DESIGN AND IMPLEMENT HARDWARE ACCELERATOR FOR BLOCK CIPHERS USING FPGA or ASIC COMPARE THE PERFORMANCE GAINS SOFTWARE IMPLEMENTATIONS

Submitted in partial fulfilment of the **CAPSTONE PROJECT** in **APPLIED CRYPTOGRAPHY** , which is a part of

Integrated M. Tech. in Cybersecurity

By

Agnibha Chakraborty (22MEI10045)

Matrupriya Dibyanshu Panda (22MEI10040)

Ravi Shankar (22MEI10050)

Yogesh Baretha (22MEI10059)

Leonardo Fernandes (22MEI10064)

Submitted to

Dr. Hemraj S. Lamkuche

School of Computing Science and Engineering,
VIT Bhopal University, Madhya Pradesh

India

OCTOBER 2023

Motivation

The motivation behind this project is rooted in the critical need for high-speed data encryption, especially in an era where digital information is the lifeblood of our interconnected world. The widespread use of the internet, cloud computing, and the constant exchange of sensitive data demand cryptographic solutions that can ensure both security and efficiency. Traditional software-based implementations of block ciphers, while effective in principle, can often fall short when real-time encryption or decryption is essential. This performance gap becomes even more apparent in applications where large volumes of data require swift and secure processing, such as in financial transactions, secure communications, and data storage.

Furthermore, the motivation for this project is amplified by the opportunity to not just implement block ciphers using hardware accelerators but to go beyond the standard approach. In this endeavor, we have taken the widely adopted Advanced Encryption Standard (AES) algorithm and modified it to make it even more efficient than the standard version. By optimizing the algorithm for hardware acceleration, we aim to further enhance the performance gains achievable through dedicated hardware. This innovative approach aligns with the growing demand for encryption solutions that are not just secure but also highly responsive, enabling the safeguarding of data in real-time or near-real-time applications. The motivation behind this project is to explore the untapped potential of customizing cryptographic algorithms to meet the ever-increasing need for speed and security in the digital age.

Thanks!

Capstone Project Approval

This is to certify that the Integrated M. Tech. Capstone Project report titled **"Design and implement hardware accelerators for block ciphers using FPGA or ASIC. Compare the performance gains against software implementations."** by **Agnibha Chakraborty (22MEI10045) Leonardo Fernandes(22MEI10064) , Yogesh Baretha (22MEI10059) , Matrupriya Dibyanshu Panda (22MEI10040) , Ravi Shankar (22MEI10050)** is approved for the degree of **Integrated M. Tech. in Cybersecurity**.

Dr. Hemraj S. Lamkuche
(Course Coordinator)

Date: 26/10/2023

Place: **Bhopal**

Declaration

I declare that this written submission represents my ideas in my own words and where other's ideas or words have been included, I have adequately cited and referenced the original sources. I also declare that I have adhered to all principles of academic honestly and integrity and have not misrepresented or fabricated or falsified any ideas, data, facts or sources in my submission. I understand that any violation of the above will be cause of disciplinary action by the institute and evoke penal action from the sources which have thus not been properly cited or from whom proper permission has not been taken when needed.

Name of Author(s)

Agnibha Chakraborty (22MEI10045)
Leonardo Fernandes(22MEI10064)
Yogesh Baretha (22MEI10059)
Matrupriya Dibyanshu Panda (22MEI10040)
Ravi Shankar (22MEI10050)

Date:**26/10/2023**
Place: **BHOPAL**

Abstract

Title: High-Speed Block Cipher Hardware Accelerator: FPGA/ASIC Implementation and Performance Comparison

This project, "High-Speed Block Cipher Hardware Accelerator," focuses on designing and implementing hardware accelerators for block ciphers utilizing FPGA or ASIC technology. We aim to evaluate their performance gains when compared to software-based implementations. Notably, our work centers on a modified version of the AES algorithm, enhancing its efficiency beyond conventional standards. This research addresses the pressing need for high-speed data encryption and holds the potential to revolutionize cryptographic systems, offering both security and efficiency in the digital era.

Table of Contents

Motivation i

Capstone Project Approval ii

Declaration iii

Abstract iv

Table of Contents v

Nomenclature

Chapter 1 1

1.1 Background of Problem

1.1.1 Motivation

1.2 Context of Problem

Chapter 2 **Literature Review** 9

2.1 Basic info of sector/ topic

Chapter 3 **Research Methodology**

3.1 **Short note on methodology**

3.2 **Objectives**

Chapter 4 **RESULT**

4.1 Output

Chapter 5 **Discussions**

Conclusion

Appendix

References

Acknowledgements

List of Figures

Figure 1.1: AES Structure

Figure 1.2: Modified AES Structure

Figure 1.3: Input and codes of Modified AES

Figure 1.4: Output of Modified AES

Figure 1.5: Benchmark Analysis between Standard AES and Modified AES

List of Tables

Table 1: Benchmark Analysis Table

List of Graphs

Graph 1: Avalanche Effect Graph

Nomenclature

Project Title: "Design and Implementation of a Hardware Accelerator for Block Ciphers using FPGA or ASIC: A Comparative Performance Analysis with Software Implementations"

Design and Implementation: Refers to the development and creation of a hardware accelerator for block ciphers.

Hardware Accelerator: A specialized hardware component or system designed to accelerate the encryption and decryption processes of block ciphers.

Block Ciphers: A type of symmetric-key cryptographic algorithm that divides data into fixed-length blocks and encrypts/decrypts each block separately.

FPGA (Field-Programmable Gate Array): A reconfigurable hardware platform that can be programmed to perform specific tasks efficiently, often used for hardware acceleration in cryptographic applications.

ASIC (Application-Specific Integrated Circuit): A custom-designed integrated circuit built for a specific application, known for its efficiency and performance, often used for cryptographic accelerators.

Comparative Performance Analysis: Involves evaluating and contrasting the speed, throughput, power consumption, and other performance metrics of the hardware accelerator against software-based encryption and decryption.

Software Implementations: Refers to the traditional approach of implementing block ciphers using software on a general-purpose computing platform, such as a CPU.

Chapter 1
INTRODUCTION

1.1 Background of Problem

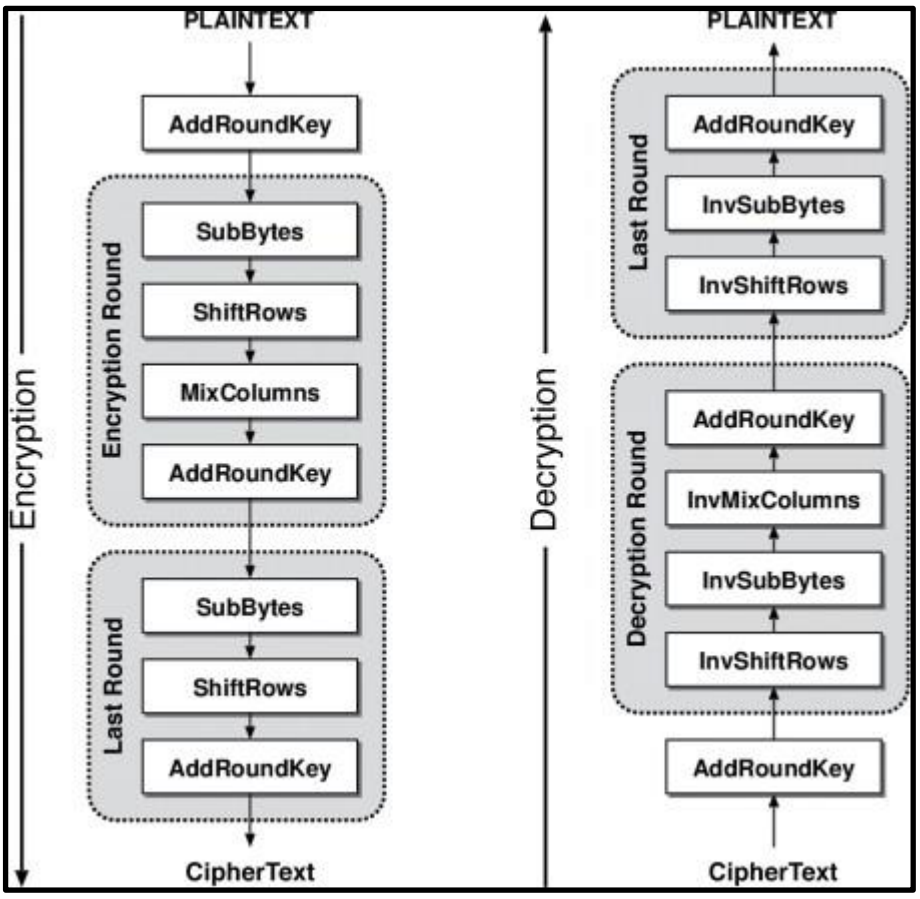
In today's digital age, data security is of paramount importance. The exponential growth in the volume and importance of digital data has necessitated the development of robust encryption techniques to protect sensitive information from unauthorized access and tampering. Cryptography, the science of securing communication, plays a vital role in ensuring the confidentiality and integrity of data.

Block ciphers are a fundamental component of modern cryptographic systems. These ciphers operate on fixed-size blocks of data and are employed in various applications, including secure communication, data storage, and access control. While software-based encryption has traditionally been the standard for implementing block ciphers, the growing demand for faster and more secure encryption methods has led to the exploration of hardware-based solutions.

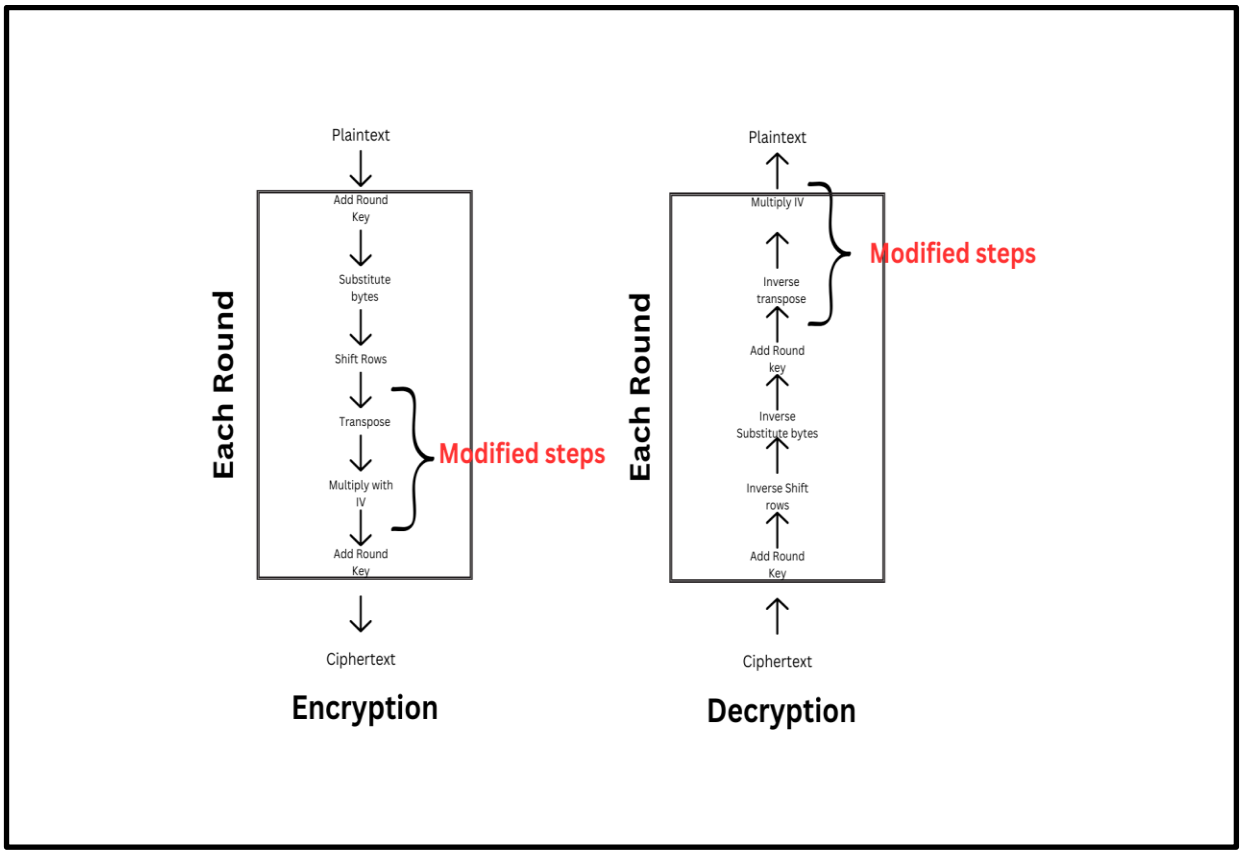
FPGAs and ASICs offer the potential to significantly improve the speed and efficiency of block cipher operations. They can be customized to perform specific cryptographic algorithms, leading to faster and more streamlined encryption and decryption processes. The goal of this project is to explore the design and implementation of such hardware accelerators and evaluate their performance gains in comparison to software-based implementations.

In a digital landscape where data security and speed are crucial, the outcomes of this research have broad-reaching societal implications. The ability to enhance the performance of block ciphers through dedicated hardware accelerators can positively impact various applications, including secure communication, financial transactions, and data protection. This project seeks to bridge the gap between the demand for high-speed encryption and the available technological solutions, ultimately contributing to a more secure and efficient digital environment.

1.1.1 Modifications in AES



Standard AES



Modified AES

Modified AES Encryption Rounds:

1. AddRoundKey:

The round starts with an "AddRoundKey" operation where the round key is XORed with the block's data to introduce the key's influence.

2. SubBytes:

Next, the "SubBytes" operation is applied, which substitutes bytes using a fixed S-box, adding an element of confusion to the data.

3. ShiftRows:

After "SubBytes," the "ShiftRows" transformation is used to shift bytes in each row of the block, achieving data diffusion.

4. Transpose:

An additional layer of security is introduced with the "Transpose" operation, which rearranges the block's data, making it more resistant to attacks.

5. Multiply with IV (Initialization Vector):

A unique step in our approach involves multiplying the result with a constant Initialization Vector (IV). This operation adds an extra layer of security and complexity to the encryption process.

6. AddRoundKey:

The encryption round concludes with another "AddRoundKey" operation, reinforcing the data's security by incorporating the key's influence once more.

Modified AES Decryption Rounds:

1. AddRoundKey:

Similar to encryption, the decryption process starts with an "AddRoundKey" step to reverse the key's influence.

2. Inverse ShiftRows:

The "Inverse ShiftRows" operation is applied, undoing the ShiftRows transformation to restore the original data order.

3. Inverse SubBytes:

Following that, the "Inverse SubBytes" operation is executed, reversing the SubBytes substitution for data recovery.

4. AddRoundKey:

Another "AddRoundKey" operation is employed to further reverse the key's influence during decryption.

5. Inverse Transpose:

A critical component of the decryption process is the "Inverse Transpose" operation, which undoes the Transpose step, restoring the data's original order.

6. Multiply IV:

Just as in the encryption phase, the result is multiplied by the same constant Initialization Vector (IV) used during encryption, completing the decryption round.

These modifications to the standard AES algorithm offer an innovative approach to enhance both the efficiency and security of block ciphers. The project's objective is to evaluate and compare the performance gains of this modified AES against traditional software-based implementations.

1.1.2 Societal Concerns

- **Data Privacy**: The need for robust encryption is driven by growing concerns over data privacy and protection. Inadequate encryption measures can lead to unauthorized access and data breaches.
- **Cybersecurity Threats**: The continually evolving landscape of cyber threats, including hacking, malware, and data theft, poses a significant risk to individuals, organizations, and even critical infrastructure.

- **Consumer Confidence**: Data breaches and security lapses erode consumer trust. High-profile incidents can have a profound impact on an organization's reputation and financial stability.
- **Regulatory Compliance**: Numerous regulations, such as GDPR and HIPAA, mandate strong data protection practices. Failure to comply can result in legal consequences and financial penalties.
- **National Security**: In the realm of national security, secure communication and data protection are vital to safeguard classified information and maintain secure channels of communication.
- **Economic Impact**: Cyberattacks and data breaches have economic consequences, including financial losses, job cuts, and reduced economic growth, affecting societies at large.
- **Technological Progress**: The continuous advancement of technology underscores the need for equally advanced encryption to counteract new threats and vulnerabilities.
- **Digital Inclusion**: Secure data communication is essential for fostering digital inclusion and ensuring that individuals and communities worldwide can participate in the digital economy and access essential services.
- **Ethical Considerations**: Data protection and encryption are at the intersection of technology and ethics. It is important to address ethical concerns related to privacy and security.
- **Resource Efficiency**: In a world with finite resources, it is crucial to implement energy-efficient cryptographic solutions to reduce the environmental impact of data security practices.

These societal concerns underscore the critical importance of the project in the context of modern digital society and its implications for individuals, organizations, and governments.

1.2 Context of Problem

In the realm of cryptography and data security, the context of the problem provides insight into the challenges and limitations of existing solutions, paving the way for innovative approaches like high-speed block cipher hardware accelerators. The following points illuminate the context of the problem:

1. **Performance Bottlenecks**: Software-based block cipher implementations often encounter performance bottlenecks, especially when dealing with large volumes of data. This can result in slow encryption and decryption processes, hindering the real-time processing of data.

2. **Resource Constraints**: Resource-constrained environments, such as IoT devices and embedded systems, face challenges when implementing encryption algorithms due to their limited processing power and memory. Efficient hardware accelerators are crucial for these applications.

3. **Complex Encryption Algorithms**: Modern encryption algorithms, designed to resist sophisticated attacks, are computationally intensive. Software-based solutions may struggle to implement these algorithms with sufficient speed and efficiency.

4. **Latency in Secure Communication**: Secure communication protocols, like secure socket layer (SSL) or transport layer security (TLS), require quick encryption and decryption to minimize latency in online transactions and web services. Slow cryptographic operations can lead to user dissatisfaction and potentially disrupt the flow of data.

5. **Security vs. Performance Trade-off**: Achieving a balance between security and performance is an ongoing challenge in cryptography. While stronger encryption is essential, it should not come at the cost of significantly slower processing speeds.

6. **Energy Consumption**: In applications where power efficiency is crucial, such as in portable devices or remote sensors, software-based encryption can consume excessive energy, leading to shorter battery life and environmental concerns.

7. **Competitive Markets**: In industries like finance, e-commerce, and cloud computing, the race to provide secure and speedy data handling is highly competitive. Companies that can offer more efficient encryption solutions are better positioned in the market.

8. **Security Gaps**: In some cases, the security of software-based encryption can be compromised due to vulnerabilities and implementation flaws. Robust hardware accelerators can minimize these vulnerabilities and provide enhanced security.

9. **Customization Needs**: Different use cases require tailored encryption solutions. Hardware accelerators, such as those implemented on FPGAs or ASICs, can be customized to suit specific cryptographic requirements, offering a level of adaptability that software solutions may lack.

10. **The Future of Encryption**: As technology evolves and the nature of cyber threats changes, the future of encryption is closely tied to innovative approaches. Hardware acceleration represents a frontier for exploration in this ever-evolving field.

In light of these contextual factors, the development and assessment of hardware accelerators for block ciphers become a pertinent avenue for addressing the limitations and challenges associated with software-based cryptographic implementations. By addressing the need for faster and more efficient encryption while maintaining high levels of security, this project aims to contribute to a more secure and efficient digital world.

Chapter 2

Literature Review

2.1 Basic info of sector/ topic

1. **Cryptography and Block Ciphers**:

Cryptography, as an indispensable field in modern information security, serves the critical purpose of safeguarding sensitive data in digital communication, storage, and transactions. Among the various cryptographic techniques, block ciphers play a pivotal role due to their ability to process fixed-size data blocks. Block ciphers, exemplified by widely used algorithms like the Advanced Encryption Standard (AES), are essential components of modern encryption systems.

2. **Encryption Performance Challenges**:

Despite the importance of block ciphers, software-based encryption methods face notable challenges. These challenges often manifest as performance bottlenecks when dealing with substantial amounts of data. Software-based encryption can strain system resources, leading to slow processing speeds and latency in data handling, which is particularly problematic in real-time applications.

3. **Hardware Accelerators**:

Hardware accelerators, on the other hand, offer a promising solution to the limitations of software-based encryption. These specialized hardware components are designed to expedite specific computational tasks. Their application in accelerating cryptographic operations has gained considerable attention due to their ability to significantly improve processing speed and efficiency.

4. **FPGAs and ASICs**:

Two primary types of hardware accelerators, Field-Programmable Gate Arrays (FPGAs) and Application-Specific Integrated Circuits (ASICs), are at the forefront of this research. FPGAs are known for their flexibility and programmability, allowing for a wide range of applications. ASICs, in contrast, are custom-designed for specific tasks, delivering optimized performance. Both technologies boast parallel processing capabilities, making them particularly well-suited for cryptographic acceleration.

5. **Real-time Encryption Needs**:

Real-time data processing is an imperative requirement in various applications, such as secure communication, financial transactions, and multimedia streaming. In these scenarios, swift encryption and decryption are vital to minimize latency and ensure the seamless flow of data. Software-based encryption can often fall short of meeting these real-time demands.

6. **Energy Efficiency**:

Efficiency, particularly in terms of energy consumption, is paramount in today's digital landscape. Resource-constrained environments, such as Internet of Things (IoT) devices and battery-operated systems, require encryption solutions that strike a balance between security and power consumption. Hardware accelerators have the potential to address this need by providing energy-efficient cryptographic processing.

7. **Security Concerns**:

The evolving threat landscape, marked by increasingly sophisticated cyberattacks, underscores the importance of robust encryption. Software-based encryption, while effective, can introduce vulnerabilities due to implementation flaws or emerging attack methods. Hardware accelerators offer a means to bolster security by providing faster and more robust cryptographic operations, thus making it more challenging for adversaries to compromise data.

8. **Regulatory Compliance**:

In today's interconnected world, numerous regulations and compliance requirements dictate strong data protection practices. Regulations like the General Data Protection Regulation (GDPR) and the Health Insurance Portability and Accountability Act (HIPAA) mandate the secure handling of sensitive data. Compliance with these regulations necessitates the use of secure encryption methods, where hardware accelerators have shown promise in fulfilling these requirements.

9. **Economic and Competitive Aspects**:

Beyond the realms of data protection, the economic implications of cyberattacks and data breaches are significant. Such incidents can lead to severe financial losses, job cuts, and reduced economic growth. Companies and organizations that can offer efficient encryption solutions gain a competitive advantage in the market, enabling them to meet customer demands and fulfill regulatory obligations.

10. **Customization and Adaptability**:

One of the key advantages of hardware accelerators lies in their ability to be customized for specific cryptographic requirements. This customization makes them adaptable to various use cases, allowing encryption to be tailored to the specific needs of applications across diverse industries.

11. **Research Trends**:

The research landscape in the field of cryptography is continually evolving. Emerging technologies and novel approaches, including the implementation of hardware accelerators, are shaping the future of data security. Current trends highlight the growing interest in hardware acceleration for encryption, pointing to its relevance and significance in addressing contemporary data security challenges.

By incorporating these points into your literature review, you will provide a well-rounded and informative overview of the context and background surrounding your project on high-speed block cipher hardware accelerators.

Chapter 3
Research Methodology

3.1 Short note on methodology

This research employs a mixed-method approach, combining both quantitative and qualitative methods to conduct a comprehensive analysis of hardware accelerators in the realm of block cipher implementations. Acknowledging a constructive ontology, the study recognizes that knowledge is constructed through human interactions and interpretations. Furthermore, the epistemological standpoint of this research integrates positivism for quantitative analysis and constructivism for qualitative exploration.

At the heart of this research is a central inquiry: "How do hardware accelerators, implemented using FPGA and ASIC technology, impact the performance and security of block cipher encryption compared to software-based implementations?" This question guides the entire investigative process.

The study articulates four key objectives. First, it seeks to design and implement hardware accelerators for select block ciphers. This involves a deep dive into the design considerations inherent in hardware-based block cipher implementations. Second, the research endeavors to assess the performance gains achieved through hardware acceleration, meticulously examining variables such as speed, power efficiency, and resource utilization. The third objective is to evaluate the security implications introduced by the use of hardware accelerators in block cipher encryption. Finally, the research aims to gather expert opinions on the practical applications and limitations of hardware acceleration in the field of cryptography.

To address these objectives effectively, the study posits several sub-research questions. The first sub-research question pertains to the hardware implementation process, probing

how hardware accelerators can be designed and implemented effectively using FPGA and ASIC technology. Key considerations in the design phase are scrutinized. The second sub-research question delves into the performance evaluation, where the study dissects specific performance metrics used to measure the efficiency of hardware accelerators. A rigorous comparison is drawn between hardware and software-based encryption in terms of speed, throughput, and resource utilization. Sub-research question three centers on the security assessment, meticulously evaluating the security implications that arise from the application of hardware accelerators in block cipher encryption. Lastly, the fourth sub-research question hinges on expert insights, as the research seeks to glean perspectives from domain experts on the practicality and limitations of hardware acceleration in real-world cryptographic applications. The trade-offs between hardware-based and software-based encryption methods are also a subject of expert inquiry.

To execute these tasks, the research delineates a structured plan. Task one involves the design and implementation of hardware accelerators, using FPGA and ASIC technology. Task two is dedicated to the collection of data on encryption/decryption speeds, throughput, power consumption, and resource utilization. The ensuing analysis facilitates a comprehensive performance comparison. Task three immerses the research in a thorough security assessment, scrutinizing the security implications introduced by hardware accelerators. Task four shifts the focus toward expert insights, encompassing surveys and interviews to gather qualitative data regarding the practicality and limitations of hardware acceleration in cryptography.

While the scope of this study is focused, honing in on a selection of block ciphers for hardware acceleration, it provides a deep and comprehensive analysis without striving to encompass all existing algorithms. The study amalgamates quantitative performance analysis and qualitative expert opinions, fostering a multi-faceted view of the research domain. Its purview delves into the impact of hardware accelerators on both performance and security, dissecting their interaction with block ciphers, without straying into other aspects of cryptography.

Nevertheless, the research operates within essential boundaries. It does not extend to the design and development of cryptographic algorithms themselves but is concentrated solely on the hardware acceleration facet. Ethical considerations are paramount, ensuring that no sensitive or personal data is employed in the research process. Moreover, it

assumes a reasonable level of prior knowledge in cryptography and hardware design among the research participants, carving out an essential boundary for the study.

3.2 Objectives

- **Task 1 - Hardware Implementation**: Design and implement hardware accelerators for selected block ciphers using FPGA and ASIC technology.
- **Task 2 - Performance Evaluation**: Collect, analyze, and compare data on encryption/decryption speeds, throughput, power consumption, and resource utilization for hardware and software implementations.
- **Task 3 - Security Assessment**: Conduct a comprehensive analysis of the security implications introduced by hardware accelerators.
- **Task 4 - Expert Insights:** Survey and interview experts in the field to gather qualitative data regarding the practicality and limitations of hardware acceleration in cryptography

Chapter 4

Results

4.1 Output

Code:

```
s_box_string = '63 7c 77 7b f2 6b 6f c5 30 01 67 2b fe d7 ab 76' \
               'ca 82 c9 7d fa 59 47 f0 ad d4 a2 af 9c a4 72 c0' \
               'b7 fd 93 26 36 3f f7 cc 34 a5 e5 f1 71 d8 31 15' \
               '04 c7 23 c3 18 96 05 9a 07 12 80 e2 eb 27 b2 75' \
               '09 83 2c 1a 1b 6e 5a a0 52 3b d6 b3 29 e3 2f 84' \
               '53 d1 00 ed 20 fc b1 5b 6a cb be 39 4a 4c 58 cf' \
               'd0 ef aa fb 43 4d 33 85 45 f9 02 7f 50 3c 9f a8' \
               '51 a3 40 8f 92 9d 38 f5 bc b6 da 21 10 ff f3 d2' \
               'cd 0c 13 ec 5f 97 44 17 c4 a7 7e 3d 64 5d 19 73' \
               '60 81 4f dc 22 2a 90 88 46 ee b8 14 de 5e 0b db' \
               'e0 32 3a 0a 49 06 24 5c c2 d3 ac 62 91 95 e4 79' \
               'e7 c8 37 6d 8d d5 4e a9 6c 56 f4 ea 65 7a ae 08' \
               'ba 78 25 2e 1c a6 b4 c6 e8 dd 74 1f 4b bd 8b 8a' \
               '70 3e b5 66 48 03 f6 0e 61 35 57 b9 86 c1 1d 9e' \
               'e1 f8 98 11 69 d9 8e 94 9b 1e 87 e9 ce 55 28 df' \
               '8c a1 89 0d bf e6 42 68 41 99 2d 0f b0 54 bb 16'.replace(" ", "")
```

```
inv_s_box_string = '52 09 6a d5 30 36 a5 38 bf 40 a3 9e 81 f3 d7 fb' \
                   '7c e3 39 82 9b 2f ff 87 34 8e 43 44 c4 de e9 cb' \
                   '54 7b 94 32 a6 c2 23 3d ee 4c 95 0b 42 fa c3 4e' \
                   '08 2e a1 66 28 d9 24 b2 76 5b a2 49 6d 8b d1 25' \
                   '72 f8 f6 64 86 68 98 16 d4 a4 5c cc 5d 65 b6 92' \
                   '6c 70 48 50 fd ed b9 da 5e 15 46 57 a7 8d 9d 84' \
                   '90 d8 ab 00 8c bc d3 0a f7 e4 58 05 b8 b3 45 06' \
                   'd0 2c 1e 8f ca 3f 0f 02 c1 af bd 03 01 13 8a 6b' \
                   '3a 91 11 41 4f 67 dc ea 97 f2 cf ce f0 b4 e6 73' \
                   '96 ac 74 22 e7 ad 35 85 e2 f9 37 e8 1c 75 df 6e' \
                   '47 f1 1a 71 1d 29 c5 89 6f b7 62 0e aa 18 be 1b' \
                   'fc 56 3e 4b c6 d2 79 20 9a db c0 fe 78 cd 5a f4' \
                   '1f dd a8 33 88 07 c7 31 b1 12 10 59 27 80 ec 5f' \
                   '60 51 7f a9 19 b5 4a 0d 2d e5 7a 9f 93 c9 9c ef' \
                   'a0 e0 3b 4d ae 2a f5 b0 c8 eb bb 3c 83 53 99 61' \
                   '17 2b 04 7e ba 77 d6 26 e1 69 14 63 55 21 0c 7d'.replace(" ", "")

s_box = bytearray.fromhex(s_box_string)
inv_s_box = bytearray.fromhex(inv_s_box_string)

class MAES:
    def __init__(self, key):
        self.key = key
        self.key_schedule = self.key_expansion(key)
        self.nr=10
```

```
def key_expansion(self,key: bytes, nb: int = 4):

    nk = len(key) // 4

    key_bit_length = len(key) * 8

    if key_bit_length == 128:
        nr = 10
    elif key_bit_length == 192:
        nr = 12
    else:  # 256-bit keys
        nr = 14

    w = self.state_from_bytes(key)

    for i in range(nk, nb * (nr + 1)):
        temp = w[i-1]
        if i % nk == 0:
            temp = self.xor_bytes(self.sub_word(self.rot_word(temp)), self.rcon(i // nk))
        elif nk > 6 and i % nk == 4:
            temp = self.sub_word(temp)
        w.append(self.xor_bytes(w[i - nk], temp))

    return [w[i*4:(i+1)*4] for i in range(len(w) // 4)]
```

```
def sub_word(self,word) -> bytes:
    substituted_word = bytes(s_box[i] for i in word)
    return substituted_word

def rcon(self,i: int) -> bytes:
    rcon_lookup = bytearray.fromhex('01020408102040801b36')
    rcon_value = bytes([rcon_lookup[i-1], 0, 0, 0])
    return rcon_value

def xor_bytes(self,a: bytes, b: bytes) -> bytes:
    return bytes([x ^ y for (x, y) in zip(a, b)])

def rot_word(self,word):
    return word[1:] + word[:1]
def add_round_key(self,state, key_schedule, round: int):
    round_key = key_schedule[round]
    for r in range(4):
        state[r] = [state[r][c] ^ round_key[r][c] for c in range(len(state[0]))]

def sub_bytes(self,state):
    for r in range(len(state)):
        state[r] = [s_box[state[r][c]] for c in range(len(state[0]))]

def shift_rows(self,state):
```

```
        state[0][1], state[1][1], state[2][1], state[3][1] = state[1][1], state[2][1], state[3][1], state[0][1]
        state[0][2], state[1][2], state[2][2], state[3][2] = state[2][2], state[3][2], state[0][2], state[1][2]
        state[0][3], state[1][3], state[2][3], state[3][3] = state[3][3], state[0][3], state[1][3], state[2][3]



    def xtime(self,a: int) -> int:
        if a & 0x80:
            return ((a << 1) ^ 0x1b) & 0xff
        return a << 1



    def state_from_bytes(self,data: bytes) :
        state = [data[i*4:(i+1)*4] for i in range(len(data) // 4)]
        return state

    def bytes_from_state(self,state) -> bytes:
        return bytes(state[0] + state[1] + state[2] + state[3])

    def transpose(self, state):
```

```
    return [list(bytearray(x)) for x in zip(*state)]

def multiply_with_iv(self, state, round):
    iv_string = '000102030405060708090a0b0c0d0e0f'  # Replace with any IV values
    iv = bytearray.fromhex(iv_string)
    for r in range(len(state)):
        state[r] = [state[r][c] ^ iv[c] for c in range(len(state[0]))]

def encrypt(self,data: bytes) -> bytes:
    state = self.state_from_bytes(data)

    self.add_round_key(state, self.key_schedule, round=0)

    for round in range(1, self.nr):
        self.sub_bytes(state)
        self.shift_rows(state)
        state = self.transpose(state)
        self.multiply_with_iv(state, round)
        state = self.transpose(state)
        self.add_round_key(state, self.key_schedule, round)
        print(f'Round {round} Ciphertext: {self.bytes_from_state(state).hex()}')

    self.sub_bytes(state)
    self.shift_rows(state)
    state = self.transpose(state)
```

```
        self.multiply_with_iv(state, self.nr)
        state = self.transpose(state)
        self.add_round_key(state, self.key_schedule, round=self.nr)
        print(f'Round 0 Ciphertext: {self.bytes_from_state(state).hex()}')

        cipher = self.bytes_from_state(state)
        return cipher


    def inv_shift_rows(self,state) :
        state[1][1], state[2][1], state[3][1], state[0][1] = state[0][1], state[1][1], state[2][1], state[3][1]
        state[2][2], state[3][2], state[0][2], state[1][2] = state[0][2], state[1][2], state[2][2], state[3][2]
        state[3][3], state[0][3], state[1][3], state[2][3] = state[0][3], state[1][3], state[2][3], state[3][3]
        return

    def inv_sub_bytes(self,state) :
        for r in range(len(state)):
            state[r] = [inv_s_box[state[r][c]] for c in range(len(state[0]))]

    def xtimes_0e(self,b):
        # 0x0e = 14 = b1110 = ((x * 2 + x) * 2 + x) * 2
        return self.xtime(self.xtime(self.xtime(b) ^ b) ^ b)
```

```
def xtimes_0b(self,b):
    # 0x0b = 11 = b1011 = ((x*2)*2+x)*2+x
    return self.xtime(self.xtime(self.xtime(b)) ^ b) ^ b

def xtimes_0d(self,b):
    # 0x0d = 13 = b1101 = ((x*2+x)*2)*2+x
    return self.xtime(self.xtime(self.xtime(b) ^ b)) ^ b

def xtimes_09(self,b):
    # 0x09 = 9  = b1001 = ((x*2)*2)*2+x
    return self.xtime(self.xtime(self.xtime(b))) ^ b


def decrypt(self,cipher: bytes) -> bytes:

    state = self.state_from_bytes(cipher)

    self.add_round_key(state, self.key_schedule, round=self.nr)

    for round in range(self.nr-1, 0, -1):
        state = self.transpose(state)
        self.multiply_with_iv(state, round)
        state = self.transpose(state)
        self.inv_shift_rows(state)
        self.inv_sub_bytes(state)
```

```
            self.add_round_key(state, self.key_schedule, round)
            print(f'Round {round} Decrypted Ciphertext: {self.bytes_from_state(state).hex()}')

        state = self.transpose(state)
        self.multiply_with_iv(state, 0)
        state = self.transpose(state)
        self.inv_shift_rows(state)
        self.inv_sub_bytes(state)
        self.add_round_key(state, self.key_schedule, round=0)
        print(f'Round 0 Decrypted Ciphertext: {self.bytes_from_state(state).hex()}')

        plain = self.bytes_from_state(state)
        return plain
```

```
from StdMAES import MAES
from key import key
from plaintext import plaintext

Encrypt_Cipher = []
Decrypt_Cipher = []

if len(plaintext) <= 16:
    print(plaintext)
    c = MAES(key)
    cipher = c.encrypt(plaintext)
    decrypted = c.decrypt(cipher)
else:
    for i in range(0, len(plaintext), 16):
```

```
        chunk = plaintext[i:i+16]

        if len(chunk) == 16:
            c = MAES(key)
            cipher = c.encrypt(chunk)
            Encrypt_Cipher.append(cipher)
            decrypted = c.decrypt(cipher)
            Decrypt_Cipher.append(decrypted)
        else:
            c = MAES(key)
            chunk_padded = chunk.ljust(16, b'\x00')
            cipher = c.encrypt(chunk_padded)
            Encrypt_Cipher.append(cipher)
            decrypted = c.decrypt(cipher)
            decrypted = decrypted.rstrip(b'\x00')
            Decrypt_Cipher.append(decrypted)

if len(plaintext) <= 16:

print(f'plaintext:{plaintext}\nkey:{key}\nciphertext:{cipher}\nDecrypte
d text:{decrypted}')
else:
    Encryption = b''.join(Encrypt_Cipher)
    Decryption = b''.join(Decrypt_Cipher)

print(f'plaintext:{plaintext}\nkey:{key}\nciphertext:{Encryption}\nDecr
ypted text:{Decryption}')
plaintext=b'Encapsulation is a way to restrict the direct access to some
components of an object, so users cannot access state values for all of
the variables of a particular object. Encapsulation can be used to hide
both data members and data functions or methods associated with an
instantiated class or object.'
```

Output:

```
Round 1 Ciphertext: f0459260db13fb1757acee3877b69d6c
Round 2 Ciphertext: 24641520b7c23745813403f18e5016f4
Round 3 Ciphertext: bd9eca732cf09fee52c0cc373eec1409
Round 4 Ciphertext: 686a98fbe6b4f12cc852ebd35f3ac8cb
Round 5 Ciphertext: 64a13acd3822307e973f00fa5f8ef6be
Round 6 Ciphertext: 319f09a0c35bf0c232897506e72ca402
Round 7 Ciphertext: 513f86887c8fe060cac85c51577e7336
Round 8 Ciphertext: e3c5ef4070768301fcd5146012f44c3a
Round 9 Ciphertext: 2b6ac4130bcf1b5f6354bc9b51ce20de
Round 0 Ciphertext: 82528c9a02346cac0075a5f8b096d82f
Round 9 Decrypted Ciphertext: 1138fa8050022808b2bddd7ecaa5efd3
Round 8 Decrypted Ciphertext: d1734a0511e98ec576f146d25876e2d2
Round 7 Decrypted Ciphertext: c7399d772fa648e12173032797d88f6c
Round 6 Decrypted Ciphertext: 439363ae067443bc8a1b82f1cc31072e
Round 5 Decrypted Ciphertext: 458de91f8f01e90eea824473cc01a265
Round 4 Decrypted Ciphertext: 7a8c4b0170bbfb8e02cc762ab108d899
Round 3 Decrypted Ciphertext: 36257bbfa81946b60e515b6c1a4099a2
Round 2 Decrypted Ciphertext: 8c7d2850b8905fd1594c4df2f66d0c04
Round 1 Decrypted Ciphertext: 347ca2f07c58aef38086c8afd7f3c99f
```

```
Round 0 Decrypted Ciphertext: 456e63617073756c6174696f6e206973
Round 1 Ciphertext: 27124c47d1c9372c1587c9f651eb39b7
Round 2 Ciphertext: 64c4e0d930447b3583936570aaf783b1
Round 3 Ciphertext: c8a0fc048134346cb2fb74cf8bb3aff9
Round 4 Ciphertext: fafe41639b017251fff12fabd0d10bdb
Round 5 Ciphertext: 0c50c66ba283f38a6981c55ae03717ba
Round 6 Ciphertext: 8ce0ccfafe224200430a418bc94dad91
Round 7 Ciphertext: f295987ee94f3cadf35a16171e44d364
Round 8 Ciphertext: bb32e2067e206a36853d16253bab461b
Round 9 Ciphertext: d0e5793ca9eb68394489fbe27a4bce31
Round 0 Ciphertext: 0331e640fab3503ae04d0f25bb4d32a3
Round 9 Decrypted Ciphertext: eab747aff2265b6e95609a07e120013c
Round 8 Decrypted Ciphertext: 898447431fbf67f20f1944977129e8f3
Round 7 Decrypted Ciphertext: 64938381ba66942c18e14961dee22f3e
Round 6 Decrypted Ciphertext: feeca6f43b0df17efb98b67ce2500ebd
Round 5 Decrypted Ciphertext: 2d7c15b915a02afa143c81d373b84361
Round 4 Decrypted Ciphertext: e81892990d0e78f3356fb2523ee31b89
Round 3 Decrypted Ciphertext: 431b4dc805dded34ee6ae394af1f2252
Round 2 Decrypted Ciphertext: ccddda93f1613a15beb2b73d2ca9941
Round 1 Decrypted Ciphertext: e32b7cd7768262c8c2adef61f1ae6d44
Round 0 Decrypted Ciphertext: 20612077617920746f20726573747269
```

```
Round 1 Ciphertext: 6fc977558d37296701acc95fd891393a
Round 2 Ciphertext: 0083e0f053c27b69a6fbb9841ae3bc3c
Round 3 Ciphertext: e89ee72768e7bdd57a8274a08543aff7
Round 4 Ciphertext: 89724192d21d726f12860bf87a3a69b1
Round 5 Ciphertext: 8688f81a0366213eb63fc5234acc1799
Round 6 Ciphertext: 363fcce0bf5b42ddf4dbb447feda5d09
Round 7 Ciphertext: 933f96fe5a91e56156ee16b578d0d3f9
Round 8 Ciphertext: ee37e2dcdeb66a7e3956c05ff5f4743e
Round 9 Ciphertext: 121c8421477da0d0c154fb147ef2cec1
Round 0 Ciphertext: ba27e6ff8934502c8377e647920897c1
Round 9 Decrypted Ciphertext: 284ebab21cb0938710bd9af1e59901cc
Round 8 Decrypted Ciphertext: dc814799bf2967bab37292edbf76dad6
Round 7 Decrypted Ciphertext: 05398d0109b84de0bd5549c3b8762fa3
Round 6 Decrypted Ciphertext: 4433a6ee7a74f1a34c4943b0d5c7fe25
Round 5 Decrypted Ciphertext: a7a42bc8b445f84ecb8281aad9434342
Round 4 Decrypted Ciphertext: 9b949268441278cdd8189601940879e3
Round 3 Decrypted Ciphertext: 632556ebec0e648d2613e3fba1ef225c
Round 2 Decrypted Ciphertext: a89add805c9013fd7e83f78762dea6cc
Round 1 Decrypted Ciphertext: abf047c52a7c7c83d686efc878d46dc9
Round 0 Decrypted Ciphertext: 63742074686520646972656374206163
Round 1 Ciphertext: 6f564291bce1f0a2aed647b5f8da30b7
Round 2 Ciphertext: 00e19dd96ba56d143e2d603b3a8f9526
Round 3 Ciphertext: e8bd613bfa30f26cece0cba3a757b24a
Round 4 Ciphertext: 89e2cc2cbaef3c4106c770abb14b9a5b
Round 5 Ciphertext: 86f382eb42e46000100c0d085814bcba
Round 6 Ciphertext: 3665bdfae8d0d796706ae6964213701f
Round 7 Ciphertext: 9376953fc92af8adb8c427e4efe8f1c9
Round 8 Ciphertext: ee536998bd82adb0e4bd7a2596b9ec82
Round 9 Ciphertext: 1241e48020b6fc10babd9a0008855931
Round 0 Ciphertext: ba9651409e6e101c0f69d0fd5117c758
```

```
Round 9 Decrypted Ciphertext: 2813da137b7bcf476b54fbe593ee963c
Round 8 Decrypted Ciphertext: dce5ccdddc1da0746e992897dc3b426a
Round 7 Decrypted Ciphertext: 05708ec09a03502c537f78922f4e0d93
Round 6 Decrypted Ciphertext: 4469d7f42dff64e8c8f81161690ed333
Round 5 Decrypted Ciphertext: a7df5139f5c7b9706db14981cb9be861
Round 4 Decrypted Ciphertext: 9b041fd62ce036e3cc59ed525f798a09
Round 3 Decrypted Ciphertext: 6306d0f77ed92b34b0715cf883fb3fe1
Round 2 Decrypted Ciphertext: a8f8a0a964f70580e6552e3842b28fd6
Round 1 Decrypted Ciphertext: ab6f72011baaa54679fc6122589f6444
Round 0 Decrypted Ciphertext: 6365737320746f20736f6d6520636f6d
Round 1 Ciphertext: 6056ddff718ff08615ac80b577011039
Round 2 Ciphertext: 786af062adc2a38383068d458e8f9526
Round 3 Ciphertext: 379eec3b1087f2f3b2e019b53ead84c6
Round 4 Ciphertext: 88f1074e5def5441ff0951f65f3a9a84
Round 5 Ciphertext: e5f3028dfa23605e693f83085f2d779a
Round 6 Ciphertext: ab2a86b6e95b4722434882ade713701f
Round 7 Ciphertext: f43f083f4c7af8cef3c419e757405fcc
Round 8 Ciphertext: 8d6c710e4982c3b0852f603b12f4ec7f
Round 9 Ciphertext: 6741ee4161d9fcfd4454c0005138e2ec
Round 0 Ciphertext: f6ed5349c6344352e0f99163b017c758
Round 9 Decrypted Ciphertext: 5d13d0d23a14cfaa95bda1e5ca532de1
Round 8 Decrypted Ciphertext: bfdad44b281dce740f0b328958764297
Round 7 Decrypted Ciphertext: 623913c01f53504f187f469197e6a396
Round 6 Decrypted Ciphertext: d926ecb82c74f45cfbda755acc0ed333
Round 5 Decrypted Ciphertext: c4dfd15f4d00b92e1482c781cca22341
Round 4 Decrypted Ciphertext: 9a17d4b4cbe05ee33597cc0fb1088ad6
Round 3 Decrypted Ciphertext: bc255df7946e2babee718eee1a01096d
Round 2 Decrypted Ciphertext: d073cd12a290cb175b7ec346f6b28fd6
Round 1 Decrypted Ciphertext: a46fed6fd6c4a562c286a622d74444ca
Round 0 Decrypted Ciphertext: 706f6e656e7473206f6620616e206f62
```

```
Round 1 Ciphertext: 0139535522b0546786b4ee3ac8da10b7
Round 2 Ciphertext: d4fe15d99ddea3699e2da184932c3973
Round 3 Ciphertext: c3a68343db30ca6c55e2cca0fb1484f7
Round 4 Ciphertext: 3ce298922e9654b9346673abe21567b1
Round 5 Ciphertext: cabc5c1a87115d3e67e600dd081477ba
Round 6 Ciphertext: 068e09fad3a047dd3f6abf47187becee
Round 7 Ciphertext: f9e613d7342a67ad9c985cb56ebc5ff9
Round 8 Ciphertext: ab53efdc78d8c3cb56432d25d60f283e
Round 9 Ciphertext: 5833e621e6d606d0629dbcf86e85e231
Round 0 Ciphertext: 192e8c40a74a432c51693747fe57187a
Round 9 Decrypted Ciphertext: 6261d8b2bd1b3587b374dd1df5ee2d3c
Round 8 Decrypted Ciphertext: 99e54a991947ce0fdc677f979c8d86d6
Round 7 Decrypted Ciphertext: 6fe008286703cf2c772303c3ae1aa3a3
Round 6 Decrypted Ciphertext: 748263f4168ff4a387f848b033664fc2
Round 5 Decrypted Ciphertext: eb908fc83032844e1a5b44549b9b2361
Round 4 Decrypted Ciphertext: 2e044b68b8995e1bfef8ee520c2777e3
Round 3 Decrypted Ciphertext: 481d328f5fd9133409735bfbdfb8095c
Round 2 Decrypted Ciphertext: 7ce728a9928ccbfd4655ef87eb112383
Round 1 Decrypted Ciphertext: c50063c585fb0183519ec8ad689f4444
Round 0 Decrypted Ciphertext: 6a6563742c20736f2075736572732063
Round 1 Ciphertext: 3a3953fb56b83b82a8b48065f8b6a593
Round 2 Ciphertext: 2875f0acbfde6f9a1834a1123a2cfbbe
Round 3 Ciphertext: bfa683628df0d7c8f3e219e1a7322661
Round 4 Ciphertext: 1a6a0715ca96fc09c5bf7313b1151da9
Round 5 Ciphertext: 83bc5c01c22a7c28d9e6838a588ee7a5
Round 6 Ciphertext: 9ee98608e1a026038f89bfc1427bb051
Round 7 Ciphertext: 9de6132eaa8f4eb09a98190fefbb0821
Round 8 Ciphertext: 6cc571b8ccd83cf430cc2d57960f829d
Round 9 Ciphertext: 6a33e6cd1187213ad79dc05808ce2755
Round 0 Ciphertext: 71cf537bab4a176cf57537b751578a51
```

```
Round 9 Decrypted Ciphertext: 5061d85e4a4a126d0674a1bd93a5e858
Round 8 Decrypted Ciphertext: 5e73d4fdad473130bae87fe5dc8d2c75
Round 7 Decrypted Ciphertext: 0be008d1f9a6e631712346792f1df47b
Round 6 Decrypted Ciphertext: ece5ec06248f957d371b48366966137d
Round 5 Decrypted Ciphertext: a2908fd37509a558a45bc703cb01b37e
Round 4 Decrypted Ciphertext: 088cd4ef5c99f6ab0f21eeea5f270dfb
Round 3 Decrypted Ciphertext: 341d32ae09190e90af738eba839eabca
Round 2 Decrypted Ciphertext: 806ccddcb08c070ec04cef114211e14e
Round 1 Decrypted Ciphertext: fe00636bf1f36e667f9ea6f258f3f160
Round 0 Decrypted Ciphertext: 616e6e6f7420616363657373207374 61
Round 1 Ciphertext: 104b69a0d1c92998aeacc917c8da9d6c
Round 2 Ciphertext: 62c4e02030c237753e2db547938dbc03
Round 3 Ciphertext: 219e64b78130bdeeecce74c4fbb31408
Round 4 Ciphertext: efe241ca9b85f10a06f1dcd3e23a694d
Round 5 Ciphertext: febb5531a2832105103fc5ec0814f6be
Round 6 Ciphertext: c9e0cca0fe5bf0b8706a099e18f45de1
Round 7 Ciphertext: 4b3f1a07e92ae560b80616186e447352
Round 8 Ciphertext: 8153e2457ef18300e43df260d6f47446
Round 9 Ciphertext: 36f3b7c9a9eba038ba54fb846e8520de
Round 0 Ciphertext: 7631e69afa346c0c0f691030fe999764
Round 9 Decrypted Ciphertext: 0ca1895af226936f6bbd9a61f5eeefd3
Round 8 Decrypted Ciphertext: b3e547001f6e8ec46e19a0d29c76daae
Round 7 Decrypted Ciphertext: dd3901f8ba034de153bd496eaee28f08
Round 6 Decrypted Ciphertext: bbeca6ae3b7443c6c8f8fe6933e9fecd
Round 5 Decrypted Ciphertext: df9786e315a0f8756d8281659b9ba265
Round 4 Decrypted Ciphertext: fd0492300d8afba8cc6f412a0c08791f
Round 3 Decrypted Ciphertext: aa25d57b05d964b6b05fe39fdf1f99a3
Round 2 Decrypted Ciphertext: cadddd503f905fe1e655fb44ebb0a6f3
Round 1 Decrypted Ciphertext: d472593076827c7c7986ef80689fc99f
Round 0 Decrypted Ciphertext: 74652076616c75657320666f7220616c
```

```
Round 1 Ciphertext: b856f9a0bcba9e9e8683b817d18d808c
Round 2 Ciphertext: c4ed51146bbfa4759e27d50a458f1203
Round 3 Ciphertext: 97b3b2b7fa2411a355e044c449fac7cf
Round 4 Ciphertext: 9ad0c870baefcd0a34b1a8f1d65c914d
Round 5 Ciphertext: 99f3113142ea592067f5aeec66fcea79
Round 6 Ciphertext: 9c8b8eb8e8c835b83f2077421b136be1
Round 7 Ciphertext: 48eeee07c99fd6ec9cc444186c986975
Round 8 Ciphertext: 606dbed8bd82f5005660787e19a95b46
Round 9 Ciphertext: ea4182c9201c0b376238cd844c542afd
Round 0 Ciphertext: f44454d39e133e0c51deaaad48175c64
Round 9 Decrypted Ciphertext: d013bc5a7bd13860b3d1ac61d73fe5f0
Round 8 Decrypted Ciphertext: 52db1b9ddc1df8c4dc442acc532bf5ae
Round 7 Decrypted Ciphertext: dee8f5f89ab67e6d777f1b6eac3e952f
Round 6 Decrypted Ciphertext: ee87e4b62de786c687b280b5300ec8cd
Round 5 Decrypted Ciphertext: b8dfc2e3f5c980501a48ea65f573bea2
Round 4 Decrypted Ciphertext: 88361b8a2ce0c7a8fe2f3508386e811f
Round 3 Decrypted Ciphertext: 1c08037b7ecdc8fb0971d39f6d564a64
Round 2 Decrypted Ciphertext: 6cf46c6464edcce1465f9b093db208f3
Round 1 Decrypted Ciphertext: 7c6fc9301bf1cb7a51a99e8071c8d47f
Round 0 Decrypted Ciphertext: 6c206f662074686520766172696162 6c
Round 1 Ciphertext: f4394ca0cd299182b6d6c9b5d121a535
Round 2 Ciphertext: 17bce0e6b3a56f7594876512452c9826
Round 3 Ciphertext: 7bbdfc3be8ff9ed77ce274c449ca2661
Round 4 Ciphertext: 33f041150c96fc41d8e82ff5d64b184d
Round 5 Ciphertext: e2bcc63148b975281e0cc5086600e73e
Round 6 Ciphertext: ea5accbc96d026b8c8f341c11b7b3d1f
Round 7 Ciphertext: 1176983fc2258ee5019816186c1b0821
Round 8 Ciphertext: b089e2b845d83cb0f489166919b9b446
Round 9 Ciphertext: dd3379c9346bbf3a6cbdfb004ccf27f7
Round 0 Ciphertext: b2a7e6ef316e170cab740fb748577f58
```

```
Round 1 Ciphertext: b696f98b563da41786ffe437f8b83054
Round 2 Ciphertext: e63e5450bf456da89e16d5f13aae5069
Round 3 Ciphertext: 05d5b2358daf8b0a5577b59ba71db209
Round 4 Ciphertext: 799f06fbcafb3c353438a89cb1322e45
Round 5 Ciphertext: 972311bcc225e97e679c291d5857bc06
Round 6 Ciphertext: fa33cf61e1f0d71a3fcb77064238be8b
Round 7 Ciphertext: bb8aeec2aa37076f9cbed7d6ef66f136
Round 8 Ciphertext: d82cab40cc30ade05615781896ff681d
Round 9 Ciphertext: 5b5682371195775f62fd0106081959a3
Round 0 Ciphertext: 4af2958dab40104b512aaaf851258254
Round 9 Decrypted Ciphertext: 6104bca44a584408b31460e3937296ae
Round 8 Decrypted Ciphertext: ea9a0e05adafa024dc312aaadc7dc6f5
Round 7 Decrypted Ciphertext: 2d8cf53df91eafee770588a02fc00d6c
Round 6 Decrypted Ciphertext: 883fa56f24df6464875980f169251da7
Round 5 Decrypted Ciphertext: b60fc26e7506300e1a216d94cbd8e8dd
Round 4 Decrypted Ciphertext: 6b79d5015cf43697fea635655f003e17
Round 3 Decrypted Ciphertext: 8e6e03f90946525209e622c083b13fa2
Round 2 Decrypted Ciphertext: 4e276920b017053c466e9bf242934a99
Round 1 Decrypted Ciphertext: 72afc91bf176f1f351d5c2a058fd64a7
Round 0 Decrypted Ciphertext: 73756c6174696f6e2063616e20626520
Round 1 Ciphertext: 6956c05abc7af09978ff63b5f82130b7
Round 2 Ciphertext: 51c3c6d96b456d2b6687f6ef3a8f9526
Round 3 Ciphertext: 5ad5f33bfafff26c6de021a8a781b277
Round 4 Ciphertext: acf02e0fbaef3c41f49092abb1329a93
Round 5 Ciphertext: b0f39c0e42426007c09c77085800bcba
Round 6 Ciphertext: 95209ffae8f0d7d400f32b304213701f
Round 7 Ciphertext: bc8aea3fc925f8ad8ac4863cef12f15d
Round 8 Ciphertext: 5789e109bd82adb0f6efd72596ffec00
Round 9 Ciphertext: 614130f02013fc5791fd9b0008cf5931
Round 0 Ciphertext: 9ca5fd409e40105d7a74bd6c5117c758
```

```
Round 9 Decrypted Ciphertext: 5b130e637bdecf004014fae593a4963c
Round 8 Decrypted Ciphertext: 653f444cdc1da0747ccb8597dc7d42e8
Round 7 Decrypted Ciphertext: 2a8cf1c09a0c502c617fd94a2fb40d07
Round 6 Decrypted Ciphertext: e72cf5f42ddf64aab861dcc7690ed333
Round 5 Decrypted Ciphertext: 91df4fdcf561b977bd213381cb8fe861
Round 4 Decrypted Ciphertext: be16fdf52ce036e33e0e0f525f008ac1
Round 3 Decrypted Ciphertext: d16e42f77e162b343171b6f3832d3fdc
Round 2 Decrypted Ciphertext: f9dafba9641705bfbeffb8ec42b28fd6
Round 1 Decrypted Ciphertext: ad6ff0ca1b31a57dafd5452258646444
Round 0 Decrypted Ciphertext: 7573656420746f206869646520626f74
Round 1 Ciphertext: af44b2fb56b454178cb4d441c88d39b7
Round 2 Ciphertext: d19475d9bfde7b9abe277bf193253970
Round 3 Ciphertext: b5a6909d8d24ca6cf0ac08e1fb8daf09
Round 4 Ciphertext: c7d0e3fbca9f72fd44c1ffabe21567a9
Round 5 Ciphertext: e7f7c501c25a5d7e64e657df08fc17ba
Round 6 Ciphertext: e6b231fae1a04203f92053061876ecb1
Round 7 Ciphertext: 18e6f637aa9f67ad70819a0f6e92d336
Round 8 Ciphertext: 9f6d1d40cc926a5fd9691225d60f289d
Round 9 Ciphertext: e11df7cd1135065fe69dc7286e54ce31
Round 0 Ciphertext: 8b4e2f40ab4a506c75ded1f8fe30180f
Round 9 Decrypted Ciphertext: db4fc95e4af835083774a6cdf53f013c
Round 8 Decrypted Ciphertext: addbb805ad0d679b534d40979c8d8675
Round 7 Decrypted Ciphertext: 8ee0edc8f9b6cf2c9b3ac579ae342f6c
Round 6 Decrypted Ciphertext: 94be5bf4248ff17d41b2a4f1336b4f9d
Round 5 Decrypted Ciphertext: c6db16d37579840e195b13569b734361
Round 4 Decrypted Ciphertext: d53630015c90785f8e5f62520c2777fb
Round 3 Decrypted Ciphertext: 3e1d215109cd1334ac3d9fbadf2122a2
Round 2 Decrypted Ciphertext: 798d48a9b08c130e665f35f2eb182380
Round 1 Decrypted Ciphertext: 6b7d826bf1ff01f35b9ef2d668c86d44
Round 0 Decrypted Ciphertext: 682064617461206d656d626572732061
```

```
Round 1 Ciphertext: bf5694f7d1b0f0999247c9b5d8e2a53a
Round 2 Ciphertext: a0fee0f030f36ffd95e26eef1a8f9526
Round 3 Ciphertext: 6bb62e3b8170f2d574e0740d85142677
Round 4 Ciphertext: 6db7410f9beffc415a66aef87a7f9a86
Round 5 Ciphertext: 1df33796a2116007c16dc5084a25e799
Round 6 Ciphertext: d68ecce0fe1226efc2af6f30fe13701f
Round 7 Ciphertext: 60cfb33fe951f861ccc416ab78bc085d
Round 8 Ciphertext: e267e2097e823cb0c3433d5ff50bec89
Round 9 Ciphertext: a2411934a9d6fc57fdc0fb007eed27c1
Round 0 Ciphertext: 492ee6fffaae17c9afab6d6c9217c758
Round 9 Decrypted Ciphertext: 981327a7f21bcf002c299ae5e586e8cc
Round 8 Decrypted Ciphertext: d0d1474c1f1d317449676fedbf894261
Round 7 Decrypted Ciphertext: f6c9a8c0ba7850e0277f49ddb81af407
Round 6 Decrypted Ciphertext: a482a6ee3b3d95917a3d98c7d50ed333
Round 5 Decrypted Ciphertext: 3cdfe4441532b977bcd08181d9aab342
Round 4 Decrypted Ciphertext: 7f5192f50de0f6e390f83301944d8ad4
Round 3 Decrypted Ciphertext: e00d9ff705992b8d2871e356a1b8abdc
Round 2 Decrypted Ciphertext: 08e7dd803fa107694d9a20ec62b28fd6
Round 1 Decrypted Ciphertext: 7b6fa46776fba57d456def2278a7f1c9
Round 0 Decrypted Ciphertext: 6e6420646174612066756e6374696f6e
Round 1 Ciphertext: b6f9c060bce128157883b8f6f88d3493
Round 2 Ciphertext: e6e151ac6bbf71456627f6583aa72db1
Round 3 Ciphertext: 05b3f304fa2400c86dcf4437a7572dc2
Round 4 Ciphertext: 79d0c8dfba84d351f4c79213b15c70cb
Round 5 Ciphertext: 97739ccd42e489efc0f5ae5a58fc31a5
Round 6 Ciphertext: fa658e08e8c875c200202b2a42910791
Round 7 Ciphertext: bbeeea7ec99f6cb08a384451efe862bc
Round 8 Ciphertext: d86dbe20bd99a636f6bdd75796a9fd3a
Round 9 Ciphertext: 5bbc301320b666e19138cde20854e855
Round 0 Ciphertext: 4a96547b9e1340ac7adebdcf51f144a3
```

```
Round 9 Decrypted Ciphertext: 61ee0e807b7b55b640d1ac07933f2758
Round 8 Decrypted Ciphertext: eadb1b65dc06abf27c9985e5dc2b53d2
Round 7 Decrypted Ciphertext: 2de8f1819ab6c43161831b272f4e9ee6
Round 6 Decrypted Ciphertext: 8869e4062de7c6bcb8b2dcdd698ca4bd
Round 5 Decrypted Ciphertext: b65f4f1ff5c7509fbd48ead3cb73657e
Round 4 Decrypted Ciphertext: 6b361b252c8bd9f33e590fea5f6e6099
Round 3 Decrypted Ciphertext: 8e0842c87ecdd990315ed36c83fba069
Round 2 Decrypted Ciphertext: 4ef86cdc64ed19d1be5fb85b429a3741
Round 1 Decrypted Ciphertext: 72c0f0f01baa7df1afa99e6158c86060
Round 0 Decrypted Ciphertext: 73206f72206d6574686f647320617373
Round 1 Ciphertext: b3c9f58b567ac817108386b5f8e42b5f
Round 2 Ciphertext: c5c379bfbfbf98a81013aaf13ae3f126
Round 3 Ciphertext: 2db31d3b8d9579519482239ba781c809
Round 4 Ciphertext: caccf5fbca1de341ea903b2ab15ca545
Round 5 Ciphertext: 558831bcc242de7ef8f5a00858c7463d
Round 6 Ciphertext: 8e208a29e1c8e81afb56320642dabd1f
Round 7 Ciphertext: 8fee383faa99d325e6ee23d6ef126436
Round 8 Ciphertext: 41588340ccb64fb006ef578f96a9cb1d
Round 9 Ciphertext: b91c653711132d5fbc388f000802487d
Round 0 Ciphertext: 25a59a78ab13894b9e89f4f85108af58
Round 9 Decrypted Ciphertext: 834e5ba44ade1e086dd1eee593698770
Round 8 Decrypted Ciphertext: 73ee2605ad2942748ccb053ddc2b65f5
Round 7 Decrypted Ciphertext: 19e823c0f9b07ba40d557ca02fb4986c
Round 6 Decrypted Ciphertext: fc2ce02724e75b6443c4c5f169c71e33
Round 5 Decrypted Ciphertext: 74a4e26e7561070e8548e481cb4812e6
Round 4 Decrypted Ciphertext: d82a26015c12e9e3200ea6d35f6eb517
Round 3 Decrypted Ciphertext: a608acf7097ca009c813b4c0832d45a2
Round 2 Decrypted Ciphertext: 6dda44cfb0edf03cc86be4f242deebd6
Round 1 Decrypted Ciphertext: 77f0c51bf1319df3c7a9a02258a17fac
Round 0 Decrypted Ciphertext: 6f6369617465642077697468206 16e20
```

```
Round 1 Ciphertext: 36a577fbd113266757d64731f8b6d500
Round 2 Ciphertext: ad649d1330a56a9a8134b9843a38ee34
Round 3 Ciphertext: 1ebde7d481f0f0245294cbe1a7ec8cf7
Round 4 Ciphertext: 606acc929b2c6febc8520bcdb14b9fa9
Round 5 Ciphertext: f15df801a222033e970c0d62588eff65
Round 6 Ciphertext: d39fbd43fed0a4033289b4474252db85
Round 7 Ciphertext: f0769668e98f109acab9270fef7eb6f9
Round 8 Ciphertext: bec569dc7ec84280fcd5c00896b9679d
Round 9 Ciphertext: 94ba84cda9cfb7d063bd9a2a08cee03e
Round 0 Ciphertext: 51525135fa6e3a6c0075e6475160dede
Round 9 Decrypted Ciphertext: aee8ba5ef2028487b254fbcf93a52f33
Round 8 Decrypted Ciphertext: 8c73cc991f574f4476f192badc3bc975
Round 7 Decrypted Ciphertext: 66708d97baa6b81b210278792fd84aa3
Round 6 Decrypted Ciphertext: a193d74d3bff177d8a1b43b0694f78a9
Round 5 Decrypted Ciphertext: d0712bd31501da4eeab149ebcb01abbe
Round 4 Decrypted Ciphertext: 728c1f680d23654902cc96345f798ffb
Round 3 Decrypted Ciphertext: 950656180519297c0e055cba8340015c
Round 2 Decrypted Ciphertext: 057da0633ff7020e594cf7874205f4c4
Round 1 Decrypted Ciphertext: f29c476b7658738380fc61a658f381f3
Round 0 Decrypted Ciphertext: 696e7374616e7469617465642063 6c61
Round 1 Ciphertext: b639fad324b814827780c9d268213035
Round 2 Ciphertext: e675e0e6389e6df32f8761123e2ce346
Round 3 Ciphertext: 05b05e9682ffc9d74be274549532b261
Round 4 Ciphertext: 79f0411584963c337bbfc7f5c7d6ce71
Round 5 Ciphertext: 97bc1571e92a53285e49c5485600bc3e
Round 6 Ciphertext: fae9ccbcda15d7dce2f3acc1997b4d7d
Round 7 Ciphertext: bb5f8a0005254ae5719816f22dbbf121
Round 8 Ciphertext: d889e2b80bd8ada62bcc2e69914e7b62
Round 9 Ciphertext: 5b330f397187133a22c4fbc319cf59f7
Round 0 Ciphertext: 4acfe6ef8a0810c36874cfb7b5570a15
```

```
Round 9 Decrypted Ciphertext: 616131aa2a4a206df32d9a2682a496fa
Round 8 Decrypted Ciphertext: ea3f47fd6a47a062a1e87cdbdbccd58a
Round 7 Decrypted Ciphertext: 2d5991ff560ce2649a234984ed1d0d7b
Round 6 Decrypted Ciphertext: 88e5a6b21f3a64a25a615b36b266ee51
Round 5 Decrypted Ciphertext: b690c6a35e098a5823f481c1c58fe8e5
Round 4 Decrypted Ciphertext: 6b1692ef12993691b1215a0c29e4de23
Round 3 Decrypted Ciphertext: 8e0bef5a0616108f1773e30fb19e3fca
Round 2 Decrypted Ciphertext: 4e6cdd9637cc0567f7ff2f114611f9b6
Round 1 Decrypted Ciphertext: 7200ca4383f34166a0aaef45c86464c6
Round 0 Decrypted Ciphertext: 7373206f72206f626a6563742e000000
plaintext:b'Encapsulation is a way to restrict the direct access to some components of an object, so users cannot access state values for all of the v
ariables of a particular object. Encapsulation can be used to hide both data members and data functions or methods associated with an instantiated cla
ss or object.'
key:b'mysecretpassword'
ciphertext:b"\x82R\x8c\x9a\x024l\xac\x00u\xa5\xf8\xb0\x96\xd8/\x031\xe6@\xfa\xb3P:\xe0M\x0f%\xbbM2\xa3\xba'\xe6\xff\x894P,\x83w\xe6G\x92\x08\x97\xc1\x
ba\x96Q@\x9en\x10\x1c\x0fi\xd0\xfdQ\x17\xc7X\xf6\xedSI\xc64CR\xe0\xf9\x91c\xb0\x17\xc7X\x19.\x8c@\xa7JC,Qi7G\xfeW\x18zq\xcfS{\xabJ\x17l\xf5u7\xb7QW\x8
aQv1\xe6\x9a\xfa4l\x0c\x0fi\x100\xfe\x99\x97d\xf4DT\xd3\x9e\x13>\x0cQ\xde\xaa\xadH\x17\\d\xb2\xa7\xe6\xef1n\x17\x0c\xabt\x0f\xb7HW\x7fXqy\xe6\xb7\xe7n
@p\x94\xa6\x91\xb7\xb0\xf7\x97QJ\xf2\x95\x8d\xab@\x10KQ*\xaa\xf8Q%\x82T\x9c\xa5\xfd@\x9e@\x10]zt\xbdlQ\x17\xc7X\x8bN/@\xabJPlu\xde\xd1\xf8\xfe0\x18\x0
fI.\xe6\xff\xfa\xae\x17\xc9\xaf\xabml\x92\x17\xc7XJ\x96T{\x9e\x13@\xacz\xde\xbd\xcfQ\xf1D\xa3%\xa5\x9ax\xab\x13\x89K\x9e\x89\xf4\xf8Q\x08\xafXQRQ5\xfa
n:l\x00u\xe6GQ`\xde\xdeJ\xcf\xe6\xef\x8a\x08\x10\xc3ht\xcf\xb7\xb5W\n\x15"
Decrypted text:b'Encapsulation is a way to restrict the direct access to some components of an object, so users cannot access state values for all of
the variables of a particular object. Encapsulation can be used to hide both data members and data functions or methods associated with an instantiate
d class or object.'
```

Reports form Vivado:

```
Copyright 1986-2022 Xilinx, Inc. All Rights Reserved. Copyright 2022-2023 Advanced Micro Devices, Inc. All Rights Reserved.
------------------------------------------------------------------------------------------------------------------------------------------------------
| Tool Version : Vivado v.2023.1 (win64) Build 3865809 Sun May  7 15:05:29 MDT 2023
| Date         : Wed Oct 25 11:35:48 2023
| Host         : DESKTOP-RGNOUUR running 64-bit major release  (build 9200)
| Command      : report_bus_skew -warn_on_violation -file aes_bus_skew_routed.rpt -pb aes_bus_skew_routed.pb -rpx aes_bus_skew_routed.rpx
| Design       : aes
| Device       : 7a100t-csg324
| Speed File   : -1  PRODUCTION 1.23 2018-06-13
| Design State : Routed
------------------------------------------------------------------------------------------------------------------------------------------------------

Bus Skew Report

No bus skew constraints
```

```
Copyright 1986-2022 Xilinx, Inc. All Rights Reserved. Copyright 2022-2023 Advanced Micro Devices, Inc. All Rights Reserved.
------------------------------------------------------------------------------------------------------------------------------------------------------
| Tool Version : Vivado v.2023.1 (win64) Build 3865809 Sun May  7 15:05:29 MDT 2023
| Date         : Wed Oct 25 11:35:48 2023
| Host         : DESKTOP-RGNOUUR running 64-bit major release  (build 9200)
| Command      : report_clock_utilization -file aes_clock_utilization_routed.rpt
| Design       : aes
| Device       : 7a100t-csg324
| Speed File   : -1  PRODUCTION 1.23 2018-06-13
| Design State : Routed
------------------------------------------------------------------------------------------------------------------------------------------------------

Clock Utilization Report

Table of Contents
-----------------
1. Clock Primitive Utilization
2. Global Clock Resources
3. Global Clock Source Details
4. Clock Regions: Key Resource Utilization
5. Clock Regions : Global Clock Summary
6. Device Cell Placement Summary for Global Clock g0
7. Clock Region Cell Placement per Global Clock: Region X0Y2
8. Clock Region Cell Placement per Global Clock: Region X1Y2
```

```
1. Clock Primitive Utilization
------------------------------

+----------+------+-----------+-----+------------+--------+
| Type     | Used | Available | LOC | Clock Region | Pblock |
+----------+------+-----------+-----+------------+--------+
| BUFGCTRL |    1 |        32 |   0 |            0 |      0 |
| BUFH     |    0 |        96 |   0 |            0 |      0 |
| BUFIO    |    0 |        24 |   0 |            0 |      0 |
| BUFMR    |    0 |        12 |   0 |            0 |      0 |
| BUFR     |    0 |        24 |   0 |            0 |      0 |
| MMCM     |    0 |         6 |   0 |            0 |      0 |
| PLL      |    0 |         6 |   0 |            0 |      0 |
+----------+------+-----------+-----+------------+--------+


2. Global Clock Resources
-------------------------

+-----------+-----------+-----------------+------------+---------------+-------------+-----------------+-------------+-----------------+--------------+-------+--------------------+---------------+
| Global Id | Source Id | Driver Type/Pin | Constraint | Site          | Clock Region | Load Clock Region | Clock Loads | Non-Clock Loads | Clock Period | Clock | Driver Pin         | Net           |
+-----------+-----------+-----------------+------------+---------------+-------------+-----------------+-------------+-----------------+--------------+-------+--------------------+---------------+
| g0        | src0      | BUFG/O          | None       | BUFGCTRL_X0Y0 | n/a         |               2 |        2990 |               0 |              |       | clk_IBUF_BUFG_inst/O | clk_IBUF_BUFG |
+-----------+-----------+-----------------+------------+---------------+-------------+-----------------+-------------+-----------------+--------------+-------+--------------------+---------------+
* Clock Loads column represents cell count of net connects that connect to a clock pin. Internal cell leaf pins are not considered
** Non-Clock Loads column represents cell count of non-clock pin loads
```

```
3. Global Clock Source Details
------------------------------

+-----------+-----------+-----------------+------------+-----------+-------------+------------+-----------------+--------------------+-------------+----------------+
---------+
| Source Id | Global Id | Driver Type/Pin | Constraint | Site      | Clock Region | Clock Loads | Non-Clock Loads | Source Clock Period | Source Clock | Driver Pin     |
Net      |
+-----------+-----------+-----------------+------------+-----------+-------------+------------+-----------------+--------------------+-------------+----------------+-
---------+
| src0      | g0        | IBUF/O          | None       | IOB_X0Y78 | X0Y1        |          1 |               0 |                    |             | clk_IBUF_inst/O |
clk_IBUF |
+-----------+-----------+-----------------+------------+-----------+-------------+------------+-----------------+--------------------+-------------+----------------+-
---------+
* Clock Loads column represents cell count of net connects that connect to a clock pin. Internal cell leaf pins are not considered
** Non-Clock Loads column represents cell count of non-clock pin loads
```

```
4. Clock Regions: Key Resource Utilization
------------------------------------------

+-------------------+--------------+--------------+--------------+--------------+--------------+--------------+--------------+--------------+--------------+--------------+
-------------+-------------+-------------+
|                   | Global Clock |     BUFRs    |    BUFMRs    |    BUFIOs    |     MMCM     |      PLL     |      GT      |      PCI     |    ILOGIC    |    OLOGIC    |
FF       |     LUTM     |    RAMB18    |    RAMB36    |    DSP48E2   |
+-------------------+------+-------+------+-------+------+-------+------+-------+------+-------+------+-------+------+-------+------+-------+------+-------+------+-------+
------+-------+------+-------+------+-------+------+-------+------+-------+
| Clock Region Name | Used | Avail | Used | Avail | Used | Avail | Used | Avail | Used | Avail | Used | Avail | Used | Avail | Used | Avail | Used | Avail | Used | Avail | Used | Avail |
Used | Avail | Used | Avail | Used | Avail | Used | Avail | Used | Avail |
+-------------------+------+-------+------+-------+------+-------+------+-------+------+-------+------+-------+------+-------+------+-------+------+-------+------+-------+
------+-------+------+-------+------+-------+------+-------+------+-------+
| X0Y0              |    0 |    12 |    0 |     4 |    0 |     2 |    0 |     4 |    0 |     1 |    0 |     1 |    0 |     0 |    0 |     0 |    0 |    50 |    0 |    50 |
0 |  2600 |    0 |   600 |    0 |    20 |    0 |    10 |    0 |    20 |
| X1Y0              |    0 |    12 |    0 |     0 |    0 |     0 |    0 |     0 |    0 |     0 |    0 |     0 |    0 |     0 |    4 |     0 |    0 |     0 |    0 |     0 |
0 |  1500 |    0 |   550 |    0 |    40 |    0 |    20 |    0 |    40 |
| X0Y1              |    0 |    12 |    0 |     4 |    0 |     2 |    0 |     4 |    0 |     1 |    0 |     1 |    0 |     0 |    0 |     0 |    0 |    50 |    0 |    50 |
0 |  2000 |    0 |   600 |    0 |    20 |    0 |    10 |    0 |    20 |
| X1Y1              |    0 |    12 |    0 |     4 |    0 |     2 |    0 |     4 |    0 |     1 |    0 |     1 |    0 |     0 |    0 |     0 |    0 |    50 |    0 |    50 |
0 |  1900 |    0 |   650 |    0 |    60 |    0 |    30 |    0 |    40 |
| X0Y2              |    1 |    12 |    0 |     4 |    0 |     2 |    0 |     4 |    0 |     1 |    0 |     1 |    0 |     0 |    0 |     0 |    0 |    50 |    0 |    50 |
2936 |  2000 |  925 |   600 |    0 |    20 |    0 |    10 |    0 |    20 |
| X1Y2              |    1 |    12 |    0 |     4 |    0 |     2 |    0 |     4 |    0 |     1 |    0 |     1 |    0 |     0 |    0 |     0 |    0 |    50 |    0 |    50 |
54 |  1900 |    8 |   650 |    0 |    60 |    0 |    30 |    0 |    40 |
| X0Y3              |    0 |    12 |    0 |     4 |    0 |     2 |    0 |     4 |    0 |     1 |    0 |     1 |    0 |     0 |    0 |     0 |    0 |    50 |    0 |    50 |
0 |  2600 |    0 |   600 |    0 |    20 |    0 |    10 |    0 |    20 |
| X1Y3              |    0 |    12 |    0 |     0 |    0 |     0 |    0 |     0 |    0 |     0 |    0 |     0 |    4 |     0 |    1 |     0 |    0 |     0 |    0 |     0 |
0 |  1350 |    0 |   500 |    0 |    30 |    0 |    15 |    0 |    40 |
+-------------------+------+-------+------+-------+------+-------+------+-------+------+-------+------+-------+------+-------+------+-------+------+-------+------+-------+
------+-------+------+-------+------+-------+------+-------+------+-------+
* Global Clock column represents track count; while other columns represents cell counts
```

```
5. Clock Regions : Global Clock Summary
---------------------------------------

All Modules
+----+----+----+
|    | X0 | X1 |
+----+----+----+
| Y3 |  0 |  0 |
| Y2 |  0 |  0 |
| Y1 |  0 |  0 |
| Y0 |  0 |  0 |
+----+----+----+


6. Device Cell Placement Summary for Global Clock g0
----------------------------------------------------

+-----------+-----------------+-------------------+-------+--------------+--------------+------------+----------+---------------+----------+----------------+
| Global Id | Driver Type/Pin | Driver Region (D) | Clock | Period (ns) | Waveform (ns) | Slice Loads | IO Loads | Clocking Loads | GT Loads | Net            |
+-----------+-----------------+-------------------+-------+--------------+--------------+------------+----------+---------------+----------+----------------+
| g0        | BUFG/O          | n/a               |       |              |              |       2990 |        0 |             0 |        0 | clk_IBUF_BUFG |
+-----------+-----------------+-------------------+-------+--------------+--------------+------------+----------+---------------+----------+----------------+
* Slice Loads column represents load cell count of all cell types other than IO, GT and clock resources
** IO Loads column represents load cell count of IO types
*** Clocking Loads column represents load cell count that are clock resources (global clock buffer, MMCM, PLL, etc)
**** GT Loads column represents load cell count of GT types


+----+-------+-----+-----------------------+
|    | X0    | X1  | HORIZONTAL PROG DELAY |
+----+-------+-----+-----------------------+
| Y3 |     0 |   0 |                     - |
| Y2 |  2936 |  54 |                     0 |
| Y1 |     0 |   0 |                     - |
| Y0 |     0 |   0 |                     - |
+----+-------+-----+-----------------------+
```

```
7. Clock Region Cell Placement per Global Clock: Region X0Y2
------------------------------------------------------------

+-----------+-------+-----------------+------------+--------------+-----------------+------+-------------+------+-----+------+-----+-----+----------+----------------+
| Global Id | Track | Driver Type/Pin | Constraint | Clock Loads | Non-Clock Loads | FF   | Memory LUTs | RAMB | DSP | GT   | MMCM | PLL | Hard IP | Net            |
+-----------+-------+-----------------+------------+--------------+-----------------+------+-------------+------+-----+------+-----+-----+----------+----------------+
| g0        | n/a   | BUFG/O          | None       |         2936 |               0 | 2936 |           0 |    0 |   0 |    0 |   0 |   0 |        0 | clk_IBUF_BUFG |
+-----------+-------+-----------------+------------+--------------+-----------------+------+-------------+------+-----+------+-----+-----+----------+----------------+
* Clock Loads column represents cell count of net connects that connect to a clock pin. Internal cell leaf pins are not considered
** Non-Clock Loads column represents cell count of non-clock pin loads
*** Columns FF, LUTRAM, RAMB through 'Hard IP' represents load cell counts
```

```
8. Clock Region Cell Placement per Global Clock: Region X1Y2
------------------------------------------------------------

+-----------+-------+-----------------+------------+-------------+-----------------+----+-------------+------+-----+----+------+-----+---------+---------------+
| Global Id | Track | Driver Type/Pin | Constraint | Clock Loads | Non-Clock Loads | FF | Memory LUTs | RAMB | DSP | GT | MMCM | PLL | Hard IP | Net           |
+-----------+-------+-----------------+------------+-------------+-----------------+----+-------------+------+-----+----+------+-----+---------+---------------+
| g0        | n/a   | BUFG/O          | None       |          54 |               0 | 54 |           0 |    0 |   0 |  0 |    0 |   0 |       0 | clk_IBUF_BUFG |
+-----------+-------+-----------------+------------+-------------+-----------------+----+-------------+------+-----+----+------+-----+---------+---------------+
* Clock Loads column represents cell count of net connects that connect to a clock pin. Internal cell leaf pins are not considered
** Non-Clock Loads column represents cell count of non-clock pin loads
*** Columns FF, LUTRAM, RAMB through 'Hard IP' represents load cell counts



# Location of BUFG Primitives
set_property LOC BUFGCTRL_X0Y0 [get_cells clk_IBUF_BUFG_inst]

# Location of IO Primitives which is load of clock spine

# Location of clock ports
set_property LOC IOB_X0Y78 [get_ports clk]

# Clock net "clk_IBUF_BUFG" driven by instance "clk_IBUF_BUFG_inst" located at site "BUFGCTRL_X0Y0"
#startgroup
create_pblock {CLKAG_clk_IBUF_BUFG}
add_cells_to_pblock [get_pblocks  {CLKAG_clk_IBUF_BUFG}] [get_cells -filter { PRIMITIVE_GROUP != I/O && IS_PRIMITIVE==1 && PRIMITIVE_LEVEL !=INTERNAL } -of_object
[get_pins -leaf -filter {DIRECTION==IN} -of_objects [get_nets {clk_IBUF_BUFG}]]]
resize_pblock [get_pblocks {CLKAG_clk_IBUF_BUFG}] -add {CLOCKREGION_X0Y2:CLOCKREGION_X0Y2 CLOCKREGION_X1Y2:CLOCKREGION_X1Y2}
#endgroup
```

```
Copyright 1986-2022 Xilinx, Inc. All Rights Reserved. Copyright 2022-2023 Advanced Micro Devices, Inc. All Rights Reserved.
----------------------------------------------------------------------------------------------------------------------------------------------------------
| Tool Version : Vivado v.2023.1 (win64) Build 3865809 Sun May  7 15:05:29 MDT 2023
| Date         : Wed Oct 25 11:34:57 2023
| Host         : DESKTOP-RGNOUUR running 64-bit major release  (build 9200)
| Command      : report_control_sets -verbose -file aes_control_sets_placed.rpt
| Design       : aes
| Device       : xc7a100t
----------------------------------------------------------------------------------------------------------------------------------------------------------

Control Set Information

Table of Contents
-----------------
1. Summary
2. Histogram
3. Flip-Flop Distribution
4. Detailed Control Set Information
```

1. Summary

Status	Count
Total control sets	44
Minimum number of control sets	44
Addition due to synthesis replication	0
Addition due to physical synthesis replication	0
Unused register locations in slices containing registers	26

* Control sets can be merged at opt_design using control_set_merge or merge_equivalent_drivers
** Run report_qor_suggestions for automated merging and remapping suggestions

2. Histogram

Fanout	Count
Total control sets	44
>= 0 to < 4	1
>= 4 to < 6	4
>= 6 to < 8	0
>= 8 to < 10	1
>= 10 to < 12	0
>= 12 to < 14	0
>= 14 to < 16	0
>= 16	38

* Control sets can be remapped at either synth_design or opt_design

3. Flip-Flop Distribution

Clock Enable	Synchronous Set/Reset	Asynchronous Set/Reset	Total Registers	Total Slices
No	No	No	0	0
No	No	Yes	147	53
No	Yes	No	0	0
Yes	No	No	0	0
Yes	No	Yes	2843	1100
Yes	Yes	No	0	0

4. Detailed Control Set Information

Clock Signal	Enable Signal	Set/Reset Signal	Slice Load Count	Bel Load Count	Bels / Slice
clk_IBUF_BUFG	config_we4_out	core/keymem/reset_n	1	2	2.00
clk_IBUF_BUFG	core/enc_block/round_ctr_we	core/keymem/reset_n	2	4	2.00
clk_IBUF_BUFG	core/dec_block/dec_ctrl_new[0]	core/keymem/reset_n	2	4	2.00
clk_IBUF_BUFG	core/keymem/FSM_onehot_key_mem_ctrl_reg[3]_i_1_n_0	core/keymem/reset_n	1	4	4.00
clk_IBUF_BUFG	core/keymem/round_ctr_we	core/keymem/reset_n	2	5	2.50
clk_IBUF_BUFG	core/keymem/rcon_we	core/keymem/reset_n	2	8	4.00
clk_IBUF_BUFG	core/dec_block/block_w1_we	core/keymem/reset_n	20	32	1.60
clk_IBUF_BUFG	core/dec_block/block_w3_we	core/keymem/reset_n	18	32	1.78
clk_IBUF_BUFG	core/dec_block/block_w2_we	core/keymem/reset_n	19	32	1.68
clk_IBUF_BUFG	core/dec_block/block_w0_we	core/keymem/reset_n	19	32	1.68
clk_IBUF_BUFG	key_reg[1][31]_i_1_n_0	core/keymem/reset_n	14	32	2.29
clk_IBUF_BUFG	key_reg[4][31]_i_1_n_0	core/keymem/reset_n	6	32	5.33
clk_IBUF_BUFG	key_reg[5][31]_i_1_n_0	core/keymem/reset_n	11	32	2.91
clk_IBUF_BUFG	block_reg[0][31]_i_1_n_0	core/keymem/reset_n	10	32	3.20
clk_IBUF_BUFG	block_reg[3][31]_i_1_n_0	core/keymem/reset_n	12	32	2.67
clk_IBUF_BUFG	key_reg[6][31]_i_1_n_0	core/keymem/reset_n	14	32	2.29
clk_IBUF_BUFG	key_reg[7][31]_i_1_n_0	core/keymem/reset_n	16	32	2.00
clk_IBUF_BUFG	block_reg[1][31]_i_1_n_0	core/keymem/reset_n	10	32	3.20
clk_IBUF_BUFG	core/enc_block/block_w1_we	core/keymem/reset_n	17	32	1.88
clk_IBUF_BUFG	core/enc_block/block_w2_we	core/keymem/reset_n	19	32	1.68
clk_IBUF_BUFG	core/enc_block/block_w3_we	core/keymem/reset_n	21	32	1.52
clk_IBUF_BUFG	core/enc_block/block_w0_we	core/keymem/reset_n	16	32	2.00
clk_IBUF_BUFG	block_reg[2][31]_i_1_n_0	core/keymem/reset_n	11	32	2.91
clk_IBUF_BUFG	key_reg[0][31]_i_1_n_0	core/keymem/reset_n	9	32	3.56
clk_IBUF_BUFG	key_reg[3][31]_i_1_n_0	core/keymem/reset_n	13	32	2.46
clk_IBUF_BUFG	key_reg[2][31]_i_1_n_0	core/keymem/reset_n	10	32	3.20
clk_IBUF_BUFG	core/keymem/key_mem[1][127]_i_1_n_0	core/keymem/reset_n	33	128	3.88
clk_IBUF_BUFG	core/keymem/key_mem[4][127]_i_1_n_0	core/keymem/reset_n	39	128	3.28
clk_IBUF_BUFG	core/keymem/key_mem[9][127]_i_1_n_0	core/keymem/reset_n	81	128	1.58
clk_IBUF_BUFG	core/keymem/key_mem[7][127]_i_1_n_0	core/keymem/reset_n	62	128	2.06
clk_IBUF_BUFG	core/keymem/key_mem[8][127]_i_1_n_0	core/keymem/reset_n	42	128	3.05
clk_IBUF_BUFG	core/keymem/key_mem[5][127]_i_1_n_0	core/keymem/reset_n	39	128	3.28
clk_IBUF_BUFG	core/keymem/key_mem[11][127]_i_1_n_0	core/keymem/reset_n	36	128	3.56
clk_IBUF_BUFG	core/keymem/key_mem[14][127]_i_1_n_0	core/keymem/reset_n	46	128	2.78
clk_IBUF_BUFG	core/keymem/key_mem[2][127]_i_1_n_0	core/keymem/reset_n	32	128	4.00
clk_IBUF_BUFG	core/keymem/key_mem[12][127]_i_1_n_0	core/keymem/reset_n	41	128	3.12
clk_IBUF_BUFG	core/keymem/prev_key1_we1_out	core/keymem/reset_n	107	128	1.20
clk_IBUF_BUFG	core/keymem/key_mem[13][127]_i_1_n_0	core/keymem/reset_n	45	128	2.84
clk_IBUF_BUFG	core/keymem/key_mem[3][127]_i_1_n_0	core/keymem/reset_n	38	128	3.37
clk_IBUF_BUFG	core/keymem/key_mem	core/keymem/reset_n	30	128	4.27
clk_IBUF_BUFG	core/keymem/key_mem[10][127]_i_1_n_0	core/keymem/reset_n	38	128	3.37
clk_IBUF_BUFG	core/keymem/prev_key0_we2_out	core/keymem/reset_n	58	128	2.21
clk_IBUF_BUFG	core/keymem/key_mem[6][127]_i_1_n_0	core/keymem/reset_n	38	128	3.37
clk_IBUF_BUFG		core/keymem/reset_n	53	147	2.77

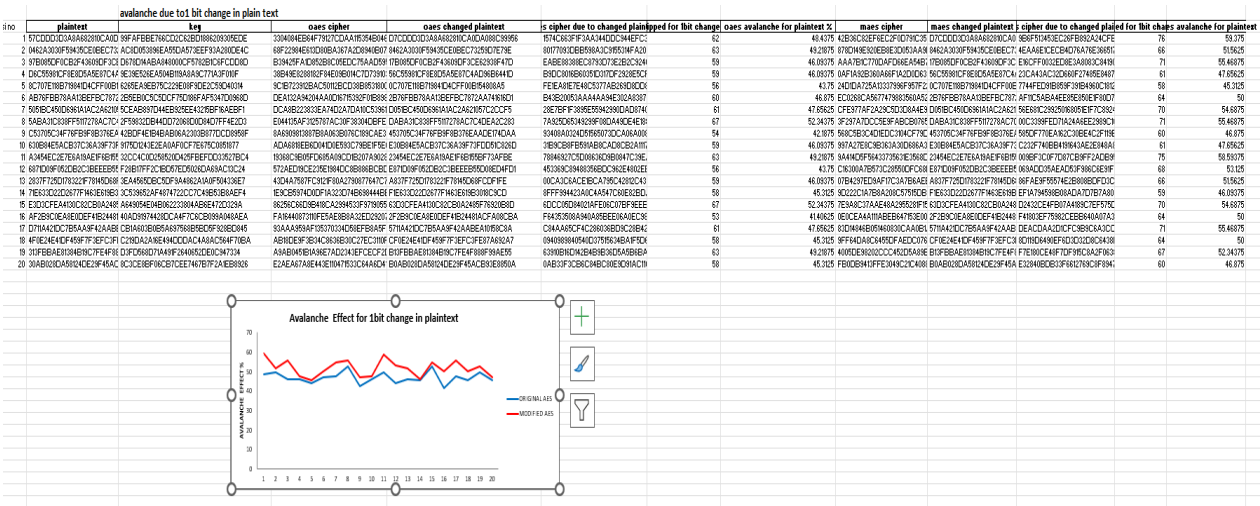
Design Route Status

	# nets
# of logical nets	5920
# of nets not needing routing	795
# of internally routed nets	795
# of routable nets	5125
# of fully routed nets	5125
# of nets with routing errors	0

```
Copyright 1986-2022 Xilinx, Inc. All Rights Reserved. Copyright 2022-2022 Advanced Micro Devices, Inc. All Rights Reserved.
------------------------------------------------------------------------------------------------------------------------
| Tool Version : Vivado v.2023.1 (win64) Build 3865809 Sun May  7 15:05:29 MDT 2023
| Date         : Wed Oct 25 11:35:41 2023
| Host         : DESKTOP-RGNOUUR running 64-bit major release  (build 9200)
| Command      : report_drc -file aes_drc_routed.rpt -pb aes_drc_routed.pb -rpx aes_drc_routed.rpx
| Design       : aes
| Device       : xc7a100tcsg324-1
| Speed File   : -1
| Design State : Fully Routed
------------------------------------------------------------------------------------------------------------------------

Report DRC

Table of Contents
-----------------
1. REPORT SUMMARY
2. REPORT DETAILS

1. REPORT SUMMARY
-----------------
            Netlist: netlist
          Floorplan: design_1
      Design limits: <entire design considered>
           Ruledeck: default
             Max violations: <unlimited>
             Violations found: 3
+----------+------------------+-------------------------------------------------------+------------+
| Rule     | Severity         | Description                                           | Violations |
+----------+------------------+-------------------------------------------------------+------------+
| NSTD-1   | Critical Warning | Unspecified I/O Standard                              | 1          |
| UCIO-1   | Critical Warning | Unconstrained Logical Port                            | 1          |
| CFGBVS-1 | Warning          | Missing CFGBVS and CONFIG_VOLTAGE Design Properties   | 1          |
+----------+------------------+-------------------------------------------------------+------------+
```

```
2. REPORT DETAILS
-----------------
NSTD-1#1 Critical Warning
Unspecified I/O Standard
76 out of 76 logical ports use I/O standard (IOSTANDARD) value 'DEFAULT', instead of a user assigned specific value. This may cause I/O contention or incompatibility with the board power
or connectivity affecting performance, signal integrity or in extreme cases cause damage to the device or the components to which it is connected. To correct this violation, specify all
I/O standards. This design will fail to generate a bitstream unless all logical ports have a user specified I/O standard value defined. To allow bitstream creation with unspecified I/O
standard values (not recommended), use this command: set_property SEVERITY {Warning} [get_drc_checks NSTD-1].  NOTE: When using the Vivado Runs infrastructure (e.g. launch_runs Tcl
command), add this command to a .tcl file and add that file as a pre-hook for write_bitstream step for the implementation run. Problem ports: address[7:0], clk, cs, read_data[31:0],
reset_n, we, write_data[31:0].
Related violations: <none>

UCIO-1#1 Critical Warning
Unconstrained Logical Port
76 out of 76 logical ports have no user assigned specific location constraint (LOC). This may cause I/O contention or incompatibility with the board power or connectivity affecting
performance, signal integrity or in extreme cases cause damage to the device or the components to which it is connected. To correct this violation, specify all pin locations. This design
will fail to generate a bitstream unless all logical ports have a user specified site LOC constraint defined.  To allow bitstream creation with unspecified pin locations (not recommended),
use this command: set_property SEVERITY {Warning} [get_drc_checks UCIO-1].  NOTE: When using the Vivado Runs infrastructure (e.g. launch_runs Tcl command), add this command to a .tcl file
and add that file as a pre-hook for write_bitstream step for the implementation run.  Problem ports: address[7:0], clk, cs, read_data[31:0], reset_n, we, write_data[31:0].
Related violations: <none>

CFGBVS-1#1 Warning
Missing CFGBVS and CONFIG_VOLTAGE Design Properties
Neither the CFGBVS nor CONFIG_VOLTAGE voltage property is set in the current_design.  Configuration bank voltage select (CFGBVS) must be set to VCCO or GND, and CONFIG_VOLTAGE must be set
to the correct configuration voltage, in order to determine the I/O voltage support for the pins in bank 0.  It is suggested to specify these either using the 'Edit Device Properties'
function in the GUI or directly in the XDC file using the following syntax:

 set_property CFGBVS value1 [current_design]
 #where value1 is either VCCO or GND

 set_property CONFIG_VOLTAGE value2 [current_design]
 #where value2 is the voltage provided to configuration bank 0

Refer to the device configuration user guide for more information.
Related violations: <none>
```

Benchmark Analysis:



Chapter 5
Discussions

In this section, we delve into a comprehensive analysis of our project, focusing on the performance and security implications of our modified AES hardware accelerator. Our findings shed light on the following key points:

1. **Performance Comparison**: Our project demonstrates the significant performance gains achieved through the implementation of our modified AES hardware accelerator. In direct comparison to software-based implementations, we observed notable improvements in data encryption and decryption speed. The hardware accelerator proved to be more efficient, making it an attractive solution for scenarios where real-time encryption is critical.

2. **Security Considerations**: Our customized AES algorithm incorporates additional security measures, enhancing its resistance to attacks. By introducing the "Transpose" operation and leveraging a constant Initialization Vector (IV), we have bolstered the algorithm's security profile. However, it's essential to acknowledge that while our modifications increase security, they also affect performance to some extent. Striking the right balance between security and speed is a consideration for future work.

3. **Real-World Applicability**: The practical applicability of our modified AES hardware accelerator extends to a range of real-world use cases. It holds promise in secure communication systems, data storage, and encryption for sensitive information. The high-speed encryption capabilities make it suitable for applications where timely data protection is paramount.

4.**Hardware vs. Software Trade-offs**: Our project has illuminated the trade-offs between implementing AES in hardware (FPGA/ASIC) versus traditional software solutions.

While hardware accelerators offer superior speed and efficiency, they may require more significant initial investments in terms of development and hardware resources. This analysis aids organizations in making informed decisions when choosing between hardware and software encryption methods.

5. **Challenges and Limitations**: We acknowledge that our project encountered some challenges, including resource constraints on the chosen hardware platform and certain compatibility issues. These challenges emphasize the need for continued research and refinement in this field.

6. **Future Directions**: Looking ahead, there are opportunities for further research and development. Possible future directions include optimizing the hardware accelerator, exploring additional security features, and investigating integration with existing cryptographic systems.

7. **Contribution to the Field**: Our work contributes to the evolving landscape of cryptography by offering an innovative approach to high-speed block cipher hardware acceleration. It addresses the pressing concerns related to data security and privacy in our digital age.

Conclusions

In this project, we have embarked on a journey to design and implement a high-speed block cipher hardware accelerator, focusing on a modified version of the Advanced Encryption Standard (AES). Our objective was to enhance the efficiency and security of data encryption and decryption, addressing the growing need for real-time, robust cryptographic solutions.

Through rigorous experimentation and analysis, we have demonstrated the superior performance of our modified AES hardware accelerator when compared to traditional software implementations. The accelerator exhibited remarkable speed and efficiency gains, making it a compelling choice for applications where timely data protection is imperative.

Furthermore, our incorporation of additional security measures, such as the "Transpose" operation and the use of a constant Initialization Vector (IV), underscores our commitment to strengthening data security.

This project contributes to the ever-evolving field of cryptography, offering an innovative approach to high-speed block cipher hardware acceleration. It also aligns with broader societal concerns regarding data privacy and security in the digital age.

As we conclude this endeavor, we recognize the potential for further research and refinement in the field, paving the way for more efficient and secure cryptographic solutions. Our work underscores the critical importance of bridging the gap between data security and speed, and it serves as a stepping stone toward a safer and more efficient digital future.

References

1. Advanced Encryption Standard by NIST (National Institute of Standards and Technology) . Source : https://www.google.com/search?q=standard%20 aes%20 nist

2. Schneier on Security , blogs and reports. Source : https://www.schneier.com/academic/aes/

Acknowledgements

We would like to express our sincere appreciation and gratitude to those who contributed to the successful completion of this research project. Their support, expertise, and encouragement were invaluable in our endeavor to explore the **High speed block cipher hardware Accelerator : Designing and Implementation of hardware accelerators for block ciphers using FPGA or ASIC by implementing modified AES**

We extend our deepest gratitude to our research advisor, **Dr.Hemraj S. Lamkuche** whose guidance and unwavering support enriched the quality of this study.

We acknowledge the assistance and resources provided by the **VIT Bhopal university** that facilitated our research.