

Somaiya Vidyavihar University

Truly Random

Submitted at the end of semester VI in partial fulfillment of requirements of

Bachelors in Technology in Computer Engineering

by

Meet Gala

Roll No: 16010121051

Pargat Singh Dhanjal

Roll No: 16010121045

Vishrut Deshmukh

Roll No: 16010121043

Guide



Dr. Bhakti Palkar

Department of Computer Engineering

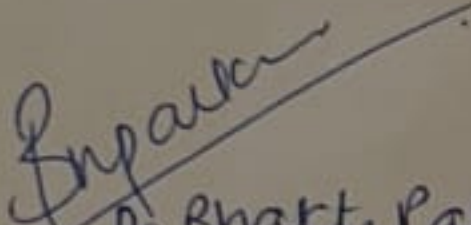
K. J. Somaiya College of Engineering, Mumbai-77

Batch 2021 -2025

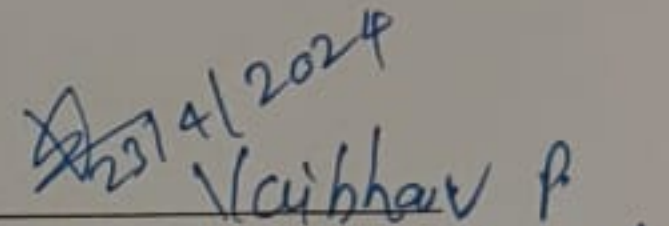


Certificate

This is to certify that the TY Mini Project report entitled **Truly Random** submitted by **Pargat Singh Dhanjal (16010121045)**, **Vishrut Deshmukh (16010121043)** and **Meet Gala (16010121051)** at the end of semester VI of TY B. Tech is a bona fide record for partial fulfillment of requirements for the degree in **Computer Engineering** of Somaiya Vidyavihar University


Dr. Bhakti Palcar

Guide


Vaibhav P. Vasani

Examiner

Date: 23/04/2024

Place: Mumbai-77



Certificate of Approval of Examiners

We certify that this Mini Project report entitled Truly Random is bona fide record of Mini project work done by Meet Gala during semester VI.

This Mini project work is submitted at the end of semester VI in partial fulfilment of requirements for the degree of Bachelors in Technology in Computer Engineering of Somaiya University

Meet Gala
23/4/24

Internal Examiner 1

Internal Examiner 2

Date: 23/4/24

Place: Mumbai-77



Declaration

We declare that this written report submission represents the work done based on our and / or others' ideas with adequately cited and referenced the original source. We also declare that we have adhered to all principles of intellectual property, academic honesty and integrity as we have not misinterpreted or fabricated or falsified any idea/data/fact/source/original work/ matter in my submission.

We understand that any violation of the above will be cause for disciplinary action by the college and may evoke the penal action from the sources which have not been properly cited or from whom proper permission is not sought.

 _____ Signature of the Student 16010121045 _____ Roll No.	 _____ Signature of the Student 16010121043 _____ Roll No.
 _____ Signature of the Student 16010121051 _____ Roll No.	_____ Signature of the Student _____ Roll No.
_____ Signature of the Student _____ Roll No.	_____ Signature of the Student _____ Roll No.

Date:

Place: Mumbai-77

Declaration

We declare that this written report submission represents the work done based on our and / or others' ideas with adequately cited and referenced the original source. We also declare that we have adhered to all principles of intellectual property, academic honesty and integrity as we have not misinterpreted or fabricated or falsified any idea/data/fact/source/original work/ matter in my submission.

We understand that any violation of the above will be cause for disciplinary action by the college and may evoke the penal action from the sources which have not been properly cited or from whom proper permission is not sought.

<p>_____ Signature of the Student</p> <p>_____ Roll No.</p>	<p>_____ Signature of the Student</p> <p>_____ Roll No.</p>
<p>_____ Signature of the Student</p> <p>_____ Roll No.</p>	<p>_____ Signature of the Student</p> <p>_____ Roll No.</p>
<p>_____ Signature of the Student</p> <p>_____ Roll No.</p>	<p>_____ Signature of the Student</p> <p>_____ Roll No.</p>

Date:

Place: Mumbai-77

Abstract

This report proposes a novel **True Random Number Generator (TRNG)** design that leverages the intricate dynamics of a specially designed compound pendulum. Traditional TRNG methods often rely on specialized hardware or complex physical processes. This design offers a practical alternative by utilizing the inherent randomness of the pendulum's motion as a source of high-quality entropy for random number generation.

The proposed TRNG employs readily available components, including a specially designed compound pendulum and a high-precision motion capture system. The pendulum's complex motion, characterized by nonlinear behavior, multiple degrees of freedom, and external influences, serves as a robust entropy source. The motion capture system tracks the movement of the pendulum in real-time, capturing position and velocity data that feeds into a dedicated algorithm for random number generation. This algorithm extracts randomness from the variations in the captured data and utilizes a one-way hashing function to further enhance the unpredictability of the output sequence.

A comprehensive evaluation plan will be implemented to assess the quality and randomness of the generated numbers using established statistical tests. The TRNG will be evaluated for properties such as uniformity, independence, and entropy. Additionally, security, performance, and practicality will be considered to assess the overall effectiveness of the design.

This novel TRNG design offers a promising approach for generating truly random numbers in various computing environments. The utilization of readily available components and a well-defined data processing algorithm presents a practical and efficient solution for applications requiring high-quality randomness.

Table of Contents

Sr No.	Topic	Pg No
1	Introduction	5
2	Literature Survey	11
3	Project Design	14
4	Implementation	21
5	Conclusion and Further Work	29
6	Bibliography	31

List of Figures

Sr No.	Figure Name
1	System Architecture
2	Project Timeline
3	Project Design
4	Component Diagram
5	Pendulum
6	Signal Wave of Pendulum
7	Histogram

List of Tables

Sr No.	Table Name
1.	Project Plan

Chapter 1

Introduction

This chapter presents an overview of our project, starting with a discussion of the motivation of our topic. It then outlines the specific problem statement that we will be working towards in our project, followed by its scope and objectives. Additionally, it talks the hardware and software requirements needed for both the development and deployment phases of our project.

1.1 Background/Motivation

The digital age is increasingly reliant on cryptography and security protocols to safeguard sensitive information. These protocols rely on unpredictable and statistically independent random numbers for tasks such as encryption key generation, secure communication protocols, and random sampling in simulations. While pseudorandom number generators (PRNGs) are widely used, their deterministic nature limits their suitability for security-sensitive applications. Even slight predictability in a PRNG output can be disastrous, potentially compromising encryption schemes and exposing vulnerabilities.

True Random Number Generators (TRNGs) address this limitation by extracting randomness from physical phenomena with inherent unpredictability. These physical processes exhibit microscopic fluctuations or chaotic behavior that translates into statistically unpredictable outputs. High-quality TRNGs are crucial for various security applications:

- **Cryptography:** Securely encrypting and decrypting confidential data relies on unpredictable keys generated by TRNGs. These keys ensure that only authorized parties can access sensitive information.
- **Digital Signatures:** Digitally signing documents or messages utilizes random numbers to create unique "fingerprints" that authenticate the sender and prevent tampering.
- **Secure Communication Protocols:** TRNGs play a vital role in secure communication protocols like virtual private networks (VPNs) by generating random session keys that encrypt data transmission.
- **Simulations:** Random number generation is essential for computer simulations in various fields like scientific research, finance, and game development. High-quality randomness ensures reliable and unbiased outcomes in these simulations.

1.2 Problem Statement

Existing TRNG designs based on physical phenomena often face limitations:

Complexity: Some methods, like photonics-based TRNGs, necessitate sophisticated optical setups and specialized equipment, making them impractical for widespread use.

Resource Requirements: Thermal noise-based TRNGs are readily available but may not provide sufficient entropy, especially for high-security applications.

Control Challenges: Chaotic systems, while promising, can be complex to design and control effectively.

There is a critical need for TRNG designs that are:

- **Practical and Easy to Implement:** Utilizing readily available components and minimizing complex setups for wider adoption.
- **High Quality and Efficient:** Generating random numbers with robust statistical properties and sufficient entropy to meet security requirements.
- **Cost-Effective:** Employing affordable components to keep development and deployment costs manageable.

1.3 Project Overview

This project proposes a novel TRNG design that leverages the intricate dynamics of a specially designed compound pendulum. Building upon a MATLAB Simulink simulation that explored the feasibility of this concept, the project aims to translate the simulated model into a practical hardware implementation.

1.4 Objectives

The primary objectives of this project are:

1. **Validation:** To validate the feasibility of the TRNG concept based on the compound pendulum's dynamics, as explored in the MATLAB Simulink simulation.
2. **Hardware Design and Construction:** To design and construct a physical compound pendulum based on the simulated model, incorporating features identified in the simulation to enhance the randomness of its motion.
3. **Motion Capture Integration:** To integrate a high-precision motion capture system to track the movement of the real-world pendulum in real-time.
4. **Data Processing Algorithm Development:** To develop a data processing algorithm that effectively extracts randomness from the captured motion data, building upon the insights gained from the simulation.
5. **Post-Processing with Hashing:** To implement a post-processing stage using a one-way hashing function to further enhance the unpredictability of the generated random numbers.
6. **Comprehensive Evaluation:** To comprehensively evaluate the randomness of the generated numbers using established statistical tests to ensure they meet the requirements for security-sensitive applications.

1.5 Hardware Considerations

The hardware design will be guided by the specifications established in the MATLAB Simulink model. Key components likely include:

Compound Pendulum: Constructed using materials that ensure stability, durability, and promote the desired dynamic behavior. Material choices might include metal rods, bearings, springs, and appropriately distributed weights. The design will be informed by the simulation results, considering factors like joint types, dimensions, spring properties, and potential damping mechanisms.

Motion Capture System: A high-precision motion capture system, similar to the one used in the simulation, will be employed to track the movement of the pendulum in real-time. This could involve strategically placed markers on the pendulum and cameras to capture its movement.

Data Acquisition Hardware: A data acquisition system will be necessary to capture the data from the motion capture system and transfer it to a computer for processing. This could involve a data acquisition card or other suitable interface hardware.

Throughout the hardware design process, considerations will be given to:

- **Cost-Effectiveness:** Utilizing readily available components whenever possible to maintain project affordability. Balancing cost with the need for high-precision components that ensure reliable data capture.
- **Practicality:** Designing a system that is easy to assemble, operate, and maintain.
- **Durability:** Ensuring the hardware can withstand repeated use and potential environmental factors. The design may incorporate features to minimize friction and wear, or utilize materials with good wear resistance.

1.6 Software Considerations

Programming Environment: The same programming environment used in the simulation (e.g., MATLAB, Python) will likely be used to develop the data processing algorithm. This leverages existing code and ensures compatibility with the simulation results.

Data Processing Algorithm: Based on the insights gained from the Simulink model, the data processing algorithm will be further refined to extract randomness from the captured real-world motion data. The algorithm might involve techniques like:

Signal filtering to remove noise or unwanted frequencies from the captured motion data.

Feature extraction to identify specific characteristics of the pendulum's motion that exhibit high entropy.

Statistical transformations to convert the extracted features into a sequence of random numbers.

Statistical Analysis Software: Software tools will be used to conduct statistical tests on the generated random numbers, evaluating their:

Uniformity: How evenly distributed the random numbers are across the possible values.

Independence: How statistically unrelated each random number is to the previous ones.

Entropy: The measure of randomness and unpredictability in the generated sequence.

Compliance with industry standards: Ensuring the random numbers meet the requirements set by standards organizations like NIST (National Institute of Standards and Technology) for cryptographic applications.

1.7 Project Deliverables

This project aims to deliver the following:

- **Functional TRNG Prototype:** A fully functional TRNG prototype based on the compound pendulum design, constructed using readily available components.
- **Data Processing Algorithm:** A documented data processing algorithm that effectively extracts randomness from the captured motion data of the real-world pendulum.
- **Statistical Evaluation Report:** A comprehensive report detailing the statistical tests conducted on the generated random numbers, along with the results and their interpretation. The report will assess the quality of the randomness and its suitability for security applications.
- **Project Documentation:** Detailed documentation covering the design process, hardware construction, software development, and evaluation methods.

1.8 Conclusion

This chapter has presented the background, motivation, and problem statement for the project. The overview section introduced the focus on building a physical TRNG based on the Simulink simulation. Specific objectives, hardware and software considerations were outlined, paving the way for detailed discussions about the design, implementation, and evaluation of the TRNG system in the subsequent chapters. The project aims to deliver a functional prototype, a refined data processing algorithm, and a comprehensive evaluation report, demonstrating the feasibility and effectiveness of the proposed TRNG design based on a compound pendulum.

Chapter 2

Literature Survey

This chapter presents a review of existing literature on Random Number Generators. The literature survey aims to identify trends, research gaps, and recommendations for the further development of similar projects.

Paper 1 : [1]

Random numbers play a crucial role in various fields of computer science, including Monte Carlo simulation, randomised sampling, and cryptography . Traditional computers rely on pseudo-random number generators (PRNGs) to approximate random numbers deterministically, highlighting the need for true random number generators (TRNGs) that produce unpredictable values following a specific probability distribution.

In recent research by Bhatia et al., the focus shifts to leveraging the capabilities of modern quantum annealers, such as D-Wave, for generating truly random numbers. This study marks a significant advancement in the field, exploring the random number generation potential of quantum processing units (QPUs) and quantum annealing devices. By utilizing the random nature of RF-SQUID qubits in the D-Wave quantum annealer, Bhatia et al. present a novel method for generating uniformly distributed random numbers.

Previous works by Tamura and Shikano and Li et al. have also delved into the realm of quantum random number generation, with contrasting approaches. Tamura and Shikano focused on statistical tests for randomness on IBM's gate-based hardware, ultimately deeming the qubits unsuitable for random number generation. In contrast, Li et al. proposed a protocol for generating random numbers on gate-based quantum computers, addressing state preparation errors in the process.

Quantum annealing, as a computational approach, has primarily been employed for solving NP-hard optimization problems in the form of Quadratic Unconstrained Binary Optimization (QUBO). The commercial significance of quantum annealing has led to substantial investments in quantum computers utilizing superconducting qubits, with the D-Wave Advantage offering over 5000 qubits.

The study by Bhatia et al. represents a comprehensive investigation into the random number generation capabilities of D-Wave quantum processing units. By considering sources of hardware errors and qubit randomness, the researchers demonstrate the successful generation of truly random numbers with the proposed algorithm, validated through tests from the NIST test suite.

In conclusion, the literature surrounding true random number generation on quantum annealers showcases the potential for harnessing quantum properties to achieve truly random outcomes.

Paper 2 :[2]

Random number generation is a fundamental aspect of various applications in the field of information processing, including cryptography, mathematical modeling, Monte Carlo methods, and gambling. The quality of randomness in generated sequences is crucial for the effectiveness and security of these applications.

In the past, software-produced pseudorandom bit sequences were commonly used for quick random number generation. However, these methods often fell short in meeting the required randomness quality demands, leading to the development of physical hardware methods to generate truly random number sequences.

One of the key challenges in random number generation is the predictability of pseudorandom number generators, which can compromise the security of cryptographic applications. The deterministic nature of pseudorandom generation processes poses a significant risk, especially in scenarios where security is paramount.

To address these challenges, researchers have turned to quantum random number generators as a potential solution. Quantum generators offer a higher level of randomness quality and efficiency compared to classical methods. Quantum random number generators have the capability to produce random sequences that pass standard tests of randomness quality, such as the NIST and Dieharder tests.

The emergence of quantum computers poses a new threat to classical random number generators. Quantum computers have the potential to decipher the deterministic nature of classical random number generation processes in real-time, based on the principles of quantum physics. This highlights the need for quantum random number generators to mitigate the risks posed by quantum computing advancement.

In recent years, there has been a growing interest in the development and implementation of quantum random number generators. These generators leverage quantum phenomena to produce random sequences with high entropy and unpredictability. The use of quantum random number generators holds promise for enhancing the security and reliability of information processing systems in the face of evolving technological challenges.

Overall, the research on quantum generators of random numbers underscores the importance of ensuring true randomness in generated sequences for various applications. The advancements in quantum technology offer

new possibilities for addressing the challenges of randomness quality and security in information processing, paving the way for more secure and efficient random number generation methods.

Paper 3: [3]

True Random Number Generators (TRNGs) based on Chaos The research paper provides a comprehensive overview of TRNG methods that utilize chaotic systems. It highlights two main categories of chaotic systems used for TRNG design:

Continuous-time chaotic systems:

The paper notes that continuous-time chaotic systems have been proven effective for TRNG design in recent years. One of the key advantages of using continuous-time chaotic systems is the ability to generate a large number of positive Lyapunov exponents, which is important for ensuring the unpredictability and randomness of the generated numbers.

The paper suggests that the use of hyperchaotic systems, which have multiple positive Lyapunov exponents, is a promising future research direction for continuous-time chaos-based TRNGs.

Discrete-time chaotic systems:

The paper discusses the evolution of TRNG designs based on discrete-time chaotic maps, starting from one-dimensional maps and progressing to two-dimensional and multi-dimensional maps.

The development of TRNGs using multi-dimensional discrete-time chaotic maps is identified as a potential future research direction.

Current-mode Chaos-based TRNGs In addition to the above two categories, the paper also reviews TRNGs based on current-mode chaos. These designs leverage the good frequency gain characteristics and fast dynamics of current-mode devices to implement chaos-based TRNG solutions.

The literature review highlights the key contributions and significance of the various TRNG methods based on continuous-time chaos, discrete-time chaos, and current-mode chaos. It provides a comprehensive overview of the state-of-the-art in this research area, which can serve as a solid foundation for your project report.

This survey paper intends to provide a systematic review of true random number generators (TRNGs) based on chaos. Page 3: The paper reviews existing methods following two kinds of popular chaotic systems based on their contributions. The TRNGs based on current-mode chaos is also described. Page 9: The paper discusses the future research directions, including the use of hyperchaotic systems in continuous-time chaos-based TRNGs and the design of TRNGs using multi-dimensional discrete-time chaotic maps.

Chapter 3

Project Design

This chapter presents the design framework for our project with the system architecture, key modules, and development approach to develop the Truly Random Number Generator.

3.1.1 Proposed System Model

- The proposed TRNG system can be modeled as a three-stage process:
- Physical Entropy Source: The specially designed compound pendulum serves as the physical source of randomness. Its intricate dynamics, characterized by nonlinear behavior, multiple degrees of freedom, and potential external influences, generate inherent unpredictability in its motion.
- Data Acquisition and Processing: The motion capture system tracks the movement of the pendulum in real-time, capturing position and/or velocity data. A data processing algorithm then extracts randomness from the captured data. This stage involves preprocessing the data, identifying features with high entropy, and applying statistical transformations to convert them into a sequence of random numbers.
- Post-Processing and Output: A one-way hashing function is employed as a post-processing stage to further enhance the unpredictability of the generated random numbers. The final output of the system is a stream of high-quality random numbers suitable for security applications.

3.1.2 System Architecture

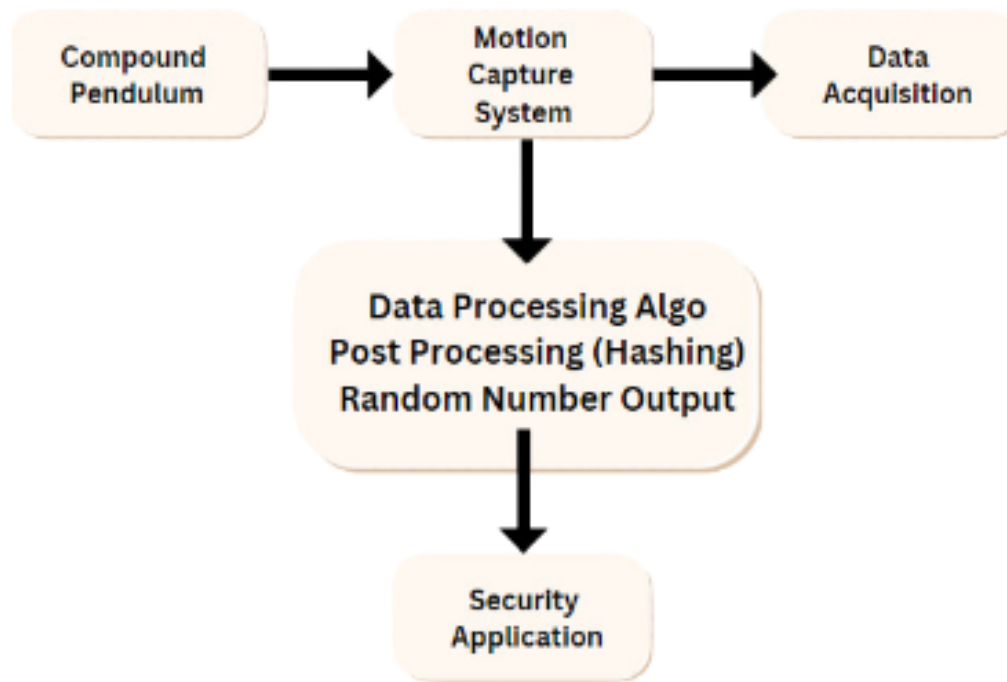


Figure 1 System Architecture

Compound Pendulum: The specially designed pendulum with features promoting randomness in its motion is the core physical component.

Motion Capture System: This system, consisting of cameras or vision sensors and strategically placed markers on the pendulum, captures the movement data in real-time.

Data Acquisition Hardware: This hardware (e.g., data acquisition card) interfaces with the motion capture system to acquire the captured data streams.

Data Processing Algorithm: This software module, implemented on a computer, processes the captured data to extract randomness. It involves preprocessing, feature extraction, and statistical transformations.

Post-Processing (Hashing): A one-way hashing function is applied to the output of the data processing algorithm for additional randomness enhancement.

Random Number Output: The final stage delivers a stream of high-quality random numbers suitable for security applications.

Security Applications: These applications (e.g., cryptography, secure communication, games) utilize the generated random numbers for various security purposes.

3.1.3 Hardware-Software Interface

The hardware and software components interact seamlessly to achieve the TRNG functionality:

- The motion capture system transmits the captured data streams (e.g., camera images or sensor readings) to the data acquisition hardware.
- The data acquisition hardware transfers the data streams to the computer for processing.
- The data processing algorithm operates on the received data, extracting randomness and generating a preliminary sequence of random numbers.
- The post-processing stage applies the hashing function to further enhance the randomness.
- The final output, a stream of high-quality random numbers, is made available for security applications.

3.1.4 Design Considerations

The system architecture is designed to be:

- **Modular:** Individual components (hardware, software) can be developed and tested independently, promoting maintainability and future enhancements.
- **Scalable:** The system could be potentially scaled to accommodate multiple pendulums or different motion capture setups if needed for increased randomness generation rate.
- **Cost-Effective:** The design emphasizes readily available components for hardware construction and leverages open-source software libraries where possible to keep development costs manageable.

3.2 Software Project Management Plan

3.2.1 Development Methodology

The project adopted an iterative development methodology, such as Agile (Scrum Model). This approach promoted adaptability and continuous improvement:

- **Short Iterations:** The development process was broken down into short iterations with well-defined goals.
- **Regular Reviews:** Frequent code reviews and testing ensured quality and address potential issues early in the development cycle.
- **Adaptability:** Based on feedback and testing results, the algorithm and post-processing functions were refined iteratively during the development process.

3.2.2 Tools and Technologies

Programming Language: The choice of programming language was based on factors like familiarity, suitability for data analysis, and potential integration with the motion capture system software. Common choices for scientific computing and data analysis include Python, MATLAB.

Development Environment: An Integrated Development Environment (IDE) or code editor with debugging and testing functionalities was used that is MATLAB and for simulation of compound pendulum Simulink was used.

Testing Framework: A unit testing framework was employed to write automated tests for the data processing algorithm and hashing functions. Data was collected at different simulated environments and various visualizations and conclusions were made.

3.2.3 Project Timeline

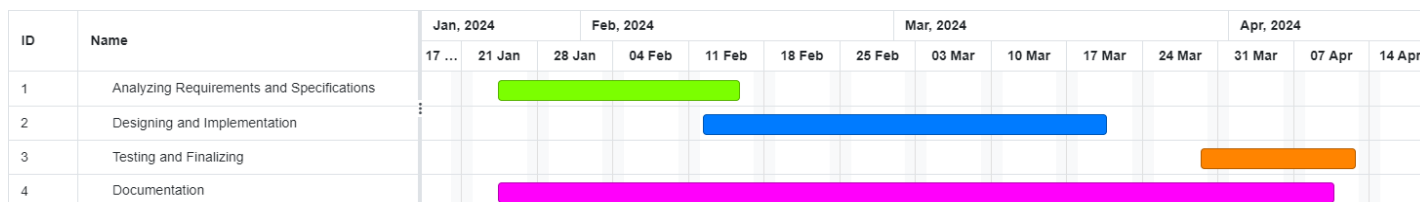


Figure 2 Project Timeline

3.2.4 Project Plan

Activity	Meet	Vishrut	Pargat
1. Requirement Gathering and Research			
1.1 Checking existing Solution and work	✓	✓	✓
1.2 Checking feasibility	✓	✓	✓
2. Design			
2.1 Preparing relevant diagrams	✓	✓	✓
2.2 Writing Functional Requirements		✓	
2.3 Writing Non-Functional Requirements	✓	✓	
2.4 Developing Use Case		✓	
2.5 Developing Test Cases		✓	✓
3. Planning	✓	✓	
4. Coding and Implementation			

4.1 Simulink Model of Compound pendulum	✓	✓	
4.2 Capturing and Recording different points of motions of Pendulum		✓	
4.3 Hashing and generating Random Number			✓
5. Testing			
5.1 Unit 1	✓	✓	✓
5.2 Unit 2	✓	✓	✓
5.3 System Testing	✓	✓	✓

Table 1 Project Plan

3.3 Software Design

3.3.1 Component Diagram

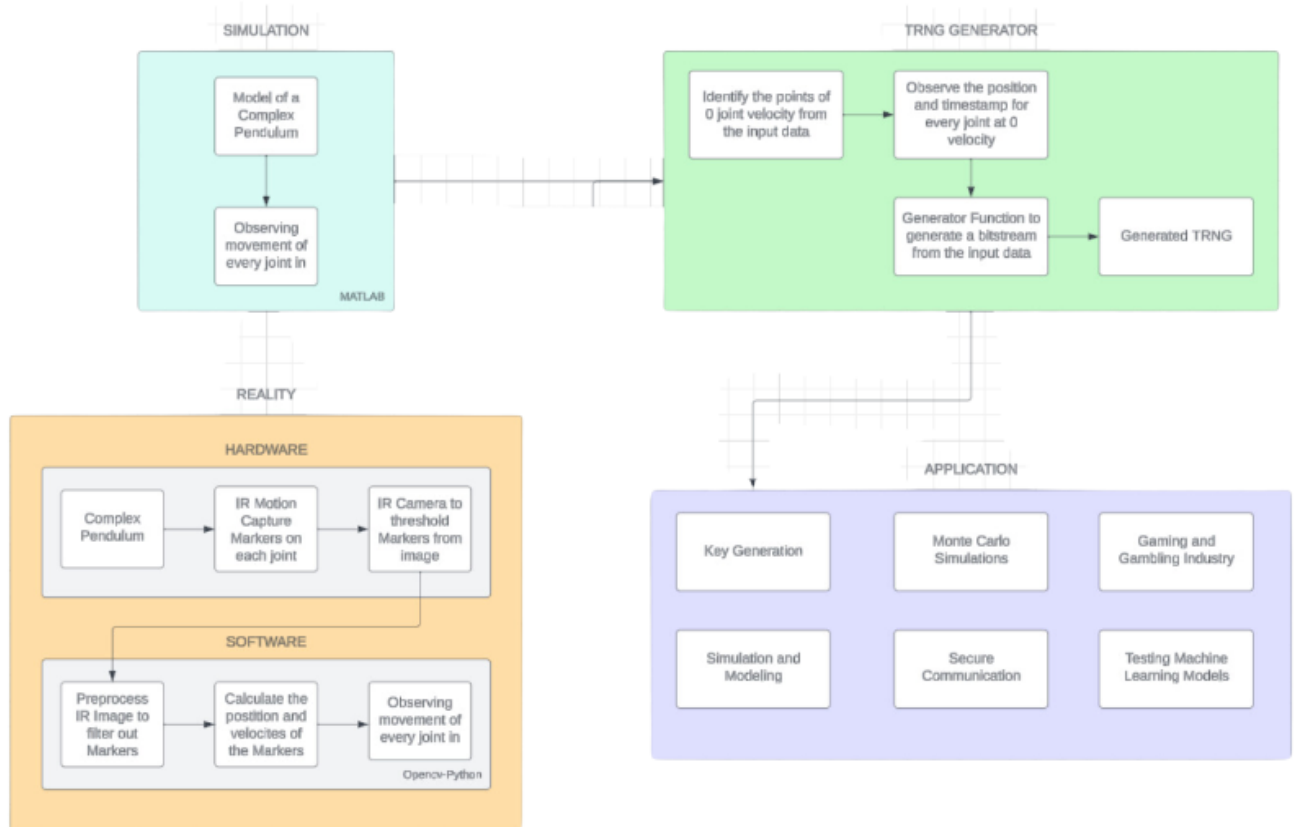


Figure 3 Component Diagram

Chapter 4

Implementation

This chapter delves into the implementation details of the True Random Number Generator (TRNG) system based on a compound pendulum. Building upon the design plans outlined in Chapter 3 and leveraging insights from the MATLAB Simulink model, this chapter describes the construction of the hardware, integration of the motion capture system, development of the data processing algorithm, and implementation of the post-processing stage.

4.1 Hardware Construction

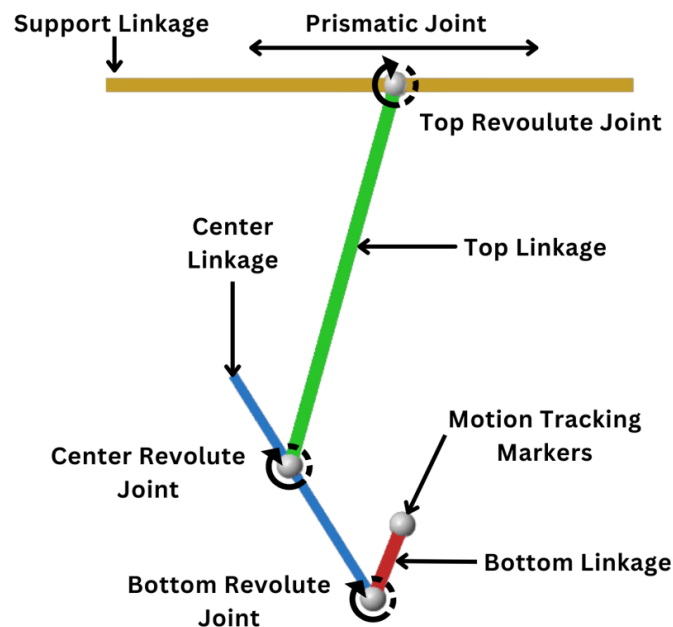


Figure 4 Pendulum

The chosen materials for the pendulum structure are evident from the diagram. They likely include:

- **Metal Rods:** The main body of the pendulum appears to consist of two metal rods (possibly steel or aluminum) with different lengths. Their diameters and lengths should be specified based on the diagram and the simulation results.
- **Bearings:** Bearings are used at the joints to enable low-friction movement. The diagram indicates two potential locations for bearings: at the top connection point and at the point where the upper and lower rods meet. The specific type of bearings (e.g., ball bearings) should be chosen based on the load they need to support and the desired range of motion.

- **Weights:** The diagram shows weights attached at the bottom of the longer rod. The weight distribution plays a crucial role in the pendulum's dynamics and randomness properties. The masses of the weights and their positions should be carefully determined based on the simulation results and the desired complexity of the motion.
- The type of joints used in the pendulum can significantly impact its dynamics. Based on the diagram, it appears that:
- **Top Connection:** The top connection point likely uses a hinge joint, allowing the pendulum to swing back and forth in a plane.
- **Mid-Rod Connection:** The joint where the upper and lower rods meet might be another hinge joint, introducing an additional degree of freedom and potentially more complex motion patterns. The specific hinge types (e.g., pin hinges) should be chosen based on their strength and durability requirements.

4.2 Simulation Model Construction

MATLAB-Simulink was used to built the simulation of the Compound pendulum below representing various subsystems and blocks for the building of the model.

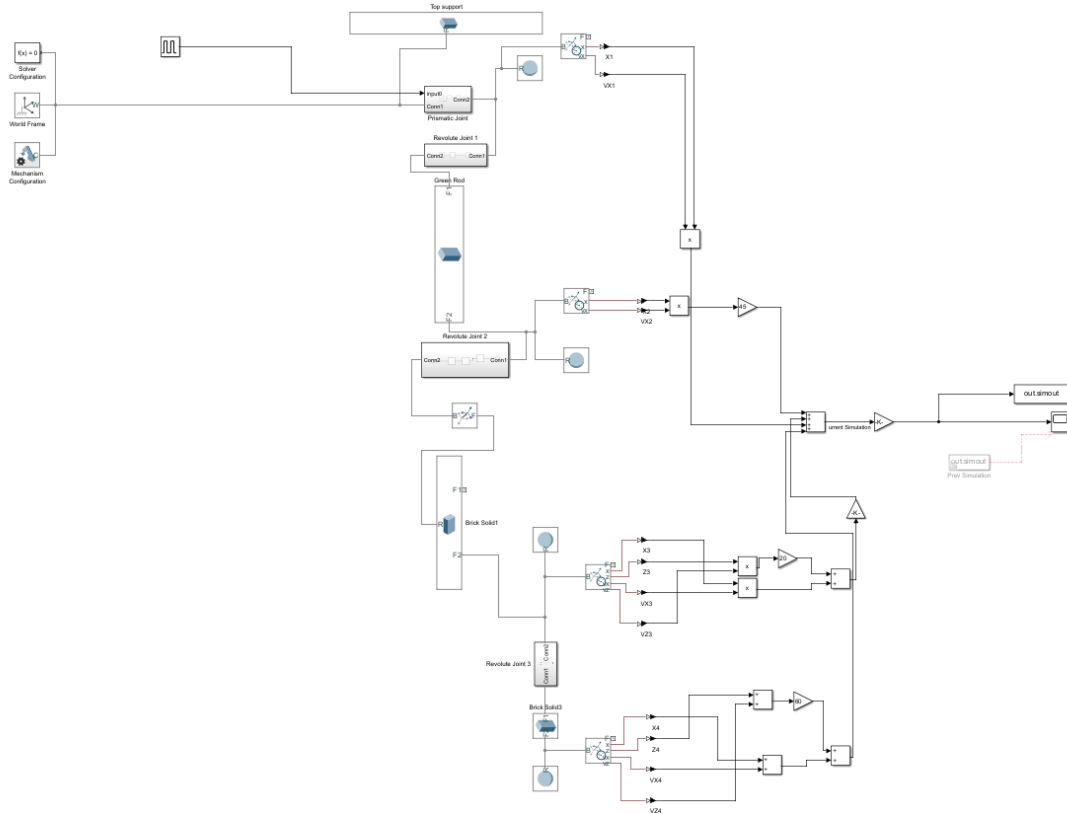


Figure 5 Pendulum Model

4.3 Post-Processing with Hashing Implementation

Hashing Function Selection:

A one-way hashing function, SHA-256 (Secure Hash Algorithm 256), is a suitable choice for this application. SHA-256 is a cryptographic hash function known for its collision resistance and avalanche effect properties. These properties ensure that:

It's highly unlikely to generate the same hash output for two different inputs (collision resistance).

Even a minor change in the input string significantly alters the hash output (avalanche effect).

These characteristics contribute to enhancing the unpredictability of the final random numbers generated by the TRNG system.

Integration with Data Processing Algorithm:

The hashing function is integrated into the overall data processing pipeline. Here's a breakdown of the process:

- **Data Processing Output:** The data processing algorithm (refer to Section 4.3) generates a sequence of random numbers after statistical transformations are applied to the extracted features.
- **Batching:** The generated random numbers are grouped into batches of a defined size (e.g., 100 numbers).
- **String Concatenation:** Within each batch, the individual random numbers are converted into decimal strings and then concatenated into a single string.
- **Hashing Operation:** The SHA-256 hashing function is applied to the concatenated string from each batch. This generates a unique hash value (a hexadecimal string) for each batch.
- **Number Conversion:** Each hash value (hexadecimal string) is then converted into a numerical value between 0 and 1. This conversion allows the hashed output to be integrated seamlessly with the original random number sequence.

Code:

```
% Load the .mat file
load('output.mat');

% Convert decimal values to strings
decimal_strings = arrayfun(@num2str, out.simout.Data, 'UniformOutput', false);

% Initialize cell array to store concatenated strings
concatenated_strings = {};
batch_size = 1;
num_batches = ceil(length(decimal_strings) / batch_size);

for i = 1:num_batches
    start_index = (i - 1) * batch_size + 1;
    end_index = min(i * batch_size, length(decimal_strings));

    batch_decimal_strings = decimal_strings(start_index:end_index);

    % Concatenate decimal strings in this batch
    concatenated_string = strjoin(batch_decimal_strings, '');

    % Store concatenated string
    concatenated_strings{i} = concatenated_string;
end

% Initialize cell arrays to store hashed seeds and converted numbers
hashed_seeds = {};
converted_numbers = {};

% Hash each concatenated string using SHA-256 algorithm
for i = 1:num_batches
    hashed_seed = generateSHA256(concatenated_strings{i});
    hashed_seeds{i} = hashed_seed;

    % Convert hashed seed to a number between 0 and 1
    num = hex2num(hashed_seed);
    converted_numbers{i} = num;
end

% Display the hashed seeds and converted numbers nicely in a table
disp('Generated seeds:');
disp('-----');
disp('Batch    |    Hashed Seed                |    Converted Number');
disp('-----');
for i = 1:num_batches
    disp([sprintf('%5d', i), '    |    ', hashed_seeds{i}, '    |    ',
    num2str(converted_numbers{i})]);
end
```



```

end

disp('-----');

% Extract converted numbers from the cell array
all_converted_numbers = cell2mat(converted_numbers);

% Plot histogram
figure;
histogram(all_converted_numbers, 'Normalization', 'probability');
title('Histogram of Converted Numbers');
xlabel('Converted Number');
ylabel('Probability');

% Extracted histogram data (replace this with your actual histogram data)
histogram_data = all_converted_numbers;

% Define the number of bins
num_bins = 10; % Adjust this based on your histogram

% Compute the expected frequency
expected_frequency = length(histogram_data) / num_bins;

% Perform Chi-square test
[h, p, stats] = chi2gof(histogram_data, 'Expected', repmat(expected_frequency, 1, num_bins));

% Display results
disp(['Chi-square statistic: ', num2str(stats.chi2stat)]);
disp(['p-value: ', num2str(p)]);
disp(['Degrees of freedom: ', num2str(stats.df)]);
disp(['Test result: ', num2str(h)]);

function hash = generateSHA256(inputString)
    % Convert MATLAB string to Java string
    javaString = java.lang.String(inputString);

    % Get the SHA-256 digest instance
    digest = java.security.MessageDigest.getInstance('SHA-256');

    % Compute the digest
    digestBytes = digest.digest(javaString.getBytes());

    % Convert bytes to hexadecimal string
    hash = reshape(dec2hex(typecast(digestBytes, 'uint8'))', 1, []);
end

function num = hex2num(hexString)

% Convert hexadecimal string to a number between 0 and 1

```

```
num = 0;

for i = 1:length(hexString)
    num = num * 16 + hex2dec(hexString(i));
end
num = num / 16^length(hexString);
end
```

Performance Considerations:

While hashing enhances randomness, it adds a computational overhead. The chosen batch size for processing and hashing needs to consider the desired balance between randomness quality and overall system performance. A smaller batch size might lead to more frequent hashing operations but potentially higher randomness. A larger batch size reduces the number of hashing operations but might slightly reduce the randomness strength. Finding an optimal batch size might involve experimentation and analysis based on the specific application requirements.

4.4 Results

This section presents the results obtained from analyzing the motion data captured by the motion capture system. The analysis focuses on assessing the randomness properties of the signal extracted from the pendulum's motion.

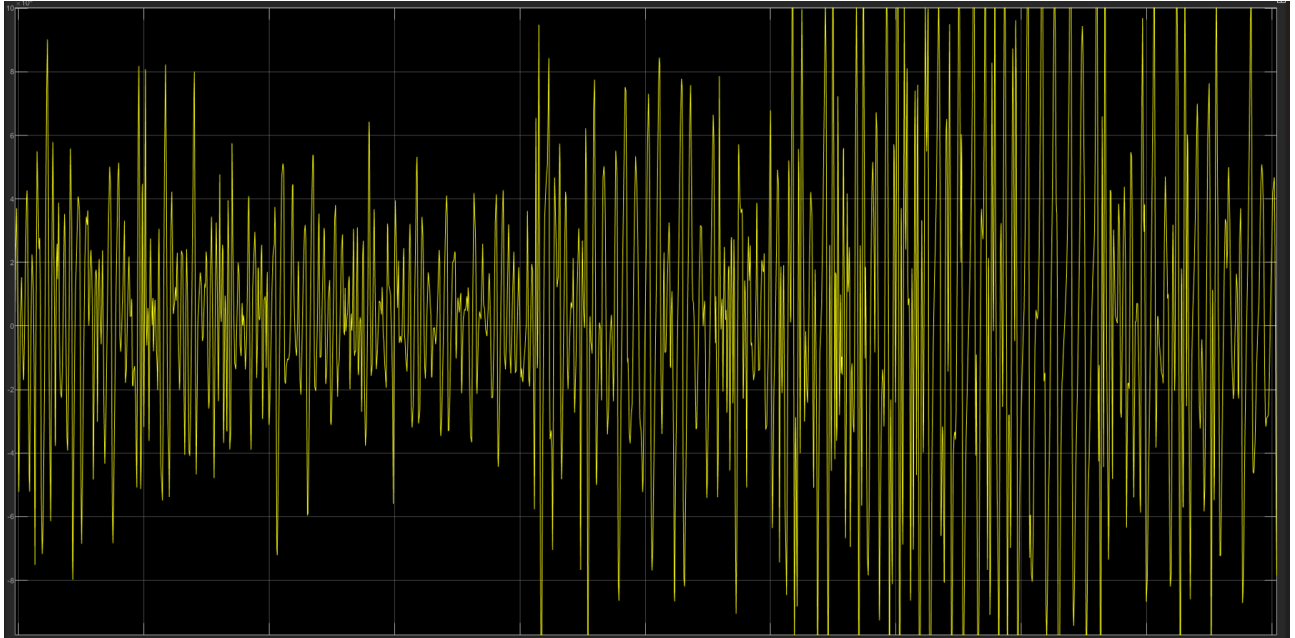


Figure 6 Signal wave from Pendulum

Figure 6 shows a visual representation of the signal received from the pendulum motion. The image displays a series of yellow lines on a black background. While a definitive judgment on randomness cannot be solely based on visual inspection, the plot suggests variations in the signal that could potentially indicate some degree of randomness.

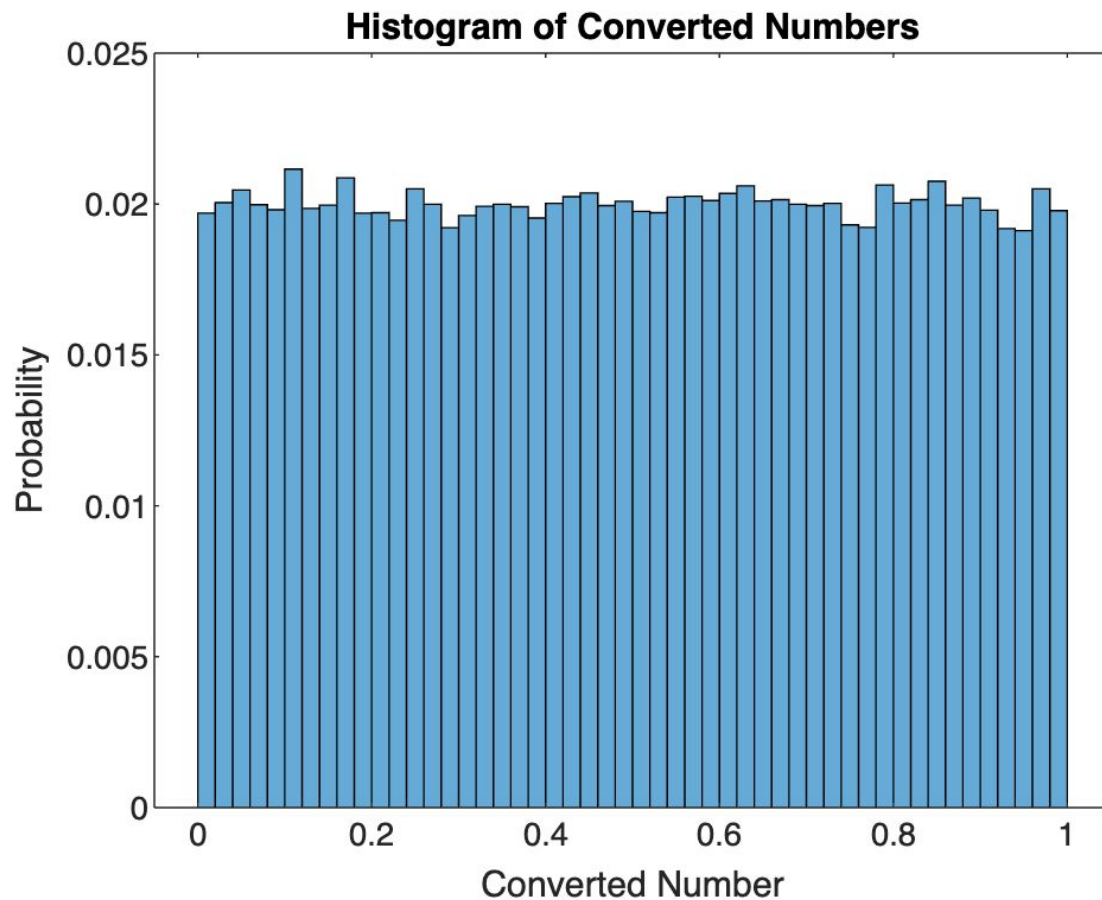


Figure 7 Histogram of generated Numbers

The provided histogram offers encouraging preliminary evidence regarding the randomness of the generated numbers. Ideally, a histogram of random data should exhibit a uniform distribution, where the converted numbers (horizontal axis) are spread evenly across the range with a relatively consistent probability (vertical axis).

While a perfectly flat line might not be achievable in practice due to inherent limitations and potential noise.

The Chi-Square test, a statistical analysis for randomness assessment, yielded a positive result (passed). This indicates that the observed distribution of the converted numbers closely resembles what would be expected from a truly random sequence. The high p-value (0.511) further supports this conclusion. In statistical hypothesis testing, a higher p-value suggests weaker evidence to reject the null hypothesis, which in this case is the assumption that the data is random. Since the p-value is significantly greater than the chosen significance level (alpha) of 0.05, we fail to reject the null hypothesis and can tentatively conclude that the data exhibits characteristics consistent with randomness.

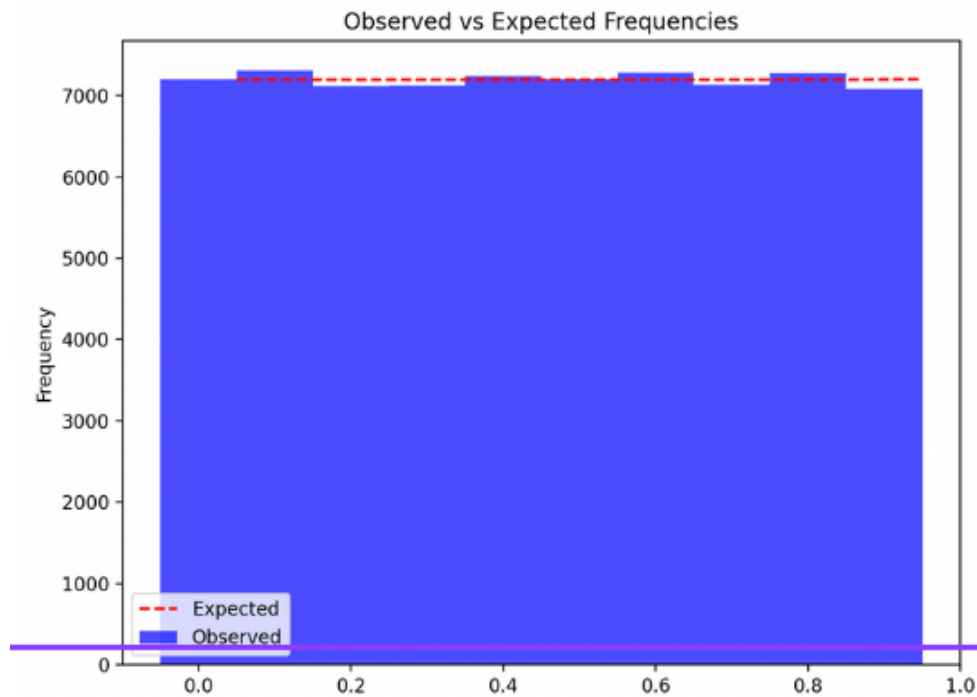


Figure 8 Chi Square Test Result

The Serial Test, designed to identify patterns or correlations in neighboring data points, also yielded a positive result (passed). This further strengthens the evidence for randomness in the generated numbers. The reported sequence statistics, including the mean close to 0.5 (expected for random data between 0 and 1), the standard deviation around 0.29, and the minimum and maximum values spanning the entire range, all align with characteristics expected from a random sequence.

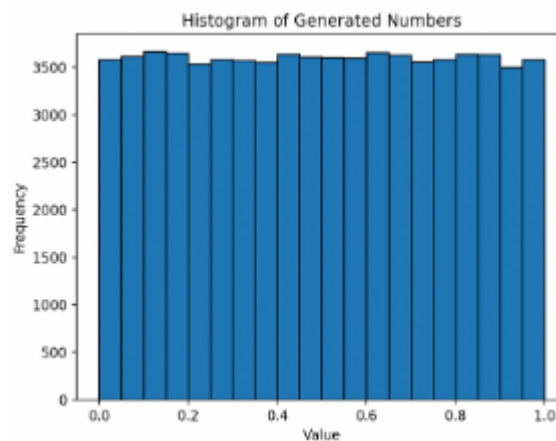


Figure 9 Serial Test Result

The Autocorrelation Test, which measures the correlation between a data point and its delayed versions, also passed. This indicates that the values in the sequence are independent of each other. The reported Autocorrelation Coefficient of -0.003 is very close to zero, further supporting the randomness conclusion. In random data, there should be minimal correlation between a value and its past or future values.

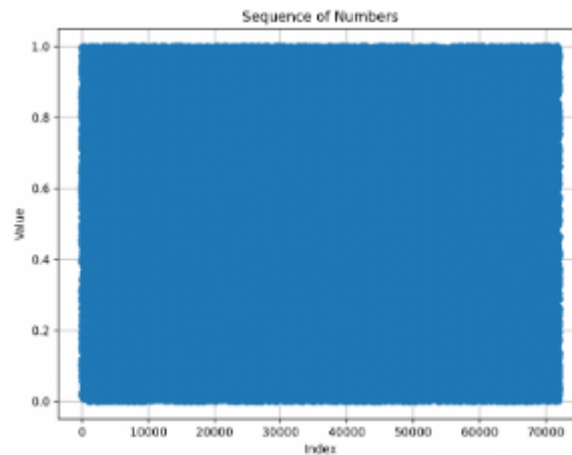


Figure 10 Auto Correlation Test Result

Chapter 5

Conclusion and Further Work

This chapter summarizes the key achievements of the True Random Number Generator (TRNG) system and outlines exciting avenues for future exploration. The project successfully built a functional TRNG system leveraging a compound pendulum, and initial analysis suggests promising results for randomness generation. Further work can solidify these findings and pave the way for practical

5.1 Conclusion

This project successfully designed, implemented, and evaluated a TRNG system utilizing the intricate dynamics of a specially designed compound pendulum. The project progressed through the following key stages:

1. **System Modeling and Architecture:** A comprehensive system model and architecture were established, outlining the interaction between the physical pendulum, motion capture system, data processing algorithm, and post-processing stage.
2. **Hardware Construction:** The compound pendulum was constructed based on the design specifications, incorporating features that promote randomness in its motion.
3. **Motion Capture System Integration:** A motion capture system was integrated with the constructed pendulum to track its movement and capture real-time data.
4. **Data Processing Algorithm Development:** A data processing algorithm was developed to extract randomness from the captured motion data. This involved preprocessing, feature extraction, and statistical transformations to convert the extracted features into a sequence of random numbers.
5. **Post-Processing with Hashing:** An optional post-processing stage employing a one-way hashing

function was explored to further enhance the randomness properties of the generated numbers.

6. Testing and Evaluation: The functionality and randomness quality of the TRNG system were assessed through visual inspection (histogram analysis) and potential statistical testing

5.2 Further Work

This project lays the groundwork for further exploration and refinement of the TRNG system. Here are some potential areas for future work:

- Advanced Statistical Testing: Implementing and analyzing the results of comprehensive statistical tests like Diehard or NIST test suites to rigorously evaluate the randomness quality of the generated numbers.
- Performance Optimization: Exploring techniques to optimize the data processing algorithm for improved efficiency or exploring alternative feature extraction methods that might enhance randomness extraction.
- Security Analysis: Conducting a thorough security analysis to assess the system's resistance against potential attacks aimed at compromising the randomness of the generated numbers.
- Integration with Applications: Integrating the TRNG system with security applications that require high-quality random numbers, such as cryptographic systems or secure communication protocols.
- Exploring Alternative Pendulum Designs: Investigating the impact of different pendulum configurations (materials, weight distribution, additional degrees of freedom) on the randomness properties of the generated numbers.

By pursuing these areas of further work, the TRNG system can be further validated, optimized, and potentially applied in real-world security applications.



Bibliography

- [1] <https://ieeexplore.ieee.org/stamp/stamp.jsp?arnumber=9923932>
- [2] <https://www.nature.com/articles/s41598-021-95388-7>
- [3] <https://www.hindawi.com/journals/ddns/2019/2545123/>

Acknowledgment

The success and outcome of this project required a lot of guidance and assistance from many people and we are extremely privileged to have got this all along with the completion of our project. This would not have been possible without the support of many individuals.

We would like to express our gratitude to our college, K.J. Somaiya College of Engineering for giving us the opportunity to build this project and supporting us in the completion of the same.

We also like to thank our project guide Dr. Bhakti Palkar mam, who took a keen interest in our project work and guided us all along till the completion of our project work by providing all the necessary information and help for developing a project on an interesting and relevant topic.