DETECTING SUBTHRESHOLD SIGNALS IN FIELD EFFECT TRANSISTORS BY USING STOCHASTIC RESONANCE PHENOMENON

AHMET ÇAĞRI ARLI

AUGUST, 2015

DETECTING SUBTHRESHOLD SIGNALS IN FIELD EFFECT TRANSISTORS BY USING STOCHASTIC RESONANCE PHENOMENON

A THESIS SUBMITTED TO
THE GRADUATE SCHOOL OF NATURAL AND APPLIED SCIENCES OF
CANKAYA UNIVERSITY

BY

AHMET ÇAĞRI ARLI

IN PARTIAL FULLFILLMENT OF THE REQUIREMENTS
FOR
THE DEGREE OF MASTER OF SCIENCE
IN
THE DEPARTMENT OF
ELECTRONICS AND COMMUNICATION ENGINEERING

AUGUST, 2015

Title of the Thesis : **Detecting of Subthreshold Signals in Field Effect Transistors by Using Stochastic Resonance Phenomenon**

Submitted by **Ahmet Çağrı ARLI**

Approval of the Graduate School of Natural and Applied Sciences, Çankaya University

Prof. Dr. Halil T. EYYUBOĞLU

Director

I certify that this thesis satisfies all the requirements as a thesis for the degree of Master of Science.

Prof. Dr. Halil T. EYYUBOĞLU

Head of Department

This is to certify that we have read this thesis and that in our opinion it is fully adequate, in scope and quality, as a thesis for the degree of Master of Science.

O. Gazi

Assoc. Prof. Dr. Orhan GAZİ

Supervisor

Examination Date : 18.08.2015

Examination Committee Members:

Assoc.Prof. Dr. Özgür ERTUĞ (Gazi Univ.)

Assoc.Prof. Dr. Orhan GAZİ (Çankaya Univ.) O. Gazi

Asst. Prof. Dr. Barbaros PREVEZE (Çankaya Univ.)

STATEMENT OF NON-PLAGIARISM PAGE

I hereby declare that all information in this document has been obtained and presented in accordance with academic rules and ethical conduct. I also declare that, as required by these rules and conduct, I have fully cited and referenced all material and results that are not original to this work.

Name, Last Name: Ahmet Çağrı, ARLI

Signature : A. Çagrı A.

Date : 18.08.2015

ABSTRACT

DETECTING SUBTRESHOLD SIGNALS IN FIELD EFFECT TRANSISTORS BY USING STOCHASTIC RESONANCE PHENOMENON

ARLI, Ahmet Çağrı

M.Sc., Department of Electronics and Communication Engineering

Supervisor: Assoc. Prof. Dr. Orhan GAZİ

August 2015, 48 Pages

Until recently it is accepted that noise is always a degrading factor for the performance of electronic communication and signal processing systems. For this reason many of the studies on signal processing and communication systems focused on eliminating the negative effects of noise signal on information bearing signal. However, recently it is discovered that noise can play a positive role for the detection of weak signals for electronic systems employing nonlinear electronic devices. It is seen that the detection of the weak signals in nonlinear systems is possible when an optimum amount of noise is added to the information bearing signal before passing the signal through the nonlinear system. And this concept is named as stochastic resonance, i.e., SR.

In this thesis it is shown that it is possible to develop a training based model for the SR systems that decides optimum noise signal to be added to the weak information signal directly for the best system performance.

In the second part of the thesis study a practical application of the SR concept on a nonlinear electronic device, field effect transistor (FET), is demonstrated. For this purpose a user interface program using C# is developed and some part of the signal processing is done using the field programmable gate arrays (FPGA).

Keywords: Signal Detection, FET, FPGA, Stochastic Resonance

ÖZ

STOKASTİK REZONANS FENOMENİNİN ALAN ETKİLİ TRANSİSTÖRLERDE EŞİK ALTI SİNYALLERİN TESPİT EDİLMESİ İÇİN KULLANIMI

ARLI, Ahmet Çağrı

Yüksek Lisans, Elektronik ve Haberleşme Mühendisliği Anabilim Dalı

Tez Yöneticisi: Doç. Dr. Orhan Gazi

Ağustos 2015, 48 Sayfa

Gürültü şimdiye kadar daima elektronik haberleşme ve sinyal işleme sistemleri için performans düşürücü etmen olarak kabul edilmiştir. Bu sebeple sinyal işleme ve haberleşme sistemleri üzerinde yapılan pek çok çalışma gürültü sinyalinin bilgi taşıyan sinyal üzerindeki negatif etkisini ortadan kaldırmaya odaklanmıştır. Ancak son zamanlarda, gürültünün doğrusal olmayan elektronik aygıtlarda zayıf sinyal algılamasında olumlu etkisi olduğu keşfedilmiştir. Zayıf bilgi sinyallerinin üzerine doğrusal sisteme girmeden önce makul miktarda gürültü eklendiğinde algılanmasının mümkün olduğu görülmüştür. Ve bu kavram stokastik rezonans (SR) olarak isimlendirilmiştir.

Bu tez çalışmasında SR sistemlerinde kullanılmak üzere, en iyi sistem performansına erişebilmek için zayıf veri sinyaline eklenecek olan en makul gürültü sinyaline direkt olarak karar verebilecek olan talim tabanlı modelin mümkün olduğu gösterilmiştir.

Tez çalışmasının ikinci kısmında doğrusal olmayan bir elektronik aygıt olan alan etkili transistör (FET) üzerinde SR kavramını temel alan pratik bir uygulama gösterilmiştir. Bu amaçla C # kullanılarak oluşturulan bir kullanıcı ara yüzü programı ve bazı sinyal işleme kısımlarını gerçekleştiren alanda programlanabilir kapı dizileri (FPGA) kullanılmıştır.

Anahtar Kelimeler: Sinyal Tespiti, FET, FPGA, Stokastik Rezonans

ACKNOWLEDGEMENTS

I would like to express my deepest gratitude to my supervisor Assoc. Prof. Dr. Orhan GAZİ for his guidance and encouragement during my thesis study. I am very grateful for his leading role, advices and insight throughout the research.

I also wish to thank to all my family for their support and encouragement during my studies.

TABLE OF CONTENTS

CHAPTERS:

LIST OF FIGURES

FIGURES

FIGURES

FIGURES

LIST OF ABBREVIATIONS

ADC	Analog to Digital Converter
CAD	Computer Aided Design
CMOS	Complementary Metal-Oxide Semiconductor
DAC	Digital to Analog Converter
DAQ	Data Acquisition Unit
DLL	Dynamic Link Library
FET	Field Effect Transistor
FPGA	Field Programmable Gate Array
GUI	Graphical User Interface
Op-amp	Operational Amplifier
SCI	Serial Communication Interface
SCPI	Commands for Programmable Instruments
SNR	Signal-to-Noise Ratio
SR	Stochastic Resonance
UART	Universal Asynchronous Receiver/transmitter
USB	Universal Serial Bus

CHAPTER 1

1. INTRODUCTION

1.1 Background

Noise is generally accepted as a degrading factor for the performance of electronic systems. Counter measures are taken in order to eliminate the degrading effects of noise. After a couple of decades of study to wipe out the noise, remarkable advancements are achieved that reduces its negative effect. Nonetheless, noise can play a constructive role in signal detection. Stochastic resonance (SR) phenomenon allows noise to play such a beneficial role. SR shows that performance of a system in a noisy environment can be better at some level of noise intensity. In this thesis work, SR phenomenon has been demonstrated with both simulation and experimental studies. Furthermore, training based model is developed to determine the optimum noise variance needed to enhance the system performance.

SR as a helpful phenomenon is introduced at 1981 by Benzi [1] where analysis of geophysical data exhibited that, ice ages were occurred with average periodicity of about 100,000 years. Additionally, an oddity exits in the Earth’s orbit, which causes energy fluctuations on the received energy of earth from sun. This oddity coincides with ice age eras. Benzi proposed that addition to this oddity, small changes in the earth’s climate leads to ice age conditions. Then, the proposed mechanism is named as stochastic resonance ever since. In model they proposed, SR has three elements: weak input signal, noise signal and a bistable nonlinear system [1, 2]. This kind of system can be described by a Langevin equation given as

$$\dot{x}(t) = ax(t) - bx^3(t) + s(t) + n(t) \qquad (1.1)$$

where s(t) is weak input signal, n(t) is zero mean Gaussian noise and x(t) is the output signal. SR is not only the subject of modeling of climate changes but also is valid for nonlinear threshold systems.

SR is proved to be applicable in biology [3-9]. In two main branches of biology, SR finds different applications. One branch is about the explanation of the working mechanism of biological structures. Since, these structures can be described by nonlinear threshold systems, SR can help to explain their functionality. For instance, synaptic transmission [4], developing neuron models to explain brain functions [4], implementation of signal enhancement at semiconductor neuron models [5] can be given as examples. In these studies, biological fluctuations are described as internal noise, and this kind of noise is inserted into model formulas in order to explain working mechanism accurately. Douglas et. al. experimental results on crayfish mechanoreceptors proved that real neurons can be modeled by SR concept [8]. These studies aim in improving biological weak signal detection by the use of external noise. Measuring weak EEG signals [4], improving performance of measurement of EMG signals from the forearm [6], improving paddlefish's prey detection ability [9] can be given as other studies where SR is used.

Improving object and pattern detection performance in image processing is another application area of SR. In this application noise is added to low quality MRI images of brain to achieve improved diagnosis over conventional methods [10]. As a pattern recognition example low quality fingerprint images can be detected by adding optimum amount of noise [11]. Circles as objects of interest in an image are detected by enhancing the image with noise for better detection performance [12].

Electronic devices are also subject to SR studies. Detection of weak input signal will have paramount importance in electronic systems. Experiments on electronics mostly focus on improving performance of electronic devices' base components such as field effect transistors (FET). In addition, SR experiments are in vogue on nanoscale devices like nano-wire FET's [6, 13-14] and CNT-FET's [15-20]. Common operation cycles of these studies are as follows. First noise signal is added to a subthreshold input signal; and generated noisy signal is applied to the transistor's gate terminal. Output performance is calculated between input signal's voltage value and drain to source current value for all noise intensities. When all data are collected and all performance calculations are done, noise intensity versus performance graph is plotted. As a result, bell shape of SR is observed which is similar to Fig. 1. As these studies have common operation cycles, they also have common results which show that SR seems to be useful for weak signal detection and information transmission. Other related studies in electronics can be given as: threshold reduction in a Schmitt trigger [21], IR sensor performance boosting by using noise [21], Signal-to-Noise Ratio (SNR) improvement in Complementary Metal-Oxide Semiconductor (CMOS) inverters [22].

A subthreshold input signal may have no effect on a nonlinear threshold system. By the addition of weak noise signal the subthreshold signal can exceed the threshold value and becomes observable. Amount of noise that enhances the input signal is vital in this process. In order to observe SR effect, noise signal with intensities from 0 to some x value are mounted on the subthreshold input signal before passing through the threshold part. At each trial, output performance is calculated. When noise intensity versus output performance graph is plotted, characteristic SR curve is observed. Fig. 1 shows such a curve. A valid SR curve has concave shape. Consequently, benefit of noise in signal detection becomes a fact for specific set of noise values. Notice that if there is no noise signal introduced to this kind of system, output performance will be zero since input signal is under threshold level.

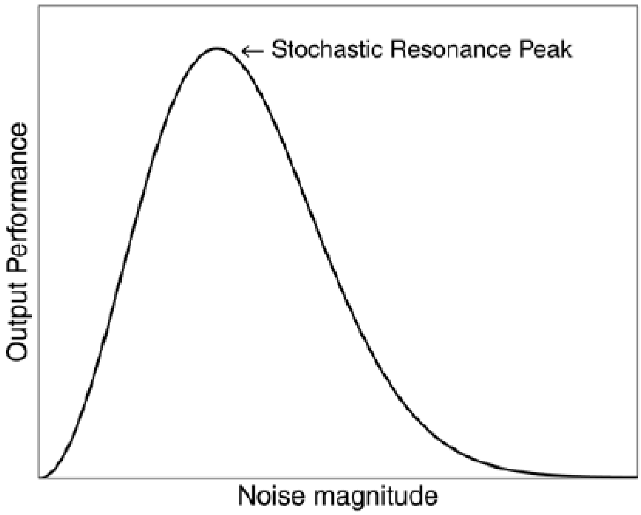


Figure 1 Characteristic SR curve [3]

1.2 Objectives

In the simulation part of this thesis, first we do system training; then the trained system is used to estimate optimum noise intensity that is needed for the best output performance for any weak signal. Estimation is going to be done by using a set of equations. Inputs of these equations are subthreshold signals with different variances and the threshold of the nonlinear system, and output of these equations is the variance of noise which produces maximum correlation value between system output and system input. MATLAB is used as the simulation platform.

In the experiment part of this thesis, SR effect on a conventional FET is going to be observed. To be able to accomplish such a task, firstly, experimental tools and results are examined from previous studies. Taking into account of pervious researcher's experiences are important in order to have successful operation cycles. Therefore, experimental studies on FETs are analyzed in terms of figuring out FET characteristics, input and noise signal types, tools that are used to generate them, tools to acquire analog data and conversion to the digital data and system performance measurement techniques. After analysis, best suited items are chosen among all and are going to be explained in detail in Chapter 3.

As stated in the previous part, SR experiments in electronic devices are focuses on FETs. In order to observe SR effect on such a nonlinear system; threshold value, drain to source current level should be known utterly. I-V characteristics curve can provide such information, thus achieving these characteristics is the first step of operation. Semiconductor parameter analyzers are used to obtain I-V curves of the FETs [15-16, 18-19].

Second stage of operation involves choosing the type of the input signal from the signal set, pulse train [13, 15-17, 23], sinus wave [18], and triangle wave [14]. Frequency of the input signal changes from 10 Hz [17] to 28 KHz [5]. However, the most important property of the input signal is that its amplitude should be below threshold value of the nonlinear system. Zero mean Gaussian noise is used in experimental studies [6,13-20,22] because; it's the closest noise type that nature offers. In addition, other noise distributions like Uniform and Cauchy [15] are proved to be effective in SR systems. Bandwidth of the noise changes from 50 KHz [13] to 26 MHz [17]. Function generators are used to generate input and noise signals and Data Acquisition Units (DAQ) and oscilloscopes are used to collect data.

Last stage of the operation involves some mathematical calculations which shows the SR effect in terms of output performance. Output performance is represented by three different measures: signal-to-noise ratio [2, 22], mutual information [15] and correlation coefficient [13-14, 17, 19-20]. In this thesis work, correlation coefficient is used as measurement unit of output performance. Correlation between two data sets, let say x and y, is calculated by using (1.2) as

$$\mathrm{Corr(x,y)} = \frac{\mathrm{Cov(x,y)}}{\sigma_{\mathrm{x}}\sigma_{\mathrm{y}}} \tag{1.2}$$

where Corr (x,y) represents correlation operation, while Cov (x,y) stands as covariance between two and σ_x and σ_y stands as variance of data sets. Covariance and variance equations are given in (1.3) and (1.4), respectively.

$$\mathrm{Cov}(X,Y) = E[XY] - E[X]E[Y] \tag{1.3}$$

$$\sigma_x = E[X^2] - (E[X])^2 \tag{1.4}$$

As a conclusion, objective of the experimental study is to show that SR phenomena can be used for conventional p-channel FET. Overall operation can be seen in Fig. 2.

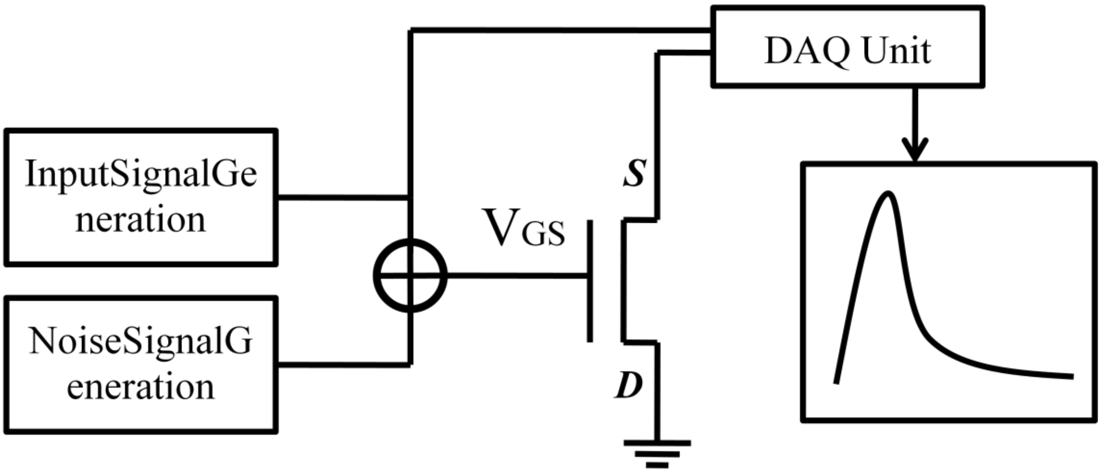


Figure 2 Overall operation

1.3 Organization of the Thesis

This thesis contains four chapters. In Chapter 1 SR is introduced and informs about brief information is give about the application areas of SR. In Chapter 2 training based modeling for SR systems is proposed and it is shown that using the trained model optimum noise variances can be determined for any type of weak input signal. Several simulations on MATLAB are explained and as a result a set of equations are represented. In Chapter 3, experimental study on a FET is explained. In order to achieve SR curve, experimental components and steps are explained in detail. Finally Chapter 4, conclusions are drawn.

CHAPTER 2

2. TRAINING BASED MODELING OF SR SYSTEMS

Aim of this chapter is to find the amount of noise to be added to the subthreshold input signal for the best correlation coefficient value between input and output signals through derived equations. In this way, there won’t be any need to try all noise intensities to find optimal value. Thus, a training based modeling is going to be obtained. In the training operation, by using different input signals and different thresholds we first obtain a set of SR curves. Once we have sufficient SR curves equation derivation operation follows. A set of equations whose inputs are the signal variances and system thresholds and their outputs are the optimum noise variances are obtained. To check whether final equations are realistic or not, classical SR simulation data sets are going to be used.

Training operation contains three stages where the first stage is about obtaining one SR curve for a certain level of input signal with a certain threshold. Second stage of operation involves obtaining several SR curves with respect to different levels of input signals while keeping threshold constant. Last stage covers repeating second stage for different threshold values. Fig. 3 shows the overall operation scheme and stages, and steps for each stage are colored accordingly for better understanding. Green arrows represent first stage’s operation steps, while red ones represent second and the blue ones represent the final stage’s steps. In the following text, stages are going to be explained in subsections.

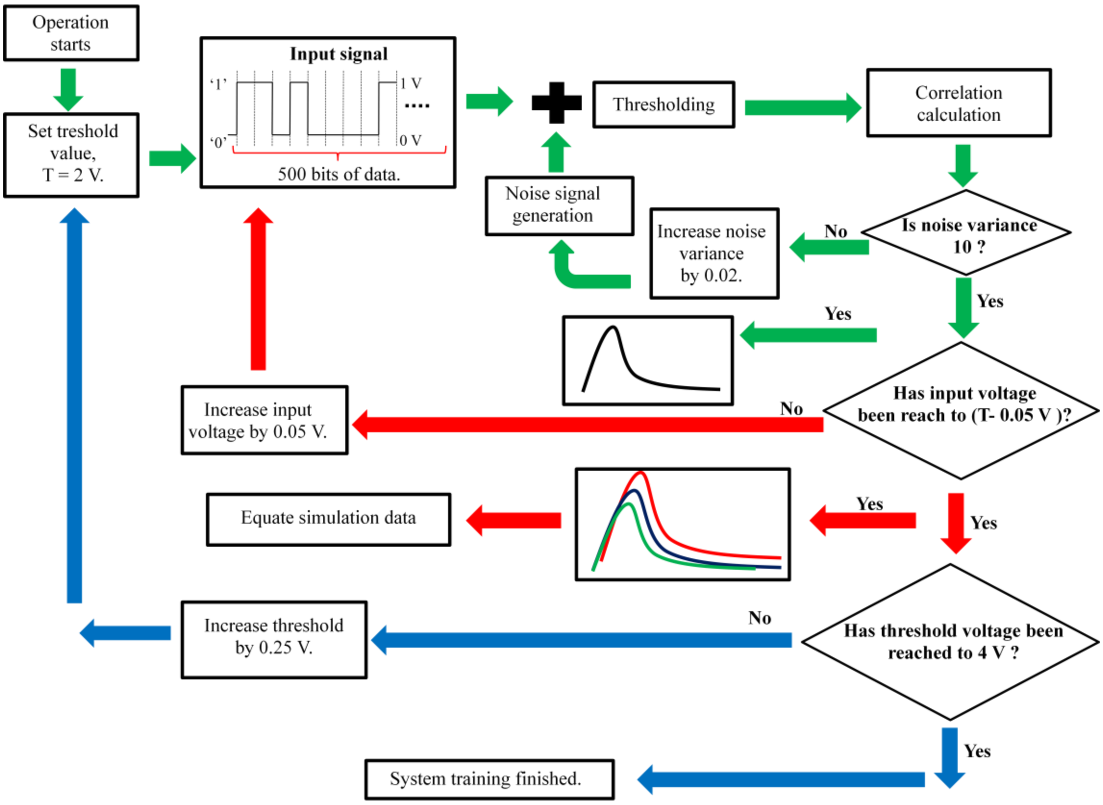

Operation starts
Set treshold value, T = 2 V.
Input signal
'1'
'0'
1 V
0 V
....
500 bits of data.
Thresholding
Correlation calculation
Noise signal generation
Increase noise variance by 0.02.
No
Is noise variance 10 ?
Yes
Yes
Increase input voltage by 0.05 V.
No
Has input voltage been reach to (T- 0.05 V)?
Equate simulation data
Yes
Yes
Increase threshold by 0.25 V.
No
Has threshold voltage been reached to 4 V ?
Yes
System training finished.

Figure 3 System training operation steps

2.1. Obtaining a Single SR Curve

2.1.1. Method of stage 1

A Bernoulli sequence of 500 bits generated randomly is used as input signal. Equation to create one bit of such a sequence is given in (2.1) where $f(x)$equals the input signal and 'r' represents a random number between 0 and 100. Hereafter, same operation is repeated 500 times to render whole sequence.

$$f(x) = \begin{Bmatrix} 1, \ r > 50 \\ 0, elsewhere \end{Bmatrix} \qquad (2.1)$$

At the beginning, logic '1's in the input signal are assigned to the 1 V and logic '0's are assigned to the 0 V. Then, zero mean Gaussian noise in different variance values, starting from 0 to ending at 10 with steps of 0.02, are added to the input signal in order to observe SR. After addition, a predefined threshold is applied as in (2.2) and correlation between input and output calculated using (1.2).

$$g(x) = \begin{cases} 1, f(x) + noise \geq T \\ 0, f(x) + noise < T \end{cases} \tag{2.2}$$

$f(x)$is the input and $g(x)$is the output of the nonlinear threshold system and threshold value (T) has initial value 2 V.

Enhancement of noise signal to the input signal and calculation of correlation coefficients are repeated 100.000 times at each step and averaged to reach more reasonable data. For the computer simulations MATLAB platform is used. Considering 100.000 repetition and 500 bits of input data, obviously 50.000.000 correlation calculations are needed. In this case, “corrcoef” command in MATLAB is observed as time consuming. As an alternative, “MEX” feature of MATLAB is used. This property allows using C/C++ codes as subroutines of main MATLAB code for performance boosting. A third party correlation calculation C file, named as “prcorr2” is used by the help of “MEX” feature. It should be stated that, in this method only c code runs on a C compiler, rest of operations are performed on MATLAB. Consequently, with the help of “MEX” property, approximately 3 times faster operation is accomplished.

2.1.2. Stage-1 result

Resulting correlation values versus noise variance values from 0 to 10 are plotted and characteristic stochastic resonance curve is observed in Fig. 4. After this operation stage 1 is complete. An SR data set for a fixed input signal and fixed threshold value (T = 2V) is achieved.

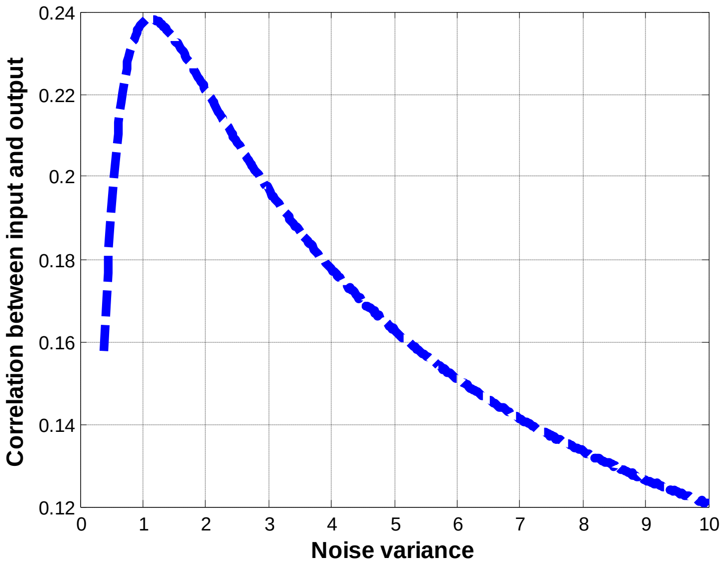


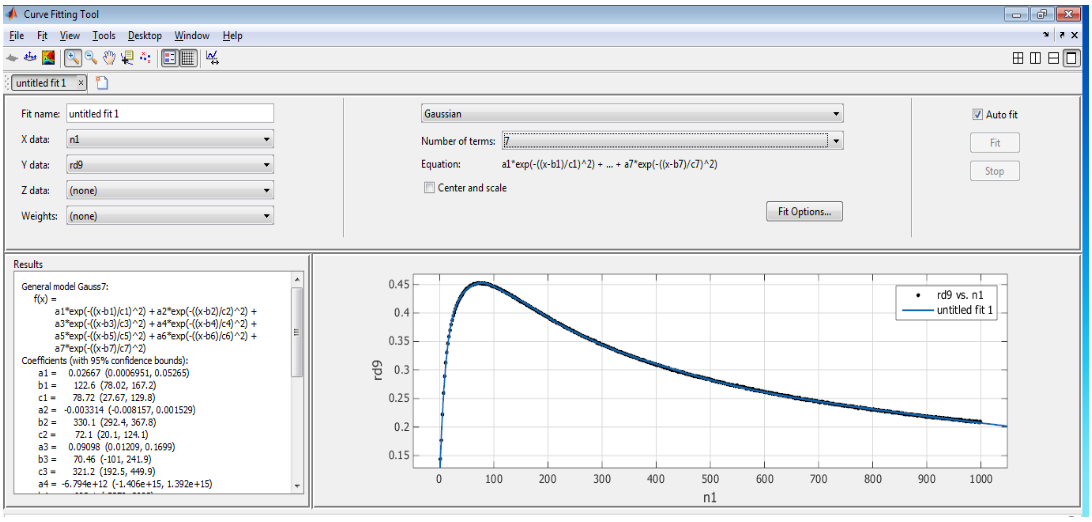
Figure 4 SR curve for the input signal voltage of 1V

2.2. Consecutive SR Curves for a Constant Threshold

2.2.1. Method of Stage-2

Once one SR curve is obtained, second stage starts in order to estimate effect of input signal’s voltage level on SR; nonetheless more data points are needed. For adequate guidance through steps, red arrows in Fig. 3 should be followed. Input signal’s logic ‘1’’s values are incremented with steps of 0.05 V till to threshold voltage, while logic ‘0’s stay at 0 V. In this way, variances of the input signal are increased automatically and one SR curve is achieved at each trial. 1.95 V is the last value for the trials where the threshold value is 2 V. Because, after that point, input signal does not need the noise signal to pass the threshold. Moreover, system performance degrades directly in that case.

We get 20 different SR curves at the end of second stage. Next step in stage two is to derive their mathematical expressions for further use. Matlab’s Curve Fitting Toolbox is used to create equations of SR curves. Curve Fitting Toolbox is a kind of interface that allows users to change the degree or the type of mathematical expressions in order to reach best fitting curve for the discrete data set. Since, polynomial representation results are observed problematic, after some trials on the interface, as the best method to derive expressions from simulation data is selected ‘Gaussian’ with 6, 7 or 8 numbers of terms. Fig. 5 shows the interface and additionally a sample data set and suitable curve for it. It should be stated that when no fitting choice is available for simulation data, that raw data is used directly in calculations.



Fit name: untitled fit 1
X data: n1
Y data: rd9
Z data: (none)
Weights: (none)

Gaussian
Number of terms: 7
Equation: a1*exp(-((x-b1)/c1)^2) + ... + a7*exp(-((x-b7)/c7)^2)
□ Center and scale
☑ Auto fit

```
General model Gauss7:
  f(x) =
       a1*exp(-((x-b1)/c1)^2) + a2*exp(-((x-b2)/c2)^2) +
       a3*exp(-((x-b3)/c3)^2) + a4*exp(-((x-b4)/c4)^2) +
       a5*exp(-((x-b5)/c5)^2) + a6*exp(-((x-b6)/c6)^2) +
       a7*exp(-((x-b7)/c7)^2)
Coefficients (with 95% confidence bounds):
     a1 =    0.02667  (0.0006951, 0.05265)
     b1 =      122.6  (78.02, 167.2)
     c1 =      78.72  (27.67, 129.8)
     a2 =  -0.003314  (-0.008157, 0.001529)
     b2 =      330.1  (292.4, 367.8)
     c2 =       72.1  (20.1, 124.1)
     a3 =    0.09098  (0.01209, 0.1699)
     b3 =      70.46  (-101, 241.9)
     c3 =      321.2  (192.5, 449.9)
     a4 = -6.794e+12  (-1.406e+15, 1.392e+15)
```

Figure 5 Matlab Curve Fitting Tool

The Gaussian Method fits the peak given as

$$y=\sum_{i=1}^{n} a_i e^{\left[-(\frac{x-b_i}{c_i})^2\right]} \tag{2.3}$$

where a_i is the amplitude, b_i represents location, c_i is related to peak width and n is the number of the peaks to fit. Maximum possible value of the n can be 8 in the Curve Fitting Toolbox.

2.2.2. Stage 2 results

At the end of second stage, 20 different mathematical expressions for 20 different voltage levels of subthreshold input signals are obtained. In Fig. 6, these equations are plotted against the noise variance whose values changes from 0 to 10 with steps of 0.02. The line in the lowest position the figure corresponds to the input signal whose logic ‘1’ value equals to 1 V while SR curve at the top belongs to the input signal whose logic ‘1’ value equals to 1.95 V. First observation from Fig. 6 can be stated as follows. As the variance of the subthreshold input signal increases maximum correlation can be achieved via the use of noise signals with smaller variances.

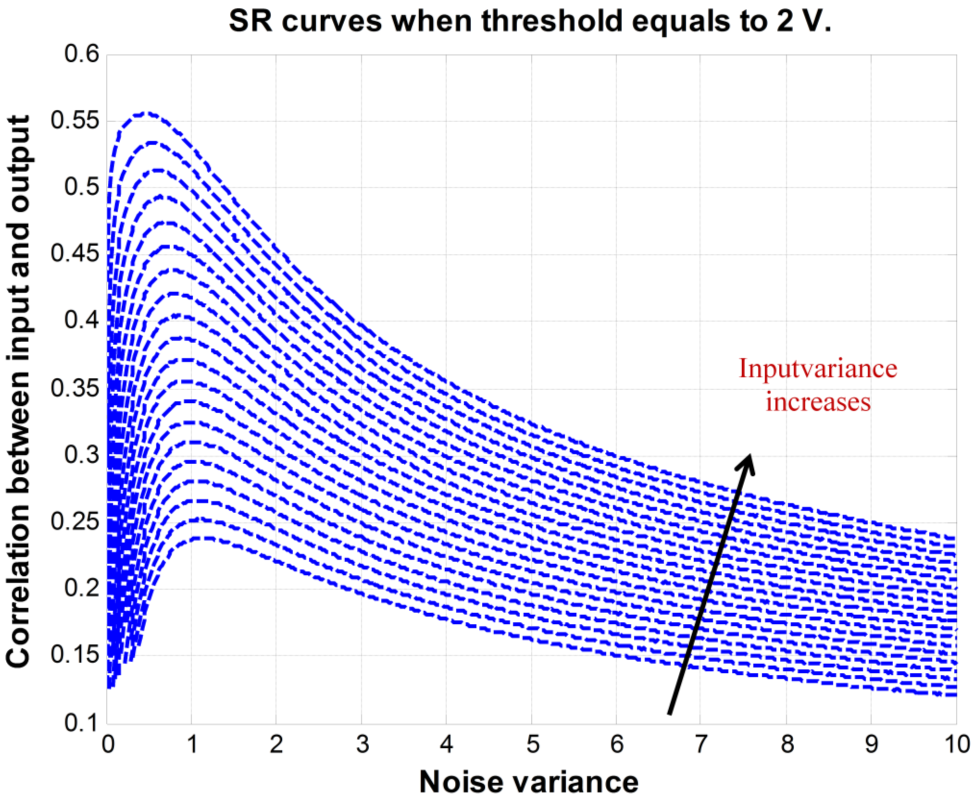


Figure 6 SR curves for input signal with increasing variances

Second observation is about the characteristic of the curves: as the voltage level of the input signal gets closer to the threshold value, reaching to the maximum point of an SR curve occurs faster than the lower voltage levels of input signal. In the Fig. 7, lines for the correlation increment from zero point to maximum correlation value are shown with red color. Maximum correlation values are calculated by using first derivative of the functions. When the first derivative of the function is set to 0 and solved according to x, local extreme points can be obtained. In these functions, there are local maxima points which are equals to maximum correlation values. In order to guess in which noise value the best correlation can be achieved for the certain variance value of input, these two observations and further comparisons are going to be used. It is important to state that, when no fitting choice available for data set, simply “max” function of Matlab is used.

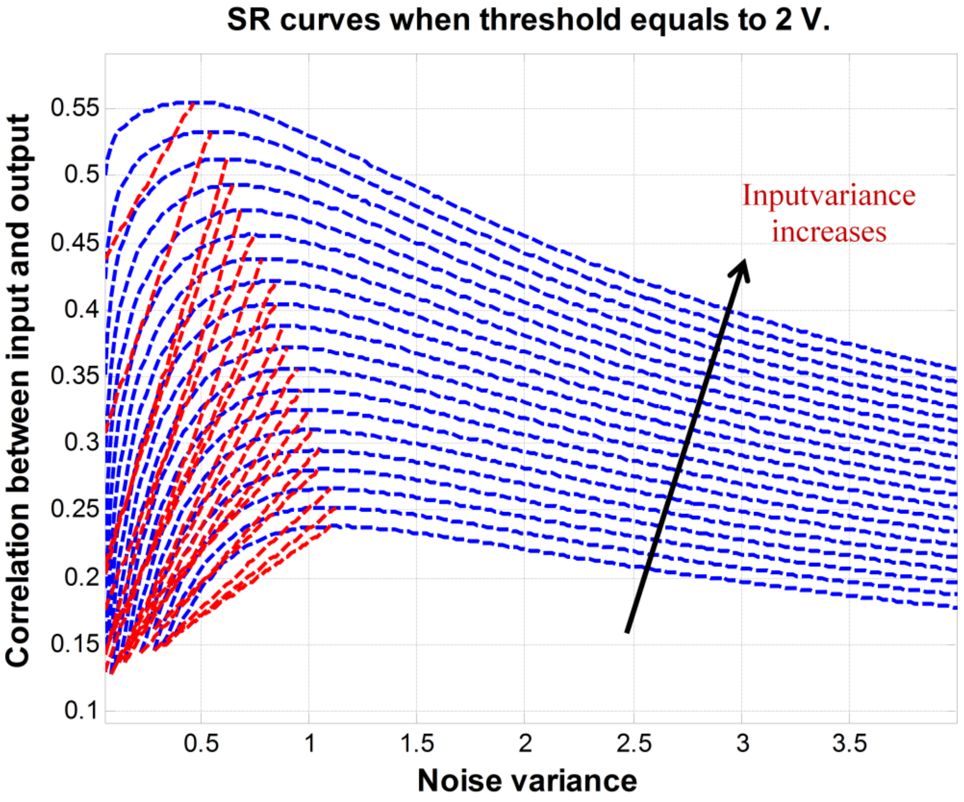


Figure 7 Correlation increment on SR curves

Lines for the correlation increment from zero point to local maxima points for each input signal are plotted separately from the SR curves for better visualization in Fig. 8. As it is seen from Fig. 8 (a), while the input signal’s variance increases, slopes of the lines increase as well. Moreover, in Fig. 8 (b), slopes of the lines versus input variance graph is plotted. The largest slope belongs to the input signal whose logic ‘1’ value equals to 1.90 V and the lowest one belongs to the input whose logic ‘1’ value equals to 1 V. 1.95 V of input signal seems to be unsuitable for the overall situation and isn’t taken into account.

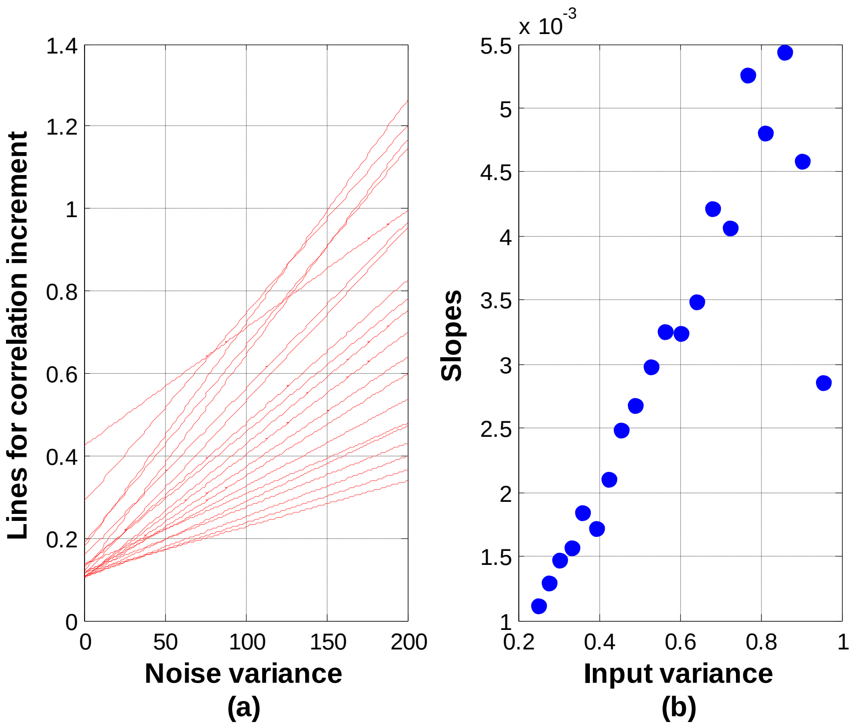


Figure 8 (a)Correlation increments and (b) input variance versus slopes

We will discuss the relation between maximum correlation values of the SR curves, and noise signal and input signal variances. Fig. 8 shows relation between maximum correlation values and noise signal variances.

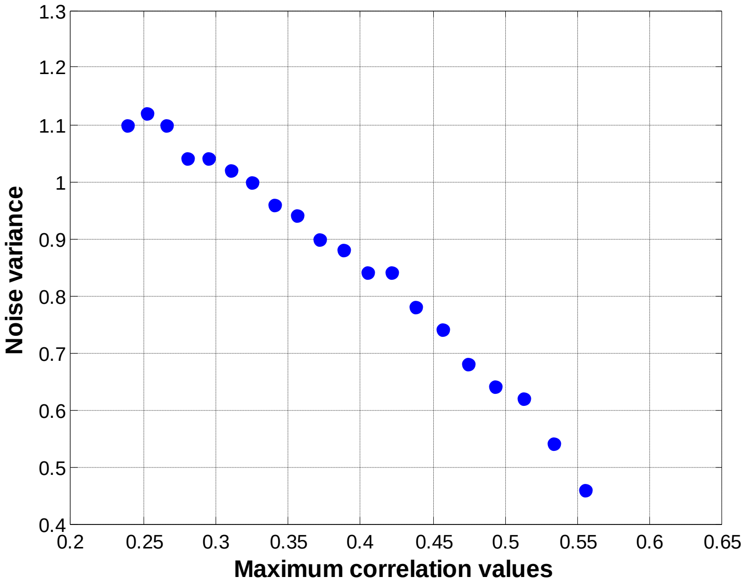


Figure 9 Relation between noise variances and maximum correlation values

In Fig. 9 maximum correlation values versus consecutive input variance values are plotted. Fig. 9 and Fig. 10 show that the relation between these parameters can be represented as polynomial functions.

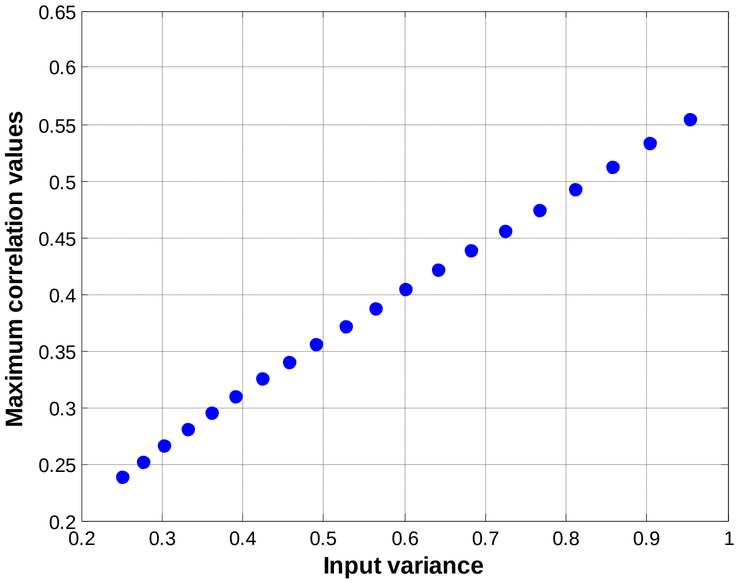


Figure 10 Maximum correlation values versus input variances

Basic fitting property of Matlab is used to find out these polynomial functions. Fig. 11 (a, b) shows the best suited equations for stage 2 of simulation data. After some trials, as the best fitting choice second degree polynomial for Fig. 11 (a) and third degree polynomial for Fig. 11 (b) are selected.

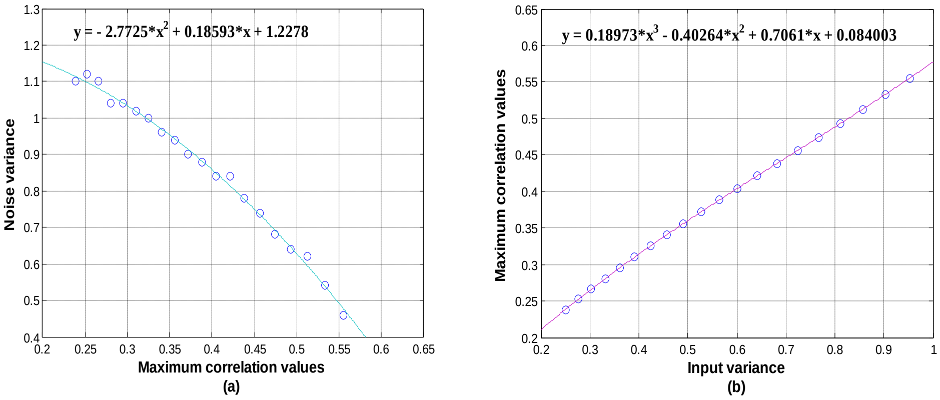


Figure 11 Polynomial fitting solutions between (a) maximum correlation values-noise variance (b) input variance-maximum correlation values

By using "Basic Fitting" property of the Matlab, equations for the Fig. 11 can be written as

$$N.V. = -2.7725 * (M.C.V.)^2 + 0.18593 * (M.C.V.) + 1.2278 \qquad (2.4)$$

$$(M.C.V) = 0.18973 * (I.V.)^3 + 0.40264 * (I.V.)^2 + 0.7061 * (I.V.) + 0.084003 \qquad (2.5)$$

where M.C.V. stands for maximum correlation value, I.V. represents input variance and N.V. is used as noise variance of best output performance. Equations (2.4) and (2.5) are used to find optimum noise variance for an input signal whose variance is known (I.V.) in order to reach the best output performance in the case of constant threshold of T=2V. In the following parts we extent these equations so that similar estimations become possible for different threshold values.

2.3. Consecutive SR Curves for Consecutive Thresholds

2.3.1. Method of stage 3

Aim of the final stage is to insert nonlinear system’s threshold values in to the equations. Thus, more simulations are performed to cover different threshold values. Stage 1 and stage 2 are repeated for different threshold values ranging from 2V to 4V with incremental step size of 0.25 V, and a total of 9 trials have been performed. At each threshold, a different equation similar to (2.4) and (2.5) is obtained. These 18 equations are used to construct a more common formula.

Method to reach a more generalized formula starts with gathering all coefficients of 18 polynomials in an array. At first, M.C.V. - I.V. equation with maximum degree 3 and having 4 coefficients, i.e., (2.5) is considered. Below, a sample matrix is given as

$$\begin{bmatrix} M.C.V._{T=2V} \\ M.C.V._{T=2.25V} \\ M.C.V._{T=2.5V} \\ M.C.V._{T=2.75V} \\ M.C.V._{T=3V} \\ M.C.V._{T=3.25V} \\ M.C.V._{T=3.5V} \\ M.C.V._{T=3.75V} \\ M.C.V._{T=4V} \end{bmatrix} = \begin{bmatrix} I.V.^{3} & I.V.^{2} & I.V.^{1} & I.V.^{0} \\ a_{11} & a_{12} & a_{13} & a_{14} \\ a_{21} & a_{22} & a_{23} & a_{24} \\ a_{31} & a_{32} & a_{33} & a_{34} \\ a_{41} & a_{42} & a_{43} & a_{44} \\ a_{51} & a_{52} & a_{53} & a_{54} \\ a_{61} & a_{62} & a_{63} & a_{64} \\ a_{71} & a_{72} & a_{73} & a_{74} \\ a_{81} & a_{82} & a_{83} & a_{84} \\ a_{91} & a_{92} & a_{93} & a_{94} \end{bmatrix} \qquad (2.6)$$

After this point; each column of the matrix are plotted against threshold values and curve fitting for each one is performed. Fig. 12 shows curve fitting results for each coefficient column of (2.6).

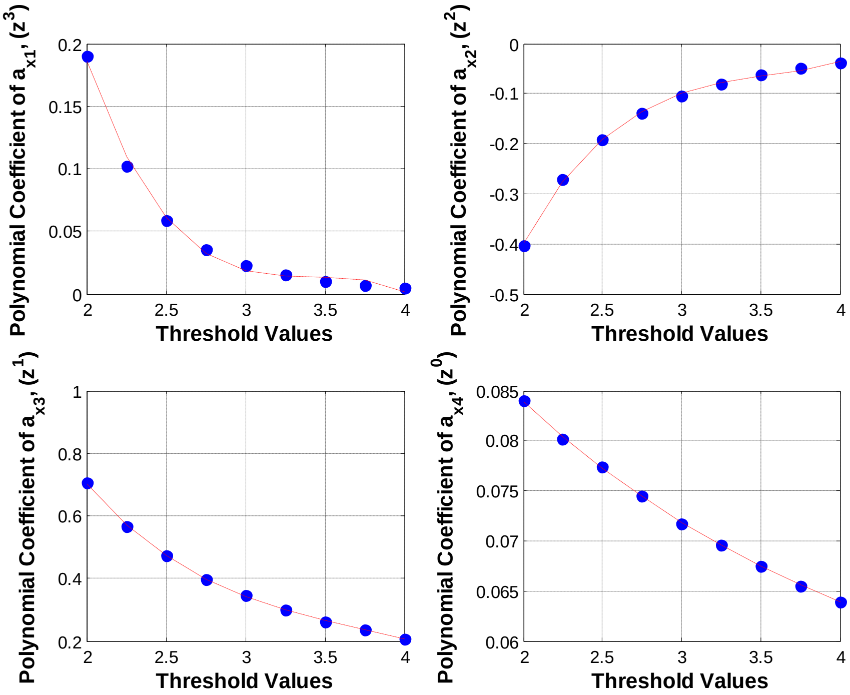


Figure 12 Coefficient fitting for M.C.V.-I.V. equation

As the best fitting polynomial, third degree representation is chosen for each coefficient. Fitting is accomplished by the use of Matlab function named as “polyfit”.

$$cof\_z^{3}=-0.0595*T.V.^{3}+0.6104*T.V.^{2}-2.0893*T.V.+2.3980 \tag{2.7}$$

$$cof\_z^{2}=0.0727*T.V.^{3}-0.7717*T.V.^{2}+2.7768*T.V.-3.4461 \tag{2.8}$$

$$cof\_z^{1}=-0.0559*T.V.^{3}+0.6168*T.V.^{2}-2.3843*T.V.+3.4507 \tag{2.9}$$

$$cof\_z^{0}=-0.0003*T.V.^{3}+0.0045*T.V.^{2}-0.0294*T.V.+0.1268 \tag{2.10}$$

Coefficients of generalized form of (2.5) can be calculated for any threshold value (T.V.) by using equations (2.7)-(2.11). Generalized M.C.V.-I.V. equations can be represented as,

$$(M.C.V.) = cof\_z^3 * (I.V.)^3 + cof\_z^2 * (I.V.)^2 + cof\_z^1 * (I.V.) + cof\_z^0 \qquad (2.11)$$

Same operation is going to be accomplished for M.C.V.-N.V. equation such that the function at which noise variance maximum correlation value is achieved when input signal variance is known. In this case, since its maximum degree is 2, coefficient matrix will be in different dimensions.

$$\begin{bmatrix} N.V._{T=2V} \\ N.V._{T=2.25V} \\ N.V._{T=2.5V} \\ N.V._{T=2.75V} \\ N.V._{T=3V} \\ N.V._{T=3.25V} \\ N.V._{T=3.5V} \\ N.V._{T=3.75V} \\ N.V._{T=4V} \end{bmatrix} = \begin{bmatrix} M.C.V.^2 & M.C.V.^1 & M.C.V.^0 \\ a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \\ a_{41} & a_{42} & a_{43} \\ a_{51} & a_{52} & a_{53} \\ a_{61} & a_{62} & a_{63} \\ a_{71} & a_{72} & a_{73} \\ a_{81} & a_{82} & a_{83} \\ a_{91} & a_{92} & a_{93} \end{bmatrix} \qquad (2.12)$$

Figure 13 shows curve fitting results for each coefficient column of (2.12). As best fitting polynomial, second degree representation is chosen for each coefficient. Fitting is accomplished by the use of Matlab function named as “polyfit”.

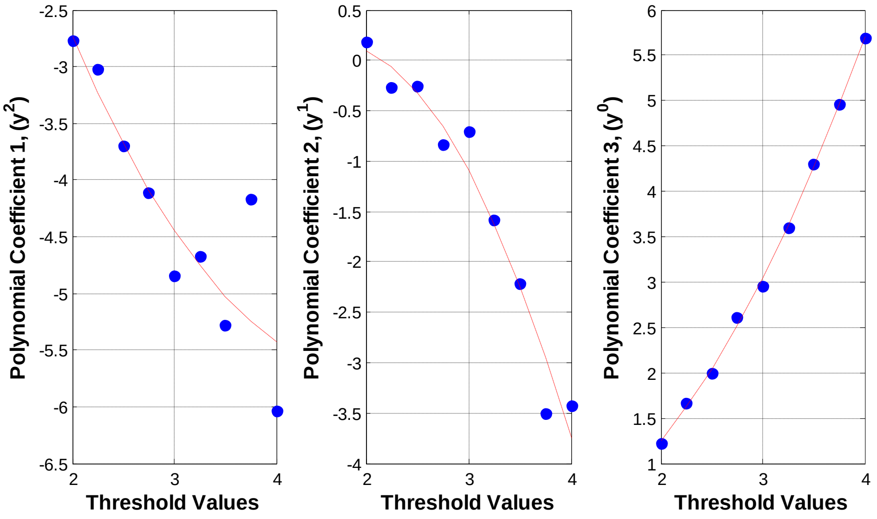


Figure 13 Coefficient fitting for M.C.V.- N.V. equation

$$cof\_y^2 = 0.36 * T.V.^2 - 3.5287 * T.V. + 2.8724 \tag{2.13}$$

$$cof\_y^1 = -0.7294 * T.V.^2 + 2.4563 * T.V. - 1.8983 \tag{2.14}$$

$$cof\_y^0 = 0.4310 * T.V.^2 - 0.3637 * T.V.^1 + 0.2574 \tag{2.15}$$

By the help of the mathematical expressions (2.13) up to (2.15) coefficients of generalized form of (2.4) can be found for any threshold value (T.V.) of a nonlinear system. Generalized M.C.V.-N.V. equation can be represented as:

$$(N.V.) = cof\_y^2 * (M.C.V.)^2 + cof\_y^1 * (M.C.V.) + cof\_y^0 \tag{2.16}$$

In the final stage of system training, we explain the method used to obtain the equation for a given input signal variance and system threshold value in order to achieve best detection performance. Moreover, it is important to state that operation is mostly based on curve fitting method rather than the simulation data. In the next topic, simulation and equation results are going to be compared to be able check performance of the proposed system training products.

2.3.2. Final stage results

As it's explained in part 2.3.1., equations are obtained for the threshold values of 2V, 2.25V, 2.5V, 2.75V, 3V, 3.25V, 3.5V, 3.75V and 4V. To check equations represented in the previous part, new threshold values that is not used to obtain any equations are needed. In this way, 3 different threshold values are fed into the equations and their computation and simulation results are compared.

First threshold level is chosen as 2.35V. At the beginning, this threshold value is used in equations (2.7), (2.8), (2.9), and (2.10) and its corresponding M.C.V.-I.V. equation is found. Then for input signal levels starting from 1 V up to 2.30 V, input variances are calculated and fed into this equation. Moreover, in parallel, SR simulation results for threshold of 2.35V are gathered. Fig. 14 displays comparison between experimental data and calculation results.

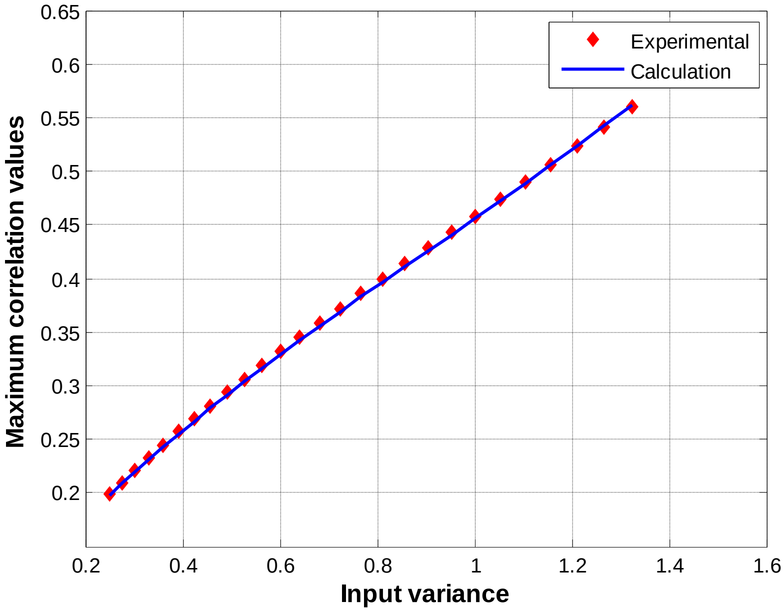


Figure 14 Experimental and system training results for M.C.V.- I.V. equation at

T = 2.35 V

M.C.V. - I.V. equation is constructed for T = 2.35 V and maximum correlation value for the given subthreshold data set is found. Results seem to fit simulation data perfectly. However, further calculation is needed. Although maximum correlation that can be reached for the certain input signal is known; it isn't known that at which noise intensity that maximum correlation occurs. Further calculation to estimate optimum noise intensity is done by the use of the equations (2.13), (2.14), (2.15) and (2.16) for T.V. of 2.35 V. Overall, resulting experimental data and calculation results are shown in Fig. 15.

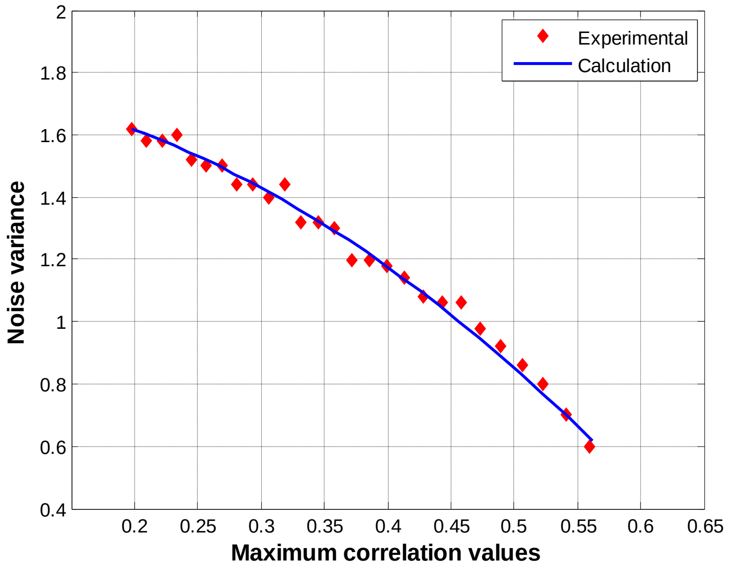


Figure 15 Experimental and system training results for M.C.V.-N.V. equation at

T = 2.35 V

Although some of the simulation data fit the calculation results, most of them do not fit completely but stands very close to the curve. Solution to this drawback of the presented model can be found in the description of the SR phenomenon itself, and is going to be explained after representing other two threshold values.

The second threshold value of 3.15 V and third threshold value of the 3.85 V are selected for more trials; their operation steps to compare their calculation results with the simulation data are the same as the first one. Results of their corresponding M.C.V.-I.V. equations are found and compared in Fig. 16.

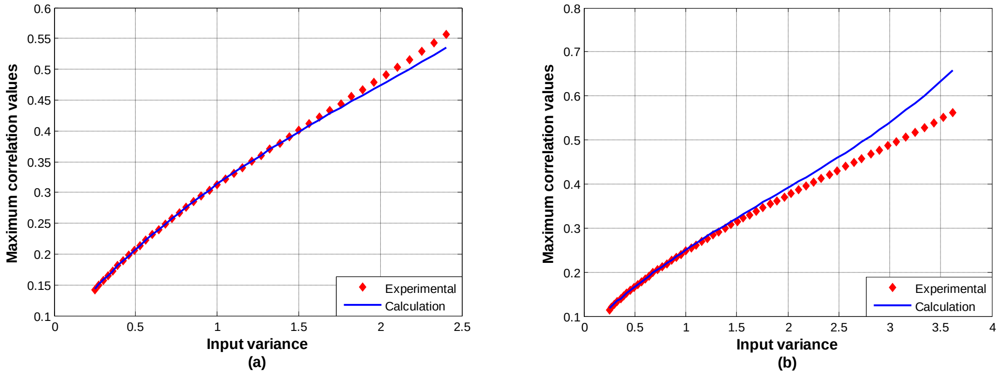


Figure 16 System training result for M.C.V.-I.V. equation (a) at T = 3.15 V and (b) T = 3.85 V

Furthermore, M.C.V.- N.V. equations are constructed and compared to experimental results in Fig. 17.

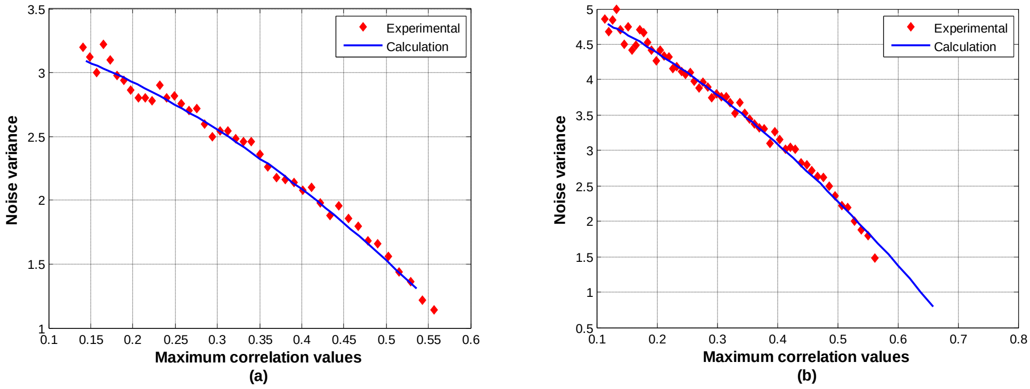


Figure 17 Experiment and system training result for M.C.V.- N.V. equation (a) at T = 3.15 V and (b) at T = 3.85 V

Figure 17 shows similarity with Fig. 15 in case of fitting problem of simulation data with respect to equation results. As mentioned before, this drawback is going to be eliminated by using a characteristic property of SR itself and is going to be explained in the next section.

2.3.3. Further discussions

Resonance as a term of physics stands as one value that causes oscillations and sudden amplitude changes on a given system. Although, name of the stochastic resonance contains the word of “resonance”, its meaning isn’t the same as the general usage of the word. SR occurs when a set of high values are observed with consecutive noise intensities. In other words, SR occurrence represents a zone not a single value. This zone can be represented as in Fig. 18 and it’s determined by eye inspection for graphical illustration.

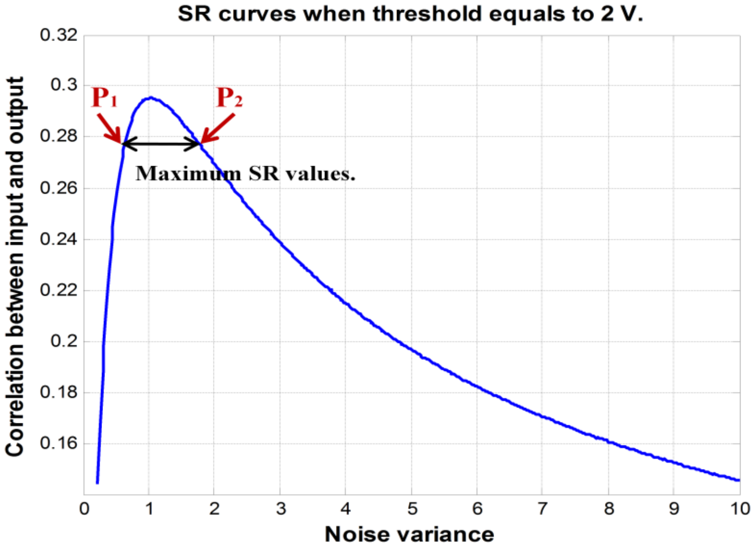


Figure 18 SR occurrence zone

Existence of zones will help to improve the system training’s detection performance. It’s observed that some calculation results unfit the simulation results in Fig. 15, 16, 17, so performance will be higher by introducing SR zones for every threshold value. Fig. 18 shows SR result for an input signal of 1.25 V with constant threshold of 2 V. Since there is no mathematical expression to define such a zone in literature, determination of an SR zone for any input signal and threshold is going to be done by the following equation

$$P_{1,2} = (A \pm k\Delta), where \quad \Delta = \frac{A}{5}, k = 1.5 \qquad (2.17)$$

where P$_1$ and P$_2$ represent points that bounds SR zone, A is the point (noise variance) that maximum correlation occurs, Δ is related with A where k is a constant number. Equation (2.17) is applied for all input signals and all threshold values (2 V to 4 V with incremental step size of 0.25 V) and optimum value of the k and Δ is achieved.

Figure 19 shows the results of (2.17) and related coefficents. Red dots represent zone borders for each input signal whose voltage level falls into the range 1 V - 1.95 V while the threshold value is kept at T = 2 V.

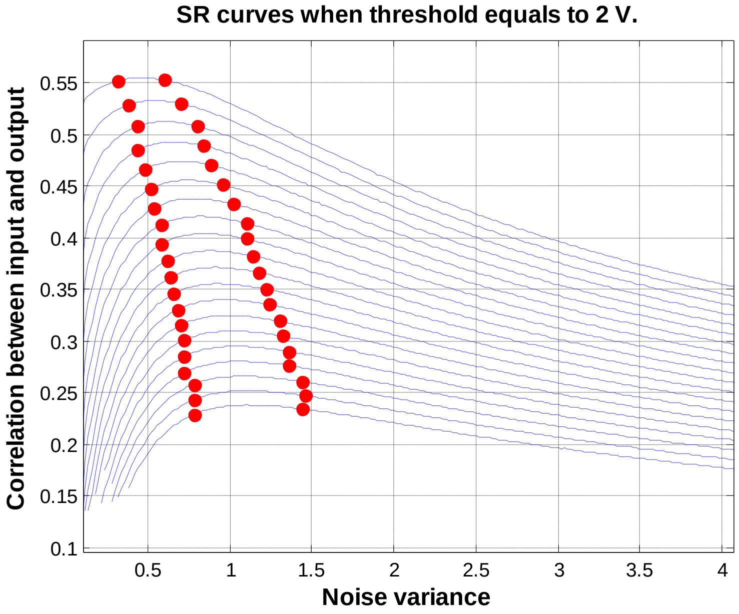


Figure 19 Zone borders for SR

To define a generic SR occurance region equation for any threshold value, further simulation is performed. SR occurrence zone border points at Fig. 19 are formed as two seperate lines and Fig. 20 shows the resulting lines for T = 2V.

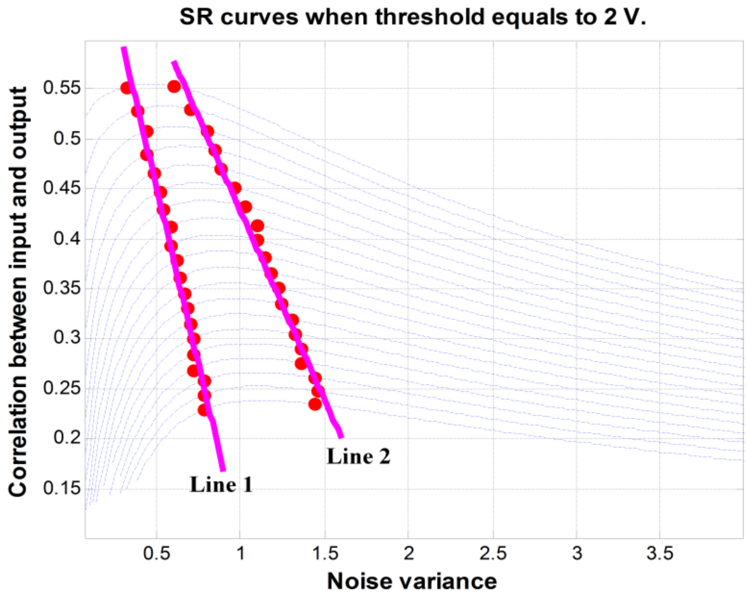


Figure 20 SR occurrence zone boundaries for different levels of input signals

Equations (2.18) and (2.19) represents 'Line 1' and 'Line 2' respectively.

$$Line1 = M.C.V. = -0.70867*(N.V.) + 0.80531 \tag{2.18}$$

$$Line2 = M.C.V. = -0.37785*(N.V.) + 0.80537 \tag{2.19}$$

Next step is to find all boundary line equations for different threshold values in the range 2V - 4V with incremental steps of 0.25 V in order to observe relation between coefficients of line polynomials for further calculation. All coefficients of line equations are stored in an array, when each column of the array is plotted separately Fig. 21 is observed. In this Figure, second degree polynomial representation is selected as the best fitting option to represent discrete data points.

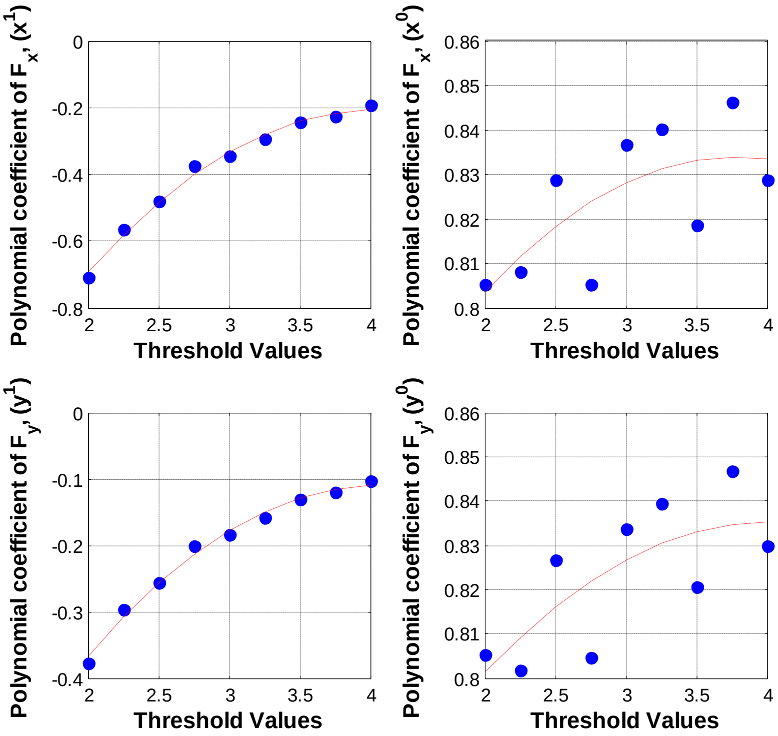


Figure 21 Coefficient fitting for generic SR occurrence region boundaries

$$cof\_x^1 = -0.1139 * T.V.^2 + 0.9261 * T.V. - 2.0875 \tag{2.20}$$

$$cof\_x^0 = -0.0095 * T.V.^2 + 0.717 * T.V. + 0.6983 \tag{2.21}$$

$$cof\_y^1 = -0.0606 * T.V.^2 + 0.4924 * T.V. - 1.1094 \tag{2.22}$$

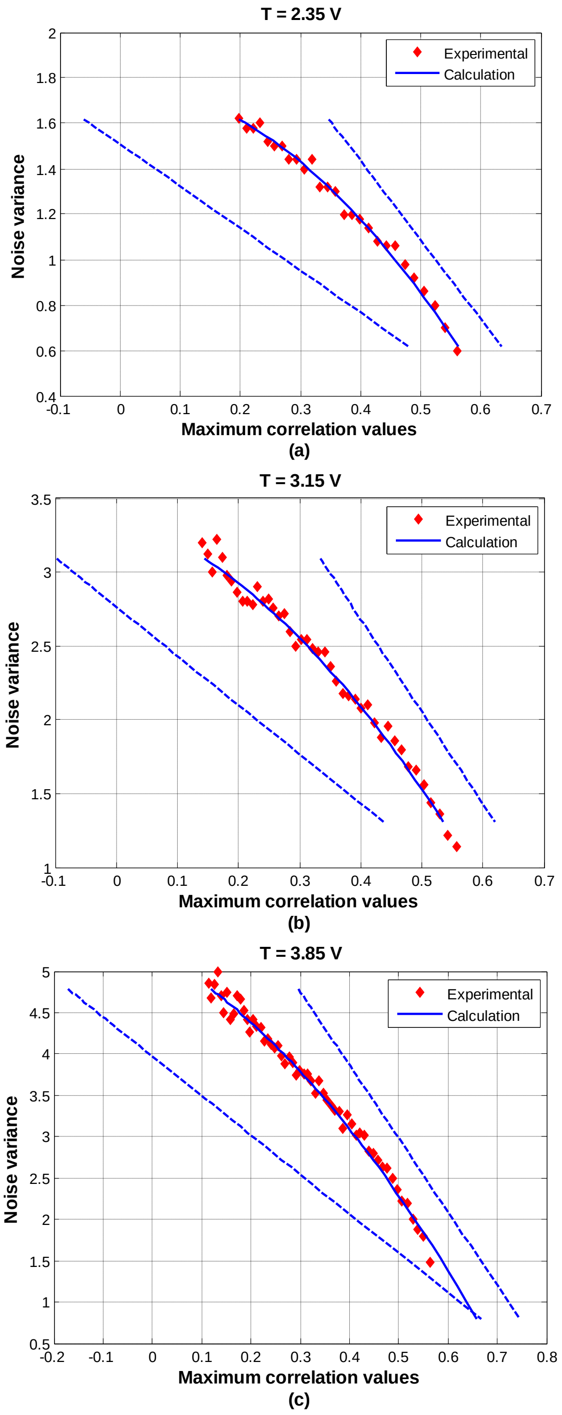
$$cof\_y^0 = -0.0082 * T.V.^2 + 0.0663 * T.V.^1 + 0.7071 \tag{2.23}$$

Generalized form of ‘Line 1’ and ‘Line 2’ for thresholds in the range 2V - 4V can be constructed using equations (2.20) - (2.23). Borders of SR for an input signal, can be found by putting system threshold value of T.V. into (2.20) - (2.23). Then ‘Line 1’ and ‘Line 2’ might be constructed using. (2.24) and (2.25).

$$Line1 = cof\_x^1 * (T.V.) + cof\_x^0 \tag{2.24}$$

$$Line2 = cof\_y^1 * (T..V.) + cof\_y^0 \tag{2.25}$$

In this part SR zone for any threshold value and any input signal variance is explained. In Figs. 15 and 17 simulation data and derived equation results are depicted together for threshold values T = 2.35V, T = 3.15V and T = 3.85V for adequate comparison purpose. In addition, in Fig. 22, SR zone definitions for each threshold value is calculated using (2.24) and (2.25) and SR occurrence borders are added to the graph.



(a)

(b)

(c)

Figure 22 Experiment and system training results with SR occurrence boundaries for (a) at T = 2.35 V, (b) at T = 3.15 V and (c) at T = 3.85 V

Figure 22 shows that unfitted simulation results stays in the SR occurrence zone and as noticed, none of simulation data crosses the boundary lines. In this case, presented model for SR seems to be successful.

CHAPTER 3

3. SR EFFECT ON A FIELD EFFECT TRANSISTOR

FET as a semiconductor device has three terminals which are named as drain, source and gate. A channel is formed in between drain and source terminals when sufficient amount of voltage (V_{GS}) is applied to the gate terminal. In this case, a current, I_{DS}, flows through terminals. In other words, when sufficient amount of voltage is available at the gate terminal. I_{DS}, drain to source current becomes observable and FET goes on state. The phrase of "sufficient amount of voltage" refers to a threshold, moreover in this case FET represent a nonlinear system.

In this chapter, implementation of SR effect on a FET and demonstration of overall SR curves are studied. In order to complete defined task, a set of hardware-software combination of operations are going to be performed. A classical SR effect demonstration process needs three main components. First one is a nonlinear threshold system while second one is a configurable signal generation device to create subthreshold input signal and noise signal. Consequently, the last one is a tool to compare input and output signals of the system in terms of performance metrics.

This chapter consists of two sections; one of them is dedicated to the explanation of tools that are going to be used for observation of SR effect on a transistor. Second part contains explanations about obtaining SR curves and results.

3.1. Operation Cycles

Starting point of operation will be the "Graphical User Interface" part of Fig. 23. Input and noise signals' characteristics are adjusted from the graphical user interface (GUI). A Field Programmable Gate Array (FPGA) is used as the controller of the hardware. FPGA controls the data flow and drives the analog digital converter.

FPGA implementation is achieved through VHDL (VHSIC Hardware Description Language) programming language. Operation cycle of the experiment is shown in Fig. 23. Operation ends in the Graphical User Interface block. More information about the operation steps are going to be described in following subsections.

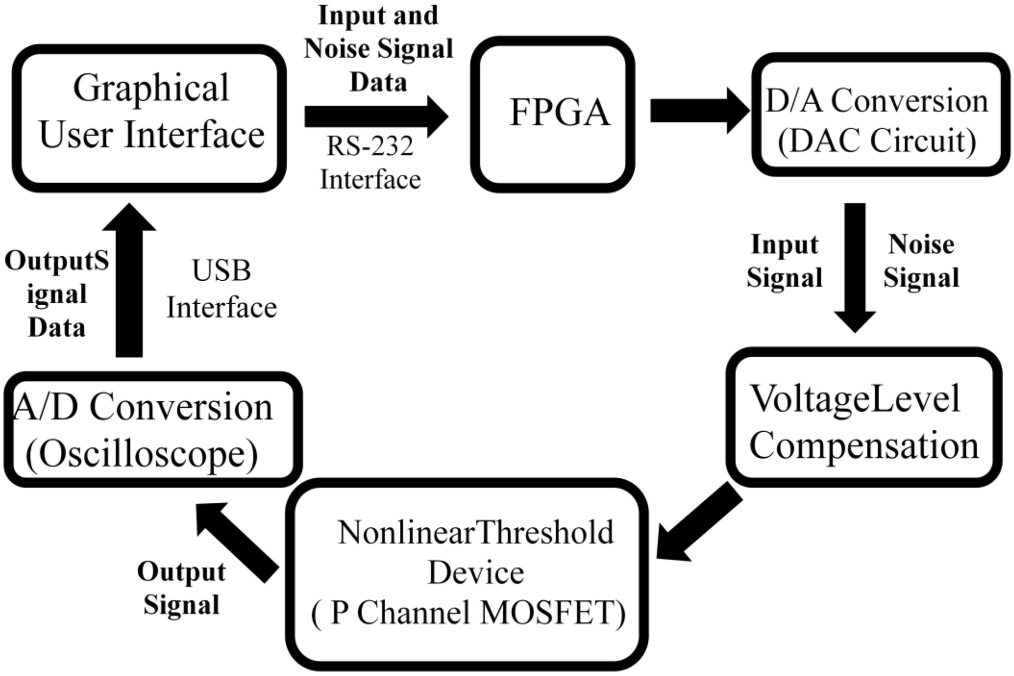
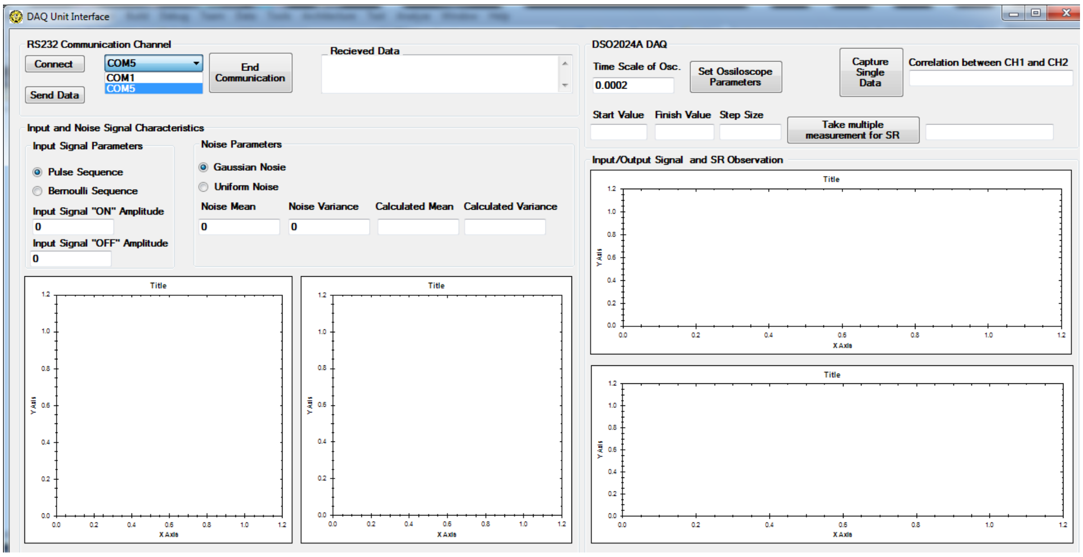

Graphical User Interface
Input and Noise Signal Data
RS-232 Interface
FPGA
D/A Conversion (DAC Circuit)
Input Signal
Noise Signal
VoltageLevel Compensation
NonlinearThreshold Device (P Channel MOSFET)
Output Signal
A/D Conversion (Oscilloscope)
OutputSignal Data
USB Interface

Figure 23 Operation Schematic for Practical Observation of SR

3.1.1. Graphical user interface

The operation cycle starts and ends at GUI which is developed on Microsoft Visual Studio 2010 C# .NET as a Windows Application Form. To be able to manage the SR effect observation process, interface sends digital data and noise signals to the FPGA through RS-232 communication interface cable using Serial Communication Interface (SCI) protocol. Then it takes input and output data from the oscilloscope through the Universal Serial Bus (USB) interface to perform correlation calculations.



COM5

● Pulse Sequence
○ Bernoulli Sequence

Input Signal "ON" Amplitude: 0

Input Signal "OFF" Amplitude: 0

● Gaussian Nosie
○ Uniform Noise

Noise Mean: 0
Noise Variance: 0
Calculated Mean:
Calculated Variance:

Time Scale of Osc.: 0.0002

Figure 24 First view of GUI

Fig. 24 shows a general view of GUI. As noticed, GUI design is separated into four parts which are named as “RS232 Communication Channel”, “Input and Output Signal Characteristics”, “DSO2424A DAQ” and “Input / Output Signals and SR observation” and each part has a unique duty. First part’s duty is to open, close or configure the SCI protocol’s communication channel. Channel links hardware and software part through a RS232 cable.

Second part adjusts input and noise signal properties like type, magnitude and peak to peak value of the input signal, to configure type, mean and variance value of the noise signal. There are two choices for input signal; pulse signal and a Bernoulli sequence. Frequency of the pulse signal isn’t configurable and set to 770 Hz which isn’t chosen for a particular reason but it is a result of hardware design. Bernoulli sequence consists of 32 bits and constructed by using (2.1). Moreover, there are two choices for noise signal distribution, Gaussian and Uniform. Since there is no ready-to-use function to create zero mean Gaussian noise signal on C #, Box-Muller method [24] is used for noise generation. For comparison purposed zero mean uniform noise signal is generated using “rand” function in Matlab environment. For both noise signals, validity of created signals through the GUI is proven by using Matlab. Fig. 25 shows noise signal data distributions, which are created at GUI, on a

Matlab histogram figure. Additionally, their mean values are calculated -0.0499 and 0 respectively.

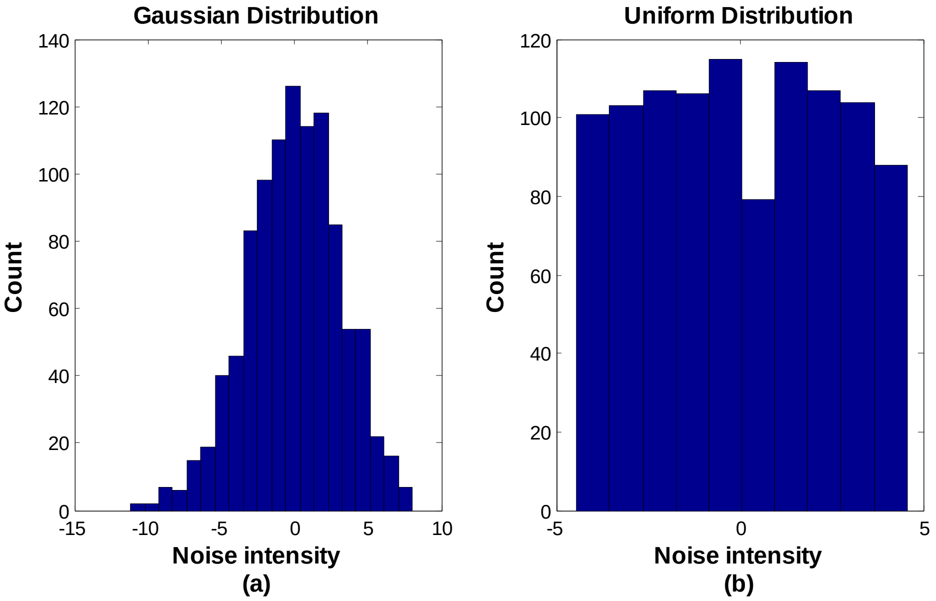
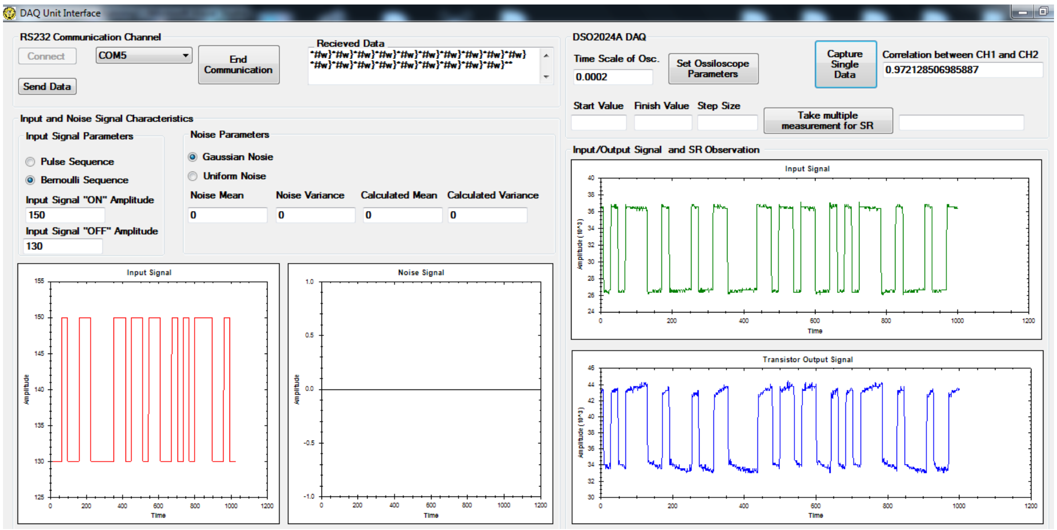


Figure 25 Noise signal distributions (a) Gaussian (b) Uniform

After configuration, we send data to hardware by simply pressing “Send Data” button on GUI using the first part. At the same time input and noise signal are plotted on two separate figures. Besides, third part is dedicated to data acquisition; it configures data acquisition device and control the process such that whether a single or consecutive sets of data capturing are going to be taken place or not. As a data acquisition device, oscilloscope is used and it is going to be explained through the following Section of 3.1.6 in details. The part of “Input / Output Signals and SR observation” in the GUI, are used to show the operation results. If “Capture Signal Data” button is pressed, both input and output signals are plotted on two separate graphs for accurate inspection. However, if one presses “Take multiple measurements for SR” button; it controls the process as presented in Chapter 2 operation steps of system training part of stage 1. At the end it plots the overall curve with respect to correlation versus noise variance.

Figure 26 shows a sample operation for the case of single set of data capturing. As seen, a Bernoulli sequence is created without any noise presence and graph is available on the left side. On the right side of Fig. 26 input and output signals coming from data acquisition unit of the threshold system are shown and correlation between these two signals is approximately calculated as 0.972. Since, threshold value and nonlinear system characteristics aren't introduced yet; further experiments are going to take place in the part 3.2.



COM5

Recieved Data
#w}#w}*#w}*#w}*#w}*#w}*#w}*#w}*#w}*#w}
#w}#w}*#w}*#w}*#w}*#w}*#w}*#w}*#w}**

○ Pulse Sequence
● Bernoulli Sequence

Input Signal "ON" Amplitude: 150

Input Signal "OFF" Amplitude: 130

● Gaussian Nosie
○ Uniform Noise

Noise Mean	Noise Variance	Calculated Mean	Calculated Variance
0	0	0	0

Time Scale of Osc.: 0.0002

Correlation between CH1 and CH2: 0.972128506985887

Figure 26 Data flow and management

Additionally, plotting operations on GUI are succeeded with a third party toolbox named as ZedGraph. After inserting ZedGraph's Dynamic Link Library (DLL) file into GUI project, plotting of desired data sets is properly enabled.

A practical GUI easy to use is developed as the main part of the SR effect observation process. AND, plotting a graph, input and noise signal generation, controlling RS232 interface, controlling a device, calculating correlation, variance, and thresholding functions are implemented on C # code successfully and used adequately in order to reach objectives.

3.1.2. FPGA Implementation

FPGA is used to control data flow between GUI and hardware; moreover it is used as a component of signal generation operation. Consequently, a system development platform that includes an FPGA chip with suitable peripheral units is needed. The

Nexys 3 digital system development platform is chosen to implement the desired functions on FPGA. Nexys 3 development board, shown in Fig. 27 (a), contains a Xilinx Spartan-6 FPGA (XC6LX16-CS324) chip and ready to use peripheral units as shown in Fig. 27 (b). Basic I/O pins, 100 MHz clock and USB-UART connection are used among presented peripherals. UART stands for Universal Asynchronous Receiver/Transmitter which is a kind of integrated circuit and is commonly used with the serial communication interface of RS-232.

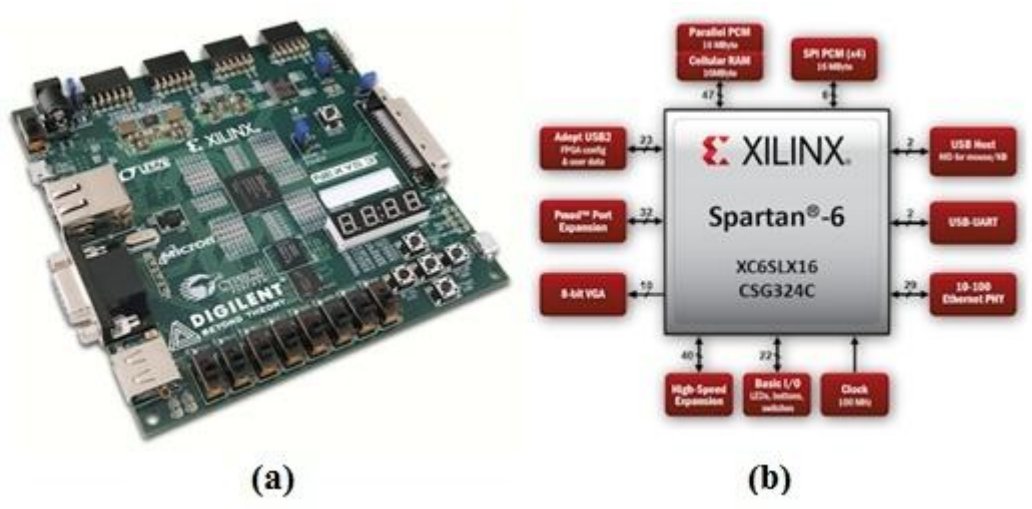


Figure 27 (a)FPGA platform and (b) Its peripherals [25]

FPGA implementations are accomplished by using VHDL. VHDL is an official IEEE standard called as IEEE 1164. It was developed as a documentation language in order to describe complex digital circuits and also was used to model behavior of digital circuits. Nowadays, in addition to these features it is widely used in hardware implementation in Computer Aided Design (CAD) systems [26]. FPGA code development is done using VHDL hardware programming language, and it is developed at Xilinx’s CAD system which is named as ISE Design Suite. Webpack Edition of ISE Design Suite version 14.2 is used on 32 bit Windows 7 operating system.

In the name of flawless handling of internal operation and data flow management, implementation is divided into blocks and furthermore, blocks are also divided into sub-blocks. As a result, FPGA implementation is consisted of three main blocks. One of them is SCI_CONTROL block, the other one is DAC_DRIVER block and the last one is the CLOCK_DIVIDER block, and these blocks are shown in Fig. 28. The main unit has 3 inputs and 3 outputs. Clock frequency is 100 MHz, “RX” is used as receiver pin for the serial communication in between computer and FPGA, ”SW” represents a switch on Nexys 3 board and it is used as on/off switch of signal generation. Outputs are “TX”, “DAC1_DATA” and, “DAC2_DATA”. “TX” is used as transmitter pin for the serial communication between computer and FPGA, both “DAC1_DATA” and “DAC2_DATA” has 8 pins which are connected to the two separate digital analog converter circuits.

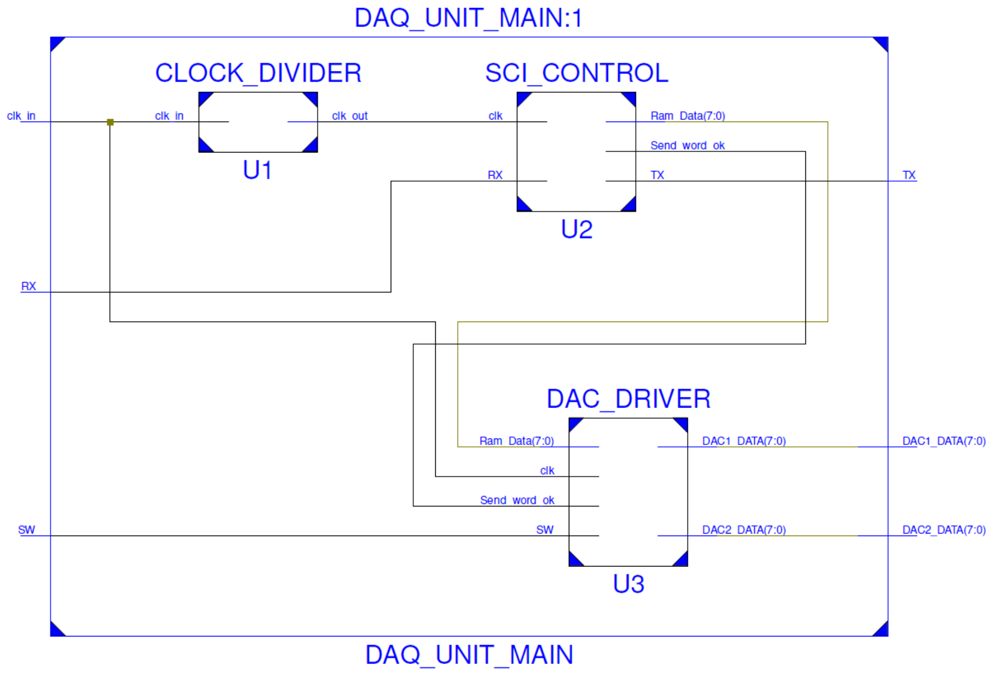


Figure 28 Main blocks of hardware implementation

SCI_CONTROL block consists of three sub-blocks, which are named as RX_MODULE, TX_MODULE and DATA_FLOW_CONTROL as shown in Fig. 29. Baud rate of the serial communication is configured as 115200 bits per second with 1 start bit, 1 stop bit and no parity. SCI_CONTROL block takes the input signal and noise signal from the GUI through RS-232 communication interface sends both data sets simultaneously to an array which is constructed in DAC_DRIVER block. It’s also important to state that same data set is sent to the host (GUI) to be able to check whether the input signal is transmitted without any loss or not.

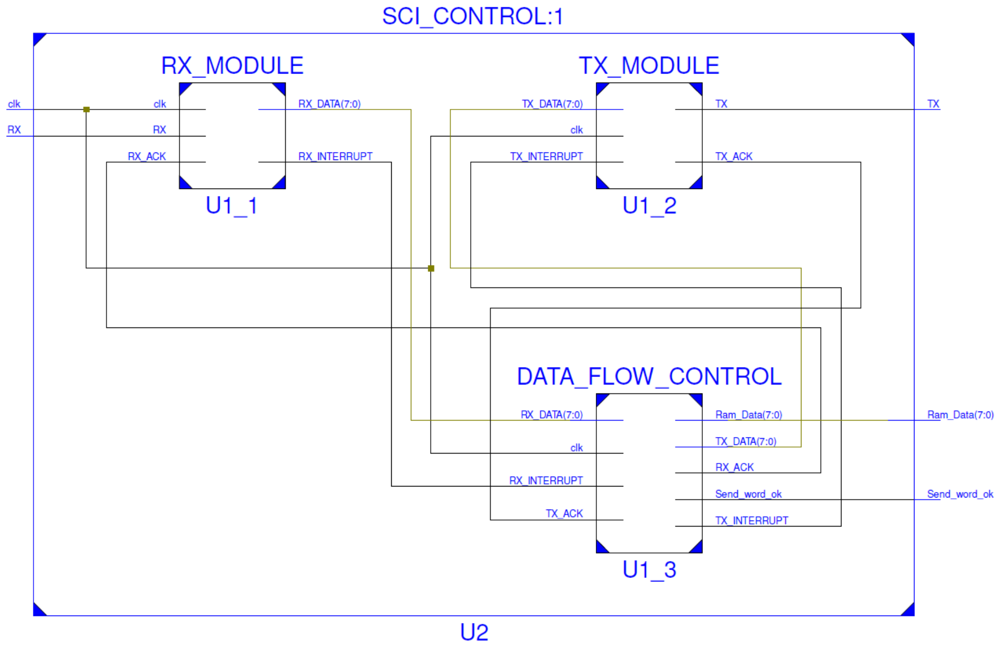


Figure 29 SCI control block

DAC_DRIVER consists of one block and handles two operations. First operation is filling the data array according to SCI_CONTROL block's ready flag. When the ready flag logic is '1', then 8 bits of data from the SCI_CONTROL block is assigned into array and index is incremented by one. Array is constructed to hold 2048 bytes of data consisting of two equally divided data sets. 1024 of them hold noisy data (noise is mounted on to input) and 1024 of them are used for the original input data. These two separate data sets are sent to two different DAC circuits simultaneously.

3.1.3. Digital analog converter circuit

Digital Analog Converter circuit takes the digital data in parallel (as eight bit) from the FPGA board and converts it to an analog signal. To accomplish conversion, Texas Instrument's DAC0800 IC is chosen. DAC0800 is a high speed, current output DAC [27] and it is easy to use, since it has no activation pin and no need to clocking. Since it is a current output DAC, some additional circuit elements should be added. In order to use DAC adequately the circuit in Fig. 30 is constructed.

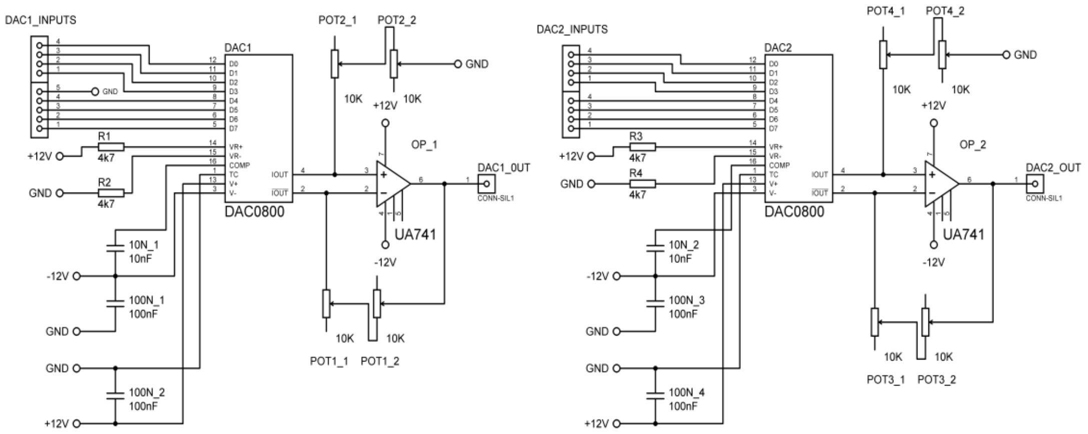


Figure 30 DAC circuits

As seen in the Fig. 30, there are two separate DAC circuits; one of them generates noisy signal and the other one generates the input signal. “DAC1_INPUTS” and “DAC2_INPUTS” are the eight-bit parallel input pins of the DAC0800 and are fed from the FPGA board directly. Since the DAC0800 is a current output type integrated circuit, output should be converted to voltage by using some additional circuit elements. In this study, current-to-voltage amplifier configuration is used. DACs and Op-amps are fed with +12V and -12V. Peak to peak value of the resulting signals can be adjusted by using potentiometers. Peak to peak value of the signals are set to approximately 22 V i.e., voltage swing is from +11 V to -11 V. Outputs of the overall circuit, named as “DAC1_OUT” and “DAC2_OUT”, are connected to the voltage level compensation circuits.

3.1.4. Voltage level compensation circuits

Voltage level compensation circuits are used to transform DAC outputs into a proper level such that SR effect conditions are satisfied. The circuit that accomplishes this task includes amplifying and shifting. One operational amplifier (op-amp), LF 412 (U1: A) is used to perform shift operation of in between - 3 V to - 4 V. Amount of shift in voltage level can be adjusted by using a potentiometer (R4). Other op-amp in LF412 (U1: B) is used degrade overall signal by 0.5. After U1:A and U1:B noisy signal is ready to be applied to the nonlinear threshold device. Since at the output of

FET current is available, it's converted into voltage and buffered by using LF 411 op-amp to be able to capture output signal through the oscilloscope, DAQ unit.

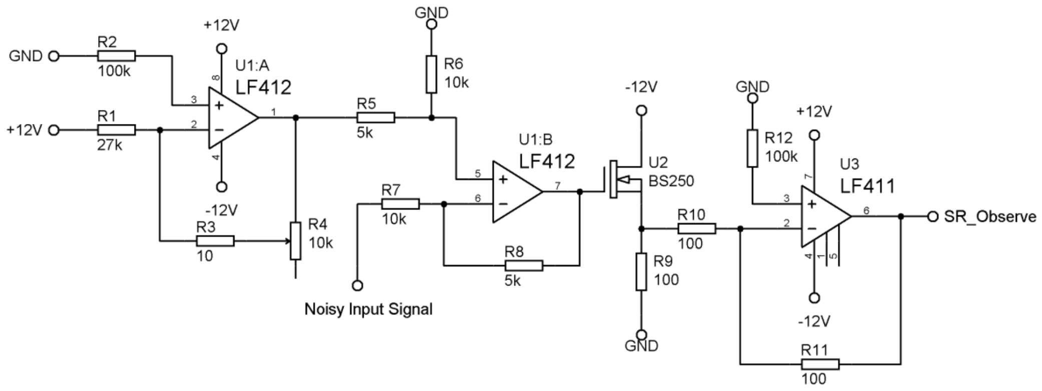


Figure 31 Voltage compensation circuits

3.1.5. Nonlinear Threshold Device

Vishay Intertechnology‘s p-channel MOSFET, BS250, is chosen (Fig. 31) as nonlinear threshold device. BS250 is chosen because it has fast switching speed like 16 ns [28], low turn on and turn off time, for instance 10 ns [28], and it has low threshold, i.e., between -1 V and -3.5 V.

To be able to apply and observe SR effect, transistor's characteristics should be known very well. Transistor I-V curve graph, provides some useful information like, saturation region, pinch off voltage, and subthreshold region. I-V curve can be obtained by measuring current flow and applied voltage across an electronic component's terminals. In the case of field effect transistor, I stands for I_{DS} (Drain to Source current) and V stands for V_{DS} (Drain to Source voltage). I_{DS} and V_{DS} values are measured at different gate voltage, V_{GS} values. As a result, I_{DS} –V_{DS} curves for different V_{GS} values are obtained as in Fig. 32.

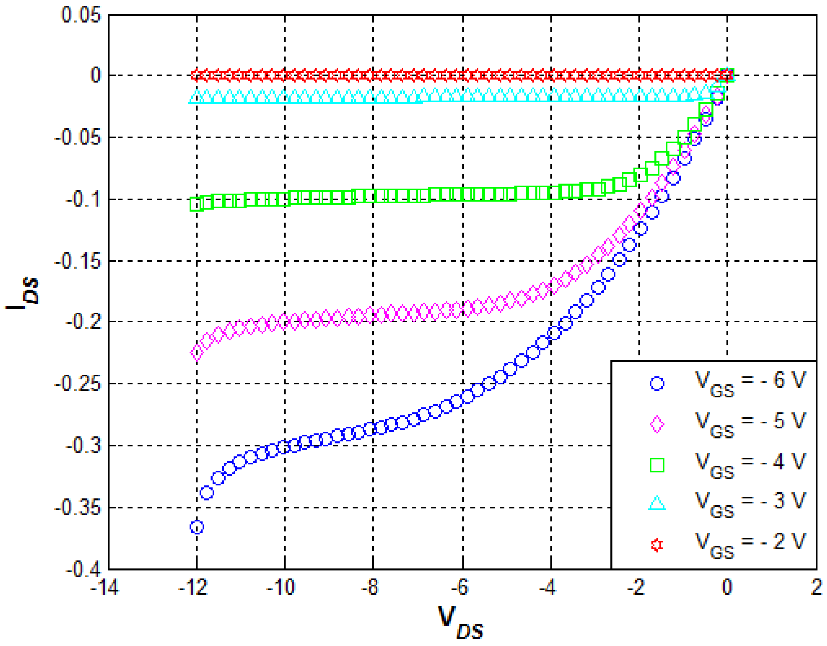


Figure 32 BS250 I-V curve

As mentioned before, BS250 is a p-channel MOSFET. Negative gate-to-source and gate-to-drain voltages are applied in order to obtain the I-V characteristic curve of field effect transistor.

Experimental setup in Fig. 33 shows that a MOSFET's I_{DS} -V_{DS} characteristic can be obtained through a half automatic operation flow. The operation is automatically performed because it is managed from a source measurement unit named as Keithley 2400 through a Labview graphical user interface. Interface allows using the instrument as voltage source, voltage and current measurement unit. Consecutive V_{DS} voltages are applied from 0 V to 12 V at certain V_{GS} then, voltage and current are being measured and saved in an array. Operation is half automatic because, to be able to obtain data for different V_{GS} values, ranging from -2 V to -6 V, voltage of gate terminal must be changed manually from the DC power supply.

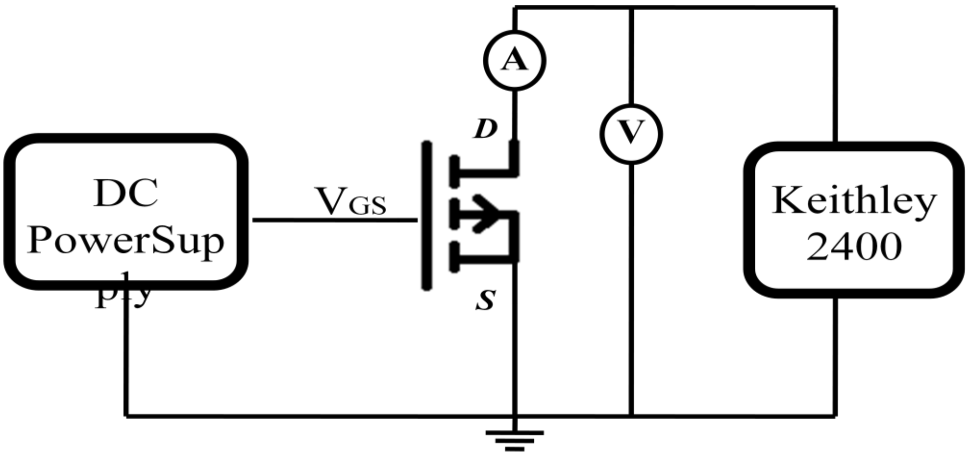


Figure 33 Experiment setup to obtain I-V curve of BS250

3.1.6. Oscilloscope

Agilent Technologies' DSO2024 2000 series oscilloscope is used for signal observation and analog to digital conversion. It has 2 GSa/s sampling rate with 16 bit resolution.

In order to use oscilloscope as a data acquisition unit, Standard Commands for Programmable Instruments (SCPI) are used by the GUI unit. SCPI is a common language for instrument control which is defined with the IEEE 488.2 specification. SCPI allows users to program the instrument and to handle data usage.

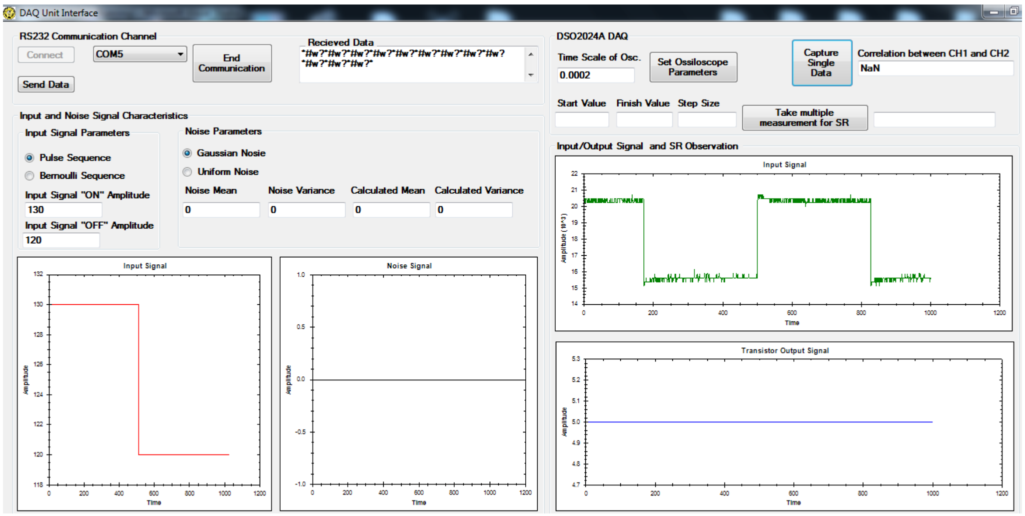
Standard Commands for Programmable Instruments are used to control the oscilloscope through the GUI developed using C# .Net on Microsoft Visual Studio 2010. A DLL file for the control of Agilent 2000 series oscilloscopes is inserted into GUI project. This DLL file allows using SCPI commands in the form of C# syntax.

Communication and data acquisition between computer and oscilloscope are accomplished using USB protocol. An USB cable between two instruments allows high rate data flow. In this study, there is no need for real time data flow. It is sufficient to take 10000 data points from one period of both signals in order to calculate the correlation coefficient between input and output signals at each operation cycle.

Channel 1 and Channel 2 on the oscilloscope is dedicated for data acquisition. Channel 1 is tied to the output of the circuit shown by “SR_Observe” in Fig. 32. On the other hand, Channel 2 is tied directly to the input signal.

3.2. Operation Results

In the previous parts, the construction of an experiment setup to observe SR effect on a threshold device is explained. It is mentioned that there are three things to perform robust experimental process. First thing is to have a threshold system; this condition is satisfied by using a FET device and its characteristics are examined in part 3.1.5. Second thing is a tool to generate input and noise signals at desired amplitude. Three trials are going to be shown in the following text in order to explain GUI operation clearly.


DAQ Unit Interface

RS232 Communication Channel

COM5

Recieved Data: *#w?*#w?*#w?*#w?*#w?*#w?*#w?*#w?*#w? *#w?*#w?*#w?*

Input and Noise Signal Characteristics

Input Signal Parameters

● Pulse Sequence
○ Bernoulli Sequence

Input Signal "ON" Amplitude: 130

Input Signal "OFF" Amplitude: 120

Noise Parameters

● Gaussian Nosie
○ Uniform Noise

Noise Mean	Noise Variance	Calculated Mean	Calculated Variance
0	0	0	0

DSO2024A DAQ

Time Scale of Osc.: 0.0002

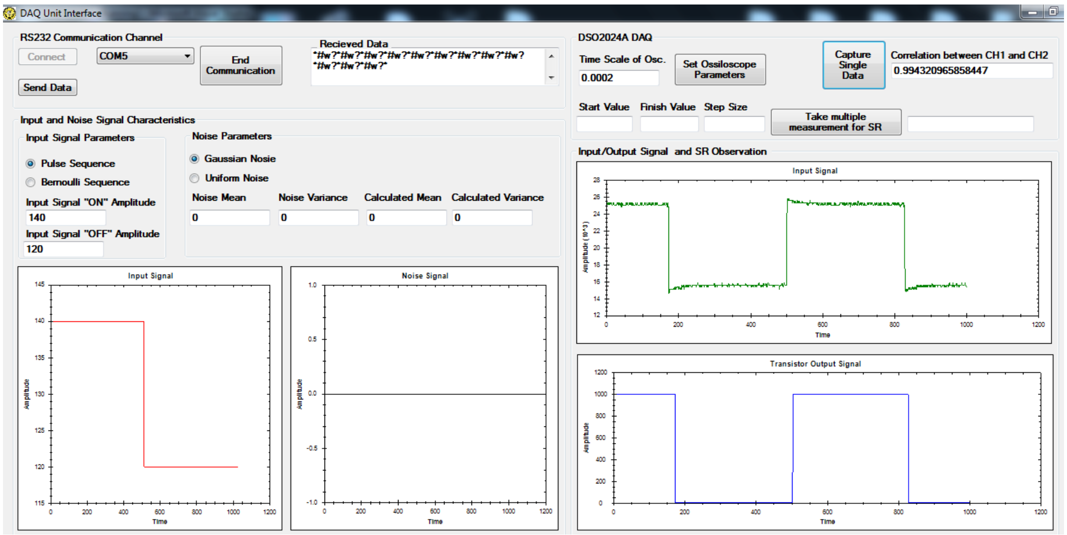
Correlation between CH1 and CH2: NaN

Start Value | Finish Value | Step Size

Input/Output Signal and SR Observation

Figure 34 Subthreshold signal is applied without noise

In Fig. 34, a subthreshold signal is applied to the threshold device without any noise presence. In the “Input and Noise Signal Characteristics” part, input signal type selected as pulse and “ON” amplitude of the signal is set as 130 and correspondingly “OFF” amplitude is set to 120. These two numbers represent decimal value of bit representation. Since, DAC that is used in the hardware has 8 bit resolution; numbers between 0 and 255 can be used. It is important to state that in the experiments, one bit equals to 40 mV. Circuitry is configured as the value of 136 equals to the threshold voltage value of the BS250. As a result, input signal presented in Fig. 34 is below threshold consequently, no output signal is observed.



RS232 Communication Channel

COM5

Recieved Data

#w?#w?*#w?*#w?*#w?*#w?*#w?*#w?*#w?
#w?#w?*#w?*

Input and Noise Signal Characteristics

Input Signal Parameters

● Pulse Sequence
○ Bernoulli Sequence

Input Signal "ON" Amplitude: 140

Input Signal "OFF" Amplitude: 120

Noise Parameters

● Gaussian Nosie
○ Uniform Noise

Noise Mean	Noise Variance	Calculated Mean	Calculated Variance
0	0	0	0

DSO2024A DAQ

Time Scale of Osc.: 0.0002

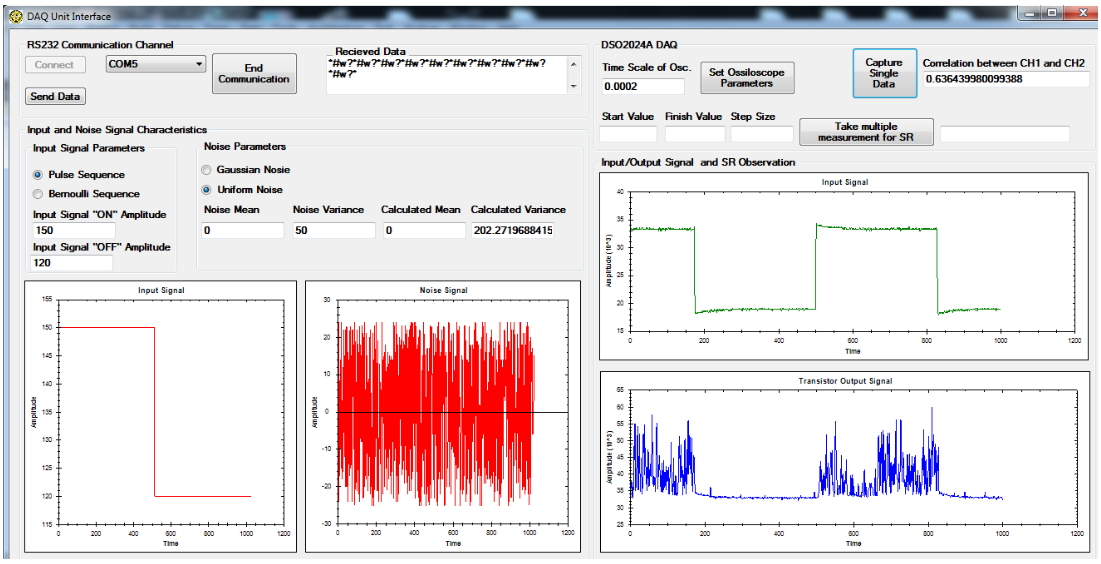
Correlation between CH1 and CH2: 0.994320965858447

Start Value | Finish Value | Step Size

Input/Output Signal and SR Observation

Figure 35 Subthreshold signal is applied without noise

In Fig. 35, amplitude of the input is chosen such that it can cross the threshold voltage of the transistor. This kind of a signal can be called subthreshold signal. Accordingly, an output signal is observed and plotted in the GUI. Also, correlation is calculated as 0.994 which shows GUI can handle the operation.



DAQ Unit Interface

RS232 Communication Channel

COM5

Recieved Data
#w?#w?*#w?*#w?*#w?*#w?*#w?*#w?*#w?
#w?

Input and Noise Signal Characteristics

Input Signal Parameters

● Pulse Sequence
○ Bernoulli Sequence

Input Signal "ON" Amplitude: 150
Input Signal "OFF" Amplitude: 120

Noise Parameters

○ Gaussian Nosie
● Uniform Noise

Noise Mean	Noise Variance	Calculated Mean	Calculated Variance
0	50	0	202.2719688415

DSO2024A DAQ

Time Scale of Osc.: 0.0002

Correlation between CH1 and CH2: 0.636439980099388

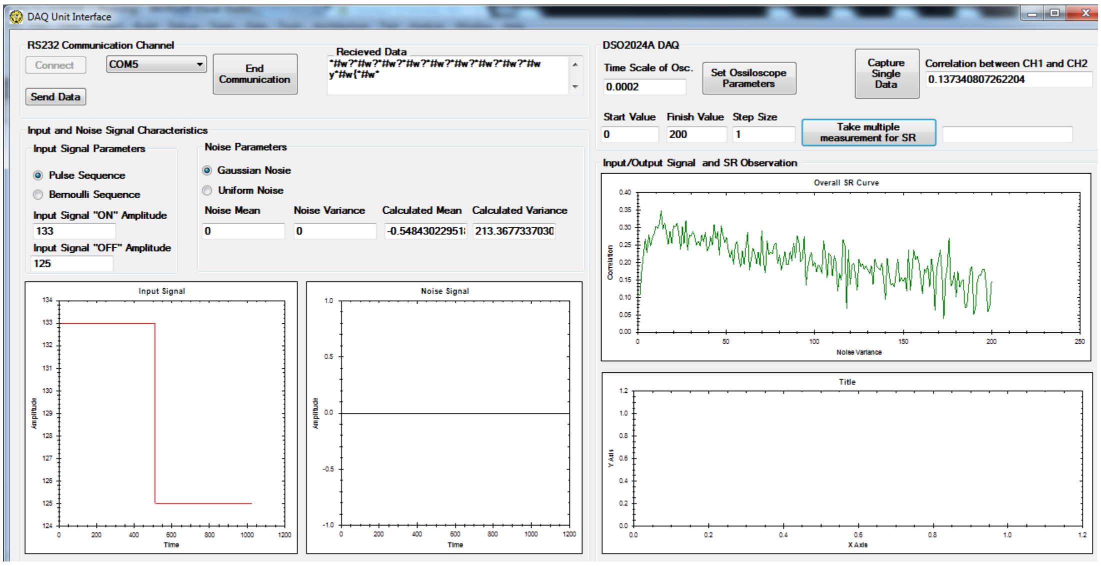
Start Value | Finish Value | Step Size

Input/Output Signal and SR Observation

Figure 36 Subthreshold signal is applied with noise signal

Moreover, a simple experiment is performed in the case of a noise presence. As seen in the Fig. 36, a uniform noise signal with variance of 50 is generated and added to subthreshold input signal. GUI shows the resulting wave form; noise presence in the pulse signal is half diminished since “OFF” state of the pulse signal is set to 120, it couldn’t pass the threshold even noise added to the signal.

Above experiments are about capturing single data sets while the following text is going to state the results of consecutive data capturing in order to observe SR effect. After many trials, two of the results seem to be presentable. One of the experiments includes a subthreshold pulse input signal and zero mean Gaussian noise is used to reach objective. Variance of zero mean Gaussian noise is changed from 0 to 200 with incremental step size of 1. At each step measurement is performed 50 times and their average is taken. Resulting SR curve is depicted in Fig. 37. Measurement time is selected as 50 after some experience on the experiment setup, when more repetition time is chosen computer enters a time out error and locks the oscilloscope. This drawback is avoided to achieve smoother curve.



COM5

Recieved Data:

```
*#w?*#w?*#w?*#w?*#w?*#w?*#w?*#w?*#w
y*#w{*#w*
```

Time Scale of Osc.: 0.0002

Correlation between CH1 and CH2: 0.137340807262204

Start Value: 0 Finish Value: 200 Step Size: 1

● Pulse Sequence
○ Bernoulli Sequence

Input Signal "ON" Amplitude: 133

Input Signal "OFF" Amplitude: 125

● Gaussian Nosie
○ Uniform Noise

Noise Mean	Noise Variance	Calculated Mean	Calculated Variance
0	0	-0.54843022951	213.3677337030

Figure 37 SR effect on a FET with pulse input and zero mean Gaussian noise

For better visualization, SR data is used in MATLAB and plotted in Fig. 38.

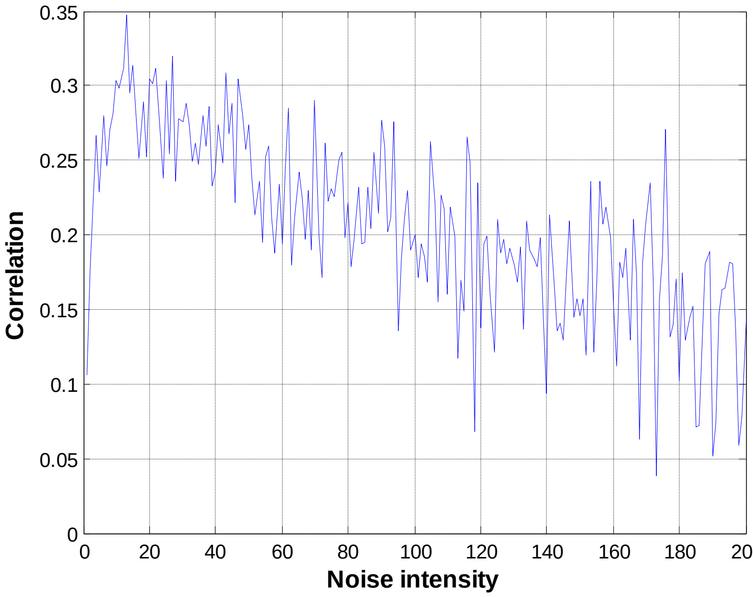
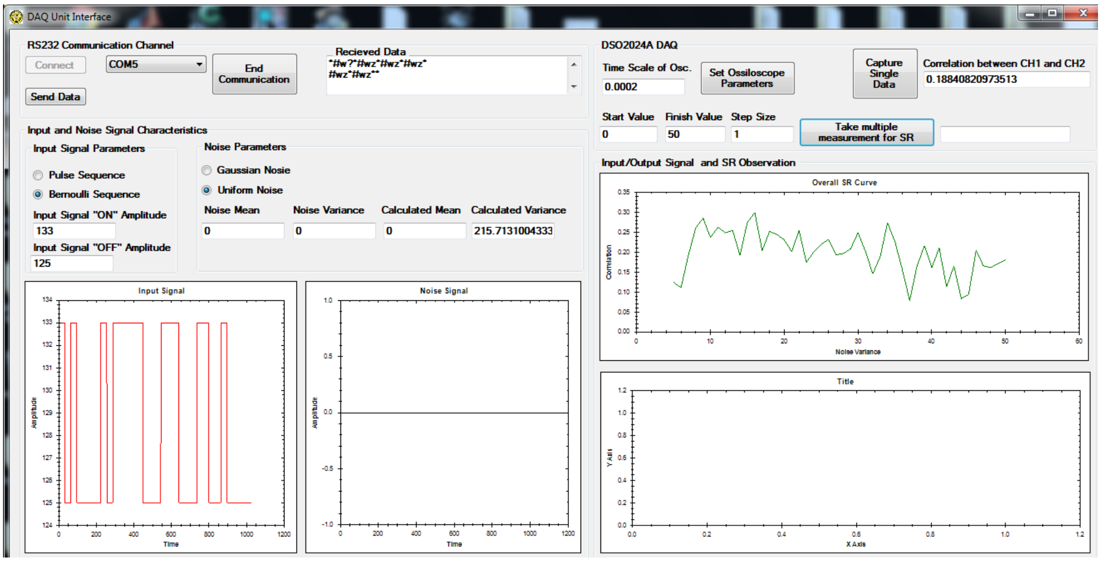


Figure 38 SR effect on a FET with pulse input and zero mean Gaussian noise

Second experiment is performed by using Bernoulli input sequence of 32 bits and uniform noise distribution. Variance of zero mean uniform noise is changed from 0 to 50 with incremental step size of 1. Measurement is done 50 times at each step and their average is taken. Resulting SR curve is presented in Fig. 39.



DAQ Unit Interface

RS232 Communication Channel: COM5

Recieved Data: *#w?*#wz*#wz*#wz* #wz*#wz**

Input and Noise Signal Characteristics

Input Signal Parameters: ○ Pulse Sequence ● Bernoulli Sequence

Input Signal "ON" Amplitude: 133

Input Signal "OFF" Amplitude: 125

Noise Parameters: ○ Gaussian Nosie ● Uniform Noise

Noise Mean: 0 | Noise Variance: 0 | Calculated Mean: 0 | Calculated Variance: 215.7131004333

DSO2024A DAQ

Time Scale of Osc.: 0.0002

Correlation between CH1 and CH2: 0.18840820973513

Start Value: 0 | Finish Value: 50 | Step Size: 1

Input/Output Signal and SR Observation

Figure 39 SR effect on a FET with Bernoulli input and zero mean uniform noise

As a conclusion, SR phenomenon is observed in a nonlinear threshold device by using a set of hardware and software components. In order to achieve given task an embedded system is developed using VHDL. In the software part of the study mainly C# is used, MATLAB is used as supplementary platform to verify the functionality of some scripts that is developed by C#. Moreover, synchronization between software and hardware devices is completed flawlessly. At the end, operation becomes successful in order to achieve thesis' goals.

CHAPTER 4

4. CONCLUSION

Noise is accepted as a degrading factor for the performance of electronic systems and it is usually tried to be eliminated from electronic systems. Through a century of electronics history, techniques to suppress noise presence have been developed. On the other hand in past few decades, it is observed that noise can play a constructive role on the performance of nonlinear threshold systems.

A subthreshold input signal can be enhanced by noise such that the resulting signal can reach and pass the threshold of a nonlinear system. Benefits of noise in signal detection have application in many different areas, such as in climate changes, in biology, in image processing and in electronic devices.

In this thesis, using the SR phenomenon noise is used to improve weak signal detection on a nonlinear threshold semiconductor device selected as FET for our study. The work accomplished in this thesis consists of two parts. In the first part (Chapter-2) a system training based model of SR is developed. In the second part (Chapter-3) the practical application of SR concept on FET using FPGA platform is demonstrated.

Main focus of Chapter-2 is developing a model that finds optimal noise intensity for a given input signal and threshold value of nonlinear system. System training based modeling was accomplished on Matlab via step by step operation. First step involves obtaining one SR curve for a constant input voltage level and for a constant threshold level. Second step consists of obtaining several SR curves for consecutive input voltage level and for a constant threshold level. Last step is about obtaining several SR curves for consecutive input voltage levels and consecutive threshold levels.

After all steps were completed, simulation SR datasets were processed to achieve a set of equations. Resulting calculations were compared with further simulation data and feasibility of the presented model was displayed.

In Chapter-3, experimental studies were taken place in order to observe SR effect. Experimental setup was constructed using a GUI developed by C#, FPGA, external analog circuits and oscilloscopes. Among these components GUI was the main part responsible for signal processing operations, while FPGA was used to control signal generation. Additionally, an oscilloscope was used to convert analog signal to digital one sent to GUI in order to observe signals and SR effect on GUI. Overall, SR was observed for subthreshold input signals for pulse and Bernoulli sequences with both Gaussian and Uniform noise signals. Future studies can focus on providing a mathematical derivation for the training based SR system studied in Chapter-2.

REFERENCES

1. **Benzi R.,Sutera A., Vulpiani A., (1981)**, *The Mechanism of Stochastic Resonance*, J.Phys. A: Math. Gen., vol. 14, pp. 453-457.

2. **Wang J., Xiao Q., Li X.,(2009),**"*The High-Frequency Weak Signal Detection Based on Stochastic Resonance"*, 2009 International Conference on Test and Measurement, Shenyang, China.

3. **McDonnell M. D., Abbott D.,** (2009), "*What is Stochastic Resonance? Definitions, Misconceptions, Debates and its Relevance to Biology",* PLoS Computational Biology, vol. 5, no 5.

4. **Moss F., Ward L. M., Sannita W. G.,(2004),** "*Stochastic Resonance and Sensory Information Processing: a Tutorial and Review of Application"*, International Federation of Clinical Neurophysiology, vol. 115, pp. 267-281.

5. **Samardak A., et. al., (2009),** "*Noise-Controlled Signal Transmission in a Multithread Semiconductor Neuron"*, Physical Review Letters, vol. 102, no 226802.

6. **Imai Y., et. al., (2014),***"Detection of Weak Biological Signal Utilizing Stochastic Resonance in a GaAs-based Nanowire FET and Its Parallel Summing Network",* Jpn. J. Appl. Phys., vol. 53.

7. **Lyttle D., (2008),***"Stochastic Resonance in Neurobiology"*,.

8. **Douglass J. K., et. al., (1993),** *"Noise Enhancement of Information Transfer in Crayfish Mechanoreceptors by Stochastic Resonance"*, Nature, vol. 365, pp. 337-340.

9. **Russell D. F., Wilkens L. A., Moss F., (1999),** *"Use of Behavioural Stochastic Resonance by Paddle Fish for Feeding",* Nature, vol. 402, pp. 291-294.

10. **Rallabandi V. P. S., Roy P. K., (2010),** *"Magnetic Resonance Image Enhancement Using Stochastic Resonance in Fourier Domain",* Magnetic Resonance Imaging, vol. 28, pp. 1361-1373.

11. **Ryu C., Kong S. G., Kim H., (2011),** *"Enhancement of Feature Extraction for Low-Quality Fingerprint Images Using Stochastic Resonance"*, Pattern Recognition Letters, vol. 32, pp. 107-113.

12. **Arlı. A. Ç., Bozkurt İ., Yengel E., (2015),** *"Improving Circle Detection Performance in Subthreshold Region by Using Stochastic Resonance",* 8. Mühendislik ve Teknoloji Sempozyumu Kitapçığı, pp. 173-176.

13. **Kasai S., Asai T., (2008),** *"Stochastic Resonance in Schottky Wrap Gate-Controlled GaAs Nanowire Field Effect Transistors and Their Networks"*, Applied Physics Express, vol. 1, no 8.

14. **Fujiwara A., Nishiguchi K., (2012**), *"Detecting Signals Buried in Noise via Nanowire Transistors Using Stochastic Resonance",* Applied Physics Letters, vol. 101, no 193108.

15. **Lee I. Y., et. al., (2003),** *"Nanosignal Processing: Stochastic Resonance in Carbon Nanotubes That Detect Subthreshold Signals"*, Nano Letters, vol. 3, no 12, pp. 1683-1686.

16. **Lee I. Y., et. al., (2006),***"Noise-Enhanced Detection of Subthreshold Signals with Carbon Nanotubes",*IEEE Transactions on Nanotechnology, vol. 5, no 6, pp. 613-627.

17. **Hakamata Y., et. al., (2010),***"Enhancement of Weak-signal Response Based on Stochastic Resonance in Carbon Nanotube Field-Effect Transistors"*, Journal of Applied Physics, vol. 108, no 104313.

18. **Kawahara T., et. al., (2011),** *"Gate Voltage Control of Stochastic Resonance in Carbon Nanotube Field Effect Transistors",*21st International Conference on Noise and Fluctuations, Toronto, Canada.

19. **Hakamata Y., et. al., (2011),***"Robust Noise Characteristics in Carbon Nanotube Transistors Based on Stochastic Resonance and Their Summing Networks",* Japanese Journal of Applied Physics, vol. 50.

20. **Hakamata Y., et. al., (2011),***"External-Noise-Induced Small-Signal Detection with Solution-Gated Carbon Nanotube Transistor",* Applied Physics Express, vol. 4, no 045102.

21. **Ando B., Graziani S., (2001),***"Adding Noise to Improve Measurement"*, IEEE Instrumentation & Measurement Magazine, vol. 1094, pp. 24-31.

22. **Ueda M., (2010),***"Improvement of Signal-to-Noise Raito by Stochastic Resonance in Sigmoid Function Threshold Systems, Demonstrated Using a CMOS inverter",*Physica A, vol. 389, pp. 1978-1985.

23. **Kanki T., et. al., (2010)** , "*Noise-Driven Signal Transmission Using Nonlinearity of VO2 Thin Films"*, Applied Physics Letters, vol. 96, no 242108.

24. **Box G. E. P., Muller M. E., (1958)**, *"A Note on the Generation of Random Normal Deviates."* Ann. Math. Stat., vol. 29, pp. 610-611.

25. *http://www.digilentinc.com/Products/Detail.cfm?NavPath=2,400,897&Prod =NEXYS3* (Data Download Date: 22.05.2015)

26. **Brown S., Vranesic Z. (2008),***"Fundamentals of Digital Logic with VHDL Design"*, McGraw-Hill Education, pp. 56-65.

27. http://www.ti.com/lit/ds/symlink/dac0800.pdf (Data Download Date: 01.08.2015)

28. http://www.vishay.com/docs/70209/70209.pdf (Data Download Date: 01.08.2015)

APPENDICESA

CURRICULUM VITAE

PERSONAL INFORMATION

Surname, Name: Arlı, Ahmet Çağrı

Nationality: Turkish (TC)

Date and Place of Birth: 03 October 1988, Ankara

Marital Status: Single

Phone: +90 506 615 54 93

e-mail: cagriarli@cankaya.edu.tr

EDUCATION

Degree	**Institution**	**Year of Graduation**
M.Sc	Çankaya University Electronic and Communication Engineering	August, 2015
B.Sc	Çankaya University Electronic and Communication Engineering	June, 2011
High School	Mehmet Emin Resulzade Anadolu Lisesi, Ankara	June, 2006

FOREIGN LANGUAGE

Advanced English

PUBLICATION

ArlıA. Ç., KaraaslanF.,Çil C. Z., Büke G. C. , Güvenç Z. B. , Dede M., Ertuğ A. M.,“Düşük Maliyetli Taramalı Tünelleme Mikroskobu”, Çankaya Üniversitesi 4. Mühendislik ve Teknoloji Sempozyumu, Ankara, Türkiye, Apr. 2011. Çankaya Üniversitesi 5. Mühendislik ve Teknoloji Sempozyumu Kitapçığı, 130-141, 28-29 Nisan 2011.

Arlı A. Ç., Bozkurt İ., Yengel E.,“Improving Circle Detection Performance in Subthreshold Region bu Using Stochastic Resonance”, Çankaya Üniversitesi 8.Mühendislik ve Teknoloji Sempozyumu, Ankara, Türkiye, May. 2015. Çankaya Üniversitesi 8. Mühendislik ve Teknoloji Sempozyumu Kitapçığı, 173-176, 14-15 Mayıs 2015.