2017-06-20

One-to-Cloud One-Time Pad Data Encryption: Introducing Virtual Prototyping with PSpice

Paul Tobin
Technological University Dublin, paul.tobin@tudublin.ie

Lee Tobin
University College Dublin, lee.tobin@ucdconnect.ie

Roberto Gandia Blanquer DR
Flowcode, Roberto.Gandia@flowcad.de

Michael McKeever
Technological University Dublin, mick.mckeever@tudublin.ie

Jonathan Blackledge
Technological University Dublin, jonathan.blackledge@tudublin.ie

One-to-Cloud One-Time Pad Data encryption: Introducing Virtual Prototyping with PSpice

P. Tobin, L. Tobin, R. Gandia Blanquer, M. McKeever, J. Blackledge

School of Electrical and Electronic Engineering
Dublin Institute of Technology, Ireland

E-mail: paul.tobin@dit.ie lee.tobin@ucdconnect.ie
Roberto.Gandia@flowcad.de mick.mckeever@dit.ie
Jonathan.blackledge59@gmail.com

Abstract In this paper, we examine the design and application of a one-time pad encryption system for protecting data stored in the Cloud. Personalising security using a one-time pad generator at the client-end protects data from break-ins, side-channel attacks and backdoors in public encryption algorithms. The one-time pad binary sequences were obtained from modified analogue chaos oscillators initiated by noise and encoded client data locally. Specific "one-to-Cloud" storage applications returned control back to the end user but without the key distribution problem normally associated with one-time pad encryption. Development of the prototype was aided by "Virtual Prototyping" in the latest version of Cadence OrCAD PSpice©. This addition allows the prototype simulation schematic to be connected to an actual microcontroller in real time using device model interfacing for bi-directional communication.

Keywords —**One-time pads, chaos, noise, one-to-Cloud, PSpice, virtual prototyping, device model interfacing, microcontroller.**

I Introduction: Cloud computing

Lack of confidence in Cloud security led to the development of a one-time pad (OTP) random binary generator for encoding data locally in the Cloud. This encoder provides control and security for "one-to-Cloud" storage applications which have no OTP key distribution problems. Personalising encryption, in addition to public encryption algorithms such as the Advanced Encryption Standard (AES), provides a layer of extra protection. However, many Cloud security issues are mainly side-channel and other less sophisticated attacks which justify this approach, despite AES encryption being present. If the rules for OTP encoding are obeyed then decoding the encrypted data is impossible without the original OTP [1].

The prototype uses standard chaos oscillators modified with analogue delays and initialised with noise from an FM data receiver. These novel additions increased the entropy of the final OTP. Cadence OrCAD PSpice© v 17.2 now allows exporting data to MATLAB® for further processing. Also, virtual prototyping (VP) connects the simulation schematic to a microcontroller in real time using device model interface (DMI). This interface uses C/C++ code to establish bi-directional communication between the Arduino microcontroller and the schematic. The OTP from the microcontroller is exclusively OR-gated with the data in a Javascript application which adds additional entropy to the encoded output.

a) Structure of the Paper

In Section I, we explain how a OTP system protected conversations between heads of state during WWII and in the sixties. Applications for the OTP prototype generator explain how it protects data stored in the Cloud but with no key distribution problems. Section II outlines the design of the

analogue chaos oscillator and threshold circuits. In Section III, we introduce a JavaScript application for processing the OTP and data. Section IV details how the encoder was tested and simulated using PSpice virtual prototyping and conclusions and future work are in Section V.

b) A brief history of the One-Time Pad

In previous papers [2], [3], we explained the rationale for using OTP encryption to secure data stored in the Cloud. Cloud traffic and security issues will grow exponentially [4], [5] and we argue why personalising encryption is necessary before using systems such as OneDrive, Google Drive, Dropbox, etc.

Churchill and Roosevelt used a OTP system called SIGSALY in WWII to mask conversations about war strategy [6], [7], [8], and a 'Hotline' OTP between Kennedy and Khrushchev during the sixties Cuban crisis. These systems which protected World peace were large (SIGSALY weighed 55 tonnes) and had key distribution problems. For example, SIGSALY recorded the OTP key on vinyl and was flown across the Atlantic, hence the key distribution problem. These factors effectively consigned the OTP to history - until now.

c) OTP Prototype Overview

The block diagram in Figure 1 presents an overview of the encoder showing chaos sources initialised by noise. The OTP and data are exclusively OR-gated (modulo two arithmetic) in a JavaScript application and the encoded data stored in the Cloud. Here, the client encrypts data at one location and then carries the OTP (the "Sneakernet" method) to decode data from the Cloud at another location. For example, a barrister bringing legal case documents in ring binder files to court may replace them with a OTP and an Android device for storing the downloaded files. Similarly, medical scans transported between the doctor's surgery and the hospital, are replaced by the OTP. Figure 2 demonstrates how a barrister in court may download encoded case documentation to an Android device and decode it using the OTP key. This method creates a secure paperless court case [9] and also gives excellent document search capabilities. Losing the OTP in transit will not create security issues, and hence client confidentiality is protected.

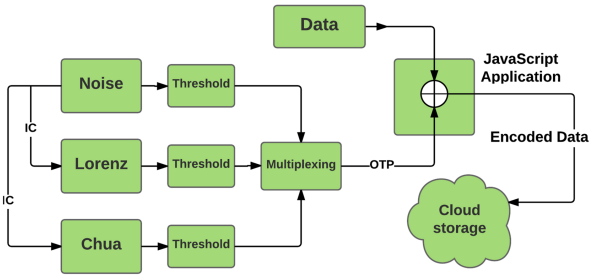

Fig. 1: A OTP encryption system.

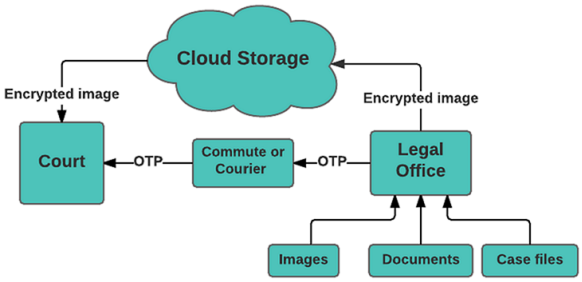

Fig. 2: A "one-to-Cloud" encoder legal application.

II Chaos Cryptography

Claude Shannon, in his 1945 and 1949 papers [10] [11], discussed chaos stretching and folding mechanisms and compared topological (ergodic) mixing to confusion in cryptography, and diffusion to sensitivity to initial conditions (SIC). Chaos oscillators may be implemented using analogue or digital systems, however, binary sequences from digital chaos oscillators will have a finite cycle length and hence will not be truly random. For this reason, the team chose analogue chaos oscillators which produce signals with an infinite number of states and the prototype may be classified as a true random number generator (TRNG) [12]. Noise from a detuned FM data receiver initialised analogue Lorenz and Chua chaos sources and when thresholded correctly, generate non-repeating cycle length random OTP sequences.

a) The Lorenz chaotic oscillator

A second-order non-linear differential equation modelled Edward Lorenz's chaotic oscillator published in a 1963 paper [14]. For electronic implementation, the second-order equation was replaced by three coupled first-order equations in (1).

$$\begin{aligned} x(t) &= -P\int_{t_0}^{t}\{x(t)-y(t)\}dt \\ y(t) &= -\int_{t_0}^{t}\{-Rx(t)+y(t)+x(t)z(t)\}dt \\ z(t) &= -\int_{t_0}^{t}\{Bz(t)-x(t)y(t)\}dt \end{aligned} \tag{1}$$

Lorenz used $B = 2.666$, *Prandtl* $P = 10$, $R = 28$, but slightly different values were determined experimentally which improved the OTP entropy. A brief history of chaos and electronic designs may be examined in [13].

b) Threshold circuit design

The OTP entropy was increased by choosing the two analogue thresholds as the fixed points (FP) shown in the 3-D Lorenz strange attractor in Figure 3. The path of $v(x)$ $v(y)$ and $v(z)$ circu-

late around these points of equilibrium in a random fashion and was the choice for generating the binary 'ones' and 'zeroes'. The attractor has

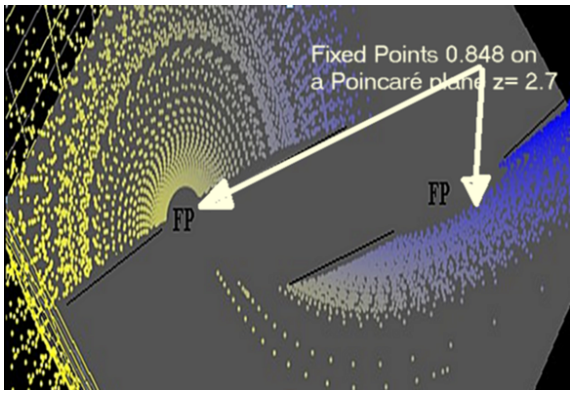

Fig. 3: The 3-D strange attractor and Poincaré section.

a Poincaré section cutting the vertical $v(z)$ axis through the FPs. We chose the threshold voltage levels to generate 'ones' and 'zeroes' as the FP x y coordinates at the lobe centres of the attractor. These were calculated by solving (1) for $x = y$ to yield:

$$FP_{1,2} = \{+\sqrt{B(P-1)}, -\sqrt{B(P-1)}, (R-1)\} \tag{2}$$

The FPs for the standard Lorenz parameters are ± 8.485 V, for z equal to 27 V, but scaling by ten for normal electronic signal amplitude, modifies the FPs to $FP_{1,2}$ to ± 0.8485 V, as shown in Figure 3.

c) The delayed Lorenz oscillator

A heuristic experimental approach improved the randomness of the prototype by adding an analogue delay to the polar z signal in the Lorenz chaos source. This modified the original equations as:

$$\begin{aligned} x(t) &= -10 \int_{t_0}^{t} \{x(t) - y(t)\} dt \\ y(t) &= -\int_{t_0}^{t} \{-28x(t) + y(t) + x(t)z(t-\tau)\} dt \\ z(t) &= -\int_{t_0}^{t} \{2.666z(t-\tau) - x(t)y(t)\} dt \end{aligned} \tag{3}$$

The idea to add a delay was inspired by the fact that chaotic maps such the logistic and Hénon [15], have more noise-like outputs and make better random number generators but are harder to implement electronically. The analogue delay used a Padé approximation technique rather than a sample and hold design, and the z-transform, defined in [16], expresses an analogue delay, τ, as:

$$z = e^{s\tau} = \frac{e^{s\tau/2}}{e^{-s\tau/2}} \approx \frac{1 + s\tau/2}{1 - s\tau/2} \tag{4}$$

The transfer function for the operational amplifier delay shown in Figure 4, is:

$$\frac{V_{out}}{V_{in}} = -\frac{sCdRd}{1 + sCdRd} \tag{5}$$

Comparing (4) to (5), yields a delay, $\tau = 0.5{*}C\mathrm{d}R\mathrm{d} = 0.5$ us.

d) The Chua chaotic oscillator

Three first-order coupled equations in (6) model the Chua analogue chaos oscillator. In 1983, Leon Chua created a new chaos oscillator system trying to prove the Lorenz oscillator was chaotic. The standard Chua oscillator consists of a parallel-tuned circuit and connected across it is a 'Chua diode', a segmented negative resistance formed from operational amplifiers. A comprehensive paper on this is [17]. However, we used a different approach using AD633 four-quadrant multipliers for the nonlinear cubic term in (6) [18]. A Padé delay was also added to the y signal line.

$$\begin{aligned} x(t) &= -\int_{t_0}^{t} \{-2.7x(t) - 2.4y(t) + 3.95x(t)^3\} dt \\ y(t) &= -\int_{t_0}^{t} \{4.167x(t) - y(t-\tau) - 7.083z(t)\} dt \\ z(t) &= -\int_{t_0}^{t} \{2.099y(t)\} dt \end{aligned} \tag{6}$$

e) Generating the One-Time Pads

The OTP is generated by thresholding the chaos signals at the value of the two FPs. The threshold design converted the Lorenz bipolar x signal to polar form by adding 4 V DC using a potential divider. The LM339 comparator reference voltage is Vref = 1.24 V and another potential divider sets the threshold voltages to the FP values calculated as ± 0.848 V plus 4 V DC, or 3.15 V and 4.84 V. The 74121 monostable then converts the comparator set and reset pulses to constant widths and stores the OTP in a microcontroller shield. A similar threshold design process is used in the Chua oscillator.

III JavaScript application

Figure 5 is the JavaScript application [19] which performs modulo two arithmetic between the pixel array data from the bitmap magnetic resonance imaging (MRI) scan and the OTP. To see the effects of bias on the scanned encoded image, the application repeated parts of the OTP to pad it out to the same length as the MRI image. However, this was for demonstration purposes only as bias weakens the encryptor strength making it easier to decode the data. The encoded image in the bottom right pane shows bias as horizontal lines.

XORing the two uncorrelated data chaos binary streams produced dibit pairs processed in a von Neumann (vN) deskewing algorithm to remove any

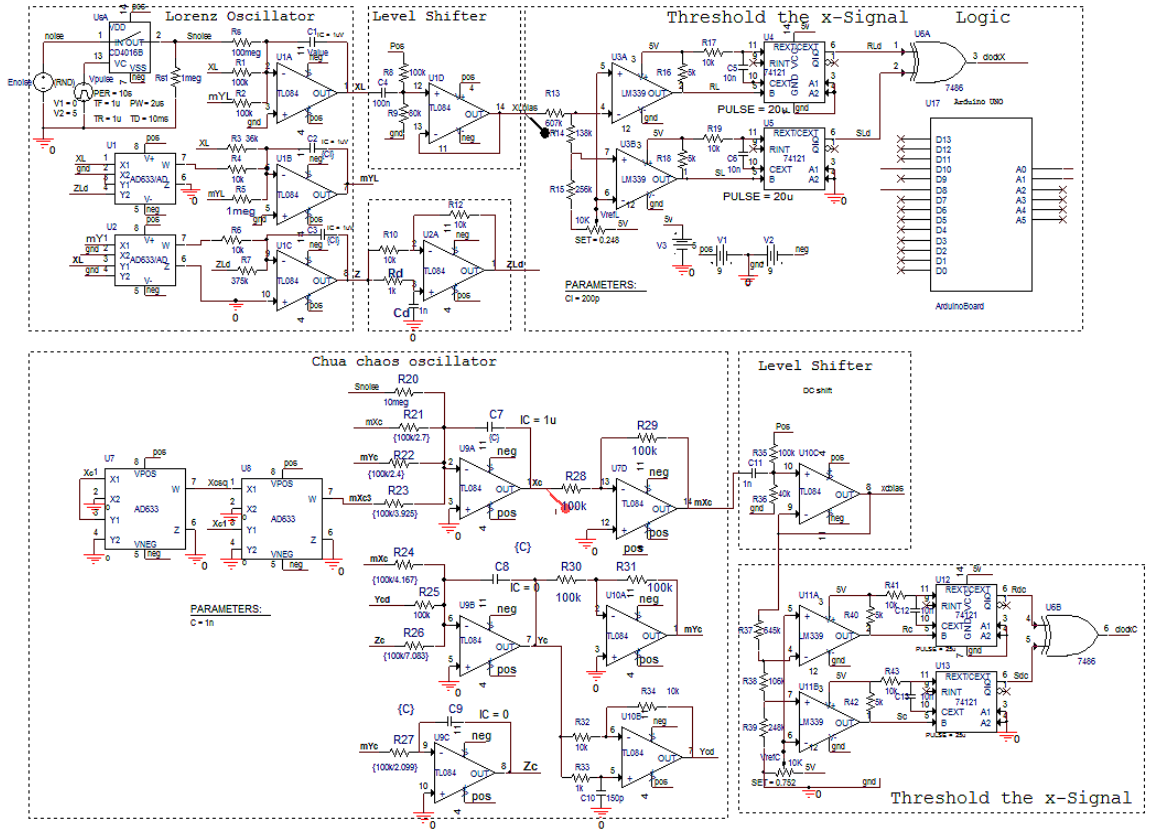

Fig. 4: Lorenz and Chua oscillator circuits.

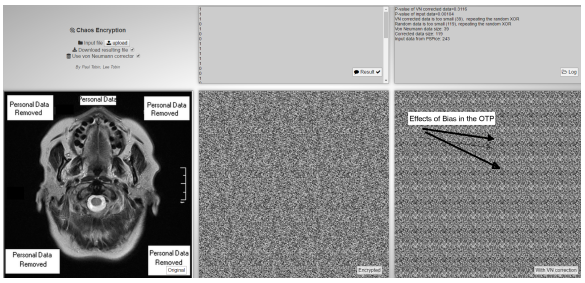

Fig. 5: JavaScript application: encrypting an MRI scan.

bias by rejecting 00 and 11 dibit pairs [20]. The middle pane is the encoded image with no vN applied, and as can be seen, no bias is present.

IV Virtual prototyping and testing

The National Institute of Standards and Technology (NIST) suite of fifteen statistical hypothesis tests evaluated the cryptographic strength of individual parts of the system [21]. However, there was insufficient time before paper submission to check the total system entropy. Each chaos source was tested separately and passed the NIST tests, so it is not unreasonable to assume the system entropy will be greater than the individual sources. Randomness tests were also carried out using new features in Cadence OrCAD PSpice© v 17.2. Figure 6 shows how we exported the OTP using a new menu item in the PSpice Probe menu. After

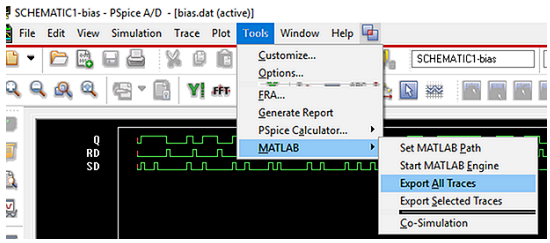

Fig. 6: Exporting the OTP data to MATLAB ®

selecting the simulated variables, a sub menu calls the following Matlab mfile:

```
>>>time=PSpiceData_1.Analysis.Sweep(1).
Digital_Traces(1).Data.Time
>>val=PSpiceData_1.Analysis.Sweep(1).
Digital_Traces(1).Data.Val
>>stairs(time,val)
>>axis([0,0.015,-0.2,1.2]);
```

Figure 7 shows the non-return-to-zero (NRZ) OTP exported from PSpice and plotted in Matlab. The major advantage of this technique is that only transitions and not PSpice simulation points, are recorded, otherwise the OTP is incorrect, very large and will give incorrect NIST results. Figure 8 shows the Matlab® results from the OTP

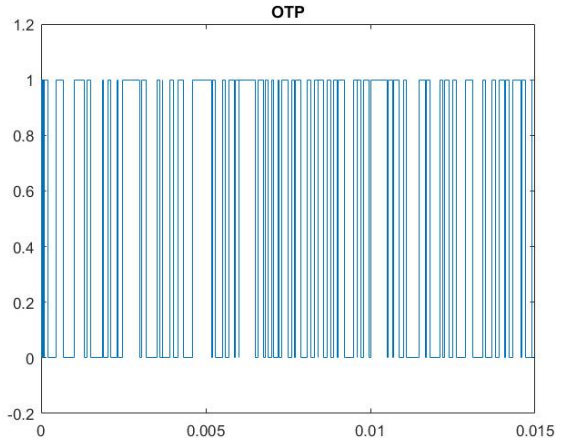

Fig. 7: The OTP exported to MATLAB®

data exclusively Or-gated with the bitmap from the Lena image. Histogram and power spectral density (PSD) plots performed in Matlab in addition to the NIST randomness tests. The bottom pane shows the recovered images. Virtual

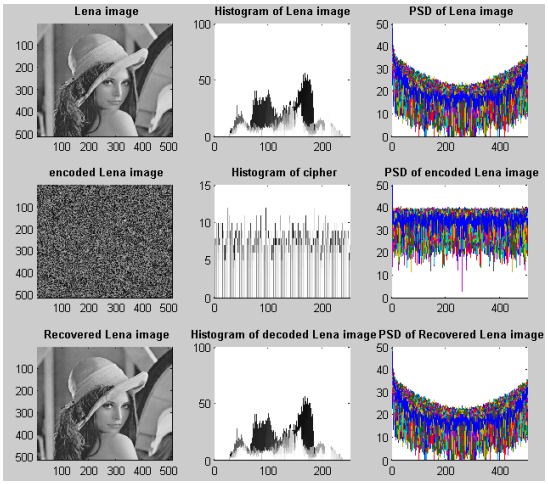

Fig. 8: (a) The Lena image (b) Histogram of unencoded image (c) Power Spectral Density of unencoded image.

Prototyping in V17.2 Cadence OrCAD PSpice© allows bi-directional communications in real time between a simulation schematic and a microcontroller. Device Model Interfacing (DMI) allows the designer to connect in real time to microcontrollers such as the Arduino by defining a system environment using C/C++, as explained in application notes by one of the authors [22] [23]. Closing the loop between simulation and real-time measurements broadens the scope of simulation and decreases the time from initial designs to final production prototype. What this means is that we may investigate in real time, the effects of parameter variation in the simulation circuit.

a) Averaging the x signal

Figure 9 shows the output from a novel method for evaluating the "equality' of the ones and zeroes in the random bit stream. The average value

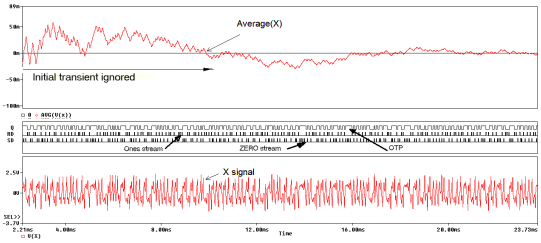

Fig. 9: Averaging the x signal.

of the x signal should oscillate about zero for best entropy results. We disregard the transient part of the OTP because it is positive for a greater amount of the time and hence biased. This measure of bias is an approximate method but useful, nevertheless. This plot displays the two digital signals which form the OTP. The x in the bottom pane is now different to the normal Lorenz x signal because of the added Padé delay.

V Conclusion

Cloud computing is suffering from poor security problems that are growing almost exponentially. Some cloud service providers do not encrypt client data while others encrypt using a weaker 128-bit AES encryption, such as in iCloud. However, even with secure algorithms, Cloud sites are still successfully attacked, and our solution encrypts data at the client end that can be decoded only with the original OTP. Creating OTPs for encoding data is not new, but our prototype incorporates some novel aspects such as an analogue delay in the chaos sources to increase the overall entropy. Furthermore, the chaos sources were initialising using a novel noise initialising technique which classifies the prototype generator as a true source of randomness. Specific one-to-Cloud applications given in the paper ensures there are no key distribution problems. A JavaScript application processes the data and the OTP and adds a vN deskewing algorithm to increase the cipher entropy.

Virtual prototyping and device model interfacing is a new feature in PSpice and allowed the simulation schematic to communicate with real hardware in real time. Another new feature exports data from PSpice to MATLAB® and provides an additional tool for analysing data not possible in PSpice. Personalising the encryption process locally, ultimately gives greater control and security to the client who wishes to encrypt sensitive data but retains the usefulness of Cloud storage. Future research will investigate applications for the

device between two people, but the key distribution problem has to be solved.

Acknowledgements

The authors are grateful to Professor Michael Conlon and Dr Marek Rebow, Dublin Institute of Technology, for arranging the collaborative research programme.

References

[1] S. Ergn, U. Gler and K. Asada, "A High Speed IC Truly Random Number Generator Based on Chaotic Sampling of Regular Waveform", IEICE Transactions on Fundamentals of Electronics, Communications and Computer Sciences, Vol. E94 A, 1, 2011, pp. 180-190.

[2] P. Tobin, L. Tobin, M. McKeever and J. Blackledge, "Chaos-based Cryptography for Cloud Computing", 27th ISSC conference Ulster University, Londonderry, June 21-22, doi: 10.1109, 2016, pp. 1-6.

[3] P. Tobin, L. Tobin, M. McKeever and J. Blackledge, "On the Development of a One-Time Pad Generator for Personalising Cloud Security", Athens, Conference on CLOUD COMPUTING, Feb 19-23, 2017.

[4] D. Shumow, N. Ferguson "On the possibility of a back door in the NIST SP800-90 Dual Ec Prng", CRYPTO Rump Session, August 2007. [Online]. Available: `http://rump2007.cr.yp.to/15-shumow.pdf`.

[5] The New York Times, " Secret Documents Reveal N.S.A. Campaign Against Encryption", 2013.

[6] W. Bennett, "Secret Telephony as a Historical Example of Spread-Spectrum Communication", IEEE TRANSACTIONS ON COMMUNICATION COM-31.1 (1983): 98-104.

[7] P. Weadon,"Sigsaly Story", National Security Agency / Central Security Service. National Security Agency, 15 Jan 2009.

[8] J. V. Boone, and R. R. Peterson."Sigsaly - The Start of the Digital Revolution", National Security Agency / Central Security Service. National Security Agency, 15 Jan 2009. Web. 19 Oct 2010.

[9] [Online].Available: `http://www.ecourt.ie/`

[10] C.E. Shannon, "A mathematical theory of cryptography", Memorandum MM, 45, 1945, pp.110-02.

[11] C. E. Shannon, "Communication Theory of Secrecy Systems", Bell Technical Journal, vol.28-4, 1949, pp. 656 715.

[12] E. Barker, E. Kelsey K, "Recommendation for the Entropy Sources Used for Random Bit Generation (draft)", NIST SP800-90B, August 2012.

[13] [online] Available: `http://www.pspice.com/blog`

[14] E. Lorenz, "Chaos and Strange Attractors: The Lorenz Equations", pp. 532-538, 1963.

[15] Zaid, Osama M. Abu, et al.,"A Proposed Encryption Scheme based on Henon Chaotic System (PESH) for Image Security", International Journal of Computer Applications 61.5, 2013.

[16] P. Tobin, "PSpice for Digital Signal Processing, `http://www.morganclaypool.com/action/doSearch?AllField=tobin&x=0&y=0&SeriesKey=` ISBN:1598291629, 2007.

[17] P. Kennedy, "Genealogy of Chuas Circuit In Chaos, CNN, Memristors and Beyond: A Festschrift for Leon Chua" pp. 3-24, 2013

[18] R. Kili, "A practical guide for studying Chua's circuits", Vol. 71. World Scientific, 2010.

[19] [online] Available: `http://jork.byethost7.com/chaosencrypt/`.

[20] J. von Neumann, "Various techniques used in connection with random digits", Applied Math Series, 12:3638, 1951.

[21] A. Rukhin, et al., "A statistical test suite for the validation of random number generators and pseudo-random number generators for cryptographic applications", [online] Available: `http://csrc.nist.gov/groups/ST/toolkit/rng/`, 2010.

[22] [online]. Available: `http://www.flowcad.de/Application_Notes.htm`.

[23] [online]. Available: `http://www.pspice.com/resources/video-library/pspice-simulation-arduino-driven-sensors`