

POLITECNICO DI TORINO

MASTER OF SCIENCE IN ELECTRONIC ENGINEERING

MASTER'S THESIS

Modelling of the electric field in molecular FCN structures

Author:
Luciano MACRI'

Supervisors:
Prof. Gianluca PICCININI
Prof. Mariagrazia GRAZIANO
Prof. Maurizio ZAMBONI



Contents

1	Introduction	15
1.1	Magnetic FCN	17
1.2	Semiconductor FCN	17
1.3	Metallic FCN	18
1.4	Molecular QCA and its principle	18
1.5	Bis-Ferrocene Molecule	20
1.6	The organization of clock system	21
2	Modelling of electric field for a 3D structure of electrodes for FCN system.	24
2.1	Electric field model for a finite wire	26
3	Evaluation of the linear charge distribution λ along wires	31
3.1	Capacitance between the two upper wires applying the driver mode	31
3.2	Capacitance between wires applying the clock mode	35
3.3	Combination of driver and hold mode	37
3.3.1	Logic State '1' combined with the Hold State	38
3.3.2	Logic State '0' combined with the Hold State	39
4	Verification of the finite wire model	41
4.1	Geometry	41
4.1.1	Directions and planes for which the electric field, in the driver mode, has been considered	44
4.1.2	Directions and planes for which the electric field, in the clock mode, has been considered	46
4.2	$E_x(x,y,z)$ for finite wire model in driver mode	49
4.3	$E_y(x,y,z)$ for finite wire model in driver mode	54
4.4	$E_z(x,y,z)$ for finite wire model in driver mode	55
4.5	$E_y(x,y,z)$ for finite wire model in reset mode	57
4.6	$E_y(x,y,z)$ for finite wire model in hold mode	58
4.7	$E_y(x,y,z)$ for finite wire model in hold+driver mode	61

4.7.1	Hold+driver by forcing the logic state '1'	61
4.7.2	Hold+driver by forcing the logic state '0'	62
5	Modelling of the electric field of a finite wire obtained with charges	64
5.1	Configuration number 1	65
5.2	Configuration number 2	66
5.2.1	Charge evaluation for the charge model according to the configuration 2	67
5.2.1.1	Charge evaluation for the charge model(2) in the driver mode	68
5.2.1.2	Charge evaluation for the charge model(2) in the hold mode	69
5.2.1.3	Charge evaluation for the charge model(2) in the reset mode	69
5.2.2	Verification of the charge model(2)	70
5.2.2.1	$E_x(x,y,z)$ for charge model(2) in driver mode	70
5.2.2.2	$E_y(x,y,z)$ for charge model(2) in driver mode	73
5.2.2.3	$E_y(x,y,z)$ for charge model(2) in hold mode	75
5.2.2.4	$E_y(x,y,z)$ for charge model(2) in reset mode	76
5.2.3	Final consideration about the configuration number 2	77
5.3	Configuration number 3	78
5.3.1	Charge evaluation for the charge model according to the configuration 3	80
5.3.2	Verification of the charge model(3)	81
5.3.2.1	$E_x(x,y,z)$ for charge model (3) in the driver mode	82
5.3.2.2	$E_y(x,y,z)$ for the charge model (3) in the reset mode	84
5.3.2.3	$E_y(x,y,z)$ for the charge model (3) in the hold mode	88
5.3.3	Parametric analysis for the configuration number 3	88
5.3.3.1	Parametric analysis for D_w	89
5.3.3.2	Parametric analysis for D_v	89
5.3.3.3	Parametric analysis for the applied voltage	91
5.3.4	Final consideration about the configuration number 3	92
6	Physical implementation for a Molecular QCA and relative limits	93
6.1	Considerations about different shapes for the driver mode	94
6.2	Considerations about different shapes for the clock mode	96
7	Towards Scerpa	99
7.1	Layout of a structure	99
7.1.1	Selection of the molecule under test	99
7.1.2	Definition of the structure	100
7.2	Definition of the drivers	104
7.3	Definition of the phases	105

7.4	Computation of the circuit under test	106
7.4.1	Initialization step	106
7.4.2	Molecular-interaction computation step	107
7.4.3	Final charge distribution evaluation	108
7.5	Example for SCERPA: Majority Voter	108
7.6	Electric field algorithm	112
7.6.1	Simulation of L, T and Crossroad structures	120
7.6.2	Simulation of L, T and Crossroad structures with overlapping . . .	124
7.7	Simulation of the Majority Voter	128
8	Conclusion	135

List of Figures

1.1	Evolution of the systems following the More than Moore's Law.	16
1.2	Standard configurations for a magnetic FCN cell.	17
1.3	Standard configuration for metallic FCN cell.	18
1.4	Standard unpolarized cell of molecular QCA.	19
1.5	Definition of the admitted states for a molecular QCA.	19
1.6	Elementary gates obtained through molecular QCA	20
1.7	Implementation of wires and inverter through molecular QCA	20
1.8	Bis-Ferrocene Molecule.[1]	21
1.9	Evolution of the clock signal inside a QCA system. [2]	22
1.10	Elementary gates obtained through molecular QCA by imposing a multi-phase clock system in order to guarantee the correct propagation of the information.	23
1.11	Implementation of wires and inverter through molecular QCA by imposing a multi-phase clock system in order to guarantee the correct propagation of the information.	23
2.1	Structure of electrodes for FCN system.	24
2.2	Structure of electrodes for FCN system with circular section.	25
2.3	Reference system for a finite wire according to the model [3].	26
3.1	Model used to estimate the capacitance during the driver mode.	32
3.2	Possibles logic states due to the driver: '1' or '0'.	34
3.3	Clarification of hold and reset state, highlighting the connection and the behaviour of electrons of molecules [4].	35
3.4	Combination of the Hold states with the logical value '1'.	38
3.5	Combination of the Hold states with the logical value '0'.	40
4.1	Reference system used to verify the model in the verification chapter. . . .	41
4.2	Considered line along x-direction, for y=0nm and z=0nm, in the driver mode.	44
4.3	Considered plane XY, for z=0nm in the driver mode.	45
4.4	Considered line along z-direction, for x=0nm and y=0nm, in the driver mode.	45

4.5	Considered plane XZ, for $y=0\text{nm}$ in the driver mode.	46
4.6	Considered plane XY, for $z=0\text{nm}$ in the clock mode.	47
4.7	Considered line along y-direction, for $x=0\text{nm}$ and $z=0\text{nm}$, in the clock mode.	47
4.8	Considered line along z-direction, for $x=0\text{nm}$ and $y=-3.6\text{nm}$, in the clock mode.	48
4.9	Comparison between Matlab and Cmsol about $E_x(x,y=0\text{nm},z=0\text{nm})$ by forcing the static logic '1'.	49
4.10	Comparison between Matlab (on the top) and Cmsol (on the bottom) about $E_x(x,y,z=0\text{nm})$ by forcing the static logic '1'.	50
4.11	Comparison between MatLab and Cmsol about $E_x(x,y=0\text{nm},z=0\text{nm})$ by forcing the static logic '0'.	51
4.12	Comparison between Matlab (on the top) and Cmsol (on the bottom) about $E_x(x,y,z=0\text{nm})$ by forcing the static logic '0'.	51
4.13	Comparison between Matlab and Cmsol about $E_x(x=0\text{nm},y=0\text{nm},z)$ by forcing the static logic '1'.	52
4.14	Comparison between Matlab and Cmsol about $E_x(x=0\text{nm},y=0\text{nm},z)$ by forcing the static logic '1' with length for each wire equal to 40 nm.	53
4.15	Comparison between Matlab (on the top) and Cmsol (on the bottom) about $E_x(x,y=0\text{nm},z)$	53
4.16	Comparison between Matlab (on the top) and Cmsol (on the bottom) about $E_y(x,y,z=0)$	54
4.17	Comparison between Matlab and Cmsol about $E_y(x=-0.5\text{nm},y,z=0\text{nm})$ (on the left) and $E_y(x=+0.5\text{nm},y,z=0\text{nm})$ (on the right) by forcing the static logic '1'.	55
4.18	Comparison between Matlab and Cmsol about $E_z(x=-0.5\text{nm},y=0\text{nm},z)$ (on the left) and $E_z(x=+0.5\text{nm},y=0,z)$ (on the right) by forcing the static logic '1'.	56
4.19	Comparison between Matlab and Cmsol about $E_z(x=-0.5\text{nm},y=0,z)$ (on the left) and $E_z(x=+0.5\text{nm},y=0\text{nm},z)$ (on the right) by forcing the static logic '1' with length for each wire equal to 40 nm.	56
4.20	Comparison between Cmsol (on the left) and Matlab (on the right) by imposing the reset mode.	57
4.21	Comparison between Cmsol and Matlab by imposing the reset mode considering the $E_y(x=0\text{nm},y,z=0\text{nm})$	58
4.22	Comparison between Cmsol (on the left) and Matlab (on the right) by imposing the Hold mode.	58
4.23	Comparison between Cmsol and Matlab by imposing the Hold mode considering the $E_y(x=0\text{nm},y,z=0\text{nm})$	59
4.24	Comparison between Cmsol and Matlab by imposing the Hold mode considering the $E_y(x=0\text{nm},y=-3.6\text{nm},z)$	60

4.25	Comparison between Comsol and Matlab by imposing the '1' static logic through the Hold+Driver mode considering the $E_y(x,y,z=0\text{nm})$	61
4.26	Comparison between Comsol and Matlab by imposing the '0' static logic through the Hold+Driver mode considering the $E_y(x,y,z=0\text{nm})$	62
4.27	Highlight of the field lines in the Driver+Hold mode: on the left the '1' static logic is imposed while on the right the '0'.	62
4.28	Comparison between Comsol (on the left) and Matlab (on the right) by imposing the '1' static logic with Hold+Driver mode considering the $E_y(x=0\text{nm},y,z)$	63
5.1	Reference structure from a 3D point of view.	64
5.2	Grid of charges used to obtain a lot of possibles shapes for electrodes.	65
5.3	Different organizations which have been tested in order to determine the best model following the configuration 2.	67
5.4	Example of how to evaluate the total amount of spheres which build a wire.	69
5.5	Comparison between Matlab and Comsol about $E_x(x,y=0\text{nm},z=0\text{nm})$ by forcing the static logic '1' with the charge model(2).	70
5.6	Comparison between Matlab (on the top) and Comsol (on the bottom) about $E_x(x,y,z=0\text{nm})$ by forcing the static logic '1' with the charge model(2).	71
5.7	Comparison between Matlab and Comsol about $E_x(x,y=0\text{nm},z=0\text{nm})$ by forcing the static logic '0' with the charge model(2).	72
5.8	Comparison between Matlab and Comsol about $E_x(x=0\text{nm},y=0\text{nm},z)$ by forcing the static logic '1' with the charge model(2).	73
5.9	Comparison between Matlab and Comsol about $E_y(x=-0.5\text{nm},y,z=0\text{nm})$ (on the left) and $E_y(x=+0.5\text{nm},y,z=0\text{nm})$ (on the right) by forcing the static logic '1' with the charge model(2).	73
5.10	Comparison between Matlab (on the top) and Comsol (on the bottom) about $E_y(x,y,z=0\text{nm})$ by forcing the static logic '1' with the charge model(2).	74
5.11	Comparison between Comsol (on the left) and Matlab (on the right) about $E_y(x,y,z=0\text{nm})$ by forcing hold mode with the charge model(2).	75
5.12	Comparison between Comsol and Matlab about $E_y(x,y=-3.6\text{nm},z=0\text{nm})$ by forcing hold mode with the charge model(2).	76
5.13	Comparison between Comsol (on the left) and Matlab (on the right) about $E_y(x,y,z=0\text{nm})$ by forcing reset mode with the charge model(2).	77
5.14	Configuration number 3, on the left there is a real case of how one electrode of 10 nm is modelled, while on the right there is top view of the entire 3D structure	79
5.15	Comparison between Matlab and Comsol about $E_x(x,y=0\text{nm},z=0\text{nm})$ by forcing the static logic '1' with the charge model(3).	82
5.16	Comparison between Matlab and Comsol about $E_x(x=0\text{nm},y=0\text{nm},z)$ by forcing the static logic '1' with the charge model(3).	83

5.17	Comparison between Matlab and Cmsol about $E_x(x=0nm,y=0nm,z)$ by forcing the static logic '1' with the charge model(3) for a structure with length's wire of 20 nm on the left and 30 on the right.	84
5.18	Comparison between Cmsol and Matlab about $E_y(x=0nm,y,z=0nm)$ by forcing reset mode with the charge model (3).	85
5.19	Comparison between Cmsol (on the left) and Matlab (on the right) about $E_y(x,y,z=0nm)$ by forcing reset mode with the charge model (3).	86
5.20	Comparison between Cmsol and Matlab about $E_y(x=0nm,y=-4.5nm ,z)$ by forcing reset mode with the charge model (3).	86
5.21	Comparison between Cmsol (on the left) and Matlab (on the right) about $E_y(x=0nm,y=-4.6nm ,z)$ by forcing reset mode with the charge model (3) for different lengths.	87
5.22	Comparison between Cmsol (on the left) and Matlab (on the right) about $E_y(x,y,z=0nm)$ by forcing hold mode with the charge model (3).	88
5.23	Parametric analysis for the horizontal distance between the right and left electrode.	89
5.24	Parametric analysis for the vertical distance between the upper electrodes and the clock electrode by considering an absolute point.	90
5.25	Parametric analysis for the vertical distance between the upper electrodes and the clock electrode by considering a relative point.	90
5.26	Parametric analysis for the applied voltage in the clock mode.	91
6.1	Real structure of the electrodes by considering also the support for the nano-wires.	93
6.2	Considered shapes for the driver mode due to technological limits.	94
6.3	Comparison between different profiles about $E_x(x,y=0nm,z=0nm)$ by forcing the '1' static logic with the driver mode.	95
6.4	Considered shapes for the clock mode due to technological limits.	96
6.5	Comparison between different profiles about $E_y(x=0nm,y,z=0nm)$ by imposing the reset mode.	97
6.6	Comparison of the electric field in reset mode between structure of electrodes with and without support.	97
6.7	Comparison of the electric field in reset mode between structure of electrodes with(on the left) and without(on the right) support about the field lines.	98
7.1	Wire composed from eight molecules under the same phase by using the SCERPA algorithm [5].	101
7.2	Bus obtained through the algorithm SCERPA [5].	102
7.3	Inverter obtained through the algorithm SCERPA [5].	103
7.4	Majority voter obtained through the algorithm SCERPA [5]	103

7.5	Definition of drivers through the algorithm SCERPA [5].	104
7.6	Logic states imposed from the drivers through the algorithm SCERPA [5].	105
7.7	Example of definition of phases for a generic circuit through the algorithm SCERPA.	106
7.8	Declaration of Majority voter's layout through the algorithm SCERPA [5].	109
7.9	Definition of Majority voter's time steps thanks to the declaration of phases through the algorithm SCERPA [5].	110
7.10	Evolution of Majority voter thanks to the declaration of relative phase zones through the algorithm SCERPA [5].	111
7.11	Reference axes for structure of electrodes used on algorithm SCERPA. . . .	112
7.12	Wire, composed from three structures of electrodes, used to test the electric field model on SCERPA algorithm.	113
7.13	Timing diagram for the simulation of the wire composed from three structures of electrodes.	113
7.14	Vertical electric field for each steps of the timing diagram shown in figure 7.13 by comparing, as always, Matlab and Comsol.	115
7.15	Vertical electric field, but with a 2D view, for each steps of the timing diagram shown in figure 7.13 by comparing, as always, Matlab and Comsol.	116
7.16	Simulation of the wire with a distance between electrodes' structures equal to 6 nm and a length for each of them equal to 30 nm, by imposing the time step t_1	117
7.17	Simulation of the wire with a distance between electrodes' structures equal to 10 nm and a length for each of them equal to 30 nm, by imposing the time step t_1	118
7.18	L, T and Crossroad structures respectively starting from the top.	119
7.19	Comparison between Comsol(on the left) and Matlab(on the right) for L, T and crossroad structures, respectively starting from the top, with a length for each wire equal to 30 nm.	121
7.20	Comparison between Comsol(on the left) and Matlab(on the right) for L, T and crossroad structures with a length for each wire equal to 10 nm. . .	122
7.21	Comparison between Comsol(on the left) and Matlab(on the right) for L, T and crossroad structures, respectively starting from the top, with a length for each wire equal to 10 nm but with boundary effect's optimization. . . .	123
7.22	L, T and Crossroad structures with overlapping respectively starting from the top.	125
7.23	Comparison between Comsol(on the left) and Matlab(on the right) for L, T and crossroad structures with overlapping, respectively starting from the top, with a length for each wire equal to 30 nm.	126

7.24	Comparison between Comsol(on the left) and Matlab(on the right) for L, T and crossroad structures, respectively starting from the top, with overlapping with a length for each wire equal to 10 nm but with boundary effect's optimization.	127
7.25	Qualitative scheme for the Majority Voter used just to explain how the simulation is arranged.	128
7.26	Majority voter circuit build with a crossroad structure long 5 nm for each wire. On the top, it is shown from 2D point of view while on the bottom from 3D.	130
7.27	Simulation for the majority voter with a crossroad structure long 5 nm. By starting from the top towards the bottom, the results for time steps t1, t2 and t3 are shown. Comsol's results on the left while Matlab's results on the right.	131
7.28	Majority voter circuit build with a crossroad structure long 3 nm for each wire. On the top, it is shown from 2D point of view while on the bottom from 3D.	132
7.29	Simulation for the majority voter with a crossroad structure long 3 nm. By starting from the top towards the bottom, the results for time steps t1, t2 and t3 are shown. Comsol's results on the left while Matlab's results on the right.	133

List of Tables

3.1	Comparison between the two different models for the capacitance per unit length with respect to the reference values taken from Comsol, in the column C_C/l ,(pF/m).	33
3.2	Relative error between the reference value for the capacitances obtained through Comsol and the value which will be used in the model [6].	33
3.3	Comparison between the reference value for the capacitances obtained through Comsol Simulator and the model [6] regards the capacitance C_{LC}/l in the clock mode.	36
3.4	Comparison between the reference value for the capacitances obtained through Comsol Simulator and the model [6] regards the capacitance C_{RC}/l in the clock mode.	36
3.5	Partition of electrons between dots in function of the clock field[7] independently on the driver voltage.	37
3.6	Comparison between Comsol and Matlab about the charge distribution for each wire in the Hold state with logical value '1'.	39
3.7	Comparison between Comsol and Matlab about the charge distribution for each wire in the Hold state with logical value '0'.	40
4.1	Geometrical parameters of the left electrode.	42
4.2	Geometrical parameters of the right electrode.	42
4.3	Geometrical parameters of the clock electrode.	42
4.4	Considered volume during the testing of driver mode.	43
4.5	Considered volume during the testing of clock mode.	43
5.1	Computational cost for the charge model(2) following the configuration number 2.	77
7.1	Partition of electrons between dots in function of the clock field for the Bisferrocene [5].	100
7.2	Coordinates of dots, in the 3D space, for the Bisferrocene [5].	100

7.3 Optimal potentials used during the simulation of the Majority Voter through
the different time steps [4]. 129

Abstract

The employment of technological applications in distinct fields and contexts, like for instance governmental and public, has been the force for engineers and specialist to discover new powerful devices to able to realize more complex tasks.

With this idea, to overcome the limits of the technology, an incredible improvement has been possible by optimization of the design and production of circuits. Nowadays, we can perform the scaling of the size of transistors to increase their number inside a single chip. This is known as Moore's Law, where the number of transistors duplicates every one year and a half, more or less. As you can easily imagine, sooner or later, the saturation will be reached, since it's impossible to think to scale infinitely an object. Please, pay attention, devices with a size of 7 nanometres are coming which is incredible.

Anyway, what mentioned above remains true. As a consequence, there are a lot of resources and engineers, or if you want pioneers, that are working to discover alternative technologies. One field, which is strongly under investigation and development, is the FCN, which means field coupling nano-computing, where the devices can work and propagate the information thanks to the field's interactions. The field and the relative elementary block of this new technology can be magnetic or electric.

In case of electric field, QCA (Quantum-dot Cellular Automata) FCN the information is associated not to the propagation of the charges, so current, but to the position of this latter, by having a lot of advantages from it like for instance the small dimensions of obtained devices, since molecules are the foundation of this technology and the speed of them.

Nowadays, different molecular simulators are available for simulating their properties. In the specific, inside the VLSI Lab, which is an innovation laboratory in the electronic department of the Politecnico of Turin, a circuit molecular simulator, called SCERPA, is under development. The thesis aim is to model the electric field inside particular structures of electrodes strongly used in this technology, by guaranteeing a link between the applied voltage on them and the management of the molecules and their polarisation.

As you will discover by reading this report, the target is to get an algorithm that describes precisely the electric field to fuse it inside SCERPA to increase the contact between the circuit molecular simulation and the physical world presents inside this type of devices.

Summary

Chapter 1

In the first chapter, a very short but essential introduction has been explained, to prepare the reader to this topic, by giving to him the most important needed knowledge to understand what we are talking about.

Chapter 2

In this case, about chapter 2, the finite wire model is determined by showing and demonstrating, mathematical speaking, how the equations have been got with the relative assumptions.

Chapter 3

This is one of the first most important chapters, where the link between the applied voltage and the charge per unit length, λ , is defined. In this case, considerations have been performed by evaluating all possible modes that are implemented on Matlab: the driver mode, used to impose the logic state in input, the clock mode, used to help and manage the propagation of the information along different complex circuits like wires, majority voter, bus and so no.

Chapter 4

Here, the first verification step has been done, by comparing the results got from the model implemented in Matlab and the results obtained from the numerical simulator, Comsol, where problems regard to the finite wire model have been highlighted.

Chapter 5

Probably, as you will discover, this is the core of the thesis where the charge model has been defined. Like you will see, the best results will be reached with configuration

number 3, where all problems will be solved and an incredible number of cases, with different dimensions, will be optimized with a fantastic final resolution.

Chapter 6

In this part, a fundamental check has been done between the model implemented in Matlab and the physical world, by trying to relax the assumption done in chapter 2, like the wires' section and others, that will be explained during the report.

Chapter 7

Here, a link between the algorithms regard the electric field and SCERPA has been done, to demonstrate that the thought organization is ready to be fused with the molecular simulator.

In particular, a huge amount of circuit has been tested, like the L, T and crossroad structure, and as a consequence, by combining these also the majority voter has been verified, that as you will discover, it's a very complex circuit with a tricky organization for electrodes that build it.

Chapter 8

This is the last chapter of this thesis, and like always happens, here conclusion are deduced. Be careful, as you will discover, generally for each chapter there are the relative conclusions to help the reader to keep the contact with what's happening. As a consequence, in this final chapter, a general and conclusive deductions are performed.

Chapter 1

Introduction

Starting from the end of 1950 years, the base of technology is the silicon, by using integrated circuits (IC) to implement every type of device inside a single chip. Of course, to reach this aim, a lot of years have been necessary since there were many problems due to the technological limits.

Thanks to the improvements of processes used to realize IC, engineers and specialists have been able to get smaller devices, so that, scaling has been possible to increase the number of transistors inside a single component. But, as you know, due to the reduction of size for transistors, a lot of problems appear, like leakage, heat due to power dissipation and so on.

This has been the reason for which silicon technology has moved from BULK to SOI technique, and, nowadays, up to FIN-FET devices, to reduce problems regard scaling and, of course, increase the density.

Another aspect, beyond the *Moore's Law*, that has been introduced above, is that there is also a parallel approach, which is called *More Than Moore*, by opening the world to integration for sensors, radio-frequency devices, MEMs and in the end, biochip applications.

All these considerations can be synthesized with the following figure 1.1, where it is possible to appreciate qualitatively the evolution of the systems. In the vertical axis, it is shown the scaling of the gate length concerning time while on the horizontal axis the integration of different applications.

Theoretically, sooner or later, the saturation for the scaling will be reached, and so, what's happening now is that researchers and companies are investing a lot of resources and money to find, or better, discover alternative devices that may substitute the silicon technology.

Of course, this is not an easy task, but like always happens, it is important to start and then, with hard work and determination, something always arrives.

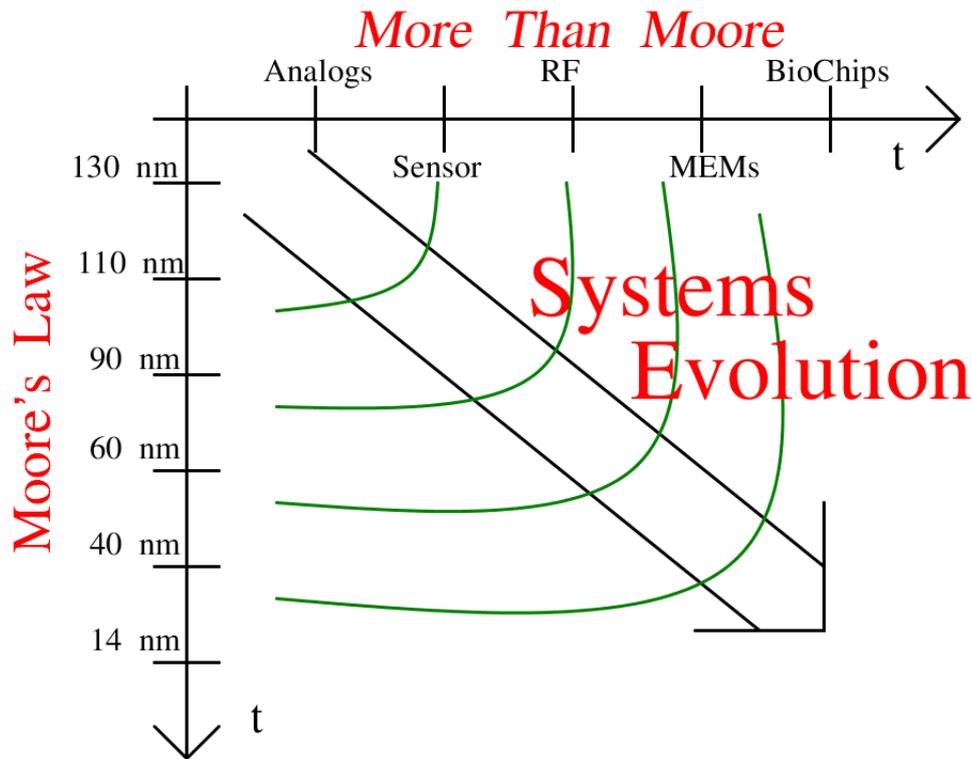


Figure 1.1: Evolution of the systems following the More than Moore's Law.

A possible choice, which is the subject of this thesis, is the *Field coupled nano-computing* technology [8], that can be abbreviated with "FCN", where the elementary blocks that build the system work through interaction's field. In base on which type of new technology is used, the information could be associated to different states, like the orientation of grains for magnetic FCN or polarisation of the molecules for Quantum-dot Cellular Automata (QCA) where the information is associated not to the current propagation but the position of charges.

Before to continue the explanation of this argument, it could be interesting to mention the other alternative new technologies which are, in parallel, under development. What is curious is that, for instead, one of these is already used to realize stable memories. So, it is something that is not just theoretical but starts to become concrete with some application.

1.1 Magnetic FCN

In this possible alternative, the standard cell is represented from nanomagnet which has just two possible stable conditions:

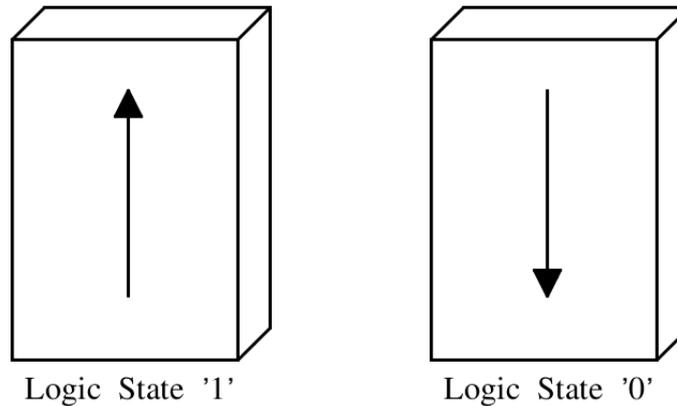


Figure 1.2: Standard configurations for a magnetic FCN cell.

By applying an external field, the basic cell follows the imposed orientation and all grains, that build the cell, are aligned along the same direction as it is shown qualitatively in the figure 1.2. Thanks to it, we can get a device with very low power consumption, no leakage, intrinsically a memory, since at the equilibrium, it maintains the information until a strong perturbation appears, feasible and, in the end, can be integrated with standard technology. The draw-back is that in a standard configuration it is not a fast device and, to guarantee a stable technology, it cannot be scaled a lot. By combining different standard cells, many configurations and basic gates, like AND, OR and wire can be realized. As a consequence, complex devices can be built by using these elementary gates.

This is a concrete alternative, in fact, Magnetic FCN is already underdevelopment to implement magnetic memories.

1.2 Semiconductor FCN

In this case, an alternative semiconductor solution may be considered based on heterostructure composed by different semiconductor compounds by trying to emulate the molecular QCA principle. Unfortunately, due to technological problems, it results so much complex to guarantee a correct functioning with nano-size dimensions.

1.3 Metallic FCN

This is the last alternative that will be mentioned. Here, the basic cell is obtained by combining island and metallic wires, as it is possible to appreciate in the figure 1.3. What happens is that in the islands, electron population is quantized, more or less, like in a standard QCA but unfortunately the dimension are so much big to work.

From a practical point of view, this technology is not a real possible alternative and, nowadays, is still a prototype to study.

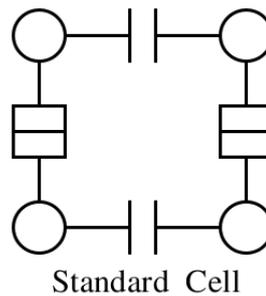


Figure 1.3: Standard configuration for metallic FCN cell.

1.4 Molecular QCA and its principle

Also in the Molecular QCA system, like it has been mentioned above, the information inside a single cell is associated with the charge position. The cell is based on oxidized molecules, where charges are aggregated in two different redox center quantum dots, and so, electrons can move from one to the other but always inside the molecule. In the following figure 1.4, it is possible to see the standard cell not polarized, which consists of two near molecules, where for each molecule it is possible to define 3 dots that work as a figure of merit to simplify the model.

The figure 1.4 has not a physical meaning, but it is used just to describe the system, since the electrons inside the cell evolve, of course, in the state with minimal energy, and since, as you know, between electrons there is a repulsive Coulomb's Force, the only two possible stable states are those where charges are aligned along the two diagonals where the distance is the maximum admitted and so the minimal energy is reached. By wanting to control the cells, another unstable state is defined like the NULL state, that can be reached by applying a determined signal to force all electrons to go inside the central dots[9].

Everything can be clarified by looking at the figure 1.5 where all admitted states are proposed.

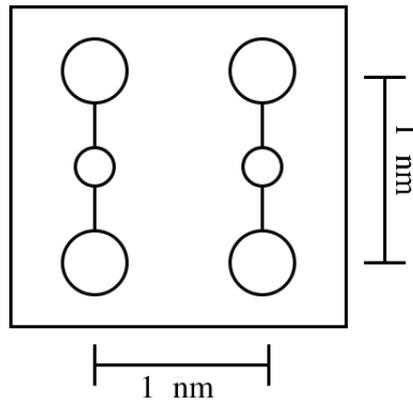


Figure 1.4: Standard unpolarized cell of molecular QCA.

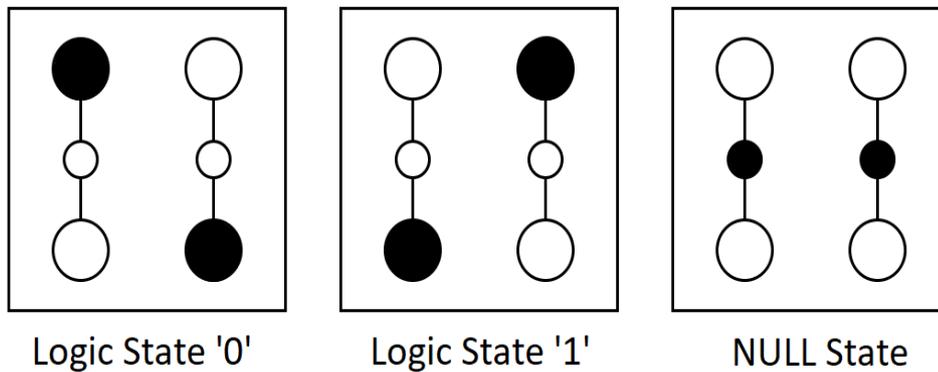


Figure 1.5: Definition of the admitted states for a molecular QCA.

Using the standard cell, elementary gates can be realized to build the simplest tools to implement more complex devices, like half-adder, full-adder and so no.

What is clear, by looking at the figure 1.6, is that the most important gate is the majority voter, since the AND, OR gates are obtained from it by forcing one input to a specific value. For the AND it is necessary to force one input to the logic state '0' while for the OR to '1'.

Then, like it is possible to appreciate in the figure 1.7, also the inverter gate and wires are realised to complete the set of elementary gates and to propagate the information.

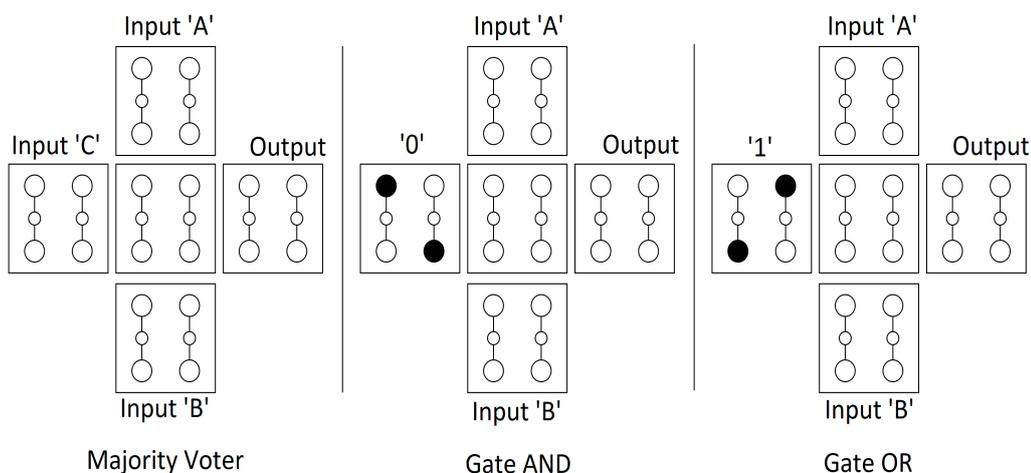


Figure 1.6: Elementary gates obtained through molecular QCA

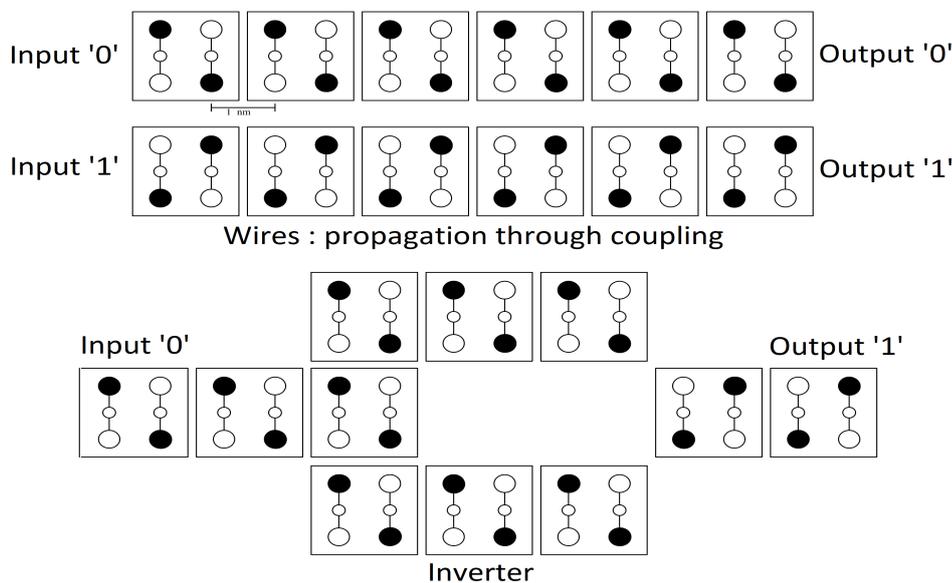


Figure 1.7: Implementation of wires and inverter through molecular QCA

1.5 Bis-Ferrocene Molecule

Up to now, nothing has been mentioned about which molecules can be used to realise these type of circuits.

A plausible candidate is the Bis-Ferrocene molecule, which has been strongly analysed from a lot of contest and brilliant scientists. [10] [11] [12] [13].

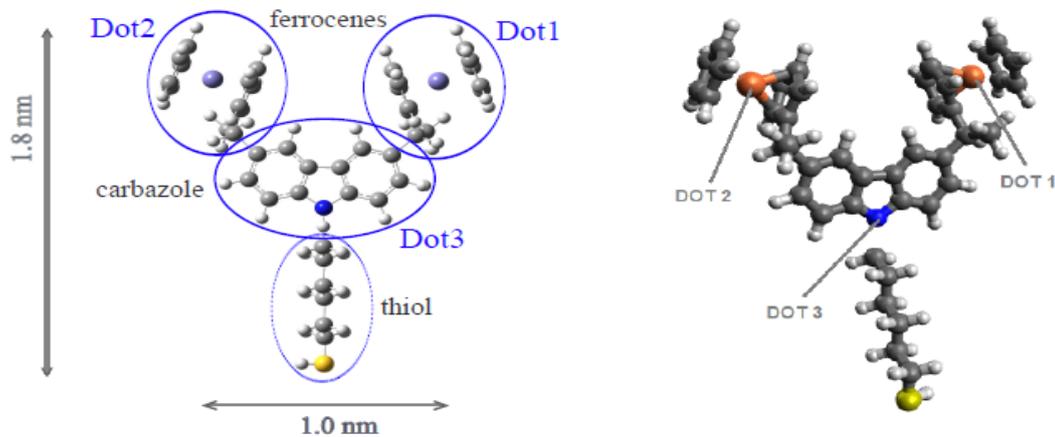


Figure 1.8: Bis-Ferrocene Molecule.[1]

By looking at the figure 1.8, on the left, there is a representation of the molecule and on the right, a screenshot of the molecule realised through Avogadro [1]. A very interesting thing has to be highlighted: the dimensions. So this is the demonstration for which a lot of resources are employed to develop this technology.

Finally, it is possible to appreciate the different dots, that have been mentioned a lot of times, which are, by continuing to look at the figure 1.8, aggregated charges. This method is used to simplify the model and have better control of the system from a theoretical point of view. To complete the explanation, there is also another part, the thiol which is the part that links the molecules and the physical structure. Since the thiol has a sort of mechanical task, it is not included in the model.

After this short but necessary introduction, we can start the journey inside the topic which has been analysed in this thesis.

1.6 The organization of clock system

The first thing which must be clarified is that, in the molecular QCA technology, the meaning of the clock is different from the standard digital world, since it mustn't be interpreted as a timing signal but like a vertical field used to drive the electrons between upper dots, dot1 and dot2, and the lowest dot3.

It becomes necessary to use a field like that since, as it has been demonstrated [14], there is a maximum number of cells that can work properly, by propagating correctly the information inside the structure, without a control signal like the clock. To simplify the explanation, it's enough to think about wires where input is imposed, like in the example shown in the figure 1.7. Without a clock, nobody can guarantee that the information

propagates only from the left, where there is the input, towards the right, where there is the output since the molecules are strongly unstable. Due to noise, in this case, is impossible to determine which is the final polarization that they reached.

By wanting to overcome this problem, to get a controllable system, a switching model is used, which is known as Adiabatic Switching [15][16][17][18].

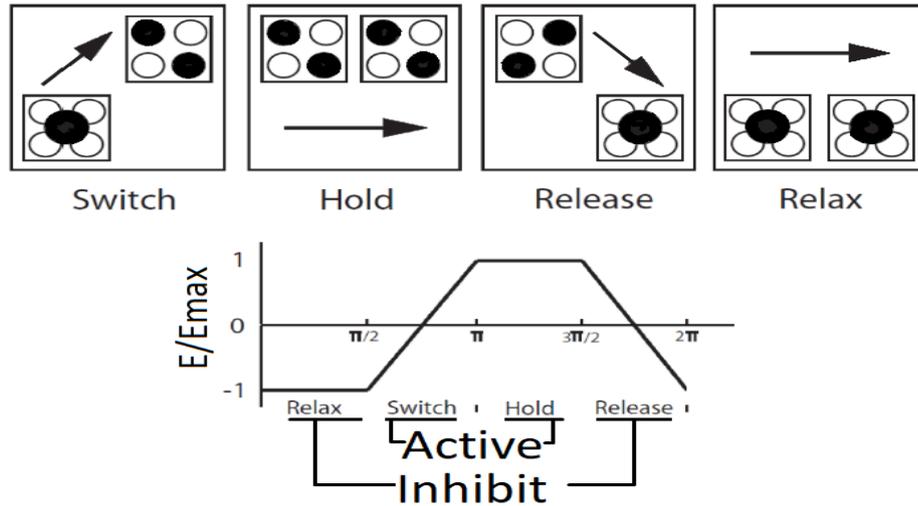


Figure 1.9: Evolution of the clock signal inside a QCA system. [2]

As a consequence, the system will be implemented like a multi-phase clock system where electrons will be driven between dots by imposing the Active state, which is the sum of Switch plus Hold state, and the Inhibit state, that is the combination of the Relax and Release State. Of course, electrons move between dots through the tunnelling effect. All these considerations are evidenced in the figure 1.9 shown above .

From a controlling point of view, we are able just to impose the steady-states and not the transitions states. So, to synthesize the clock field is used to modulate the barriers, to allow electrons to move between dots.

Before to continue, it is important to clarify one fundamental aspect: multi-phase means that in the circuit, there are different zones with distinct clock field which operate with different timing. So, the number of phases that are present in the circuit is determined from how much zones with different clock are implemented and there is no correlation with the plot shown in the figure 1.9. By wanting to apply this method, all previous gates have been re-analysed by determining the clock organization, by highlighting the region under control from the same clock zone. Of course, like it is intuitive, the cells with the same colour are managed from the identical phase.

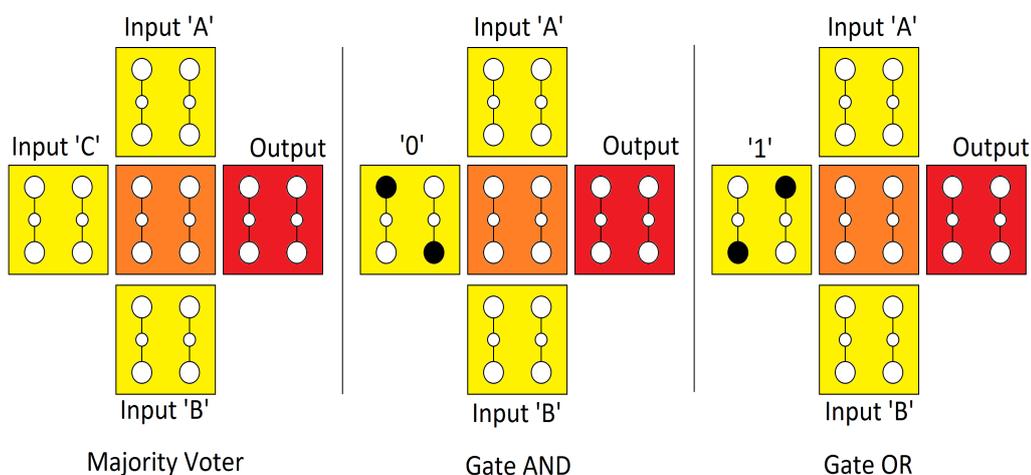


Figure 1.10: Elementary gates obtained through molecular QCA by imposing a multi-phase clock system in order to guarantee the correct propagation of the information.

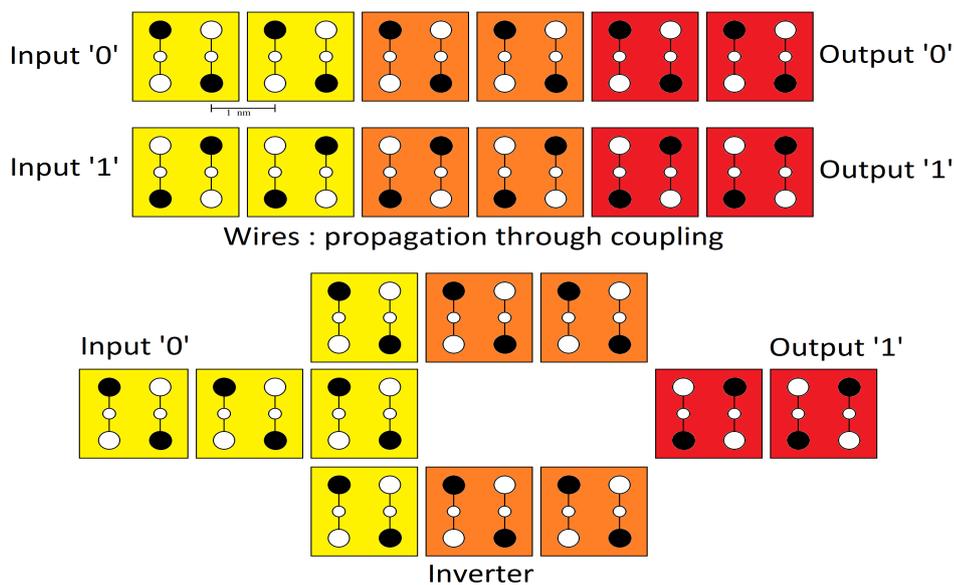


Figure 1.11: Implementation of wires and inverter through molecular QCA by imposing a multi-phase clock system in order to guarantee the correct propagation of the information.

Chapter 2

Modelling of electric field for a 3D structure of electrodes for FCN system.

The typical structure of electrodes which is used inside a molecular system can be modelled like three wires, for which by applying the right signal, it's possible to impose the different working states: Hold, Reset and Driver Mode. As a matter of fact, it could be interesting to show the structure to clarify what we are talking about.

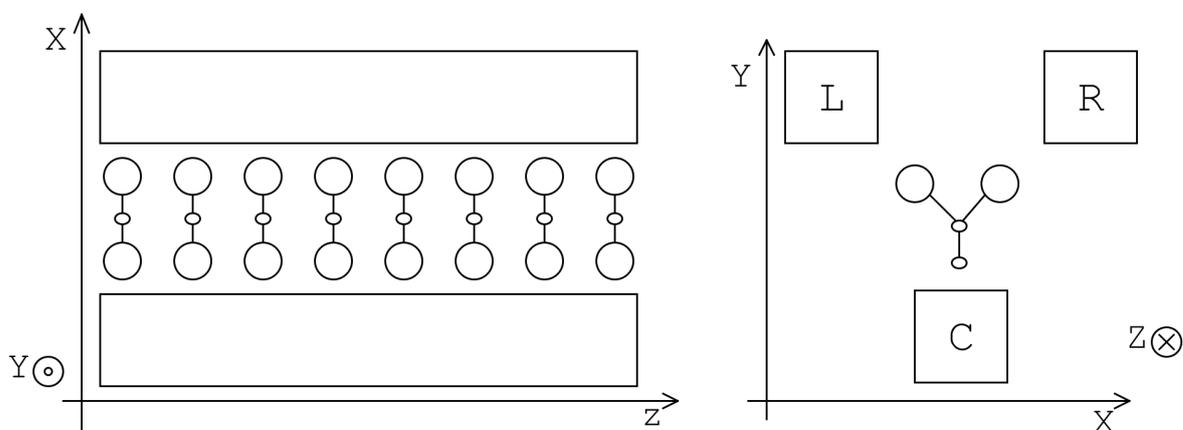


Figure 2.1: Structure of electrodes for FCN system.

Let's give some short but important names to the electrodes in order to simplify the explanation, so, by looking at the figure 2.1, and in the particular on the right, the letter L means Left Electrode, R is for Right and, in the end, C means Clock since, as it has been anticipated in the chapter dedicated to the introduction, to realise the hold and reset states, it's necessary to apply a vertical field with respect to the molecule.

This part aim is to determine a model which gives outgoing the electric field and the potential inside a specific volume where the integrated circuit is implemented.

From a practical point of view, it is very complex to describe in a closed-form the behaviour of the electric field near to the edge, so, the first approximation that can be assumed is to consider wires with circular section.

Anyway, this is not far away from the real case, since, as it's known, to realise an integrated circuit, the lithography process is exploited. Due to its limited resolution, by combining the resolution about the masks, the photoresist and in the end, problems regard to diffraction, it is practically impossible to get perfect edge. So this allows to consider reasonable the assumption. As a consequence, the structure of electrodes which is used and modelled is shown in the following figure 2.2.

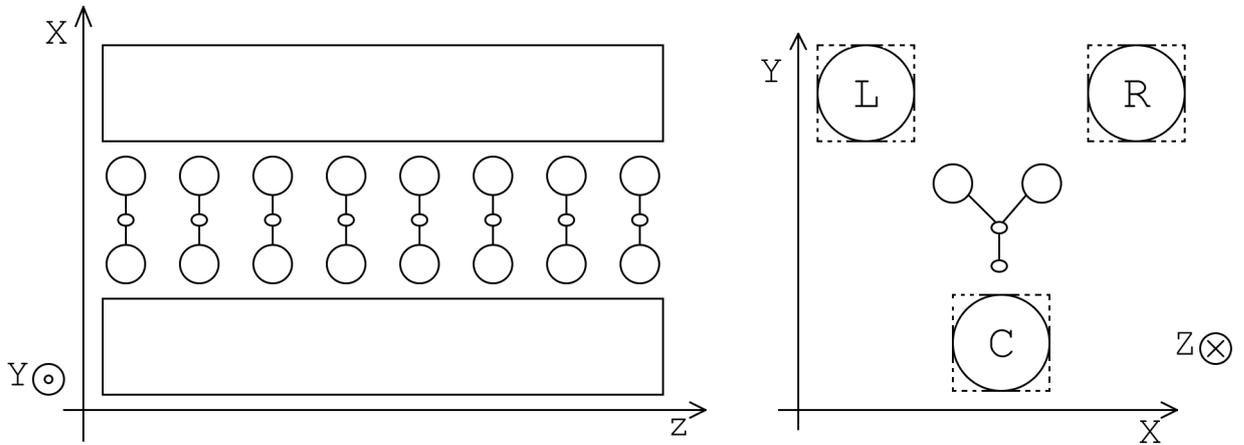


Figure 2.2: Structure of electrodes for FCN system with circular section.

By modelling a structure of electrodes, to determine the entire electric field, it's practicable to use the superposition principle. It consists in evaluating the field for each wire independently from others and then, sum each term together, one for each electrode. As a matter of fact, it's possible to summarise like follows:

$$E_X(x, y, z) = E_{X,L}(x, y, z) + E_{X,R}(x, y, z) + E_{X,C}(x, y, z) \quad (2.1)$$

$$E_Y(x, y, z) = E_{Y,L}(x, y, z) + E_{Y,R}(x, y, z) + E_{Y,C}(x, y, z) \quad (2.2)$$

$$E_Z(x, y, z) = E_{Z,L}(x, y, z) + E_{Z,R}(x, y, z) + E_{Z,C}(x, y, z) \quad (2.3)$$

2.1 Electric field model for a finite wire

Gauss's Law [3] is used as a powerful tool to solve the problem where a cylindrical or spheric symmetry is present. By considering an infinite wire, with a uniform linear charge distribution λ , the expression of the electric field is [19]:

$$\mathbf{E}(r) = \frac{\lambda}{2\pi\epsilon_0} \frac{1}{r} \hat{\mathbf{r}} \quad (2.4)$$

where, obviously, ϵ_0 is the dielectric constant in the empty space, r is the distance from the point $P(r,z)$, where the electric field has to be evaluated, and the position of the wire, and, in the end, $\hat{\mathbf{r}}$ is the unitary vector along the direction r , defined analogously like the radial component in a system of cylindrical coordinates.

The equation 2.4, of course, it is true only from a theoretical point of view. As a matter of fact, in the real case, an infinite wire doesn't exist. As a consequence, the equation 2.4 cannot be used inside this model.

For this reason, results necessary to derive the electric field for the situation which is described in the figure 2.3.

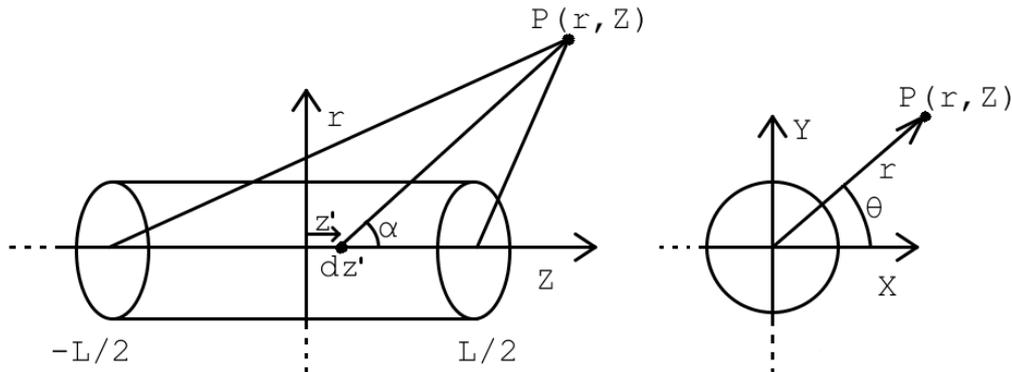


Figure 2.3: Reference system for a finite wire according to the model [3].

In this case, r is again the radial component, and so, to determine the electric field along the x and y axes, it's necessary to project the radial field with the mathematical functions: sine and cosine.

While the z coordinate is the longitudinal coordinate which crosses the wire along its length L .

By applying the superposition principle, it's possible to define the electric field in the point $P(r,z)$ like the sum of all infinite contributes $d\mathbf{E}$, given from the small elements of

the wire dz' , on which there is a charge $dq' = \lambda dz'$. As a consequence, for each element $d\mathbf{E}$ follows:

$$d\mathbf{E} = dE_z \hat{\mathbf{z}} + dE_r \hat{\mathbf{r}} = dE \cos(\alpha) \hat{\mathbf{z}} + dE \sin(\alpha) \hat{\mathbf{r}} \quad (2.5)$$

where

$$dE = \frac{k\lambda dz'}{\sqrt{(z-z')^2 + r^2}} \quad (2.6)$$

with $k = 1/(4\pi\epsilon_0)$. Therefore, the equation 2.5, can be written in the following way:

$$E_z(r, z) = k\lambda \int_{-L/2}^{L/2} \frac{z-z'}{[(z-z')^2 + r^2]^{3/2}} dz' \quad (2.7)$$

$$E_r(r, z) = k\lambda r \int_{-L/2}^{L/2} \frac{1}{[(z-z')^2 + r^2]^{3/2}} dz' \quad (2.8)$$

Now, it becomes just a mathematical exercise, since the aim is to solve these two integrals to get the electric field in all positions. By going on, it could be useful to substitute in the equation 2.7 the new variable:

$$\mu = (z-z')^2 + r^2; \quad \mu_{\pm} = (z \mp \frac{L}{2})^2 + r^2 \quad (2.9)$$

So, by going on:

$$E_z(r, z) = \frac{k\lambda}{2} \int_{\mu_+}^{\mu_-} \frac{1}{\mu^{3/2}} d\mu = k\lambda \left(\frac{1}{\sqrt{\mu_+}} - \frac{1}{\sqrt{\mu_-}} \right) \quad (2.10)$$

In the end, by substituting the expression of μ_+ and μ_- , the final expression is:

$$E_z(r, z) = k\lambda \left(\frac{1}{\sqrt{(z-L/2)^2 + r^2}} - \frac{1}{\sqrt{(z+L/2)^2 + r^2}} \right) \quad (2.11)$$

The equation 2.11 is valid also on the z axis, except for the points $z_{\pm} = \pm \frac{L}{2}$, where the electric field $E_z(r, z)$ diverges.

By proceeding with the calculation, it will be shown that, even if $E_z(r, z)$ exists also inside the charge distribution along the wire, it's impossible to define the electric field \mathbf{E} since the component $E_r(r, z)$, in the interval $z \in [-L/2, L/2]$ and $r = 0$, presents a discontinuity of the second type in every point where the linear electric charge is present.

Now, since the component $E_z(r, z)$ has been evaluated and determined, it's time to go on with the equation 2.8, in order to get $E_r(r, z)$.

By doing reference to the figure 2.3, by looking at the point on the z axis, the following substitution can be imposed: $(z - z') = r \cot \alpha$. Then, it's possible to notice that:

$$(z - z')^2 + r^2 = \frac{r^2}{\sin^2(\alpha)} ; dz' = \frac{r}{\sin^2(\alpha)} \quad (2.12)$$

As a consequence, the integral 2.8 becomes:

$$E_r(r, z) = \frac{k\lambda}{r} \int_{\alpha_-}^{\alpha_+} \sin \alpha \, d\alpha = \frac{k\lambda}{r} (\cos(\alpha_-) - \cos(\alpha_+)) \quad (2.13)$$

where

$$\cos \alpha_{\pm} = \frac{z \mp L/2}{\sqrt{(z \mp L/2)^2 + r^2}} \quad (2.14)$$

Finally, after a lot of calculations, the last component of the electric field along the radial direction is:

$$E_r(r, z) = \frac{k\lambda}{r} \left(\frac{z + L/2}{\sqrt{(z + L/2)^2 + r^2}} - \frac{z - L/2}{\sqrt{(z - L/2)^2 + r^2}} \right) \quad (2.15)$$

At this point, the electric field has been obtained on the plane r - z , except for the points which appertain to the charge distribution. Anyway, the structure of electrodes which is shown in the figure 2.2, is based on a 3D coordinate system. So, the radial component has to be projected on the two axes x and y . By looking at the figure 2.3, and in the specific on the right, it is clear that:

$$E_x(x, y, z) = E_r(r, z) \cdot \cos(\theta); \quad (2.16)$$

$$E_y(x, y, z) = E_r(r, z) \cdot \sin(\theta); \quad (2.17)$$

Now, by knowing the electric field in the space (x, y, z) for each wire, with the superposition principle, as it has been anticipated before, it is possible to know \mathbf{E} by using the formulas 2.1, 2.2 and 2.3.

Below it is shown the code of the core function used to evaluate the electric field in every point of the volume around the structure of electrodes for each wire.

```

function [Er,Ex,Ey,Ez] = Electric_field_3D (Py,Px,Pz,L,radius ,xc ,yc ,
      chle)
%The outputs of this function are the components of the electric
%field in all possible directions
%Er(x,y,z)
%Ex(x,y,z)
%Ey(x,y,z)
%Ez(x,y,z)

%V=(Px,Py,Pz) is the volume where the electric field is evaluated
%C=(xc,yc) is the center of the wire
%chle is the charge distribution per unit length
%L is the length of the wire

eps0=8.854*1e-12;
ke=1/(4*pi*eps0);

%preallocate step used in order to speed up the algorithm
Er=zeros (length (Px) ,length (Py) ,length (Pz));
Ex=zeros (length (Px) ,length (Py) ,length (Pz));
Ey=zeros (length (Px) ,length (Py) ,length (Pz));
Ez=zeros (length (Px) ,length (Py) ,length (Pz));

%the external loop, in function of k, corresponds to the z axis
for k=1:length (Pz)

    %the middle loop, in function of j, corresponds to the y axis
    for j=1:length (Py)

        %the inner loop, in function of i, corresponds to the x axis
        for i=1:length (Px)

            r=sqrt ( (Px(i)-xc).^2 + (Py(j)-yc).^2 );
            Dn=sqrt( (Pz(k)-L/2).^2 + r.^2 );
            Dp=sqrt( (Pz(k)+L/2).^2 + r.^2 );

            if (sqrt(((Px(i)-xc).^2 + (Py(j)-yc).^2)) > radius)
                Er(i,j,k)=(chle*ke./r).*((Pz(k)+L/2)./Dp)-((Pz(k)-L
                    /2)./Dn);
                Ez(i,j,k)=(chle*ke).*((1./Dn - 1./Dp));
            else

```

```
%           Er(i,j,k)=0;
%           Ez(i,j,k)=0;
           end
%notice that the else is commented since it's not necessary
%thanks to the preallocation where all vector are initialized
           Ex(i,j,k)=Er(i,j,k)*cos(atan2((Py(j)-yc),(Px(i)-xc)));
           Ey(i,j,k)=Er(i,j,k)*sin(atan2((Py(j)-yc),(Px(i)-xc)));
           end
           end
end
end %last end close the function
```

So, like it is possible to understand by looking at the function which is shown above, it is like the wire is divided into slices and the electric field is evaluated since vectors are used.

As a matter of fact, for each component of the electric field, there is a 3D matrix, where for each position, inside of the considered volume around the structure, the relative value is computed.

Chapter 3

Evaluation of the linear charge distribution λ along wires

Up to this point, nothing has been told about λ even if it has been mentioned a lot of times in the formulas shown above.

Since the aim is to describe the electric field for an integrated circuit based on molecular technology, becomes very important to link the charge along the wire, on which depends the electric field, with the applied potential on the structure.

A possible solution is to consider wires like parallel cylindrical capacitors, to take advantage of the following basic expressions:

$$Q = C \cdot V ; \lambda = Q/L = \frac{C}{L} \cdot V$$

3.1 Capacitance between the two upper wires applying the driver mode

During the driver mode, in first approximation, the only two electrodes which are implied are the two on the top, so the electrodes L and R.

The system that we are talking about is shown in the figure 3.1.

By giving some definition, before to continue with the explanation, it's possible to find again in the figure 3.1:

- r , the radius of the wires;
- D_w , the distance between electrodes, on which depends strongly the capacitance;
- l is the length of the wire;

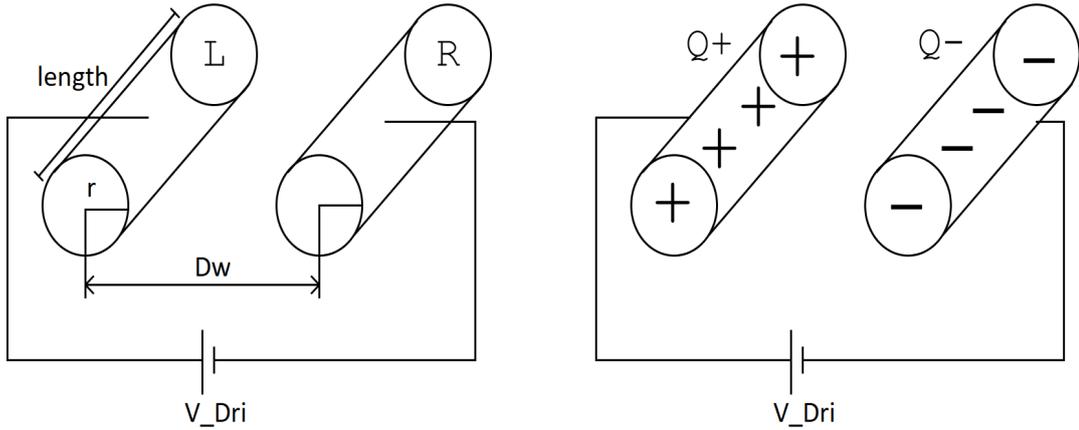


Figure 3.1: Model used to estimate the capacitance during the driver mode.

- V_{Dri} , the applied voltage used to drive the charge along wires during the driver mode.

After a lot of researches and calculations, it's possible to take into account two models from different sources, to establish which is the best and more accurate. As will be shown, some problems have been discovered due to the tiny size of the structure of electrodes. There is nothing to be surprised, since as it's known, model works just in some safe condition with specific assumptions, while in the other cases, numerical solutions are employed. The first model that could be implemented in the algorithm for the evaluation of the capacitance per unit length between a couple of wire is [20]:

$$\frac{C_1}{l} = \frac{4\pi\epsilon}{2(\ln(\frac{l}{Dw}) - 0.307) + \ln(\frac{Dw}{r})} \quad \text{when } l \gg Dw \quad (3.1)$$

Then, the second model which could be implement is [6]:

$$\frac{C_2}{l} = \frac{\pi\epsilon}{\cosh^{-1}(Dw/2r)} \quad \text{when } l \gg Dw \quad (3.2)$$

By wanting to determine which is the most accurate model, a comparison table is realised to summarise all data and simplify the discussion.

The method, which has been used, consists in keeping constant a specific distance and then increase the length to respect the assumption for which $l \gg Dw$.

What we can understand, by looking at the table 3.1, is that the model describes through the formula 3.2 is much more accurate than the other.

In the first four simulations, biggest values for r , Dw and l parameters have been considered just to verify the model with easier dimensions, then by keeping the ratio constant, and so by performing the scaling equal for each parameter, it's possible to appreciate that

Simulation	r	dw	l	$C_C/l,(\text{pF}/\text{m})$	$C_1/l,(\text{pF}/\text{m})$	$C_2/l,(\text{pF}/\text{m})$
#1	1 mm	4 mm	10 mm	27.77	42.21	21.12
#2	1 mm	4 mm	20 mm	25.1	27.8	21.12
#3	1 mm	4 mm	100 mm	22.96	15.42	21.12
#4	1mm	4 mm	200 mm	22.7	12.94	21.12
#5	1 nm	4 nm	10 nm	27.77	42.21	21.12
#6	1 nm	4 nm	20 nm	25.1	27.8	21.12
#7	1 nm	4 nm	100 nm	22.96	15.42	21.12
#8	1 nm	4 nm	200 nm	22.7	12.94	21.12

Table 3.1: Comparison between the two different models for the capacitance per unit length with respect to the reference values taken from Comsol, in the column $C_C/l,(\text{pF}/\text{m})$.

the values of the capacitance, from each source, don't change. This is true by comparing the simulations from 1 to 4 with that from 5 to 8.

So the stablest equation, which will be taken into account in the model, is the second C_2/l got from the equation (3.2). As a consequence, from this point up to the end, just this capacitance will be considered.

Unfortunately, by considering the last four simulations highlighted in the table 3.2, it's possible to see that the two values converge just when $l \gg Dw$. As a matter of fact, it's very interesting to highlight the relative error between the reference values got from Comsol and the values which are obtained from the equation 3.2.

Simulation	r	dw	l	$C_C/l,(\text{pF}/\text{m})$	$C_1/l,(\text{pF}/\text{m})$	$C_2/l,(\text{pF}/\text{m})$
#5	1 nm	4 nm	10 nm	27.77	21.12	23.95%
#6	1 nm	4 nm	20 nm	25.1	21.12	15.86
#7	1 nm	4 nm	100 nm	22.96	21.12	8%
#8	1 nm	4 nm	200 nm	22.7	21.12	6.96%

Table 3.2: Relative error between the reference value for the capacitances obtain through Comsol and the value which will be used in the model [6].

The typical dimension, that our structure of electrodes has, are shown in the simulation number 5, where the error is not negligible. Anyway, there is no solution, since, with those dimensions, the only way to get a precise value is a numeric solution, which is completely in contradiction with the aim of this thesis. Others solutions will be considered in the future, like realises wires through a combination of aligned charges to overcome the assumption of the model and, as a consequence, its limitations.

The two possible configurations are shown in the figure 3.2:

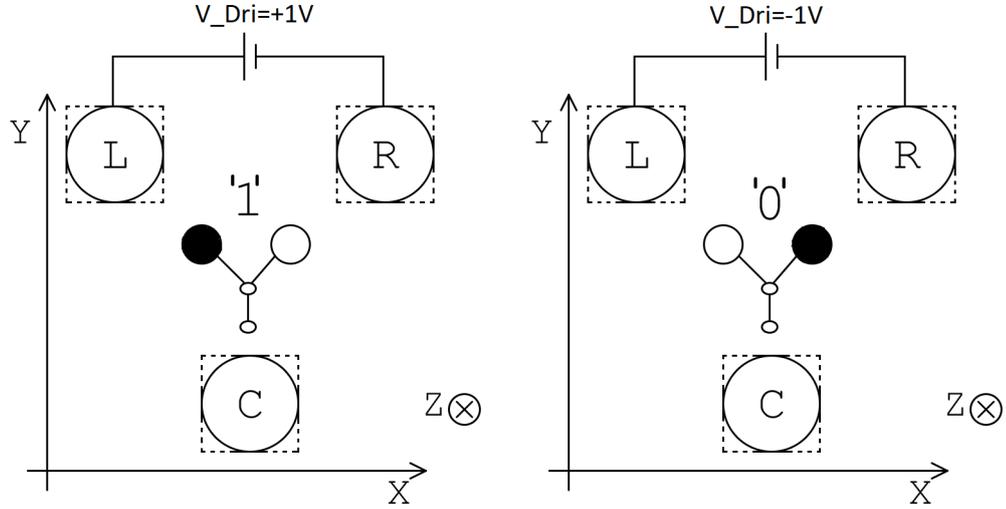


Figure 3.2: Possibles logic states due to the driver: '1' or '0'.

By doing reference to the figure 3.2, it's possible to determine which are the charge distributions for each wire in the two different cases:

- **Logic Value '0'**: by looking on the right, in this case a negative voltage is applied, as a consequence what happens is:
 - * $\lambda_L = -\frac{C}{L} \cdot |V_{Dri}|$;
 - * $\lambda_R = +\frac{C}{L} \cdot |V_{Dri}|$;
- **Logic Value '1'**: while, in this case, by looking on the left, a positive voltage is applied, as a consequence what happens is:
 - * $\lambda_L = +\frac{C}{L} \cdot |V_{Dri}|$;
 - * $\lambda_R = -\frac{C}{L} \cdot |V_{Dri}|$.

While, in this case, for the electrode C dedicated to the clock field, it's assumed to be connected to High Impedance, to guarantee that it doesn't comport unwanted capacitance since, with a zero potential, a capacitance effect appears with the electrode where the potential is not zero. The High Impedance state avoids that this happens.

At this point, everything has been considered for the driver mode and it's time to go on whit the clock mode.

3.2 Capacitance between wires applying the clock mode

The first step, that could help the explanation, is shown the two possible connections between electrodes to realise the hold state, where the electrons are pushed towards the dot1 and dot2 and the reset state, which corresponds to have all electrons of molecules inside the dot3.

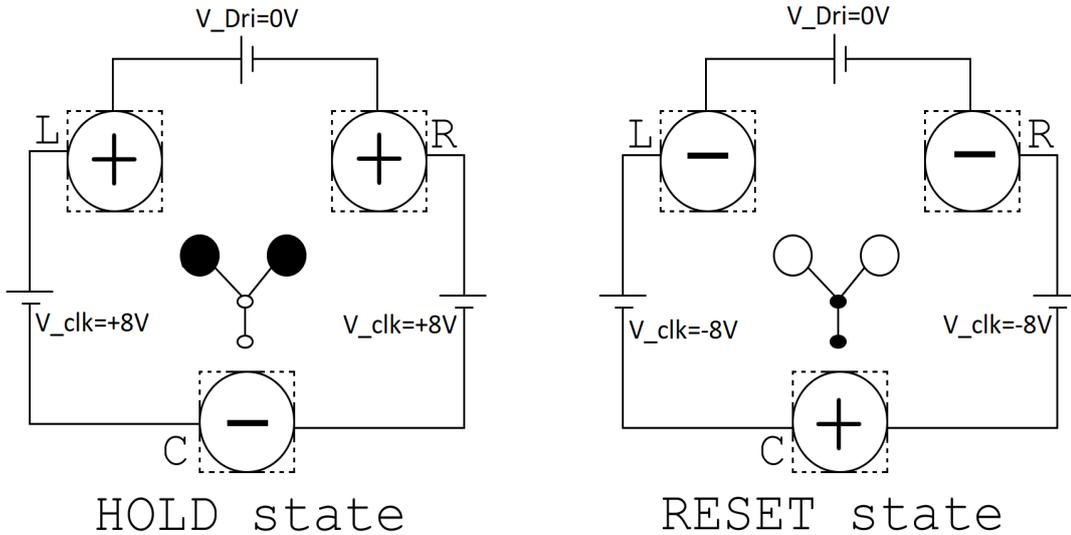


Figure 3.3: Clarification of hold and reset state, highlighting the connection and the behaviour of electrons of molecules [4].

In this case, more or less, nothing changes concerning the driver mode about the capacitances, since it's still true the equation 3.2. The only difference is that, in the first approximation, three distinct capacitances are present for both two states: hold and reset. So, to summarise, what it's able to define is:

- C_{LC} , which is the capacitance between left and clock electrodes;
- C_{RC} , which is the capacitance between right and clock electrodes;
- C_{LR} , which is the capacitance between left and right electrodes.

Since, like it's shown in the figure 3.3, the R and L electrodes are connected to the same potential, C_{LR} gives no impact and can be neglected.

As a consequence, it's necessary just to focus on the other two capacitances, but, it's sufficient to use the equation 3.2 imposing that the distance is the diagonal between centres' positions P of wires, where coordinates for each of them are equal to:

- $P_R = (X_R, Y_R, Z_R=0)$;
- $P_L = (X_L, Y_L, Z_L=0)$;
- $P_C = (X_C, Y_C, Z_C=0)$;

So the relative capacitances per unit length can be evaluate like follows:

$$D_{wL,C} = \sqrt{(X_L - X_C)^2 + (Y_L - Y_C)^2} \Rightarrow \frac{C_{LC}}{l} = \frac{\pi\epsilon}{\cosh^{-1}(D_{wL,C}/2r)} \quad (3.3)$$

$$D_{wR,C} = \sqrt{(X_R - X_C)^2 + (Y_R - Y_C)^2} \Rightarrow \frac{C_{RC}}{l} = \frac{\pi\epsilon}{\cosh^{-1}(D_{wR,C}/2r)} \quad (3.4)$$

Also, in this case, simulations have been done to compare which are the differences between the capacitance obtained from the numerical simulator Comsol and the model implemented on Matlab, to examined possible errors.

Simulation	r	dw	l	$(C_{LC}/l)_{COMSOL}$	$(C_{LC}/l)_{MATLAB}$	%Error
#09	1 nm	4 nm	10 nm	16.39 (pF/m)	14.57 (pF/m)	11.10%
#10	1 nm	4 nm	20 nm	15.05 (pF/m)	14.57 (pF/m)	3.20%
#11	1 nm	4 nm	100 nm	13.52 (pF/m)	14.57 (pF/m)	7.77%
#12	1 nm	4 nm	200 nm	13.34 (pF/m)	14.57 (pF/m)	9.22%

Table 3.3: Comparison between the reference value for the capacitances obtained through Comsol Simulator and the model [6] regards the capacitance C_{LC}/l in the clock mode.

The same has been done for the capacitance C_{RC}/l , obtaining the following results:

Simulation	r	dw	l	$(C_{RC}/l)_{COMSOL}$	$(C_{RC}/l)_{MATLAB}$	%Error
#13	1 nm	4 nm	10 nm	16.39 (pF/m)	14.57 (pF/m)	11.10%
#14	1 nm	4 nm	20 nm	15.05 (pF/m)	14.57 (pF/m)	3.20%
#15	1 nm	4 nm	100 nm	13.52 (pF/m)	14.57 (pF/m)	7.77%
#16	1 nm	4 nm	200 nm	13.34 (pF/m)	14.57 (pF/m)	9.22%

Table 3.4: Comparison between the reference value for the capacitances obtained through Comsol Simulator and the model [6] regards the capacitance C_{RC}/l in the clock mode.

Obviously, the two tables 3.3 and 3.4 are identical since capacitances depends only on geometrical parameters, and since the structure of electrodes is symmetric concerning the C electrode, the two tables are equal.

The last thing to analyse is the charge distribution along wires for the hold and reset states, so by considering the figure 3.3, results clear that:

- **Hold State:** in this case, the value of the voltage generator is equal to +8V [4], so, on the left and right electrodes, there are positive charges while on the clock wire there is negative linear charge distribution. As a consequence:

$$\begin{aligned}
 * \lambda_L &= +\frac{C_{LC}}{L} \cdot |V_{Clk}|; \\
 * \lambda_R &= +\frac{C_{RC}}{L} \cdot |V_{Clk}|; \\
 * \lambda_C &= -(|\lambda_L| + |\lambda_R|).
 \end{aligned}$$

- **Reset State:** in this case, the value of the voltage generator is equal to -8V [4], so, on the left and right electrodes, there are negative charges while on the clock wire there is positive linear charge distribution. As a consequence:

$$\begin{aligned}
 * \lambda_L &= -\frac{C_{LC}}{L} \cdot |V_{Clk}|; \\
 * \lambda_R &= -\frac{C_{RC}}{L} \cdot |V_{Clk}|; \\
 * \lambda_C &= +(|\lambda_L| + |\lambda_R|).
 \end{aligned}$$

3.3 Combination of driver and hold mode

What has been explained in section 3.1 is true but it's not the entire story.

The aim of the driver mode consists in polarising the electrons in one of the two dots (dot1 and dot2) dedicated to logic states, which are, of course, '0' or '1'. But it's also true that to have all electrons in the highest dots, a vertical field, which pushes the electrons towards up, must be applied. Consequently, the best action to get the right logic state consists in combining the hold plus driver mode.

By wanting to determine the linear charge distribution for each electrode it's necessary to use again the superposition principle but before to go on, it could be useful to look at the table 3.5, where it's able to appreciate the partition of electrons between dots in function of the vertical electric field obtained in the clock mode, independently on the driver voltage value, which is completely in agreement with the figure 3.3.

Simulation	V_{clk}	E_{clk}	DOT1	DOT2	DOT3
Hold	+8V	+2 V/nm	+0.470	+0.470	+0.060
No Signal	0V	+0 V/nm	+0.370	+0.350	+0.270
Reset	-8V	-2 V/nm	+0.027	+0.026	+0.947

Table 3.5: Partition of electrons between dots in function of the clock field[7] independently on the driver voltage.

By looking at the Hold State simulation, it's evident that in this case, all electrons are in the upper dots and so, at this point, by applying the right driver voltage, it's possible to polarise the charges where needed.

3.3.1 Logic State '1' combined with the Hold State

In this case, the situation is shown in the figure 3.4, and in order to determine the total amount of charge along each wire, also the equivalent circuit has been realised to use the standard electronic method.

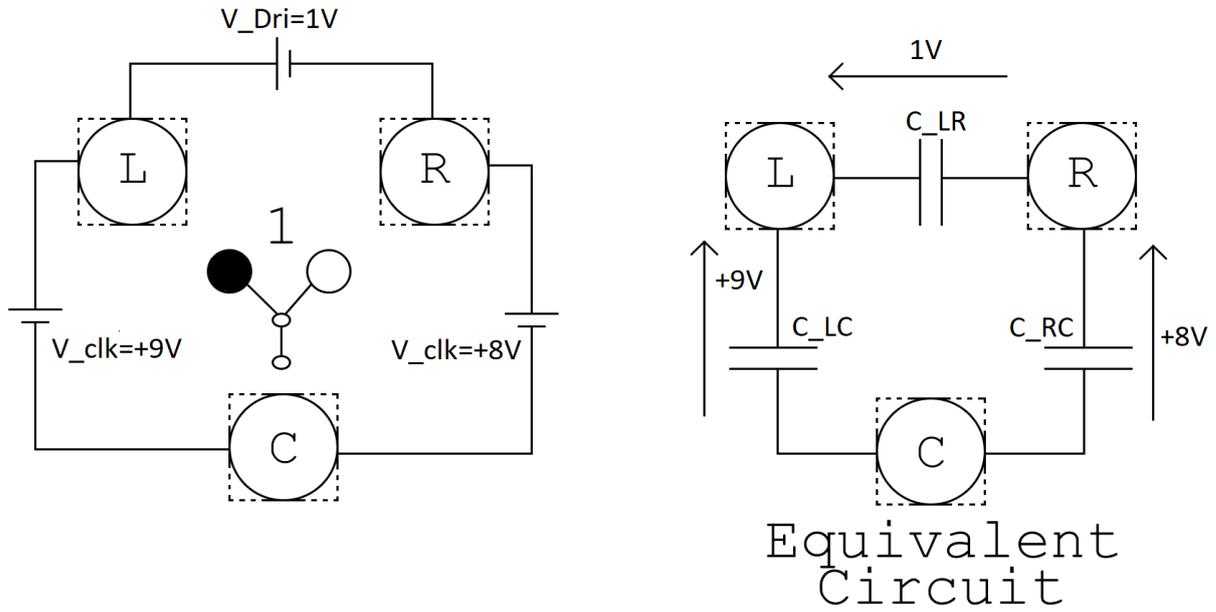


Figure 3.4: Combination of the Hold states with the logical value '1'.

By starting from the left electrode:

$$\lambda_L = \frac{C_{LR}}{l} \cdot (V_L - V_R) + \frac{C_{LC}}{l} \cdot (V_L - V_C); \quad (3.5)$$

then, for the right electrode:

$$\lambda_R = \frac{C_{LR}}{l} \cdot (V_R - V_L) + \frac{C_{RC}}{l} \cdot (V_R - V_C); \quad (3.6)$$

and in the end about the clock electrode:

$$\lambda_C = \frac{C_{LC}}{l} \cdot (V_C - V_L) + \frac{C_{RC}}{l} \cdot (V_C - V_R); \quad (3.7)$$

where, of course, V_R , V_L and V_C are the potentials on the right, left and clock electrodes respectively. As a matter of fact, in this case, the different potentials are:

$$\begin{cases} V_L = 9V \\ V_R = 8V \\ V_C = 0V \end{cases} \quad (3.8)$$

By comparing the values obtained from the model implemented on Matlab and Comsol, like it's possible to appreciate in the table 3.6, there is a confirmation.

Simulation		COMSOL	Matlab	Error
#17	λ_L	174.40 pC/m	152.25 pC/m	12.7 %
#18	λ_R	109.95 pC/m	95.44 pC/m	13.2%
#19	λ_C	-284.37 pC/m	-247.70 pC/m	13.9%

Table 3.6: Comparison between Comsol and Matlab about the charge distribution for each wire in the Hold state with logical value '1'.

Notice that, differently with the driver mode, the results obtained with the driver plus hold mode are more precise. Consequently, it could be a good idea to use also this mode to impose the input inside the circuit.

3.3.2 Logic State '0' combined with the Hold State

Now, the same analysis has to be done by imposing the '0' logic state.

The equations 3.5, 3.6 and 3.7 are, of course, still true. As a matter of fact, to obtain the values of each linear charge distribution for every wire, it's necessary to substitute in these equations the new voltage potentials that are present on the electrodes, which are:

$$\begin{cases} V_L = 8V \\ V_R = 9V \\ V_C = 0V \end{cases} \quad (3.9)$$

This can be simplified by looking at the equivalent circuit that represents the case under test, which is shown below in the figure 3.5. By comparing again the values obtained from the model implemented on Matlab and Comsol, like it's possible to appreciate in the table 3.7, also in this case, there is a confirmation.

By comparing the table 3.6 and the table 3.7, it is really interesting to notice that the linear charge on the C electrode doesn't change, while between the right and left electrode there is an inversion of charge. This is right since in both cases the hold mode is imposed and since this is managed from the clock field which is based on the C electrode,

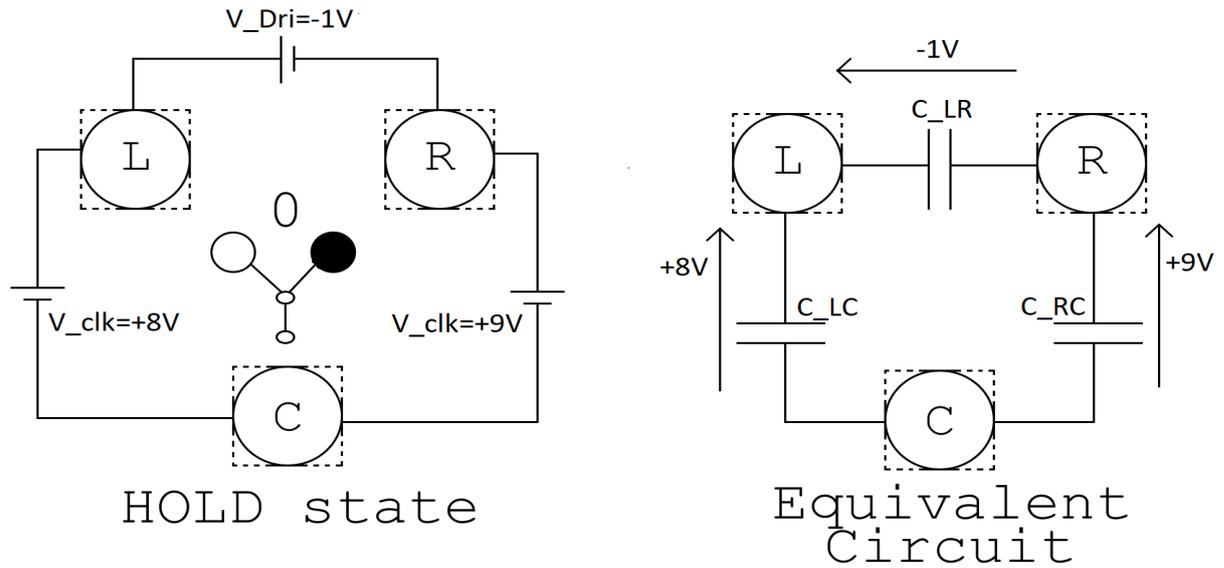


Figure 3.5: Combination of the Hold states with the logical value '0'.

Simulation		COMSOL	Matlab	Error
#20	λ_L	109.95 pC/m	95.44 pC/m	13.2%
#21	λ_R	174.40 pC/m	152.25 pC/m	12.7 %
#22	λ_C	-284.37 pC/m	-247.70 pC/m	13.9%

Table 3.7: Comparison between Comsol and Matlab about the charge distribution for each wire in the Hold state with logical value '0'.

the linear charge on it is constant while the static logic depends on the voltage between upper dots, and so in base on which input we are imposing through the driver, there is the corresponding linear charge.

At this point, all information is known and it's able to launch the model implemented in Matlab and look at what happens. Since it is very important to demonstrate which are the critical issues of the model, for all possible states a comparison with Comsol will be considered.

Chapter 4

Verification of the finite wire model

By definition, a scientific experiment or measurement is something which must be repeatable and referable to the specific test. With this assumption, in this first part of the chapter dedicated to the verification step, used to check the model which has been implemented in Matlab, all data and references are shown.

4.1 Geometry

Just for the verification chapter, the following axes reference has been used in order to have a symmetric structure of electrodes and simplify the test.

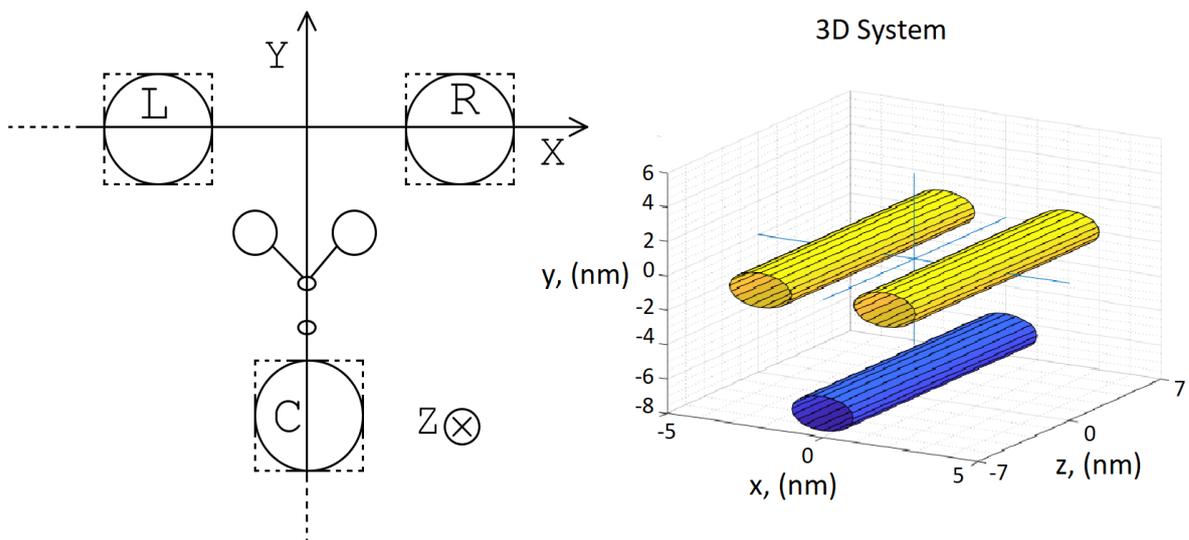


Figure 4.1: Reference system used to verify the model in the verification chapter.

Now, all parameters, for each wire, are shown, of course, by doing reference to the figure 4.1. By starting from the left electrode, its parameters are [4]:

Parameter	Symbol	Value
Radius	r_L	1 nm
Length	l_L	10 nm
x Coordinate	x_{c_L}	-2 nm
y Coordinate	y_{c_L}	0 nm
z Coordinate	z_{c_L}	0 nm

Table 4.1: Geometrical parameters of the left electrode.

For the right wire, the parameters are:

Parameter	Symbol	Value
Radius	r_R	1 nm
Length	l_R	10 nm
x Coordinate	x_{c_R}	+2 nm
y Coordinate	y_{c_R}	0 nm
z Coordinate	z_{c_R}	0 nm

Table 4.2: Geometrical parameters of the right electrode.

and, in the end, for the clock electrode:

Parameter	Symbol	Value
Radius	r_C	1 nm
Length	l_C	10 nm
x Coordinate	x_{c_C}	0 nm
y Coordinate	y_{c_C}	-6.6 nm
z Coordinate	z_{c_C}	0 nm

Table 4.3: Geometrical parameters of the clock electrode.

By considering the system shown in the figure 4.1, it's clear that the parameter D_w represents, as it's been told before, the distance between the right and left electrodes which can be easily determined like:

$$D_w = x_{c_R} - x_{c_L} = 4 \text{ nm} \quad (4.1)$$

that is the distance for which all comparisons between the different capacitance values have been done in the chapter 3.

The volumes, on which the electric field has been evaluated, are:

Parameter	Symbol	Value
x axis	Px	$[-(D_w + r_L), (D_w + r_L)]$
y axis	Py	$[-3 \cdot r_L ; 3 \cdot r_L]$
z axis	Pz	$[-(l_L + 4 \cdot r_L)/2, (l_L + 4 \cdot r_L)/2]$

Table 4.4: Considered volume during the testing of driver mode.

Parameter	Symbol	Value
x axis	Px	$[-(D_w + r_L), (D_w + r_L)]$
y axis	Py	$[+y_{c_C} - 3 \cdot r_L ; 3 \cdot r_L]$
z axis	Pz	$[-(l_L + 4 \cdot r_L)/2, (l_L + 4 \cdot r_L)/2]$

Table 4.5: Considered volume during the testing of clock mode.

Notice that everything is kept parametric to easily adapt the model to different structures with distinct geometrical parameters.

Before to go on, it must be clear that it's impossible to compare results for every point, inside the considered volumes, between the numerical simulator, Comsol, and Matlab. As a consequence, the most important plots will be shown to demonstrate that the model implemented in Matlab works, of course, with some tolerance.

Above, in the chapter 3, different states have been mentioned, like the driver mode, the hold and reset mode obtained through the clock field and, in the end, the driver combined with the hold mode. So, the results of all these states will be shown.

By wanting to help the reader, it could be very useful to show the planes and the lines along which different components, of the electric field, will be considered. As a matter of fact, it could be a bit difficult to imagine the direction under investigation inside the structure, at least, at the beginning of the report. So, with this hypothesis, all possibles cases will be plotted. As you will discover, the mode which is always the first to be analysed for every different implemented model, is the driver. So, let's start from it

4.1.1 Directions and planes for which the electric field, in the driver mode, has been considered

By starting from the driver mode, the typical directions, along which the electric field is analysed, are:

- Line along x-direction, for $y=0\text{nm}$ and $z=0\text{nm}$, which is used to understand how much the structure of electrodes polarises the molecule, which is shown below the figure 4.2;
- Plane XY, for $z=0\text{nm}$, which cuts in the middle the structure, used to see if the obtain field respects the expectations, which is shown below the figure 4.3;
- Line along z-direction, for $x=0\text{nm}$ and $y=0\text{nm}$, used to highlight the problem related to $l \gg Dw$ (don't worry if now it is not clear since it will be explained after with much more details) which is shown below the figure 4.4;
- Plane XZ, for $y=0\text{nm}$, to highlight how the entire structure works, which is shown below the figure 4.5.

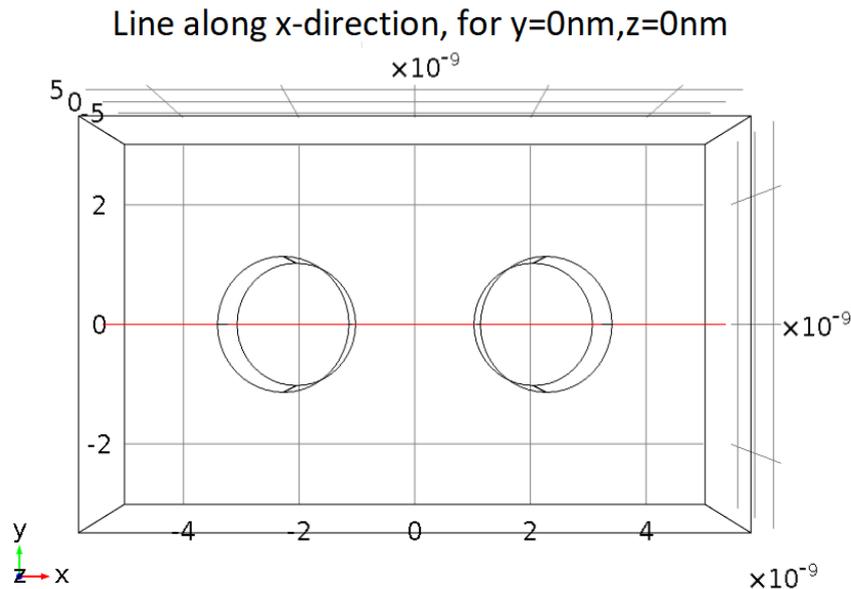


Figure 4.2: Considered line along x-direction, for $y=0\text{nm}$ and $z=0\text{nm}$, in the driver mode.

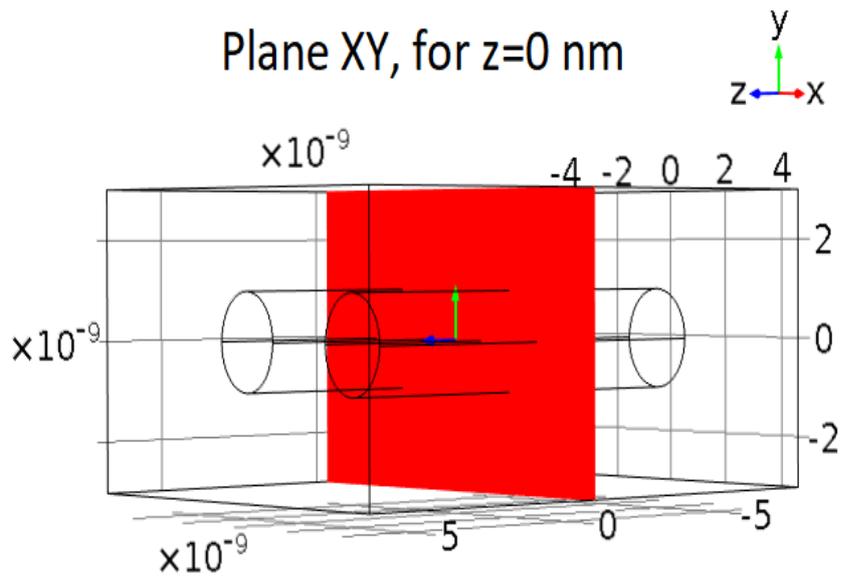


Figure 4.3: Considered plane XY, for $z=0$ nm in the driver mode.

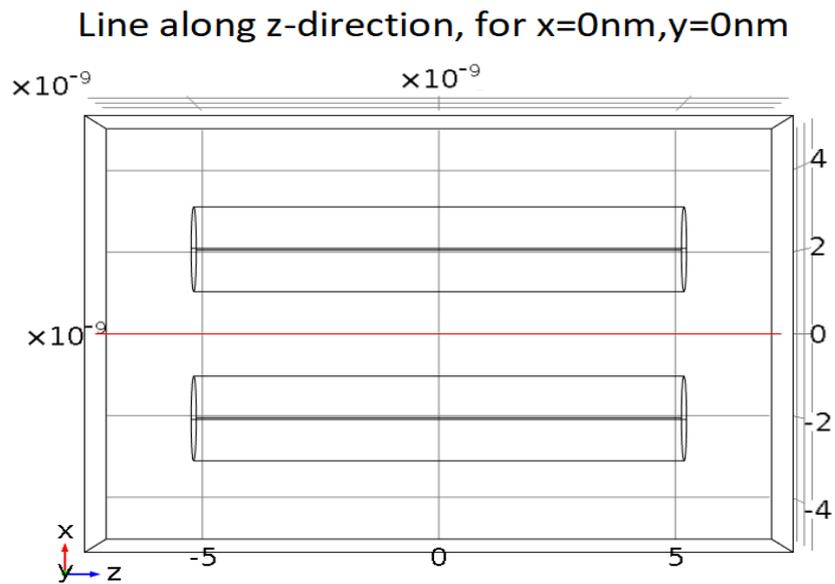


Figure 4.4: Considered line along z-direction, for $x=0$ nm and $y=0$ nm, in the driver mode.

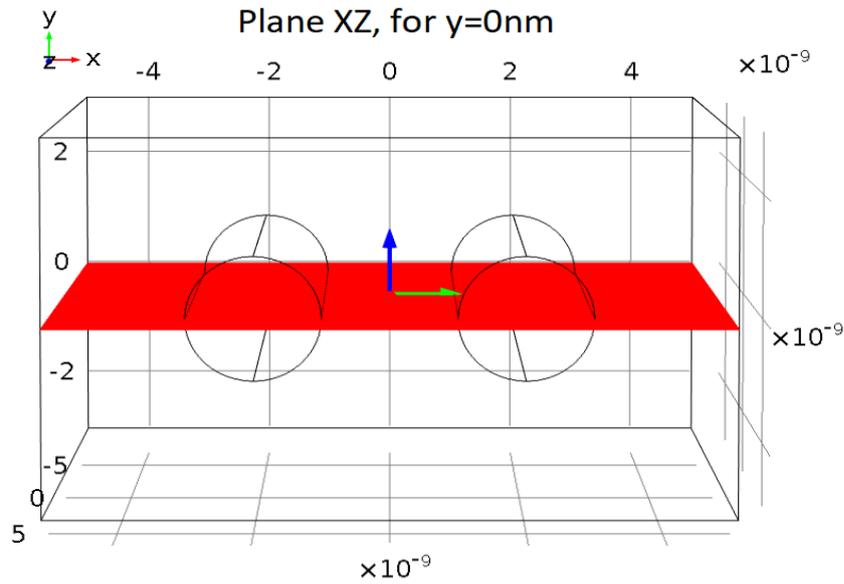


Figure 4.5: Considered plane XZ, for $y=0\text{nm}$ in the driver mode.

Now, let's do the same for the clock mode, which will be used to study the hold and reset mode, again to help the reader.

4.1.2 Directions and planes for which the electric field, in the clock mode, has been considered

As it will be highlighted a lot of times later during the report, this is the most important field since it is fundamental to guarantee the right propagation of the information along with complex structures. For it, the following directions have been analysed:

- Plane XY, for $z=0\text{nm}$, which cuts in the middle the structure, used to understand how the model works and how the electric field appears, which is shown below the figure 4.6;
- Line along the y-direction, for $x=0\text{nm}$ and $z=0\text{nm}$, used to plot the electric field along the entire structure and understand where molecules have to be placed to be polarised in the right way, which is shown below the figure 4.7;
- Line along z-direction, for $x=0\text{nm}$ and $y=-3.6\text{ nm}$, used to highlight the problem related to $l \gg Dw$, (don't worry if now it is not clear since it will be explained after with much more details), which is shown below the figure 4.8.

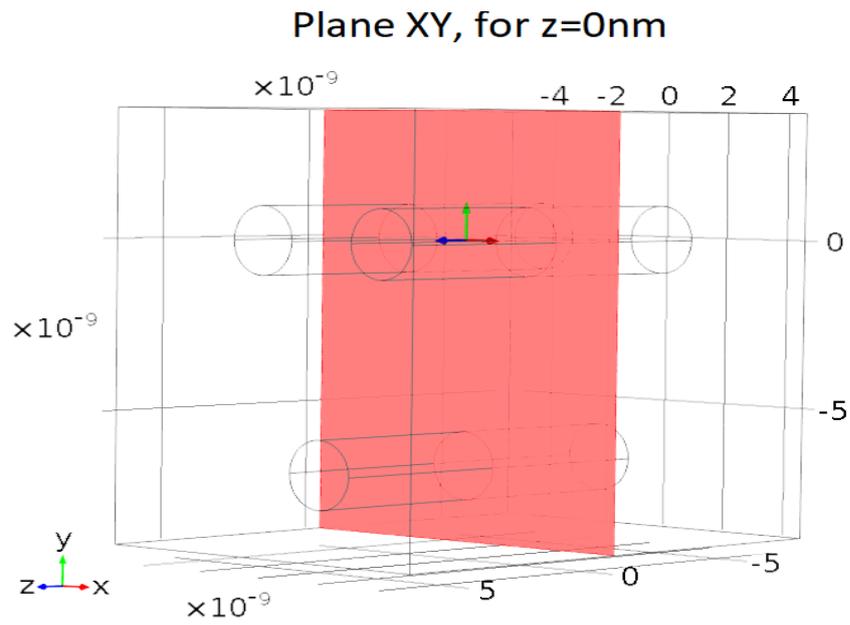


Figure 4.6: Considered plane XY, for $z=0\text{nm}$ in the clock mode.

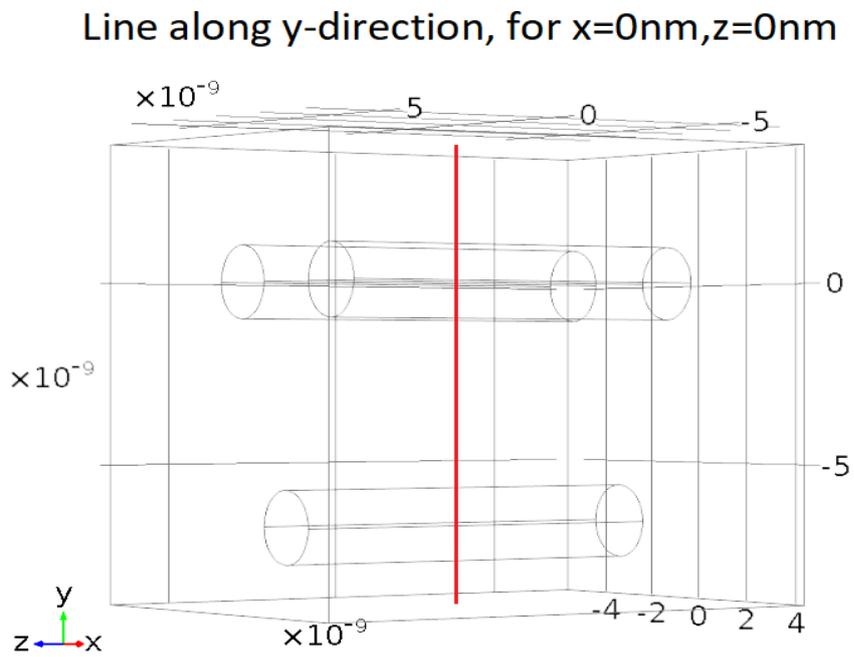


Figure 4.7: Considered line along y-direction, for $x=0\text{nm}$ and $z=0\text{nm}$, in the clock mode.

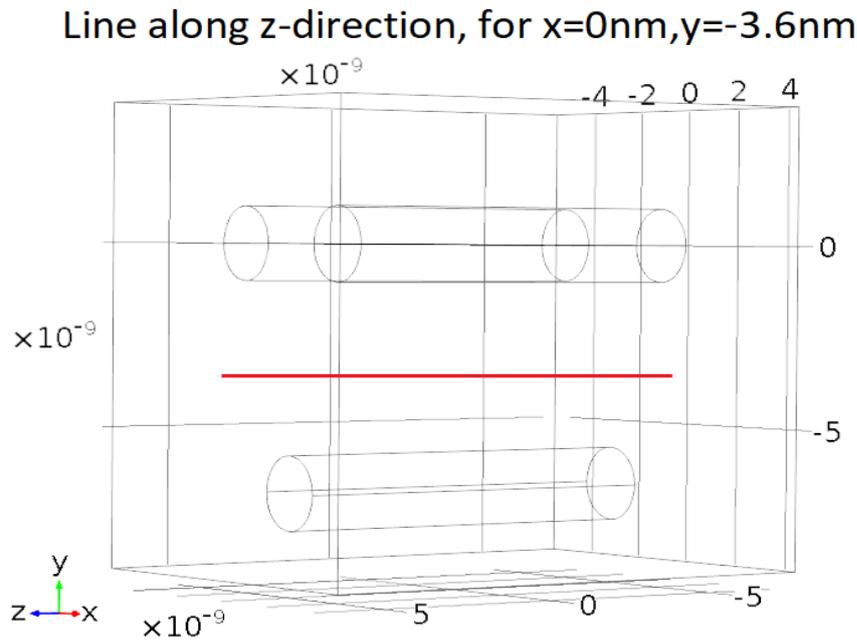


Figure 4.8: Considered line along z-direction, for $x=0\text{nm}$ and $y=-3.6\text{nm}$, in the clock mode.

By hoping that everything has been explained, to render for the reader as easy as possible this journey along this complex aim: modelling the electric field in a structure dedicated to molecular QCA, the first results are shown below starting from the driver mode.

4.2 $E_x(x,y,z)$ for finite wire model in driver mode

During the driver mode, the aim is to polarise the molecules, by partitioning the charge principally between dot1 and dot2. As a consequence, the most important component of the electric field is that along the x coordinate, by doing reference to the figure 4.2.

By starting from the driver mode, and in the particular from the static logic '1', by looking at the plots 4.9, it's very interesting to certify that the model is strongly in agreement with the numerical simulator. The model implemented in Matlab is underestimated with respect to the Comsol result. Of course, there is nothing to be surprised about that since, as it has been evaluated in the chapter dedicated to estimation of the charge distribution along wires, the capacitance got from the equation 3.2 is smaller than the reference of an amount equal to 20% which is exactly the different between the Matlab and Comsol curve. So, everything is coherent.

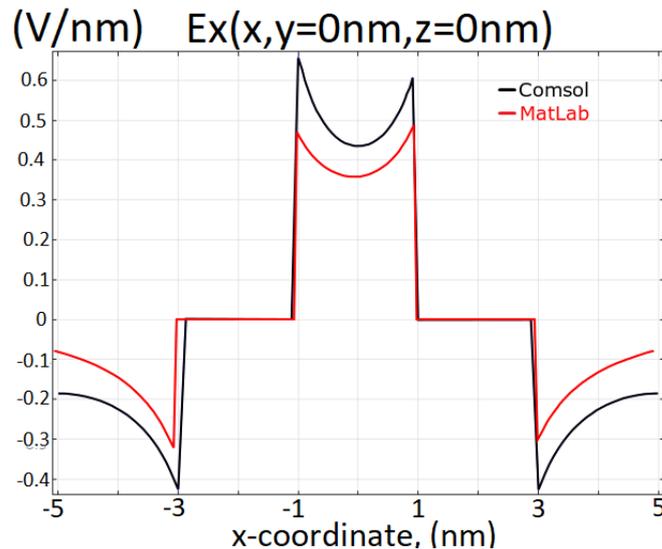


Figure 4.9: Comparison between Matlab and Comsol about $E_x(x,y=0\text{nm},z=0\text{nm})$ by forcing the static logic '1'.

It could be also interesting to compare the electric field between electrodes from another point of view, which is in function of a plane that cuts the structure in the middle for coordinate $z=0$ nm, (to help, look at the image 4.3), like it's shown in the figure 4.10. It's possible to appreciate that the field is exactly like aspected between two metal pieces with opposite charge, like in a standard capacitor.

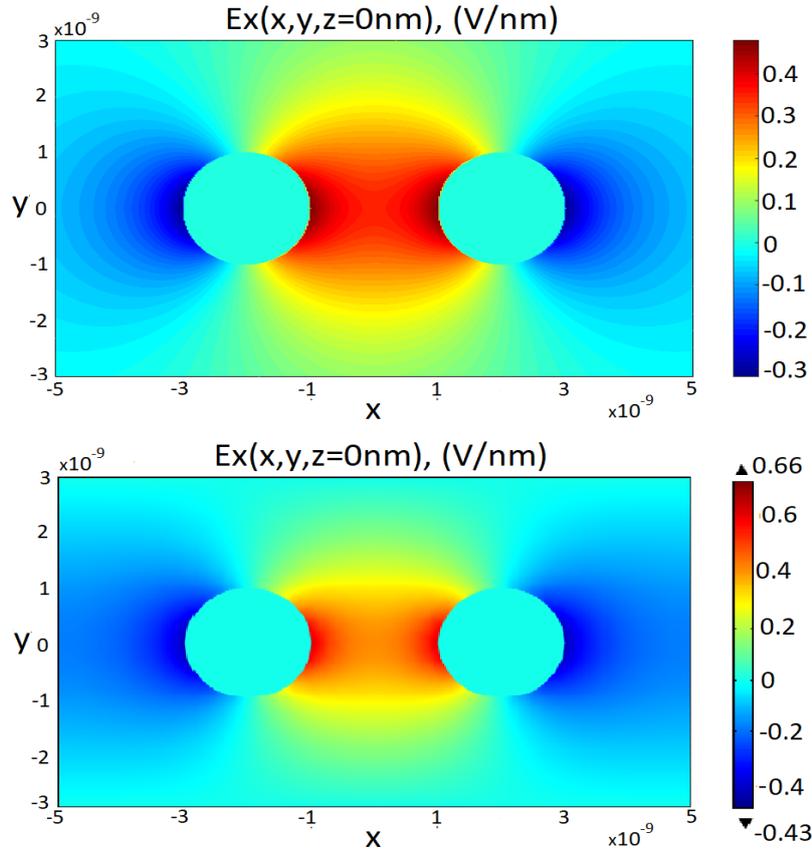


Figure 4.10: Comparison between Matlab (on the top) and Comsol (on the bottom) about $E_x(x, y, z=0\text{nm})$ by forcing the static logic '1'.

Of course, this is a wonderful starting point since the obtained results are very promising. As a consequence, let's continue by looking at what happens by forcing the static logic '0'.

By looking to the figure 4.11 and 4.12 which are shown below, again a very agreement between Matlab and Comsol simulator is obtained. Also, in this case, the same considerations are true about the fact that Matlab is underestimated due to the smaller capacitance.

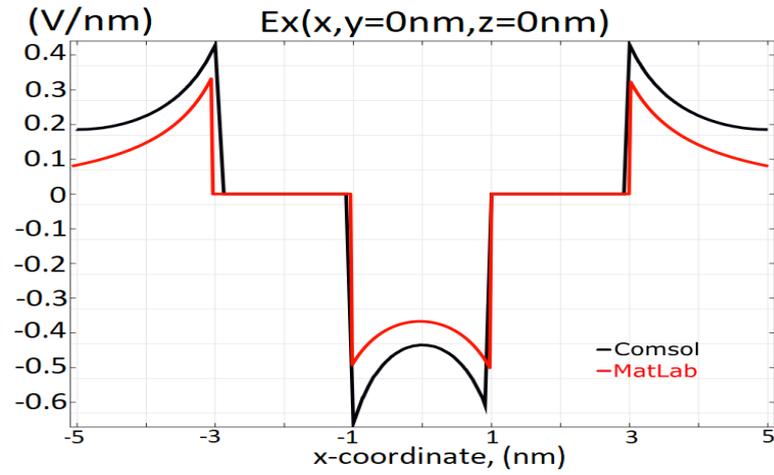


Figure 4.11: Comparison between MatLab and Comsol about $E_x(x,y=0nm,z=0nm)$ by forcing the static logic '0'.

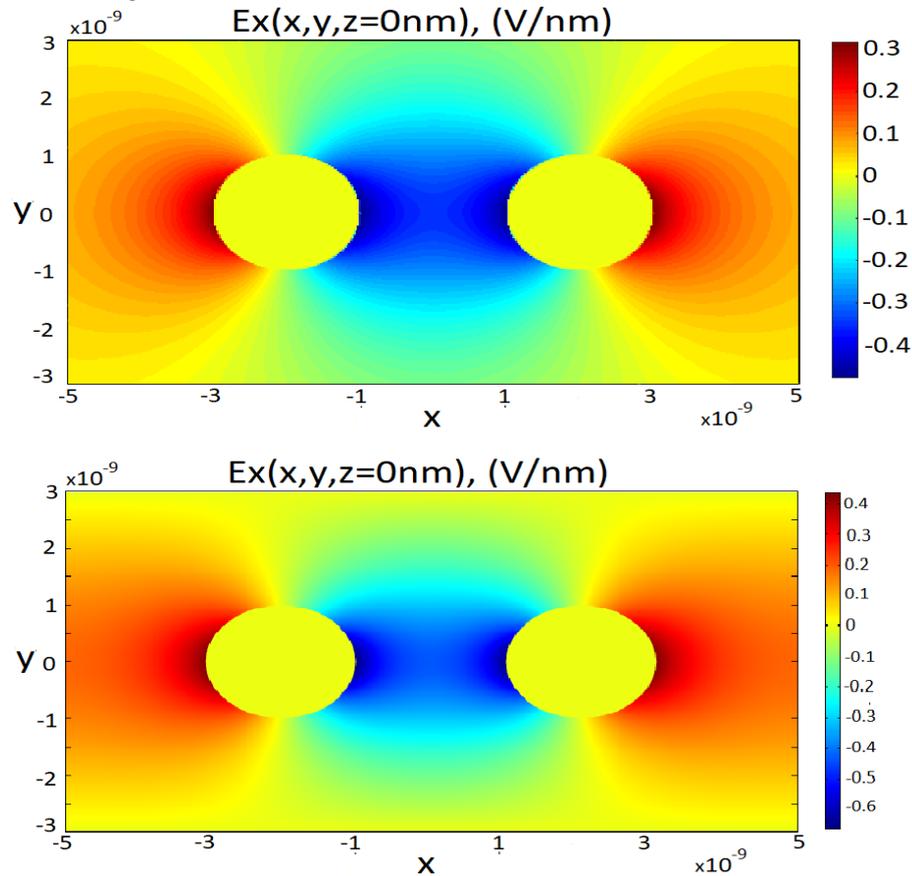


Figure 4.12: Comparison between Matlab (on the top) and Comsol (on the bottom) about $E_x(x,y,z=0nm)$ by forcing the static logic '0'.

At this point, it is interesting to analyse how the electric field is distributed along the z-axis, which is the longitudinal axis concerning the structure of electrodes, of course, by doing reference at the figure 4.4.

Here it's possible to appreciate that the two plots are enough different. We are not speaking about the maximum value since it is connected to the problem regards capacitance, but in Comsol the function is flatter than the Matlab model. So, it looks like the implemented model is more sensible at the end of the wire.

Also, in this case, this is not a surprise, since the model of the electric field is true for long wire.

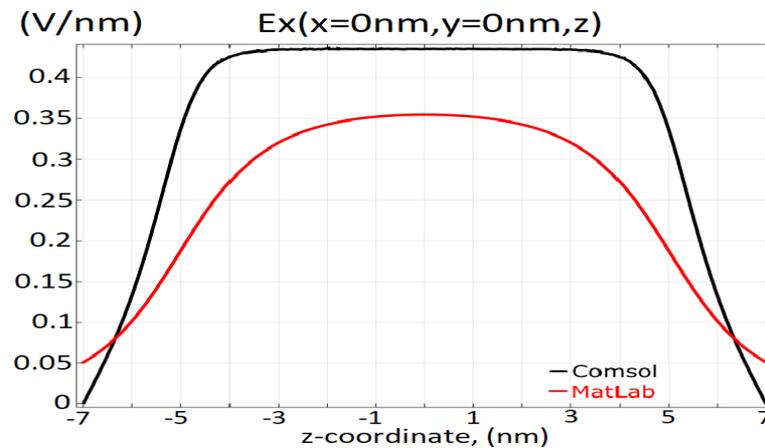


Figure 4.13: Comparison between Matlab and Comsol about $E_x(x=0\text{nm},y=0\text{nm},z)$ by forcing the static logic '1'.

By wanting to demonstrate that this is true, a comparison with a structure with longer electrodes has been done, by considering wires with a length equal to 40 nm, to respect the assumption for which $l \gg D_w$.

Like it's able to appreciate in the figure 4.14, which is shown below, now the two curves have exactly the same behaviour. The difference between the two regards, again, the different capacitance but the results is very good and confirm the limit of the model.

It will be shown that this problem will appear also with the hold, reset and driver plus hold mode since it is an intrinsic limit of the model itself. After, another solution will be considered to overcome it, but, anyway, it's very important to not forget that this is just a model with its tolerance. Since the results are distance from the numerical solution of 10,15%, more or less, we can be very proud about this starting point for the implemented model.

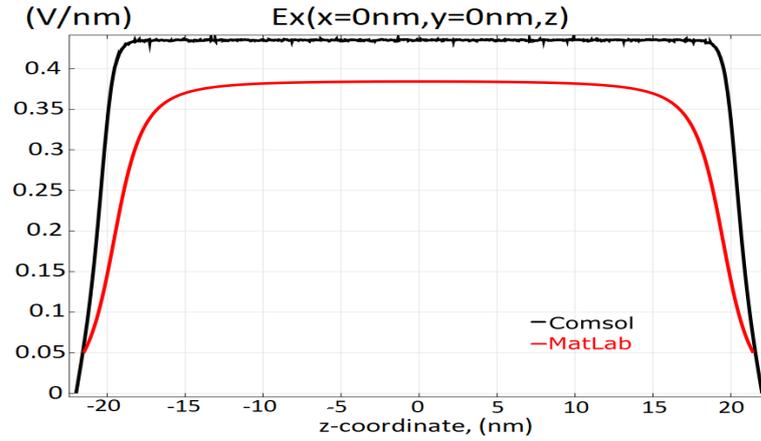


Figure 4.14: Comparison between Matlab and Comsol about $E_x(x=0nm, y=0nm, z)$ by forcing the static logic '1' with length for each wire equal to 40 nm.

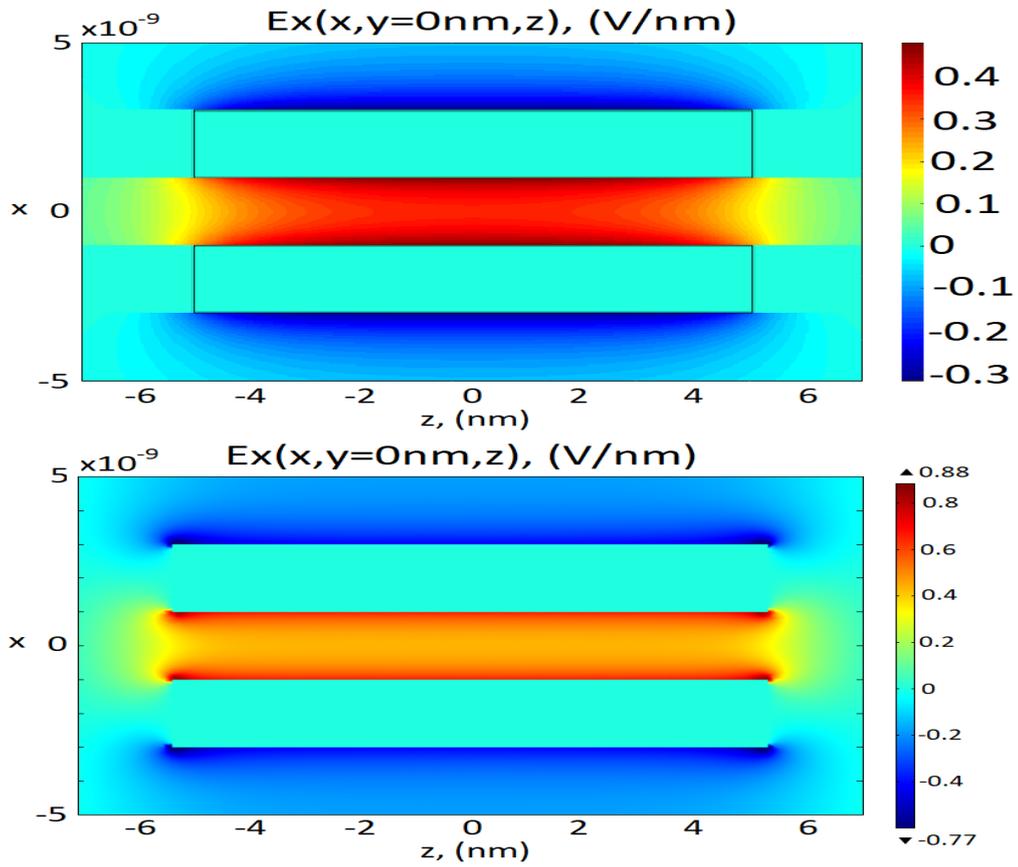


Figure 4.15: Comparison between Matlab (on the top) and Comsol (on the bottom) about $E_x(x, y=0nm, z)$.

Now that the component $E_x(x,y,z)$ of the electric field has been heavily analysed, it's time to move on by considering the vertical component.

4.3 $E_y(x,y,z)$ for finite wire model in driver mode

In this case, the obtained results are shown in the figure 4.16, and so, also, in this case, there is a very strong agreement.

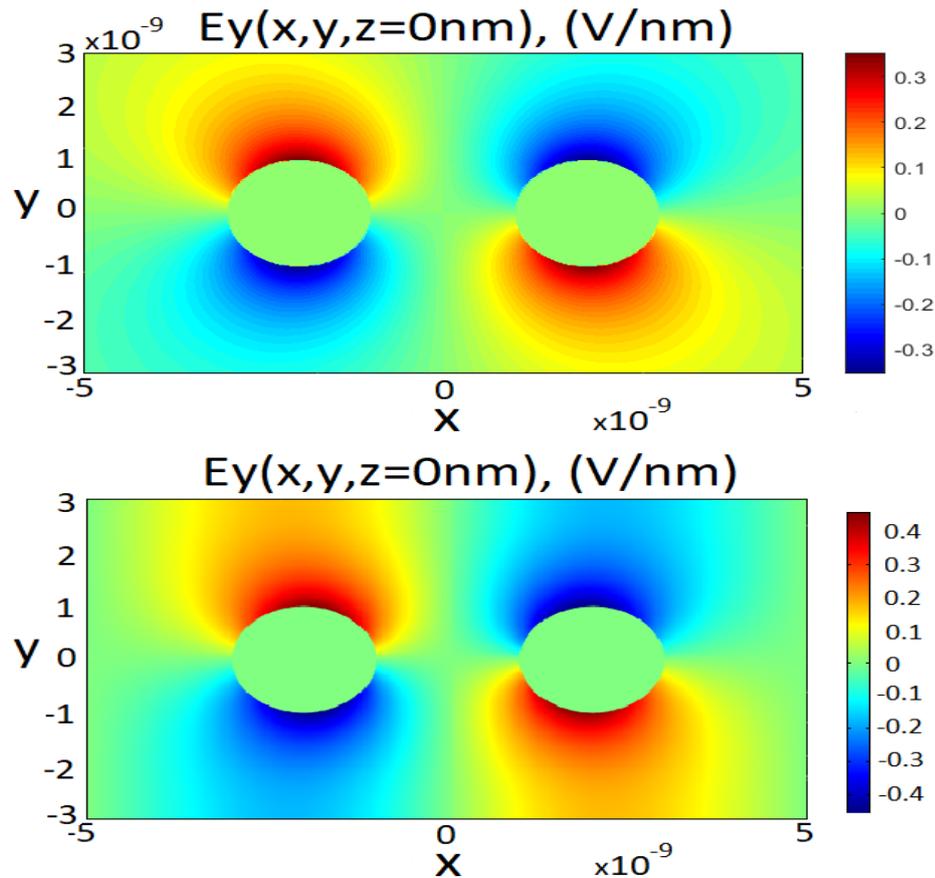


Figure 4.16: Comparison between Matlab (on the top) and Comsol (on the bottom) about $E_y(x,y,z=0)$.

It could be also interesting to analyse, by looking at the figure 4.17 shown below, the electric field in $x=-0.5$ nm and $x=+0.5$ nm, which are, theoretically, the coordinates for which the two dot1 and dot2 are presents.

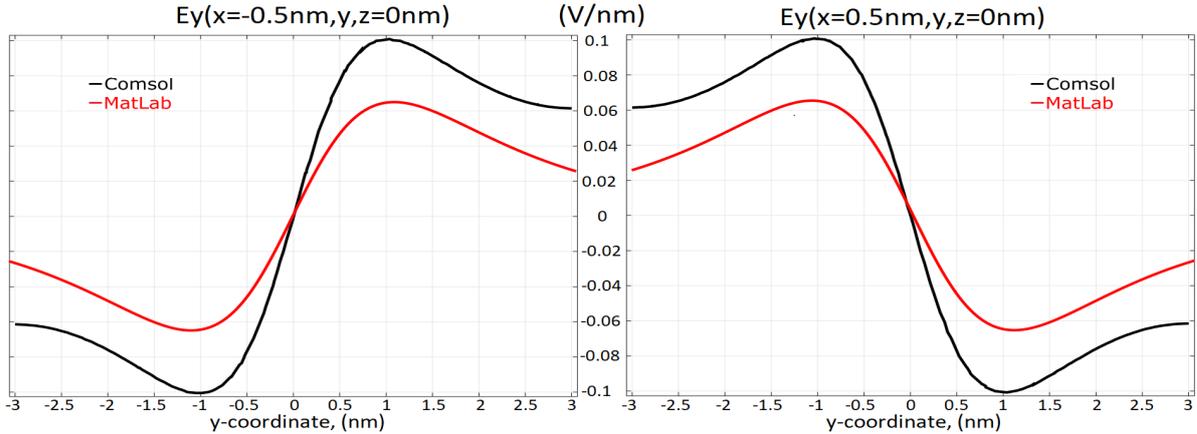


Figure 4.17: Comparison between Matlab and Comsol about $E_y(x=-0.5\text{nm},y,z=0\text{nm})$ (on the left) and $E_y(x=+0.5\text{nm},y,z=0\text{nm})$ (on the right) by forcing the static logic '1'.

From a behavioural point of view, there is a perfect match, so also here we have to be proud of it. The difference in the amplitude depends, again, on the capacitance value. What's important to highlight is that $E_y(x,y,z)$ is smaller than $E_x(x,y,z)$, almost of one order of magnitude. It's normal since, due to the fact the two electrodes have opposite charges, along the y -axis, there is a sort of compensation. Now, it is not perfect since the electric field along each axis is not linear, but it is inversely proportional concerning the distance, and so this justifies the not perfect compensation. As a matter of fact, for $x=0\text{nm}$, that is the point for which the structure of electrodes is symmetric, $E_y(x=0\text{nm},y,z)=0$ V/nm for every point.

4.4 $E_z(x,y,z)$ for finite wire model in driver mode

Finally, it's time of the last component $E_z(x,y,z)$, that is the longitudinal component concerning the reference system shown in the figure 4.1 where something strange happens.

By going on to consider the electric field in $x=-0.5\text{nm}$ and $x=0.5\text{nm}$, the curves have been obtained are shown below, in the figure 4.18. In this case, the model doesn't follow the real case. The reason, for which the worst results are obtained for this component of the electric field, is that along the z -axis, the most important violation is present, that is $l \gg D_w$. To prove that it is true, simulation with longer wires has been done like it's able to appreciate by looking at the figure 4.19.

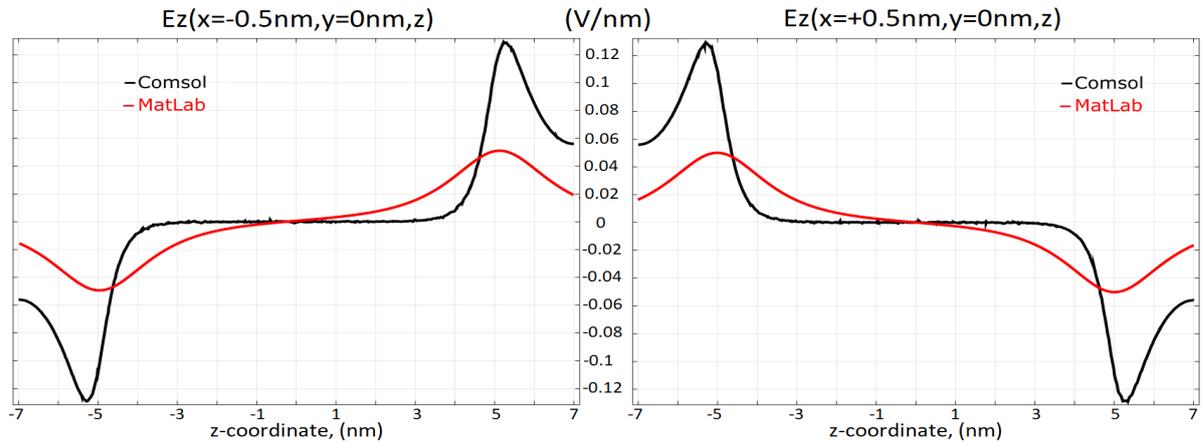


Figure 4.18: Comparison between Matlab and Comsol about $E_z(x=-0.5\text{nm}, y=0\text{nm}, z)$ (on the left) and $E_z(x=+0.5\text{nm}, y=0\text{nm}, z)$ (on the right) by forcing the static logic '1'.

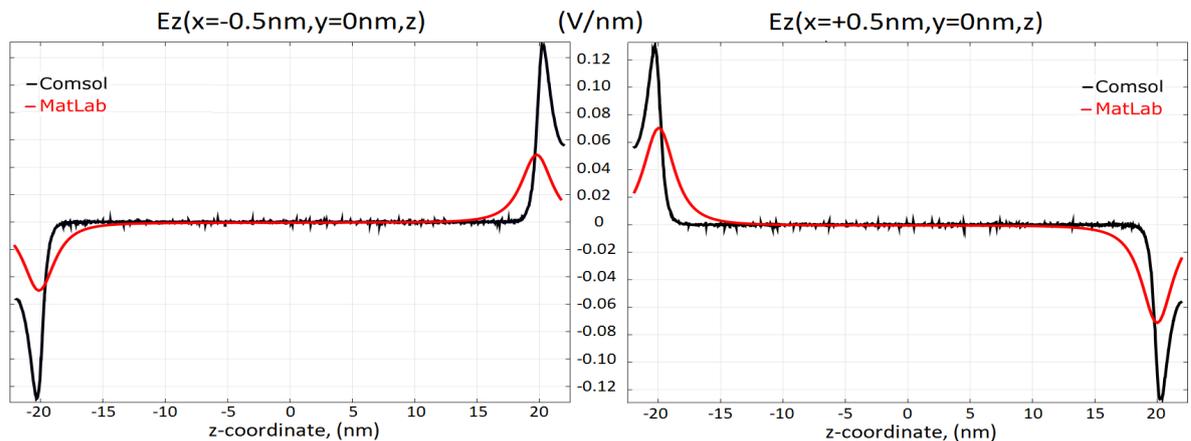


Figure 4.19: Comparison between Matlab and Comsol about $E_z(x=-0.5\text{nm}, y=0\text{nm}, z)$ (on the left) and $E_z(x=+0.5\text{nm}, y=0\text{nm}, z)$ (on the right) by forcing the static logic '1' with length for each wire equal to 40 nm.

Before to think about how to interpret these results, it's fundamental to remember what happens in an infinitely long wire.

In a theoretical case, with an infinitely long wire [19], the only component that survives is the radial, since the longitudinal is always zero.

From the curve shown in the figure 4.18 and 4.19, by considering just plots got from Comsol, it's evident that for almost the entire length of wires $E_z(x,y,z)$ is zero expect near

to the end of their, so it's like there is a sort of boundary effect.

As a consequence, let's conclude that in this model the $E_z(x,y,z)$ component can be neglected, at least, in a first approximation.

The driver mode has been analysed from all possible points of view. Now that the strengths and weaknesses of the implemented model have been shown, it's possible to go on with the hold, reset and, in the end, with the combination of hold plus driver mode. By wanting to avoid to write a boring and heavy scientist document, for the remaining states, just the vertical component $E_y(x,y,z)$ will be analysed, since it's the component used to drive the entire system by moving electrons up and down between dots.

4.5 $E_y(x,y,z)$ for finite wire model in reset mode

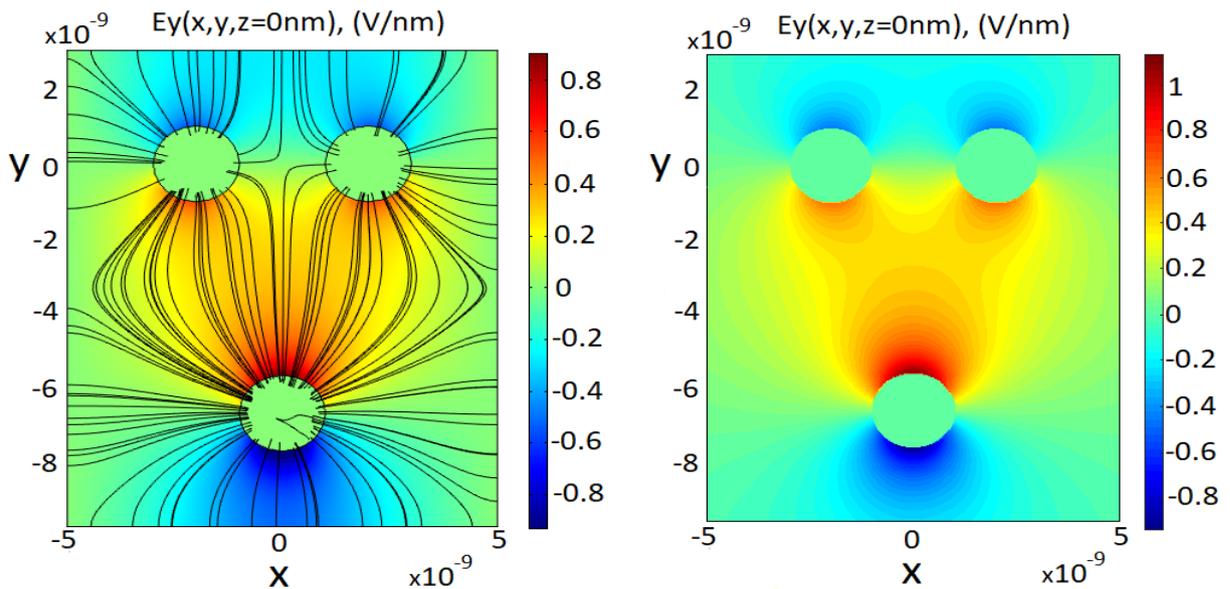


Figure 4.20: Comparison between Comsol (on the left) and Matlab (on the right) by imposing the reset mode.

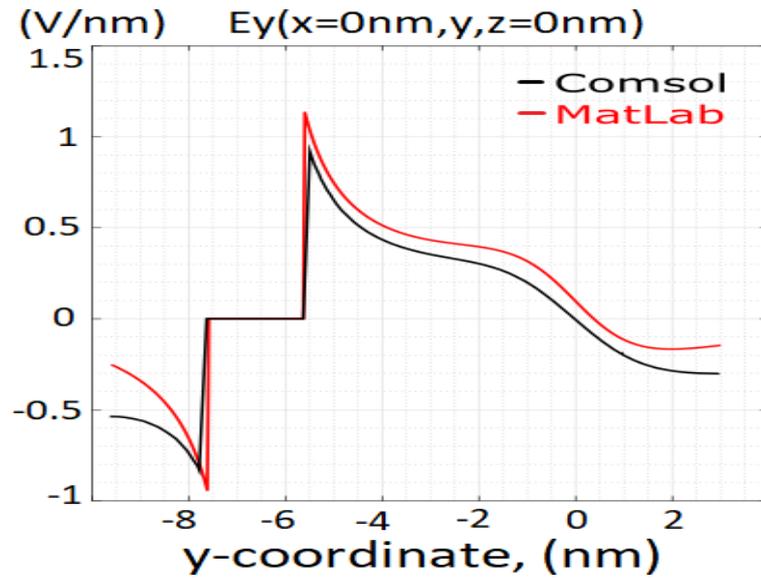


Figure 4.21: Comparison between Cmsol and Matlab by imposing the reset mode considering the $E_y(x=0nm, y, z=0nm)$.

4.6 $E_y(x, y, z)$ for finite wire model in hold mode

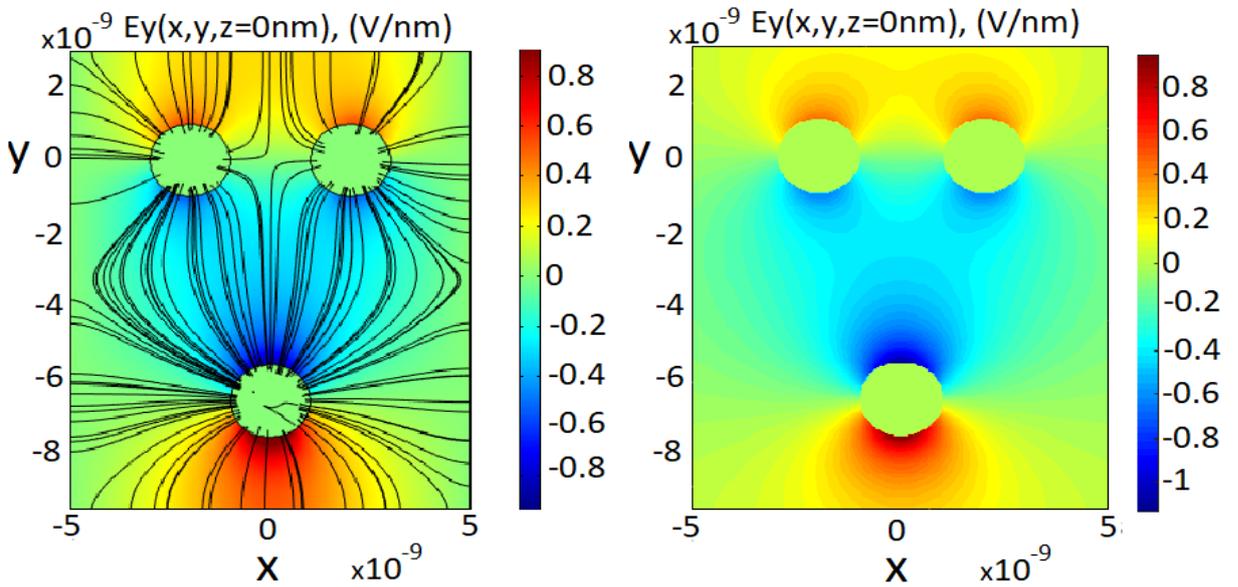


Figure 4.22: Comparison between Cmsol (on the left) and Matlab (on the right) by imposing the Hold mode.

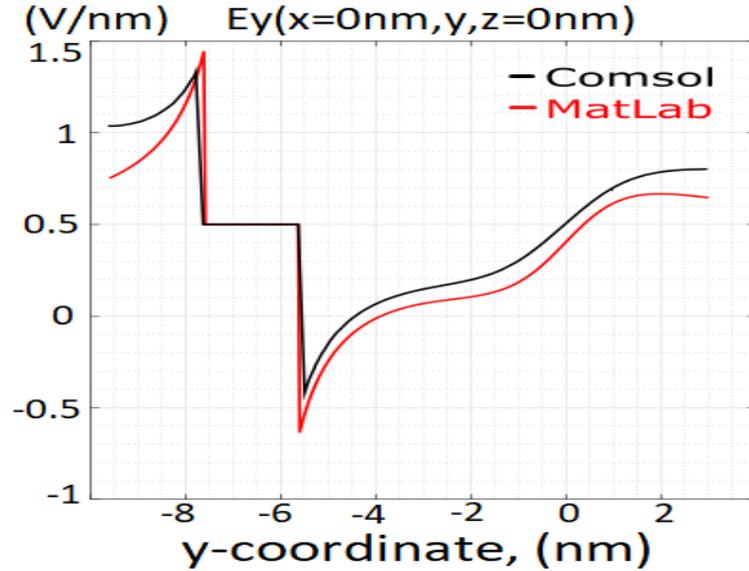


Figure 4.23: Comparison between Comsol and Matlab by imposing the Hold mode considering the $E_y(x=0\text{nm},y,z=0\text{nm})$.

By looking at the figure 4.21 and at the figure 4.23, which are shown above, it's more evident that in these conditions, the difference between results obtained from the implemented model and Comsol are smaller than the driver mode. This depends on the fact that the physical distance between the clock wire and the right and left electrodes is bigger and so, the capacitance equation can get more precise values.

This is a very important result since, by thinking about how the circuit works, once that the static logic has been imposed to the input through the driver mode, the correct behaviour and the right propagation of the information is managed with the clock electric field.

Another interesting aspect has to be highlighted. The $E_y(x,y,z)$ component is used to drive electrons vertically through dots. By looking at the figures where the field lines are shown, that are on the left of images 4.20 and 4.22, it's able to understand that the perfect position for molecules is towards the clock electrode, instead of the right and left electrodes, where the lines are practically vertical. As a consequence, very good control can be achieved.

Now, it is time to look at what happens at the vertical field along the z -coordinate, which is the longitudinal, to highlight the effect of the violation about $l \gg D_w$ assumption. The field is plotted following the orientation shown in the figure 4.8 while it's able to appreciate the result in the image 4.24. Also for this case, there are some problems for the field along the longitudinal direction, even if in this case, the field is overestimated. But, anyway, generally speaking, the results got for the clock mode are better than the driver

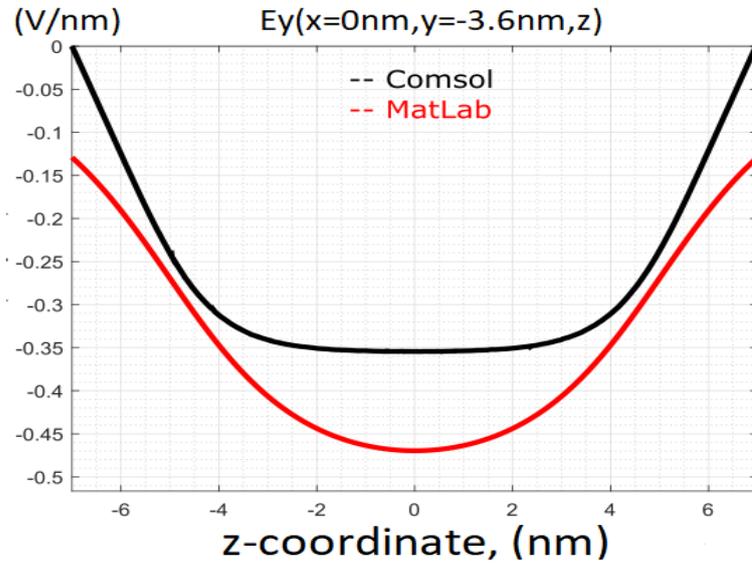


Figure 4.24: Comparison between Comsol and Matlab by imposing the Hold mode considering the $E_y(x=0\text{nm},y=-3.6\text{nm},z)$.

mode, since the distance between electrodes increases, and so, again, it is a confirmation of the limit for the model.

Since the structure with three electrodes give in output better results than the structure with just two wires, it may be reasonable try to impose the input with the driver mode combined with the hold to use the compensation effect that the model shown and to get smaller error.

4.7 $E_y(x,y,z)$ for finite wire model in hold+driver mode

In this case the aim is to push electrons towards up, thanks to the clock field, which is, as it's known, vertical and then, through to the driver mode, we are able to modulate the electrons between the dot1 and dot2, in order to impose the wanted static logic inside the circuit.

4.7.1 Hold+driver by forcing the logic state '1'

Let's start from hold mode and, contemporaneously, by forcing the logic state '1'. There is a perfect match, which is much better than the only driver mode.

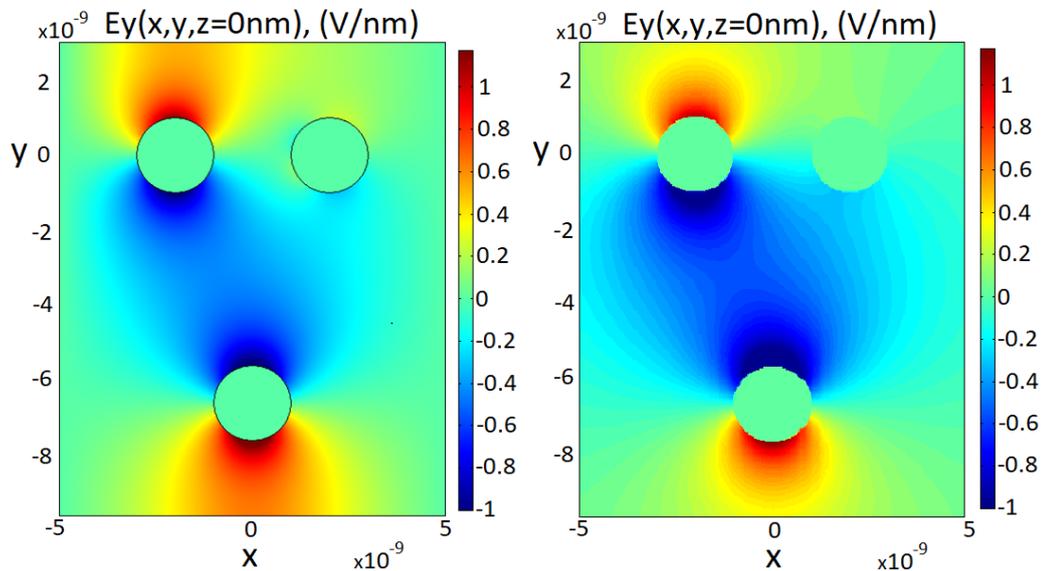


Figure 4.25: Comparison between Cmsol and Matlab by imposing the '1' static logic through the Hold+Driver mode considering the $E_y(x,y,z=0\text{nm})$.

By looking at the figure 4.25, it's curios to highlight that the vertical field is polarised, by following the imposed static logic.

Now, let's go on by looking at what happens with the '0' static logic. Like it's possible to appreciate in the figure 4.26, the same thing happens but, of course, in the opposite direction since a negative driver voltage is applied. Both cases are perfectly in agreement with the results obtain from Cmsol, so this justifies the reason for which this particular model could be used instead of the only driver mode.

4.7.2 Hold+driver by forcing the logic state '0'

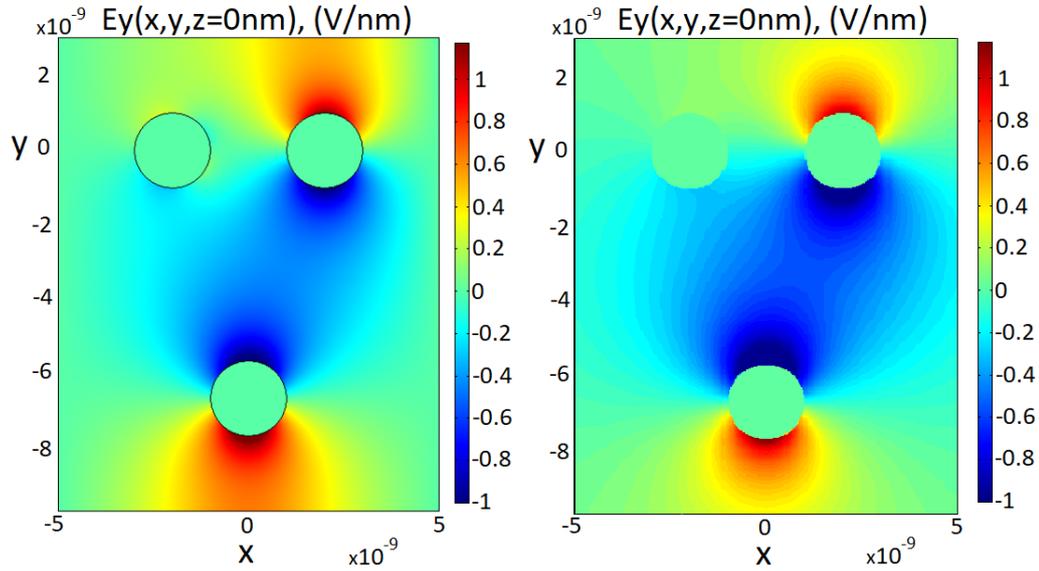


Figure 4.26: Comparison between Cmsol and Matlab by imposing the '0' static logic through the Hold+Driver mode considering the $E_y(x,y,z=0nm)$.

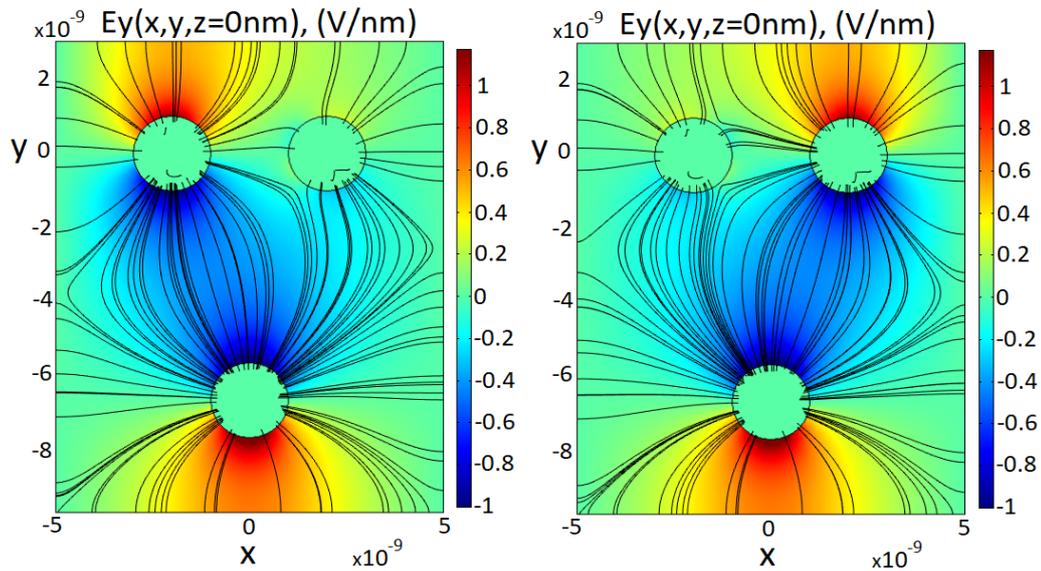


Figure 4.27: Highlight of the field lines in the Driver+Hold mode: on the left the '1' static logic is imposed while on the right the '0'.

By looking at the figure 4.27, which is shown above, it is very interesting to highlight

how the field lines move inside the structure by following the imposed input and this demonstrates that there is a polarization of the field.

The last important comparison, before to go on, is the electric field E_y presents along the plane $y-z$, for $x=0\text{nm}$, which cuts the structure in the middle, that highlights the vertical and the longitudinal trend, by continuing to impose the '1' static logic .

So, fortunately, also in this case, like it is possible to appreciate by looking at the figure 4.28, a strong agreement has been reached between the numerical simulator and the implemented model.

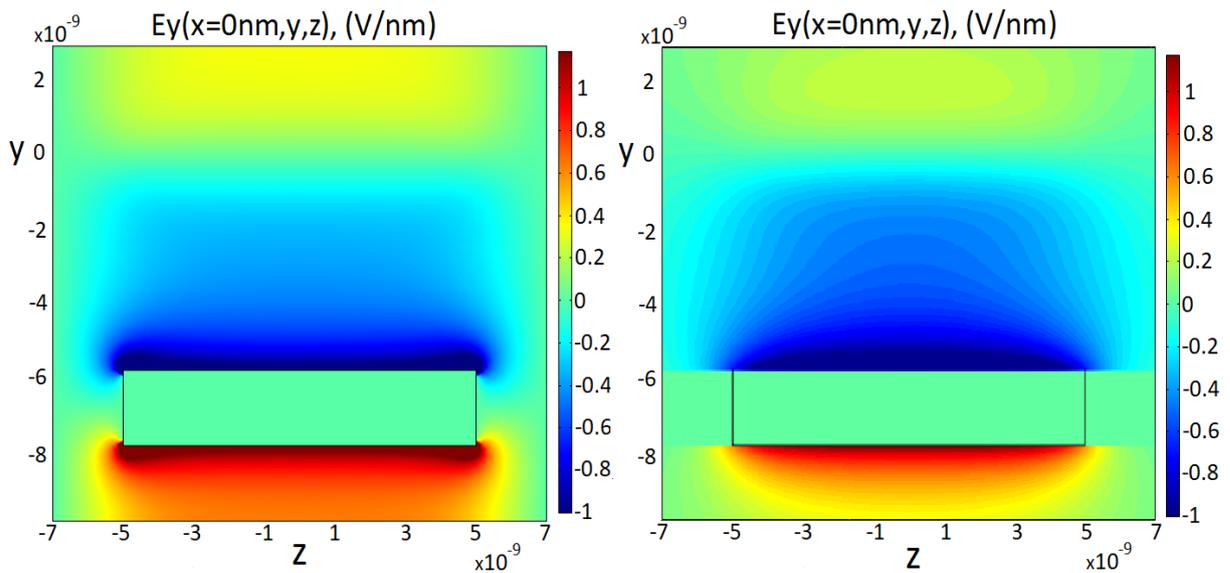


Figure 4.28: Comparison between Comsol (on the left) and Matlab (on the right) by imposing the '1' static logic with Hold+Driver mode considering the $E_y(x=0\text{nm}, y, z)$.

This model has been heavily analysed, and objectively, even if there are some problems, it is a very good starting point. Now different other models can be considered and the aim becomes to compare their results with that obtained from the finite wire model, in order to determine which gives the best solution and so, which may be implement on SCERPA, which is a molecular simulator where different integrated circuit, based, of course, on molecules, are realised like bus, basic logic gates, like AND, OR and inverter and others.

Chapter 5

Modelling of the electric field of a finite wire obtained with charges

By wanting to overcome the problems that have been highlighted in the previous chapter, which are, just to remember, the underestimation of linear charge along wires and the violation of the assumption $l \gg D_w$, for which the equations of finite wire model have been obtained, another alternative model has been implemented on Matlab.

The idea consists in building an electrode like a sum of aligned charges. Probably the right way to explain clearly what's happening is shown the structure.

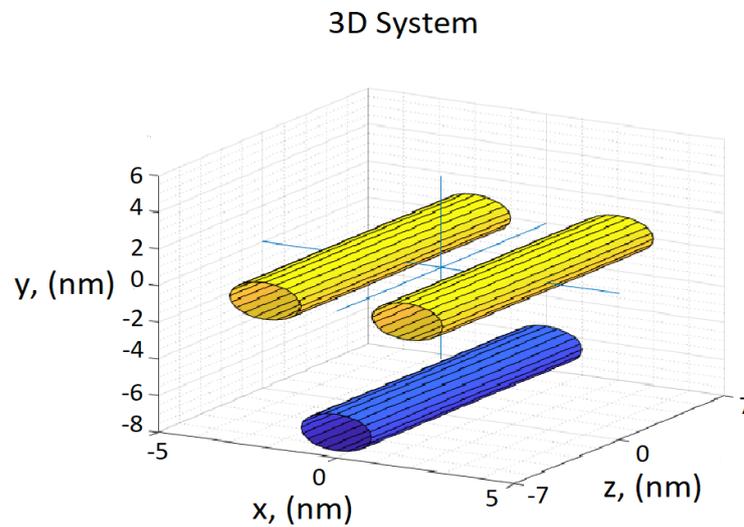


Figure 5.1: Reference structure from a 3D point of view.

Above, in the figure 5.1, there is a representation of the structure of electrodes from a 3D point of view, used just to remember what we are speaking about.

The aim consists in realising each single wire like a sum of different charges organized in slices and align each slice one after the other along the longitudinal direction.

A lot of different configurations can be used. As a consequence, the best one must be determined by taking into account different aspects like computation cost, complexity and the final resolution.

By wanting to simplify the explanation, it will show just the organization of one single slice for each studied configuration instead of the entire structure just to obtain cleaner figures.

5.1 Configuration number 1

The first configuration, which has been considered, is shown in the figure 5.2. The basic idea consists in building a grid of charges, and the model considers just that are inside the wanted shape of the electrode, which is the black profile in the image 5.2. In figure is very easy to understand which are inside since they are green, which means like they have passed the test. In this way, different shapes can be analysed, like a circular, rectangle shape or much others. Unfortunately, the results which have been obtained with it are so

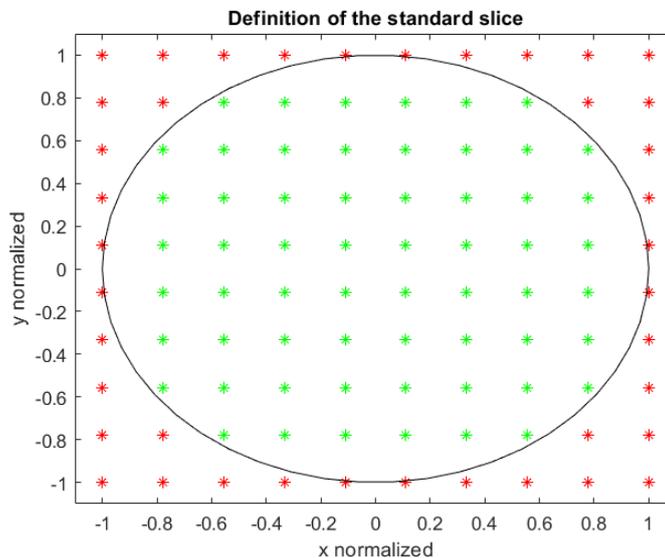


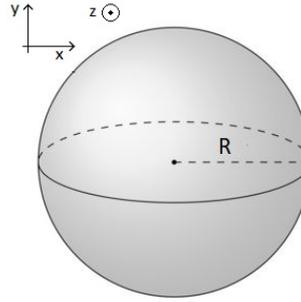
Figure 5.2: Grid of charges used to obtain a lot of possible shapes for electrodes.

far from the real case, and another important aspect is that there is an enormous impact

on the computation cost. As a consequence, this configuration has been abandoned and another has been considered.

Instead of considering just pointed charges, it could be useful, to find a trade-off between computational cost and final resolution, realise each wire like a sum of spheres.

First of all, let's remember which is the electric field and the potential for a sphere with a finite and non-zero radius.



By starting from the electric field, and by defining:

$$r = \sqrt{x^2 + y^2 + z^2}; \quad K_e = 1/(4\pi\epsilon_0);$$

$$\begin{cases} E = 0 & \text{for } r < R; \\ E = Q/(K_e R^2) & \text{for } r = R; \\ E = Q/(K_e r^2) & \text{for } r > R. \end{cases} \quad (5.1)$$

while the potential V is:

$$\begin{cases} V = Q/(K_e R) & \text{for } r \leq R; \\ V = Q/(K_e r) & \text{for } r > R. \end{cases} \quad (5.2)$$

where Q, of course, is the charge of the sphere.

5.2 Configuration number 2

In this second configuration, instead of considering a grid of charges, the single wire is obtained like a superposition of spheres.

Like it is possible to appreciate in the figure 5.3 below, different organizations have been proved, by increasing the number of spheres, for each slice, from two up to sixteen.

In the following section will be analysed and demonstrated which is the best, like a good scientist report must do. But what is intuitive is that the first organization, the one

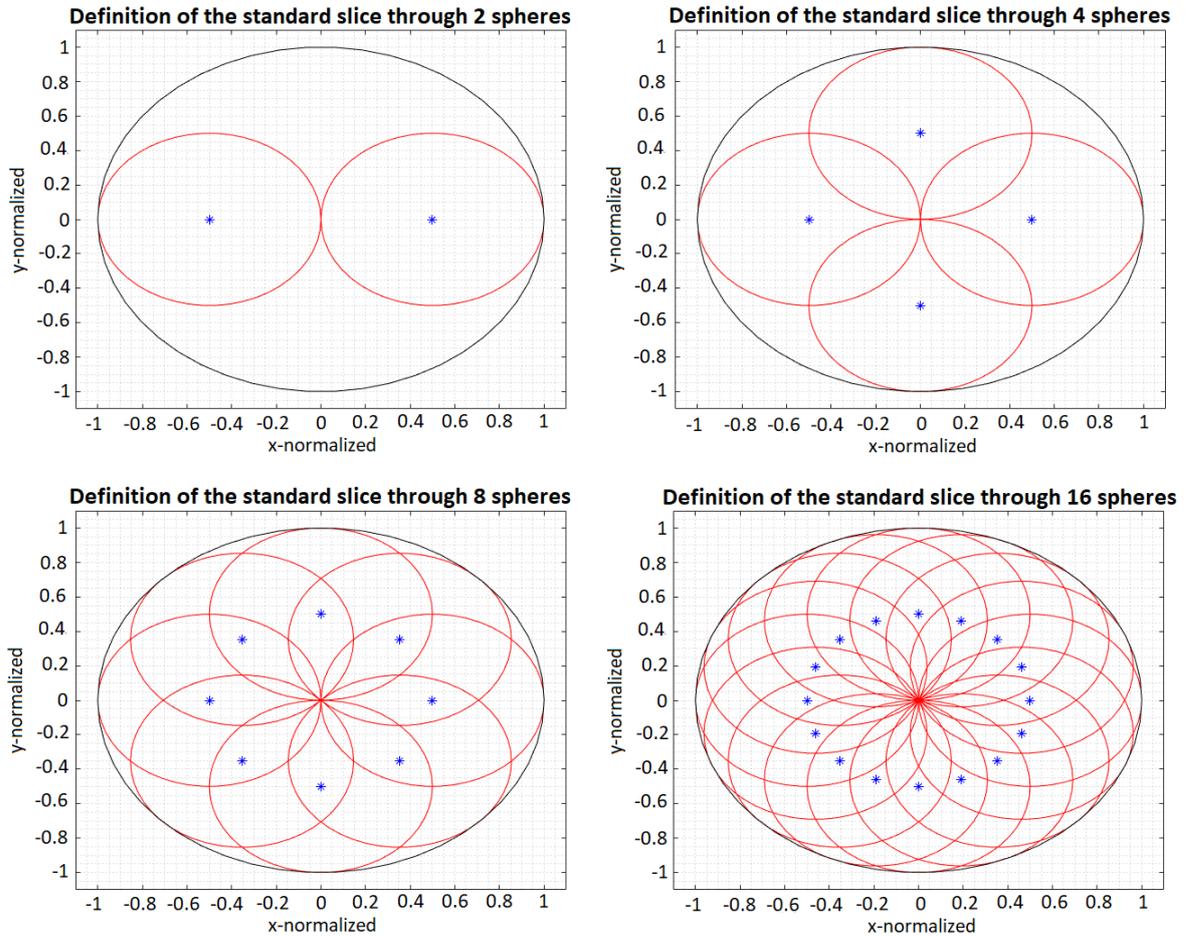


Figure 5.3: Different organizations which have been tested in order to determine the best model following the configuration 2.

with just two spheres, is the worst since it doesn't replicate a wire. Something better is obtained by using four spheres but the best are the configurations with 8 and 16 spheres, respectively looking at the down left and down right in the figure 5.3, where the regions which are not considered are very small. On the other hand, the last probably will have an important impact from a computational cost point of view. Anyway, all possible considerations will be explained below.

5.2.1 Charge evaluation for the charge model according to the configuration 2

Also, in this case, it becomes necessary to understand how to evaluate the amount of charge that the structure of electrodes has.

What has to be clear is that, even if spheres are used, the final shape is always a wire. As a consequence, the obtained results in the chapter 3, dedicated to the charge evaluation, are still true. It is necessary just to adapt these results to this case. As it has been done in the chapter 3, let's start from the driver mode.

5.2.1.1 Charge evaluation for the charge model(2) in the driver mode

With the driver mode, the aim is to impose the logic state. With the finire wire model the results were:

- **Logic Value '0'**: a negative voltage is applied, as a consequence what happens is:

$$* \lambda_L = -\frac{C}{L} \cdot |V_{Dri}|;$$

$$* \lambda_R = +\frac{C}{L} \cdot |V_{Dri}|;$$

- **Logic Value '1'**: a positive voltage is applied, as a consequence what happens is:

$$* \lambda_L = +\frac{C}{L} \cdot |V_{Dri}|;$$

$$* \lambda_R = -\frac{C}{L} \cdot |V_{Dri}|.$$

In the charge model, it is necessary to determine the charge for each sphere, but it's enough to evaluate the total amount of charge along the wire and not the linear charge, and then partition the amount of charge between the total number of spheres that build the wire.

The total number of spheres Q_N can be easily obtained like:

$$Q_N = Q_{\text{Slice}} \cdot \text{Slice}_N \quad (5.3)$$

where Slice_N is the number of slices which are used one in parallel to the other to build the wire and Q_{Slice} is the number of spheres that are present for each slice.

The idea is very simple, but just to clarify let's consider an example, which is shown in the figure 5.4, where there are three slices, so $\text{Slice}_N = 3$ where for each slice there is a number of spheres equal to $Q_{\text{Slice}} = 8$ by having on the whole $Q_N = 24$ spheres.

So, once that it has been determined, as a consequence, the charge for each sphere Q_i along wires can be easily evaluated like:

- **Logic Value '0'**: a negative voltage is applied, as a consequence what happens is:

$$* \lambda_L = -\frac{C}{L} \cdot |V_{Dri}|; \Rightarrow Q_L = \lambda_L \cdot l; \Rightarrow Q_{i_L} = \frac{Q_L}{Q_N};$$

$$* \lambda_R = +\frac{C}{L} \cdot |V_{Dri}|; \Rightarrow Q_R = \lambda_R \cdot l; \Rightarrow Q_{i_R} = \frac{Q_R}{Q_N}.$$

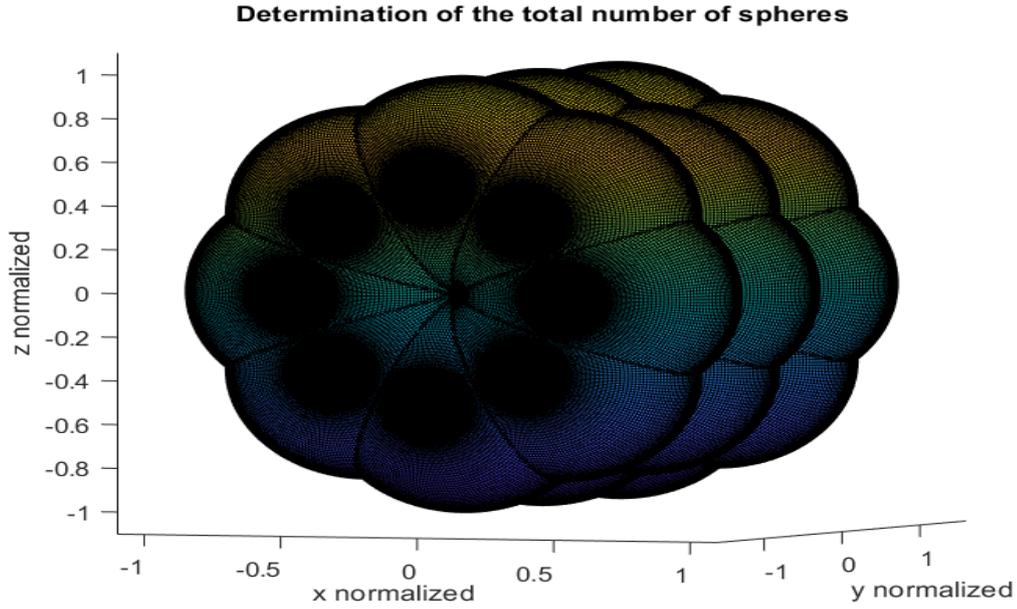


Figure 5.4: Example of how to evaluate the total amount of spheres which build a wire.

- **Logic Value '1'**: a positive voltage is applied, as a consequence what happens is:

$$\begin{aligned}
 * \lambda_L &= +\frac{C}{L} \cdot |V_{Dri}|; \Rightarrow Q_L = \lambda_L \cdot l; \Rightarrow Q_{i_L} = \frac{Q_L}{Q_N}; \\
 * \lambda_R &= -\frac{C}{L} \cdot |V_{Dri}|; \Rightarrow Q_R = \lambda_R \cdot l; \Rightarrow Q_{i_R} = \frac{Q_R}{Q_N}.
 \end{aligned}$$

where Q_{i_L} , Q_{i_R} , Q_{i_C} represent the charge for each sphere which build the relative electrode. Now that everything has been explained, it's able to consider quickly others modes, which are the hold, reset and hold plus driver modes.

5.2.1.2 Charge evaluation for the charge model(2) in the hold mode

In this case, the new results obtained by combining the consideration got from the finite wire model and the charge model according to the configuration number 2 are:

$$\begin{aligned}
 * \lambda_L &= +\frac{C_{LC}}{L} \cdot |V_{Cik}|; \Rightarrow Q_L = \lambda_L \cdot l; \Rightarrow Q_{i_L} = \frac{Q_L}{Q_N}; \\
 * \lambda_R &= +\frac{C_{RC}}{L} \cdot |V_{Cik}|; \Rightarrow Q_R = \lambda_R \cdot l; \Rightarrow Q_{i_R} = \frac{Q_R}{Q_N}; \\
 * \lambda_C &= -(|\lambda_L| + |\lambda_R|); \Rightarrow Q_C = \lambda_C \cdot l; \Rightarrow Q_{i_C} = \frac{Q_C}{Q_N}.
 \end{aligned}$$

5.2.1.3 Charge evaluation for the charge model(2) in the reset mode

And the same thing happens for the reset mode:

$$\begin{aligned}
 * \lambda_L &= -\frac{C_{LC}}{L} \cdot |V_{Clk}|; \Rightarrow Q_L = \lambda_L \cdot l; \Rightarrow Q_{iL} = \frac{Q_L}{Q_N}; \\
 * \lambda_R &= -\frac{C_{RC}}{L} \cdot |V_{Clk}|; \Rightarrow Q_R = \lambda_R \cdot l; \Rightarrow Q_{iR} = \frac{Q_R}{Q_N}; \\
 * \lambda_C &= +(|\lambda_L| + |\lambda_R|); \Rightarrow Q_C = \lambda_C \cdot l; \Rightarrow Q_{iC} = \frac{Q_C}{Q_N}.
 \end{aligned}$$

And analogous for the hold plus driver mode. Now that the electric field and the charge have been determined, all informations are known in order to move towards the simulation step and verify which model is the best, by comparing the results also with the finite wire model.

So, it will be important to understand if the computation cost that we pay inside the algorithm is useful or useless.

5.2.2 Verification of the charge model(2)

Also in this case, with the charge model, to verify the obtained results concerning Comsol, the same dimension and reference axes have been used, like described in the chapter 4. By wanting to compare which is the best and so, to determine which model can be considered the stablest, the same tests have been done.

5.2.2.1 Ex(x,y,z) for charge model(2) in driver mode

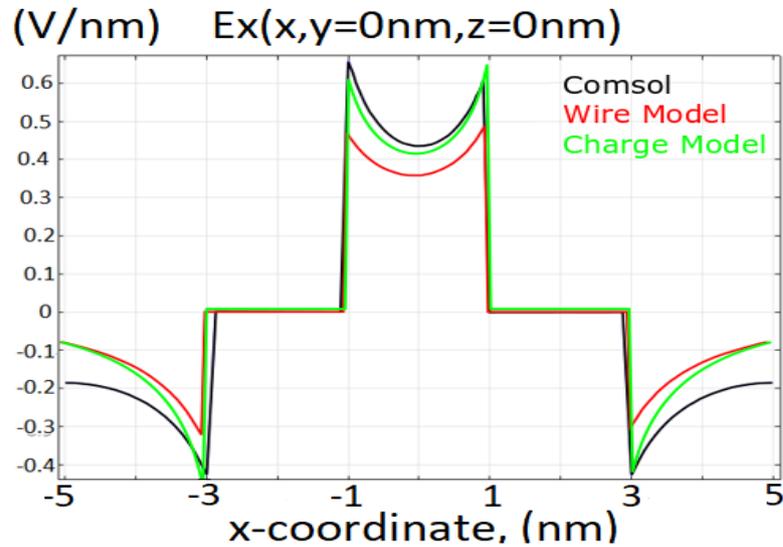


Figure 5.5: Comparison between Matlab and Comsol about $E_x(x,y=0nm,z=0nm)$ by forcing the static logic '1' with the charge model(2).

From a theoretical point of view, of course, nothing changes, in the sense that the driver mode is used to impose the logic state from the driver to molecules. In this case, like it's possible to appreciate in the figure 5.5, and coherently in the figure 5.6, there is a very important improvement, since in the figure 5.5 is almost impossible to distinguish the result obtained from Comsol and MatLab, while, on the other hand, by looking at the figure 5.6, there is practically a perfect match between the scales of the two plots. As a consequence, it's able to understand that it's a very strong and stable starting point for this configuration number 2.

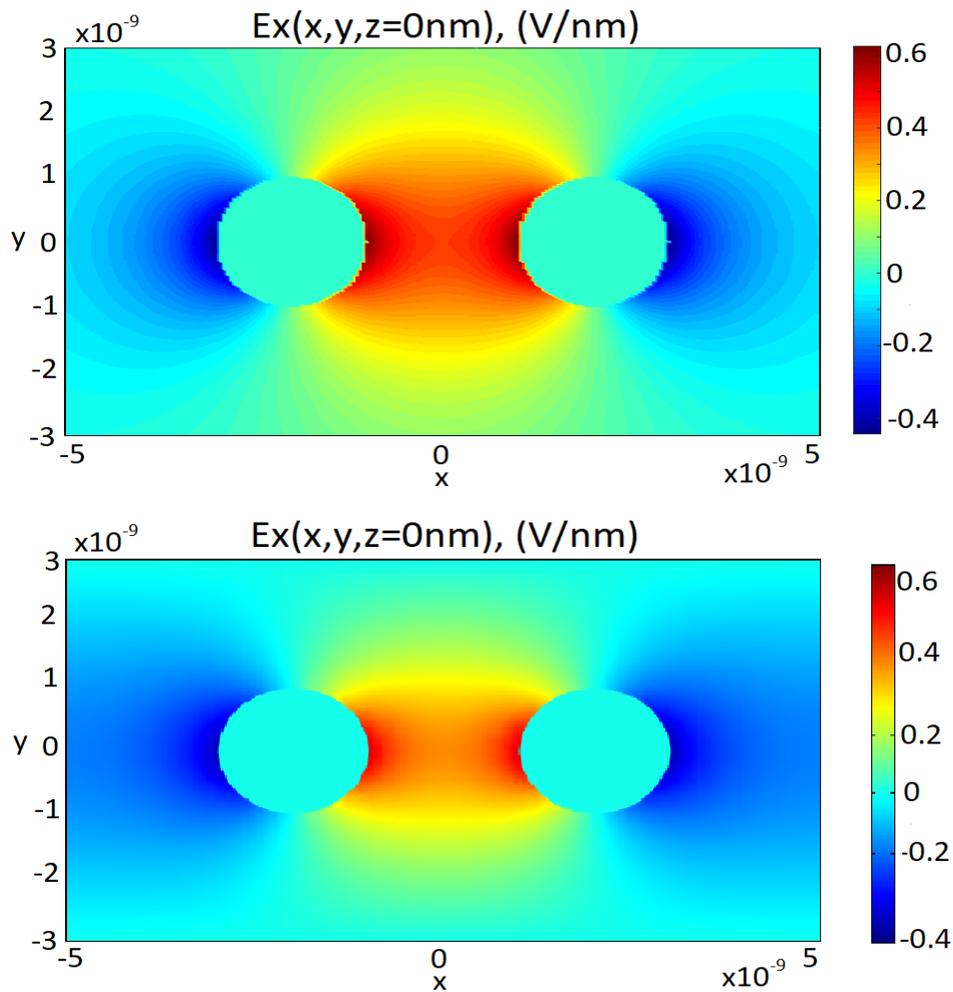


Figure 5.6: Comparison between Matlab (on the top) and Comsol (on the bottom) about $E_x(x, y, z=0\text{nm})$ by forcing the static logic '1' with the charge model(2).

Below, in the figure 5.7, it is shown the electric field by imposing the '0' logical value and also, in this case, a fantastic resolution has been reached. This results have been

obtained by using 8 sphere to build the wires, since it has been discovered that the optimal is when the number of spheres, for each slice, is greater, or equal, than eight, and then, after this value, the resolution doesn't improve so much with respect to the paid computational cost. As a consequence, from this point up to the end, for the configuration number 2, just this case will be analysed.

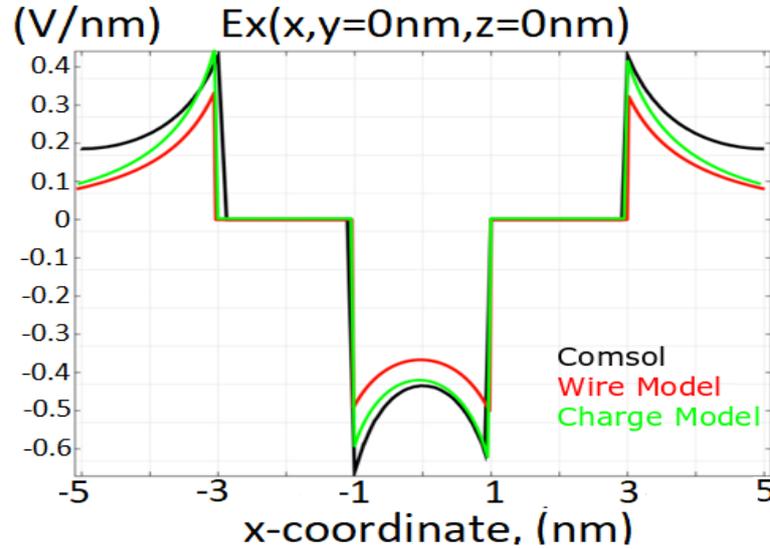


Figure 5.7: Comparison between Matlab and Cmsol about $E_x(x, y=0\text{nm}, z=0\text{nm})$ by forcing the static logic '0' with the charge model(2).

What's clear is that, in the first approximation, the problem related to the underestimation of the capacitance seems to be solved. As a consequence, let's move over and look at the plot that highlights the question related to the length: $l \gg D_w$. By looking at the figure 5.8, which is shown below, it is evident that there is, also for this case, a problem for the longitudinal direction, even if, with this configuration, is smaller than the result obtained from the finite wire model. As a consequence, this configuration is not yet the final but, as it will be explained in the section dedicated to the considerations of the configuration number 2, it will be the base for last working configuration number 3.

Now, like it has been done for the finite wire model, let's continue with the examination of this configuration by looking at other components of the electric field, to cover all possible cases and modes. So, as usual, after the horizontal component of the electric field in the driver mode, let's go on with the same mode but looking at the vertical component.

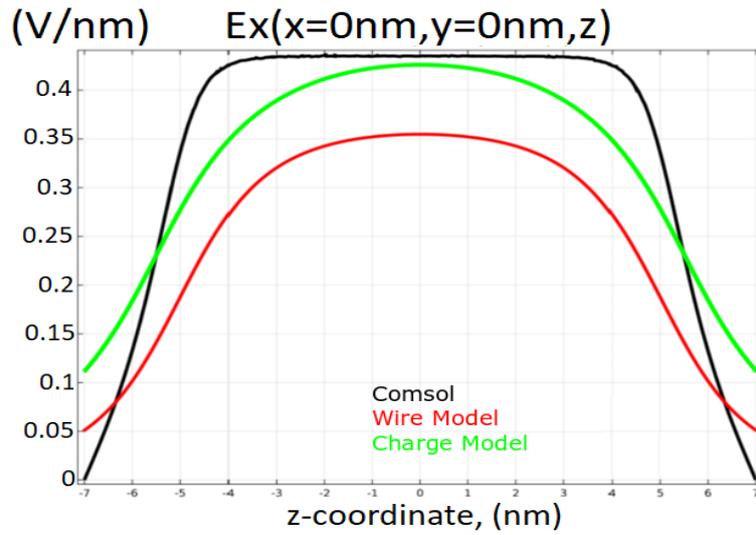


Figure 5.8: Comparison between Matlab and Cmsol about $E_x(x=0\text{nm},y=0\text{nm},z)$ by forcing the static logic '1' with the charge model(2).

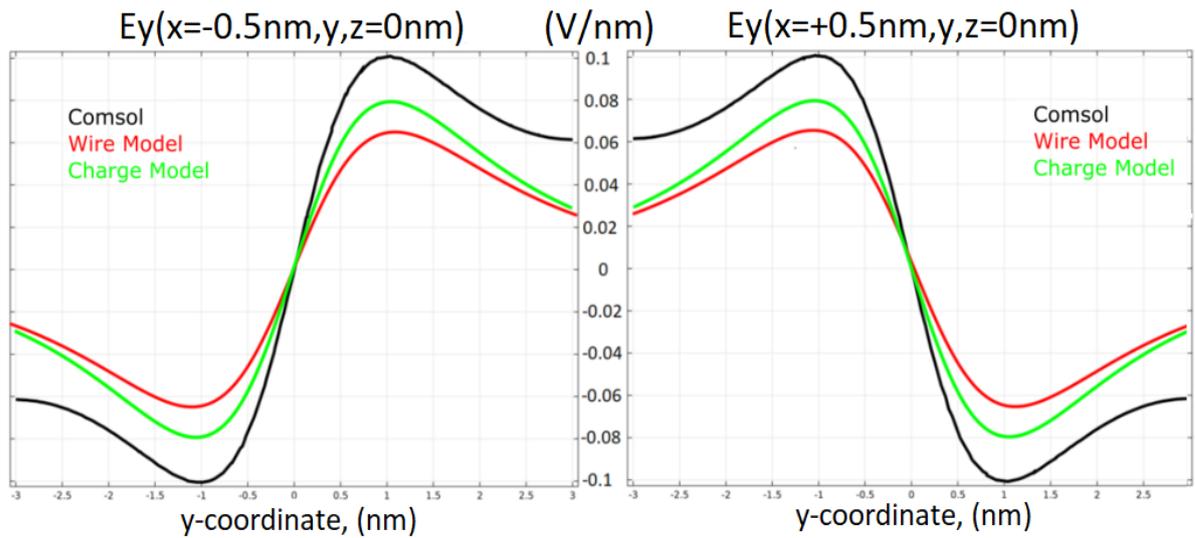


Figure 5.9: Comparison between Matlab and Cmsol about $E_y(x=-0.5\text{nm},y,z=0\text{nm})$ (on the left) and $E_y(x=+0.5\text{nm},y,z=0\text{nm})$ (on the right) by forcing the static logic '1' with the charge model(2).

5.2.2.2 $E_y(x,y,z)$ for charge model(2) in driver mode

As we know, in the driver mode, the vertical field is negligible since there is a compensation effect like it's been explanted in the chapter 4. Anyway, just to highlight how the

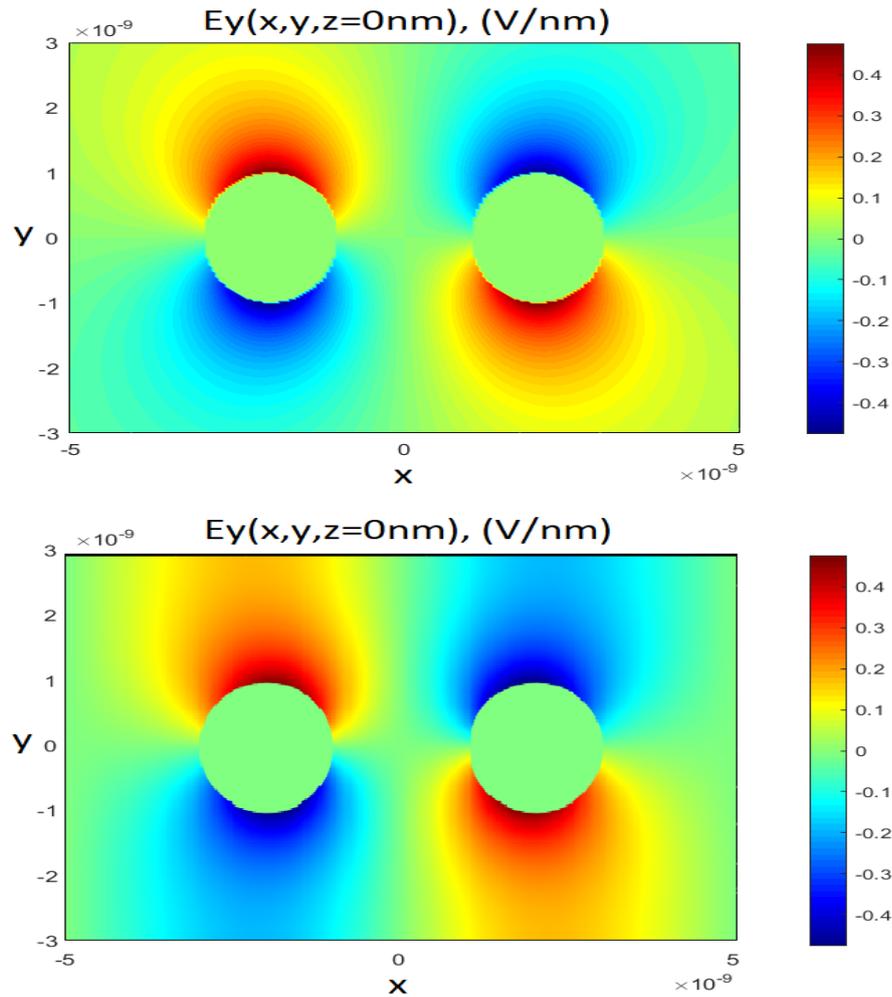


Figure 5.10: Comparison between Matlab (on the top) and Comsol (on the bottom) about $E_y(x, y, z=0\text{nm})$ by forcing the static logic '1' with the charge model(2).

new configuration works, it's interesting to notice that, also for the vertical component of the electric field, there is an improvement with respect to the finite wire model, like it is possible to appreciate in the figure 5.9.

This is another point in favour of this configuration, which is more evident also by looking at the figure 5.10, which is shown above. At this point, as a consequence, we can go on by considering the others important modes, which are the hold and the reset, that are, just to remember, used to help the propagation of the information along structures, like wire or complex gates.

5.2.2.3 $E_y(x,y,z)$ for charge model(2) in hold mode

By continuing with the hold mode, which is used, just to remember, to push towards up the electrons of molecules, it's possible to appreciate that, by looking at the figure 5.11, also with this configuration there is a very good agreement even if, like it is demonstrated in the figure 5.12, for $z=0\text{nm}$, which is the point for which we are in the middle of the structure, the electric field is, in this case, overestimated with respect to the reference value obtained from Comsol. This is not a problem but is a consequence of how the charge has been partitioned along the wires.

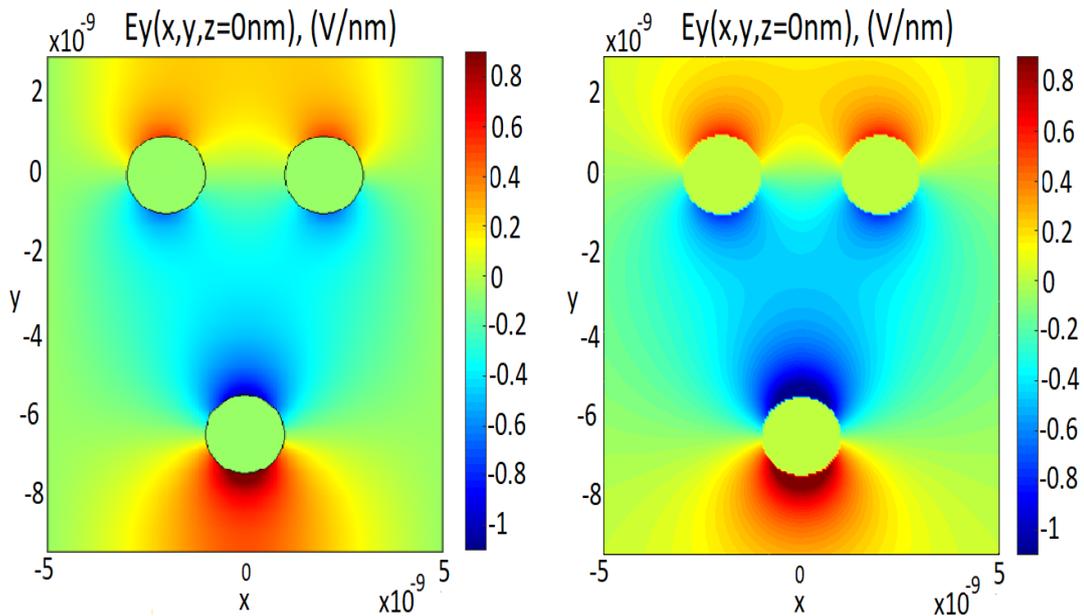


Figure 5.11: Comparison between Comsol (on the left) and Matlab (on the right) about $E_y(x,y,z=0\text{nm})$ by forcing hold mode with the charge model(2).

It's interesting to notice that, fortunately, every time that news aspects and method are used, there is an increase of the resolution, by going to reduce the total error of the implemented model on Matlab concerning the numerical solution got from Comsol. So, this is highlighted not only from the shown plots and figures but also from the scales that converge towards the same range. From it, we can conclude that this method works, even if not perfectly, but thanks to it and its final considerations, the last working configuration has been obtained to overcome and solve all problems that have been discovered along this report. Unfortunately, also in this case, the problem regards to $l \gg D_w$ is practically unchanged.

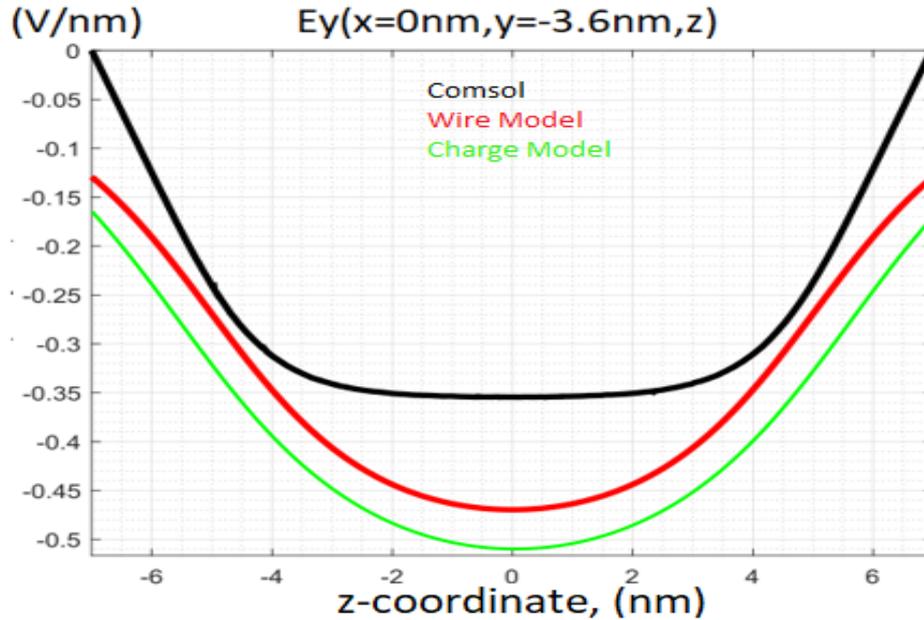


Figure 5.12: Comparison between Comsol and Matlab about $E_y(x=0\text{nm}, y=-3.6\text{nm}, z=0\text{nm})$ by forcing hold mode with the charge model(2).

5.2.2.4 $E_y(x,y,z)$ for charge model(2) in reset mode

By looking at the figure 5.13 the same happens, of course, with the reset mode, since the only difference is the sign of applied voltage, V_{clk} , while the module doesn't change.

For both two cases, hold and reset mode, anyway, the electric field has the same behaviour, which was not absolutely predictable. Remember that the aim is to realise a model with, of course, its tolerance. Anyway, as it will be shown later, better results will be obtained with the last configuration, number 3, where, by combining different degrees of freedom, an incredible resolution will be got, by obtaining a model which follows perfectly the results obtained from Comsol.

Notice that, up to this point, every time that comparisons about the hold and reset mode, between Matlab and Comsol, have been done, a $V_{\text{clk}} = \pm 2\text{V}$ has been used, even if the optimal is $\pm 8\text{V}$ [4]. Of course, this is not a problem since, up to this point, the aim was just to verify the validity of the model and not the effective electric field necessary to polarise or drive the molecule. For the last working configuration, instead, the typical value $\pm 8\text{V}$ will be used in order to be ready to implement this algorithm on SCERPA and have a powerful control of the structure, molecules' layout and electric field on them, in order to do a step towards a physical implementation of this new technology.

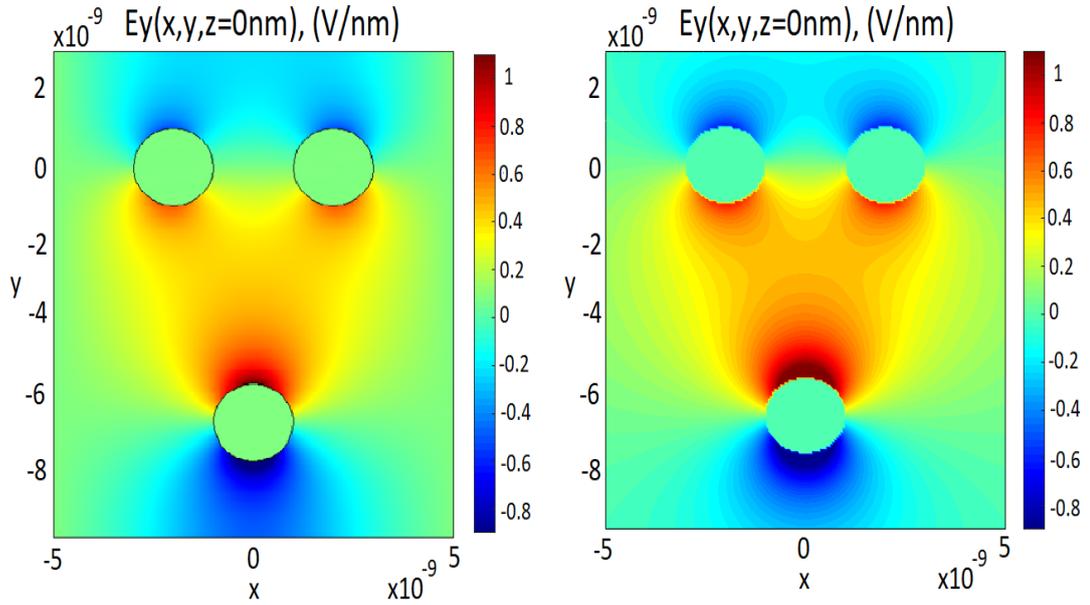


Figure 5.13: Comparison between Cmsol (on the left) and Matlab (on the right) about $E_y(x,y,z=0\text{nm})$ by forcing reset mode with the charge model(2).

5.2.3 Final consideration about the configuration number 2

A lot of times references to the computational cost have been mentioned, so it becomes almost mandatory to give some data to understand which is the impact of this configuration on the algorithm. In the table 5.1, there are times necessary to evaluate the electric field inside the volume with dimension specified in the chapter 4 for a structure of electrodes with a length, for each wire, equal to 10 nm.

Q_{Slice}	Time, (sec)
2	44
4	88
8	175
16	350

Table 5.1: Computational cost for the charge model(2) following the configuration number 2.

To comprehend the impact of this model, it's enough to think that, for the finite wire model, the cost for each structure is, more or less, three seconds. By considering that the only two admitted configurations are with eight and sixteen spheres, there is an increase of two orders of magnitude, so, not absolutely negligible.

It's important to highlight this aspect since, to realise a complex circuit, probably a lot of structure of electrodes have to be considered, and so, in the end, the simulation time explodes.

What we can conclude from this model is that, first of all, by playing with the charge distribution on the spheres, a very good result can be obtained. Then, to reduce the computational cost, by keeping unchanged the resolution, it's possible to reduce the volume for which the electric field is evaluated, by considering just the volume where we expect to place, of course with tolerance, molecules and reduce the total amount of spheres that build wires.

5.3 Configuration number 3

Finally, it's time to explain the power of this last configuration, the number 3, that a lot of times has been mentioned like the solver.

Before to explain how it works, it is very important to highlight what this configuration has to solve:

- reduce, as much as possible, the computational cost to obtain a faster algorithm able to realise simulation of complex circuits fastly;
- solve the last remaining problem, which hasn't been solved from the configuration number 2, which is $l \gg D_w$.

Like always happens, before to go on, it could be much better to show what we are speaking about to simplify.

By looking at the figure 5.14, which is shown below, and in the particular on the left, it is possible to appreciate one of the two basics principles of this configuration, that is, instead of realise each slice like a sum of a certain amount of spheres, there are just two plugs, one at the beginning and the other at the end, and in the centre of the wire, to build the body of it, just big spheres are used, where big means that they have a radius equal to the radius of the wanted wire.

The reason, for which plugs are used, is since, in this way, at the boundary, they represent better a wire, instead of use just big spheres. As a consequence, a much better electric field is got by implementing them in the algorithm. By wanting to understand which is the impact on the computational cost, it is enough trying to count the number of spheres, which are present on the left of the figure 5.14, concerning the case, that would be present, if the configuration number 2 is used.

By considering the figure mentioned before, 5.14, the number of slices are nineteen plus two plugs with eight spheres for each of them, as a consequence, the total number

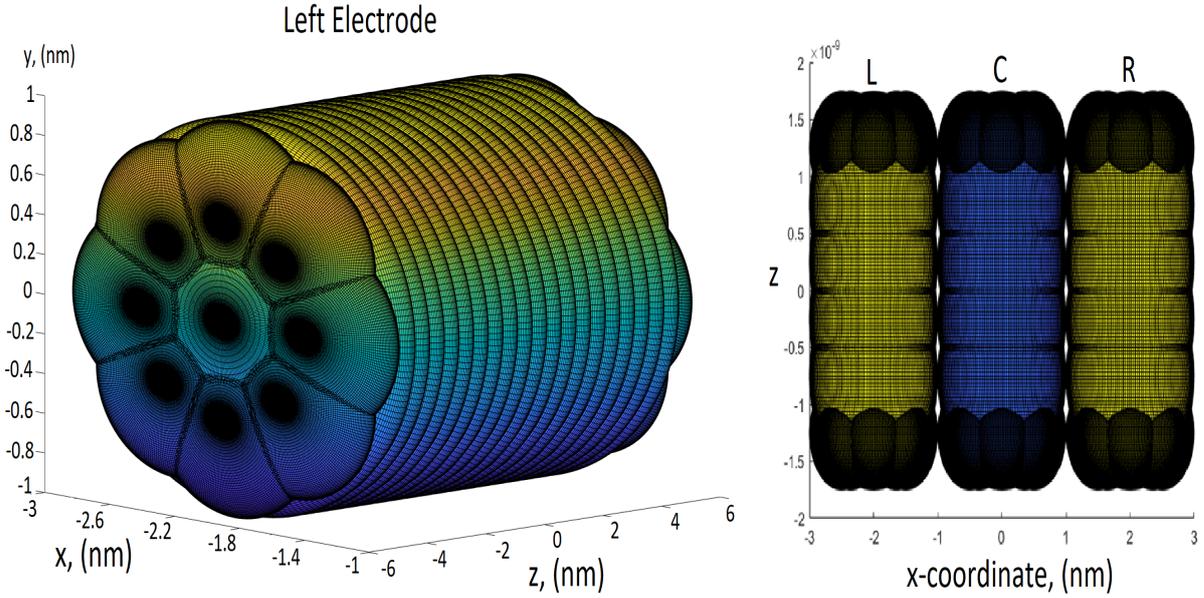


Figure 5.14: Configuration number 3, on the left there is a real case of how one electrode of 10 nm is modelled, while on the right there is top view of the entire 3D structure

of spheres, for each electrode, by continuing to use the same notation explained for the equation 5.3, is equal to

$$Q_N = \text{Slice}_N + 2 \cdot Q_{\text{plugs}} = 19 + 2 \cdot 8 = 35 \text{ spheres} \quad (5.4)$$

While with the configuration number 2, by having eight spheres for each slices, the final number would be:

$$Q_N = 8 \cdot \text{Slice}_N = 8 \cdot 19 = 168 \text{ spheres} \quad (5.5)$$

So an incredible reduction of 75%, more or less, for each electrode, which means one order of magnitude for the computation cost!

On the right of the figure 5.14, instead, is shown a top view of an example for the entire structure, where it's possible to see that each wire follows the right organization of the configuration number 3, so the body of wire between the two plugs.

At this point, one of the two aims has been reached. Now, it's time to move over and try to solve the problem related to $l \gg D_w$. Independently on which field we consider, along the longitudinal direction, there is always a parabolic effect instead of a flat field up to the end of wires. By wanting to help the description of this scenario, it could be useful to look at the figure 5.8. The idea, to overcome this problem, consists in adding some charge, used to realise a sort of compensation, to get the wanted results in black. Now, instead, to add physically news spheres, it is much better to use spheres that are already

used to realise the wire but playing with the charge partition.

By wanting to determine the charge along each wire, an analogous method used with the configuration number 2 has to be re-used with, of course, some adjustment.

5.3.1 Charge evaluation for the charge model according to the configuration 3

Like in the configuration number 2, it's enough to adapt the result get from the chapter 3. In this way, again, the total amount of charge along each wire can be evaluated by multiplying the charge per unit length and the length of each wire. At this point, the new partition method starts.

First of all, it's necessary to determine the total number of spheres Q_N which build the wires, by using exactly the equation 5.4, like the sum of all bigger spheres that realise the body of electrodes and the two plugs. As a consequence, we can determine the charge for each slice, and, the charge of each sphere since for this configuration, in the body, one slice corresponds to one sphere.

Then it remains to understand how to partition the charge in the two plugs, and so, the charge for each sphere in the plugs is equal to the charge of the slice divides from total number of charges in the plugs, which are, in our case, eight. So, up to this point, there is nothing new, since the method is similar to the configuration number 2 by changing just the way of how each slice is realised.

The powerful tool, which is used in this charge model(3), consists of realising and developing a compensation algorithm used to solve the problem regards to $l \gg D_w$. The first step consists of the partition mentioned before, then the second step is performed to adapt the charge along the wire in the right way to get a good final electric field. By wanting to help the explanation, the core of the code for this part is shown, by highlighting also the used degrees of freedom.

Like it is possible to appreciate below, there are three different degrees of freedom:

- n , which is the position of the spheres, in the beginning and in the end, where the charge is increase through the "alpha_external";
- α_{external} is the coefficient used for the spheres mentioned before for n , where the aim is to increase the charge of these spheres,
- α_{center} is the coefficient used to reduce the value of the charge of the sphere in the middle of the wire, in order to compensate the increase due to α_{external} of the sphere in the position n .

```

%charge compensation
%degrees of freedom
%           n
%           alpha_external
%           alpha_center

if (zz==1+n || zz == length(Dz)-n)

    %compensation at the beginning of the wire
    %and at the end of it

    for te=1:pnt_circon
        charge_sphere(te).qi=QiL*alpha_external;
    end

elseif (zz==(length(Dz)+1)/2)

    %compensation in the middle of the wire

    for te=1:pnt_circon
        charge_sphere(te).qi=QiL*alpha_center;
    end

end
end

```

By combining these three elements, as will be shown below, fantastic results have been got. What's very interesting, and gives stability to the algorithm, is that the theoretical formulas used to get the capacitance, and the charge per unit length λ , don't change since they are always true. What changes is just how we decide to partition and manage the amount of charge to increase the final resolution. Now, it's true that, with this configuration, we lose in term of generality, since for each different length, a respective compensation has to be done but, it's necessary to accept it in order to obtain a solution in a closed-form with a very small tolerance, otherwise numerical solution must be used but this is completely in contradiction with what's the aim of this thesis. So, just to conclude, it's a typical trade-off.

5.3.2 Verification of the charge model(3)

Also in this case, with the charge model, to verify the obtained results concerning Comsol, the same dimension and reference axes have been used, like described in the chapter 4. To compare which is the best and so, to determine which model can be considered the

stablest, the same test have been done.

5.3.2.1 $E_x(x,y,z)$ for charge model (3) in the driver mode

As a tradition, after to have spoken about the charge distribution, let's go on whit the verification step by starting, like usually, from the driver mode, which is, of course, used to impose the logic state from the driver to the molecules. In this case, only results about the '1' logic state will be shown, since the behaviour concerning the '0' logic state doesn't change, but only the sign. Like it's possible to appreciate in the figure 5.15, in this case the best result is obtained. It's incredible that is also better than the configuration number 2, where there was already an improvement concerning the finite wire model. As a consequence, it's reasonable to consider destroyed the problem related to the underestimation of the capacitance. Now, it's time to try to solve, finally the problem related to

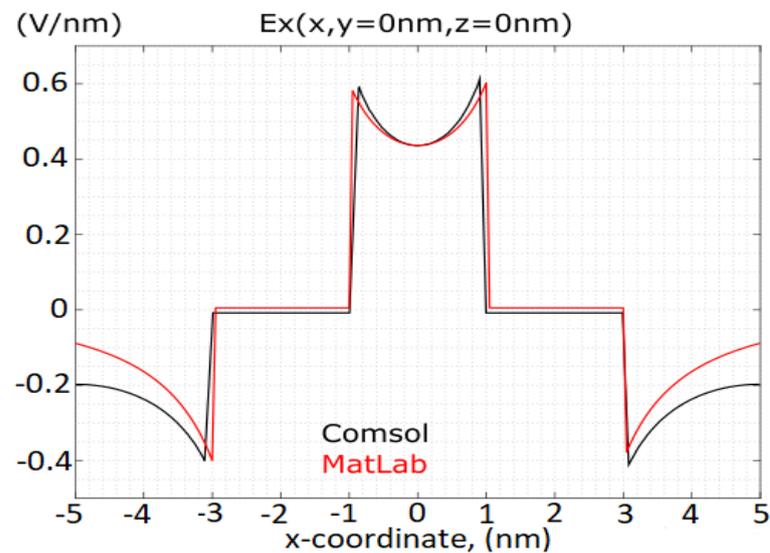


Figure 5.15: Comparison between Matlab and Comsol about $E_x(x,y=0nm,z=0nm)$ by forcing the static logic '1' with the charge model(3).

the $l \gg D_w$. By wanting to reach this aim, the charge of the spheres near to the end of electrodes has been increased and the charge of the sphere in the middle of the body has been reduced, by keeping, of course, the total amount of charge along the wire unchanged, like anticipated in the code shown above. The result is shown in the figure 5.16.

Finally, the goal has been reached. And, to highlight the result, it's incredible that practically is impossible to distinguish the two curves, the red from Matlab and the black from Comsol. This is a very strong important starting point for this configuration to go on with other cases.

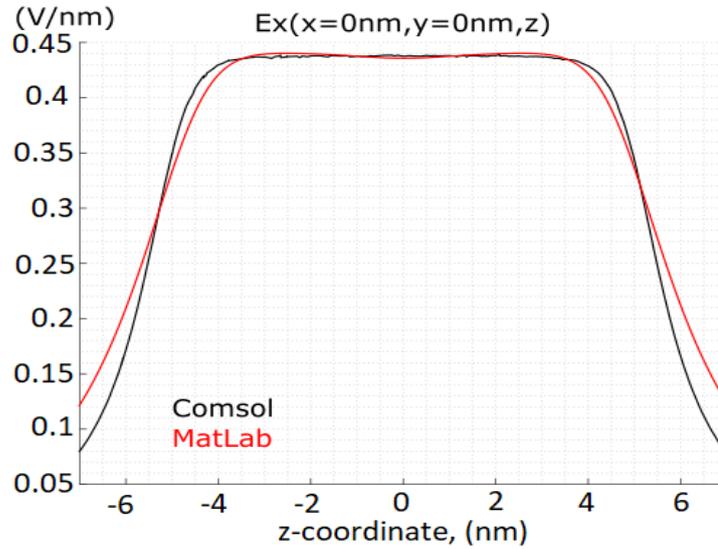


Figure 5.16: Comparison between Matlab and Comsol about $E_x(x=0\text{nm}, y=0\text{nm}, z)$ by forcing the static logic '1' with the charge model(3).

Once that the field for a structure with a length, for each wire, equal to 10 nm has been got, and it has been demonstrated that the used method works in a fantastic way, let's try to determine the electric field also for structure with other lengths, like, for instance, 20 and 30 nm, which are shown in the figure 5.17, respectively on the left and on the right.

So, by looking at all figures regard to the driver mode, which are shown above and the last below, we can consider concluded this part, since the model works very well, a not just for a specific case.

Notice that, this configuration has been optimized for wires with a length equal to 10, 20 and 30 nm. But, of course, this doesn't mean that the model doesn't work for other length, like for instance, 15, 25 and etcetera nm. The consequence is that instead to have a perfect model, there is a tolerance of, more or less, 5 %, which is, anyway, completely negligible.

At this point, let's go on with the test of this configuration by trying to see what happens for the hold and reset model. Since the aim is to realise an algorithm which will be fused with SCERPA, the simulation will be performed by imposing a $V_{\text{clk}} = \pm 8\text{V}$ [4], which is the potential necessary to have a vertical electric field $E_{\text{clk}} = \pm 2\text{V/nm}$ needed to drive towards up or down electrons inside molecules.

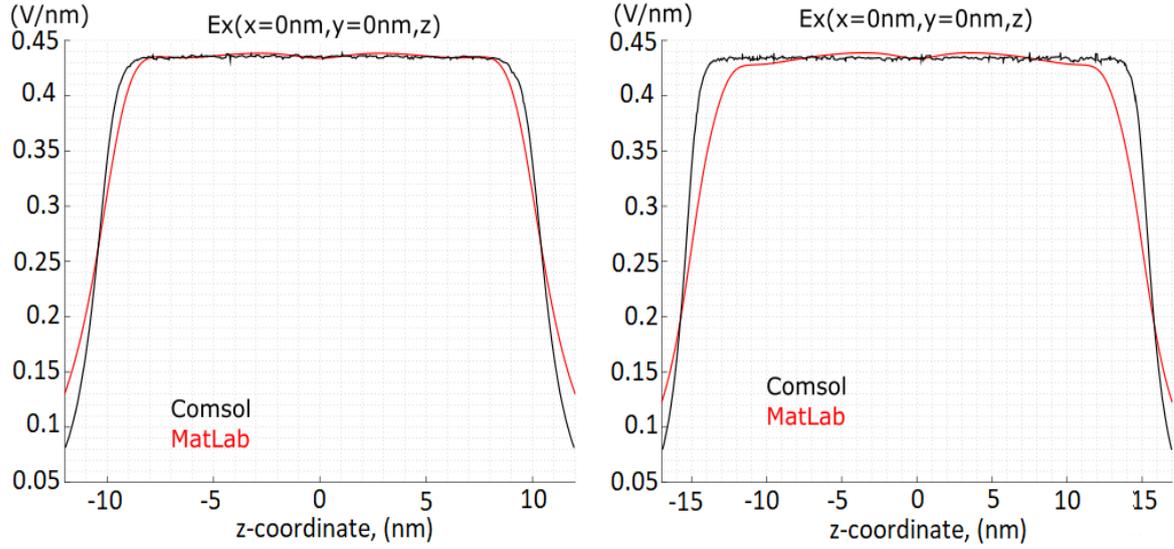


Figure 5.17: Comparison between Matlab and Comsol about $E_x(x=0\text{nm},y=0\text{nm},z)$ by forcing the static logic '1' with the charge model(3) for a structure with length's wire of 20 nm on the left and 30 on the right.

5.3.2.2 $E_y(x,y,z)$ for the charge model (3) in the reset mode

Since for the configuration 2, the hold mode has been used to verify the model, in this case, for completeness, the reset mode will be considered, since the only thing that changes is the sign, like a lot of times it has been highlighted. First of all, it's important to look at the plot shown in the figure 5.18, which is the vertical field in the middle of the wires, for $z=0\text{nm}$, and passing for the origin, $x=0\text{nm}$, along the vertical coordinate y . Also, in this case, the starting point is very promising since there is not practically different between results got from Matlab and Comsol.

From this simulation, a very important consideration must be done. By wanting to have a very good control on molecules, to guarantee the right propagation of the information along with complex structures, the field on molecules has to be, more or less, equal to $\pm 2\text{V/nm}$ in base of the fact that a hold or reset mode is imposed. Since, at this moment, the reset mode is considered, the value must be equal to $+2\text{V/nm}$. As a matter of fact, all molecules have to be placed in a coordinate y equal to -4.5 nm , more or less, which means distant about 1.5 nm from the clock electrode to be managed properly. Of course, this is just a theoretical sentence, since decide where to place molecules in a complex circuit with an incredible tolerance and the right orientation is, practically, impossible. We are able to conclude that the best place where to place the molecule is where the vertical electric field is flatter, so from -3.5 up to -2 nm , in order to relax the limits due to technological

process, but in this region there is a field equal to $\sim +1.2\text{V/nm}$ which is not enough to guarantee the right polarization. The aim of this thesis is just to describe in closed form the electric field in these type of structures, but anyway, who will continue this job, probably should take into account these considerations. A possible solution could be to increase the applied voltage V_{clk} to have $\pm 2\text{V/nm}$ in that region but for a digital circuit, 8V is already a huge value.

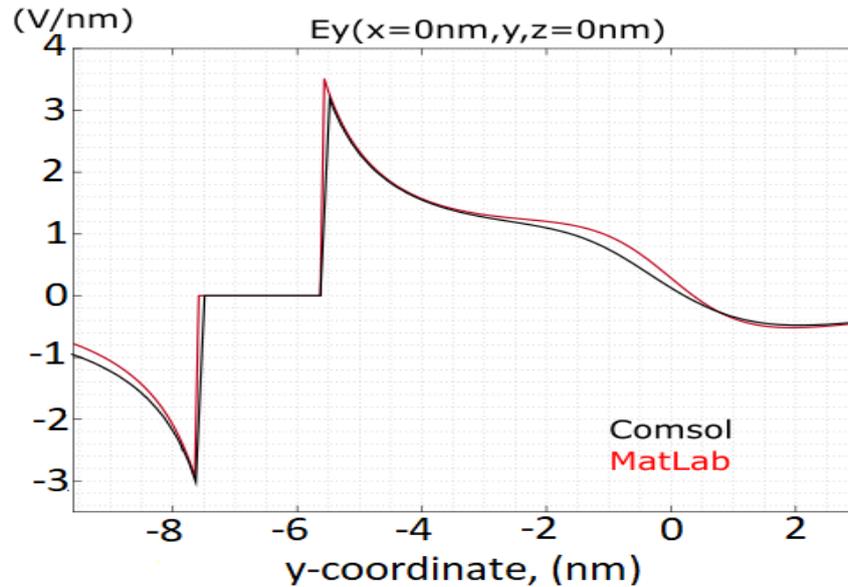


Figure 5.18: Comparison between Cmsol and Matlab about $E_y(x=0\text{nm}, y, z=0\text{nm})$ by forcing reset mode with the charge model (3).

Below in the figure 5.19, which is the usual plane that cuts the structure in the middle, along the longitudinal direction, it is shown the vertical electric field. Finally, by highlighting that for the two plots there is the same numerical scale, there is a perfect match between the numeric solution got from Cmsol and the implemented model in Matlab. To be precise, there is a different outside the region of the structure of electrodes, where the electric field is negative (region in blue) but this is not a problem since we are interesting to what happens where the molecules are placed, and so inside the structure of wires. At this point, let's go on and look at what happens along the longitudinal direction. Like it's possible to appreciate in the figure 5.20, also in this case the method used to solve the problem regards $l \gg D_w$ works in a fantastic way. Practically there is not difference, again. After it, we can be very proud since the aim has been reached. Since the idea is to get a model that works for different contexts, the method has been extended for other lengths, by optimizing the configuration for specific values.

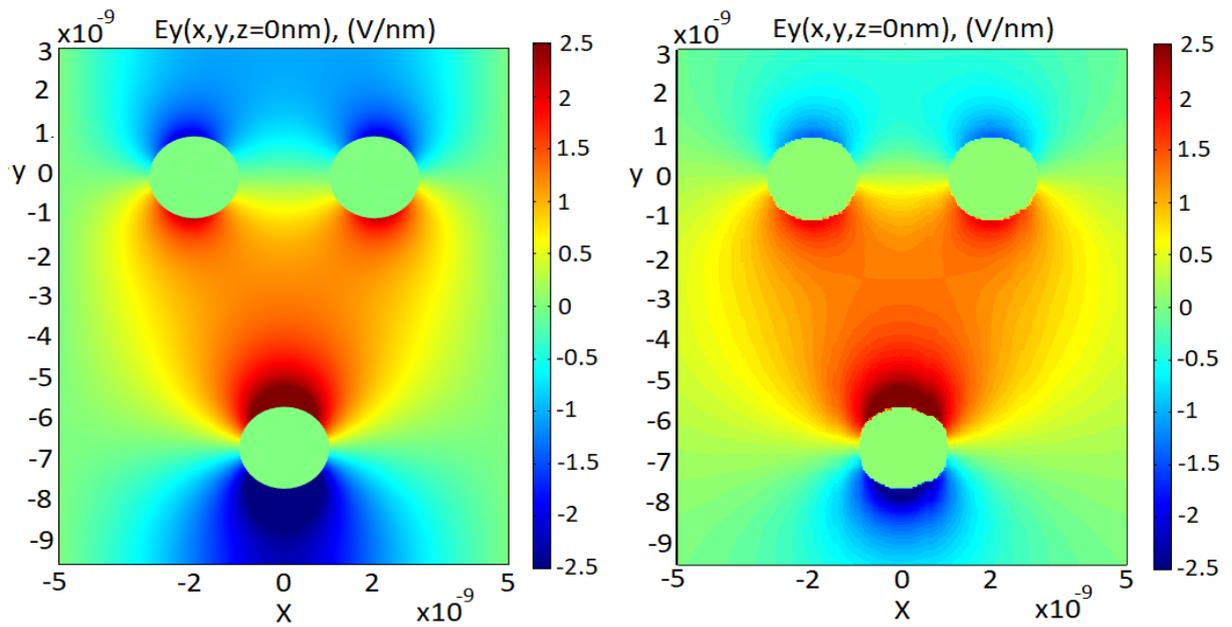


Figure 5.19: Comparison between Comsol (on the left) and Matlab (on the right) about $E_y(x,y,z=0\text{nm})$ by forcing reset mode with the charge model (3).

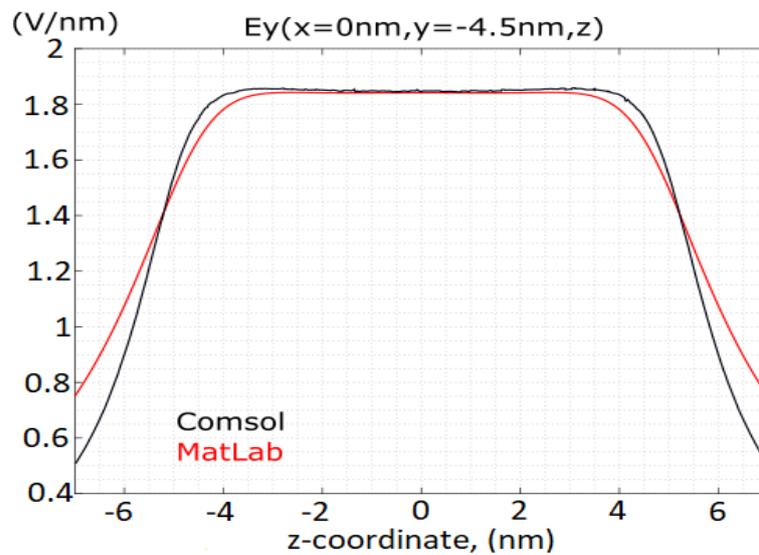


Figure 5.20: Comparison between Comsol and Matlab about $E_y(x=0\text{nm}, y=-4.5\text{nm}, z)$ by forcing reset mode with the charge model (3).

By looking at the figure 5.21, which is shown below, the results for different lengths have been obtained. In the top left, there is the comparison for a structure with a length equal to 5 nm, so very very short and complex to model, then on the top right 15 nm, while on the down left 20 nm and, in the end, in the down right 30 nm. For the clock mode, a bigger number of optimization has been done since this mode is used than the driver since it is used to help the propagation of the information along the circuit. Also for this case, other lengths can be considered, not just the optimized.

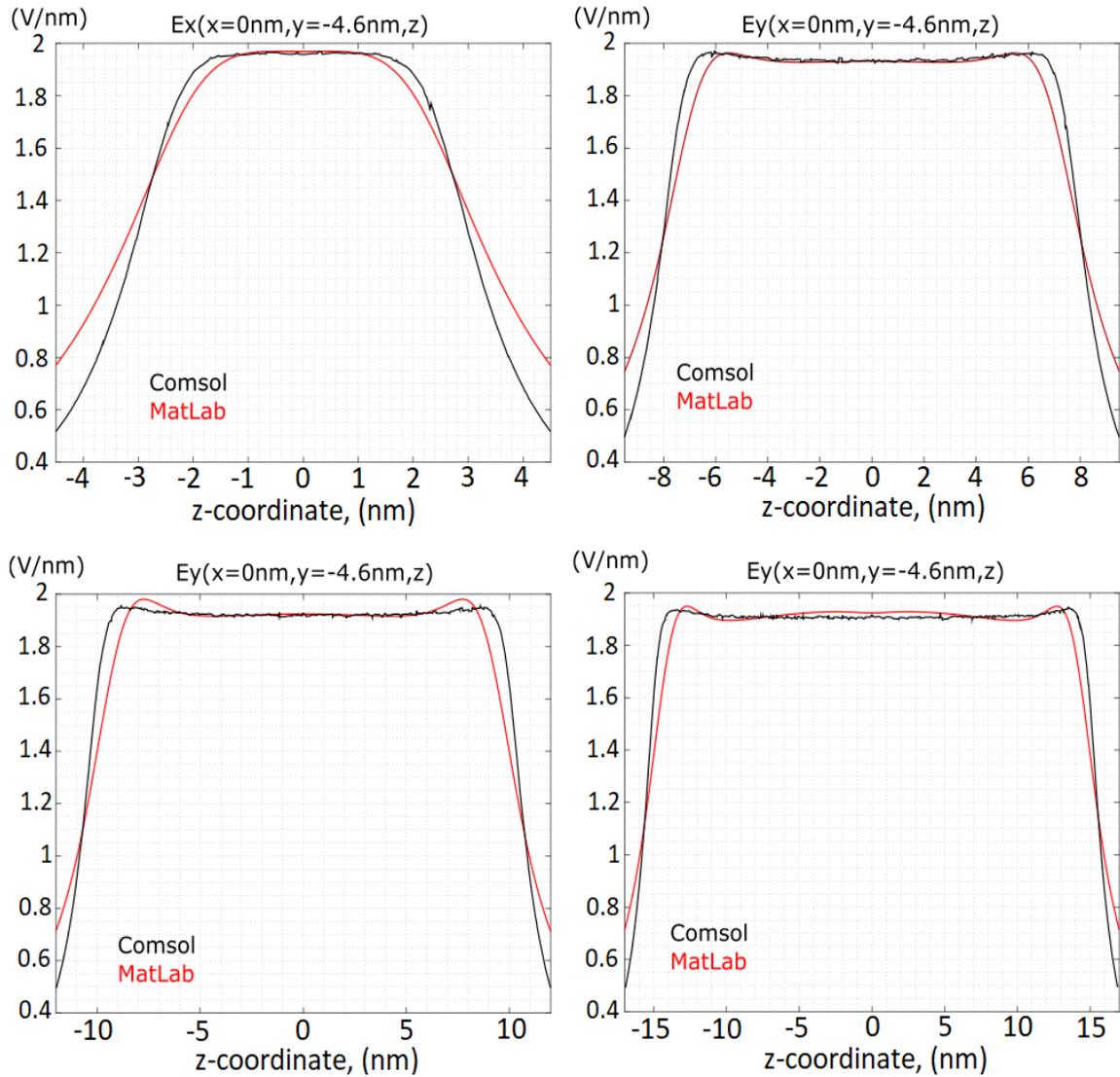


Figure 5.21: Comparison between Comsol (on the left) and Matlab (on the right) about $E_y(x=0\text{nm}, y=-4.6\text{nm}, z)$ by forcing reset mode with the charge model (3) for different lengths.

At this point, everything has been considered about the reset mode. For the hold mode is the same, for this reason, it will be shown just a plot of it, since it's a nonsense, at this point of the report, to show twice the same figures but with opposite sign.

5.3.2.3 $E_y(x,y,z)$ for the charge model (3) in the hold mode

Also for this case, a very precise result is obtained inside the structure while outside is worse, but it has been mentioned for the reset mode, this doesn't matter since what is important is to describe in a precise way the electric field inside the structure of electrodes where molecules are present.

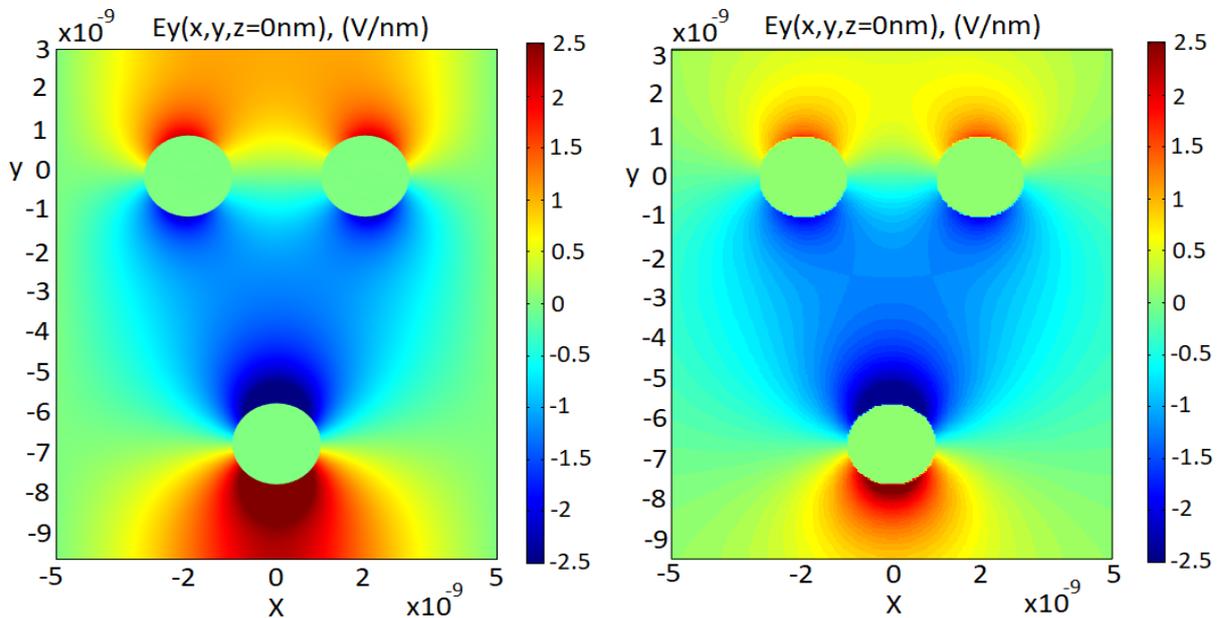


Figure 5.22: Comparison between Comsol (on the left) and Matlab (on the right) about $E_y(x,y,z=0\text{nm})$ by forcing hold mode with the charge model (3).

5.3.3 Parametric analysis for the configuration number 3

Since, like it has been demonstrated, the charge model(3) works very well, it becomes very interesting to determine its limitation by performing a parametric analysis on it. Up to this point, different geometrical parameters have been never changed, like for instance, D_v , which is the vertical distance between the upper electrodes and the clock electrode, D_w , which is the horizontal distance between the right and left electrodes and, finally, the applied voltages.

5.3.3.1 Parametric analysis for Dw

Let's start from the horizontal distance, by performing a sweep of Dw. In this case, the driver mode has been considered, since, in this case, the impact of Dw is much more evident. Like it is possible to appreciate in the figure 5.23, shown above, there is an

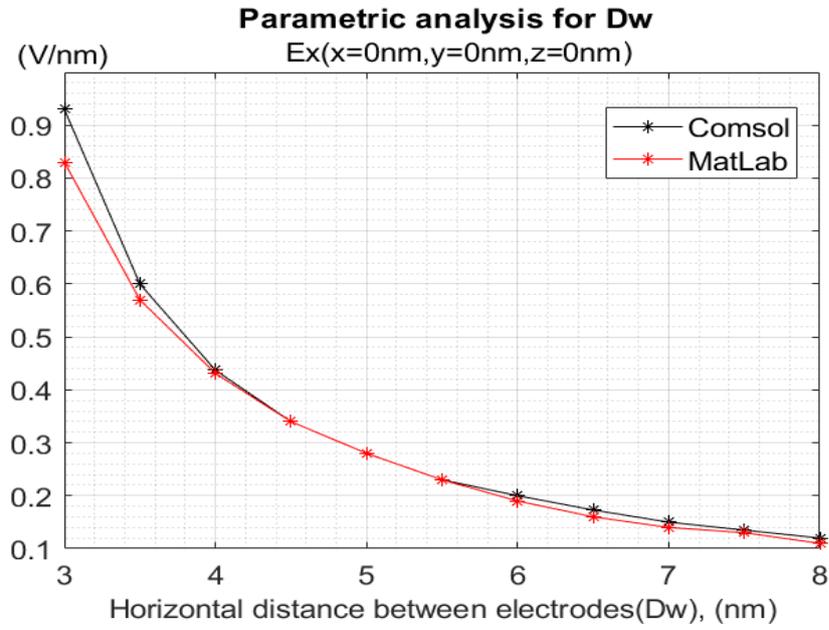


Figure 5.23: Parametric analysis for the horizontal distance between the right and left electrode.

incredible match between the results got from the numerical simulator, Comsol, and the implemented model in Matlab. This means that the charge model, in according with the configuration number 3, is extremely sensitive to Dw. As a consequence, what we can conclude is that this is not just a data but a parameter, by adding another degree of freedom to this model.

5.3.3.2 Parametric analysis for Dv

Now, the same has been done for Dv, which is the vertical distance between centres of the two upper electrodes and the clock wire. Like it can appreciate in the figure 5.24, also, in this case, there is an incredible match. Now, it is less precise than the results got for Dw, but it is possible to determine a range regards Dv for which the model works. In this case, there is a range for which Dv can change and the model go on to work. The results between Comsol and Matlab remains acceptable for values which are below -3 nm.

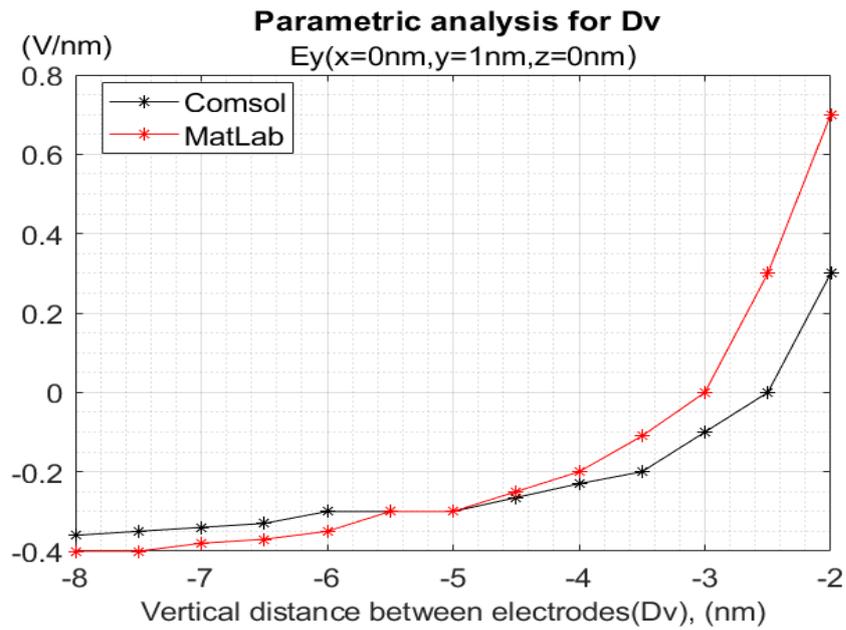


Figure 5.24: Parametric analysis for the vertical distance between the upper electrodes and the clock electrode by considering an absolute point.

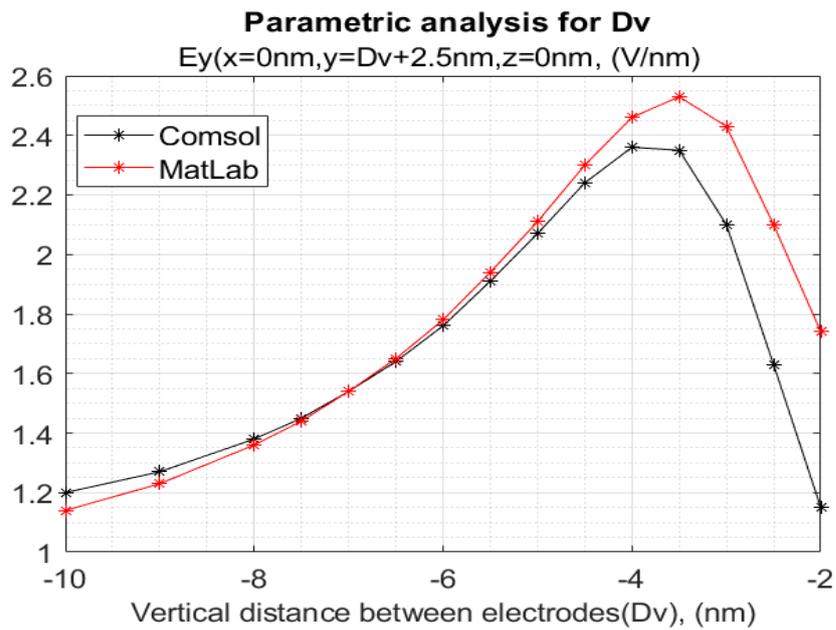


Figure 5.25: Parametric analysis for the vertical distance between the upper electrodes and the clock electrode by considering a relative point.

This is not a problem, since, as it has been mentioned in other section, it is better to put the molecules far from the two upper electrodes, to give space at the field lines of the vertical electric field to become straight. But in the figure 5.24, an absolute reference point has been considered. To complete the parametric analysis for this geometrical data, a relative point has been analysed, like it is possible to appreciate in the figure 5.25 shown above. Relative means that, by moving the clock electrode, the field was always considered at the same distance from it, that is, by going on to look at the figure 5.25, 2.5 nm from the centre of the wire, or, analogously 1.5 nm from its surface since the radius of it is 1 nm. What is important is that, also, in this case, the working area has been determined, that is up to -3 nm. As a consequence, after these considerations, in that range, also Dv moves from to be a data to be a parameter.

5.3.3.3 Parametric analysis for the applied voltage

The last parametric that has to be considered is the applied voltage. Let's start from the clock voltage. In this case, the best result, about parametric analysis, is reached, like it is possible to appreciate in the figure 5.26. But, honestly speaking, there is nothing to be surprise since, like it is shown in the figure 5.20 where an incredible resolution is got for a $V_{\text{clk}} = -8V$, the field is linearly proportional to the applied voltage, since also the linear charge λ is. As a consequence, and as it is demonstrated by looking at the figure 5.26, this remains true for different applied voltages.

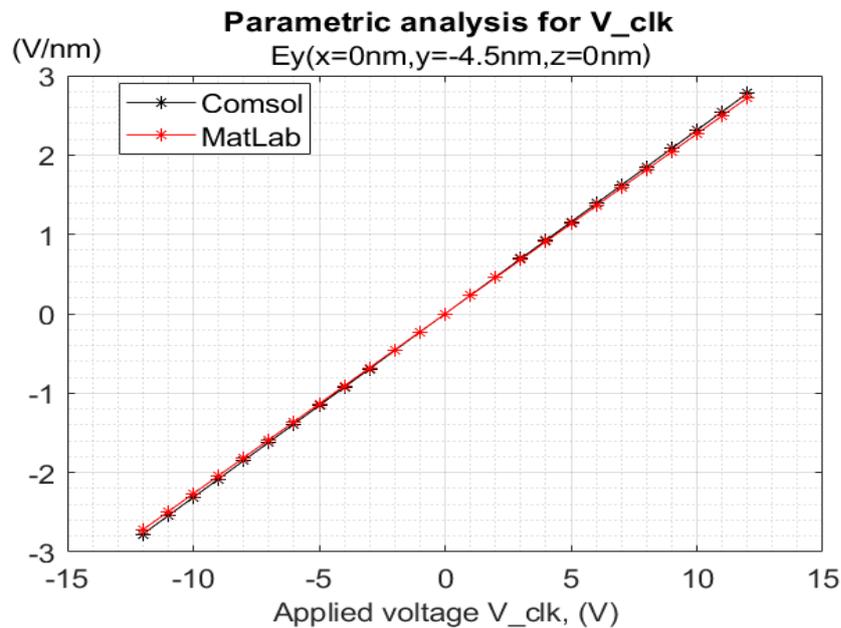


Figure 5.26: Parametric analysis for the applied voltage in the clock mode.

The same happens, of course, for the same reasoning, for the driver voltage V_{dri} . So, let's conclude that also these two arguments are parameters of the model and not just data.

5.3.4 Final consideration about the configuration number 3

As it has been demonstrated, this final configuration works fantastically, by giving a big control of the model and so a minimal tolerance is reached. Like it has been anticipated, this model is not generic even if it has been optimized for a huge number of cases, to have a complete arsenal ready to be used inside SCERPA. At this point, it is possible to consider concluded this part and move over towards other topics.

Chapter 6

Physical implementation for a Molecular QCA and relative limits

By wanting to implement this technology, it is necessary to use a photolithography process with sophisticated skills since nano-devices have to be realised by managing nano-objects like molecules.

So, first of all, the first step consists in creating the nanometric trench where the nanowires have to be deposited. Generally, the typical material, that is used to realise the trench, is Silicon Nitride (Si_3N_4), which is, by looking at the figure 6.1, the part of the section in grey. After that this part has been built, it's the time to realise the nano-wires.

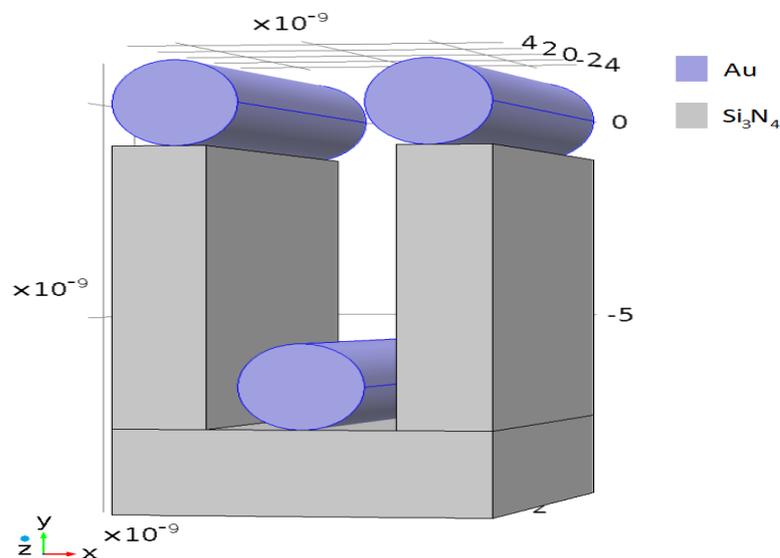


Figure 6.1: Real structure of the electrodes by considering also the support for the nano-wires.

A lot of different metals can be used during photolithography process, but the best for this aim is gold, since it is not subjected to corrosion and oxidation, can be easily managed since it is not toxic and finally, the last important property, it has a high sulphur affinity [21], which is very important to combine it with the power of molecular QCA technology: self-assembly process, where thiols create easily link with the sulphur atom of the clock electrode. By continuing to do reference to the figure 6.1, the gold wires are in violet.

At the beginning of this report, an assumption has been done: instead of considering wires with square section, which is an impossible shape to get with lithography process, especially with these dimensions, cylindrical form has been considered. This was helpful also to simplify the model, since, edges are so difficult to be described with closed equations. But at this point, it becomes almost mandatory to ask for our-self if this assumption could be a limitation or not, since also perfect cylindrical wires are impossible to get due to technological limits. As a consequence, through the numerical simulator Comsol, structures of electrodes with different shapes have been considered, by starting from the theoretical one, the square section, up to the easiest to model: the cylindrical.

6.1 Considerations about different shapes for the driver mode

Like usual, or better, at this point of the report, like a tradition, let's start from the driver mode, which is, as you know very well, used to impose the logic state on the molecules.

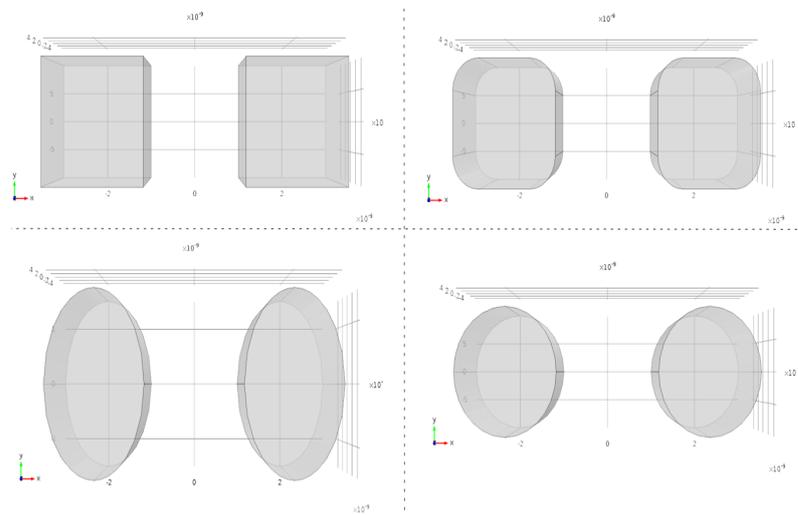


Figure 6.2: Considered shapes for the driver mode due to technological limits.

By looking at the figure 6.2, it's able to appreciate the flow that has been analysed, starting from a profile with edge and by rounding off the profiles, step by step, up to the smoothest, the cylindrical. By imposing the '1' logic static, as usual in the driver mode, the horizontal field has been considered, used to polarize the electrons of the molecules in the right way. The plots, for different profiles, are shown in the figure 6.3. What we can understand from it is that every time that the shape becomes smoother, the field and its relative field lines, start to become more bowed. This is normal since, by thinking to a capacitor with parallel flat planes, the field lines are practically straight, but since the square profile has a tiny dimension, its field lines curve a bit. This behaviour becomes more evident as the profiles have smoother shape.

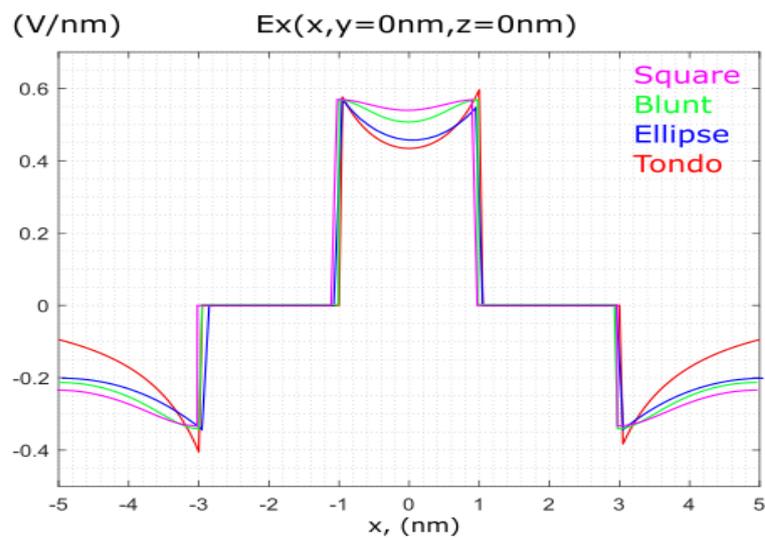


Figure 6.3: Comparison between different profiles about $E_x(x,y=0\text{nm},z=0\text{nm})$ by forcing the '1' static logic with the driver mode.

What it is important to conclude from the figure 6.3 is that, fortunately, the cylindrical profile, which is the shape that has been analysed and modelled in this thesis, is the worst case since it has the minimum value between all considered profiles. As a consequence, it's enough to guarantee that the algorithm works with this shape and so, surely, it will work for the others. The last thing that has to be highlighted is the difference, that is in the order of percentage, so probably, nothing serious.

At this point, let's go on with the most important mode: the clock and look at what happens.

6.2 Considerations about different shapes for the clock mode

The same shapes, that have been analysed for the driver mode have been considered for the clock mode, like it is possible to appreciate in the figure 6.4. Notice that, in order to perform these comparisons, the geometrical data for the structures are, of course, always the same declared in the chapter 4, where all data are printouts.

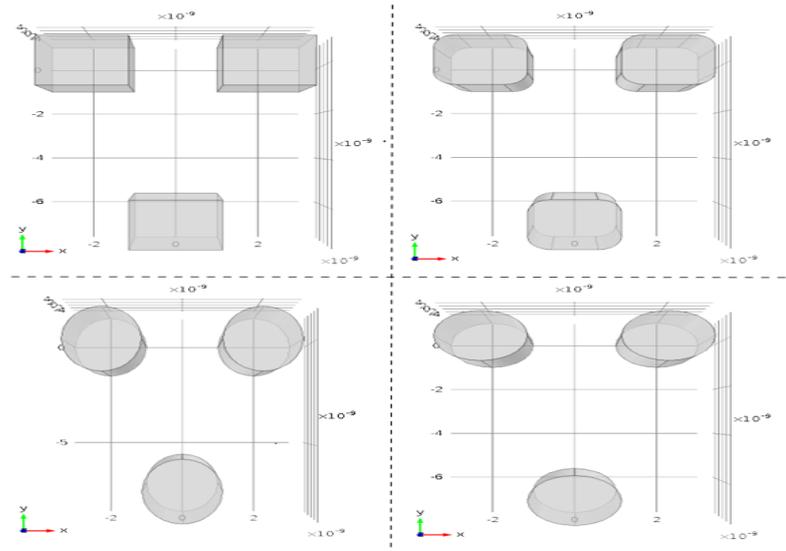


Figure 6.4: Considered shapes for the clock mode due to technological limits.

By wanting to choose between the hold or reset mode, since nothing changes except for the sign, the reset mode has been considered. By looking at the figure 6.5, it is able to appreciate that in this case is also better than the driver mode, since there are just some boundary effects near to the end of wire, in $y=-5.6$ nm, but after it, for increasing y -coordinate's values, it is practically impossible to distinguish curves between different profiles. Instead, the differences that are present in $y=-7.6$ nm, between the ellipse's plot and the others depends just on the fact that the dimension of the ellipse profile is bigger, by definition. But since in that region there are not molecules, it doesn't matter while in the opposite direction there is a fantastic match.

Now, surely all these considerations are true since the reported fields, in the chapter 4 and 5, are enough far away from regions where difference boundary effects are present and by remembering that the aim is to study and model complex structures of electrodes in specific safe zone, where it's fundamental to guarantee a good description of the field.

The last remaining thing to analyse regards the presence or not of the support, that is

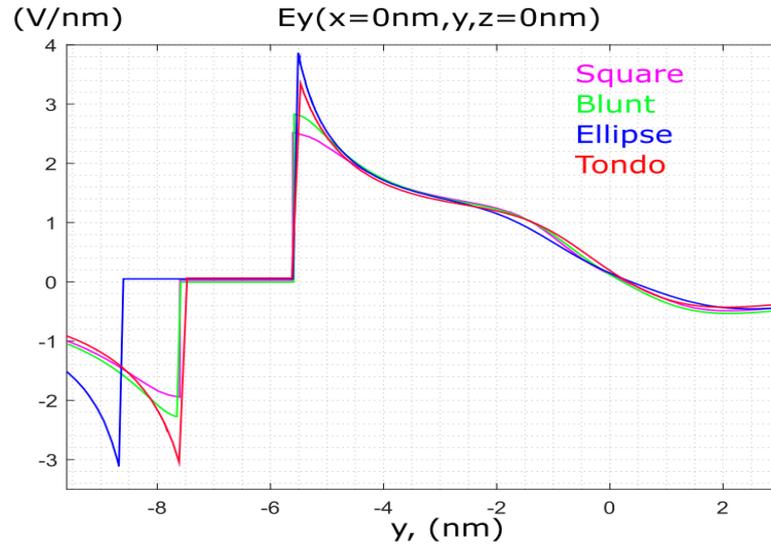


Figure 6.5: Comparison between different profiles about $E_y(x=0\text{nm},y,z=0\text{nm})$ by imposing the reset mode.

shown in the figure 6.1, and which are the consequences of it. This is another important and crucial point since, from a practical point of point, it is impossible to realise a structure of electrodes without the support but up to this moment, it has never been mentioned and never been taken into account.

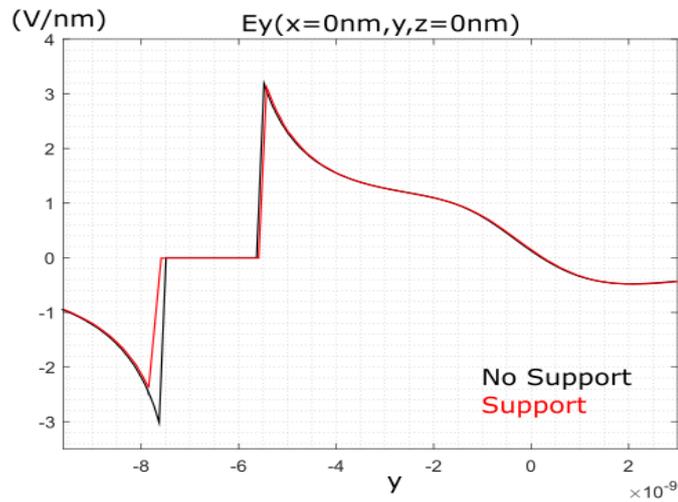


Figure 6.6: Comparison of the electric field in reset mode between structure of electrodes with and without support.

Honestly speaking, it is not able to include it inside a model in a closed form, but by remembering which is the aim of this report, it is enough that, in the safe zone region, no effects are appreciable. By going on to consider the reset mode, the first plot which is proposed is the typical for this case: $E_y(x=0\text{nm},y,z=0\text{nm})$. Like it is possible to appreciate in the figure 6.6, it is practically impossible to distinguish the two plots. There is a small difference outside of the safe zone region, for $y \leq -7.6 \text{ nm}$, but, as usual, it doesn't matter. This is a fundamental result since it solidifies all done job up to this moment since a link with the physical implementation has been guaranteed and demonstrated. Below it is shown how the field lines are canalised in the support, on the left of the figure 6.7, with respect to the other case, on the right, without support. Also, in this case, happens what is intuitive: the support works like a sort of vehicle for the field. Again, where molecules are present, in the middle of the structure, nothing changes. To be precise, the support focuses more the field towards the clock electrode, so a better control is guaranteed in a real case.

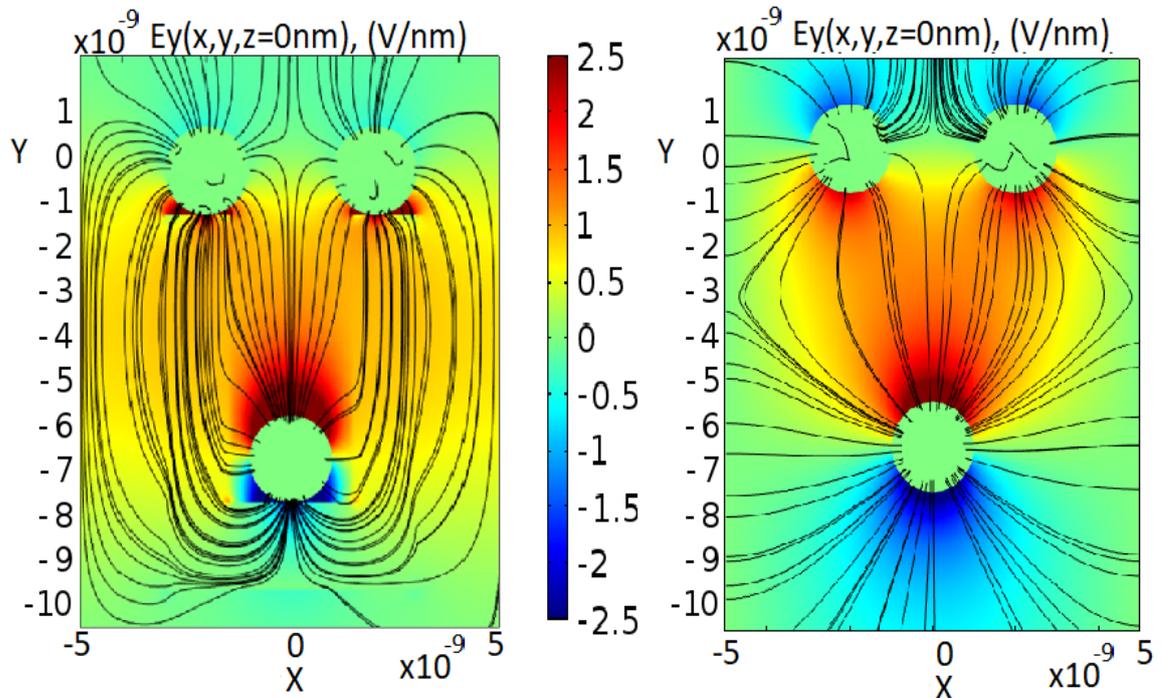


Figure 6.7: Comparison of the electric field in reset mode between structure of electrodes with (on the left) and without (on the right) support about the field lines.

Chapter 7

Towards Scerpa

A lot of times, references to a particular algorithm, called SCERPA, have been done. This is a program under development in the VLSI laboratory of the Politecnico of Turin. This laboratory is dedicated to many different topics, and one of these is innovation, and Molecular QCA is one of the studied arguments. Another aspect, that has been mentioned many times along this report, is that the algorithm devoted to the description of the electric field inside molecules' structure for the QCA has to be fused with SCERPA to realise another step ahead between physical and logic world.

Before to see how the algorithm for the electric field works, it could be useful to undertake a journey inside SCERPA to understand how it works and then to comprehend how to fuse the two algorithms.

7.1 Layout of a structure

Like every complex well-done program has to be organized, SCERPA is divided in section, in order to simplify the management.

The first part consists in realising the layout of the circuit that has to be analysed, by imposing the position of each molecule. In order to reach this aim, a matrix organization has been thought, by considering a sort of grid which models what we want to evaluate. By wanting to help the reader, some example, of course, may be proposed.

7.1.1 Selection of the molecule under test

By going on to speak about the layout, the first thing to do is the selection of a molecule that is used to realise the wanted structure. Each molecule is described through three different matrixes.

The first one is dedicated to the selection of the initial charge for the four different dots, in base on which vertical electric field is applied, by depending on which clock mode

is imposed, reset or hold. Just to give an example, for the Bisferrocene with 4 charges, the charge partition between four dots depending on the applied vertical field is synthesized in the following table 7.1.

	Mode	DOT1	DOT2	DOT3	DOT4
Initial charge	Hold	0.475	0.471	-0.223	0.277
Initial charge	No signal	0.370	0.352	-0.023	0.307
Initial charge	reset	0.027	0.026	0.141	0.806

Table 7.1: Partition of electrons between dots in function of the clock field for the Bisferrocene [5].

Then, the second matrix is dedicated to explicate coordinates in the 3D space about the four different dots. Again, by wanting to give an example and by going on to speak about the bisferrocene, the data are organized like it's possible to appreciate in the following table 7.2:

	X, (Å)	Y, (Å)	Z, (Å)	
Position	-3.622	-5.062	-0.094	DOT1
Position	-3.588	+5.083	-0.094	DOT2
Position	+3.133515	-0.011731	-0.755	DOT3
Position	+11.776298	-0.053777	+0.409	DOT4

Table 7.2: Coordinates of dots, in the 3D space, for the Bisferrocene [5].

and in the end, there is the last matrix, that some time for specific molecules could be just a vector, which is called draw association, used just for some settings about plots.

Now, this is what happens for each molecule, so, by selecting which one has to be used in the algorithm, the relative data will be uploaded in the code. As it has been mentioned once, nowadays, thanks to the great skill to synthesize molecules, there is a sort of catalogue where different of them can be chosen in base on what is needed and which are the needed characteristics. To demonstrate that this is true, it is enough to think that, just for SCERPA, at least six different molecules can be used, like, for instance, decatriene, butane and others.

7.1.2 Definition of the structure

This is one of the two most important parts about to the layout section of the algorithm. Here is declared how the structure is realised and which are, for each molecule, the phase zones that manage the relative molecules. By wanting to simplify the explanation, an

easy example could be used, like a single wire composed of eight molecules under the same phase.

Here, it's shown the needed code used to define this wire, called layout 1.

$$\text{structure} = \text{'Dr1' 'Dr2' '1' '1' '1' '1' '1' '1' '1' '1'} \quad (7.1)$$

and the obtained structure is shown in the figure 7.1. So, like it's intuitive to understand, for each '1' presents in the definition, a relative molecule is placed, and it is managed from the phase zone number 1.

The polarization of the driver depends on which logic value has been imposed in the simulation, since, as it's well known, the driver is used just to impose the logic input to the structure under test.

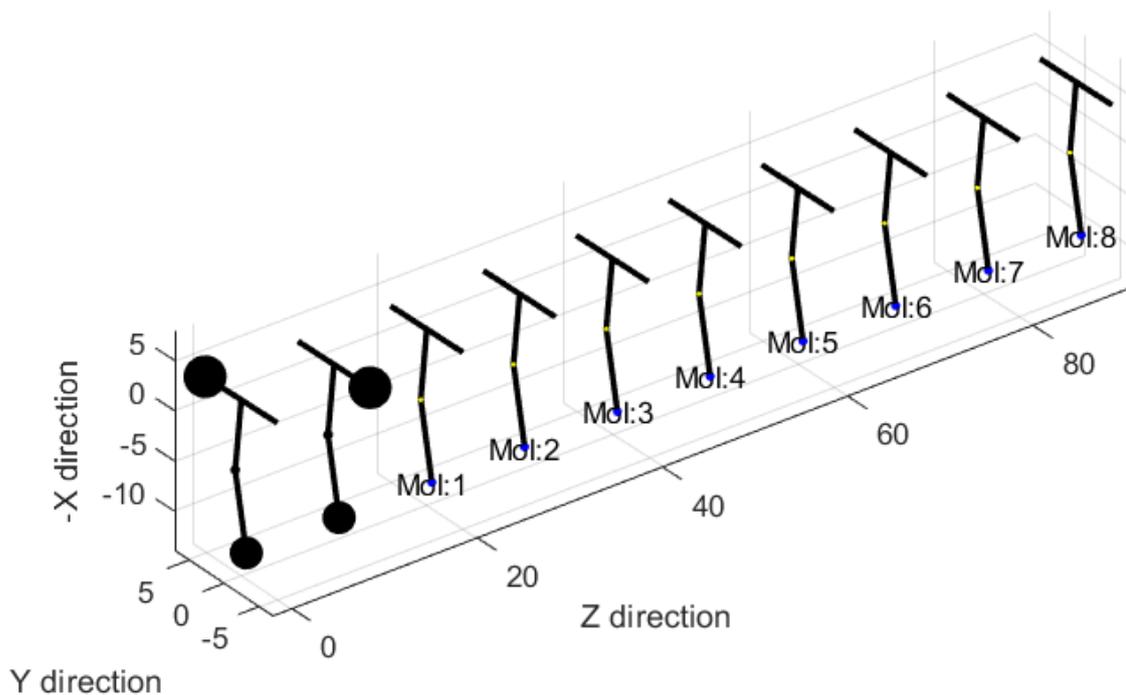


Figure 7.1: Wire composed from eight molecules under the same phase by using the SCERPA algorithm [5].

All molecules have been declared with the value '1', this means that all of them are below the same phase zone, which is, of course, the phase number 1. It's possible to obtain also the same wire but with different control, called layout 2, by imposing two different phase zones, in the following way:

$$\text{structure} = \text{'Dr1' 'Dr2' '1' '1' '1' '1' '2' '2' '2' '2'} \quad (7.2)$$

In this case, the obtained structure is again the one shown in the figure 7.1 above, but with different control, since the molecules, number 1,2,3 and 4 are under control of the phase zone number 1 while the molecule number 5,6,7 and 8 are managed from the phases zone number 2.

Thanks to this type of definition, an enormous degree of freedom is available, since complex circuits can be obtained like bus, inverter and majority voter, which are shown, respectively, in the figure 7.2, 7.3 and 7.4.

Now, thanks to the completeness of the algorithm, it can evaluate and analyse also a lot of non-ideality for the structure, like the possible rotation of each molecule, since to assume that all molecules are perfectly in parallel one to each other is practically impossible. Then another non-ideality is related to the shift between molecules, in all 3D coordinates, so the algorithm is able to analyse that, during the process, the distance between them can change and vary, since, also for this parameter is practically absurd to assume that there is a perfect placement since this type of technology is based on very tiny objects which are enormously sensitive to process variation.

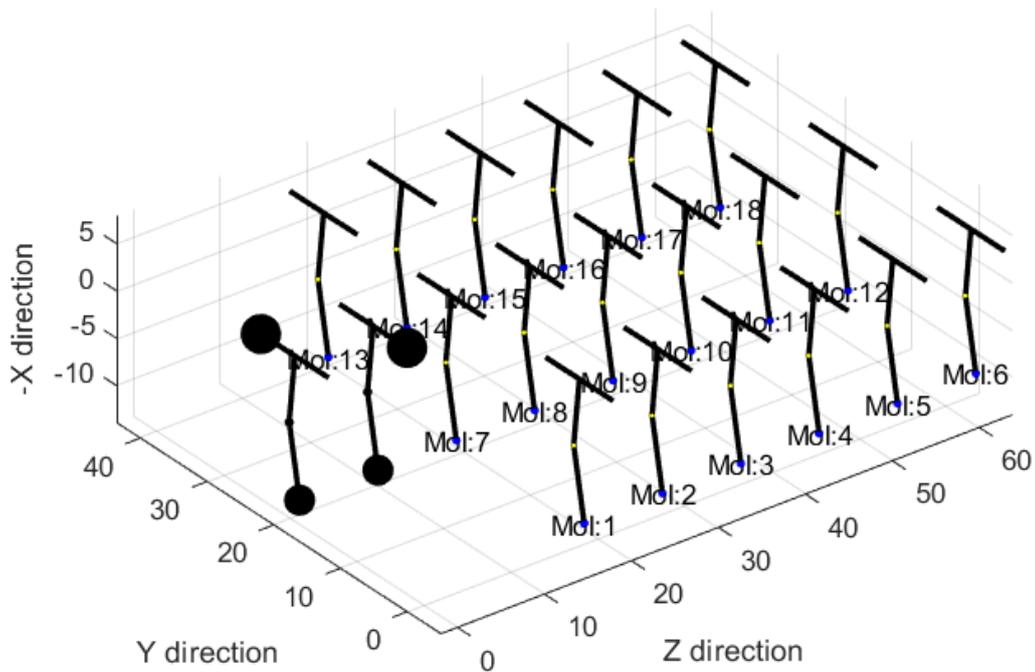


Figure 7.2: Bus obtained through the algorithm SCERPA [5].

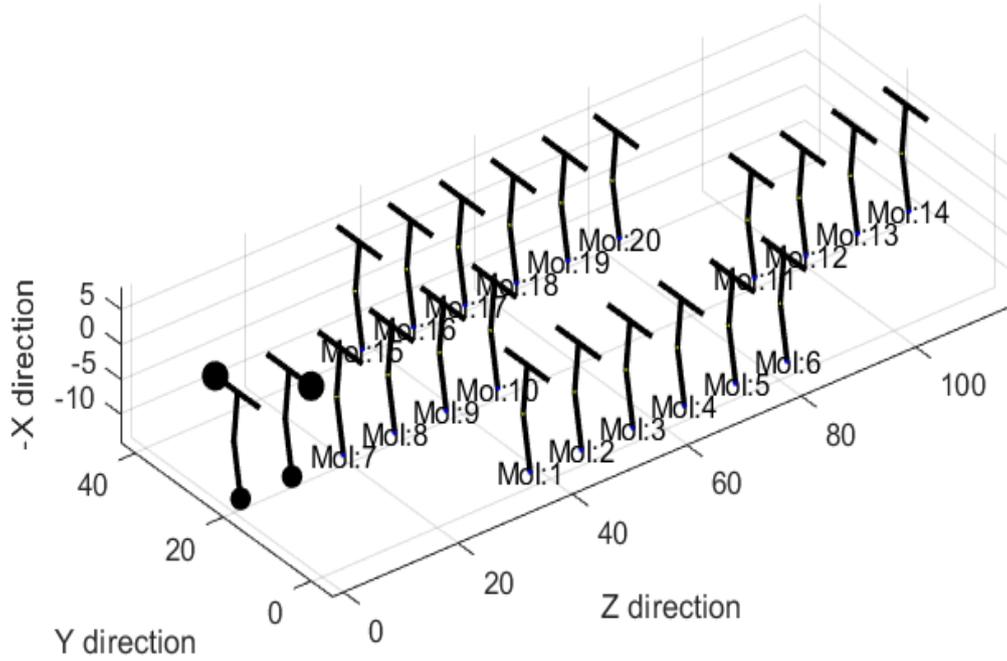


Figure 7.3: Inverter obtained through the algorithm SCERPA [5].

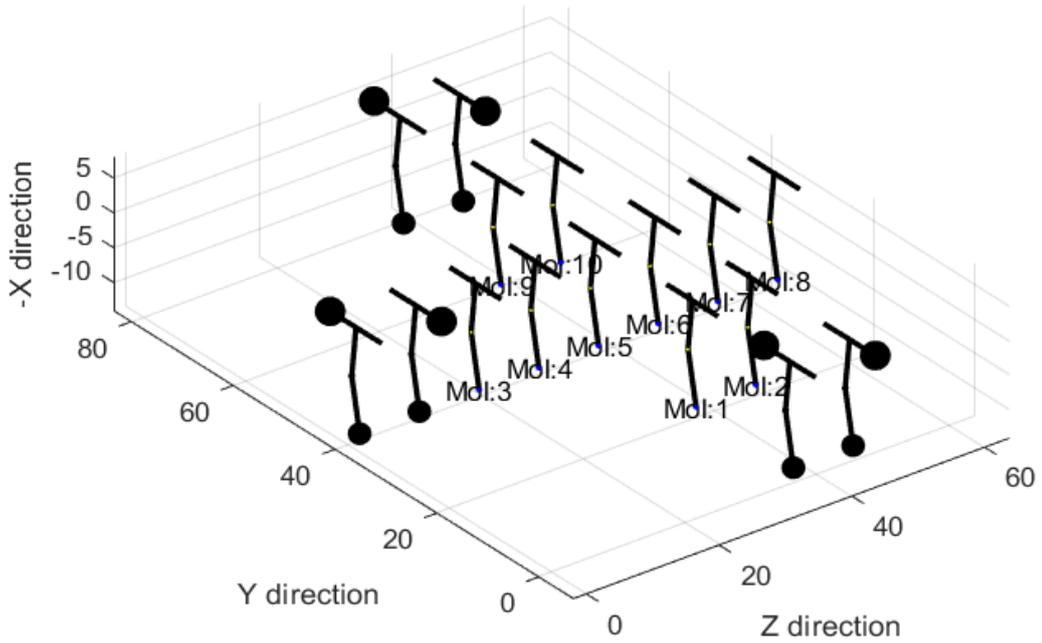


Figure 7.4: Majority voter obtained through the algorithm SCERPA [5]

At this points, all information about the layout have been considered and we are ready to continue the explanation by considering how to impose the wanted logic value to the driver and how to determine the evolution of different phase zones.

7.2 Definition of the drivers

Once that the structure has been declared and defined, it's time to determine which is the logical input that has to be imposed to the circuits under test. The algorithm is organized, both for the driver and the phases, again, in matrixes where the column represent the different time steps, and for each time, on every row there are the relative values associated to the different drivers.

```

circuit.Values_Dr = {
  'Dr1' -4.5 -4.5 -4.5 -4.5 +4.5 +4.5 +4.5 +4.5 +4.5 'end';
  'Dr2' +4.5 +4.5 +4.5 +4.5 -4.5 -4.5 -4.5 -4.5 -4.5 'end';
  'Dr3' -4.5 -4.5 -4.5 -4.5 +4.5 +4.5 +4.5 +4.5 +4.5 'end';
  'Dr4' +4.5 +4.5 +4.5 +4.5 -4.5 -4.5 -4.5 -4.5 -4.5 'end';
  'Dr5' -4.5 -4.5 -4.5 -4.5 +4.5 +4.5 +4.5 +4.5 +6.5 'end';
  'Dr6' +4.5 +4.5 +4.5 +4.5 -4.5 -4.5 -4.5 -4.5 -4.5 'end'};

      ↑   ↑   ↑   ↑   ↑   ↑   ↑   ↑   ↑
      t0  t1  t2  t3  t4  t5  t6  t7  t8

```

Figure 7.5: Definition of drivers through the algorithm SCERPA [5].

As it is possible to appreciate by looking at the figure 7.5, the typical values used to impose the logical inputs are -4.5 and +4.5. By remembering the figure 1.5, which is shown in the introduction, to apply a specific logic value to the circuit, a couple of drivers have to be used, since each box of molecules is obtained from two of them. It could be interesting to move from the representation based on a figure of merit, shown, again, in the figure 1.5 towards a physical point of view inside the algorithm like it is possible to appreciate in the figure 7.6.

Notice that this is a sort of pre-compiled matrix ready to be used with different structures with different degree of complexity, as a consequence, not in every simulation all rows of

the matrix shown in the figure 7.5 are used. To understand, in a single wire where just an input is imposed, the only used drivers are the drivers 1 and 2, while the others are neglected and also their relative rows with their associated values. On the other hand, with more complex circuits, like, for instance, the majority voter, where there are up to three inputs, all drivers are used since three couples are needed.

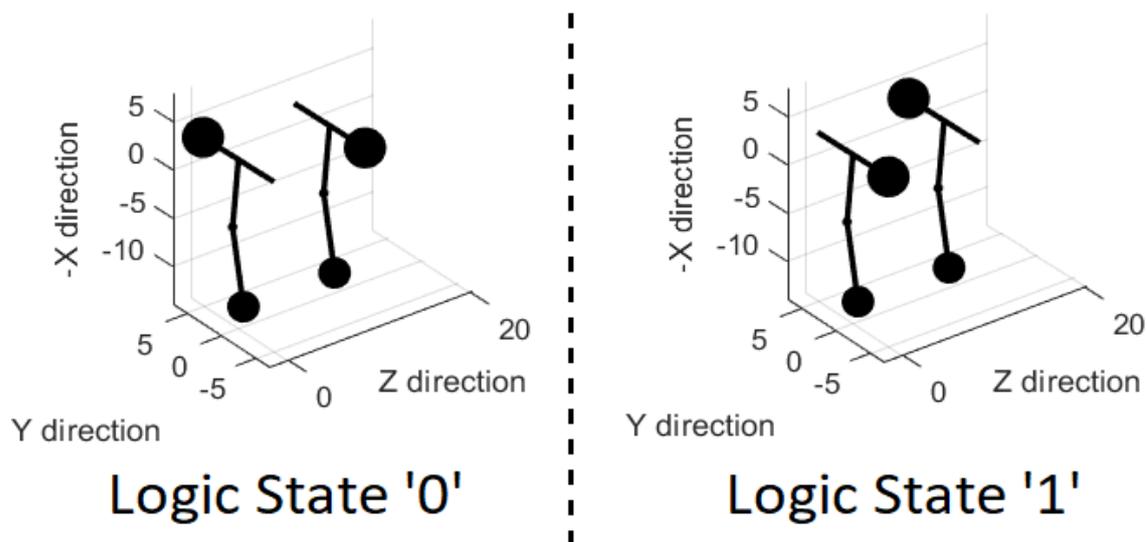


Figure 7.6: Logic states imposed from the drivers through the algorithm SCERPA [5].

7.3 Definition of the phases

This is the last important part of the algorithm, dedicated to the definition of the layout, since here the behaviour and the development of each single phase is imposed during the simulation of the circuit while the algorithm is running.

Here the organization, like it has been anticipated before, follows exactly the same of the driver, thanks to a matrix where each column represents a specific time and for each row, there are information about the relative phase zone. Also for this case, an example could be very useful in order to explain better what we are speaking about. By looking at the figure 7.7, it's able to appreciate the definition of the evolution for three different phases. So, as a consequence, by doing reference to the layout 1, explained with the definition 7.1 shown above, all molecules declared with '1' follow the evolution of the phase 1, while according to the layout 2, explained with the definition 7.2, the molecules declared with '1' follow, again, the behaviour of the phase 1 while that are defined with '2' follow the evolution of the phase 2.

```

%Phases evolution
circuit.stack_phase(1,:) = [ 2   2  -2  -2];
circuit.stack_phase(2,:) = [-2   2   2  -2];
circuit.stack_phase(3,:) = [-2  -2   2   2];
                        ↑   ↑   ↑   ↑
                        t0  t1  t2  t3

```

Figure 7.7: Example of definition of phases for a generic circuit through the algorithm SCERPA.

Now that all possible considerations about to the definition of the circuit and about to its relative evolution in time have been completely considered, let's go on whit the last big part of SCERPA, which is computational.

7.4 Computation of the circuit under test

Probably this is the most difficult part of the algorithm. Anyway, some idea and reasoning must be done to understand how it works and how these physical structure are analysed. The three fundamental steps are:

- initialization;
- molecular-interaction computation;
- final charge distribution evaluation.

Notice that, for every different time, t_0, t_1, t_2 and so on, where different settings for drivers and phases are defined, these steps have to be repeated.

7.4.1 Initialization step

The first step, as always happens in all simulator, is the initialization step which means four different sub-steps:

- definition of the positions of the molecules, thanks to the layout part of the algorithm, with some eventual considerations about non-ideality like shift or random rotation;
- definition of the initial charge, like it has been explained before, per each molecule;

- definition of the driver configuration;
- definition of the configuration of the phase, to determine how the circuit must evolve.

At this point one important thing has to be highlighted: we are assuming that these conditions are, strictly speaking, frozen but this is not absolutely true, since in a real circuit, due to its dimension, there are always variations of its conditions. Anyway, it is necessary to use some well-defined values, got from numerical simulator like AB-Initio and then stored in a Look-Up table, in order to establish a starting point for the algorithm.

7.4.2 Molecular-interaction computation step

Before to explain how the interaction computation works, it could be useful to try to explain how the information propagates inside the structure and how it is managed from a theoretical point of view. This technology, as it is known, is based on the field-coupling nano computing between different objects, molecules, where the information is associated with the position of electrons and not to current flow. As a consequence, the main principle that has to be considered and modelled is the field on a specific molecule- i due to the driver and due to other molecules- j , as a consequence, the equivalent input voltage for each single molecule- i under test has to be determined like follows:

$$V_{IN,i}^{Tot} = V_{IN,i}^{Driver} + \sum_{j=1, j \neq i}^N [V_{IN,i}^{Q_{1J}} + V_{IN,i}^{Q_{2J}} + V_{IN,i}^{Q_{3J}} + V_{IN,i}^{Q_{4J}}] \quad (7.3)$$

where N , of course, represents the total number of molecules that build the circuit and the charges Q_{1J} , Q_{2J} , Q_{3J} and finally Q_{4J} are associated, respectively, to the dot1, dot2, dot3 and dot4 of the molecule- j . So this justifies why it's practically impossible to determine a single equation that solves the entire system in one single shot but an iterative approach is employed where it is necessary to reach convergence, called iterative self-consistent solution.

After this overview, let's go on by explaining how the molecular-interaction computation step works.

First of all, per each molecule- i , it's necessary to calculate the effect of the driver $Dr-k$, so the evaluation of the electric field has to be determined on all charges that compose the molecule- i . As a consequence, the result is:

$$V_{IN,i}^{Driver} = V_{IN,i}^{Driver} \quad (7.4)$$

Then, until the error is larger than the relative admitted error ϵ , what happens is that:

- per each molecule- i , the charge Q_{1i} , Q_{2i} , Q_{3i} and Q_{4i} are evaluated;

- per each molecule- $j \neq i$, the variation $\Delta V_{\text{mol}_j \rightarrow \text{mol}_i}$ is evaluated and the result ΔV is stored, which is just the increment on the molecule- i from the previous step;
- finally, the maximum total voltage variation ΔV , which is the error, is evaluated.

Notice that, this process is done until the error ϵ is not below a certain threshold. Then, the notation

$$\Delta V_{\text{mol}_j \rightarrow \text{mol}_i}$$

means that, at each step of the loop, the charge of molecule- j changes of ΔQ_{1j} , ΔQ_{2j} , ΔQ_{3j} and ΔQ_{4j} , since it has been evaluated it-self affected by other molecules, so the only evaluated correspondence is:

$$\Delta V \Leftarrow \Delta Q$$

So that:

$$\Delta V_{\text{mol}_j \rightarrow \text{mol}_i} = V_{IN,i}^{\Delta Q_{1j}} + V_{IN,i}^{\Delta Q_{2j}} + V_{IN,i}^{\Delta Q_{3j}} + V_{IN,i}^{\Delta Q_{4j}} \quad (7.5)$$

7.4.3 Final charge distribution evaluation

This is the last step which is performed once that the convergence of the previous step is reached. At this point, when $V_{IN,i}^{Tot}$ has been calculated per each molecule- i , the values of the charges of the molecule- i are immediately derived using the characteristics of it, known thanks to the AB-Initio simulator. Like you have probably noted, just the explanation is very complex, in fact, it hopes that it is enough clear, but image instead the complexity that there is beyond in order to get reasonable results. Just to conclude, before to shown an entire example by hoping that it could be useful to clarify eventual doubts, different solvers, nowadays, are implemented in the algorithm. As it has been explained, the total field on each molecule- i depends on all other j but generally, the most important influence arrives from the nearest. So, instead of considering all N molecules for each step, a sub-system can be considered, by limiting the number to 10 for instance or others but not the entire N molecules. Probably some inaccuracy appears, but like always happens, it is enough that remains below a certain threshold. At this point, a complex example could be analysed in order to use it to clarify some eventual doubts.

7.5 Example for SCERPA: Majority Voter

This is a very complex structure with its critical issues, that will be explained later, but it is also a complete example since, thanks to it, the AND and OR gates can be realised, which are two of the three fundamental gates.

Just to remember what has been told in the introduction, by starting from the majority voter, to get the AND gate is necessary to force one of the three input to the '0' logic value, while, on the other hand, for the OR is needed to force one input to '1'. These considerations are shown in the figure 1.6. The first step consists in explicating the layout of the circuit under test through, as it has been explained before in the dedicated section, the layout matrix, which is shown below the figure 7.8. The obtained circuit is the same shown in the image 7.4.

```

% %majority voter
circuit.structure = {
    0      0      0      0  'Dr3' 'Dr4'  0      0  ;
    0      0      0      0  '1'   '1'   0      0  ;
'Dr1' 'Dr2'  '1'  '1'  '2'   '2'  '3'  '3'  ;
    0      0      0      0  '1'   '1'   0      0  ;
    0      0      0      0  'Dr5' 'Dr6'  0      0  };

```

Figure 7.8: Declaration of Majority voter's layout through the algorithm SCERPA [5].

In this case, by going on to do reference to the figure 7.4, the molecules 1,2,3,4,9 and 10 are managed from the phase zone number 1, then, the molecules 5 and 6 are under control from the phase zone number 2 and finally the phase 3 manages the molecules 7 and 8 dedicated to put in output the result. Notice that, there is a perfect match with the organization of the phases shown in the introduction about the figure 1.11, so everything is coherent.

Now, it's time to define how the different phases must evolve between the hold and reset state. Remember that, the hold mode is imposed from the value +2 while the reset from the value -2. In this case, the evolution follows the organization shown in the figure 7.9.

For the first time, t1, the phase 1 is imposed in hold mode and its relative molecules, while the other two are in reset mode, so in this way, the information starts from the drivers toward the centre of the circuit. Then, in the time t2 also the centre, composed from molecules 5 and 6, is driven in hold, in order to evaluate the right output and then finally, in the last time t3, the information goes outside towards the molecule 7 and 8 dedicated to the output while the molecules under the phase 1 come back in the reset mode to be ready for another evaluation. All these considerations are shown in the figure 7.10.

The circuit works very well, since like inputs, two drivers have '0' logic value while just

```

circuit.stack_phase(1,:) = [+2 +2 -2 ]; %phase 1
circuit.stack_phase(2,:) = [-2 +2 +2 ]; %phase 2
circuit.stack_phase(3,:) = [-2 -2 +2 ]; %phase 3
                        ↑   ↑   ↑
                        t1  t2  t3

```

Figure 7.9: Definition of Majority voter's time steps thanks to the declaration of phases through the algorithm SCERPA [5].

one has '1'. So the number of drivers with '0' logic value is bigger than the drivers with '1', as a consequence, the output is '0' like it is shown in the figure 7.10 according to the time t3.

The critical issue for this type of circuit is represented from the molecules 5 and 6, which build the cross between the three different drivers coming from opposite directions. In fact, this is the region where non-ideality must be, in some way, limited even if it is not absolutely easy, in order to guarantee a good symmetry for the circuit and have a perfect propagation of the information.

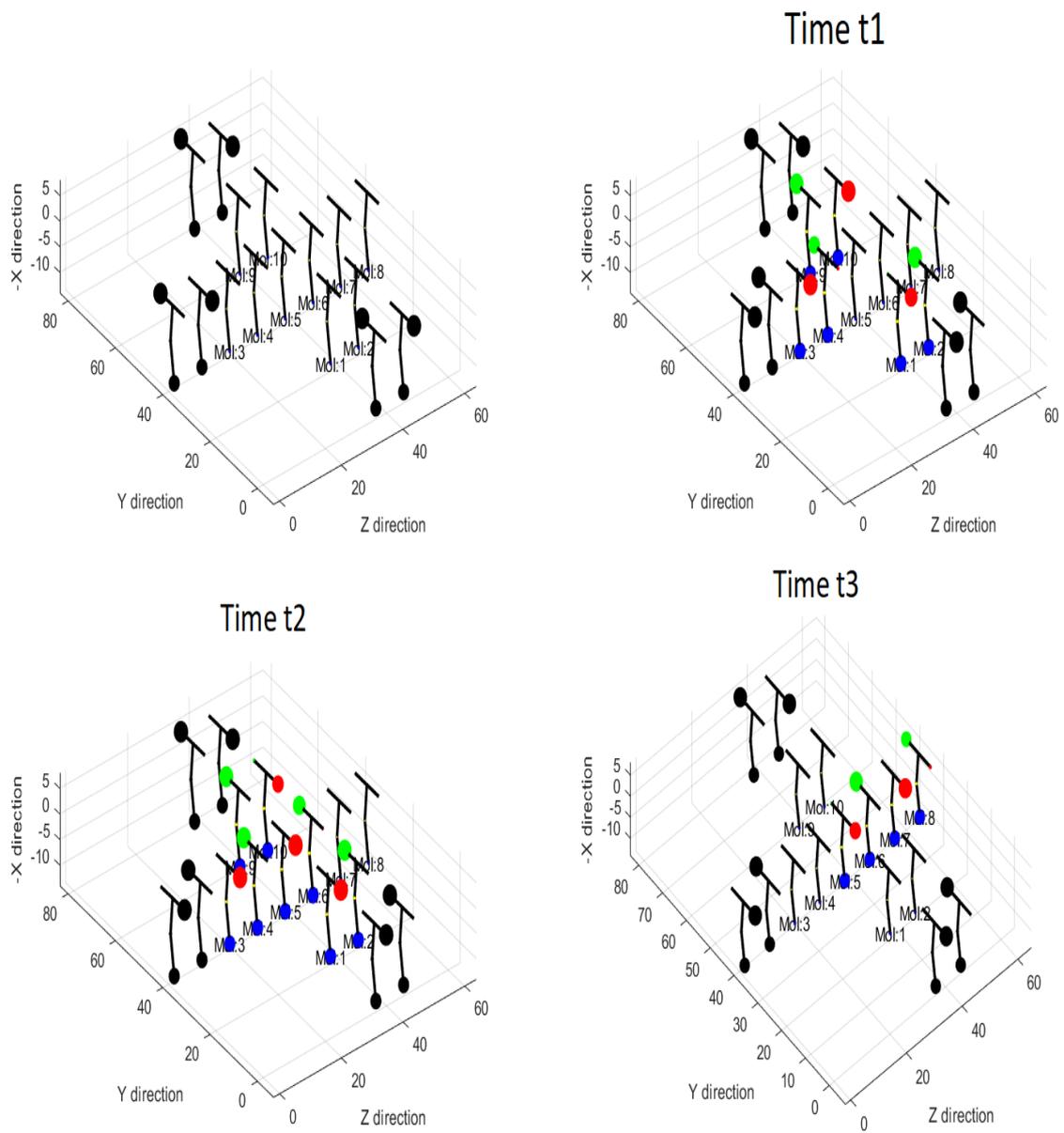


Figure 7.10: Evolution of Majority voter thanks to the declaration of relative phase zones through the algorithm SCERPA [5].

Now, let's go on by showing how the algorithm developed in this thesis works and understand what happens when different structures of electrodes are aligned one after the others. In order to demonstrate that it works, a wire has been considered, composed of three different structures managed from distinct phases.

7.6 Electric field algorithm

So, first of all, by wanting to write as much as possible a scientific report, and by definition, repeatable, the new coordinate system has to be shown, which is the used in SCERPA.

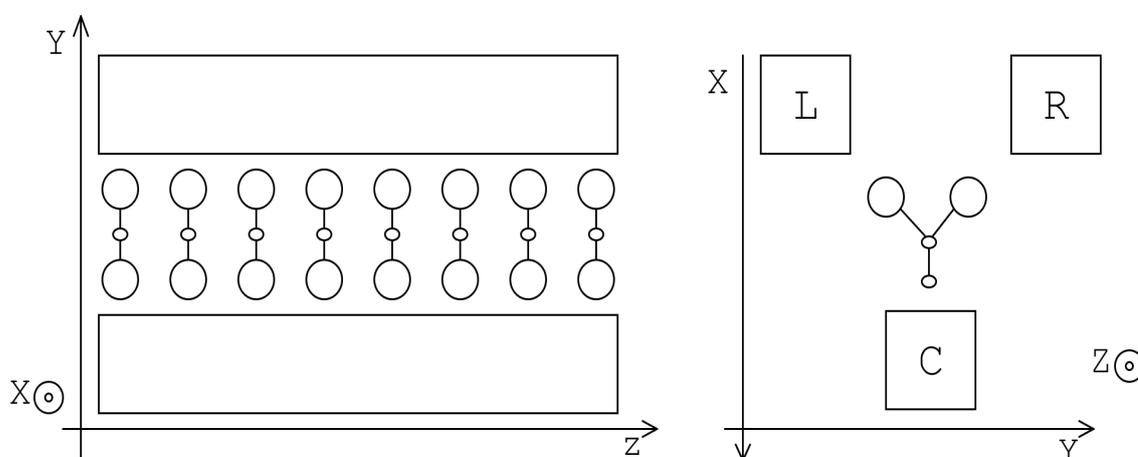


Figure 7.11: Reference axes for structure of electrodes used on algorithm SCERPA.

By comparing the new reference axes concerning the initial shown in the figure 2.1, the longitudinal axis z doesn't change while there is an inversion between the x and y axes. Of course, this doesn't change absolutely the model, but just some name and reference, but since this is an internal issue of SCERPA, a transparent approach will be used to explain how the model works, to avoid to confuse the reader.

As it has been mentioned before, the considered system is a wire composed of three structures of electrodes, like it is possible to appreciate in the following figure 7.12. In this type of simulation, molecules are not shown since this is what SCERPA uses while, in this case, again plots are considered to keep the contact with the physical world beyond these circuits. Instead, about the simulation, it has been structured exactly like SCERPA to minimize the impact during the fusion between them. So, like it's able to appreciate in the figure 7.13, also for this case there is a matrix organization where columns represent the different time steps while rows identify which are the states for each structure of

electrodes in the relative time steps. So, by going on to consider the figure 7.13 the timing diagram is exposed.

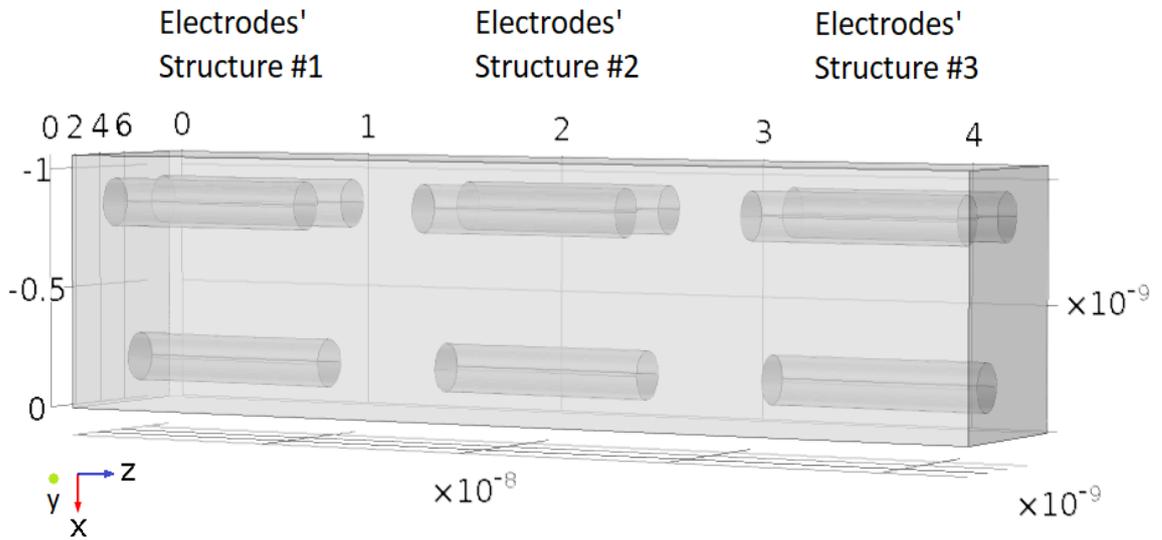


Figure 7.12: Wire, composed from three structures of electrodes, used to test the electric field model on SCERPA algorithm.

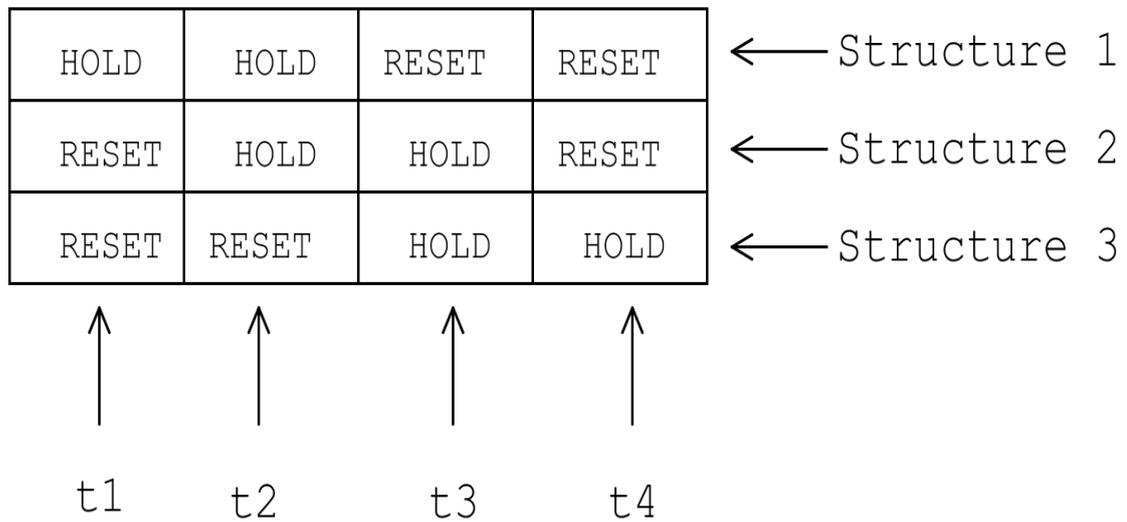


Figure 7.13: Timing diagram for the simulation of the wire composed from three structures of electrodes.

By wanting to clarify how it works, it could be useful to look at the main function and at the organization of the algorithm.

```

%the external loop, dedicated to the time evolution, is used to
  consider in sequence the different time steps, t1,t2, etc up to
  the end of the time for the simulation
for t = 1 : time

%Initialization of the electric field components
Ex = 0;
Ey = 0;

%inner loop used to move along the different structures in a
  specific time step
for ns = 1 : number_structure

  %circuit_Layout is the matrix shown in the figure 7.13 used to
    impose in the main function the relative working mode, reset or
    hold, for each structure in specific time step
  mode = circuit_Layout(ns,t);

  %Ext, Eyt are the temporally field for each structure
  % [Ext,Eyt]=f_wire(position(ns,:),r,L,Dw,Dv,mode,V_clk,V_dri,Px,Py,
    Pz);
  [Ext,Eyt]=f_charge(position(ns,:),r,L,Dw,Dv,mode,V_clk,V_dri,Px,Py,
    Pz);

  %Updating
  Ex=Ex+Ext;
  Ey=Ey+Eyt;

end

end

```

The main function is composed of different parameters: first of all, the position in the space of each structure, organised like always in matrix, then r , L , Dw and Dv are the geometrical parameters, which mean, respectively, radius, length, horizontal distance between upper electrodes and vertical, that are the same that have been used a lot of times. Then, there is the mode for which the structure is working, so driver, reset or hold mode, and the relative applied voltages, and, in the end, the considered volume where algorithm evaluates the electric field. Notice that, there are two different main functions, one is called `f_wire` while the other `f_charge`. The first one uses the finite wire model which is the first implemented on Matlab, even if it is less precise but is very fast to give an idea

of how the electric field is. On the other hand, the second is used to get a precise field paying more computational time. At this point, all knowledge about the algorithm has been provided and so, we can continue. As it is known, the photolithography process has its limits, so it is practically impossible to think to align each structure near to others with an infinite resolution. So, as a consequence, surely there is a minimum distance between them. For these simulations, this distance has been assumed to be equal to 6 nm. Consequently, there are some molecules which are not inside any structures. By looking at the figure 7.14 and 7.15, on the top left there is the simulation result for the time t1, on the top right t2, on down left t3 and, finally, t4 on down right.

From the simulation shown below, there are two contrasting news. The first is that, the model developed in this thesis, works with an incredible precision, in fact it follows very well the numeric solution obtained from Comsol, like it is possible to appreciate in the figure 7.14, which is the same field considered in the figure 5.20, but, of course, by combining three different structures together. Notice that, in this case, is called E_x due to the fact the reference axes is changed but this is just a name, what is shown is the vertical field used to modulate the electrons towards up and down to realise the hold and reset mode.

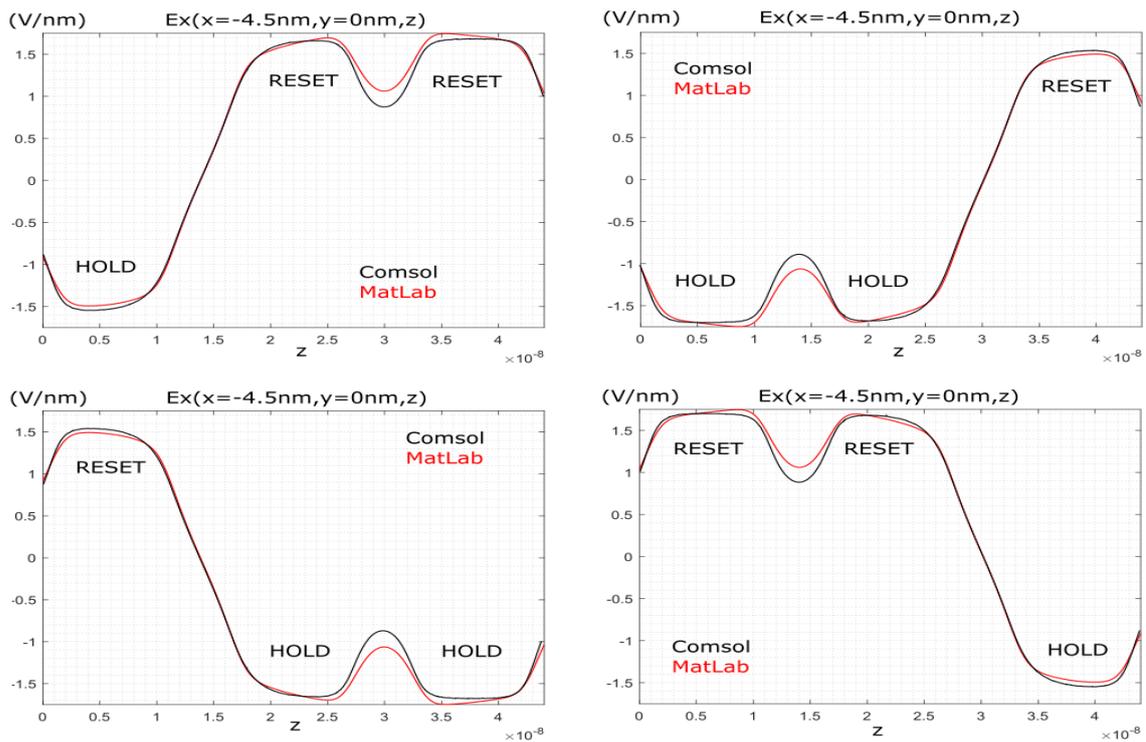


Figure 7.14: Vertical electric field for each steps of the timing diagram shown in figure 7.13 by comparing, as always, Matlab and Comsol.

On the other hand, what happens is that, with these short structures of electrodes, it's

impossible to have a stable region, for each structure, where the clock field is flat like in the figure 5.20. Please, must be clear that this doesn't depend on the implemented model, since it follows perfectly the numerical solution, but it is an intrinsic limit of the circuit. Honestly speaking, there is no reason to be surprised, since the electric field decreases outside the wanted region like $1/r^2$, so it is not selective. As a consequence, what we are able to conclude is that, like it is demonstrated with other simulations with higher length for each structure, this effect disappears.

Below instead, in the figure 7.15, it is really interesting to notice and highlight the zone between two consecutive structures of electrodes where the field has transition regions, mostly where there is the change of states from two consecutive structure with opposite mode, hold and reset. This shows up how much is difficult to have a very strong and precise control of the field to drive in the right way this type of technology.

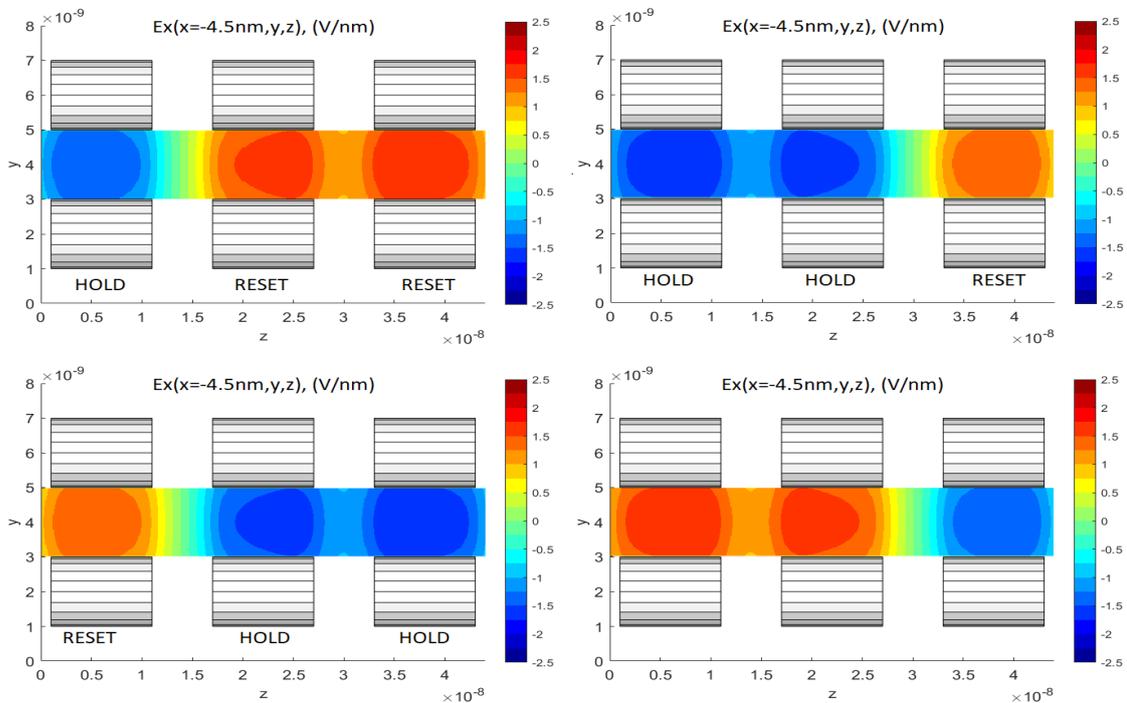


Figure 7.15: Vertical electric field, but with a 2D view, for each steps of the timing diagram shown in figure 7.13 by comparing, as always, Matlab and Comsol.

At this point becomes mandatory to try to increase the length of each electrodes' structure to see if a better control of the electric field is reached and if the flat zone, where the molecules are placed, appears again. Since the aim is to look at this phenomena, just the result for the time step t1 will be shown. Like it is possible to appreciate in the figure 7.16, a better and stabler field is obtained. This is a confirmation of the consideration done before. Also for this simulation, an incredible resolution and precision has been

reached.

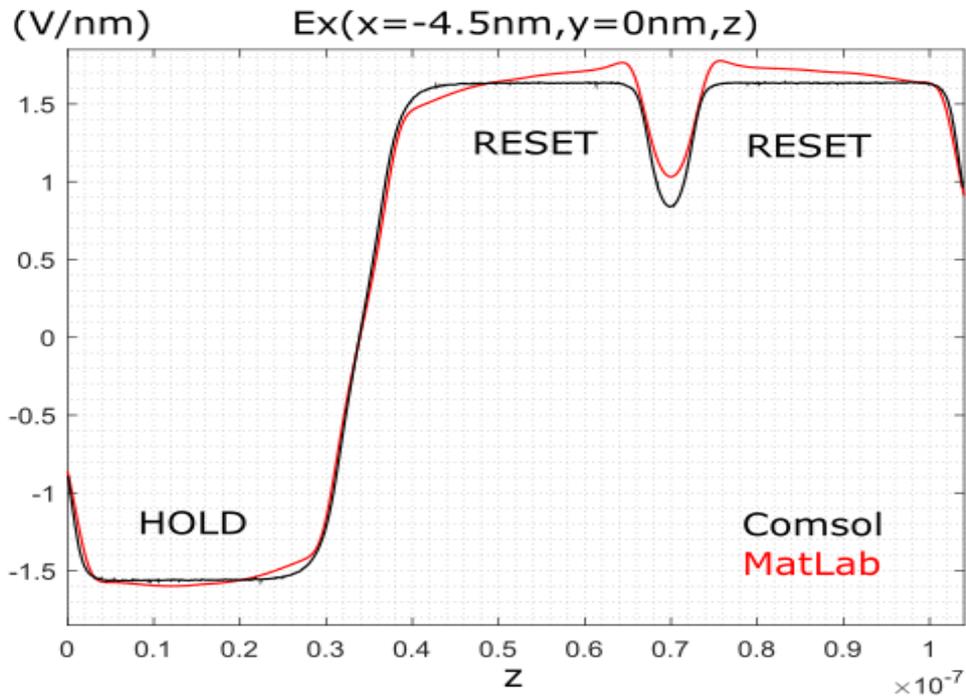


Figure 7.16: Simulation of the wire with a distance between electrodes' structures equal to 6 nm and a length for each of them equal to 30 nm, by imposing the time step t_1 .

It could be also interesting to play with the parameter D_s in the algorithm, that describes the distance between consecutive structures of electrodes and at the same time, it is very sensitive to technological limits. By looking at the figure 7.17, by imposing a distance between electrodes equal to 10 nm, it is evident that, by increasing the value of D_s , there is an important decrease of the value of the electric field between two different structures. So, as a consequence, it must be as small as possible to keep a strong control and not have a lot of molecules in that place, since, small electric field means small control of electrons inside molecules and as a consequence, not well-defined state.

Anyway, also for this case, the implemented model on Matlab follows with an incredible resolution the numerical solution got from Comsol. By having demonstrated that with different simulations with distinct parameters for each structure the final result is very precise, we can conclude that in these conditions the model works, of course, with its limits.

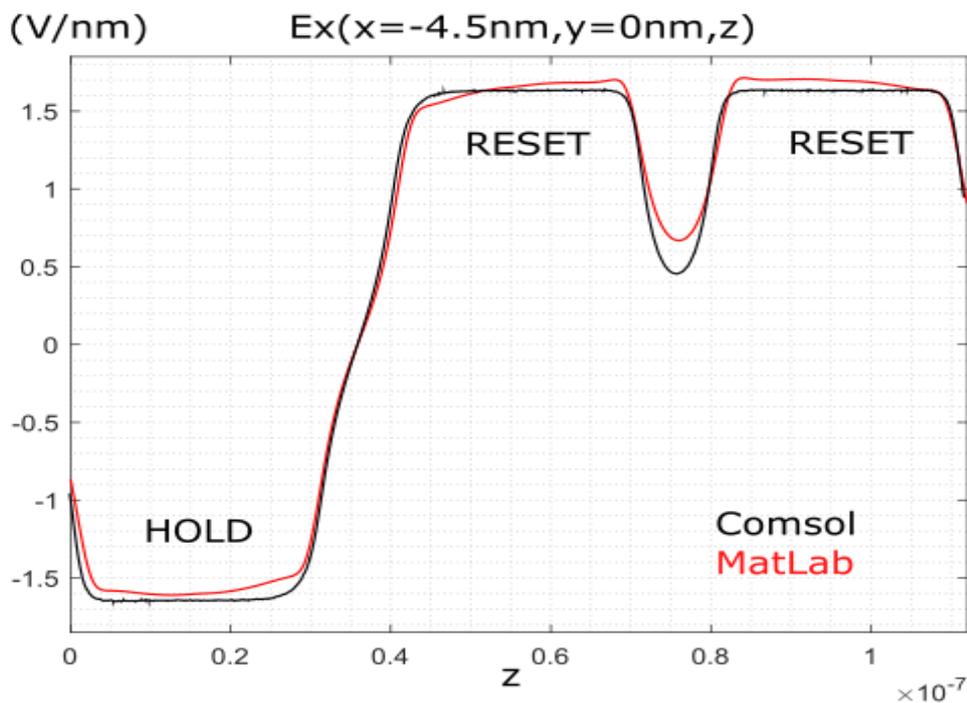


Figure 7.17: Simulation of the wire with a distance between electrodes' structures equal to 10 nm and a length for each of them equal to 30 nm, by imposing the time step t_1 .

Now, it's almost mandatory to try to change the orientation in 3D space for structures, to build much more complex devices, like the L, T and Crossroad, got by combining our elementary modelled structures of wires. Below, in the figure 7.18, it is possible to appreciate what we are speaking about. These new objects are very useful to enlarge our degrees of freedom to simulate real circuit, like the Majority Voter. As a consequence, also other fundamental gates like AND, OR and so on, since, just to remember, the latter are two special cases of the Majority Voter it-self. Of course, and this is something that it will be shown below, by combining different structures of electrodes, the relative error with respect to Comsol simulator will increase, but, as always, it is enough that the model works properly, with its limits, in the safe region where molecules have to be placed.

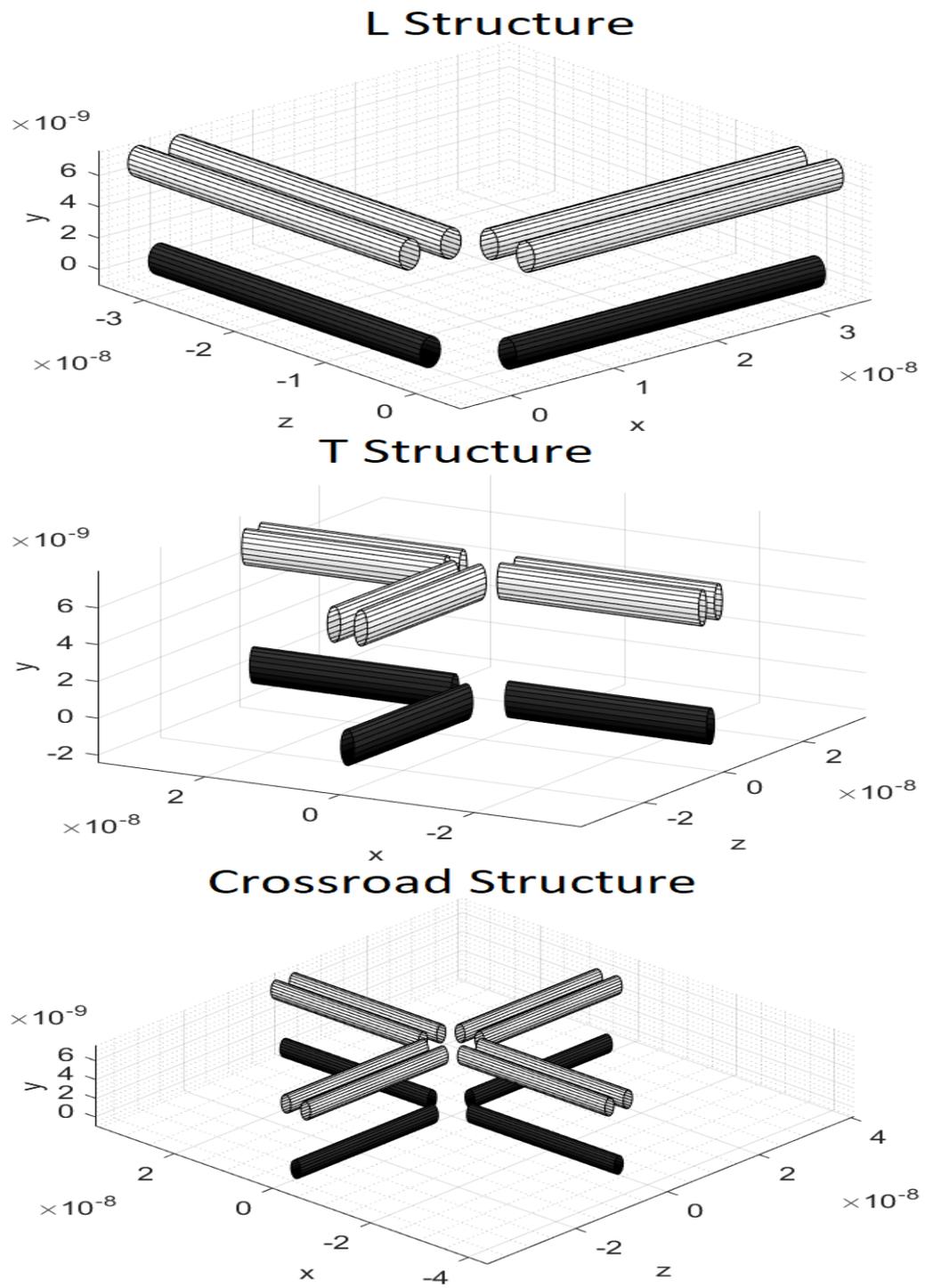


Figure 7.18: L, T and Crossroad structures respectively starting from the top.

7.6.1 Simulation of L, T and Crossroad structures

The structures shown above, in the figure 7.18, have been studied for two different lengths: 30 nm by looking at the figure 7.19 while 10 nm in the figure 7.20, by wanting to test the model in two conditions completely opposite. By starting from that regarding 30 nm, it is possible to appreciate that an incredible resolution has been got. There is just an overestimation for the T and Crossroad structures in the middle, for z and x equal to 0 nm, since in this region there is a combination of different boundary effects since the model has to take into account the end of each wire, which is not absolutely an easy task for it. This is demonstrated by looking at the figure 5.21, and in the specific on the right down where each structure finishes, there is a difference between fields got from Comsol and Matlab, since to model precisely boundary effects it is almost impossible. So, higher is the number of structures that build the device under test, higher will be the relative error just in that region since by analysing an electrostatic case, the superposition effect is present. Anyway, in general, by neglecting this region where there is just an overestimation, the results respect completely the expectations and we can consider this a very strong and important starting point for complex structures.

By looking at the figure 7.20, where simulations of wires with length equal to 10 nm are shown, it is a bit worse always in the same region mentioned above: in the middle where different electrodes' structures' match the others. This is demonstrated by considering the plot shown in the figure 5.20, where it's possible to appreciate that, with shorter wires, it is more difficult, by definition, to describe the relative boundary effects.

In fact by comparing the figure 5.21 and the figure 5.20, in the latter the error at the boundary is a bit higher. The fundamental aspect is that, first of all, everything is coherent since the superposition is still true also by combining different structures of electrodes with distinct orientation, by definition for electrostatic physical model. Then, there is just an overestimation in a particular region, while on the other points of the considered volume, there is a fantastic match. Now, since the aim is to define a model, we can conclude that it works, of course, with its limits.

Anyway, another alternative solution is proposed for wires with a length equal to 10, which is shown in the figure 7.21, where the model can get less precise value in the centre of each structure underestimating the field, but by privileging the boundary effects. Remember that it is possible thanks to the compensation tool builds for the charge model in the configuration number 3, where the degrees of freedom mentioned in the relative section, are used. As a consequence, a more precise description near to the end of each structure is got and in the critical region, for z and x coordinates equal to 0, the total error decrease dramatically.

Unfortunately, this is another trade-off since it is impossible to catch both even if at this point of the thesis, it is something for which we are inured.

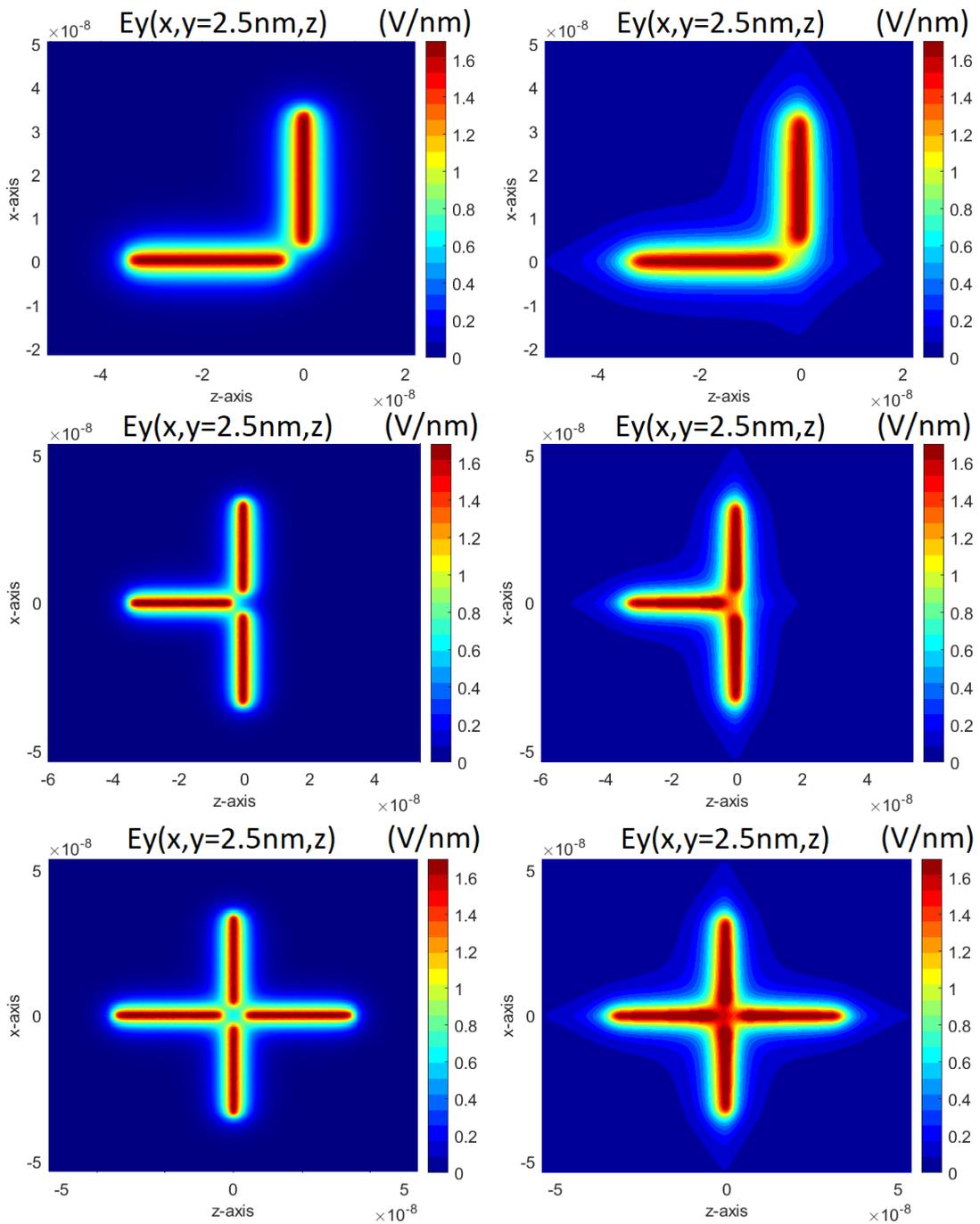


Figure 7.19: Comparison between Comsol(on the left) and Matlab(on the right) for L, T and crossroad structures, respectively starting from the top, with a length for each wire equal to 30 nm.

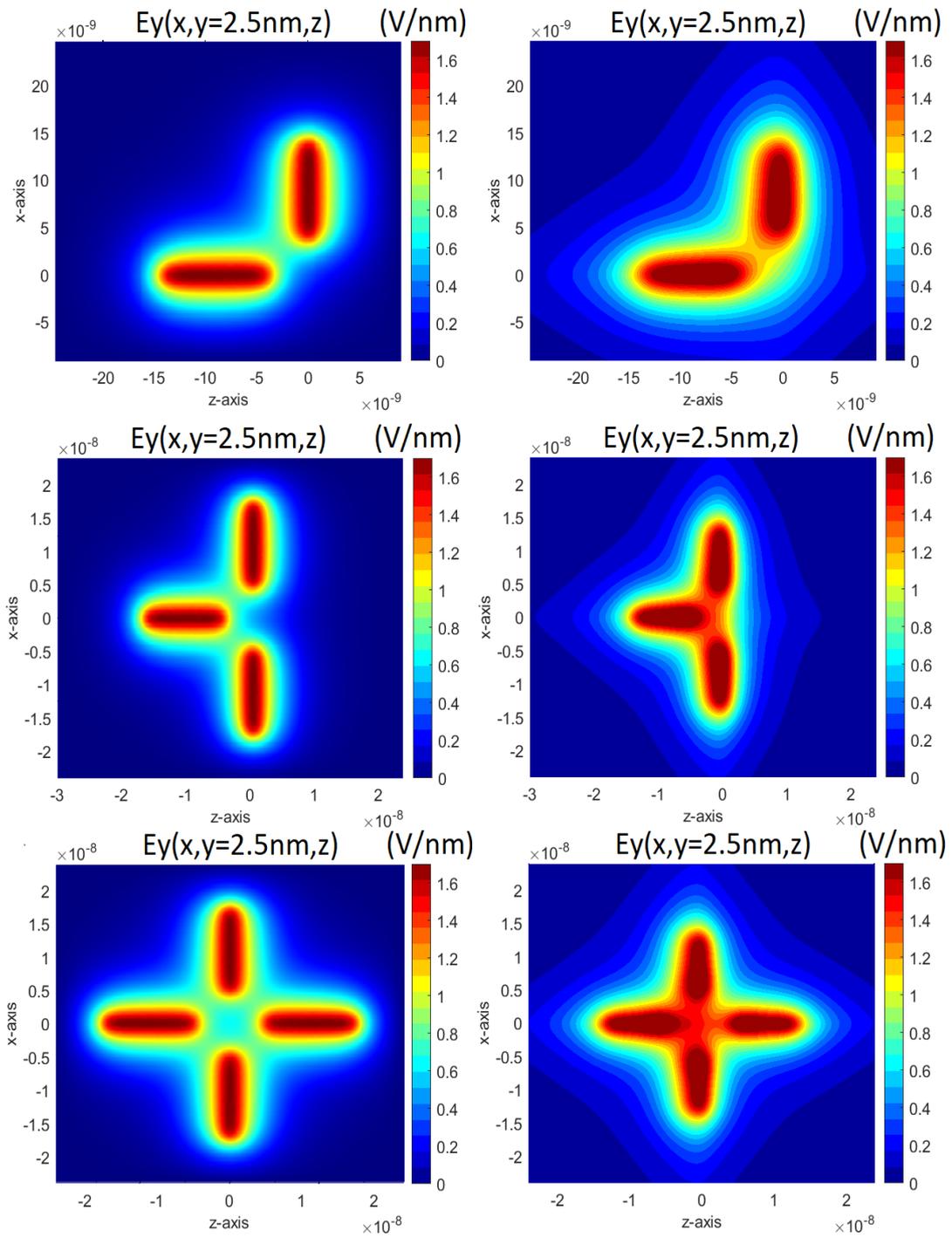


Figure 7.20: Comparison between Comsol(on the left) and Matlab(on the right) for L, T and crossroad structures with a length for each wire equal to 10 nm.

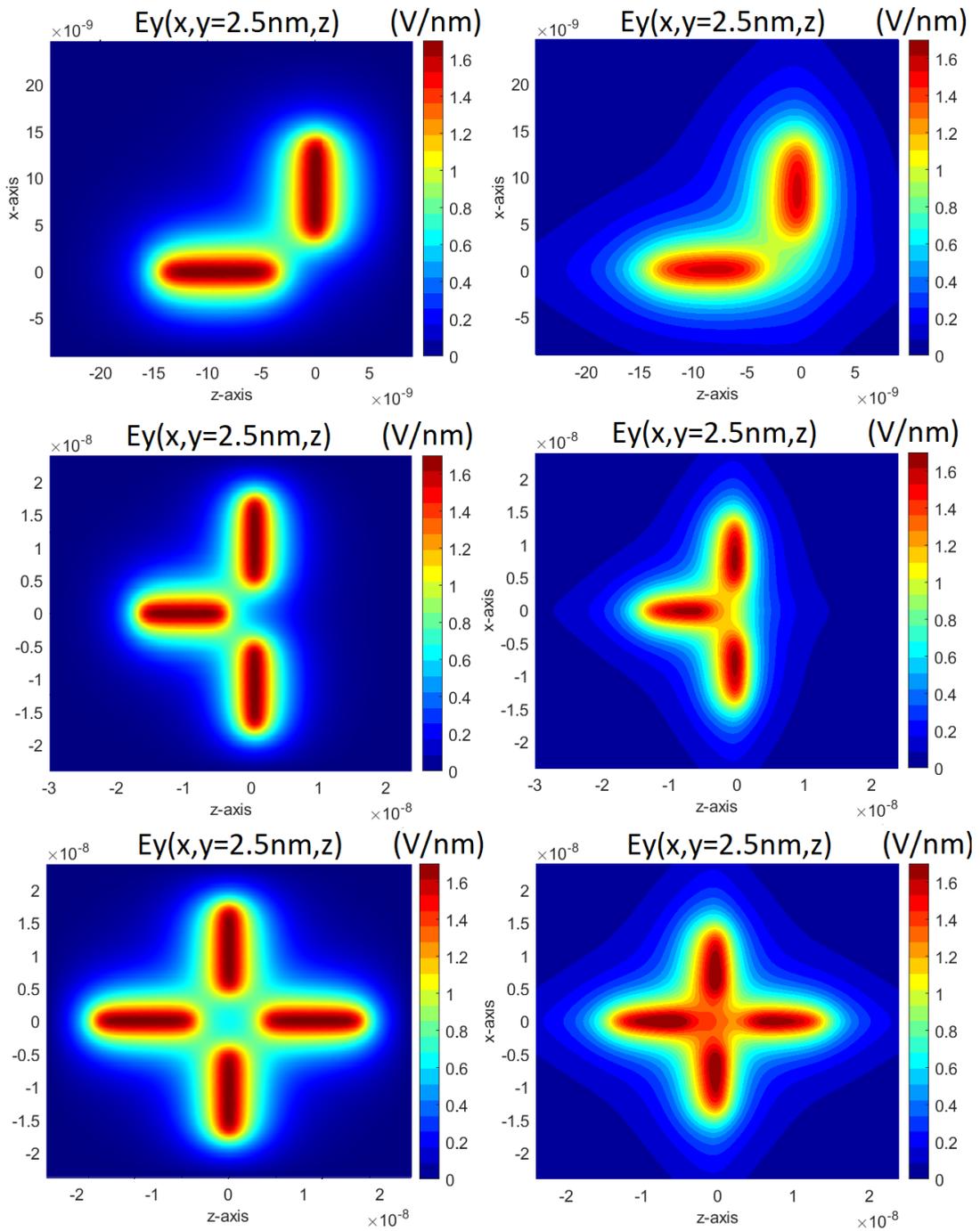


Figure 7.21: Comparison between Comsol(on the left) and Matlab(on the right) for L, T and crossroad structures, respectively starting from the top, with a length for each wire equal to 10 nm but with boundary effect's optimization.

7.6.2 Simulation of L, T and Crossroad structures with overlapping

In this case, the model is much more under stress concerning the previous case, but as it has been demonstrated [4], the structures shown in the figure 7.22 are practically mandatory to build really the Majority Voter and the relative gates got from it.

The same concept mentioned for the structures without overlapping is still true in this case. What happens is that, by overlapping the structure, manage the boundary effects is practically impossible. Anyway, how in the case analysed above, for longer wires a very precise results have been got while with shorter structures of electrodes the model implemented on Matlab loses in term of resolution. Please, pay attention. This doesn't mean that it doesn't work since, where is needed, that is where the molecules have to be placed, the same field between Comsol and Matlab is got, of course as always with the relative tolerance.

In the figure 7.23 simulations with wires long 30 nm have been performed, and again, since with this length boundary effects are easier to model, a very good match is obtained.

In the figure 7.24, the resolution is lower but again, where molecules, theoretically, have to be placed the field respect the results got from the numerical simulator, Comsol. Notice that, in this case, only the results for the model optimized for the boundary effects have been upload since in the case without optimization, the resolution is unacceptable.

Notice that all plots considered in this last section, dedicated to the L, T and crossroad devices, have a vertical y-coordinate equal to 2.5 nm. The centre of the clock electrode is placed in $y=0$ nm, so this means that, by having a radius equal to 1 nm, the field is plotted at 1.5 nm from its surface, which is the best place where deposit the molecules to guarantee a very good control of the electrons. This is just a reminder since it is not the first time that this aspect is mentioned. Instead for the other geometrical data, the typical values have been considered: for the vertical distance D_v , 6.6 nm, while for the horizontal, D_w , 4 nm. The same for simulations performed for the Majority Voter.

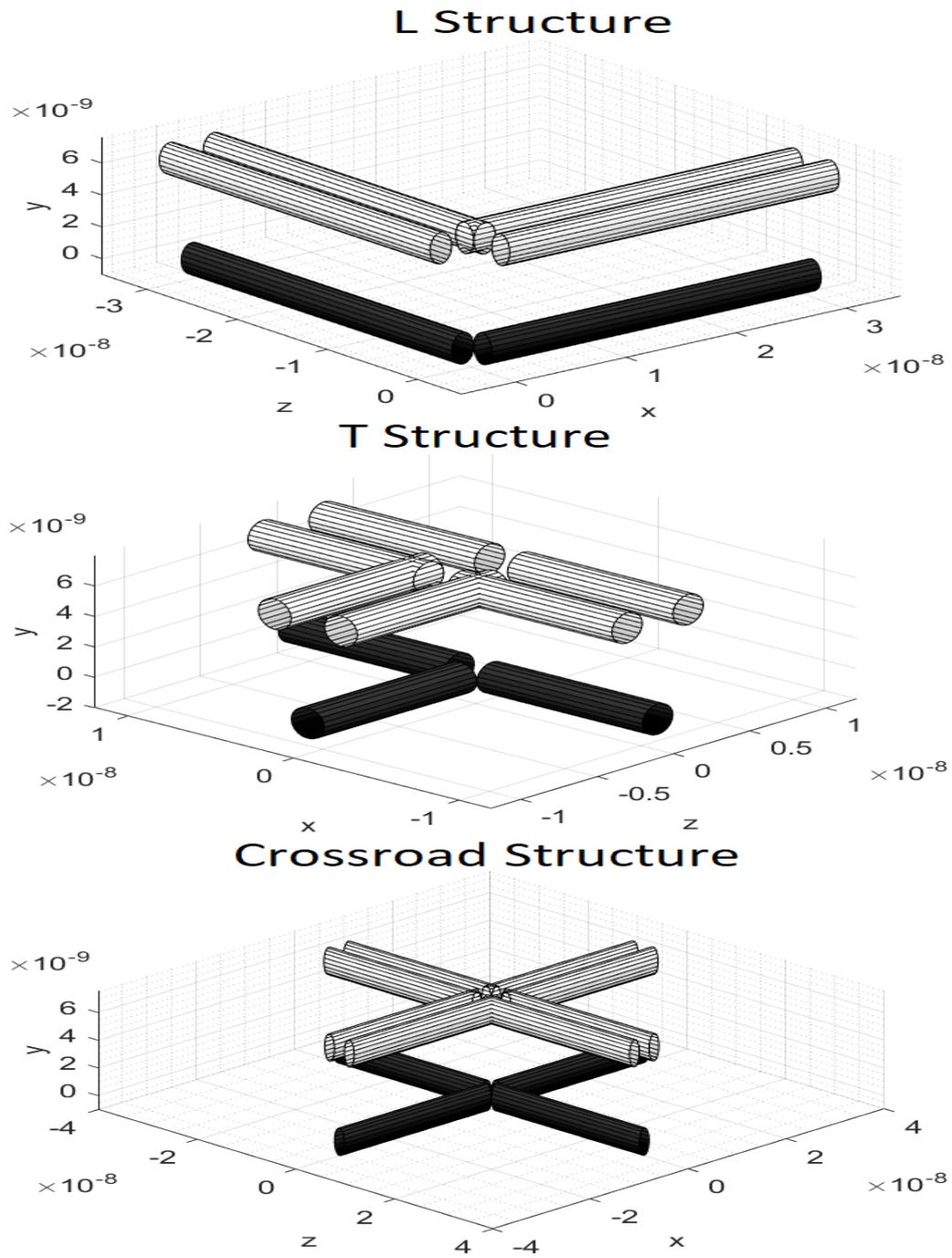


Figure 7.22: L, T and Crossroad structures with overlapping respectively starting from the top.

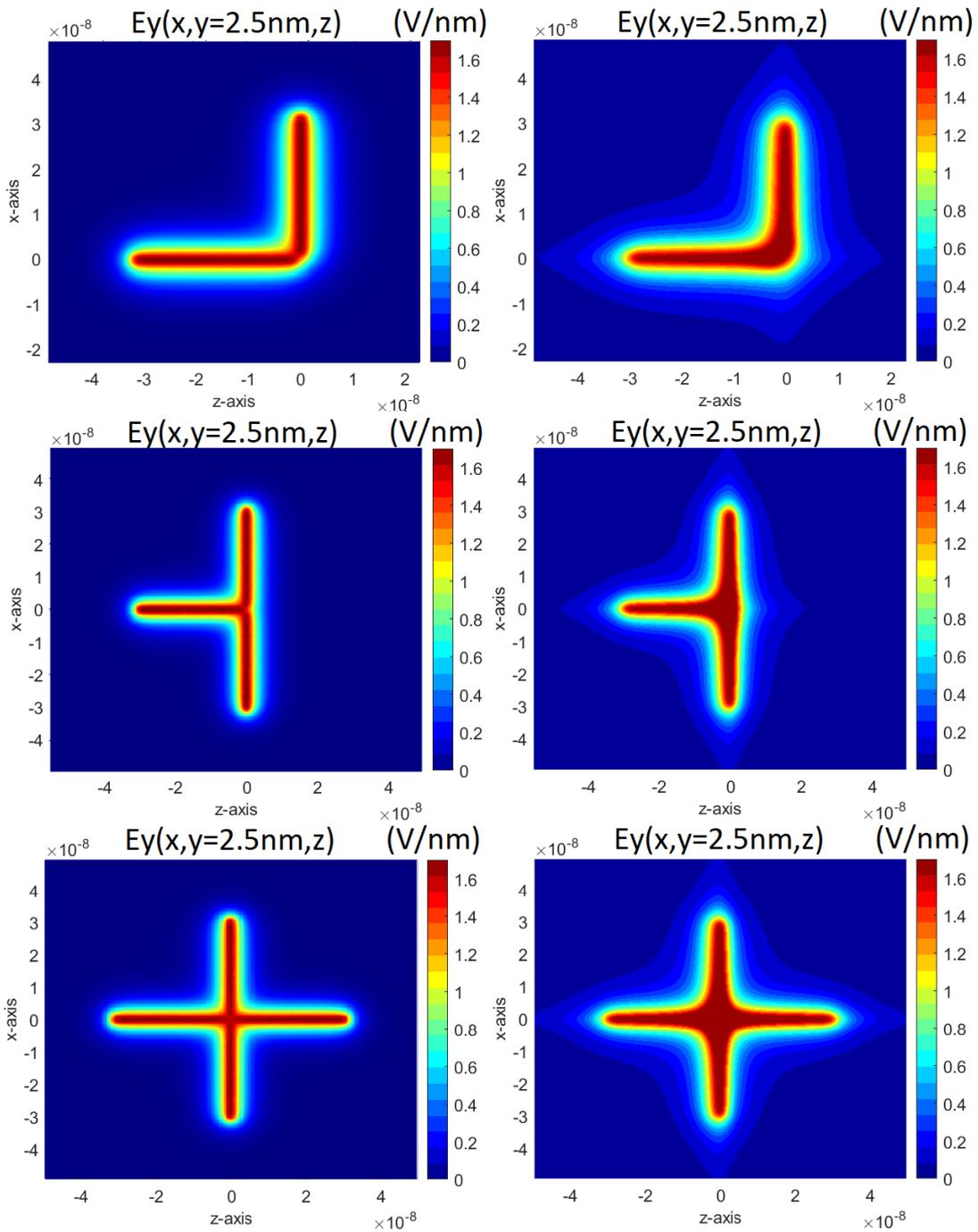


Figure 7.23: Comparison between Comsol(on the left) and Matlab(on the right) for L, T and crossroad structures with overlapping, respectively starting from the top, with a length for each wire equal to 30 nm.

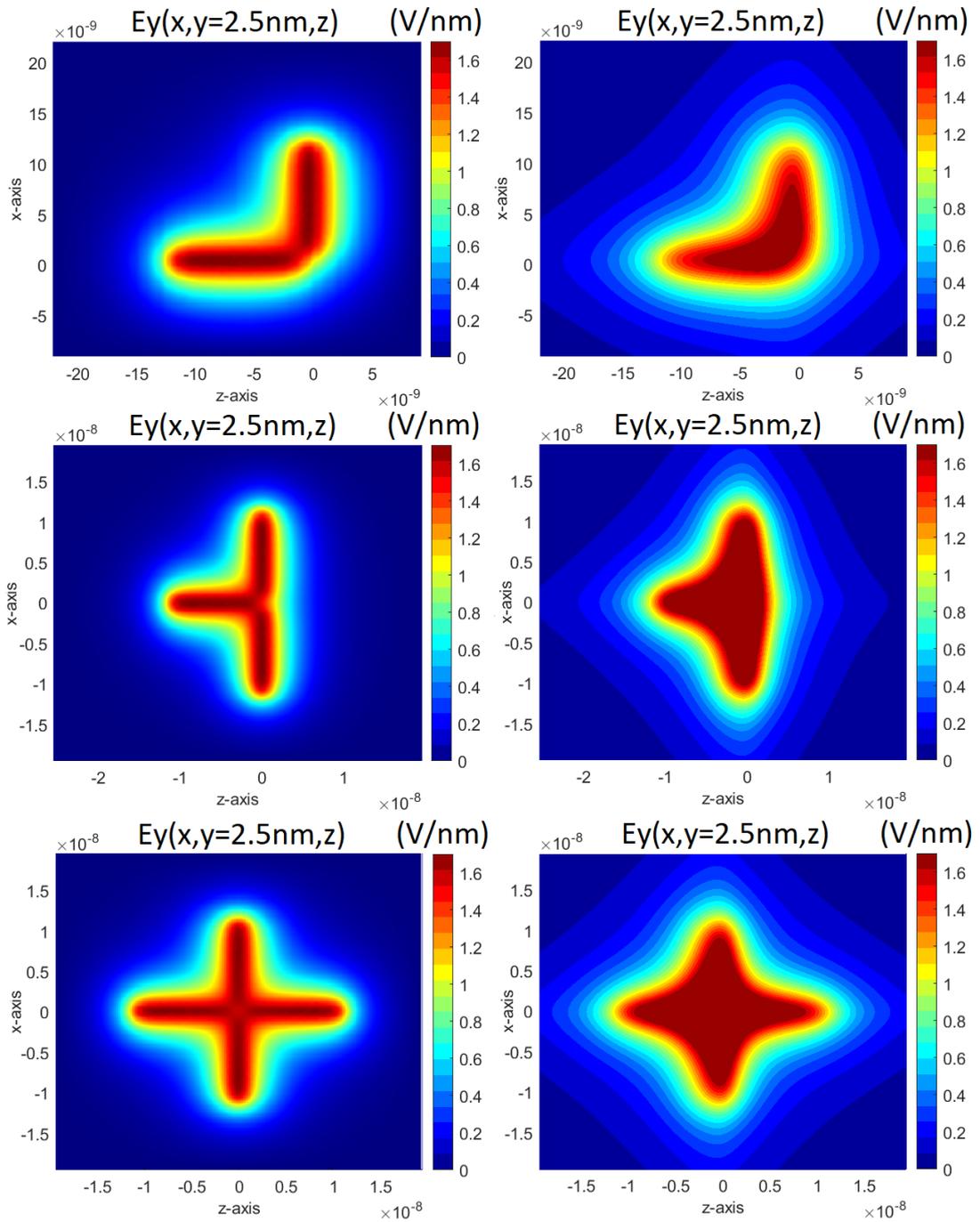


Figure 7.24: Comparison between Comsol(on the left) and Matlab(on the right) for L, T and crossroad structures, respectively starting from the top, with overlapping with a length for each wire equal to 10 nm but with boundary effect's optimization.

At this point, let's go on in the last and more complex case: the majority voter, in order to discover really if this model can be fused or not with the molecular simulator, SCERPA.

7.7 Simulation of the Majority Voter

By remembering what has been explained during this thesis, starting from the introduction, it's known that, to guarantee a good behaviour of this device a multi-phase organization has to be imposed. By considering the figure 7.25, it is possible to understand how the phases are organized and managed, by highlighting that of course, this image is just qualitative, in order to explain how the simulation is arranged.

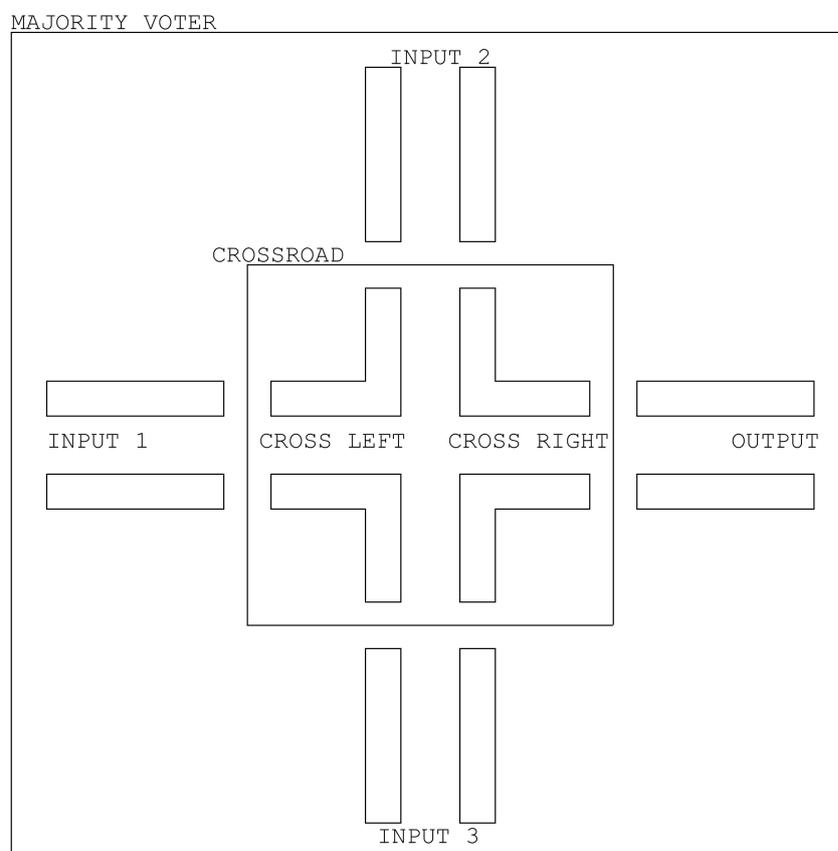


Figure 7.25: Qualitative scheme for the Majority Voter used just to explain how the simulation is arranged.

Now, before to go on, it's almost mandatory to impose a nomenclature in order to explain what is happening as clear as possible:

- V_I for $I=1,2,3$ is the potential associated with the structures of electrodes dedicated to the propagation of the different inputs towards the Crossroad structure, which is used for the output's evaluation;
- V_{CL} is the voltage that drives the left part of the Crossroad structure;
- V_{CR} is the voltage that drives the right part of the Crossroad structure;
- V_O is the potential used to manage the external structure of electrodes, called OUTPUT, used to put out the final result from the circuit.

By wanting to drive correctly the circuit in order to obtain a right final output, three different time steps have to be imposed:

- In the time step $t1$, all inputs run towards the Crossroad. As a consequence, all structures dedicated to the inputs are in hold state, while the others two, Crossroad and output, in reset;
- In the time step $t2$, the evaluation is performed. So the only thing that changes is that the crossroad structure is in hold condition, in order to perform the evaluation by combining all inputs together;
- In the last time step, $t3$, inputs' electrodes came back in reset state, in order to be ready for another evaluation with other inputs, while the structure dedicated to the output is brought in hold state in order to put out the result.

All these considerations may be synthesised in the following table 7.3:

Time step	V_I , (V)	V_{CL} , (V)	V_{CR} , (V)	V_O , (V)
$t1$	+8	-5	-3.5	-8
$t2$	+8	+2	+3.5	+8
$t3$	-8	+2	+2	+8

Table 7.3: Optimal potentials used during the simulation of the Majority Voter through the different time steps [4].

Remember that, independently from the absolute value, according to the convention used in this report, when a positive voltage is applied to a specific structure, the hold state is imposed while, on the other hand, with a negative voltage, the reset mode. At this point, below are shown the results got for two different configurations: the first with a crossroad structure composed from electrodes with a length equal to 5 nm, shown in the figure 7.26 while the second, shown in the figure 7.28, a length for crossroad's wires equal to 3 nm has been set. Instead, the electrodes used to realise the inputs and the output are long 10 nm, and the distance between different structures D_s , is equal to 1nm. Objectively, the model is strongly under stress but let's see what happens.

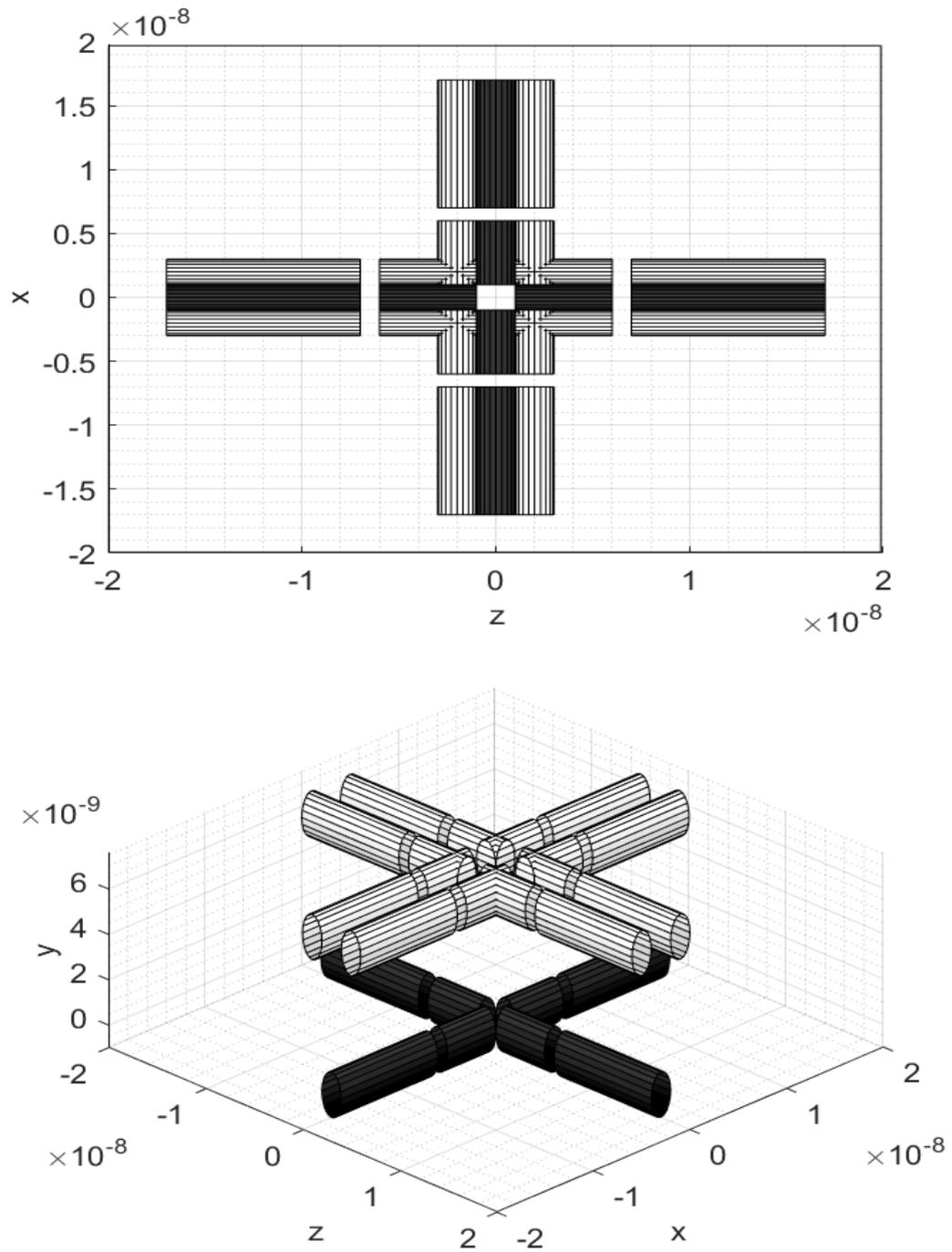


Figure 7.26: Majority voter circuit build with a crossroad structure long 5 nm for each wire. On the top, it is shown from 2D point of view while on the bottom from 3D.

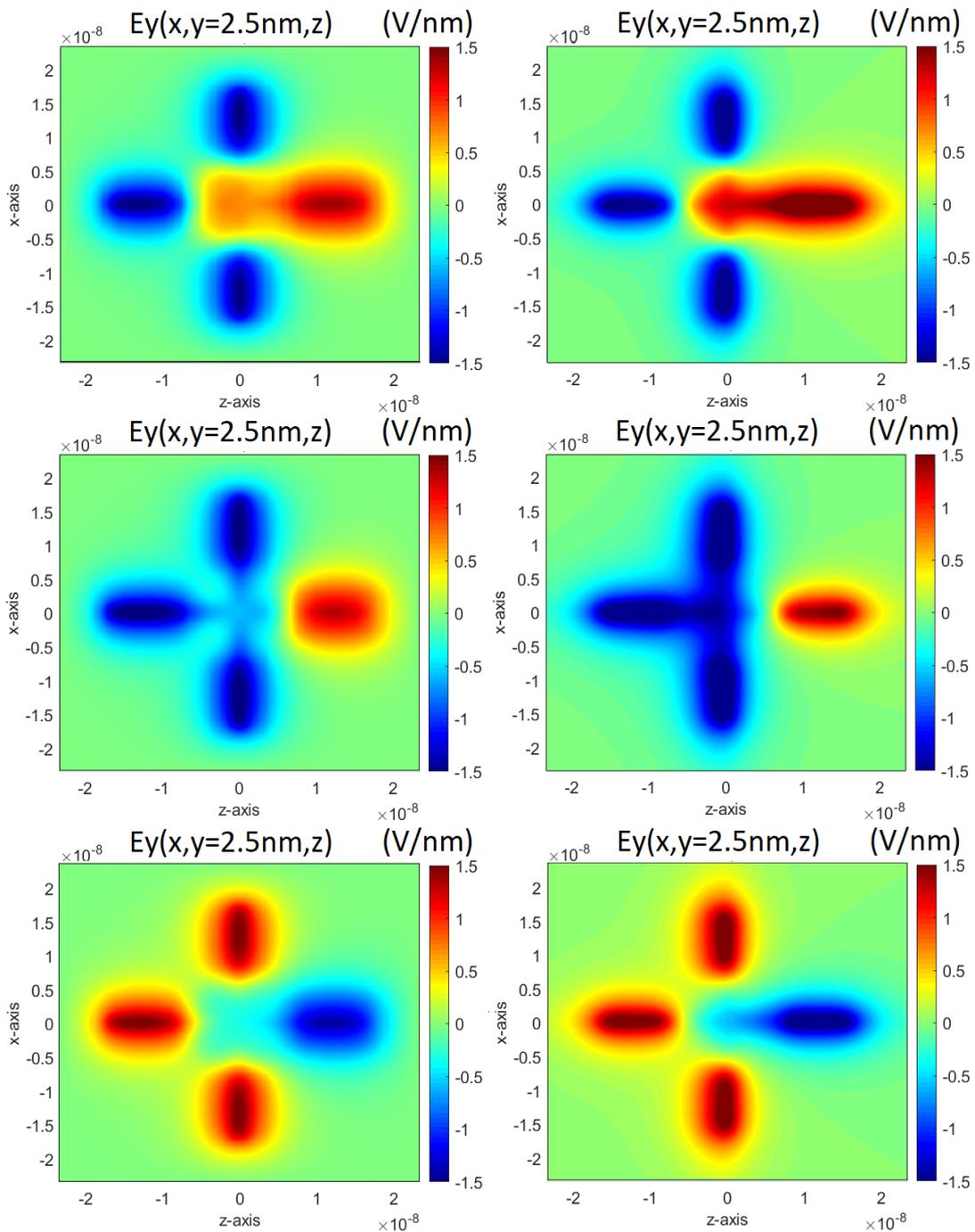


Figure 7.27: Simulation for the majority voter with a crossroad structure long 5 nm. By starting from the top towards the bottom, the results for time steps t_1 , t_2 and t_3 are shown. Comsol's results on the left while Matlab's results on the right.

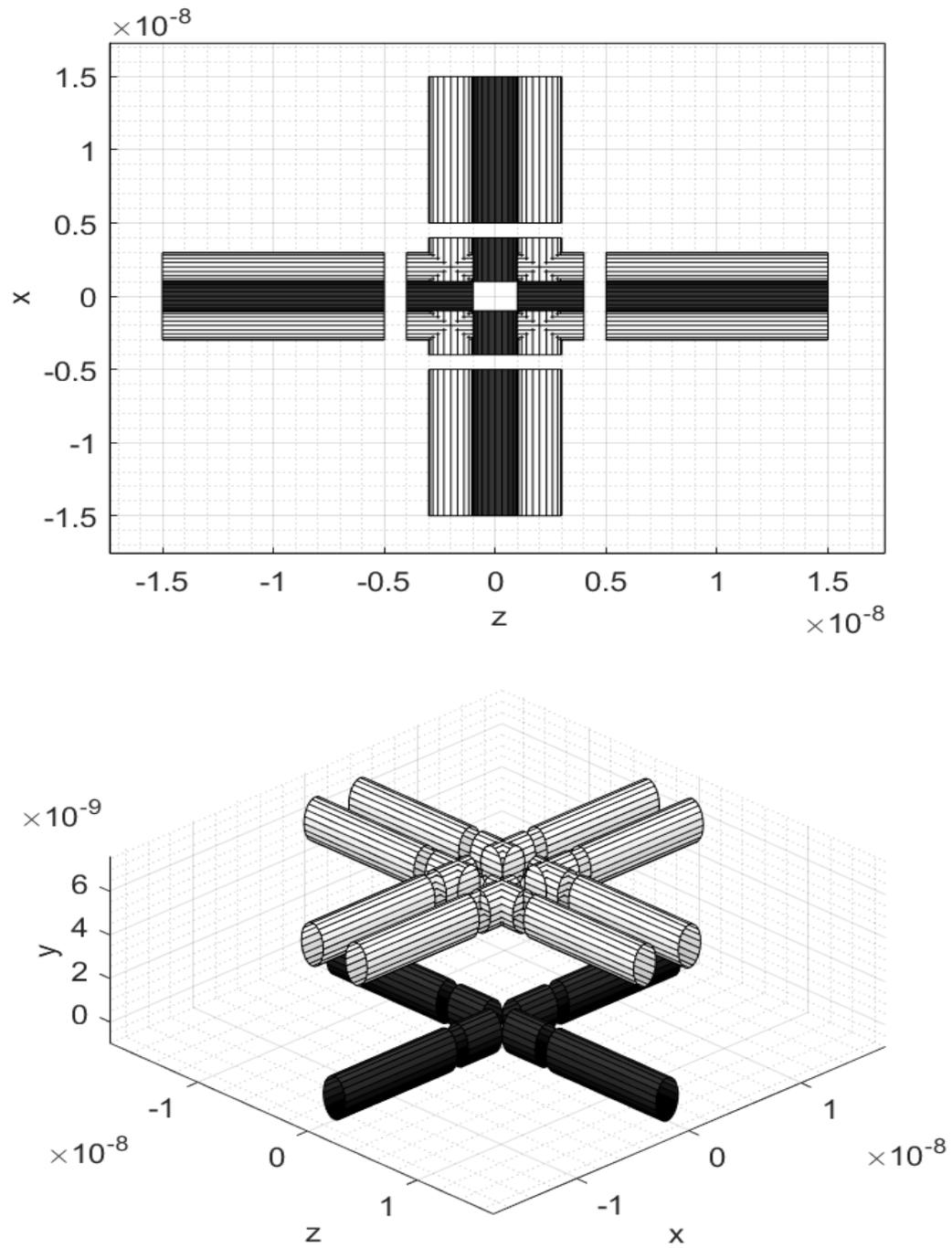


Figure 7.28: Majority voter circuit build with a crossroad structure long 3 nm for each wire. On the top, it is shown from 2D point of view while on the bottom from 3D.

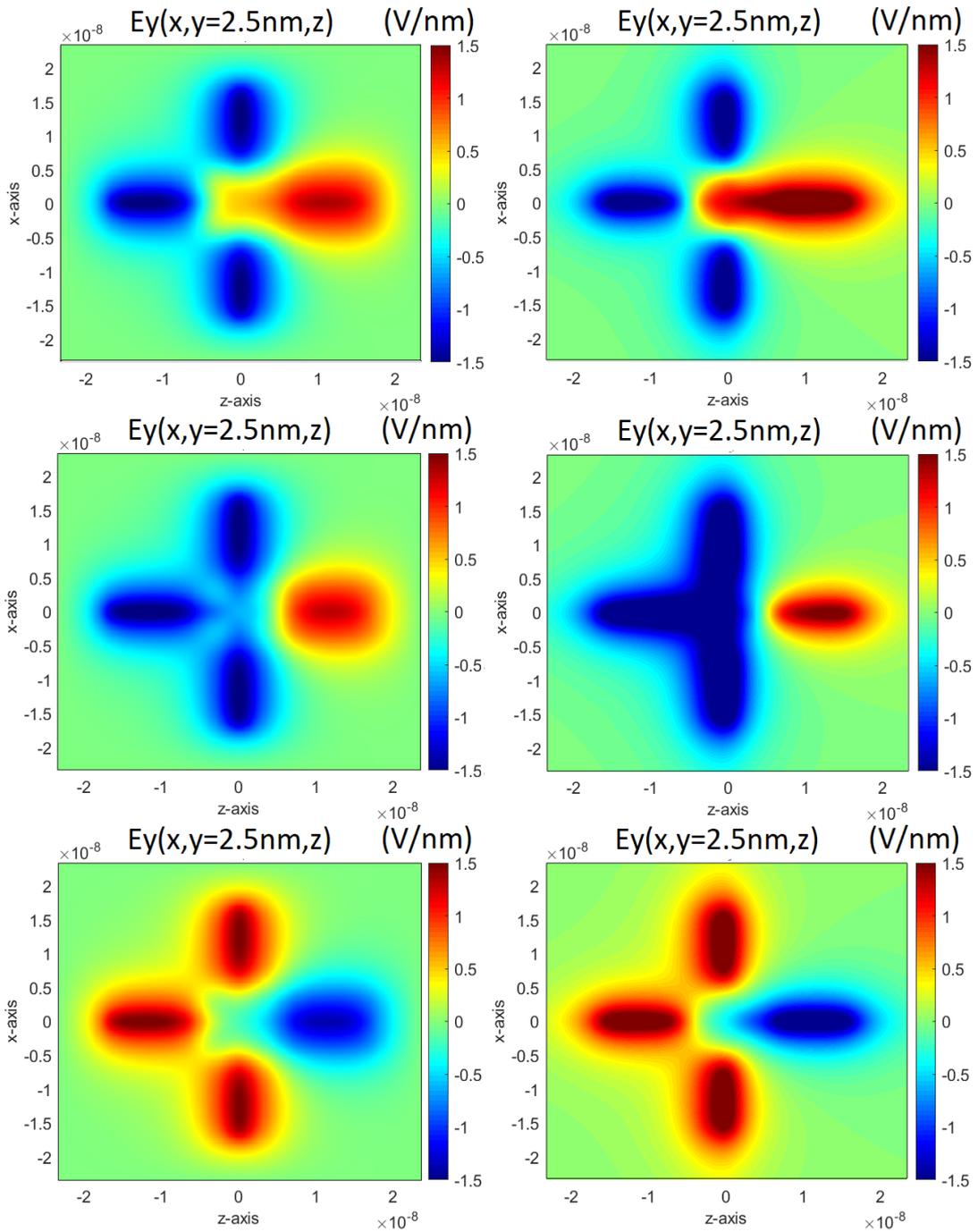


Figure 7.29: Simulation for the majority voter with a crossroad structure long 3 nm. By starting from the top towards the bottom, the results for time steps t_1 , t_2 and t_3 are shown. Comsol's results on the left while Matlab's results on the right.

Like it is possible to appreciate by looking at the figure 7.27 and 7.29, an incredible, maybe also unpredictable, resolution has been obtained. Unpredictable since a very complex and complete circuit is under test. The only time step which suffers a bit is t_2 , that is in the middle of the figures mentioned before. In that case, there is an overestimation since boundary effects sum with the same sign, since, in this time steps, both inputs and crossroad structures are in hold, with a positive applied voltage. On the other hand, independently from the length of the crossroad structure, for the time steps t_1 and t_3 is almost impossible to distinguish the results got from Comsol and Matlab.

There is nothing to be surprised since, like happens in the figure 7.16 and others, where there are consecutive electrodes with an opposite applied voltage, the relative error disappears since the model is summing two field with opposite sign but equal magnitude, while with two consecutive structures in the same state, so with the same sign of the applied potential, the field is summed and not deleted. This is justified, mathematically speaking, the obtained results.

Surely, by relaxing the geometrical parametric and so, by using longer wires with a bigger distance D_s between them, the final results would be also better. As a consequence, it is reasonable to consider these last two simulations like two worse cases. By wanting to conclude, since in this complex case the model works enough well, let's conclude that is reliable and ready to be fused with the numerical simulator SCERPA, by accepting its strengths, like ease of use, low computational cost and a very powerful control and weakness, like leak of resolution when complex circuits are under test.

Chapter 8

Conclusion

Now that the majority voter has been modelled, tested and verified, it's able to include it inside SCERPA by simulating the molecular circuits. As a consequence, by having obtained good results for this fundamental circuit, also others gates may be considered, like the AND and the OR, since, as it has been told a lot of times, these latter are just a special case of the majority voter. As a consequence, also much more complex devices can be realised, like the half-adder and the full-adder.

Probably, by combining a lot of different gates the resolution could decrease a bit, but by remembering which is the aim of the model with its limits, this is not a big problem if the final resolution is, where needed, enough high. Remember that, like it has been demonstrated in the chapter 5, by having performed a lot of parametric analyses, this model doesn't work just for specific data, but it has an incredible sensibility to different parameters, like D_w , which is the horizontal distance between the two upper wires; D_v which is the vertical distance between the two upper wires and the clock electrodes and so on. As a consequence, by combining different geometrical data, an enormous number of cases can be simulated.

By having collected a huge amount of experience with this topic, it is my suggestion to use this model to get quickly an idea of the electric field for specific combinations of the structures of electrodes. As a consequence, after having designed and tested the circuit with the molecular simulator, it could be a good idea to run the numerical simulator Comsol, at least once at the end, to check that everything is fine and correct. In the electronic world, this type of approach is not absolutely new. For instance, during the design of a digital circuit, in the first step all delays, like the raise, falling, set-up, etc, are taken into account by hand from the designer. Then, before to affirm that the circuit really works, a numerical simulator is used at least once, like for instance Synopsis.

Bibliography

- [1] Yuri Ardesi. Energy analysis and bistability study of molecular fcn. *Master Thesis*, December 2017.
- [2] M. Ottavi. F. Lombardi V. Vankamamidi. Two-dimensional schemes for clocking timing of qca circuits. *Computer-Aided Design of Integrated Circuits and Systems, IEEE Transactions on*, vol.27, no.1., 2008.
- [3] Roberto De Luca. Fili, finiti e infiniti, uniformemente carichi. *DIIMA - Università degli Studi di Salerno*, April 1993.
- [4] Alemanno Giorgio. Study of clocking system for molecular fcn. *Politecnico di Torino*, Academic Year 2018/2019.
- [5] Wang Ruiyu. Analysis and modulation of molecular quantum-dot cellular automata (qca) wire. *Politecnico of Turin*, 2017.
- [6] Jack Vanderlinde. Classical electromagnetic theory, second edition. *University of New Brunswick*, pag 152, 2004.
- [7] Prof. Mariagrazia Graziano. Lecture about fcn - charge distribution between dots in function of the clock field. *Politecnico di Torino*, Academic Year 2018/2019.
- [8] P.D. Tougaw C.S. Lent and W. Porod. Bistable saturation in coupled quantum dots for quantum cellular automata. *Appl. Phys. Lett.* , vol. 62, page 714, 1993.
- [9] P.D. Tougaw C.S. Lent Bernstein and W. Porod. Quantum cellular automata. *Nanotechnology*, vol. 4, page 49-57, 1993.
- [10] L. Zoli. Active bis-ferrocene molecules as unit for molecular computation. *PhD dissertation*,, 2010.
- [11] M. Iurlo L. Zoli S. Kumar M. Piacenza F. Della Sala F. Martino G. Maruccio R. Rinaldi F. Paolucci M. Marcaccio P.G. Cozzi V. Arima and A.P. Bramanti. Toward quantum-dot cellular automata units: thiolated-carbazole linked bis-ferrocenes. *Nanoscale*, vol. 4,, pages 813-823, 2012.

- [12] M. Graziano A. Sanginario V. Cauda D. Demarchi A. Pulimeno and G. Piccinini. Bis-ferrocene molecular qca wire: ab-initio simulations of fabrication driven fault tolerance. *IEEE Transactions on Nanotechnology*,, 2013.
- [13] M. Graziano A. Pulimeno and G. Piccinini. Molecule interaction for qca computation. *IEEE NANO2012 12th International Conference on Nanotechnology, Birmingham (UK)*,, 20-23 August 2012.
- [14] P.D. Tougaw C.S. Lent Bernstein and W. Porod. Quantum cellular automata: the physics of computing with arrays of quantum dot molecules. *PHYSCOMP 94 Proceed- ings*, pages 5-13, 1994.
- [15] P.D. Tougaw and C.S. Lent Bernstein. A device architecture for computing with quantum dots. *Proceedings of the IEEE 85*,, pages 541-557, 1997.
- [16] I. Amlani R.K. Kummamuru R. Ramasubramaniam G. Toth C.S. Lent G.H. Bern- stein A.O. Orlov and G.L. Snider. Experimental demonstration of clocked single- electron switching in quantum-dot cellular automata. *App. Phys. Lett.*, Vol. 77, pages 295-297, 2000.
- [17] G. Toth and C.S. Lent. Quasi-adiabatic switching for metal-island quantum-dot cellular automata. *Journal of Applied Physics*, vol. 85,, pages 2977-2984, 1999.
- [18] M. Ottavi. F. Lombardi V. Vankamamidi. Clocking and cell placement for qca. *IEEE-NANO*, vol. 1,, pages 343-346, 2006.
- [19] C.Voci P.Mazzoldi, M.Nigro. Fisica, vol. ii. *Ed. EdiSes*, 1998.
- [20] E. S. Kochanov Yu. Ya. Iossel' and M.G. Strunskiy. The calculation of electrical capacitance. *Foreign Technology Division*, pag 65, August 1971.
- [21] Jennah K. Kriebel Ralph G. Nuzzo George M. Whitesides Christopher Love, Lara A. Estroff. Self-assembled monolayers of thiolates on metal as a form of nanotech- nology. *Department of Chemistry and the Frederick Seitz Materials Research Labora- tory, Universitary of Illinoius-Urbana-Champaign and Department of Chemistry and Chemical Biology, Harvard University*, March 2005.