Quantum Grover Attack on MIBS

H.O. Cildiroglu[1,2] [(a)], H. Basmaci[1] and O. Yayla[2]

[1] *Department of Physics Engineering, Ankara University - Dögol St. 06100, Ankara, Turkiye*
[2] *Cryptography Department, Middle East Technical University - Universiteler Mh. 06800, Ankara, Turkiye*

PACS 03.67.Ac – Quantum algorithms, protocols, and simulations
PACS 03.67.Dd – Quantum cryptography and communication security
PACS 89.70.Eg – Computational complexity

Abstract –The advent of quantum computing necessitates a rigorous reassessment of classical cryptographic primitives, particularly lightweight block ciphers (LBCs) deployed in resource-constrained environments. This work presents a comprehensive quantum implementation and security analysis of the Feistel-based LBC MIBS against quantum cryptanalysis. Using the inherent reversibility of its structure, we develop a novel ancilla-free quantum circuit that optimizes qubit count and depth. For MIBS-64 and MIBS-80, our implementation achieves quantum costs of 23,371 and 24,363, requiring 128 and 144 qubits, respectively, with a depth of 4,768. We subsequently quantify the cipher's vulnerability to Grover's key-search algorithm under the NIST PQC security constraint MAXDEPTH. By constructing Grover oracles using inner parallelization with multiple plaintext-ciphertext pairs to suppress false positives, we demonstrate total quantum attack costs of approximately 2^{94} for MIBS-64 and 2^{111} for MIBS-80. These values fall below NIST's Level-1 security threshold (2^{170}), confirming the susceptibility of both MIBS variants to quantum key-recovery attacks despite their classical lightweight efficiency.

Introduction. – The emergence of quantum computing poses a significant threat to modern cryptographic systems. Classical cryptography relies on the computational intractability of mathematical problems such as integer factorization and discrete logarithms [1, 2]. However, quantum algorithms challenge this foundational assumption, exploiting the principles of superposition and entanglement. In particular, Grover's algorithm provides a quadratic speedup for brute-force key search attacks, weakening the security of symmetric encryption schemes [3].

Block ciphers (BC) remain strong candidates for post-quantum security due to their computational efficiency and demonstrated robustness against classical cryptanalysis [4]. Their security stems from iterative rounds of key-dependent substitutions and permutations, which ensure confusion and diffusion [5]. Lightweight block ciphers, such as the MIBS cipher [6], are specifically designed for resource-constrained environments, notably in the context of the Internet of Things (IoT). However, their compact architectures demand careful evaluation of security margins against quantum adversaries.

Grover's algorithm represents the primary quantum threat to symmetric cryptosystems, effectively reducing their key search space quadratically. The National Institute of Standards and Technology (NIST) has proposed a framework to evaluate this threat by defining quantum security levels, which are based on the maximum feasible circuit depth (MAXDEPTH) that a quantum adversary could achieve [7, 8]. A critical component of this framework involves the explicit quantum circuit implementation of the BC. This enables the construction of the Grover oracle and facilitates a precise estimation of the quantum resources required to mount a key-recovery attack.

In this work, we present a concrete quantum security analysis of the lightweight Feistel cipher MIBS. Our primary contribution is the first complete quantum circuit implementation of MIBS, optimized for qubit count and circuit depth by leveraging the inherent reversibility of its Feistel structure [9]. Here, we apply the DORCIS framework [10] for S-box optimization. The resulting implementation achieves a total quantum cost of 23,371 (MIBS-64) and 24,363 (MIBS-80) using 128 and 144 qubits, respectively. Leveraging this circuit, we construct the corresponding Grover oracle and carry out a detailed quantum resource estimation for key-recovery attacks. Our analysis, performed within the NIST MAXDEPTH framework, quanti-

[(a)] Corresponding author: cildiroglu@ankara.edu.tr

fies the total quantum attack costs at 1.21×2^{94} for MIBS-64 and 1.91×2^{110} for MIBS-80. These results demonstrate that the 64- and 80-bit key variants of MIBS fall short of NIST's Level-1 quantum security threshold, highlighting their vulnerability in quantum settings. Beyond this specific case, our work provides a generalizable framework for assessing Feistel-based ciphers and underscores the urgent need for quantum-resistant lightweight designs.

The remainder of this letter is structured as follows. We first review the classical structure and design principles of the MIBS cipher, and detail our optimized quantum circuit implementation. Then, we present the construction of the Grover oracle based on this implementation and provide a comprehensive quantum resource estimation for key-recovery attacks, analyzed within the NIST `MAXDEPTH` framework. Last, we compare our results with other lightweight block ciphers (LBCs) and discuss the broader implications of our findings for the post-quantum security of Feistel-based designs.

Structure of MIBS. – The MIBS is a symmetric LBC specifically developed for resource-constrained environments like the Internet of Things (IoT) and RFID systems [6]. Its design provides a balance between security and efficiency and features a 64-bit block size with a Feistel network structure over 32 rounds. The algorithm supports two key lengths, 64 bits (MIBS-64) and 80 bits (MIBS-80), making it adaptable to different security requirements.

Table 1: S-box of MIBS

x	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
S(x)	4	F	3	8	D	A	C	0	B	5	7	E	2	6	1	9

The encryption process of MIBS employs a classical Feistel network. The 64-bit plaintext input is divided into two 32-bit halves (See Fig. 1). For each round, the lower half (D_i) is passed through a round function (KSMP) and the result is XORed with the upper half (U_i). The round output is then given by $D_{i-1} = U_i$ and $D_i = U_{i-1} \oplus P(M(S(K^{i-1} \oplus D_{i-1})))$. The round function integrates four fundamental layers to ensure confusion and diffusion: a key addition layer (K-layer), a substitution layer (S-box) adapted from Mcrypton [11], a linear mixing layer (M-layer), and a permutation layer (P-layer).

The key schedule generates the 32-bit round subkeys from the master key. The process differs for the two key variants. For MIBS-64, the initial key state is the user key. For each round i, the state is updated by applying a right circular shift by 15 bits. The four most significant bits of the resulting state are then substituted using the cipher's S-box (Table 1). A 5-bit segment of the state [15:11] is then XORed with a round counter RC. The round subkey k^i is extracted as leftmost 32 bits of the state.

For the 80-bit variant (MIBS-80), the initial state is set to the user key. In each iteration, the state undergoes a right circular shift by 19 bits. The two four-bit segments at the top of the state (bits [79:76] and [75:72]) are processed through the S-box. A 5-bit segment (bits [18:14]) is XORed with the round counter. The 32-bit round subkey is extracted from bits [79:48] of the resulting state.

Table 2: P-layer of MIBS

x	0	1	2	3	4	5	6	7
P(x)	1	7	0	2	5	6	3	4

$$
\begin{aligned}
y'_1 &= y_2 \oplus y_3 \oplus y_4 \oplus y_5 \oplus y_6 \oplus y_7 \\
y'_2 &= y_1 \oplus y_3 \oplus y_4 \oplus y_6 \oplus y_7 \oplus y_8 \\
y'_3 &= y_1 \oplus y_2 \oplus y_4 \oplus y_5 \oplus y_7 \oplus y_8 \\
y'_4 &= y_1 \oplus y_2 \oplus y_3 \oplus y_5 \oplus y_6 \oplus y_8 \\
y'_5 &= y_1 \oplus y_2 \oplus y_4 \oplus y_5 \oplus y_6 \\
y'_6 &= y_1 \oplus y_2 \oplus y_3 \oplus y_6 \oplus y_7 \\
y'_7 &= y_2 \oplus y_3 \oplus y_4 \oplus y_7 \oplus y_8 \\
y'_8 &= y_1 \oplus y_3 \oplus y_4 \oplus y_5 \oplus y_8
\end{aligned}
\tag{1}
$$

The round function integrates operations to ensure confusion and diffusion. The input is first combined with the round subkey in the K-layer via a bitwise XOR. The resulting 32-bit value is split into eight 4-bit nibbles. These nibbles are processed in parallel by the S-box layer using a single 4-bit S-box (Table 1). The output is then passed to the M-layer given in eq. (1), a linear transformation based on XOR operations to achieve diffusion. Finally, the P-layer applies a fixed nibble-wise permutation to the output of the M-layer (Table 2).

Quantum Implementation of MIBS. – Quantitatively assessing the vulnerability of cryptosystems to Grover's algorithm requires demonstrating the cipher's quantum circuit feasibility [3]. This involves modeling the cipher in a quantum oracle to determine the practical resource cost of the Grover attack. For MIBS, calculating the total resource requirements of the attack relies on the accurate quantification of each round operation's quantum cost (NOT/CNOT/Toffoli gates) and circuit depth.

The limited qubit capacity of NISQ-era quantum hardware makes minimizing qubit count critical. To address this, we develop an architecture that requires no additional qubits by leveraging the inherent reversibility of the Feistel structure. The inverse round function, KSMP^{-1}, provides the necessary uncomputation step, restoring the data qubits to their initial state to complete each Grover iteration. This strategy keeps qubit counts low: 128 qubits (64 data + 64 key) for MIBS-64 and 144 qubits (64 data + 80 key) for MIBS-80.

General Implementation Strategy. In the quantum implementation of MIBS, the 64-bit plaintext is symmetrically divided into an upper (U) and a lower (D) halves (See fig 1). The process initiates by applying a CNOT operation to the 32 qubits of the D-box (D_0) with the first round key (K_1) via the key addition layer. The S-box is then applied concurrently across eight 4-qubit nibbles,

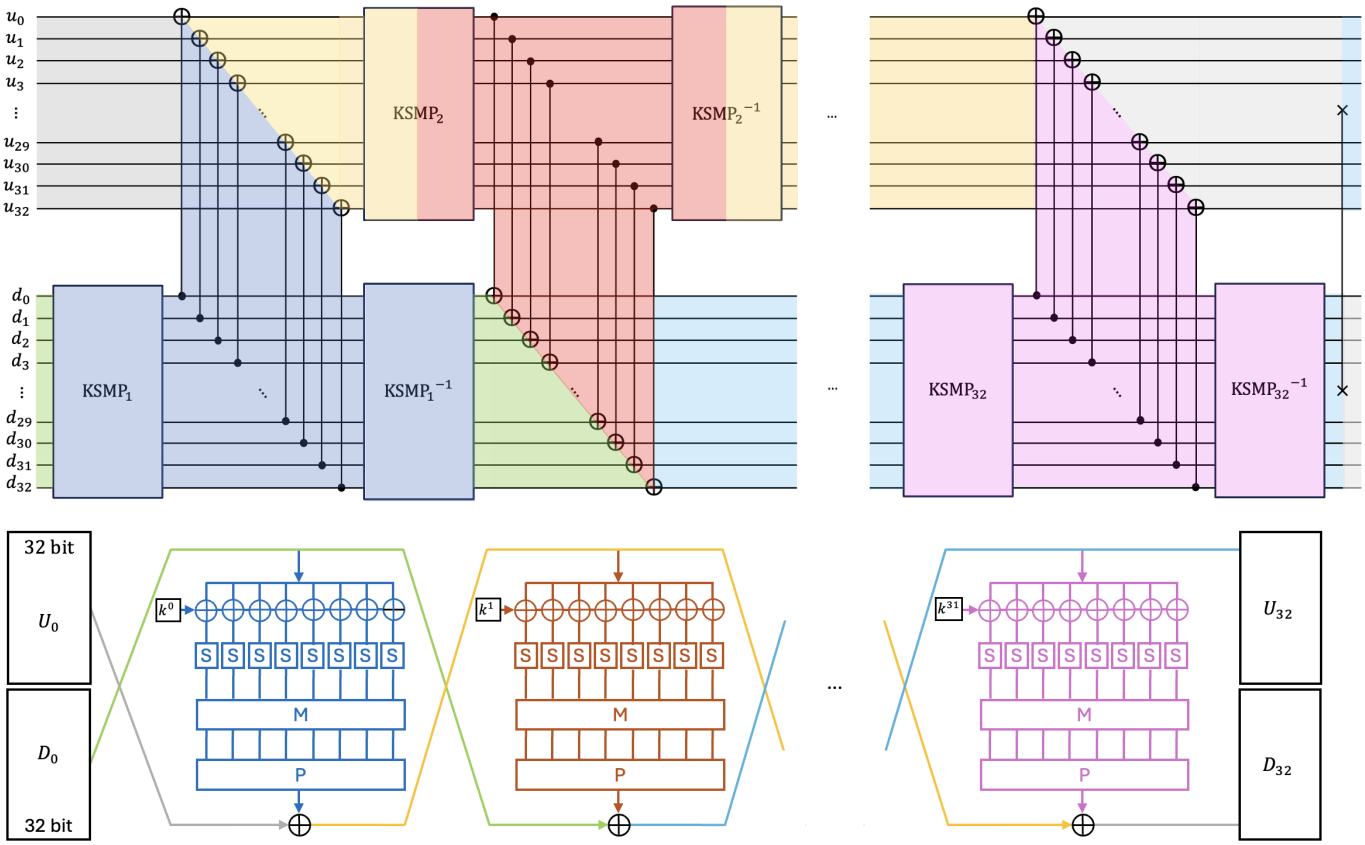


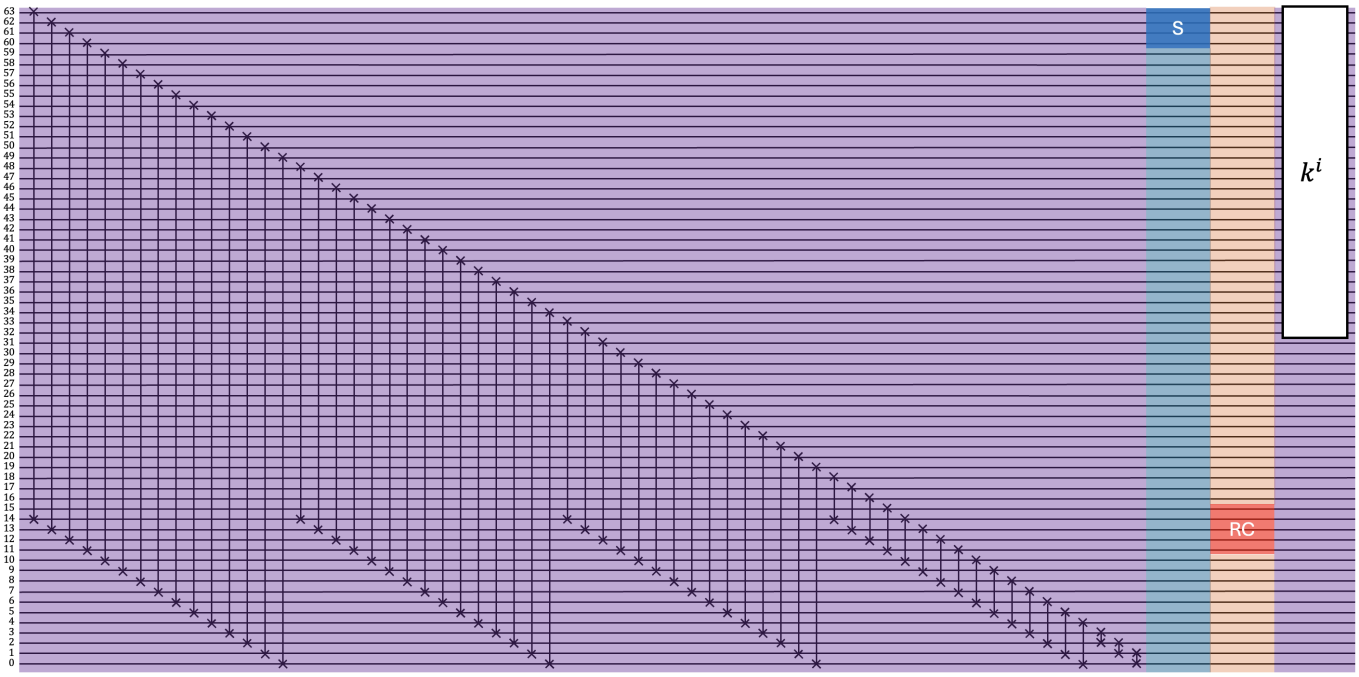
Fig. 1: Quantum implementation (top) and the classical structure (bottom) of MIBS

starting from the least significant qubits (LSQ). The outputs are then sequentially processed through the M- and P-layers. Last, the 32 qubits of the U-box (U_0) CNOTing on the processed D-box qubits, completing the first round and producing D_1. By leveraging the reversible properties of gates and the Feistel structure, the original qubit states are restored for subsequent rounds. This is achieved by applying the round function in reverse order (KSMP^{-1}), commencing from the lower branch and proceeding upward. This method effectively resets the channels for the next round, allowing U_i to function as D_{i-1} in each subsequent iteration.

Key Schedule Implementation. The MIBS algorithm features distinct quantum key scheduling mechanisms for its 64-bit and 80-bit variants across 32 rounds. For MIBS-64, the initial 64-bit key state undergoes a 15-qubit right circular shift implemented via SWAP gates. Subsequently, the four most significant qubits (MSQ) are processed by the S-box, while the 5-bit round counter (RC) is applied via CNOT gates to state bits [15:11]. The 32-bit round subkey k_i is extracted from the MSQ [63:32] of the resulting state (See fig. 2). For MIBS-80, the initial 80-bit key state undergoes a 19-qubit right circular shift. The two four-bit segments [79:76] and [75:72] are processed in parallel through the S-boxes. Then, the RC is applied through the bits [18:14]. Last, k_i is extracted from MSQ [79:48]. Here, the 5-bit RC itself is implemented as an ancilla-free counter, which uses NOT gates to increment its value for each round (See Fig. 2). The Feistel structure allows the same key to be used for the inverse operation $\mathrm{K}^{-1} = \mathrm{K}$.

S-Box Implementation. The challenge of S-box optimization is typically addressed using established tools like LIGHTER-R [12] and DORCIS [10]. LIGHTER-R fails to optimize the S-box when restricted to the standard NCT basis (NOT, CNOT, Toffoli), but succeeds when utilizing an expanded MCT set that includes higher-order gates like CCCNOT. Relying on its MCT-based solution, LIGHTER-R produces a circuit with a T-depth of 8, a full depth of 61, and a total gate cost of 45. In contrast, DORCIS yields a more efficient solution using the standard NCT basis. Specifically, DORCIS implements the S-box using 11 gates (1 NOT, 6 CNOT, 4 Toffoli) for a quantum cost of 31, a T-depth of 4, and a full depth of 35. A similar gain is seen for the S$^-$1-box, which requires 13 gates (1 NOT, 8 CNOT, 4 Toffoli) for a quantum cost of 33, a T-depth of 4, and a full depth of 34. This S-box optimization directly lowers the overall depth of the KSMP structure, which in turn enhances the efficiency of the overall quantum attack.

M-layer and P-layer Implementations. The M-layer performs a linear transformation on the eight 4-bit nibbles using XOR operations as given in (1). Its quantum



K_Q \ RC	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
$K_{15\,(18)}$																	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x
$K_{14(17)}$									x	x	x	x	x	x	x	x									x	x	x	x	x	x	x	x
$K_{13(16)}$					x	x	x	x					x	x	x	x					x	x	x	x					x	x	x	x
$K_{12(15)}$			x	x			x	x			x	x			x	x			x	x			x	x			x	x			x	x
$K_{11(14)}$		x		x		x		x		x		x		x		x		x		x		x		x		x		x		x	

Fig. 2: Key schedule (top) and round counter (bottom) implementations of MIBS

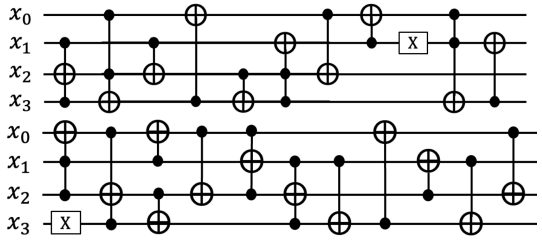


Fig. 3: S-box (top) and S^{-1}-box (bottom) realizations

implementation requires 60 CNOT gates, resulting in a circuit depth of 3 (See fig. 4). The inverse M^{-1} layer shares the same resource costs (See fig. 4-bottom left).

The P-layer performs a nibble-wise permutation according to a predefined mapping (Table 2). Its quantum implementation uses SWAP gates. While each logical SWAP operation is technically decomposable into three CNOT gates, we neglect this in our final cost estimation in order to provide a consistent comparison with the literature [15–20]. The inverse P^{-1} layer simply applies the reverse permutation, also implemented with SWAP gates (See fig. 4-bottom right).

Quantum Resource Estimation. The quantum resource requirements are evaluated using standardized metrics: NOT and CNOT cost 1 unit each, while a CCNOT costs 6 [13]. The total costs for variants across 32 rounds are detailed in Table 3. The aggregate analysis illustrates MIBS-64 requires 12,491 quantum gates with a total cost of 23,371 units, while MIBS-80 requires 12,843 gates with a cost of 24,363 units. The increased cost for MIBS-80 originates from its more complex key schedule.

Table 3: Quantum resource estimation for MIBS

LAYER	NOT	CNOT	CCNOT	TOTAL	COST
S	1	6	4	11	31
S^{-1}	1	8	4	13	33
K-64	1+RC	6	4	11+RC	31+RC
K-80	2+RC	12	8	22+RC	62+RC
KSMP-64	9+RC	146	36	191+RC	371+RC
KSMP-80	10+RC	152	40	202+RC	402+RC
$KSMP^{-1}$	8	156	32	196	356
MIBS-64	619	9696	2176	12491	23371
MIBS-80	651	9888	2304	12843	24363

The quantum depth of a single KSMP operation is 75, and 73 for its inverse. Including the CNOT operation between Feistel branches, each full round has a depth of 149. Over 32 rounds, the total quantum depth is 4,768 units for both variants, as the key size affects the gate count but not the path length.

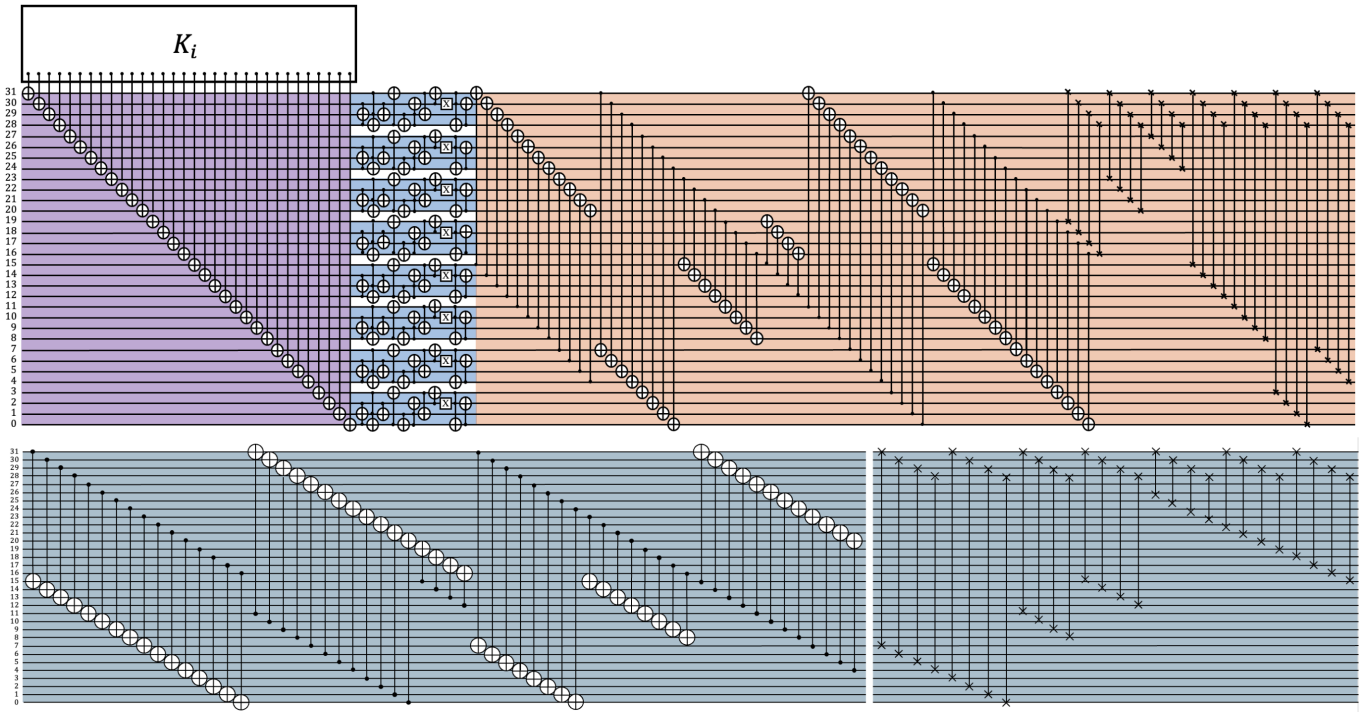


Fig. 4: (top) Single round KSMP- implementation of MIBS (bottom) M^{-1}-layer and P^{-1}-layer implementations

Table 4: Comparative Quantum Resource Requirements for LBC Implementations

CIPHER	QUBIT	NOT	CNOT	CCNOT	TOTAL	COST	REF
SIMON 32/64	96	448	2816	512	3776	6336	[15]
SPECK 32/64	96	42	4222	1290	5554	12004	[16]
RECTANGLE 64/80	144	567	4964	2000	7531	17531	[17]
PRINT 48/80	128	154	3840	2304	6298	17818	[18]
SLIM 32/80	112	1848	9452	3184	14484	30404	[19]
LBLOCK 64/80	144	877	16747	14280	31904	103304	[20]
MIBS 64/64	128	619	9696	2176	12491	23371	This Work
MIBS 64/80	144	651	9888	2304	12843	24363	This Work

Grover Analysis on MIBS. – We now present a quantitative assessment of the quantum attack resistance of the MIBS under Grover's search algorithm, which provides a quadratic speedup for searching an unstructured key space of size $N = 2^k$, reducing the brute-force attack complexity from $O(2^k)$ to $O(2^{k/2})$. First, applying $H^{\otimes k}$ to the initial state $|0\rangle^{\otimes k}$ prepares a uniform quantum superposition over all 2^k possible states: $|\psi\rangle = \frac{1}{\sqrt{2^k}} \sum_{x\in\{0,1\}^k} |x\rangle$. An ancillary qubit initialized in $|1\rangle$ is transformed via a Hadamard gate into $|-\rangle = \frac{1}{\sqrt{2}}(|0\rangle - |1\rangle)$, yielding the joint state $|\psi\rangle \otimes |-\rangle$. The algorithm consists of a quantum oracle U_f based on a Boolean function $f(x)$, where $f : \{0,1\}^k \to \{0,1\}$ that checks if a candidate key x encrypts a known plaintext m to its corresponding ciphertext c ($\mathrm{Enc}_x(m) = c$), and a diffusion operator D that amplifies the amplitude of the correct key state: $f(x) = 1$ if $\mathrm{Enc}_x(m) = c$; $f(x) = 0$ if $\mathrm{Enc}_x(m) \neq c$. The quantum oracle concurrently encrypts the known plaintext with all candidate keys ($x \in \{0,1\}^k$) in superposition. This circuit simultaneously processes all keys through the encryption mechanism, generating the corresponding intermediate ciphertext states. The resulting temporary outputs are then compared with the known target ciphertext. When a match occurs, the phase of the associated key state is inverted: $\sum_x a_x|x\rangle|y\rangle \to \sum_x a_x|x\rangle|y \oplus f(x)\rangle$. Then, the quantum circuit is reversed to uncompute the ciphertexts, leaving only the phase-altered key states. The oracle and diffusion operators are then applied as follows:

$$U_f(|\psi\rangle \otimes |-\rangle) = 2^{-\frac{n}{2}} \sum_{x=0}^{2^n-1} (-1)^{f(x)}|x\rangle \otimes |-\rangle \qquad (2)$$

$$D = 2|\psi\rangle\langle\psi| - I$$

Grover's algorithm iteratively applies the Grover operator $G = DU_f$, rotating the state vector by an angle θ, where $\sin^2\theta = M/N$ and M denotes the number of solutions (See fig. 5). When $M \ll N$, we approximate

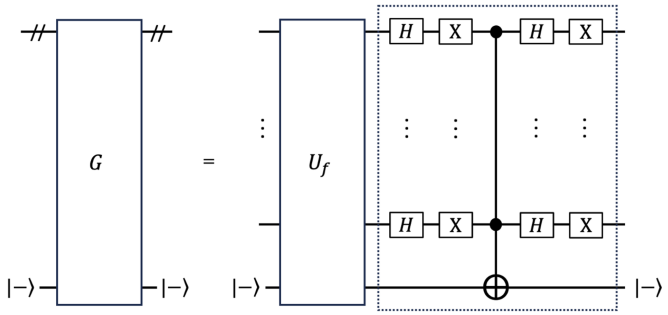


Fig. 5: Grover operator $G = U_f D$ and internal structure of D

$\theta \approx \sqrt{M/N}$. After j iterations, the success probability becomes $p(j) = \sin^2((2j+1)\theta)$, which is maximized for $j \approx \frac{\pi}{4}\sqrt{\frac{N}{M}}$, yielding $p(j) \geq 1 - \frac{M}{N}$ [14]. The efficiency of the algorithm is determined by the implementation cost of U_f, which dominates the total quantum resource usage.

For a successful key recovery, multiple plaintext-ciphertext pairs (r) are required to minimize false positives. For MIBS-64 and MIBS-80, we implement the oracle with $r = 2$ and $r = 3$ pairs (See fig. 6). The oracle encrypts r plaintexts in parallel using the candidate key, compares the results to the target ciphertexts using multi-controlled NOT gates, and flips the phase of the key register if all comparisons match. The circuit is then uncomputed. The total qubit count for the Grover oracle is $r \cdot Q + 1$, where Q is the qubit count for a single MIBS instance (128 for MIBS-64, 144 for MIBS-80). This results in 257 and 433 qubits for MIBS-64/64 ($r = 2$) and MIBS-64/80 ($r = 3$), respectively.

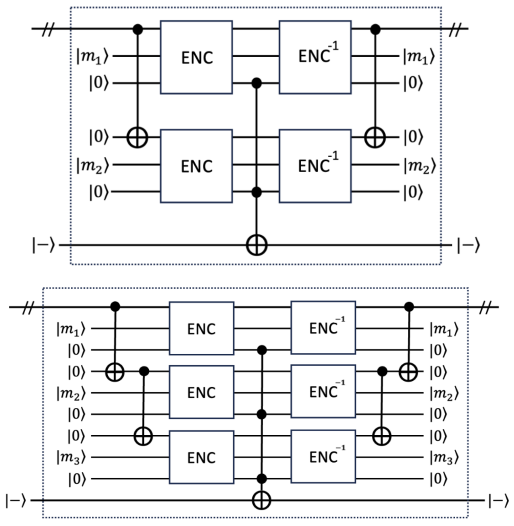


Fig. 6: Grover oracles for (top) $r = 2$ and (bottom) $r = 3$

The quantum resource estimates for the Grover oracles are summarized in Table 6. The gate counts and depths are derived from the underlying MIBS quantum implementation from Table 3. The overall cost of the Grover attack is determined by the number of iterations ($\approx \frac{\pi}{4}\sqrt{2^k}$ for $k = 64$ and $k = 80$) and the resources per oracle call.

NIST defines security levels for post-quantum cryptography based on the total cost of a Grover attack. The total computational cost is quantified by the product of the circuit depth and the total number of gates, following the `MAXDEPTH` benchmarking framework [7]. Accordingly, security Level 1 requires a cost of at least 2^{170} operations. The total costs for both MIBS-64 ($\sim 2^{94}$) and MIBS-80 ($\sim 2^{111}$) fall significantly below this threshold, indicating that they do not provide sufficient security against quantum adversaries. Table 5 presents the estimated total cost for attacking MIBS variants under this framework.

Table 5: `MAXDEPTH` analysis of Grover's algorithm

CIPHER	r	2^{40}	2^{64}	2^{96}	approximation
MIBS 64/64	2	54	30	-2	2^{94}/`MAXDEPTH`
MIBS 64/80	2	70	46	14	2^{110}/`MAXDEPTH`
MIBS 64/80	3	71	47	15	2^{111}/`MAXDEPTH`

This analysis demonstrates that despite their efficiency in classical environments, both variants of the MIBS cipher are vulnerable to key-recovery attacks using Grover's algorithm on a sufficiently powerful quantum computer. Their security levels are inadequate against the quantum threat model defined by NIST. Future lightweight cipher designs must incorporate longer key sizes or different structures to achieve quantum resistance.

Conclusions. – This study presents a comprehensive quantum security analysis of the MIBS. We provide a novel complete ancilla-free quantum circuit implementation for both its 64-bit and 80-bit key variants, exploiting the cipher's Feistel structure. Our design is optimized for qubit count, requiring only 128 and 144 qubits for the 64-bit and 80-bit versions, respectively. For the S-box implementations, we utilized the DORCIS framework, which yielded a lower quantum cost and T-depth NCT-based circuit compared to other tools.

This quantum implementation enables the construction of the Grover oracle required for key-recovery analysis. The security evaluation, performed within the NIST `MAXDEPTH` framework, demonstrates that both MIBS variants are vulnerable to quantum key-recovery attacks. The total quantum cost is approximately 2^{94} for MIBS-64 and 2^{111} for MIBS-80. These costs fall significantly below the NIST Level-1 security threshold of 2^{170}, indicating that MIBS does not offer sufficient protection against a quantum adversary.

The vulnerability of MIBS primarily stems from the cipher's short key length and mostly linear key schedule, both of which are efficiently exploited by Grover's algorithm. While MIBS offers classical efficiency for constrained environments, this efficiency constitutes a critical weakness in the quantum threat model.

Our findings highlight the necessity for a paradigm shift

Table 6: Comparative quantum resource estimates for Grover key search on LBCs

CIPHER	r	GATE COST	#GATES	FULL DEPTH	TOTAL COST
SIMON 32/64	3	38016*	1.91×2^{45}	1.29×2^{46}	1.23×2^{92}
SIMON 48/72	2	37152*	1.59×2^{49}	1.36×2^{50}	1.08×2^{100}
SPECK 32/64	3	72024*	1.78×2^{46}	1.43×2^{45}	1.27×2^{92}
SPECK 48/72	2	73488*	1.37×2^{51}	1.11×2^{50}	1.52×2^{101}
RECTANGLE 64/80	2	62284	1.79×2^{58}	1.40×2^{49}	1.25×2^{108}
LBLOCK 32/80	2	111296*	1.04×2^{56}	1.34×2^{52}	1.39×2^{108}
PRINT 48/80	2	62216	1.97×2^{58}	1.06×2^{50}	1.04×2^{108}
SLIM 32/80	2	121766	1.77×2^{56}	1.56×2^{53}	1.37×2^{110}
SLIM 32/80	3	182744	1.33×2^{57}	1.56×2^{53}	1.04×2^{111}
MIBS 64/64	2	93612	1.33×2^{48}	1.83×2^{45}	1.21×2^{94}
MIBS 64/80	2	97452	1.39×2^{56}	1.83×2^{53}	1.27×2^{110}
MIBS 64/80	3	146498	1.04×2^{57}	1.83×2^{53}	1.91×2^{110}

*Estimated from cited gate counts for table completeness.

in lightweight cipher design. Future quantum-resistant designs must incorporate longer keys, more complex nonlinear key schedules, and a higher key-to-block size ratio. The methodology and benchmarks presented here provide a framework for assessing the post-quantum security of symmetric cryptographic primitives. Our S-box analysis, in particular, highlights how strongly the total attack cost is dependent on the efficiency of these core components. This highlights the potential of emerging advanced methods, such as the memory-optimized approaches in [21,22].

REFERENCES

[1] FIPS, *NBS*, **197** (2001) USDC, Washington.

[2] FIPS, *NBS*, **46-3** (1999) USDC, Washington.

[3] Grover L. K., *Proc. of the 28th ACM Symposium on Theory of Computing-STOC* (ACM) 1996.

[4] Bernstein D. J. and Lange T., *Cryptology ePrint Archive*, **2017** (Paper 2017/314) .

[5] Shannon C. E., *Bell Syst. Tech. J.*, **28** (1949) 656.

[6] Izadi M., Sadeghiyan B., Sadeghian S. S. and Khanooki H. A., *Proc. of Int. Conf. on Cryptology and Network Security* (Springer) 2009, p. 334.

[7] NIST, *Project for Post-Quantum Cryptography Standardization. Call for Proposals (NIST IR 8105)* (U.S. Department of Commerce) 2016.

[8] NIST, *Recommendation for Key Management: Part 1 - General (NIST Special Publication 800-57 Part 1 Revision 5)* (U.S. Department of Commerce) 2020.

[9] Feistel H., *Scientific American*, **228** (1973) 15.

[10] Chun M., Baksi A. and Chattopadhyay A., *Cryptology ePrint Archive*, **2023** (Paper 2023/XXX) .

[11] Lim C. H. and Korkishko T., *Proc. 6th Int. Workshop*, edited by J. Song, T. Kwon and M. Yung, Vol. **3786** (Springer, Jeju Island, South Korea) 2005, p. 243.

[12] Dasu V. A., Baksi A., Sarkar S. and Chattopadhyay A., *Proc. 32nd IEEE Int. System-on-Chip Conf. (SOCC)* (IEEE) 2019, p. 260.

[13] Shende V. and Markov I., *Quantum Inf. Comput.*, **9** (2009) 461.

[14] Jaques S., Naehrig M., Roetteler M. and Virdia F., *Advances in Cryptology–EUROCRYPT 2020* (Springer) 2020, p. 280.

[15] Anand R., Maitra A. and Mukhopadhyay S., *Quantum Inf. Process.*, **19** (2020) 1.

[16] Jang K., Choi S. *et al.*, *Cryptology*, **2020** (Preprint) .

[17] Saravanan P., Jenitha J., Aasish S. R. and Sanjana S., *Proc. 25th Int. Symp. on VLSI Design and Test (VDAT)* (IEEE) 2021, p. 1.

[18] Paramasivam S., Jenitha J., Sanjana S. and Haghparast M., *IEEE Access*, **11** (2023) 66767.

[19] Cildiroglu, H.O. and Yayla, O., *Physica Scripta*, **100** (10) 2025105023.

[20] Jing X. Y., Li Y. J., Zhao G. Y., Xie H. and Wang Q., *Quantum Inf. Process.*, **22** (2023) 347.

[21] Tilki, O., Cildiroglu, H.O., Yayla, O. and Saran, A.N., *Proc. 18th Int. Conf. on Information Security and Cryptography (ISC2025)*, Quantum Implementation of S-Boxes: A Memory Optimized Approach (IEEE Axcess) 2025.

[22] Chung, D. and Lee, S., *Electronics Letters,* Quantum Implementation of S-Boxes Based on Polynomial Evaluation, **61** (2025) e70337.