Technische Universität München

Fakultät für Electrotechnik und Informationstechnik

Institute for Nanoelectronics

Prof. Dr. Paolo Lugli

Diploma Thesis

Circuit approaches to Physical Cryptography

Author: Tamara Jun

Supervisor: Dr. György Csaba

April 2009

Circuit approaches to Physical Cryptography

Abstract

Nowadays keeping information safe is one of the most important research topics in Computer Science and Information Technology. Consequently, many techniques of Cryptography and Security are continually being proposed.

In this thesis we will investigate a novel approach to Cryptography, **Physical Cryptography**: This suggests the application of optical and electrical nanostructures to cryptography and security, to complement standard, algorithmic procedures. Using physical objects enables security solutions with novel features.

This thesis focuses on the introduction and analysis of two specific techniques related to Physical Cryptography:

SHIC (Super High Information content) systems allow the user to keep a high amount of information safe from external attacks: The architecture of these circuits forces an extremely slow-read out of the data. This specific characteristic prevents the system from being completely characterized by the attacker when this has gained temporal access to the circuit.

UNIQUE objects are the other field to study in this work: Here, only small amount of information is protected. Its fast internal speed makes it physically impossible to being reproduced or imitated by an intruder.

We will present two techniques and propose possible physical circuits that implement SHIC and UNIQUE: SPICE and Sentaurus TCAD simulators will be used for making analog-circuit and device-level simulations respectively, in order to study and conclude the feasibility of both proposals.

Acknowledgements

Before giving formal start to this diploma thesis I would first like to thank Professor Paolo Lugli for letting me take part of the Nanoelectronics Research Group and would like to especially thank my supervisor György Csaba for all his support and kind guidance since my work in the Institute began some months ago.
I'm really grateful for all the friendly support I had from the group's members, with especial mention to Simone Locci, for all his support in helping me to get introduced to Sentaurus TCAD simulator and to my room-mates: Omar Fakhr, Jamila Rezgani, Xueming Ju and Nico Kieβling for making work hours pleasant and motivating.

Thank you to my family and friends: for their time and support, for being always there.

I'm really happy of having had the opportunity on finishing my degree in the Physical Cryptography Institute here in München: great personal experience and totally enriching.

Table of contents

LIST OF FIGURES

Chapter 1

INTRODUCTION AND MOTIVATION

Nowadays many different techniques are continually being given related to cryptography and security. Nevertheless, standard approaches in the field are inherently based on unproven mathematical assumptions and on the supposed secrecy of binary keys which can be cloned or extracted and transferred from computer systems. These facts restrict both the security and applicability of existing concepts.

The main goals in this thesis are presenting Physical Cryptography, as another possible alternative. Before developing the concept, the lector should first be introduced to the topic this thesis is based on: Cryptography and Security.

1.1. Mathematical cryptography

This is an extended field where the techniques related to it, are mainly based on the idea of a mathematical one-way function, a function "easy to compute" but "difficult to invert". RSA could be a good example of mathematical cryptography : this is an algorithm for public-key cryptography. It is the first algorithm known to be suitable for signing as well as encryption, and one of the first great advances in public key cryptography. Every user in the communications protocol has a Public key (known by all the users) and a secret key (one for each user and only this one knows).

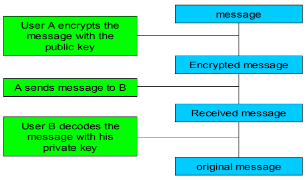

User A encrypts the message with the public key
message
Encrypted message
A sends message to B
Received message
User B decodes the message with his private key
original message

Figure 1: ***Communications protocol scheme for RSA***

Public and private keys are generated, for example, from using two large randomized prime numbers. Therefore security of this cryptosystem is basically based on the problem of factoring large numbers and the RSA problem. Full decryption of an RSA ciphertext is thought to be infeasible, but if the keys are not long enough (less than 2048 bits) then, these could become possibly breakable by using ultimate hardware or computers. Once the attacker managed to decipher the keys, then these can easily be transferred to malware. In other words, mathematical cryptoschemes are intrinsically based on the concept of secret binary keys, which can be copied, extracted from mobile systems through invasive or side channel attacks and subsequently be cloned or transferred from computer systems through malware.

1.2. Quantum cryptography

Quantum cryptography, or Quantum Key Distribution (QKD), uses quantum mechanics to guarantee secure communication. It enables two parties to produce a shared random bit string known only to them, which can be used as a key to encrypt and decrypt messages: they can detect the presence of any third party trying to gain knowledge of the key. Its security relies on the foundations of quantum mechanics, in contrast to traditional public key cryptography which relies on the computational difficulty of certain mathematical functions, and cannot provide any total guarantee of key security. However, quantum cryptography systems can only operate over small systems and are extremely sensitive to external perturbations and they require a special infrastructure of quantum communication channels.

1.3. Physical cryptography

Physical cryptography is a different and very recent approach to cryptography and security: it is based on the inherent complexity of nanoscale electronic and photonic systems. The main idea of this emerging field is to use physical nanostructures with specifically designed properties which can replace or complement standard cryptoschemes and standard binary keys, resulting in better security properties. This, leads to better security properties and enables fully new types of applications. Ravikanth Pappu defined and discussed in 2002 , the advantages of using Physical One Way Functions, based on generating keys using the properties from physical objects instead of using mathematical algorithms, which are generally based on unproven conjectures ore can present many vulnerabilities. As a result to this proposal, many research related to this topic has been made to the date and many Physical Uncloneable function (PUF) designs have been suggested.

PUFs, which extract secrets from physical characteristics of integrated circuits, so are especially interesting because they are easy to evaluate but the physical system is hard to characterize, model or reproduce, making it not easily vulnerable to the development of any new, efficient code-breaking algorithms or powerful computers. Security of traditional cryptography based on mathematical fundamentals, often relies on unproven mathematical assumptions, when physical cryptography is based on technological limits and formidable costs of characterizing and manufacturing certain objects with nanoscale features.

PUFs inherit their uncloneability property from the fact that each one has a unique and unpredictable way of mapping challenges to responses. Physical uncloneability requires an exact control over the manufacturing process, such that all parameters of the physical structure can be exactly defined, and this is a very hard task to manage. On the other hand, mathematical uncloneability means that an unknown response given the exact parameters or other Challenge Response Pair (CRPs) from the PUF should be very hard to compute. This is because a response is created as a very complex interaction of the challenge with the random components. Modelling this interaction, even if the random values are known, should take a lot of computational effort.

As a result to this previous research study, some techniques have been given to implement Physical Uncloneable Functions (PUFs) from which we will mention two methods (in the next point) proposed by MIT in 2005 on which SHIC idea will derive from.

1.3.1. Possible implementations for PUF: Arbiter and Ring Oscillator

Arbiter PUF

The circuit has a multiple-bit input X and computes a 1-bit output Y based on the relative delay difference between two paths with the same layout length. The input bits determine the delay paths by controlling the MUXes. Here, a pair of multiplexers, MUXes controlled by the same input bit $X[i]$ work as a switching box. The MUXes pass through the two delay signals from the left side if the input control bit $X[i]$ is zero. Otherwise, the top and bottom signals are switched. In this way, the circuit can create a pair of delay paths for each input X . To evaluate the output for a particular input, a rising signal is given to both paths at the same time, the signals race through the two delay paths, and the arbiter (latch) at the end decides which signal is faster. The output is one if the signal to the latch data input D is faster, and zero otherwise. Because the PUF circuit is rather simple to obtain, attackers can try to construct a precise timing model and learn the parameters from many input-output pairs. To prevent these model-building attacks, the PUF circuit output can be obfuscated by XOR'ing multiple outputs or a PUF output can be used as one of the MUX control signals.

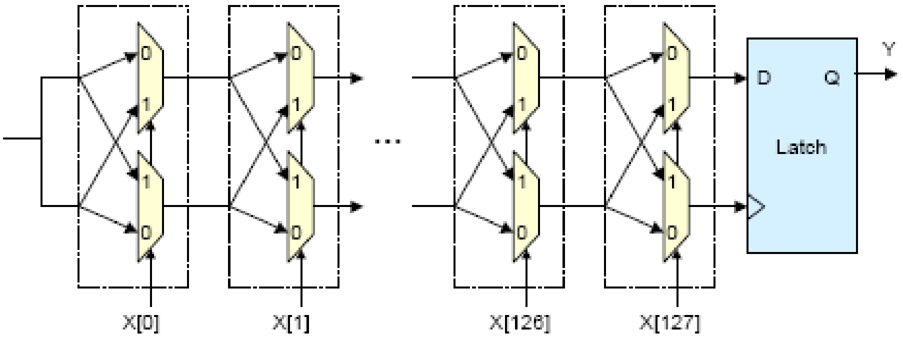


Figure 2: ***Arbiter PUF delay circuit***

There are two ways to construct a k-bit response from the 1-bit output of this PUF delay circuit. First, one circuit can be used k times with different inputs. A challenge is used as a seed for a pseudo-random number generator (such as a linear feedback shift register). Then, the PUF delay circuit is evaluated k times; using k different bit vectors from the pseudo-random number generator serving as the input X to configure the delay paths. It is also possible to duplicate the single-output PUF circuit itself multiple times to obtain k bits with a single evaluation.

Ring Oscillator PUF

This PUF design is based on delay loops (ring oscillators) and counters. Compared to the arbiter PUF previously described, the RO PUF allows an easier implementation, an easier evaluation of the entropy and higher reliability. On the other hand, the RO PUF is slower, larger and consumes more power to generate bits than the arbiter PUF. Therefore, the two designs are complementary; the arbiter PUF is appropriate for resource constrained platforms and the RO PUF is better for being used in secure processor designs.

Each ring oscillator that constitutes the structure is a simple circuit that oscillates with a particular frequency. Due to manufacturing variation, each ring oscillator oscillates with a slightly different frequency. In order to generate a fixed number of bits, a fixed sequence of oscillator pairs is selected, and their frequencies are compared to generate an output bit. The output bits from the same sequence of oscillator pair comparisons will vary from chip to chip. Given that oscillators are identically laid out, the frequency differences are determined by manufacturing variation and an output bit is equally likely to be one or zero if random variations dominate.

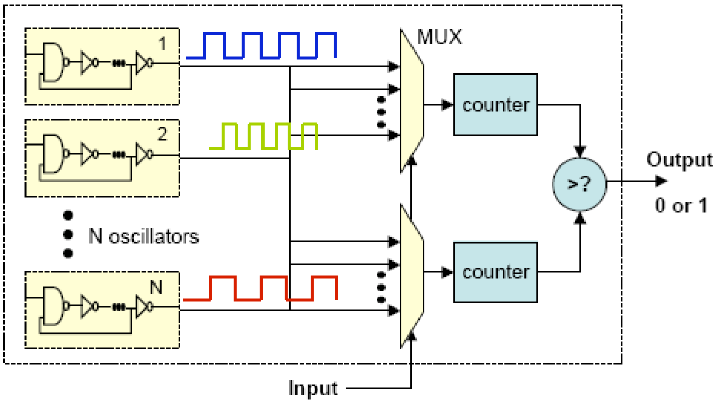


Figure 3: ***Ring Oscillator based PUF circuit***

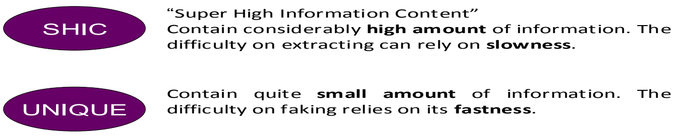
Unlike the arbiter PUF, there is no need for careful layout and routing: the paths from oscillator outputs to counters do not need to be symmetric. By counting many oscillator cycles, the difference in oscillator frequencies can be amplified and will dominate any skews in routing. Each comparison of a pair of oscillators generates a bit. There are $N(N-1)/2$ distinct pairs given N ring oscillators. However, the entropy of this circuit, which corresponds to the number of independent bits that can be generated from the circuit, is clearly less than $N(N-1)/2$ because the bits obtained from pair-wise comparisons are correlated. For example, if oscillator A is faster than oscillator B, the comparison will yield a 1. If B is in turn faster than C, the comparison will yield a 1. It is clear that when A is compared with C that the comparison will yield a 1: these bits are correlated. Fortunately, it is possible to derive the maximum entropy of this circuit assuming pair wise comparisons.

Detailed study referred to Arbiter PUFs and RO PUFs were made in 2007 by MIT and can be consulted in

Chapter 2

INTRODUCTION TO SHIC SYSTEMS AND UNIQUE OBJECTS

The main goals of this thesis are to present two systems inside the group of Physical Cryptography: SHIC "Super High Information Content" and UNIQUE. We want to present the usability of each system and propose a possible implementation to each. To do so, the next chapters will be dedicated to develop both SHIC and UNIQUE and proving their feasibility with analog circuit simulations on SPICE and device simulations on Sentaurus TCAD.



"Super High Information Content"
Contain considerably **high amount** of information. The difficulty on extracting can rely on **slowness**.

Contain quite **small amount** of information. The difficulty on faking relies on its **fastness**.

2.1. SHIC "Super High Information Content" systems

As a continuation to the work made by MIT in 2005 , Ulrich Rührmair introduced the idea to SHIC systems in . These respond to "Super High Information Content" systems and stand to be complex disordered physical systems which must contain an extraordinarily high amount of structural information. The system should satisfy the next conditions:

- The information content of the system can be extracted reliably and repeatedly through measurements with different parameters p_i and obtaining the resulting answer $M(p_i)$ of the system.
- The number of possible measurement parameters p_i is so large that the values $M(p_i)$ cannot be determined for all possible measurement parameters p_i within limited time.
- Due to the high information content in the system it must also be impossible to model, computationally learn, simulate or otherwise numerically predict the results of unknown measurement results $M(p_k)$ from known results $M(p_i)$.
- It must be prohibitively difficult to physically reproduce or clone a SHIC system.

These exposed properties promise that SHIC systems are not breakable, which makes themselves strongly suitable for cryptographic applications. More descriptions can be obtained in full detail in .

2.1.1. Applications of SHIC systems

Suppose we have a communication channel and we initially have two participants, Bob and $Alice$. If Bob is in possession of the circuit, then any participant in the protocol can verify that he is communicating with the possessor of the circuit. If first, $Alice$ measures some randomly selected bit values on the memory and then sends the memory to Bob, when this last one claims to have the memory, $Alice$ challenges Bob with the previously measured bit values and understands that Bob possesses the memory if his answers have a reasonable high success rate.

If an adversary, Eve, gains temporary access to the memory, then she may not have a suitable time to fully read it all out and successfully answer the challenges of $Alice$. The high information content and the physically limited read-out speed thwarts attempts to unrightfully claim ownership of the object.

Eve could use different strategies when wanting to fight the cryptosystem: she could try to re-fabricate the memory the same way as the original manufacturer did or can unlawfully obtain a second copy from the same memory design. If the memory content is generated by a truly random physical process beyond the manufacturers control then such effort is obviously fruitless. Duplicating the system would be possible for Eve but characterizing the circuit completely, would be physically impossible.

SHIC could also be used in the remote verification of the integrity of hardware systems. The protocol based on the fact that most SHIC systems are very sensitive to integrity violations, can be exploited in order to verify, remotely and without physical inspection, the integrity of a hardware system that has been encapsulated by a SHIC system. Such a protocol would be a key tool in the long predicted ubiquitous computing scenarios. In any case, with SHIC we could always assure a secure exchange of a secret cryptographic key between two participants in a communication protocol ($Alice$ and Bob).

2.2. UNIQUE objects

A physical object is called unique if it possesses properties with respect to a fixed measurement method that cannot be reproduced in another physical object. The inability for reproduction or cloning the object could also be impossible for the original manufacturer of the object. Furthermore, it should be possible to express the unique and non-reproducible properties of the object in a relatively short bit string, being possible to measure them in a short time interval on the order of seconds.

We are in front of physical systems whose measurable properties are extremely difficult to duplicate or reproduce.

A fruitful approach for realizing mass produced unique objects is to exploit random manufacturing variations, which are beyond the control of the manufacturers.

2.2.1. Applications of UNIQUE systems

We have as typical applications of unique objects, the labelling of products, identifying bank cards and passports; which can be implemented in combination with a central database of all valid objects. The additional use of special crypto techniques such as digital signatures even allows label verification in a highly practical, offline and decentral process, without central databases.

We could also use them for the generation of copy protected digital content. Here, a digital signature is used to inseparably link the content with the unique properties of the unique object. If the content is to be played, the playing device checks the presence of the unique object, and does not play the content otherwise.

Chapter 3

INTRODUCTION TO SPICE AND SENTAURUS SIMULATORS

As previously pointed out in the introduction, Physical Cryptography involves using optical and electrical nanostructures for cryptography and security. Therefore, if we want to investigate the feasibility of using electronic circuits for this purpose, simulation tools could be useful to understand and evaluate the behaviour of future circuit proposals.

In this chapter, we will briefly introduce SPICE and Sentaurus TCAD, as they will be the simulators used in the next lines when studying SHIC and UNIQUE systems.

3.1. Introduction to SPICE simulator

SPICE, Simulation Program with Integrated Circuit Emphasis, is a general-purpose analog electronic circuit simulator, especially used in Integrated Circuits to check the integrity of circuit designs and to predict circuit behaviour.

In the first part of our study we propose a circuit based on a memory array as a possible implementation to SHIC. In order to assure this structure will fit into our necessities, SPICE can be an interesting tool to use: we can obtain voltage or current measurements in any chosen node in the circuit and also transient analysis, which will be very useful when studying crossbars' feasibility.

3.1.1. Possible analysis with SPICE

Any simulation with SPICE requires establishing which kind of analysis we want to make. We now mention the most important for our study.

DC analysis: this is automatically performed prior to a transient analysis to determine the transient initial conditions. With this we can generate dc transfer curves: a specified independent voltage or current source is stepped over a user-specified range and the dc output variables such as current or voltage can be observed.

Transient analysis: This method is used for time-dependent analysis of the circuit. At least one voltage or current time dependent source is required; the rest can be set to a dc value. The results are tabulated as output voltage or current versus time.

3.1.2. File types for SPICE

To model a circuit with Spice we can use a text file type .CIR. or draw the circuit manually with Schematics Editor which will create the *.CIR file, along with several auxiliary file types: the *.SCH (the schematic data, itself) and *.NET (network connection files).

The output file generated by Spice is a text file that has the file type .OUT. When running a DC analysis, results will be saved in that file type. When running a transient analysis, there will be too much data for the *.OUT file, so the numerical data will be saved in a *.DAT file. Other files used with Spice are *.LIB files where the details of complex parts are saved.

3.1.3. Subcircuits creation with Spice

Subcircuits allow the designer to create a model or circuit for a device that may be used many times in a schematic (and saved for later use in other circuits). Each instance of the device can use the same parameters or this can be changed on an instance-by-instance basis. The advantage of a subcircuit is that one symbol on a schematic can represent another circuit. Whenever that symbol is placed on the schematic, the entire subcircuit is placed. This allows for ease of design with more clarity on the schematic. And as it happened for a circuit design, we can use text editor or schematics for its definition.

These will be very useful for future designs in the next chapter. Annex I

3.1.4. Example application: Diode simulation

As a first rapprochement to using SPICE simulator we will design a simple Schottky diode which we will be an important component inside our proposal to SHIC. As previously commented, there are two possible ways on simulation an analog circuit in SPICE: we can design the structure with a text editor *.cir or we could do it with schematics *.asc. Annex II

Device definition with text editor

First, a definition of the parameters that compose the structure are needed: the structure is defined as a subcircuit, where behaviour of the diode is specified, this obeys the Shockley equation.
After the design's definition, what analysis should be made must be indicated and which values is the user interested on showing in the graphs.

```
*DiodeON
# Definition of the different parameters that conform the subcircuit
.param Is  1.0E-17
.param Rs  1.66E+6
.param Rp  3.0E+14
.param n   1.8

# Definition of the subcircuit "DiodeON": diodes definition and
# resistor connection to the nodes.
.subckt DiodeON in out
*Bdieq in dout I = Is*exp( (V(in)-V(dout)) / 0.025 -1)
Bdieq in dout I = Is * (exp( (V(in)-V(dout))/(n*0.025) ) -1)
Rpar  in dout {Rp}
Rserial dout out {Rs}
.ends

Xdi in 0 DiodeON
vprobe in 0 1.0

.options list node gminsteps=0

# We can observe results from a minimum to maximum value of the voltage source
.dc Vprobe -2.6 2.6 0.01
# We can indicate which values we want to visualize in the graphs
.print dc ix(di)
.print dc log(ix(di))
.end
```

Device definition with schematics

Now drawing the structure is needed. The user just has to choose the components we need to define the structure make the wirings and add the legend that indicates which kind operation we want to make: the simplest, a continuous voltage DC sweep through the voltage source.

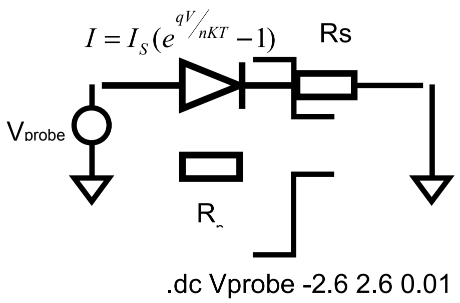


Figure 4: ***Equivalent circuit for a Schottky diode in Schematics***

Results view

Once we have the files prepared we can run the simulation and will be able to visualize results: when studying a diode, the most important graph to show could be the I-V curve, with it we can observe for which biasing it starts to work.

Depending on how voltage is applied to the junction, two different bias states can be given:

- **Forward biased (positive voltage applied to the Pcontact):** Electrons and holes are pushed towards the junction. This also reduces the height of the energy barrier and reduces the width of the depletion zone. These effects make it easier for free electrons and holes with modest amounts of thermal energy to cross the junction. As a result, we get a sizeable current through the diode when we apply a forward bias voltage, a current which varies exponentially with the applied voltage (Current obeys the Shockley equation for a diode)

- **Reversed biased (negative voltage applied to the Pcontact):** Free electrons and holes are pulled apart, and the height of the energy barrier between the two sides of the diode increases. As a result it is almost impossible for any electrons or holes to cross the depletion zone and the diode current produced is virtually zero (parasitical current). A few lucky electrons and holes may happen to pick up a lot of thermal energy. This gives them enough strength to cross the barrier; hence the reversed biased current is not zero, just very, very small: we say certain parasitical current is dissipated.

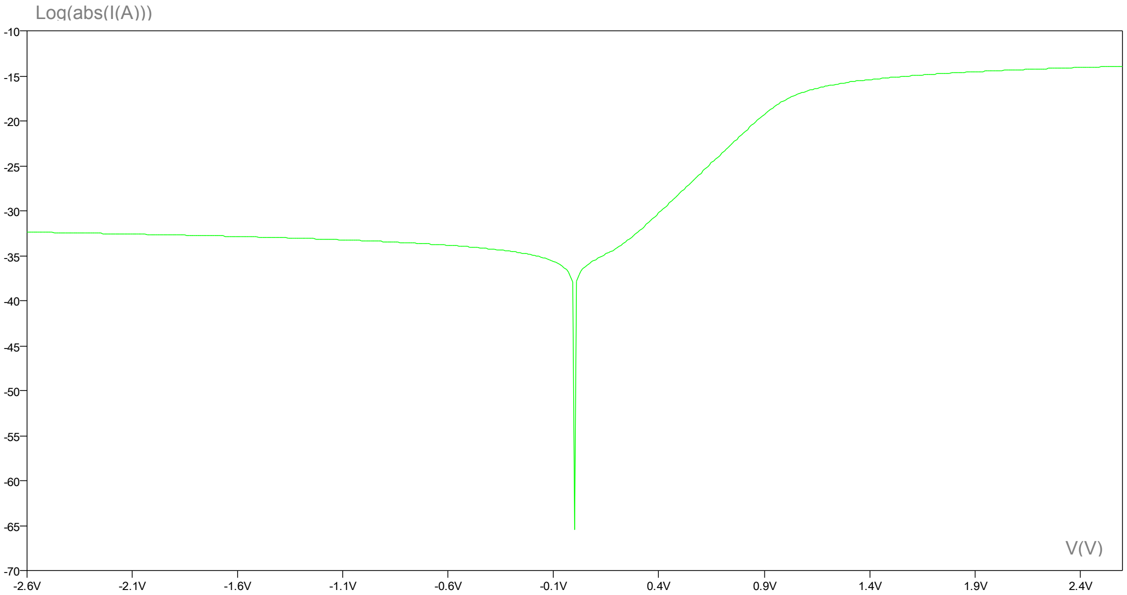
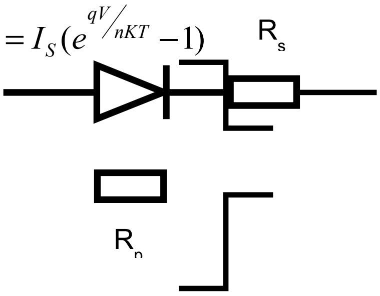


Figure 5: ***Current-Voltage characteristics of a diode backed-junction***

As it is observed the current grows exponentially with the voltage, obeying the Shockley diode equation:

$$I = I_S(e^{qV/nKT} - 1)$$



I = diode current

I_S = reverse bias saturation current

V = voltage across the diode

n = emission coefficient, also known as the ideality factor.

3.2. Introduction to Sentaurus TCAD simulator

Sentaurus Device is an advanced 1D, 2D and 3D device-level simulator capable of simulating the electrical, thermal, and optical characteristics of silicon and compound semiconductor devices and presents several benefits which include:

- Exploration of new device concepts for which fabrication processes are not yet defined.
- Characterization of electrical, thermal, and optical behaviour of semiconductor devices for fast prototyping, development, and optimization of their performance.

- Shorten development time by supplementing experimental data with deep physical insight from simulation.
- Study of the sensitivity of device characteristics to process variations for optimizing parametric yields.
- Generation of electrical data for SPICE modelling and early-silicon circuit evaluation.

In the second part of our study we propose photodiodes as a possible implementation to UNIQUE. In order to assure this device will fit into our necessities, using Sentaurus can be interesting: we can obtain voltage or current measurements for optical inputs and make internal study of the device (electron-hole behaviour analysis), which can be useful when studying a photodiode's feasibility.

3.2.1. Stages for a device creation

Device design: Mdraw and Language editor

As a first step, design of the device must be done: The user may use Mdraw (equivalent to schematics) or a language editor. Here, materials, contacts, doping and refinement regions are defined and the meshing can also be done.

The structure is basically described by two files:

- **The data** (or doping), which contains the doping profiles.
- **The grid** (or geometry), which contains a description of the regions, boundaries, material types, location of the electrical contacts.

Device simulation: Dessis

Once we have these files prepared (*.bnd and *.cmd), the grid *_msh.grd and the doping *_msh.dat files will be generated with the command `>mesh diode` necessary to running the simulation stage with Dessis. This incorporates advanced physical models and robust numeric methods for the simulation of most types of semiconductor.

***_des.cmd file description: sections and keywords**

The *_des.cmd file is particularly outstanding as it contains all the necessary information to obtain the results referred to the device on study. We can obtain internal study of the device such as electron-holes behaviour (current density, mobility, velocity, etc.) We now briefly describe each section of it:

File section

This specifies the input and output files necessary to perform the simulation.

- Input files: *_msh.grd, is the essential input file as it defines the mesh and various regions of the device structure, including contacts. *_msh.dat contains the doping profiles data for the device structure.

- Output files: The file *_des.dat is the file name for the final spatial solution variables on the structure mesh and *_des.plt, the name for electrical output data (such as currents, voltages, charges at electrodes). *_des.log is an alternate file name for the output log or protocol file that is automatically created whenever Dessis is run.

Electrode section

Defines all the electrodes to be used in the Dessis simulation, with their respective boundary conditions and initial biases:

- `Name="string"` each electrode is specified by a case-sensitive name that must match exactly an existing contact name in the structure grid file.

- `Voltage="float"` defines a voltage boundary condition with an initial value.

Physics section

It allows a selection of the physical models to be applied in the device simulation.

Mobility models including doping dependence, high field saturation (velocity saturation), and transverse field dependence are specified for this simulation.

Plot section

This Specifies all the solution variables which are saved in the output plot files (.dat).

Math section

Dessis solves the device equations (which are essentially a set of partial differential equations) self consistently, on the discrete mesh, in an iterative fashion. For each iteration, an error is calculated and Dessis attempts to converge on a solution that has an acceptably small error.

Solve section

The Solve section defines a sequence of solutions to be obtained by the solver.

As the simulation proceeds, output data for each of the electrodes (currents, voltages, and charges) is saved to the current file *_des.plt after each step and, therefore, the electrical characteristic is obtained. The final solution is saved in the plot file *_des.dat.

Plotting out of simulation results: Tecplot

Finally we could plot the results from Dessis in graphs, so we have the possibility of using two tools depending on what we want to plot. For device simulation using TCAD tools, TecPlot is best for viewing 3D images of physical quantities throughout a device and for making cross sections through a device and viewing quantities across them. TecPlot takes: *_msh.grd and *_des.dat files as inputs and generates plots that can be exported into any of the most common graphic formats.

3.2.2. Example application: simple PN junction simulation

As a first approximation to Sentaurus and to get familiarized with this tool, we will start with the design of a simple pn-junction, concretely a 10µm x 4µm silicon made device, which will be useful when we study UNIQUE objects.

As a first step, we could make a current-voltage characteristics study, to prove this has a diode's behaviour: biased/unbiased ranges, current capacitance and functional properties of the device are reflected in this curve. Once we introduce photodetectors, optical generation command will be required (To be explained).

First, we must define the device, to do so; we will use the language editor, as if we want to make any future changes of the parameters we could do it easily.

Device definition: material & contacts "pn2D.bnd"

```
Silicon "Region 0" { rectangle [ (-5,-2) (5,2) ] }
Contact "Pcontact" { line [ (-5,-2) (-5,2) ] }
Contact "Ncontact" { line [ (5,-2) (5,2) ] }
```

Definition of the regions, boundaries, material types "pn2D.cmd"

We will try first by using the same mesh for the entire device.

```
Definitions {
    # Refinement regions
    Refinement "Default Region"{
    }
    Refinement "Silicon"{     # Max and min mesh sizes. Here, one mesh for the enire device
            MaxElementSize = (0.1 0.2)        # Max(Width, Height)
            MinElementSize = (0.1 0.2)        # Min(Width, Height)
    }
    # Profiles: doping species and concentration of each doping
    Constant "P-type"{
            Species = "BoronActiveConcentration"
            Value = 1e+18
    }
    Constant "N-type"{
            Species = "PhosphorusActiveConcentration"
            Value = 1e+16
```

```
     }
}
Placements {             # Definition of the areas of the different regions
     # Refinement regions
     Refinement "Default Region"{
               Reference = "Default Region"
               #Default region
     }
     Refinement "Silicon"{         # Corresponds to the whole structure
               Reference = "Silicon"
               RefineWindow = rectangle [( -5 -2 ) , ( 5 2 )]
     }
     #Profiles
     Constant "P-type"{
               Reference = "P-type"
               EvaluateWindow{
                         Element = rectangle [( -5 -2 ) , ( 0 2 )]
                         DecayLength = 0
               }
     }
     Constant "N-type"{
               Reference = "N-type"
               EvaluateWindow{
                         Element = rectangle [( 0 -2 ) , ( 5 2 )]
                         DecayLength = 0
               }
     }
}
```

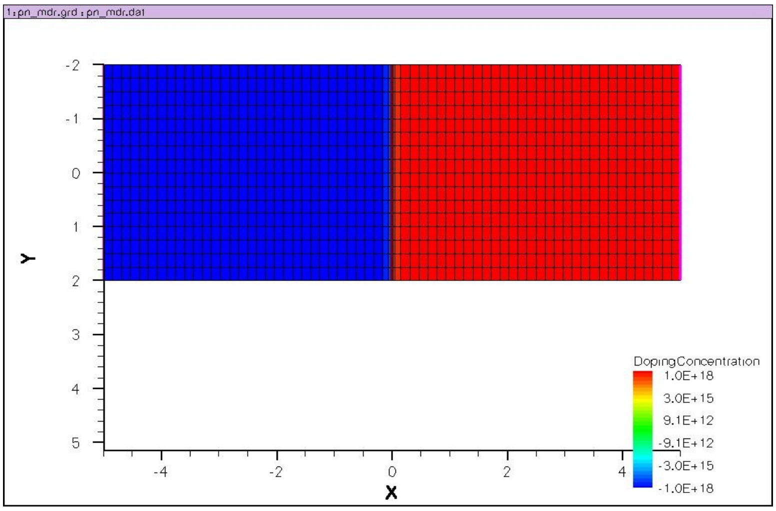


Figure 6: ***2D pn-junction structure on Sentaurus***

Once we have .bnd and .cmd files created the resultant structure in Tecplot can be observed: we can analyse its doping distribution, differentiate p-side from n-side and see the meshing through the structure.

Static solution for 2D

First of all we will make a static study of the device's behaviour and prove that we obtain the expected characteristics for a diode. Let's describe the instructions we used in the Solve section in order to obtain a static study of our device:

- `Poisson` This specifies that the initial solution is of the nonlinear Poisson equation only. Electrodes have initial electrical bias conditions as defined in the Electrode section. In this particular case, a -2V bias is applied to the P-type.
- `Coupled {Poisson Electron Hole}` This introduces the continuity equation for electrons and holes, with the initial bias conditions applied. In this case, the electron current continuity equation is solved fully coupled to the Poisson equation, taking the solution from the previous step as the initial guess.
- `Quasistationary` specifies that quasi-static or steady state 'equilibrium' solutions are to be obtained. A set of Goals for one or more electrodes is defined in parentheses. In this case, a sequence of solutions is obtained for increasing gate bias up to and including the goal of 1.5V. A fully coupled (Newton) method for the self consistent solution of the Poisson and electron continuity equations is specified in braces. Each bias step is solved by taking the solution from the previous step as its initial guess.

As for the math section we will basically use three keywords:

- `Extrapolate` In quasistationary bias ramps, the initial guess for a given step is obtained by extrapolation from the solutions of the previous two steps (if they exist).
- `NewDiscretization` Switches back to the obsolete discretization scheme for continuity equations, lattice temperature equations, and carrier temperature equations.
- `Derivatives` Switches off the analytic derivatives of the mobility and avalanche terms (switched on by default).

Command file for Dessis: "pn2D_des.cmd" static

```
File {
    # input files
    Grid= "pn2D_msh.grd"
    Doping= "pn2D_msh.dat"
    # output files
    Plot= "pn2D_des.dat"
    Current= "pn2D_des.plt"
    Output= "pn2D_des.log"
}
Electrode {
    { Name="Pcontact"    Voltage= 0.5 }
    { Name="Ncontact"    Voltage= 0.0 }
}
Physics {
    Mobility( DopingDep HighFieldsat Enormal )
    EffectiveIntrinsicDensity( OldSlotboom )  # This is the silicon band-gap narrowing model
                                              # that determines the intrinsic carrier
                                              # concentration.
}
Plot {
    eDensity  hDensity  eCurrent  hCurrent
```

```
        Potential SpaceCharge  ElectricField
        eMobility  hMobility eVelocity  hVelocity
        Doping  DonorConcentration   AcceptorConcentration
    }
    Math {
        Extrapolate
        Derivatives
        NewDiscretization
    }
    Solve {
        Poisson
        Coupled { Poisson Electron Hole }
        Quasistationary( MaxStep=0.1 Goal { Name="Pcontact" Voltage=1.5 } )
                { Coupled { Poisson Electron Hole} }
    }
```

Now we could try to evaluate the results and will mainly focus in the I-V curve: so as expected we can see a curve which logically corresponds to our expectations as it obeys the Shockley equation.

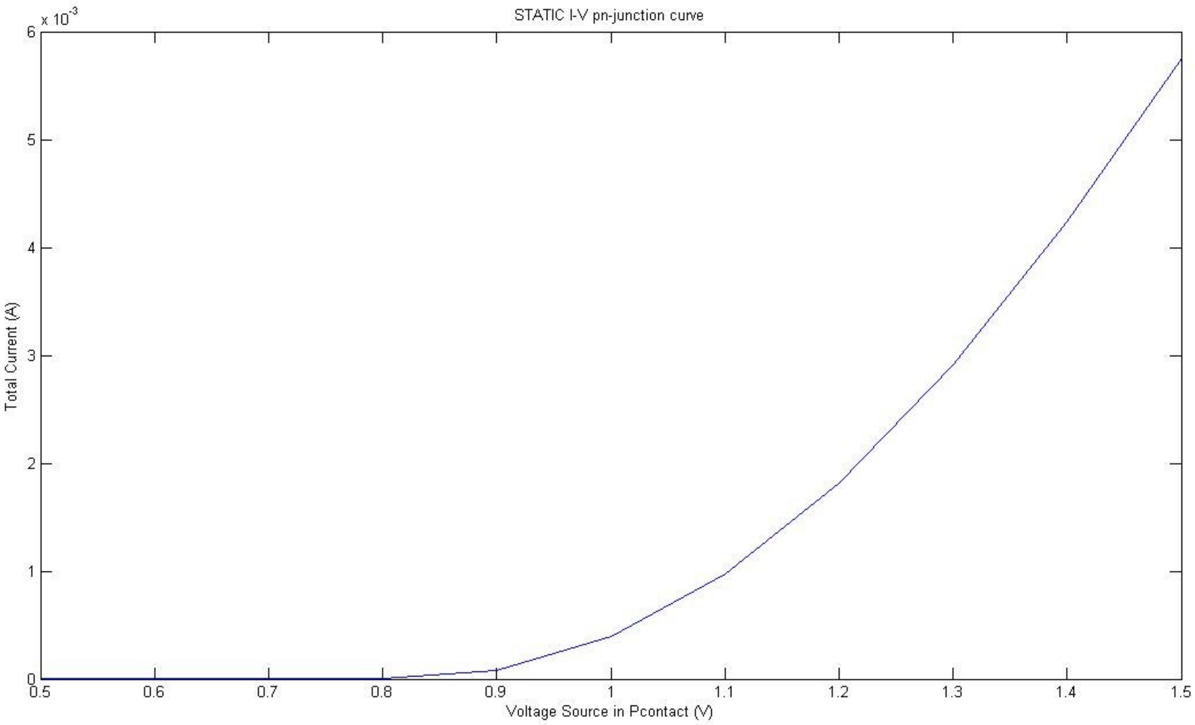


Figure 7: ***I-V curve for a pn junction: static study for $V_{Pcontact}$=(0.5,1.5)V***

Transient solution for 2D

Now let’s see if we obtain similar results when we apply a pulse in time. We should prove that for a determinate source value, the resultant current finally converges to the static value which corresponds to that same voltage, having as a result, a similar curve lines to the one we previously obtained in the static solution.

To make analysis in time, we need to specify it in the Solve section with the command `transient(…)`, which allows the user to overwrite time-step control parameters: Initial time and final time of the simulation can be specified, but the most important is to choose adequate values for `IntialStep`, `MinStep` and `MaxStep` as these determine the simulation time, the accuracy of the results and its ease to convergence: Actual step sizes are determined internally, based on the rate of convergence of the solution at the previous step. The minimum step and maximum step would be the thresholds of the size step variation.

Command file for Dessis: “pn2D_des.cmd” transient

```
File {
    Grid= "pn2D_msh.grd"
    Doping= "pn2D_msh.dat"
    Plot= "pn2D_des.dat"
    Current= "pn2D_des.plt"
    Output= "pn2D_des.log"
}
Electrode {
    { Name="Pcontact"    Voltage=0.0
            Voltage = (      0 at 0,
                             0 at 1e-11,
                             1 at 1.00001e-11,
                             1 at 10e-11          )
    }
    { Name="Ncontact"    Voltage=0.0 }
}

Physics {
    Mobility( DopingDep HighFieldsat Enormal )
    EffectiveIntrinsicDensity( OldSlotboom )
}

Plot {
    eDensity  hDensity  eCurrent  hCurrent
    Potential SpaceCharge  ElectricField
    eMobility  hMobility eVelocity  hVelocity
    Doping  DonorConcentration   AcceptorConcentration
}
Math {
    Extrapolate
    RelErrControl
    Iterations = 15
    NewDiscretization
}
Solve {
    Poisson
    Coupled { Poisson Electron Hole }
    Transient (
    InitialTime=0.0  FinalTime=10.0e-10  InitialStep=0.5e-12  MaxStep=1.0e-12  Minstep=1.0e-13
    Increment=2
    )
    { Coupled {
            Poisson Electron Hole }
    }
}
```

As a result we can have an approximate idea of the I-V curve in the transient domain, but not the precise and exact one as we had for the static case: Now we have to restrict our measurement time, and would need large time periods if we wanted to determine the real current the diode converges to for the voltage applied, aspect which requires a considerable time of simulation. In this stage we only want to prove that Dessis would be suitable as a simulation tool for much more complex future designs. So it would be enough by verifying we obtain a similar graph to an exponential (diode behaviour) and inside the current magnitudes we had for the static domain.

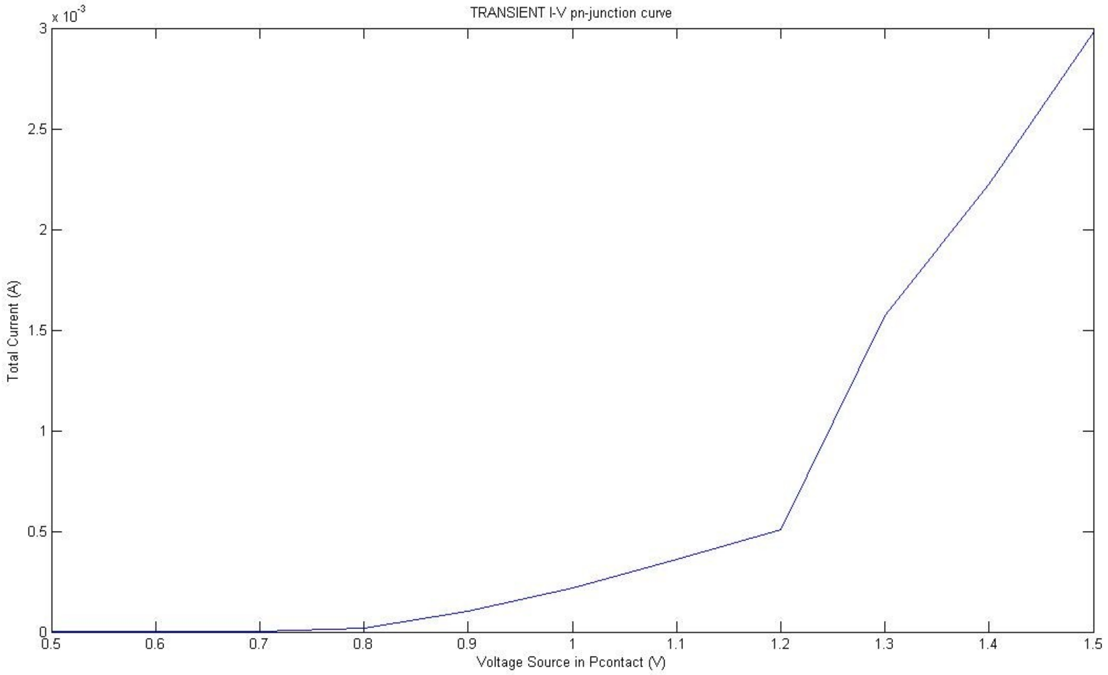


Figure 8: ***I-V curve for a pn junction: transient study for $V_{Pcontact}$=(0.5,1.5)V***

Static vs. transient solution for 2D

We can show now, both graphs and contrast behaviours and final values. With these optimal results, we can consider Sentaurus TCAD as a possible simulator tool for future use.

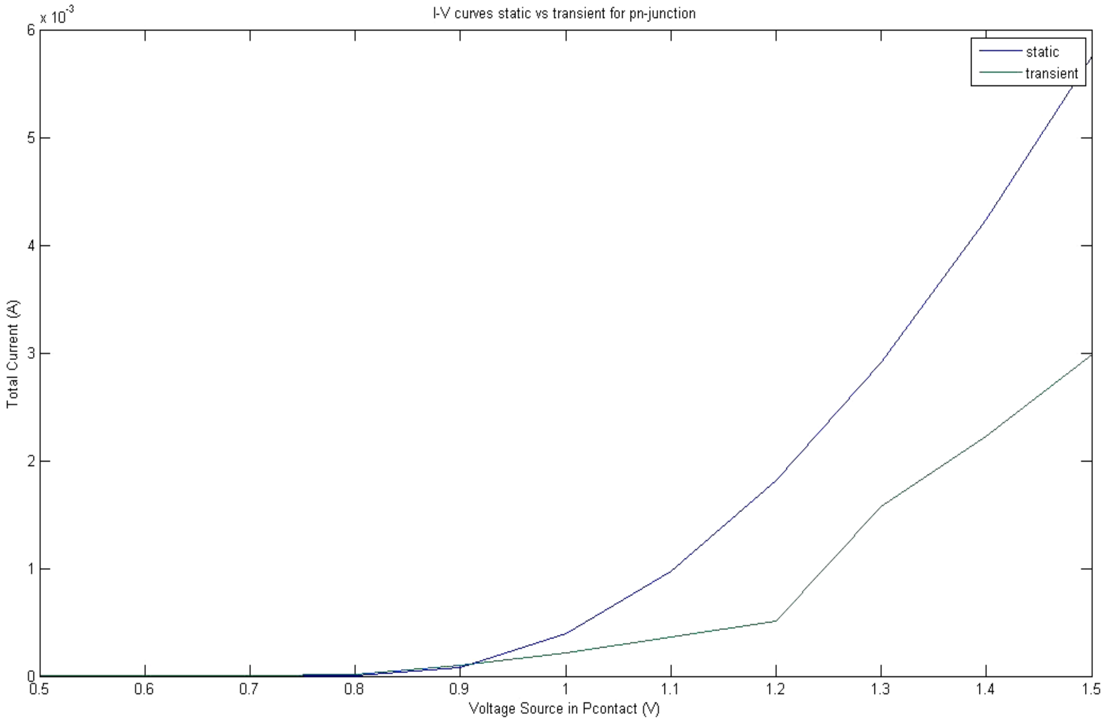


Figure 9: ***Static vs. Transient IV-curves in linear scale***

Chapter 4

CROSSBAR MEMORIES: APPROACH TO SHIC SYSTEMS

In this chapter we present a possible implementation of SHIC systems: A high capacity and hard wired memory, where high amount of data can only be accessed very slowly. The access time cannot be sped up without destroying the integrity of the memory and the information contained therein.

Our aim is to conclude that passive crossbar memories represent the highest security with the lowest cost for a two-dimensional integrated circuit technology. To do so, we will introduce the main idea of the proposal and will complete the study with analog-circuit simulations on SPICE, in order to conclude we reached our aims.

4.1. Crossbar memories: introduction to the idea

Cross-point architectures are the simplest functional nanodevices with a very regular geometry. Therefore, they hold a great promise in nanoelectronics, where fabrication challenges prohibit making a more complex, arbitrarily interconnected circuit.

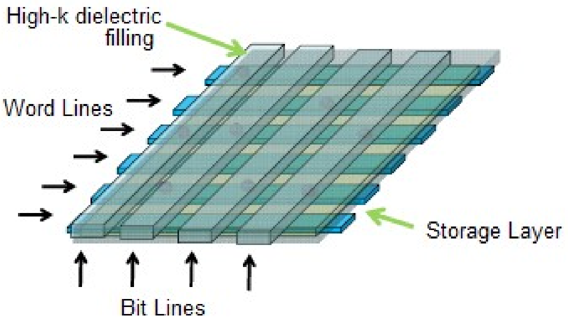


Figure 10: ***Schematic illustration of a Crossbar Memory***

Here we assume that only the storage array is implemented by crossbar technology: The user has a matrix $N \times N$ which can store N^2 bits (1bit/cell) and is distributed in N word lines (corresponding to files) and N bit lines (corresponding to the columns).
A particular bit at the intersection of the horizontal and vertical lines is addressed by activating the corresponding bit and word lines and measuring the current flowing through the crossing. Usually each junction is a multilayered structure showing nonlinear characteristics.

4.1.1. Specifications for the memory

How far can attacker Eve get with invasive attacks depends on the circuit architecture of the memory. As a result, a fixed-content memory circuit array, containing N^2 bits of information should satisfy the following requirements:

- The read-out speed is limited by the design of the circuit to k bits/second, for a small value of k, so it basically depends on the construction of the memory cells, bit (vertical) and word (horizontal) lines, and cannot be sped up by an invasive attack which uses different circuitry to access the memory.
- The time T_{full} required to manage a complete circuit characterization exceeds the application lifetime of the circuits.
- The N^2 bits content of the memory is physically random, so even the manufacturer has no control over it.

4.1.2. Particular aspects of the design

The proposed crossbar presents some specific aspects closely related to security, which are important to take into account:

- This crossbar is not writable: it carries hard wired information content, defined by the storage layer and which is unique, random for each instance of the fabricated memory.
- The space between the bit and word lines is filled with a high dielectric constant material, which creates large interwire and junction capacitances.

The entire memory array can be built as one monolithic block, which prevents the attacker from cutting access time by reading multiple memory banks in parallel.

4.1.3. Security assessment

The cryptosystem is broken if the attacker manages to read-out all the contents of the memory within a short time frame. This can be done by:

- Tampering with a silicon-based read-out circuitry, and use a lower input resistance, R_{in}
- Reading out many bits in parallel
- Cutting the circuit into pieces and reading them separately and parallel in time.

The crossbar memory provides protection against these scenarios in the following way:

- The value of the input resistance R_{in} cannot be decreased, because it would result in high currents, thus destroying the corresponding bit/word line. The time constant τ is fixed by the circuit construction.
- Parallel read-out would result in high currents, and therefore, as explained in the last point, is not possible.
- The circuit is fabricated at state of the art lithographic technology, contacting internal nodes or partitioning the circuit without seriously damaging it, is formidably difficult.

4.2. Resistive crossbars Memories

Previous work on Crossbar Memories has already been made as seen in the previous work made in 2008 and that are explained in .

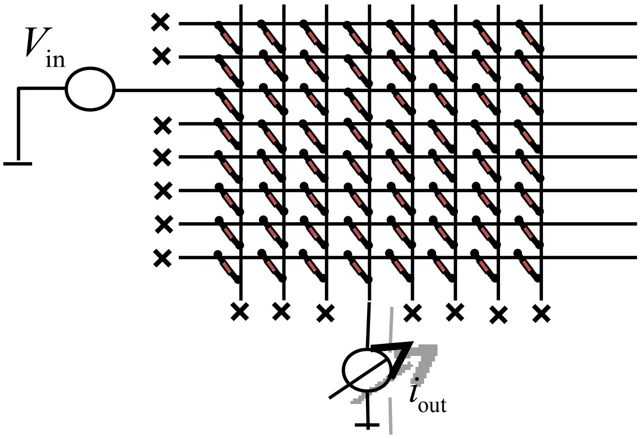


Figure 11: *Scheme of a resistive crossbar*

In the simplest crossbars the junction is only a resistive element with two possible values $R = R_{on}$ or $R = R_{off}$. In this way, multi-bit storage can be achieved by the continuous distribution of these components.

If we assume a maximum-information content (maximum entropy memory), then half of the junctions are in the $R = R_{on}$ and the other half are in the $R = R_{off}$ state.
To read the crossbar the V+ and V- voltages are set to zero, so the parasitic current paths end up in a shortcut to ground. By performing a current measurement on R_{sens} (selected Bit Line) it is possible to find out the state of the accessed junction. However, the vast majority of currents $(\approx nI_{sens})$ flow through the parasitic current paths and greatly overload the accessed word line. As a result, resistive crossbars are not scalable to the great sizes (for bigger N, the system doesn't work), so they cannot be considered as a suitable model to develop for implementing this type of memories.

4.3. Diode Crossbar memory

As a possible improvement to the previous model, placing diodes in series with the resistive element at each junction is proposed: A $N \times N$ matrix of simple multilayered structures with nonlinear characteristics (backed diodes) which are easy to implement as well as difficult to get accessed to. These cross-point architectures are the simplest functional nanodevices we can use as they present a very regular geometry characterized only with two terminal, passive devices.

If the user applies a bias scheme that assures: parasitic junctions (the non-accessed bits) are all reverse-biased or zero-biased, then this would make it possible to have mostly all the current flowing through R_{sens} ; Reverse-biased and zero-biased state in a diode contributes to some parasitical current through the unaccessed lines, but its value wouldn't be as huge as when we had a Resistive crossbar memory: The diode-backed crossbar memory is scalable to very large N array sizes, without needing extreme requirements to the diode or the R_{on} / R_{off} ratio of the resistive layer. The Crossbar memory's concern is the destruction of the wires by current overloading rather than the overall power dissipation.

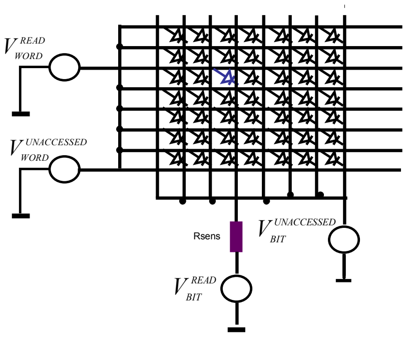


Figure 12: ***Biasing scheme of a diode crossbar memory***

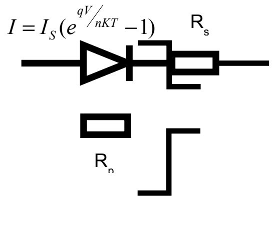
These two points: assuring parasitical currents don't overload the accessed bit line (selected bit can be correctly read) and making the structure robust enough to invasive attacks, will be the main subjects of our study in the next lines.

In the crossbar array, a particular bit at the intersection of the horizontal and vertical lines is addressed by activating the corresponding bit (column) and word (file) lines and measuring the current flowing through the crossing.

4.3.1. Structure of the junctions in the diode crossbar array

Depending on how voltage is applied to the junction, two different bias states can be given: We say forward-bias occurs when the P-type semiconductor material is connected to the positive terminal of a battery and the N-type to the negative. This enables the pn junction to its conduction as such bias causes a force on the electrons pushing them from the N side towards the P side (junction closed). On the other hand, reverse-biased supply means the voltage at the cathode is higher than the one at the anode. Therefore, no current will flow until the diode breaks down. The holes in the P-type material are pulled away from the junction, causing the width of the depletion zone to increase (junction opened). Similarly, because the N-type region is connected to the positive terminal, the electrons will also be pulled away from the junction. Therefore the depletion region widens, and does so increasingly with increasing reverse-bias voltage. This increases the voltage barrier causing a high resistance to the flow of charge carriers thus allowing minimal electric current to cross the pn junction.

When studying the I-V curve from a diode, this behaves according to the Shockley diode equation: Information, in bits, is carried by the serial resistor, $\boldsymbol{R_s}$ and as we can appreciate from two possible states for each junction are given: diode ON which means that the junction is closed, or diode OFF when this is open.



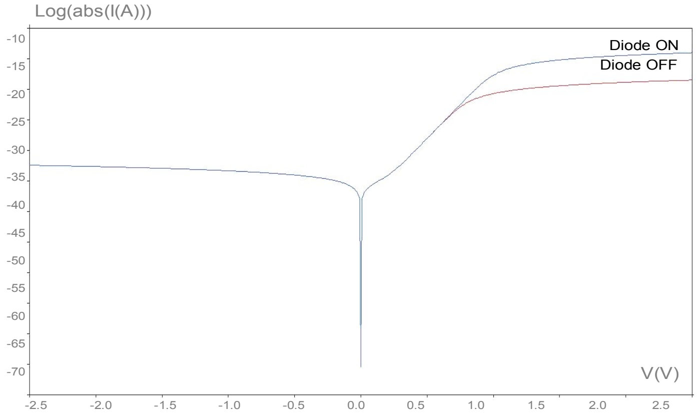


Figure 13: ***Current-Voltage characteristics of a junction in a diode crossbar array***

As we can observe, we can distinguish one state from another (ON/OFF) for $V_{dd} > 0.7V$ when having only a circuit with one diode. Now we want to see which would be the effect of having N^2 diodes and which would be the adequate V_{dd} so that both states could correctly be identified.

4.3.2. Equivalent circuit for a Diode Crossbar Memory

Before going into SPICE simulations for feasibility study we need to describe first the equivalent circuit to the $N \times N$ matrix that constitutes the described memory. When obtaining the equivalent, it is important to take into account the junction capacitances, unavoidable as well as unwanted, which are present because of the wiring and the proximity between the diodes. Serial internal resistance of the voltage generators is also a fact because of the wires and the neighbouring components of the circuit. The values we choose for these components are very important as they determine τ time constant which origins the time to read out a bit. Annex III

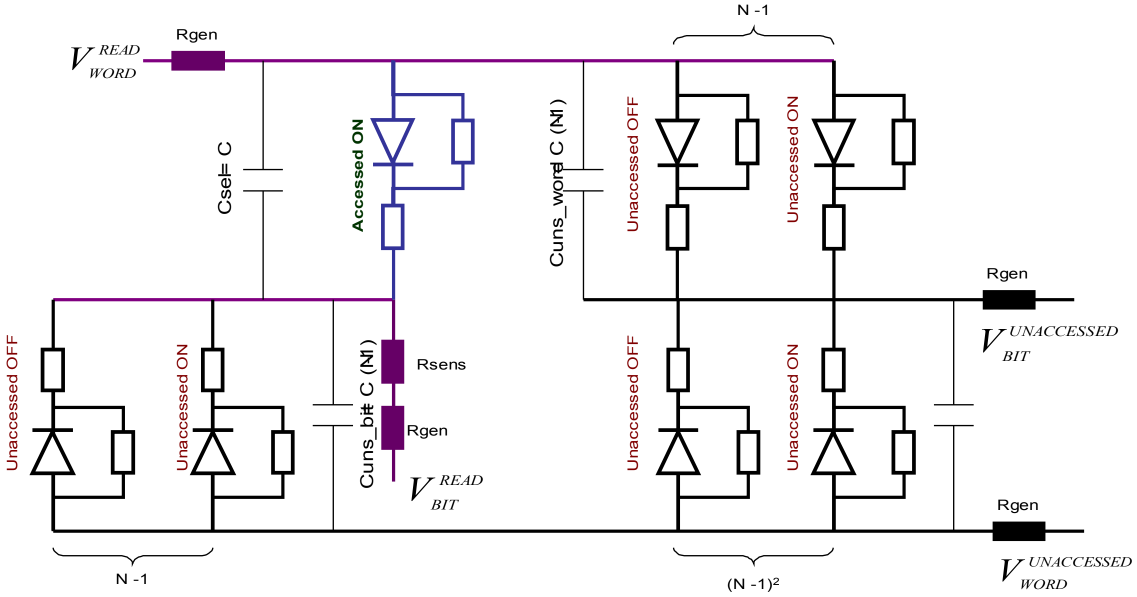


Figure 14: ***Equivalent circuit for a diode crossbar array***

If we turn back to Figure 12 we can observe: the selected junction (one) and $N-1$ unaccessed junctions are connected to the read word line. Other $N-1$ unselected junctions correspond to the read bit line and the rest of the junctions in the array, $(N-1)^2$ to the unselected bit lines.

As observed in Figure 14, each junction presents an OFF/ON state: $N-1$ junctions with the equivalent to $N-1$ capacitors in parallel, $C'=C\times(N-1)$ are connected to the read word line, $N-1$ and also $C'=C\times(N-1)$ connected to the read bit line and the rest, $(N-1)^2$ junctions and $C''=C\times(N-1)^2$ connected to the unaccessed lines. The internal resistance associated to the voltage source connected to the read word line and $C'=C\times(N-1)$ corresponding to the $N-1$ unselected junctions connected to the read word line, will determine the time constant of the circuit $\tau=R_{gen}\times C'$. This parameter we are especially interested on, as security in a crossbar relies on read out speed: we are looking for a structure that gives us a significantly slow time to read out the whole information, making it physically impossible to the attacker to fully characterize the circuit. It is important to choose R_{gen} in such a way that the driven word line is not overloaded by the capacitive peak at charge-up. $C_{junction} = 10^{-13} F$ and $R_{gen} = 100k\Omega$ are the values we used in our designs.

4.3.3. One bit read-out from the crossbar

We want to assure correct bit to bit read-out without loosing information: ideally all the current should flow through R_{sens} (the accessed bit line) but as we are using diodes, certain current leakage is present even if the diode is OFF, aspect that can contribute to a parasitical current flowing through the unaccessed wires. The first thing we want to assure is that any bit information kept in the crossbar may be read correctly whenever we want to access to this information, so ideally most of the current should flow through the Bit Line, R_{sens} but data is kept in pn junctions, then certain current is dissipated through them even if they are OFF, and bearing in mind that the matrix we are working with, has significant dimensions and consequently has many diodes, though all the unselected junctions were open, these could contribute to a parasitical current which we want to see in which grade can this affect when reading out a bit.

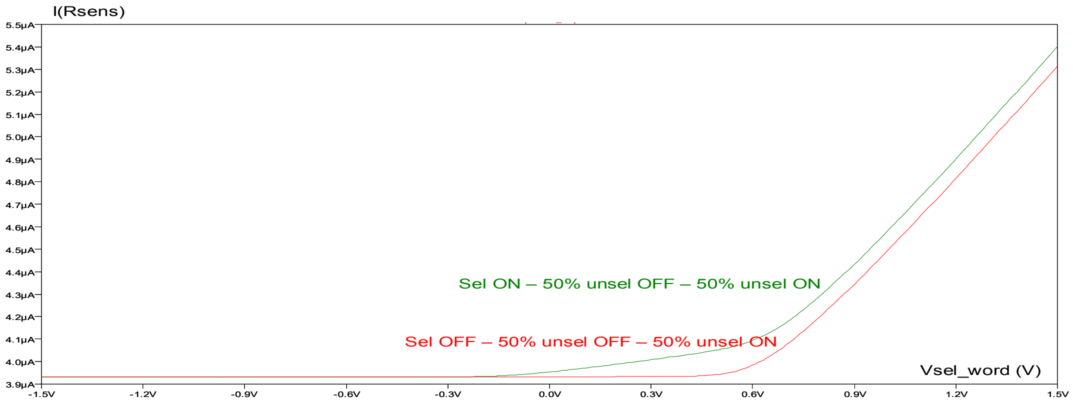


Figure 15: ***Current on Rsens depending on the Voltage applied in the selected Word Wire***

We can observe that for $V_{dd} \geq 0$ we could already distinguish between an open junction (diode OFF) and a closed junction (diode ON). When positive voltages applied to the scheme, a correct distinction between closed junction (read cell) and open junction (unread cell) can be assured. In this study $V_{dd} = 1.5V$ has been considered: this enables a correct read out of the selected bit and doesn't contribute to a extremely high current flowing through the unaccessed lines, as observed in Figure 16.

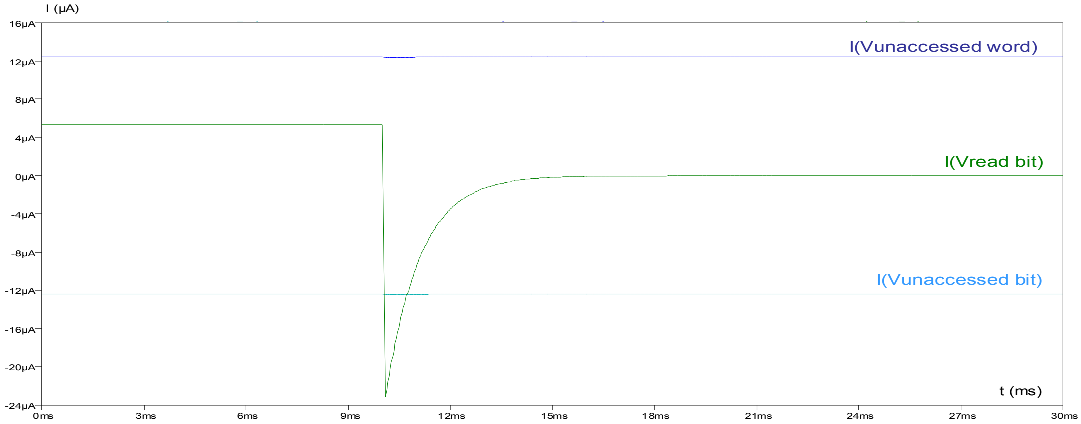


Figure 16: ***Current on Rsens (red), unaccessed bit (turquoise), unaccessed word (blue) when reading a bit***

In order to read-out a bit: selected cell, first biased and then unbiased and supposing from the unselected junctions, 50% are open (OFF) and 50% closed (ON), our aim is to be able to read that specific bit correctly, expecting practically all the current through the accessed Word line, but as we are using diodes, even if these are OFF they dissipate some leakage current so some current will flow through the unselected wires. As we can appreciate form the graph, 12,5 μA flow through the unaccessed bit line, twice bigger than the initial current flowing through R_{sens} but these won't affect in terms on reading out the data, just on an additional warming up of the wiring and consequently of the whole system but won't suppose any loose of data.

4.3.4. Slow read-out and security against invasive external attacks

The main principle of crossbar memories, is to keep vast amount of information safe from attackers: maybe they could manage to reproduce the structure but total characterization won't be possible as we are proposing such a design that forces a very slow read out; any technique the attacker could use in order to decrease this time would seriously damage the structure disabling him to future access.

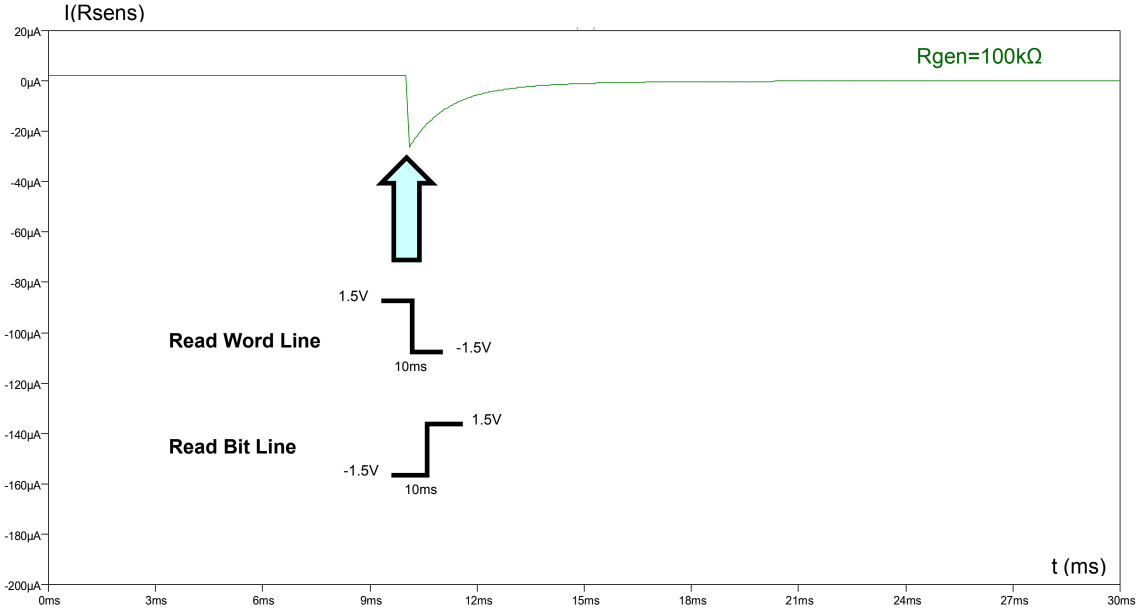


Figure 17: ***Current in Rsens: 1 bit read-out time study***

In order to read out a bit of information, if selected junction is first biased (at 10ms) and subsequently unbiased (at 10,1ms) we can appreciate from Figure 17 it takes approximately 10ms to the diode to reach a constant current value (≈0A) which means 1 bit information could be read in $k = 10ms / bit$. Considering $N = 10^5$ which means 10Gbits of total information, this entails a total time to read out the entire information (N/k) of $T_{full} = 10^8 s$, 3 years and 62 days, too long for the attacker to characterize the crossbar completely, as well as this time could be longer than the application lifetime of the circuit $T_{full} = T_{lifetime}$

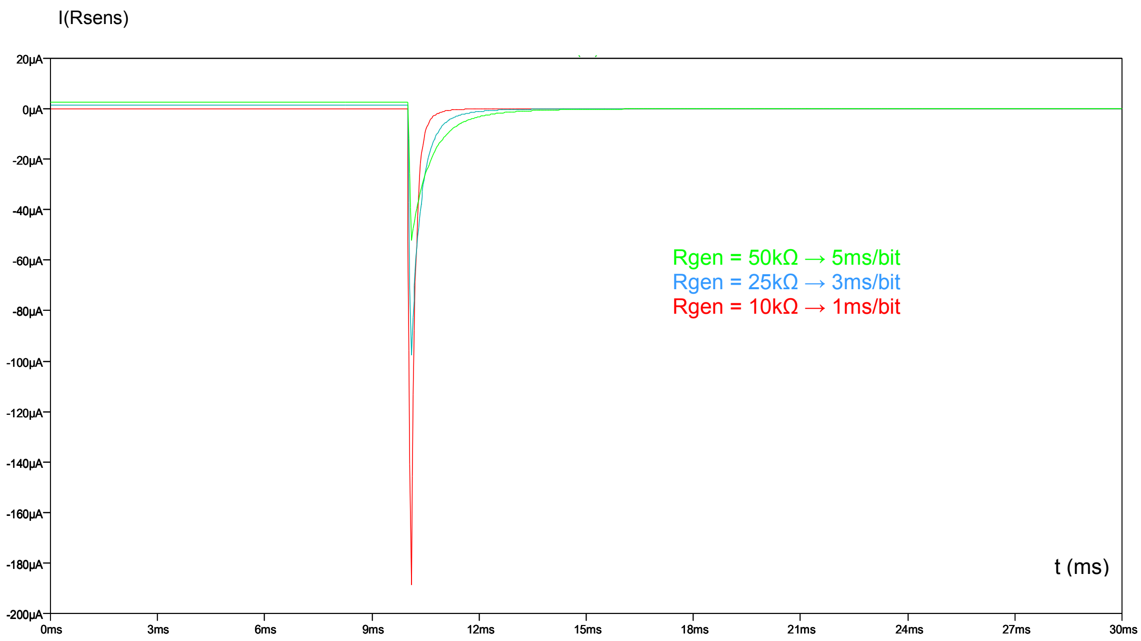


Figure 18: ***Currents in Rsens for different rgen values : 1 bit read-out time reduction analysis***

However, the attacker could manipulate, or entirely replace the read-out circuitry of the memory in order to get quick access to its content. This can be done by using a smaller value for R_{gen}, which means a smaller time constant τ, and so, a smaller time needed to read a bit. He could also read out multiple bits in parallel. Both of these attacks can be prevented if the wires have only a limited $i_{in}^{\max}$ current-carrying capability, and they are destroyed if a higher current is forced through them. The $i_{in}^{\max}$ current limit could be a few times larger than the $i_{static} = V/(R_{gen} + R_{on})$ steady-state current flow.

As we can see in the last figure, decreasing the value of R_{gen} will imply a quicker read out of the bit, but will result in a capacitive current peak at the charging of the wire and could destroy the wire, banning completely the access to data. On the other hand, reading multiple bit values (m) simultaneously, in parallel, loads the corresponding bit / word line with i_{static}^{m} current which exceeds the value of $i_{in}^{\max}$ already for small m, having the same consequences as when reducing R_{gen}'s value.

In addition, the attacker can apply smart read-out techniques, such as pre-charging bit/word lines, but to keep a reasonable noise margin he is always forced to use the read-out scheme. It cannot be avoided to charge an entire bit/word line at each read-out step and to escape the limitation imposed by τ.

4.4. Conclusions

As we have observed from the different results obtained from Spice circuit simulations, it is demonstrated a Crossbar Memory could be a suitable proposal to a SHIC system, as this let's the user keep safe a huge amount of data from possible external attacks; security which mainly relies on slow read-out which invalidates the attacker from characterizing it completely. However, when gained temporal access to the crossbar array, the intruder could reduce this read-out time but he is always limited to a time constant threshold: making τ smaller, means necessity of applying higher current to the wires, action which could suppose a precharging of the wire resulting on its destruction.

In addition, if the crossbar is fabricated by using art lithographic technology, tampering with the internal parts of the crossbar array (such as cutting it into pieces or separating the storage layer from its contacts) seems to be technologically impossible, even for an adversary with unlimited resources. Future considerations related to the physics of the crossbar could be made, like using other types of diodes and metallization as commented in .

Chapter 5

PHOTODIODES: APPROACH TO A UNIQUE OBJECT

In this chapter, we will focus on characterizing a UNIQUE object. Our goal is to implement a physical object with significant strong properties that make it extremely difficult to be reproduced into another physical object: we must stress that a secure unique object must not only prohibit exact reproduction, but also imitation by different physical structures, even by reactive electronic systems that actively generate an imitation signal. All these requirements make the task of designing unique objects so involved, and besides make us think on Photodiodes (PDs) as possible candidates to characterize a unique object as at first glance, seem to achieve the minimum conditions that make an object unique.

PDs are semiconductors devices that can detect optical signals through electronic processes and can be basically based in three processes:

- Carrier generation by incident light.
- Carrier transport and/or multiplication by whatever current-gain mechanism may be present.
- Interaction of the current with the external circuit to provide the output signal.

They are important as they convert the optical variations into electrical variations, which are subsequently amplified and further processed. Therefore, photodetectors must satisfy stringent requirements such as high sensitivity at operating wavelengths and high response speed.

As we are in front a very new idea, and no other previous research related to UNIQUE has been made, a first theoretical basis to PDs will be given in addition to the main idea of the proposal and will complete the study with analog-circuit simulations on SPICE and device-level simulations on Sentaurus TCAD, in order to conclude we reached our aims.

5.1. Photodiodes: introduction to the idea

In a PD signals can be easily applied and it is capable on converting an optical input (light) into an electrical output (charge measurement). The main principles of the idea are summarized:

- A fast photodiode excited with subnanosecond light pulses
- Read-out of the total charge of the system (can be done slowly)
- Trial of building a photodiode, where the net charge is sensitive to the timings

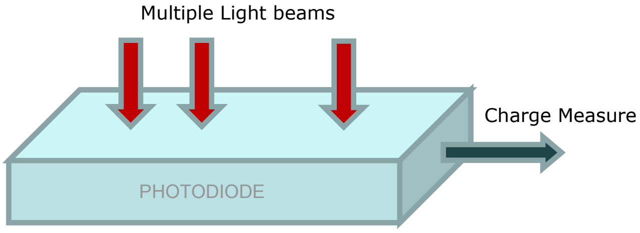


Figure 19: ***Scheme for a PD: different light inputs and electrical measurement***

Before we begin any study, a brief introduction to Photodiode's basics will be given in the next point.

5.2. Security assessment

If the attacker wants to fake the system he could do it in two possible ways:

- Reproduction: this means extracting a copy of the structure.

- Imitation: this implies obtaining a new structure that gives him the same output as the one given from the original system.

So our intention is to assure security in front of any of these two methods:

- When reproduction: looking for a complex and randomized object significantly difficult to be easily refabricated.
- When imitation: looking for a especially fast object that makes it difficult to obtain the same output even if the attacker uses the most ultimate ICs.

5.3. Theoretical basics of Photodiodes

A photodiode is a p-n junction whose reverse current increases when it absorbs photons: If we consider a reversed biased p-n junction under illumination we will observe that photons are absorbed everywhere with an absorption coefficient α. Whenever a photon is absorbed, an electron-hole pair is generated. But only where an electric field is present can the charge carriers be transported in a particular direction. Since p-n junction can support an electric field only in the depletion layer, this is the region in which is desirable to generate photocarriers.

There are, however, three possible locations where electron-hole pairs can be generated:

- Electrons and holes generated in the depletion layer quickly drift in opposite directions under the influence of the strong electric field always points in the n-p direction, electrons move to the n side and holes to the p side. As a result, the photocurrent created in the external circuit is always in the reverse direction (from the n to the p region). Each carrier pair generates in the external circuit an electric current pulse of area since recombination does not take place in the depleted region.
- Electrons and holes generated away from the depletion layer cannot be transported because of the absence of an electric field. They wander randomly until are annihilated by recombination. They do not contribute a signal to the external electric current.
- Electron-hole pairs generated outside the depletion layer, but in its vicinity have a chance on entering the depletion layer by random diffusion. An electron coming from

the p side is quickly transported across the junction and therefore contributes a charge to the external circuit. A hole coming from the n side has a similar effect.

Devices are often constructed in such a way that the light impinges normally o the p-n junction instead of parallel to it.

A photodiode has a I-V relation given:

$$I = I_s \left[\exp\left(\frac{eV}{k_B T} \right) - 1 \right] - I_p \tag{1}$$

Equation which corresponds to the usual I-V curve for a p-n junction (Shockley diode equation), but now, this presents an added photocurrent $—I_p$ proportional to the photon flux.

There are basically three classical modes of photodiode operation:

- **Open circuit (photovoltaic)** The light generates electron hole-pairs in the depletion region. So the flow of photocurrent out of the device is restricted and a voltage builds up. The diode becomes forward biased and dark current begins to flow across the junction in the direction opposite to the photocurrent. This mode is responsible for the photovoltaic effect, which is the basis for solar cells.
- **Short circuit (V=0)** The current here is just the photocurrent *Ip*.
- **Reverse biased (photoconductive)** This increases the width of the depletion layer, which decreases the junction's capacitance resulting in faster response times. The reverse bias induces only a small amount of current (known as saturation or back current) along its direction while the photocurrent remains virtually the same. The photocurrent is linearly proportional to the luminance.

In addition, two main parameters characterize a photodiode:

Quantum Efficiency and Responsivity

Referred to the number of electron-hole pairs generated per incident photon:

$$\eta = \frac{I_p / q}{P_{opt} / h\upsilon} \tag{2}$$

I_p is the photogenerated current by the absorption of incident optical power P_{opt} at a wavelength λ, corresponding to the photon energy $h\upsilon$. We must pay special attention in the responsivity, which is the ratio of the photocurrent to the optical power:

$$\Re\left[\frac{A}{W}\right] = \frac{I_p}{P_{opt}} = \frac{\eta q}{h\upsilon} = \frac{\eta\lambda(\mu m)}{1.24} \tag{3}$$

Therefore, for a given quantum efficiency, the responsivity increases linearly with the wavelength. In addition, the absorption coefficient α is one of the key factors that determines the quantum efficiency and is a strong function of the wavelength.

Response Time

A photodiode takes a certain amount of time to respond to a sudden change in light levels. It is common practice to express to response time in terms of the rise time (tR) or the fall time (tF) where:

tR = time required for the output to rise from 10% to 90% of its final value.

tF = The time required for the output to fall from 90% to 10% of its on state value.

The response time of a photodiode depends upon many factors, including the wavelength of the light, the value of the applied voltage across the diode (since this has a major effect of the junction capacitance), and the load resistance.

Simulation results, as well as measurement data, will eventually show us, that the response time is changing only slightly with the position and the size of the illuminated spot size, when very short light pulses are used. After the incidence light is over, the discharging of the photodiode continues with a time constant, which is highly independent of the size and the location of the illuminated part of the photodiode.

5.4. Analog circuit simulations with SPICE

First of all, we should verify a photodiode gives us what we are looking for: an object with very fast internal dynamics but which doesn't need to be characterized very quickly. To evaluate this, we will take an already existing PD model which was studied by an important Japanese research group in 2003 : They studied the characteristics of InAlAs / InGaAs / InP Heterostructure Metal-Semiconductor-Metal Photodetectors.

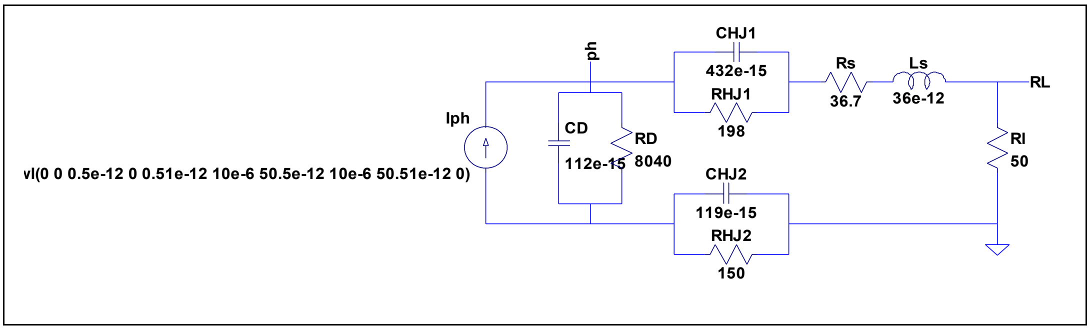
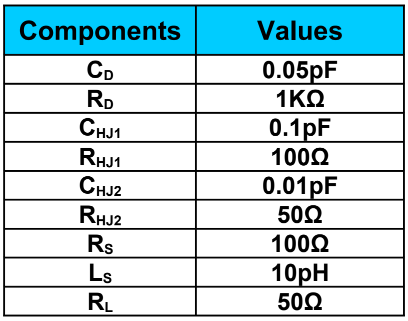


Figure 20: ***Small equivalent circuit for a MSM photodetector***

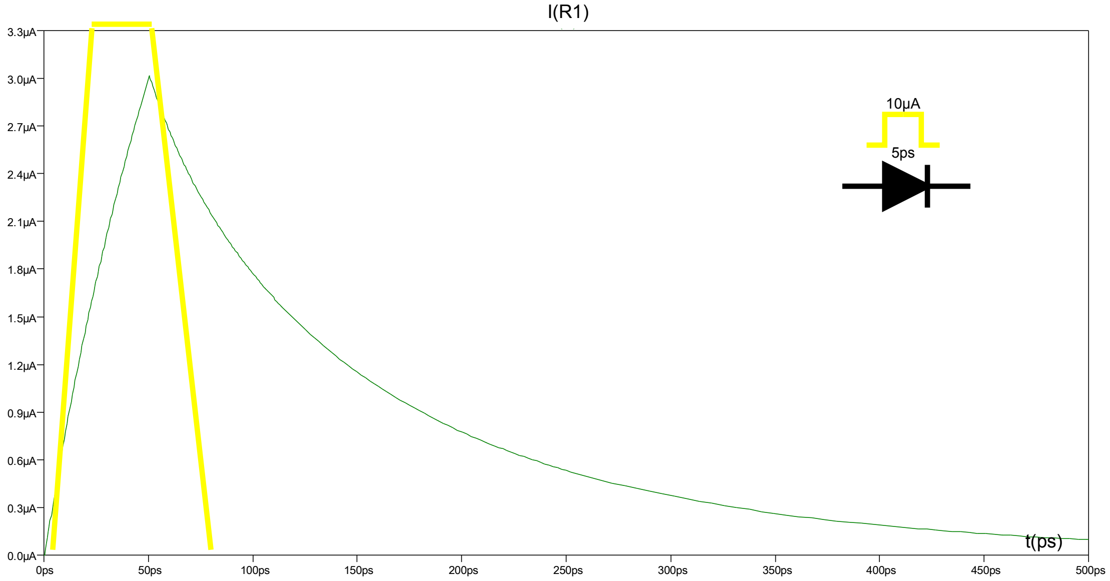


Components	Values
C_D	0.05pF
R_D	1KΩ
C_{HJ1}	0.1pF
R_{HJ1}	100Ω
C_{HJ2}	0.01pF
R_{HJ2}	50Ω
R_S	100Ω
L_S	10pH
R_L	50Ω

Figure 21: ***Values used for 1PD simulation with SPICE***

I_{ph} corresponds to the photocurrent, R_s the series resistance and L_s the series inductance due to the narrow electrode structure. R_D and C_D correspond to the resistance and capacitance in the depletion layer and R_{HJ} and C_{HJ} the resistance and capacitance at the heterobarrier, respectively, with suffix "1" and "2" meaning two heterobarriers, one for electrons and the other for holes, and R_L is the load resistance. The heterobarriers in the PD cause trapping on the photo-generated electrons and holes at conduction band discontinuity and valence band discontinuity, respectively, and negative and positive electronic charge are stored there. The charge storage is expressed by the capacitance C_{HJ}. The trapped electrons and holes then overcome the heterobarrier due to thermionic emission and then drift toward the electrodes.

We will apply a light pulse to the structure during a very small period of time, in order to see which the device's reaction to this optical input is: this should be extremely quick, if applying a pulse of picoseconds large.



access and duplicated. As a next approach, device-level simulations will be made: Sentaurus TCAD is a suitable simulator for defining or modifying the architecture of the device (doping concentration, contacts, etc) and enabling an internal analysis of the structure in order to understand its behaviour clearly. These concretely, could not be made with SPICE as, this tool doesn't support such options. Therefore, Sentaurus TCAD will be the simulator tool used in the next paragraphs.

5.5. Device-level simulations with Sentaurus TCAD

Once we've proved a photodiode could be an adequate candidate for UNIQUE we will now proceed to make some device-level simulations with Sentaurus TCAD in order to approach the nearest solution to our necessities: we are intended to design a photodiode where the net charge is sensitive to the timings of the light beams applied to it, which will give us a random and unpredictable reaction to light resulting in a very secure structure.

5.5.1. PN junction device analysis

Let's see now, what happens when we apply a light pulse (a vertical photo beam) to a simple pn junction: for such study we will use the pn junction presented in the previous chapter when introducing Sentaurus simulator.

Let's briefly sum up the new commands and functions added to the original *_des.cmd file: To indicate we are generating optical energy to the structure, we will use the OptBeam statement which is specified in the region or material Physics section: let's briefly comment the different keywords used for such statement.

- `WavePower [W/cm²]`: This specifies the incident wave power.

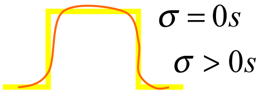
- `SemAbs(model=ODB)`: for this case, Dessis takes the absorption coefficient from the table based optical database Table ODB, in the Dessis parameter field.

- `SemSurf [cm]`: indicates the coordinate of semiconductor's surface.

- `SemWindow [cm]`: specifies the semiconductor window.

- `WaveTime [s]`: implies the interval when the incident beam is constant.

- `WaveTsigma [s]`: Decay of the incident beam. We will use a value different from 0 as for this value the Poisson electron hole equation doesn't converge.

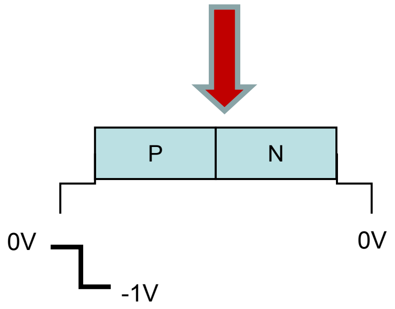


Supposing the junction is reversed biased: Ncontact with 0V and Pcontact with a step voltage, which initially is 0V and rapidly goes to -1V.

Here, a peak optical power is taken from the SWB parameter Power. The optical wavelength is set to $0{,}9\mu m$ the absorption depth is computed automatically using the optical parameter database (ODB) of Sentaurus Device. The parameter SemSurf, as previously explained, defines at which position the peak optical generation is applied (here, at the top of the pn-junction diode). The lateral extent of the light beam is controlled by specifying a window. The temporal profile of the light pulse will be defined with the options WaveTime and WaveTsigma. Here, the pulse is at its peak value in the interval from 10 ns to 15 ns. At its beginning and end, the pulse is phased in and out with a Gaussian profile with a standard deviation of 1 ns.

Behaviour of one PD to one light input

Let's start from the simplest case where we apply one light pulse to the pn junction: this should have a quick internal reaction to the optical input and subsequently, knowing the dissipated current through the diode, charge measurements could be made. Annex IV



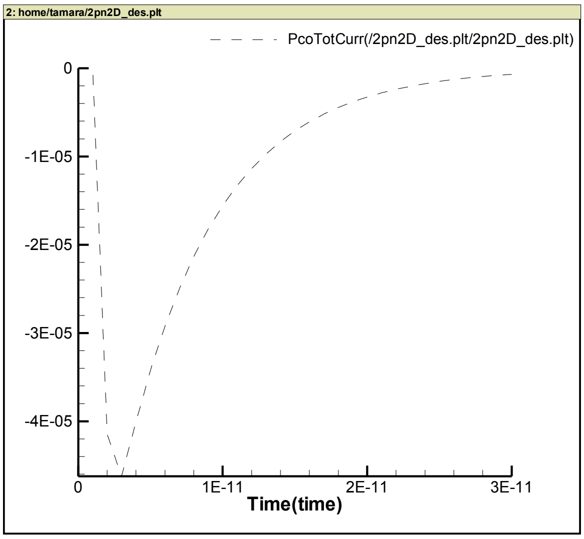
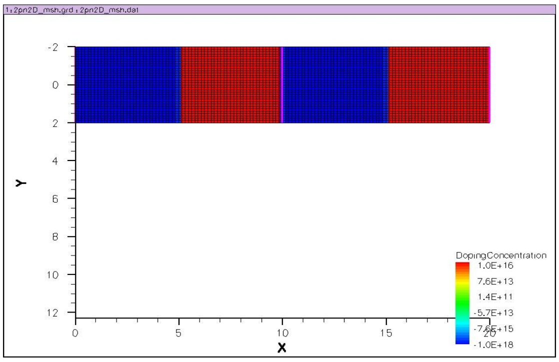


Figure 23: ***I(A)-t(s) 2D curve for transient optical generation. Light beam in (10,15)ns. 0V to -1V pulse in Pcontact. 0V in Ncontact.***

As we can see from the results, a pn junction reacts rapidly to a light excitation as wanted. Now we could go for more complex results, we want to observe what would happen when we have more than one PD and more than one light excitation. When using two identical PDs and introducing two identical light beams, the behaviour should be quite predictable: whatever the overlapping in time between the beams, the total net charge would be two times the charge obtained when having one PD and one light impulse: $Q'=2Q$. We want to prove such premise in order to progress in our design and approach to designing a Photodiode which could present a not so trivial behaviour when optical input applied.

Behaviour of two PDs to two light inputs

We would like to analyse what would happen if we had two PDs and added a light impulse to each other, let's see if we obtain the results we expected to have when considering two identical PDs and two identical light inputs in the middle of each diode. Annex V



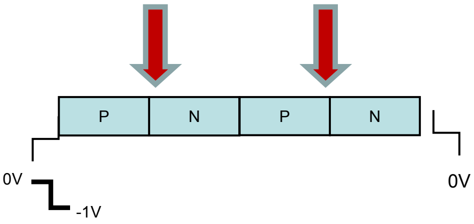


Figure 24: ***2D Sentaurus model for 2 pn-junctions in parallel***

Now the results don't arrive to convergence, so we cannot consider this structure for future development, maybe the fact of having a vertical PN distribution could affect on absorbing correctly the light which enters vertically. Therefore we will propose a silicon diode for next study: now the PN is distributed horizontally.

5.6. Silicon photodiode

A silicon nitride passivation layer is deposited onto the front face, the thickness being chosen so that the layer acts as an antireflection coating for the wavelength of operation. Between the p-type region and the lightly doped n-type region there is a depletion region which is free from mobile charges. The width of this region depends upon the resistivity of the silicon and the applied voltage; even with no externally applied bias the diffusion of electrons and holes across the junction creates a depletion region with an electric field across it which is known as the "built-in" field.

When a photon is absorbed in a semiconductor an electron-hole pair is formed. Photocurrent results when photon-generated electron-hole pairs are separated, electrons passing to the n-region and holes to the p-region. Alternatively, holes and electrons may recombine, thereby causing no charge displacement and thus no contribution to photocurrent. There is a greater probability of separation of a photon-generated electron hole pair when it is formed within the depletion region where the strongest electric field exists.

Planar diffused silicon photodiodes are simply P-N junction diodes constructed from single crystal silicon wafers similar to those used in the manufacture of integrated circuits. In this particular case we will form by diffusing a P-type impurity thin layer, such as Boron, into a N-type bulk silicon wafer. The interface between the "p" layer and the "n" silicon is known as a pn junction: here we have the depletion region which is free from mobile charges. The width of this region depends upon the resistivity of the silicon and the applied voltage; even with no externally applied bias the diffusion of electrons and holes across the junction creates a depletion region with an electric field across it which is known as the "built-in" field.

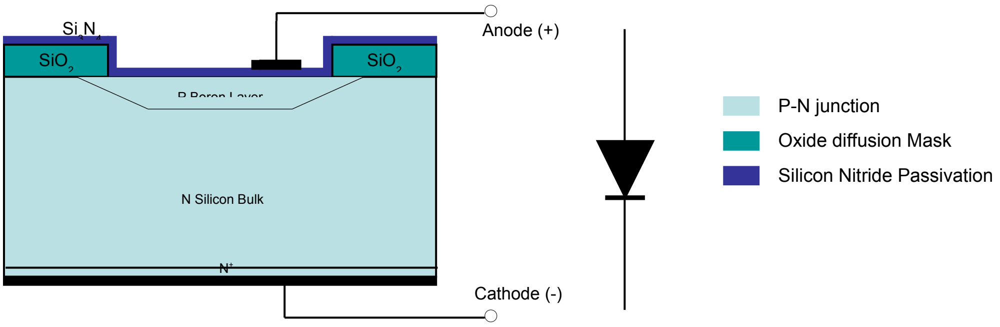


Figure 25: ***Silicon Photodiode Device***

The contact pads are deposited, on the front active area on a defined area (anode) and on the backside, it covers completely the device (cathode). The front contact is normally defined as an aluminium layer and the rear contact is by means of one of a number of alternative multilayer metallization. We can see their location in Figure 26.

We can also indicate which are the materials used in each area, but we are not specially interested on defining them: we are more interested in its architecture and doping characteristics. The active area is coated with silicon nitride (most commonly), silicon monoxide or silicon dioxide for giving protection and to serve as an anti-reflection coating. The active area is then deposited on with an anti-reflection coating to reduce the reflection of the light for a specific predefined wavelength. The non-active area on the top is covered with a thick layer of silicon oxide. By controlling the thickness of bulk substrate, the speed and responsivity of the photodiode can be controlled.

So we are proposing simply a pn junction that can detect presence or absence of light with a low capacitance planar diffusion type: a highly pure, N-type on the backside to enlarge the depletion layer and decrease the junction capacitance.

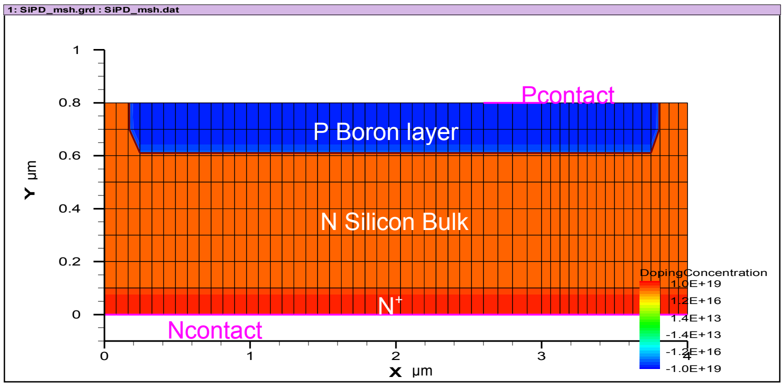


Figure 26: ***Homogeneous Silicon PD structure on Sentaurus***

For the next simulations we will use this model for silicon photodiode: a $4\mu m \times 0.8\mu m$ structure with an Ncontact that covers the whole backside and the Pcontact which is deposited on a defined area in the front side. Annex VI

5.6.1. Silicon PD operation for no light input

Suppose we have no light input in the silicon PD, then this behaves as a normal diode as its I-V curve obeys the Shockley equation: the current grows exponentially with the voltage and it goes to ON state for a forward biased $V > 0.7V$

We can also observe that when making a transient study, after many initial internal changes, after some time, the current converges to its static value, as expected. Annex VII

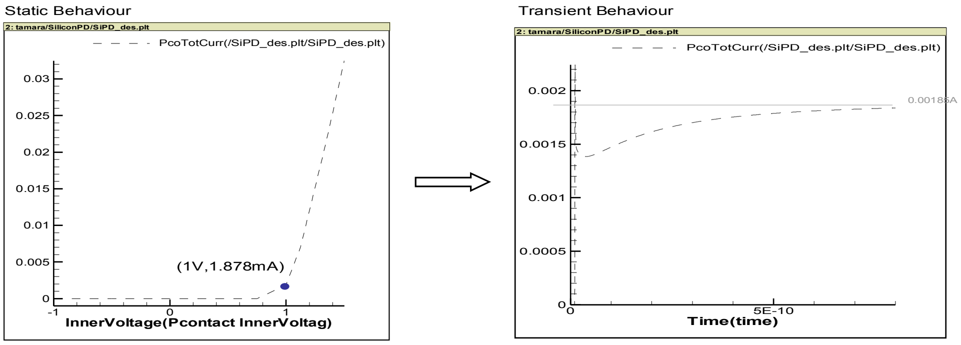
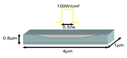


Figure 27: ***No light effect: I-V curve and I-t curve for the PD to contrast results***

5.6.2. Silicon PD operation for light input

Suppose we introduce now a light beam vertically in the center of the photodiode, let's see how electrons and holes behave to this excitement. We will first make an internal analysis of the response of electrons and holes to an optical input and next we will see which is the current curve obtained in order to know which would be the total net charge measured. Annex VIII



Internal behaviour

First of all we want to see how electrons and holes react to a light excitation in order to understand the photodiode's characteristics.

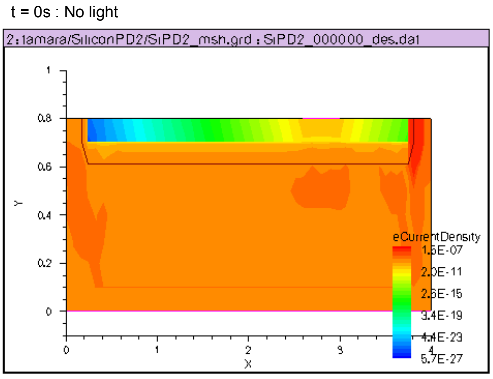


Figure 28: ***Electron current density before any light is applied***

In absence of light, Dark current is the current through the diode, practically no current is generated by electrons either holes. This current is due to the ideal diode current, the generation/recombination of carriers in the depletion region and any surface leakage, which occurs in the diode.

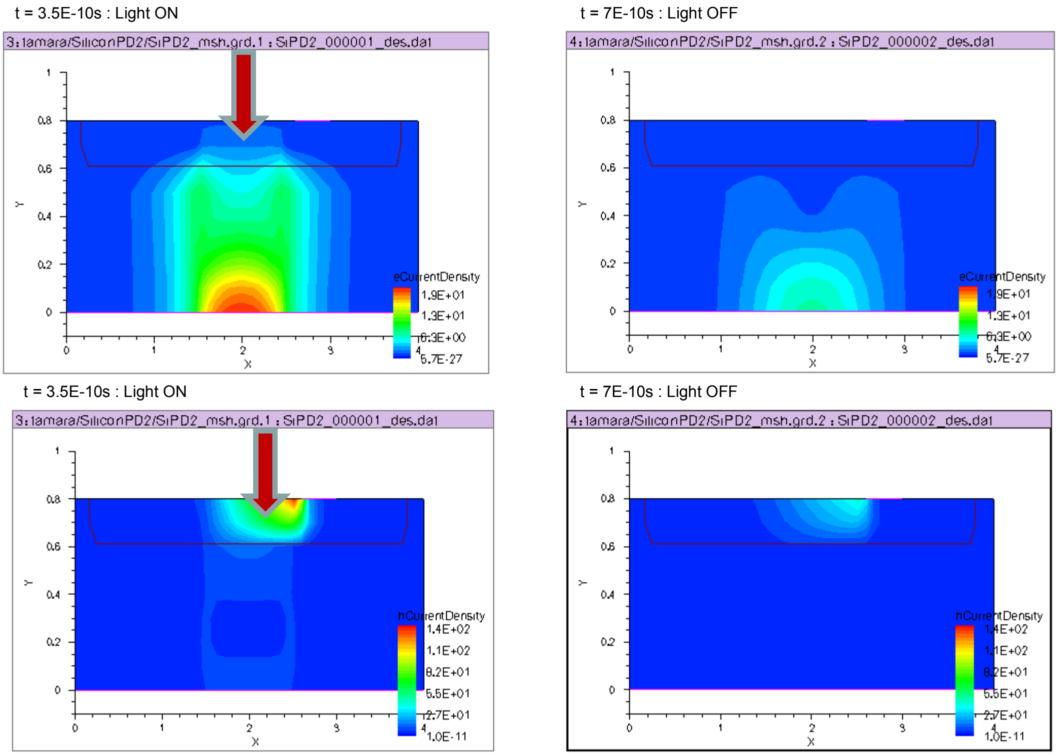


Figure 29: ***Electron current density when light IN and after light OFF***
Hole current density when light in and after light OFF

When light gets in, then electrons begin to give some current increasing from the Ncontact, while the holes do it from the Pcontact, coming most of the current from the holes. When light is gone, then electrons and holes decrease its current to its original state. As a result for a normal silicon photodiode with standard and homogeneous doping distribution (P-layer, N-bulk and thin N-highly-doped layer in the backside) shows also a standard and predictable behaviour of electrons and holes.

Quantum efficiency study

As this is the first time work on PDs with Sentaurus simulator is made in the Nanoelectronics Institute of the TUM, it is important to demonstrate first, that this behaves correctly to different light wavelengths inside the visible light spectrum. To do so, we can obtain Responsivity and Quantum Efficiency values, which define the sensitivity of the photodiode.

$$\text{Responsity} = \Re\left[\frac{A}{W}\right] = \frac{\text{ElectricalOutput}}{\text{OpticalInput}} = \frac{I_p}{P_{opt}}$$

$$\text{Quantum Efficiency} = \eta = 1.24\frac{\Re}{\lambda(\mu m)}$$

	λ = 0.5μm	**λ = 0.6μm**	**λ = 0.7μm**	**λ = 0.8μm**	**λ = 0.9μm**
Imax (A)	16.0E-8	10.5E-8	6.0E-8	3.2E-8	0.4E-8
Responsivity (A/W)	0.160	0.105	0.06	0.032	0.004
QE (%)	40.0	21.7	10.6	4.96	0.55

As we can observe in the next graph, for a wavelength of $0.5\mu m$ we have maximum Quantum Efficiency (40%) and the PD reacts to the light input in a speed of $1ns$, values that guarantee us an optimum absorption of the light and also a quite high internal speed.

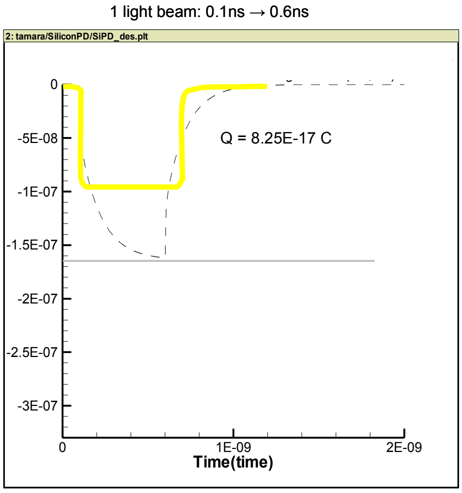


Figure 30: ***Current characteristics for an homogeneous Silicon PD One light input***

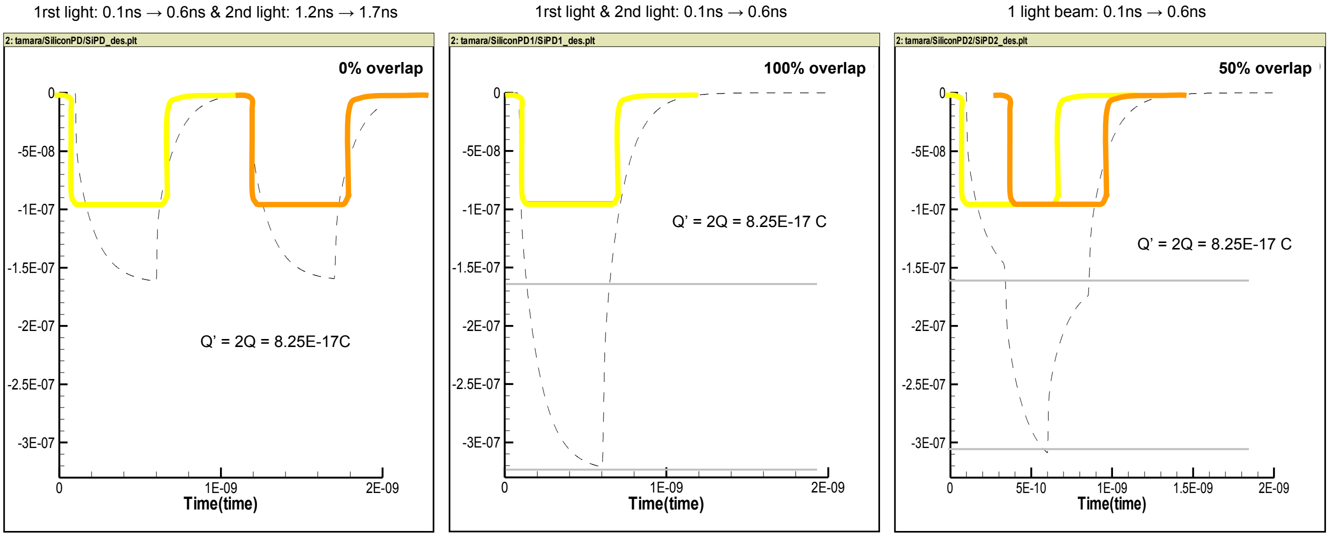


Figure 31: ***Current characteristics for an homogeneous Silicon PD***
Two light beams: (a) 0% overlap (b) 100% overlap (c) 50% overlap

Now we want to see what would happen if we applied two identical light beams to the structure: as the device presents homogeneous doping distribution, then this behaves trivially to light excitation, whatever the overlapping between the lights is the resultant total net charge would be $Q'=2Q$ as estimated. Now we have to make this behaviour much more unpredictable, in order to get closer to reach our goals: to do so, we propose adding some new doping particles to the original silicon PD structure and make the same study for this inhomogeneous new device.

5.7. Proposal of a unique Photodiode

As previously mentioned we want to obtain now a unique device that behaves electrically in a unique way to optical input. Having proved a silicon PD gives us a sufficient high internal speed, let's try to find out now how to obtain a UNIQUE structure.

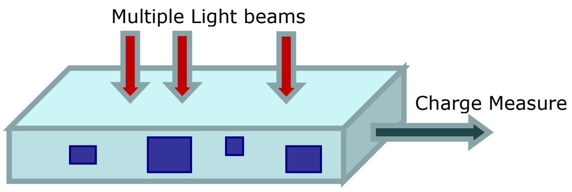


Figure 32: ***Scheme for a PD: different light inputs and electrical measurement***

As said, we will convert the homogeneous original structure into an inhomogeneous object, by adding new dopings to it. As observed when analysing the electrons-holes behaviours, most of the changes were given in the area near the contact location and the light entry, so we will add some doping particles in those areas to see which would be the effect:

Annex IX

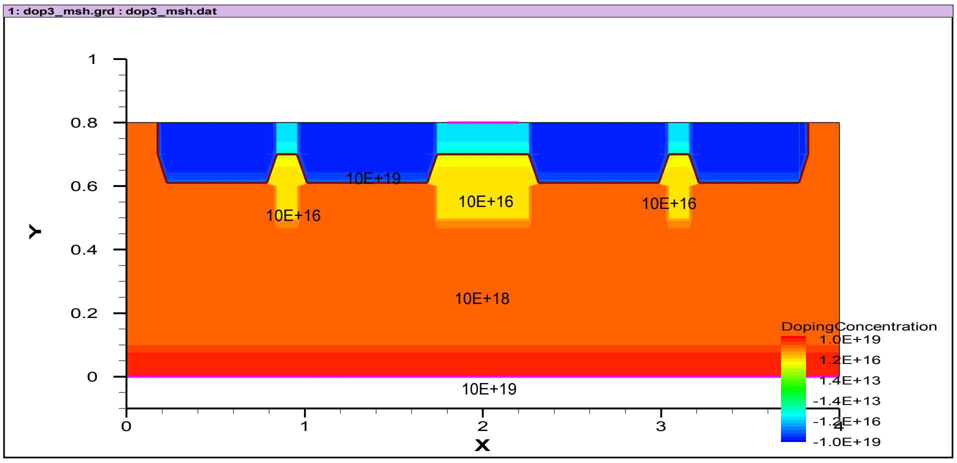


Figure 33: ***Inhomogeneous Silicon PD structure on Sentaurus: doping particles added to the original device***

Now the new object presents some very low dopings located near the Pcontact and in the areas where light will be injected, occupying principally the depletion area, where most of the changes are done. To analyse it we will act in the same way as for the homogeneous structure: we will introduce two identical lights in different timings (different percentage of overlapping between them) in order to see which the electrical result is now.

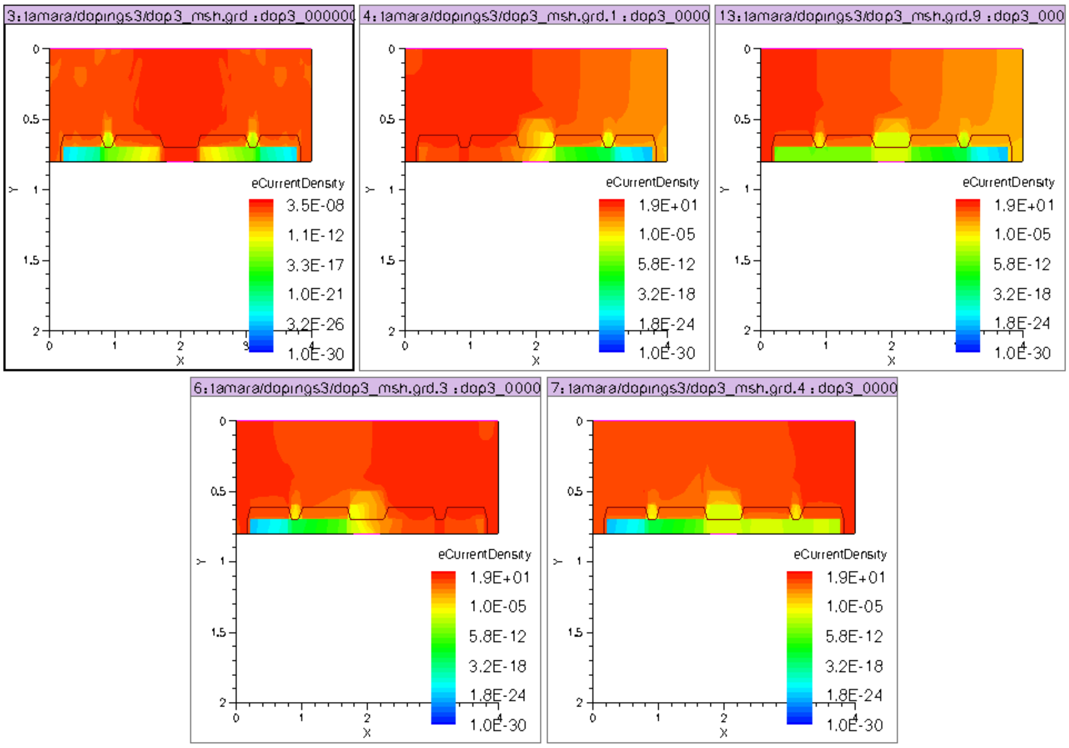


Figure 34: ***Electron Current density for an inhomogeneous PD (1) no light (2) 1rst light IN (3) 1rst and 2nd lights IN (4) 1rst light OFF (5) 2nd light OFF***

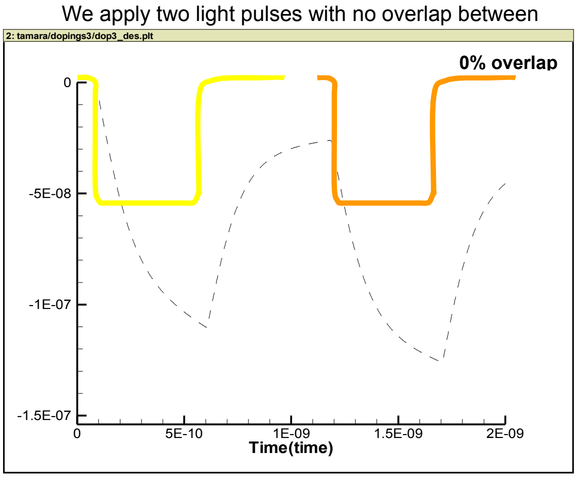


Figure 35: ***Current characteristics for an inhomogeneous Silicon PD (0% overlap)***

As we can appreciate for an inhomogeneous structure we don't have a symmetrical reaction to light; there's certain "tail" before reaching original state when light is gone, because of the doping we added. So now, when the lights present 0% we cannot say the total charge is two time one charge when only one impulse was introduced. The same will happen for other overlap as we can appreciate from the next graphs:

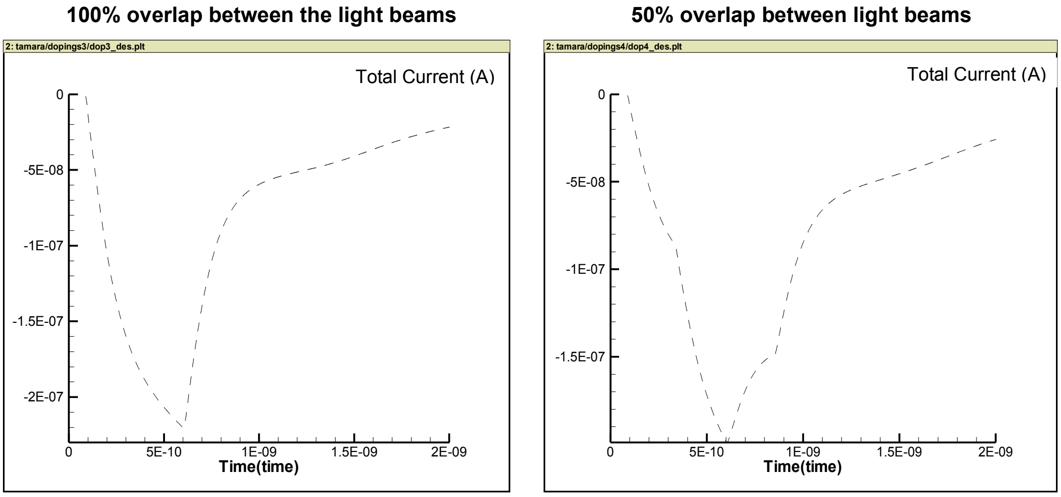


Figure 36: ***Current characteristics for an inhomogeneous Silicon PD (100% and 50% overlap)***

So now we are nearer to our objectives as for an inhomogeneous PD, the behaviour is not that trivial: for identical light pulses, Total Charge ≠ $\sum_{n} Q$

As a next step let's demonstrate that for one structure we have one unique response, not two different objects can give us the same electrical output: this will guarantee what we are looking for, a UNIQUE object.

5.8. UNIQUE object achievement

Getting near to our goals, we'll proceed to just change the particles' location (the structure is has the same architecture as the device presented in the previous point); the optical inputs will be the same as in the previous case, only position of the low dopings has changed:

Annex X

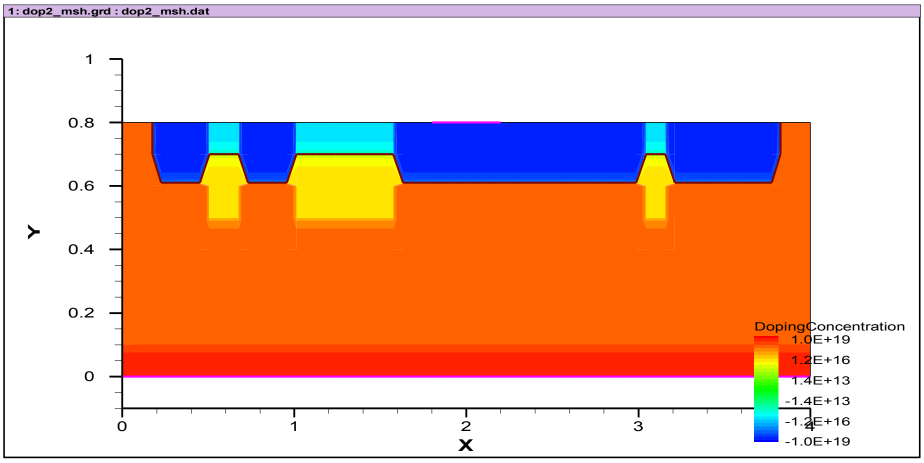
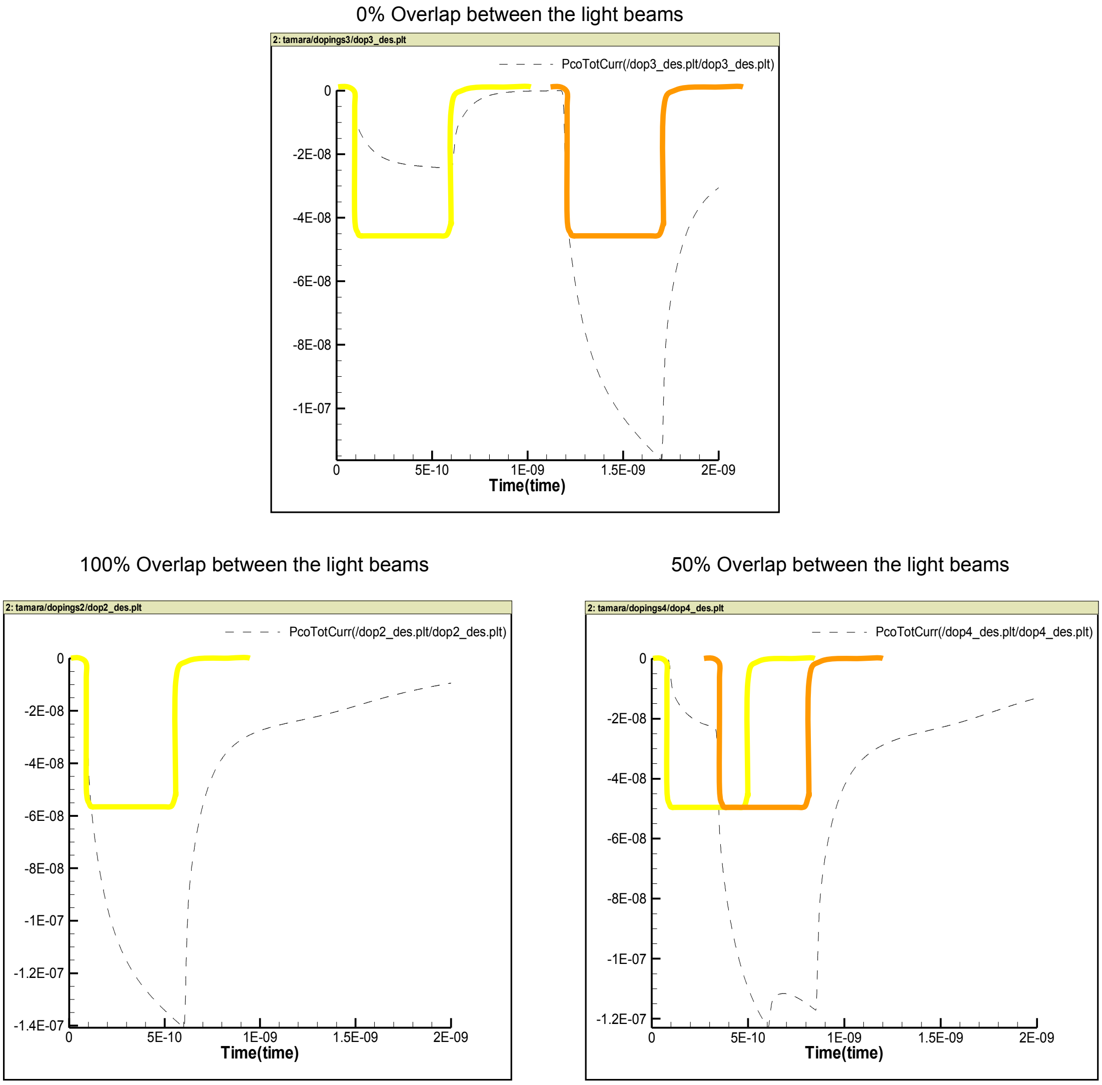


Figure 37: ***Unique Silicon PD structure on Sentaurus: redistributin of doping particles added***

Now we located more doping particles in the left side of the PD, so the reaction to light will be different on each area even if the light beams we are applying are totally identical, now most of the current is dissipated when the second light gets in. Therefore we have no more symmetrical behaviour and no more trivial behaviour in consequence.



0% Overlap between the light beams

100% Overlap between the light beams

50% Overlap between the light beams

Figure 38: ***Current characteristics for a unique Silicon PD (0%, 100% and 50% overlap)***

In addition for another distribution we obtained another charge measurement, which takes us to conclude with photodiodes we could create an UNIQUE object: random enough to make it safe from copy and quick enough to keep it safe from imitation.

5.9. Future considerations of design

Arrived to this point, we should remark we've been using silicon for implementing the previous diodes' designs, as we wanted to optimize the computational time needed to obtain simulation results. Ideally we've seen Silicon diodes could reach our necessities but practically other materials such as Gallium Arsenide or Germanium would be much more adequate materials for a PD: they absorb better the light and would give us much more adjusted results. If we think on using Gallium Arsenide (GaAs) as a substitute: economically this results much more expensive than acquiring Silicon, but in contrast GaAs contributes to important benefits:

- a high substrate bulk resistivity providing isolation and minimises parasitic capacitance
- Increased speed
- Increased temperature tolerance; reduced power dissipation
- Improved radiation hardness.

The principal benefit of using GaAs in devices is that it generates less noise than most other types of semiconductor components and, as a result, is useful in weak-signal amplification applications.

5.9.1. Gallium arsenide advantages

In the next lines we expose some electronic properties which make GaAs a special material to take into account:

- It has a higher saturated electron velocity and higher electron mobility.
- GaAs devices generate less noise than silicon devices when operated at high frequencies. They can also be operated at higher power levels than the equivalent silicon device because they have higher breakdown voltages.
- It has a direct band gap, which means that it can be used to emit light efficiently. Silicon has an indirect bandgap and so is very poor at emitting light.

5.9.2. Silicon advantages

We can stress three major advantages of silicon over GaAs for integrated circuit manufacture:

- It is abundant and cheap to process. The economy of scale available to the silicon industry has also reduced the adoption of GaAs.
- It is assumed the existence of silicon dioxide, one of the best insulators, which can easily be incorporated onto silicon circuits, and such layers are adherent to the underlying Si. GaAs does not form a stable adherent insulating layer.
- It possesses a higher electron hole mobility, which allows the fabrication of higher-speed P-channel field effect transistors. Unlike silicon cells, GaAs cells are relatively insensitive to heat.

5.10. Conclusions

After studying carefully a photodiode and its principal characteristics, we could consider them as a possible implementation to UNIQUE objects: They present sufficiently fast internal dynamics that protect the data from possible fake and can be made unique, randomized enough so that even control for the manufacturer could be difficult. As we have seen Unique can be reached by adding new doping particles to a standard PD, having as a result, a very sensitive device to the timings of the optical inputs introduced. Therefore, measuring the total net charge would be extremely difficult. We can obtain one unique object with one unique response to excitation. As previously remarked, all this study has been realized using the most simple materials and structure, in order to have a first approximation of the idea: future study could be made to reach a much robust system (physical characteristics, device's architecture, etc.)

CONCLUSIONS

With “Circuit approaches to Physical Cryptography” we basically wanted to introduce the reader another innovative and attractive technique of Cryptography and Security. Personal work has been documented and completed with already existing work made by other research groups, but specially the one made by György Csaba and Ulrich Rührmair in the Nanoelectronics Institute inside the TUM (Technische Universität München). In addition, all new data has been contrasted with simulator tools in order to support our proposals and different implementations here presented.

Physical Cryptography is still in its first years, so many studies related to it can be made and many techniques based on it can be given. However SHIC systems and UNIQUE objects, the techniques on analysis in this document, are a good example to demonstrate the advantages on using micro-nanostructures for security. The different structures proposed are only a first step which can be completed and improved in future research study. Here we propose the main basis for design but further considerations like other more physical materials for the structures ore modifying its architecture, could be made.

BIBLIOGRAPHY

Paper Sources

[1]. ***A Method for Obtaining Digital Signatures and Public-Key Cryptosystems. Communications of the ACM, vol. 21 (2), pp. 120-126, 1978.*** R. Rivest, A. Shamir, L. Adleman.

[2]. ***Physical One-Way Functions, 2002.*** Ravikanth Pappu, Ben Recht, Jason Taylor, Neil Gershenfeld

[3]. ***Extracting Secret Keys from Integrated Circuits*, 2005.** Daihyun Lim, Jae W. Lee, Blaise Gassend, G. Edward Suh, Marten van Dijk and Srinivas Devadas

[4]. ***Physical Uncloneable Functions for Device Authentication and Secret Key Generation, 2007.*** G. Edward Suh and Srinivas Devadas

[5]. ***SHIC Systems and their applications to cryptography and security. Internal Manuscript, 2007.*** Ulrich Rührmair.

[6]. ***Internal Manuscript, 2008.*** G. Csaba, J. J. Finley, C. Jirauschek, P. Lugli, U. Rührmair and M. Stutzmann: Physical Realizations of SHIC systems.

[7]. ***Read-Proof Hardware from Protective Coatings, 2006.*** Pim Tuyls, Geert-Jan Schrijen, Boris Skoric, Jan van Geloven, Nynke Verhaegh, and Rob Wolters

[8]. ***Read-out design rules for molecular crossbar architectures, 2008.*** György Csaba and Paolo Lugli

[9]. ***Cryptographic Applications for High-Capacity Crossbar Memories, 2009.*** Ulrich Rührmair, György Csaba, Christian Jaeger, Martin Stutzmann, and Paolo Lugli

[10]. ***Equivalent Circuit Model of InAlAs / InGaAs / InP Heterostructure Metal-Semiconductor-Metal Photodetectors, 2003.*** Koichi Iiyama, Junya Ashida, Akira Takemoto and Saburo Takamiya

Literary Sources

"Fundamentals of Photonics", B.E.A. Saleh and M.C. Teich, Wiley Series in Pure and applied Optics, J.W. Goodman Editor.

"Principles of Semiconductor devices", B. Van Zeghbroeck, 2004, ECEE Colorado

ANNEXES

Annex I Creating subcircuits with Schematics

Procedure:

1) Create a netlilst using a text editor such as notepad.

   a) use the *.subckt* statement followed by the name of the subcircuit followed by the pins (e.g. *.subckt myopamp 1 2 3*). The numbers will be used as node numbers for connection to other components (0 is ground). The order of the numbers will be the order used for the "pin/port netlist order" on the symbol.

   b) End the subcircuit with the *.ends* statement (e.g. *.ends myopamp*).

   c) Save the netlist as *filename.lib* or *filename.sub,* the name does not have to be the same as the name of the subcircuit (in fact, more than one subcircuit can be included in a .lib or .sub file. See "Notes on Saving"

2) Draw a symbol to represent the device.

   a) Select File: New Symbol, to open the Symbol Editor.

   b) Draw the symbol using the Draw menu and the Line, Rect, Circle, etc. commands.

   c) Place the pins. Select Edit: Add Pin/Port to get the Pin/Port Properties dialog box. At the top right is the Netlist Order box. This order corresponds to the order of the pins in the subciruit. A label can be inserted using the Label box (e.g. Vin), but has nothing to do with the function of the pin. The pin's function relates to the connection specified in the subcircuit netlist. To make the label visible select one of the Pin Label Position buttons.

   d) Some information, such as, Instance Name can be added by selecting Edit: Attributes: Attribute Window and selecting the information. Clicking OK will allow pasting on the symbol

3) Open the Symbol Attribute Editor to enter the appropriate information.

a) Select Edit: Attributes: Edit Attributes.

i) Select Cell in the Symbol Type drop-down box.

ii) Select Prefix, type X in the Prefix = box. X tells LTspice that this is a subcircuit.

iii) In the Value field type the name of the .subckt (e.g. *myopamp*)

iv) SPICELine can be used to pass parameters.

v) Leave ModelFile blank.

b) Save the symbol, File: Save As saves the file as a *.asy. See "Notes on Saving"

4) On the schematic, include a SPICE Directive with the path of the subcircuit file (e.g. *.lib C:\Program Files\LTC\SwCADIII\My Work\OpampExample.lib*).

Annex II PN junction basics

A p-n junction consists of two semiconductor regions with opposite doping type as shown in the Figure 1. The region on the left is *p*-type with an acceptor density N_a, while the region on the right is *n*-type with a donor density N_d. The dopants are assumed to be shallow, so that the electron (hole) density in the *n*-type (*p*-type) region is approximately equal to the donor (acceptor) density.

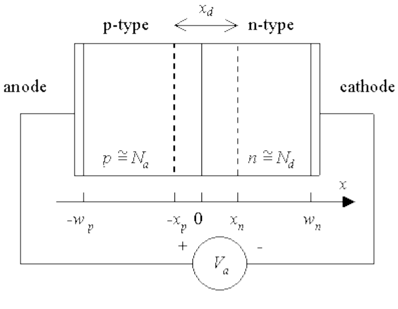


Figure 1. Cross-section of a PN-junction

We will assume, unless stated otherwise, that the doped regions are *uniformly* doped and that the transition between the two regions is abrupt. We will refer to this structure as an *abrupt* p-n junction.

Frequently we will deal with p-n junctions in which one side is distinctly higher-doped than the other. We will find that in such a case only the low-doped region needs to be considered, since it primarily determines the device characteristics. We will refer to such a structure as a *one-sided* abrupt p-n junction.

The junction is biased with a voltage V_a as shown in figure 1. We will call the junction forward-biased if a positive voltage is applied to the *p*-doped region and reversed-biased if a negative voltage is applied to the *p*-doped region. The contact to the *p*-type region is also called the anode, while the contact to the *n*-type region is called the cathode, in reference to the *anions* or positive carriers and *cations* or negative carriers in each of these regions.

Flatband diagram

The principle of operation will be explained using a *gedanken* experiment, an experiment, which is in principle possible but not necessarily executable in practice. We imagine that one can bring both semiconductor regions together, aligning both the conduction and valence band energies of each region. This yields the so-called flatband diagram shown in Figure 2.

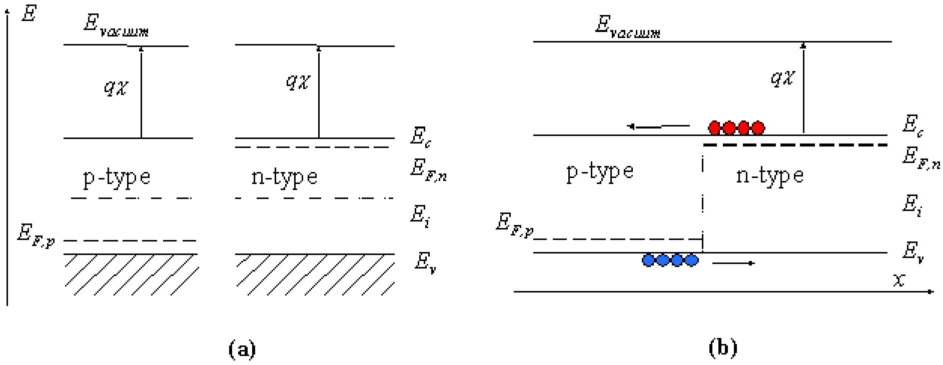


Figure 2. Energy band diagram of a p-n junction (a) before and (b) after merging the n-type and p-type regions

Note that this does not automatically align the Fermi energies, $E_{F,n}$ and $E_{F,p}$. Also, note that this flatband diagram is not an equilibrium diagram since both electrons and holes can lower their energy by crossing the junction. A motion of electrons and holes is therefore expected before thermal equilibrium is obtained. The diagram shown in Figure 2 (b) is called a flatband diagram. This name refers to the horizontal band edges. It also implies that there is no field and no net charge in the semiconductor.

Forward and reversed biased

We now consider a p-n diode with an applied bias voltage, V_a. A forward bias corresponds to applying a positive voltage to the anode (the *p*-type region) relative to the cathode (the *n*-type region). A reverse bias corresponds to a negative voltage applied to the cathode. Both bias modes are illustrated with Figure 3 the applied voltage is proportional to the difference between the Fermi energy in the *n*-type and *p*-type quasi-neutral regions. As a negative voltage is applied, the potential across the semiconductor increases and so does the depletion layer width. As a positive voltage is applied, the potential across the semiconductor decreases and with it the depletion layer width.

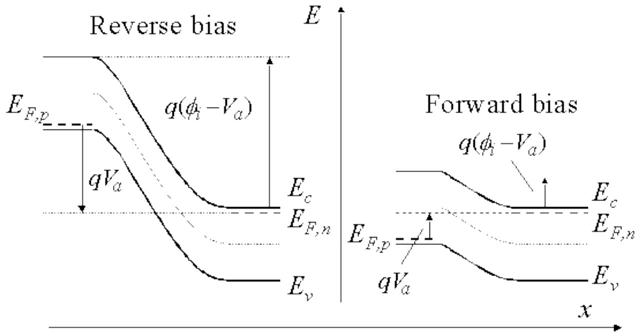


Figure 3. Energy band diagram of a p-n junction under reverse and forward bias

Annex III Crossbar memory design:

N=10^5 bits, unselected junctions: 50% open, 0% closed

LANGUAGE EDITOR DESIGN: .cir file

```
.param scalefactor  1.0E-10
.param Is  {1.0E-7*scalefactor}
.param Rp  {3.0E+4/scalefactor}
.param Rs0 {1.66E-4/scalefactor}
.param Rslow {Rs0}
.param Rshigh {100.0*Rs0}
.param idfac   1.8
.param Cjunction {1.0E-3*scalefactor}
.param Rub 1.0E+5
.param Ruw 1.0E+5
.param Rgen 1.0E+5
.param Rsens 1.0E+5

*This is the dimensionality of the array
.param n 1.0E+5

*The bias voltage
.param Vdd 3.0

*This is the number of parallel-connected diodes
*.param pd 1

.subckt Diode_on in out pd = 1
Bdieq in dout I = pd * Is * (exp( (V(in)-V(dout))/(idfac*0.025) ) -1)
Rpar  in dout {Rp/pd}
Rserial dout out {Rslow/pd}
C1 in out {Cjunction*pd}
.ends

.subckt Diode_off in out pd = 1
Bdieq in dout I = pd * Is * (exp( (V(in)-V(dout))/(idfac*0.025) ) -1)
Rpar  in dout {Rp/pd}
Rserial dout out {Rshigh/pd}
C1 in out {Cjunction*pd}
.ends

*Units connecting from the unaccessed word line
X1on read_word unaccessed_bit Diode_on  pd = {n/2.0}
```

```
X1off  read_word unaccessed_bit Diode_off  pd = {n/2.0}

*The interrogated junction
Xaccessed read_word sense_node Diode_on  pd = 1

Rsens sense_node read_bit 1.0k

*The parallel resistances connected to the same bit line
X2on unaccessed_word sense_node Diode_on  pd = {n/2.0}
X2off unaccessed_word sense_node Diode_off  pd = {n/2.0}

*The rest of the array
Xreston unaccessed_word unaccessed_bit Diode_on pd = {n*n/2.0}
Xrestoff unaccessed_word unaccessed_bit Diode_off pd = {n*n/2.0}

Vread_word read_wordG 0        +1.5 pwl(1.0E-2 +1.5 1.01E-2 -1.5)
Vread_bit read_bitG  0                -1.5 pwl(1.0E-2 -1.5 1.01E-2 +1.5)
Vunaccessed_word unaccessed_wordG 0  pwl(0.0 -1.5)
Vunaccessed_bit unaccessed_bitG 0  pwl(0.0 +1.5)


R1 read_wordG read_word {Rgen}
R2 read_bitG  read_bit  {Rsens}
R3 unaccessed_wordG unaccessed_word {Ruw}
R4 unaccessed_bitG unaccessed_bit {Rub}

.op
*.dc n
.tran 3.0E-2
*.options list node gminsteps=0
*.print i(*)

.print i(Vread_word)
.print i(Vread_bit)
.print i(Vunaccessed_word)
.print i(Vunaccessed_bit)

*.step param Rgen list 10k 100k 1meg
*.dc param Vdd -5.0 5.0 0.1
*.dc Vprobe -2.6 2.6 0.01
*.dc param n LIST 1 10 100
*.print dc ix(di)
*.print dc log(ix(di))
.end
```

SCHEMATICS DESIGN: with subcircuits

Subcircuit corresponding to Selected diode (ON / OFF)

```
.param Isoff  1.0E-17
.param Rsoff  1.66E+8
.param Rpoff  3.0E+14
.param idfacoff   1.8
.subckt SelDiodeOFF v1 vsens
Bdieq v1 dout I = Isoff * (exp( (V(v1)-V(dout))/(idfacoff*0.025) ) -1)
Rpar  v1 dout {Rpoff}
Rserial dout vsens {Rsoff}
.ends SelDiodeOFF

.param Ison  1.0E-17
.param Rson  1.66E+6
.param Rpon  3.0E+14
.param idfacon   1.8
.subckt SelDiodeON v1 vsens
Bdieq v1 dout I = Ison * (exp( (V(v1)-V(dout))/(idfacon*0.025) ) -1)
Rpar  v1 dout {Rpon}
Rserial dout vsens {Rson}
.ends SelDiodeON
```

Subcircuit corresponding to unselected bit / word lines diodes (ON / OFF): N-1 junctions

```
.param Is1  1.0E-17
```

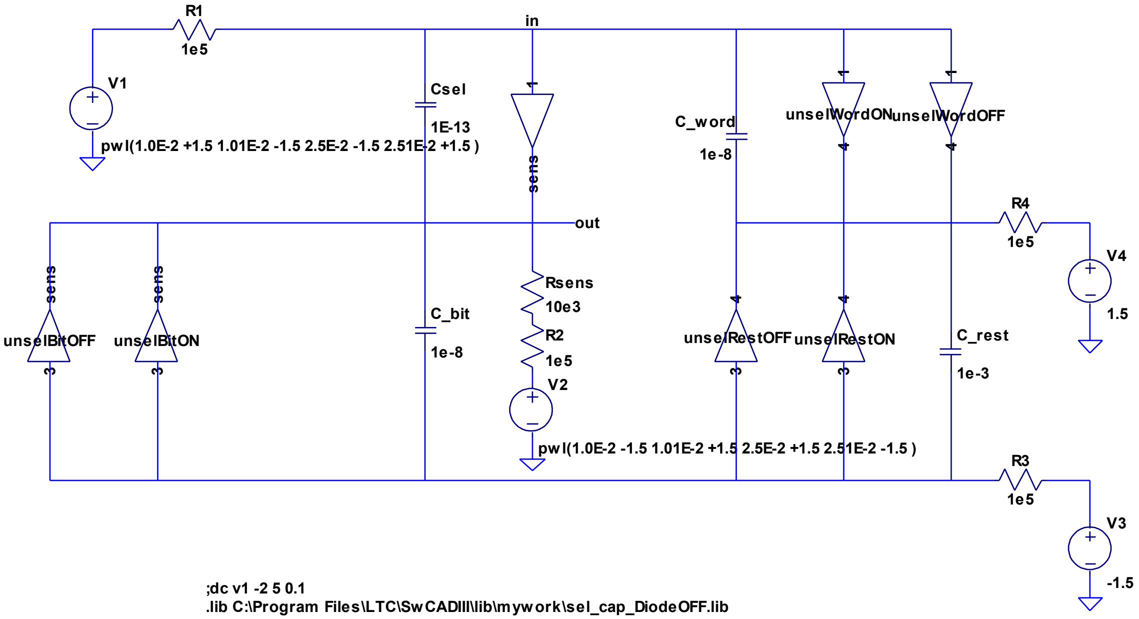
```
.param Rs1  1.66E+8
.param Rp1  3.0E+14
.param idfac1   1.8
.subckt UnselBitDiodeOFF v3 vsens n1=5e4
Bdieq v3 dout I = Is1 * n1 * (exp( (V(v3)-V(dout))/(idfac1*0.025) ) -1)
Rpar  v3 dout {Rp1/n1}
Rserial dout vsens {Rs1/n1}
.ends UnselBitDiodeOFF

.param Is2  1.0E-17
.param Rs2  1.66E+6
.param Rp2  3.0E+14
.param idfac2   1.8
.subckt UnselBitDiodeON v3 vsens n2=5e4
Bdieq v3 dout I = Is2 * n2 * (exp( (V(v3)-V(dout))/(idfac2*0.025) ) -1)
Rpar  v3 dout {Rp2/n2}
Rserial dout vsens {Rs2/n2}
.ends UnselBitDiodeON
```

Subcircuit corresponding to unselected bit / word lines diodes (ON / OFF): N^2 junctions

```
.param Is5  1.0E-17
.param Rs5  1.66E+6
.param Rp5  3.0E+14
.param idfac5   1.8
.subckt UnselRestDiodeOFF v3 v4 n5=5E9
Bdieq v1 dout I = Is5 * n5 * (exp( (V(v1)-V(v4))/(idfac5*0.025) ) -1)
Rpar  v1 dout {Rp5/n5}
Rserial dout v4 {Rs5/n5}
.ends UnselRestDiodeOFF

.param Is6  1.0E-17
.param Rs6  1.66E+6
.param Rp6  3.0E+14
.param idfac6   1.8
.subckt UnselRestDiodeON v3 v4 n6=5E9
Bdieq v3 dout I = Is6 * n6 * (exp( (V(v3)-V(dout))/(idfac6*0.025) ) -1)
Rpar  v3 dout {Rp6/n6}
Rserial dout v4 {Rs6/n6}
.ends UnselRestDiodeON
```



```
;dc v1 -2 5 0.1
.lib C:\Program Files\LTC\SwCADIII\lib\mywork\sel_cap_DiodeOFF.lib
.lib C:\Program Files\LTC\SwCADIII\lib\mywork\uns_cap_BitDiodeOFF.lib
.lib C:\Program Files\LTC\SwCADIII\lib\mywork\uns_cap_BitDiodeON.lib
.lib C:\Program Files\LTC\SwCADIII\lib\mywork\uns_cap_WordDiodeOFF.lib
.lib C:\Program Files\LTC\SwCADIII\lib\mywork\uns_cap_WordDiodeON.lib
.lib C:\Program Files\LTC\SwCADIII\lib\mywork\uns_cap_RestDiodeOFF.lib
.lib C:\Program Files\LTC\SwCADIII\lib\mywork\uns_cap_RestDiodeON.lib
.tran 20m
```

Annex IV 1 light beam generation over 1 pn junction

```
File {
  Grid= "pn2D_msh.grd"
  Doping= "pn2D_msh.dat"
  Plot= "pn2D_des.dat"
  Current= "pn2D_des.plt"
  Output= "pn2D_des.log"
}
Electrode {
  { Name="Pcontact"    Voltage=0.0
                        Voltage = (    0 at 0,
                 0 at 1e-11,
                        -1 at 1.00001e-11,
                        -1 at 10e-11        )
  }
  { Name="Ncontact"    Voltage=0.0 }
}

Physics {
  Mobility( DopingDep HighFieldsat Enormal )
  EffectiveIntrinsicDensity( OldSlotboom )
  OptBeam(
WavePower = 0.01
WaveLength=0.9e-4
SemAbs(model=ODB)
Semsurf = 0
SemWindow = (0.0 2e-4)
WaveTime= (10e-9 15e-9)
WaveTsigma= 1e-9
  )
}

Plot {
  eDensity  hDensity  eCurrent  hCurrent
  Potential SpaceCharge  ElectricField
  eMobility  hMobility eVelocity  hVelocity
  Doping  DonorConcentration   AcceptorConcentration
}
Math {
  Extrapolate
  RelErrControl
  Iterations = 15
  NewDiscretization
}
Solve {
  Poisson
  Coupled { Poisson Electron Hole }
}
```

Annex V 2 light beam generation over 2 pn junction

Device definition: material & contacts "2pn2D.bnd"

```
Silicon "Region 0" { rectangle [ (0,-2) (10,2) ] }
Contact "Pcontact" { line [ (0,-2) (0,2) ] }
Contact "Ncontact" { line [ (20,-2) (20,2) ] }
```

Definition of the regions, boundaries, material types "2pn2D.cmd"

```
Definitions {
  #Refinement regions
  Refinement "Default Region"{
  }
  Refinement "Silicon"{
         MaxElementSize = (0.1 0.1)
         MinElementSize = (0.1 0.1)
  }
  #Profiles
  Constant "P-type"{
          Species = "BoronActiveConcentration"
          Value = 1e+18
  }
  Constant "N-type"{
          Species = "PhosphorusActiveConcentration"
          Value = 1e+16
  }
}
Placements {
  #Refinement regions
  Refinement "Default Region"{
          Reference = "Default Region"
          #Default region
          }
  Refinement "Silicon"{
          Reference = "Silicon"
          RefineWindow = rectangle [( 0 -2 ) , ( 20 2 )]
  }
          #Profiles
          Constant "P1-type"{
          Reference = "P-type"
          EvaluateWindow{
                  Element = rectangle [( 0 -2 ) , ( 5 2 )]
                  DecayLength = 0
          }
          }
          Constant "N1-type"{
          Reference = "N-type"
          EvaluateWindow{
                  Element = rectangle [( 5 -2 ) , ( 10 2 )]
                  DecayLength = 0
          }
          }
          Constant "P2-type"{
          Reference = "P-type"
          EvaluateWindow{
                  Element = rectangle [( 10 -2 ) , ( 15 2 )]
                  DecayLength = 0
          }
          }
          Constant "N2-type"{
          Reference = "N-type"
          EvaluateWindow{
                  Element = rectangle [( 15 -2 ) , ( 20 2 )]
                  DecayLength = 0
          }
          }
}
```

100% overlapping of light pulses:

Main file for Dessis: “2pn2D_des.cmd” static

```
File {
  Grid= "2pn2D_msh.grd"
  Doping= "2pn2D_msh.dat"
  Plot= "2pn2D_des.dat"
  Current= "2pn2D_des.plt"
  Output= "2pn2D_des.log"
}
Electrode {
  { Name="Pcontact"    Voltage=0.0
          Voltage = (    0 at 0,
                         0 at 1e-11,
                         -1 at 1.00001e-11,
                         -1 at 10e-11 )
  }
  { Name="Ncontact"    Voltage=0.0 }
}
Physics {
  Mobility( DopingDep HighFieldsat Enormal )
  EffectiveIntrinsicDensity( OldSlotboom )
  OptBeam
  (
          WavePower = 0.01
          WaveLength=0.9e-6
          Semsurf = 0
          SemWindow = (0 2e-4)
          WaveTime= (10e-9 12e-9)
          WaveTsigma= 1e-9
  )
  (
          WavePower = 0.01
          WaveLength=0.9e-6
          Semsurf = 0
          SemWindow = (10e-4 12e-4)
          WaveTime= (10e-9 12e-9)
          WaveTsigma= 1e-9
  )
}

Plot {
  eDensity  hDensity  eCurrent  hCurrent
  Potential SpaceCharge  ElectricField
  eMobility  hMobility eVelocity  hVelocity
  Doping  DonorConcentration   AcceptorConcentration
}
Math {
  Extrapolate
  RelErrControl
  Iterations = 15
  NewDiscretization
}
Solve {
  Poisson
  Coupled { Poisson Electron Hole }
  Transient (
          InitialTime=0.0    FinalTime=1e-07    InitialStep=0.5-10    MaxStep=1.0e-10
Minstep=1.0e-12                Increment=1.5
  )
  { Coupled {
          Poisson Electron Hole }
  }
}
```

Annex VI Silicon PD

```
Definitions{
  #Refinement regions
  Refinement "Default Region"{
  }
  Refinement "Silicon"{
          MaxElementSize = (0.1 0.1)
          MinElementSize = (0.05 0.05)
  }
  #Profiles
  Constant "P-type"{
          Species = "BoronActiveConcentration"
          Value = 1e+19
  }
  Constant "N1-type"{
          Species = "PhosphorusActiveConcentration"
          Value = 1e+18
  }
  Constant "N2-type"{
          Species = "PhosphorusActiveConcentration"
          Value = 1e+19
  }
}
Placements {
  #Refinement regions
  Refinement "Default Region"{
          Reference = "Default Region"
          #Default region
          }
  Refinement "Silicon"{
          Reference = "Silicon"
          RefineWindow = rectangle [( 0,0 ) , ( 4,0.8 )]
  }
  #Profiles
          Constant "P-type"{
          Reference = "P-type"
          EvaluateWindow{
                  Element = rectangle [( 0.2,0.7 ) , ( 3.8,0.8 )]
          }
          }
          Constant "N1-type"{
          Reference = "N1-type"
          EvaluateWindow{
                  Element = rectangle [( 0,0.01 ) , ( 0.2,0.8 )]
          }
          }
          Constant "N2-type"{
          Reference = "N1-type"
          EvaluateWindow{
                  Element = rectangle [( 0.2,0.01 ) , ( 3.8,0.7 )]
          }
          }
          Constant "N3-type"{
          Reference = "N1-type"
                  EvaluateWindow{
                  Element = rectangle [( 3.8,0.01 ) , ( 4,0.8 )]
          }
          }
  Constant "N4-type"{
          Reference = "N2-type"
          EvaluateWindow{
                  Element = rectangle [(0,0) , (4,0.01)]
          }
  }
}
```

Annex VII Silicon PD: no light effect

STATIC

```
File {
  Grid= "SiPD_msh.grd"
  Doping= "SiPD_msh.dat"
  Plot= "SiPD_des.dat"
  Current= "SiPD_des.plt"
  Output= "SiPD_des.log"
}

Electrode {
  { Name="Pcontact"    Voltage=0.0 }
  { Name="Ncontact"    Voltage=0.0 }
}

Physics {
  Mobility( DopingDep HighFieldsat Enormal )
  EffectiveIntrinsicDensity( OldSlotboom )
}

Plot {
  eDensity  hDensity  eCurrent  hCurrent
  Potential SpaceCharge  ElectricField
  eMobility  hMobility eVelocity  hVelocity
  Doping  DonorConcentration   AcceptorConcentration
}
Math {
  Extrapolate
  RelErrControl
  Iterations = 30
  NewDiscretization
}
Solve {
  Poisson
  Coupled { Poisson Electron Hole }
  Quasistationary ( MaxStep = 0.1  Goal { Name = "Pcontact" Voltage = 1.5 } )
  { Coupled { Poisson Electron Hole } }
}
```

TRANSIENT

```
File {
  Grid= "SiPD_msh.grd"
  Doping= "SiPD_msh.dat"
  Plot= "SiPD_des.dat"
  Current= "SiPD_des.plt"
  Output= "SiPD_des.log"
}

Electrode {
  { Name="Pcontact"    Voltage=0.0
         Voltage = ( 0 at 0,
                        0 at 1e-11,
                        1 at 1.00001e-11,
                        1 at 10e-11 )
  }
  { Name="Ncontact"    Voltage=0.0 }
}

Physics {
  Mobility( DopingDep HighFieldsat Enormal )
  EffectiveIntrinsicDensity( OldSlotboom )
}

Plot {
  eDensity  hDensity  eCurrent  hCurrent
  Potential SpaceCharge  ElectricField
  eMobility  hMobility eVelocity  hVelocity
  Doping  DonorConcentration   AcceptorConcentration
}
Math {
  Extrapolate
  RelErrControl
  Iterations = 30
  NewDiscretization
}
Solve {
  Poisson
  Coupled { Poisson Electron Hole }
  Transient (
         InitialTime = 0.0  FinalTime = 80e-11  InitialStep = 0.5e-12   MaxStep =
1.0e-12
         Minstep = 1.0e-14  Increment = 2 Decrement = 4
  )
  { Coupled {
         Poisson Electron Hole }
  }
}
```

Annex VIII Silicon PD: light effect

```
File {
  Grid= "SiPD2_msh.grd"
  Doping= "SiPD2_msh.dat"
  Plot= "SiPD2_des.dat"
  Current= "SiPD2_des.plt"
  Output= "SiPD2_des.log"
}

Electrode {
  { Name="Pcontact"    Voltage=-1.0 }
  { Name="Ncontact"    Voltage=0.0 }
}

Physics {
  Mobility( DopingDep HighFieldsat Enormal )
  EffectiveIntrinsicDensity( OldSlotboom )
  OptBeam
         (       WavePower = 100
                 WaveLength=0.5e-4
                 SemAbs(model=ODB)
                 Semsurf = 0
                 SemWindow = (1.5e-4 2.5e-4)
                 WaveTime= (1e-10 6e-10)
                 WaveTsigma= 0.01e-10
         )
}

Plot {
  eDensity  hDensity  eCurrent  hCurrent
  Potential SpaceCharge  ElectricField
  eMobility  hMobility eVelocity  hVelocity
  Doping  DonorConcentration   AcceptorConcentration
}
Math {
  Extrapolate
  RelErrControl
  Iterations = 30
  NewDiscretization
}
Solve {
  Poisson
  Coupled { Poisson Electron Hole }
  Transient (
         InitialTime = 0.0   FinalTime = 2e-9   InitialStep = 0.5e-12    MaxStep =
1.0e-12  Minstep = 1.0e-14  Increment = 2 Decrement = 4
         Plot { Range = (0 7e-10) Intervals=2 }
  )
  { Coupled {
         Poisson Electron Hole }
  }
}
```

Annex IX Inhomogeneous Silicon PD

.bnd file

```
Silicon "Body" { rectangle [ (0,0) (4,0.8) ] }
Contact "Pcontact" { line [ (1.8,0.8) (2.2,0.8) ] }
Contact "Ncontact" { line [ (0,0) (4,0) ] }
```

.cmd file

```
Definitions {
  #Refinement regions
  Refinement "Default Region"{
  }
  Refinement "Silicon"{
         MaxElementSize = (0.1 0.1)
         MinElementSize = (0.05 0.05)
  }
  #Profiles
  Constant "P-type"{
         Species = "BoronActiveConcentration"
         Value = 1e+19
  }
  Constant "N1-type"{
         Species = "PhosphorusActiveConcentration"
         Value = 1e+18
  }
  Constant "N2-type"{
         Species = "PhosphorusActiveConcentration"
         Value = 1e+19
  }
  Constant "doping11"{
         Species = "BoronActiveConcentration"
         Value = 1e+16
  }
  Constant "doping12"{
         Species = "PhosphorusActiveConcentration"
         Value= 1e+16
  }
}
Placements {
  #Refinement regions
  Refinement "Default Region"{
         Reference = "Default Region"
         #Default region
         }
  Refinement "Silicon"{
         Reference = "Silicon"
         RefineWindow = rectangle [( 0,0 ) , ( 4,0.8 )]
  }
  #Profiles
         Constant "P1-type"{
         Reference = "P-type"
         EvaluateWindow{
                Element = rectangle [( 0.2,0.7 ) , ( 0.8,0.8 )]
         }
         }
  Constant "P2-type"{
         Reference = "P-type"
         EvaluateWindow{
                Element = rectangle [( 1,0.7 ) , ( 1.7,0.8 )]
         }
         }
  Constant "P3-type"{
         Reference = "P-type"
         EvaluateWindow{
                Element = rectangle [( 2.3,0.7 ) , ( 3.0,0.8 )]
         }
         }
  Constant "P4-type"{
         Reference = "P-type"
         EvaluateWindow{
                Element = rectangle [( 3.2,0.7 ) , ( 3.8,0.8 )]
         }
         }
         Constant "dop11"{
         Reference = "doping11"
         EvaluateWindow{
                Element = rectangle [(0.8,0.7 ) , (1,0.8 )]
         }
         }
         Constant "dop21"{
         Reference = "doping11"
         EvaluateWindow{
                Element = rectangle [(1.7,0.7 ) , (2.3,0.8 )]
         }
         }
         Constant "dop31"{
         Reference = "doping11"
         EvaluateWindow{
                Element = rectangle [(3,0.7 ) , (3.2,0.8 )]
         }
         }
  Constant "N1-type"{
         Reference = "N1-type"
```

```
        EvaluateWindow{
                Element = rectangle [( 0,0.01 ) , ( 0.2,0.8 )]
        }
        }
        Constant "N2-type"{
        Reference = "N1-type"
        EvaluateWindow{
                Element = rectangle [( 0.2,0.01 ) , (0.8,0.7 )]
        }
        }
        Constant "N3-type"{
        Reference = "N1-type"
        EvaluateWindow{
                Element = rectangle [( 0.8,0.01 ) , (1,0.4 )]
        }
        }
        Constant "N4-type"{
        Reference = "N1-type"
        EvaluateWindow{
                Element = rectangle [( 1,0.01 ) , (1.7,0.7 )]
        }
        }
        Constant "N5-type"{
        Reference = "N1-type"
        EvaluateWindow{
                Element = rectangle [( 1.7,0.01 ) , (2.3,0.4 )]
        }
        }
        Constant "N6-type"{
        Reference = "N1-type"
        EvaluateWindow{
                Element = rectangle [( 2.3,0.01 ) , ( 3,0.7 )]
        }
        }
Constant "N7-type"{
        Reference = "N1-type"
        EvaluateWindow{
                Element = rectangle [( 3,0.01 ) , ( 3.2,0.4 )]
        }
        }
        Constant "N8-type"{
        Reference = "N1-type"
        EvaluateWindow{
                Element = rectangle [( 3.2,0.01 ) , ( 3.8,0.7 )]
        }
        }
Constant "N4-type"{
        Reference = "N1-type"
        EvaluateWindow{
                Element = rectangle [( 3.8,0.01 ) , ( 4,0.8 )]
        }
        }
        Constant "dop12"{
        Reference = "doping12"
        EvaluateWindow{
                Element = rectangle [( 0.8,0.4 ) , ( 1,0.7 )]
        }
        }
        Constant "dop22"{
        Reference = "doping12"
        EvaluateWindow{
                Element = rectangle [(1.7,0.4 ) , (2.3,0.7 )]
        }
        }
        Constant "dop32"{
        Reference = "doping12"
        EvaluateWindow{
                Element = rectangle [(3,0.4 ) , (3.2,0.7 )]
        }
        }

Constant "Nplus-type"{
        Reference = "N2-type"
        EvaluateWindow{
                Element = rectangle [(0,0) , (4,0.01)]
        }
}
```

```
}
```

Annex X Approach to UNIQUE

.cmd file

```
Definitions {
  #Refinement regions
  Refinement "Default Region"{
  }
  Refinement "Silicon"{
          MaxElementSize = (0.1 0.1)
          MinElementSize = (0.05 0.05)
  }
  #Profiles
  Constant "P-type"{
          Species = "BoronActiveConcentration"
          Value = 1e+19
  }
  Constant "N1-type"{
          Species = "PhosphorusActiveConcentration"
          Value = 1e+18
  }
  Constant "N2-type"{
          Species = "PhosphorusActiveConcentration"
          Value = 1e+19
  }
  Constant "doping11"{
          Species = "BoronActiveConcentration"
          Value = 1e+16
  }
  Constant "doping12"{
          Species = "PhosphorusActiveConcentration"
          Value= 1e+16
  }
}
Placements {
  #Refinement regions
  Refinement "Default Region"{
          Reference = "Default Region"
          #Default region
          }
  Refinement "Silicon"{
          Reference = "Silicon"
          RefineWindow = rectangle [( 0,0 ) , ( 4,0.8 )]
  }
  #Profiles
          Constant "P1-type"{
          Reference = "P-type"
          EvaluateWindow{
                  Element = rectangle [( 0.2,0.7 ) , ( 0.5,0.8 )]
          }
          }
  Constant "P2-type"{
          Reference = "P-type"
          EvaluateWindow{
                  Element = rectangle [( 0.7,0.7 ) , ( 1,0.8 )]
          }
          }
  Constant "P3-type"{
          Reference = "P-type"
          EvaluateWindow{
                  Element = rectangle [( 1.6,0.7 ) , ( 3.0,0.8 )]
          }
          }
  Constant "P4-type"{
          Reference = "P-type"
          EvaluateWindow{
                  Element = rectangle [( 3.2,0.7 ) , ( 3.8,0.8 )]
          }
          }
```

```
        Constant "dop11"{
        Reference = "doping11"
        EvaluateWindow{
                Element = rectangle [(0.5,0.7 ) , (0.7,0.8 )]
        }
        }
        Constant "dop21"{
        Reference = "doping11"
        EvaluateWindow{
                Element = rectangle [(1,0.7 ) , (1.6,0.8 )]
        }
        }
        Constant "dop31"{
        Reference = "doping11"
        EvaluateWindow{
                Element = rectangle [(3,0.7 ) , (3.2,0.8 )]
        }
        }
Constant "N1-type"{
        Reference = "N1-type"
        EvaluateWindow{
                Element = rectangle [( 0,0.01 ) , ( 0.2,0.8 )]
        }
        }
        Constant "N2-type"{
        Reference = "N1-type"
        EvaluateWindow{
                Element = rectangle [( 0.2,0.01 ) , (0.5,0.7 )]
        }
        }
        Constant "N3-type"{
        Reference = "N1-type"
        EvaluateWindow{
                Element = rectangle [( 0.5,0.01 ) , (0.7,0.4 )]
        }
        }
        Constant "N4-type"{
        Reference = "N1-type"
        EvaluateWindow{
                Element = rectangle [( 0.7,0.01 ) , (1,0.7 )]
        }
        }
        Constant "N5-type"{
        Reference = "N1-type"
        EvaluateWindow{
                Element = rectangle [( 1,0.01 ) , (1.6,0.4 )]
        }
        }
        Constant "N6-type"{
        Reference = "N1-type"
        EvaluateWindow{
                Element = rectangle [( 1.6,0.01 ) , ( 3,0.7 )]
        }
        }
Constant "N7-type"{
        Reference = "N1-type"
        EvaluateWindow{
                Element = rectangle [( 3,0.01 ) , ( 3.2,0.4 )]
        }
        }
        Constant "N8-type"{
        Reference = "N1-type"
        EvaluateWindow{
                Element = rectangle [( 3.2,0.01 ) , ( 3.8,0.7 )]
        }
        }
Constant "N4-type"{
        Reference = "N1-type"
        EvaluateWindow{
                Element = rectangle [( 3.8,0.01 ) , ( 4,0.8 )]
        }
        }
        Constant "dop12"{
        Reference = "doping12"
        EvaluateWindow{
                Element = rectangle [( 0.5,0.4 ) , ( 0.7,0.7 )]
        }
```

```
         }
         Constant "dop22"{
         Reference = "doping12"
         EvaluateWindow{
                Element = rectangle [(1,0.4 ) , (1.6,0.7 )]
         }
         }
         Constant "dop32"{
         Reference = "doping12"
         EvaluateWindow{
                Element = rectangle [(3,0.4 ) , (3.2,0.7 )]
         }
         }

  Constant "Nplus-type"{
         Reference = "N2-type"
         EvaluateWindow{
                Element = rectangle [(0,0) , (4,0.01)]
         }
  }
}
```