International Journal of Advanced Research and Multidisciplinary Trends (IJARMT)
An International Open Access, Peer-Reviewed Refereed Journal
Impact Factor: 6.4 Website: https://ijarmt.com ISSN No.: 3048-9458

Optimization Delay of Blowfish Symmetric-Key Block Cipher using Modified Carry Select Adder

Sanjeet Kumar
M. Tech. Scholar, Department of Electronics and Communication Engineering, SIRT, Bhopal
Dr. Shalini Sahay
Associate Professor, Department of Electronics and Communication Engineering, SIRT, Bhopal

Abstract

The Blowfish symmetric-key block cipher is a widely used encryption algorithm known for its simplicity, flexibility, and strong security. However, in hardware-based cryptographic systems, speed and efficiency are critical factors, especially for real-time applications. This paper presents an optimized hardware implementation of the Blowfish cipher using a Modified Carry Select Adder (MCSA) to reduce computational delay during encryption and decryption processes. The Carry Select Adder, known for its high-speed performance, is further enhanced by integrating Binary to Excess-1 Converter (BEC) logic to minimize area and power consumption while maximizing speed. The Blowfish algorithm's key scheduling and data processing modules, which involve intensive arithmetic operations, are redesigned using MCSA architecture. The proposed design is modeled in Verilog and synthesized using Xilinx FPGA tools. Comparative analysis is performed against traditional adder implementations in terms of delay, area utilization, and power efficiency. Simulation results demonstrate a significant reduction in propagation delay and improved throughput without compromising the security strength of the cipher. This optimization makes the Blowfish cipher more suitable for applications requiring high-speed and low-power encryption, such as embedded systems, wireless networks, and IoT devices. The proposed approach contributes to the development of efficient and secure hardware-based cryptographic solutions.

Keywords: Blowfish, Symmetric Key, Modified Carry Select Adder

INTRODUCTION

One security measure that is mainly utilized to maintain data secrecy is encryption. It is a mathematical process that jumbles information that needs to be protected (plaintext) into a format that is difficult for computers or unauthorized individuals to understand (ciphertext). The plaintext seems random after being converted to ciphertext and does not provide any information about the original data's content. After encryption, nothing can be deciphered by a

International Journal of Advanced Research and Multidisciplinary Trends (IJARMT)
An International Open Access, Peer-Reviewed Refereed Journal
Impact Factor: 6.4 Website: https://ijarmt.com ISSN No.: 3048-9458

human or a machine reading the data in its encrypted version to learn more about the original material's substance.

The process of encryption is reversible. It is only helpful when it is possible to restore encrypted data (ciphertext) to its original, unencrypted state (plaintext). The encrypted data is deemed illegible and unusable if it cannot be reversed. Decryption is the term used to describe this reverse process. An encryption data (ciphertext) can be reversed back to its original content (plaintext) using a similar decryption method.

Key symmetry Encryption: The encryption and decryption keys for this kind of communication are the identical, and both the sender and the recipient have the key before they begin communication. The keys used for encryption and decryption might be the same, or switching between two keys could be simple.

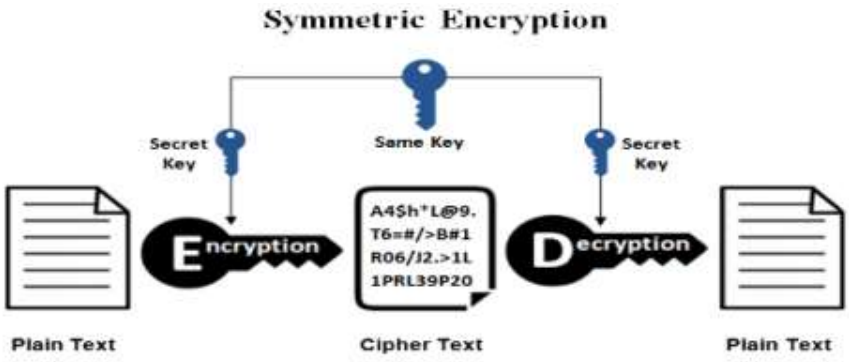

Figure 1: Symmetric Encryption

BLOWFISH SYMMETRIC-KEY BLOCK CIPHER

The Blowfish cipher is a symmetric-key block cipher developed by Bruce Schneier in 1993 as a fast, secure alternative to existing encryption algorithms like DES. It is a Feistel-based cipher that operates on 64-bit blocks of data and supports variable key lengths ranging from 32 bits to 448 bits, offering flexibility in terms of security and performance.

Blowfish consists of two main components: the key expansion and the data encryption process. The key expansion step transforms a key of up to 448 bits into 18 32-bit subkeys (P-array) and four substitution boxes (S-boxes**)** containing 256 entries each. This phase is computationally intensive, making it resistant to brute-force and dictionary attacks.

The encryption process uses a 16-round Feistel network**,** where the 64-bit input block is split into two 32-bit halves. In each round, one half is passed through a complex **F-function**, which

International Journal of Advanced Research and Multidisciplinary Trends (IJARMT)
An International Open Access, Peer-Reviewed Refereed Journal
Impact Factor: 6.4 Website: https://ijarmt.com ISSN No.: 3048-9458

uses the S-boxes, modular addition, and XOR operations, and the result is XORed with the other half. The halves are then swapped, and the process repeats. After 16 rounds, a final transformation involving the P-array completes the encryption.

The F-function is a non-linear transformation that plays a critical role in providing confusion and diffusion, making it difficult for attackers to reverse the encryption without the key.

Blowfish is notable for its speed**,** especially on 32-bit microprocessors, and its unpatented, royalty-free nature, which has led to its wide adoption in various applications such as file encryption tools, network security systems, and embedded devices. However, due to its 64-bit block size, Blowfish is less suited for modern applications that handle large volumes of data. Despite this limitation, it remains a popular choice for lightweight encryption in legacy and resource-constrained systems.

PROPOSED METHODOLOGY

In total around 4KB of memory is required to store all the entries within the P-array and all the entries from the four Sboxes. The Feistel structure along with the functionality of the F-function is summarized in Figures 2 and 3.

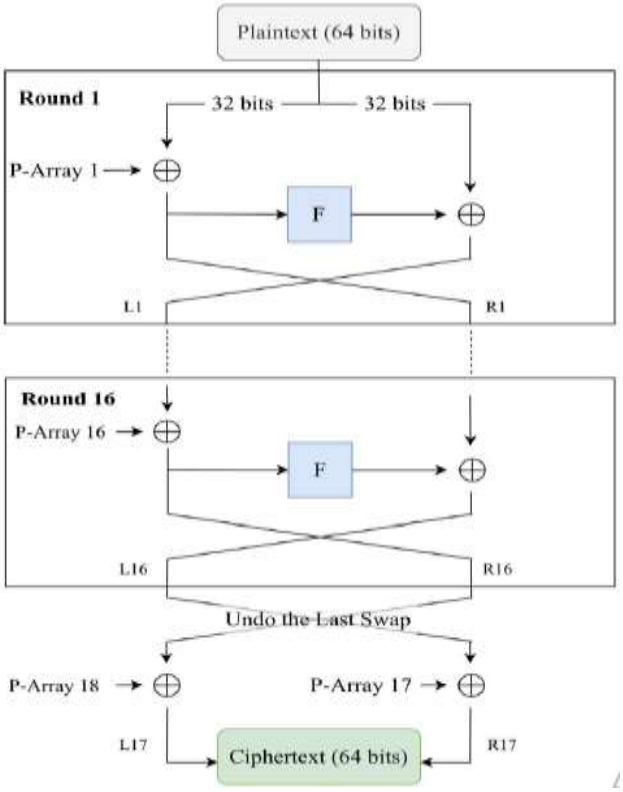

Figure 2: Sixteen Round Feistel Structure of Blowfish (encryption)

International Journal of Advanced Research and Multidisciplinary Trends (IJARMT)
An International Open Access, Peer-Reviewed Refereed Journal
Impact Factor: 6.4 Website: https://ijarmt.com ISSN No.: 3048-9458

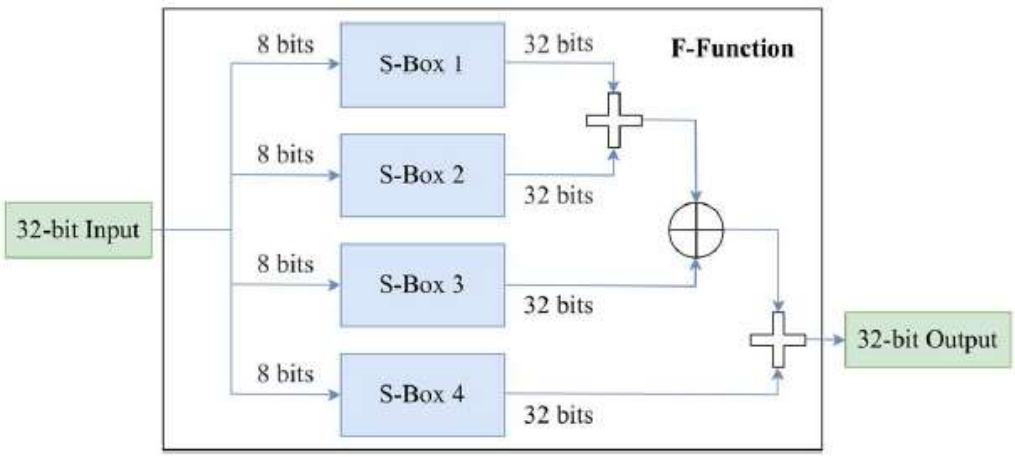

Figure 3: F-function within a single round of Blowfish

Encryption occurs through a 16-round Feistel structure with the addition of a final output whitening round which reverses the last swap and performs an XOR operation to get the output. Decryption is easily done through Blowfish by reversing the order in which the P-array entries are used. For encryption, entries are used in the order from 1 to 18 and for decryption, entries are used in the order 18 down to 1. Each round of the Feistel network utilizes an F-function. The F-function takes a 32-bit input and splits that up into 8-bit portions which are used as inputs for each S-box.

The MCSLA is consist of reduce full adder (RFA). RFA is consist of multiplexer is shown in fig. 4.

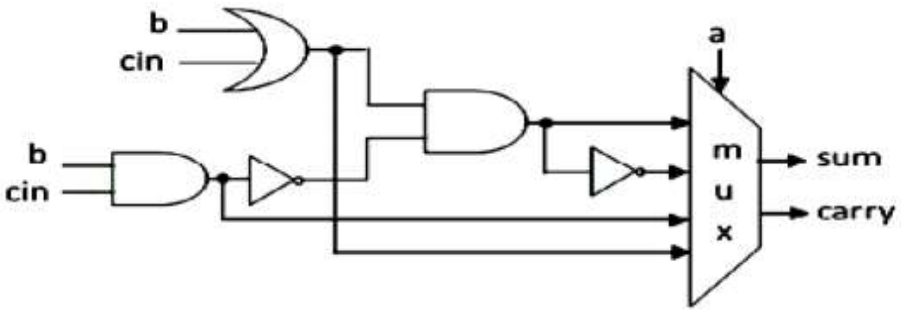

Figure 4: RFA Circuit

In the MCSLA, the full adder circuit is greatly reduced. The number of gate counts is reduced in FA circuit. The MCSLA is circuit diagram represent in fig. 4. MCSLA is a 4×4 circuit that have 4 input A, B, C, D and 4 output P, Q, R, S i.e. P = (B+C)(BC)', Q = (B+C)'+BC, R = BC

International Journal of Advanced Research and Multidisciplinary Trends (IJARMT)
An International Open Access, Peer-Reviewed Refereed Journal
Impact Factor: 6.4 Website: https://ijarmt.com ISSN No.: 3048-9458

and S = B+C. if A = 0 than output of the RMCA is P and Q but if A = 1 than output of the MCSLA is R and S.

SIMULATION RESULT

VHDL is very adaptive, owing to its architecture, allowing designers, electronic design automation companies and the semiconductor industry to experiment with new language concepts to ensure good design tolls and data interoperability.

Having designed the various DSP configurations, we now proceed to the software synthesis of this designs using VHDL. In the following sections, we have established the desired filter outputs using separate VHDL codes for every design. The codes of the designs have been shown in the Appendix. The structure so was discussed successfully implemented or synthesized on XILINX ISE design suite 14.5i.

Figure 5: 16-bit Modified Carry Select Adder

p_array1 Project Status (05/29/2025 - 21:08:53)			
Project File:	mtechproject.xise	**Parser Errors:**	No Errors
Module Name:	RFA_16bit	**Implementation State:**	Synthesized
Target Device:	xc3s50-5vq100	**• Errors:**	No Errors
Product Version:	ISE 14.5	**• Warnings:**	2 Warnings (2 new)
Design Goal:	Balanced	**• Routing Results:**	
Design Strategy:	Xilinx Default (unlocked)	**• Timing Constraints:**	
Environment:	System Settings	**• Final Timing Score:**	

Device Utilization Summary (estimated values)			[-]
Logic Utilization	**Used**	**Available**	**Utilization**
Number of Slices	18	768	2%
Number of 4 input LUTs	31	1536	2%
Number of bonded IOBs	49	63	77%

Figure 6: Utilization Summary of Modified Carry Select Adder

International Journal of Advanced Research and Multidisciplinary Trends (IJARMT)
An International Open Access, Peer-Reviewed Refereed Journal
Impact Factor: 6.4 Website: https://ijarmt.com ISSN No.: 3048-9458

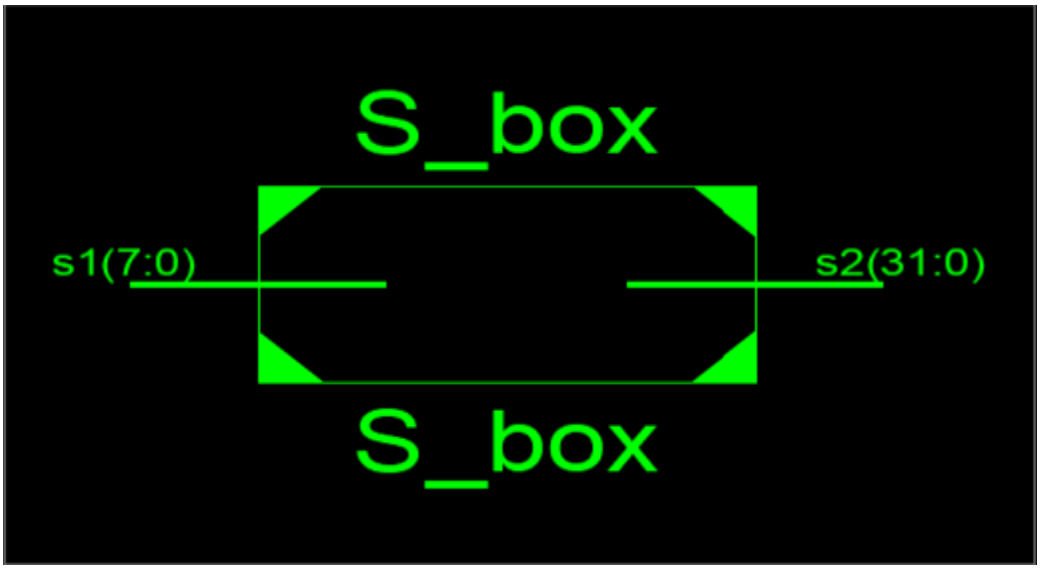

Figure 7: S-Box

p_array1 Project Status (05/29/2025 - 21:11:49)			
Project File:	mtechproject.xise	**Parser Errors:**	No Errors
Module Name:	S_box	**Implementation State:**	Synthesized
Target Device:	xc3s50-5vq100	**• Errors:**	No Errors
Product Version:	ISE 14.5	**• Warnings:**	No Warnings
Design Goal:	Balanced	**• Routing Results:**	
Design Strategy:	Xilinx Default (unlocked)	**• Timing Constraints:**	
Environment:	System Settings	**• Final Timing Score:**	

Device Utilization Summary (estimated values)			[-]
Logic Utilization	**Used**	**Available**	**Utilization**
Number of Slices	4	768	0%
Number of 4 input LUTs	7	1536	0%
Number of bonded IOBs	40	63	63%

Figure 8: Utilization Summary of S-Box

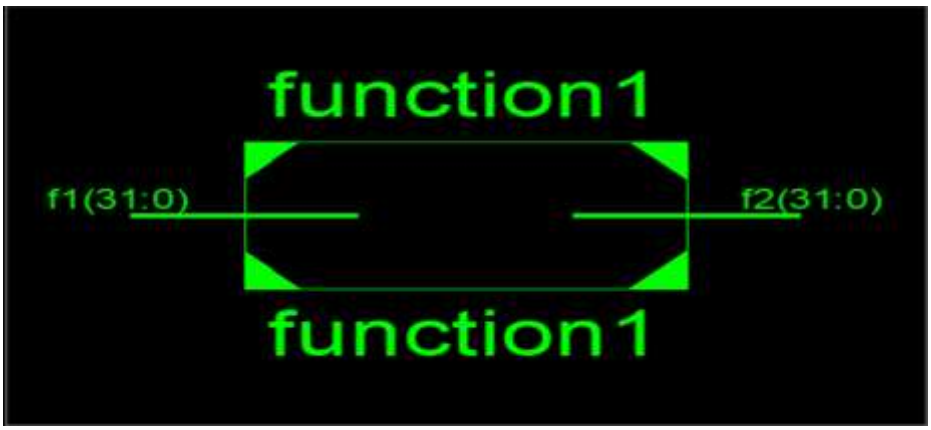

Figure 9: Function

International Journal of Advanced Research and Multidisciplinary Trends (IJARMT)
An International Open Access, Peer-Reviewed Refereed Journal
Impact Factor: 6.4 Website: https://ijarmt.com ISSN No.: 3048-9458

p_array1 Project Status (05/29/2025 - 21:13:35)			
Project File:	mtechproject.xise	**Parser Errors:**	No Errors
Module Name:	function1	**Implementation State:**	Synthesized
Target Device:	xc3s50-5vq100	**• Errors:**	No Errors
Product Version:	ISE 14.5	**• Warnings:**	4 Warnings (4 new)
Design Goal:	Balanced	**• Routing Results:**	
Design Strategy:	Xilinx Default (unlocked)	**• Timing Constraints:**	
Environment:	System Settings	**• Final Timing Score:**	

Device Utilization Summary (estimated values)			[-]
Logic Utilization	**Used**	**Available**	**Utilization**
Number of Slices	104	768	13%
Number of 4 input LUTs	193	1536	12%
Number of bonded IOBs	64	63	101%

Figure 10: Utilization Summary of Function

CONCLUSION

In this work, the delay performance of the Blowfish symmetric-key block cipher was optimized by incorporating a Modified Carry Select Adder (MCSA) into the encryption architecture. The MCSA, enhanced with reduce logic gate, significantly improved the computational speed by reducing the critical path delay associated with arithmetic operations in the cipher's key scheduling and data processing modules. The proposed design was implemented in Verilog and synthesized using FPGA tools, with performance metrics compared against conventional adder-based implementations. Results confirmed that the MCSA-based architecture achieved noticeable delay reduction, improved throughput, and maintained low area and power consumption. These improvements make the optimized Blowfish cipher more suitable for time-critical and resource-constrained applications such as real-time embedded systems, wireless sensor networks, and IoT devices. Overall, this study demonstrates that applying efficient arithmetic architectures can greatly enhance the performance of cryptographic algorithms in hardware, paving the way for faster and more secure encryption solutions.

REFERENCES

[1] R. K. Meyers and A. H. Desoky, "An Implementation of the Blowfish Cryptosystem," *IEEE International Symposium on Signal Processing and Information Technology (ISSPIT)*, pp. 346–351, 2008.

[2] M. C.-J. Lin and Y.-L. Lin, "A VLSI Implementation of Blowfish Encryption/Decryption Algorithm," *IEEE Design Automation Conference (ASP-DAC)*, pp. 1–2, 2000.

[3] B. Cody and J. Madigan, "High Speed SOC Design for Blowfish Cryptographic Algorithm," *IEEE International Conference on VLSI (IFIP)*, pp. 284–287, 2007.

[4] H. S. Zied, A. G. Abdellatif, and A. A. Elmahallawy, "An Optimized Implementation of the Blowfish Encryption Algorithm," *IEEE Access*, vol. 11, pp. 1–10, 2023.

[5] L. K. Kiran, J. E. N. Abhilash, and P. S. Kumar, "FPGA Implementation of Blowfish Cryptosystem Using VHDL," *International Journal of Engineering Research & Technology (IJERT)*, vol. 2, no. 1, pp. 1–5, 2013.

[6] A. S., "An Implementation of Blowfish Algorithm Using FPGA," *International Journal of Engineering Research & Technology (IJERT)*, vol. 2, no. 8, pp. 1–4, 2013.

[7] S. Chatterjee and S. Majumder, "FPGA Implementation of Pipelined Blowfish Algorithm," *IEEE International Symposium on Electronic System Design (ISED)*, pp. 208–213, 2014.

[8] D. Darlis, "An Implementation of Data Encryption for Internet of Things Using Blowfish Algorithm on FPGA," *International Journal of Advanced Computer Science and Applications (IJACSA)*, vol. 8, no. 5, pp. 1–6, 2017.

[9] L. Christina and V. J. Irudayaraj, "Optimized Blowfish Encryption Technique," *International Journal of Innovative Research in Computer and Communication Engineering*, vol. 2, no. 7, pp. 5009–5015, 2014.

[10] A. Mousa, "Data Encryption Performance Based on Blowfish," *47th International Symposium ELMAR*, pp. 131–134, 2005.

[11] M. Agrawal and P. Mishra, "A Modified Approach for Symmetric Key Cryptography Based on Blowfish Algorithm," *International Journal of Engineering and Advanced Technology (IJEAT)*, vol. 1, no. 6, pp. 79–83, 2012.

[12] A. E. Adeniyi, S. Misra, E. Daniel, and A. Bokolo Jr, "Computational Complexity of Modified Blowfish Cryptographic Algorithm on Video Data," *Algorithms*, vol. 15, no. 10, p. 373, 2022.

[13] R. R. Corpuz, B. D. Gerardo, and R. P. Medina, "Using a Modified Approach of Blowfish Algorithm for Data Security in Cloud Computing," *Proceedings of the 6th International Conference on Information Technology: IoT and Smart City*, pp. 157–162, 2018.

[14] R. O. Ogundokun, J. B. Awotunde, E. A. Adeniyi, and F. E. Ayo, "Crypto-Stego Based Model for Securing Medical Information on IoMT Platform," *Multimedia Tools and Applications*, vol. 80, pp. 31705–31727, 2021.

[15] M. Tayel, A. Gamal, and H. Shawky, "A Proposed Implementation Method of an Audio Steganography Technique," *2016 18th International Conference on Advanced Communication Technology (ICACT)*, pp. 180–184, 2016.