CHAPTER 8

RESULTS AND DISCUSSION

8.1 Introduction

This chapter highlights the results and discussion of the cryptanalysis conducted on the new LAO-3D algorithm and other existing block ciphers. Similar to the cryptanalysis conducted in Chapter 7, this chapter presents three avalanche effects experiments including correlation coefficient, bit error, and key sensitivity tests. In addition, randomness tests, differential cryptanalysis, and linear cryptanalysis were also executed.

The presentation of the results and its discussion are divided into two sections. In the first section, the results of LAO-3D algorithm are compared to the original RECTANGLE block cipher. Meanwhile, the second section presented the results of LAO-3D lightweight block cipher against the other existing algorithm.

In addition to the cryptanalysis results, this chapter presents the performance tests of lightweight block ciphers. For comparison, the results of LAO-3D lightweight block cipher were compared with RECTANGLE to show the superiority of the new algorithm against the original block cipher. Apart from that, comparisons of LAO-3D against other existing algorithms are also presented.

8.2 LAO-3D vs RECTANGLE

The cryptanalysis experiments conducted on RECTANGLE block cipher in this section implemented the same testing methodology, equipment, and settings of LAO-3D lightweight block cipher as presented in Chapter 7. Besides that, the same sample data sets from APPENDIX J are used to conduct the tests. A fair comparison of both LAO-3D algorithm and RECTANGLE block cipher is important to show the improvements made to the proposed algorithm which is developed based on the original block cipher.

8.2.1 Correlation Coefficient Test

The correlation coefficient results are summarized in Table 8.1. Results produced by all of the five keys indicate that LAO-3D has better correlation coefficient results than the original RECTANGLE block cipher. The majority of the correlation coefficients, r_{pc} of LAO-3D are located in the $0 < r_{pc} \leq 0.3$ and $-0.3 \leq r_{pc} < 0$ ranges which indicate a weak linear relationship between the input and output. On average, LAO-3D recorded 98.20% correlation value between 0 to 0.3 (and -0.3 to 0), and 1.80% between 0.3 to 0.7 (and -0.7 to -0.3). On the contrary, RECTANGLE obtained 57.20% correlation value between 0 to 0.3 (and -0.3 to 0), 41.66% between 0.3 to 0.7 (and -0.7 to -0.3), and 1.14% between 0.7 to 1.0 (and -1.0 to -0.7). Overall, LAO-3D block cipher has a better non-linearity than RECTANGLE algorithm.

Table 8.1: Correlation Coefficient Results

Input	Algorithm	r_{pc} = -1, r_{pc} = 0, and r_{pc} = 1	$0 < r_{pc} \leq 0.3$ and $-0.3 \leq r_{pc} < 0$	$0.3 < r_{pc} < 0.7$ and $-0.7 < r_{pc} < -0.3$	$0.7 \leq r_{pc} < 1$ and $-1 < r_{pc} \leq -0.7$
Key 1	LAO-3D	0	979	21	0
	RECTANGLE	0	574	416	10
Key 2	LAO-3D	0	977	23	0
	RECTANGLE	0	571	419	10
Key 3	LAO-3D	0	990	10	0
	RECTANGLE	0	568	422	10
Key 4	LAO-3D	0	982	18	0
	RECTANGLE	0	579	406	15
Key 5	LAO-3D	0	982	18	0
	RECTANGLE	0	568	420	12

8.2.2 Bit Error Rate Test

The comparison of bit error rate test results shown in Table 8.2 indicates that LAO-3D has obtained better results compared with the RECTANGLE version. Overall, the *BER* of LAO-3D is close to 0.5. The results generated by all five plaintexts indicate that LAO-3D block cipher achieved a 50.00% bit error rate which is the optimum test result and performed better than RECTANGLE which achieved 49.95% *BER* result.

Table 8.2: Bit Error Rate Results

Input	Algorithm	Average Different Bits	Average Bit Error Rate
Plaintext 1	LAO-3D	31.703125	0.495361
	RECTANGLE	30.656250	0.479004
Plaintext 2	LAO-3D	32.203125	0.503174
	RECTANGLE	32.937500	0.514648
Plaintext 3	LAO-3D	32.046875	0.500732
	RECTANGLE	31.828125	0.497314
Plaintext 4	LAO-3D	32.046875	0.500732
	RECTANGLE	32.875000	0.513672
Plaintext 5	LAO-3D	32.156250	0.502441
	RECTANGLE	31.546875	0.492920

8.2.3 Key Sensitivity Test

The comparison of key sensitivity test results shown in Table 8.3 indicates that LAO-3D obtained better results than the original RECTANGLE algorithm. On average, LAO-3D block cipher recorded 50.00% bit error rate compared to 50.02% obtained by RECTANGLE. The results indicate that LAO-3D has a non-linear relationship between the key and ciphertext which also represents a high sensitivity of the key to the ciphertext.

Table 8.3: Key Sensitivity Results

Input	Algorithm	Average Different Bits	Average Bit Error Rate
Key 1	LAO-3D	32.000122	0.500002
	RECTANGLE	32.437500	0.506836
Key 2	LAO-3D	32.028317	0.500442
	RECTANGLE	32.242188	0.503784
Key 3	LAO-3D	32.007694	0.500120
	RECTANGLE	31.570313	0.493286
Key 4	LAO-3D	31.948364	0.499193
	RECTANGLE	31.609375	0.493896
Key 5	LAO-3D	32.013795	0.500216
	RECTANGLE	32.218750	0.503418

8.2.4 Randomness Tests

Randomness tests results using two significance levels are summarized in Table 8.4. The same data set, testing method, and settings were used to evaluate the randomness of RECTANGLE block cipher. From the experiments, LAO-3D algorithm performed better than the original RECTANGLE. LAO-3D managed to pass all of the randomness tests, while RECTANGLE failed a number of tests.

Table 8.4: Randomness Tests Results

Algorithm	Significance Level	Results	Data Category	Statistical Test
LAO-3D	0.1%	Pass	9	15
		Fail	0	0
	1%	Pass	9	15
		Fail	0	0
RECTANGLE	0.1%	Pass	0	8
		Fail	9	7
	1%	Pass	6	10
		Fail	3	5

8.2.5 Differential Cryptanalysis

Differential cryptanalysis results of LAO-3D algorithm are compared with RECTANGLE block cipher as shown in Table 8.5. In addition, the number of active S-boxes of the differential cryptanalysis are also presented to show the advantage of the new algorithm. The results indicate that LAO-3D recorded the lowest allowable probabilities of differential trails (2^{-44}) with 17 active S-boxes at the fifth round compared to RECTANGLE which achieved the lowest allowable probabilities results in the 14th round.

Table 8.5: Active S-Boxes and Probabilities of Differential Trails

Rounds	LAO-3D		RECTANGLE	
	Active S-Boxes	Probability	Active S-Boxes	Probability
1	1	2^{-2}	1	2^{-2}
2	3	2^{-8}	2	2^{-4}
3	6	2^{-15}	3	2^{-7}
4	11	2^{-27}	4	2^{-10}
5	17	2^{-44}	6	2^{-14}
6	*25	$*2^{-64}$	8	2^{-18}
7			11	2^{-25}
8			13	2^{-31}
9			14	2^{-36}
10			16	2^{-41}
11			18	2^{-46}
12			20	2^{-51}
13			22	2^{-56}
14			24	2^{-61}
15			26	$*2^{-66}$

* indicates no effective trail from the encryption round onwards

8.2.6 Linear Cryptanalysis

Results of the linear cryptanalysis of LAO-3D algorithm are compared with RECTANGLE lightweight block cipher as shown in Table 8.6. The number of active S-boxes of the linear cryptanalysis are also displayed to present the advantage of the proposed algorithm. The results show that LAO-3D achieved the lowest allowable correlation potentials of linear trails (2^{-32}) at the sixth round with 11 active S-boxes which is better than RECTANGLE block cipher.

Table 8.6: Active S-Boxes and Correlation Potentials of Linear Trails

Rounds	LAO-3D		RECTANGLE	
	Active S-Boxes	Correlation Potential	Active S-Boxes	Correlation Potential
1	1	2^{-2}	1	2^{-1}
2	3	2^{-8}	2	2^{-2}
3	5	2^{-14}	3	2^{-4}
4	7	2^{-20}	4	2^{-6}
5	9	2^{-26}	6	2^{-8}
6	11	2^{-32}	8	2^{-10}
7	*13	$*2^{-38}$	10	2^{-13}
8			12	2^{-16}
9			14	2^{-19}
10			16	2^{-22}
11			18	2^{-25}
12			20	2^{-28}
13			22	2^{-31}
14			*24	$*2^{-34}$

* indicates no effective trail from the encryption round onwards

8.3 LAO-3D vs Other Block Ciphers

The cryptanalysis experimental results obtained from LAO-3D lightweight block cipher are compared with the other existing algorithms. Cryptanalysis results of the other block ciphers are obtained from published research articles. These comparisons are important to observe the security strength of the proposed LAO-3D lightweight block cipher against its established competitors.

8.3.1 Correlation Coefficient Test

A comparison of the correlation coefficient between LAO-3D algorithm against other existing block ciphers is presented in Table 8.7 (Ariffin, 2012; Zakaria et al., 2020). Majority of the correlation coefficients, r_{pc} of the compared algorithms are located in the $0 < r_{pc} \leq 0.3$ and $-0.3 \leq r_{pc} < 0$ ranges which indicate a weak linear relationship between the input and output. Overall, LAO-3D recorded the highest correlation value in the mentioned ranges, followed by 3D-AES, AES, and 3D RECTANGLE. In conclusion, LAO-3D block cipher has a better non-linearity than the other block ciphers.

Table 8.7: Correlation Coefficient Results

Algorithm	r_{pc} = -1, r_{pc} = 0, and r_{pc} = 1	$0 < r_{pc} \leq 0.3$ and $-0.3 \leq r_{pc} < 0$	$0.3 < r_{pc} < 0.7$ and $-0.7 < r_{pc} < -0.3$	$0.7 \leq r_{pc} < 1$ and $-1 < r_{pc} \leq -0.7$
LAO-3D	0%	98.20%	1.80%	0%
AES	0%	64.06%	19.53%	16.41%
3D-AES	0%	64.84%	35.16%	0%
3D RECTANGLE	0%	58.78%	40.38%	0%

8.3.2 Bit Error Rate Test

On average, LAO-3D produced 32 different bits which is equivalent to 50.00% bit error rate which verifies that the ciphertext is entirely modified with a single alteration in the plaintext bit. For better observation, a comparison of the avalanche effect on the modification of the plaintext against other existing block ciphers is presented in Table 8.8 (Biswas et al., 2020). The result achieved by LAO-3D surpasses the earlier works, this indicates that the new lightweight block cipher has a non-linear relationship of its corresponding plaintext to the ciphertext.

Table 8.8: Bit Error Rate Results

Algorithm	Average Avalanche Effect
LAO-3D	50.00%
LED	52.83%
LRBC	58.00%
PRINCE	51.18%
PRINT	49.08%
QTL	52.56%
SIMECK	53.00%
TEA	49.12%

8.3.3 Key Sensitivity Test

Observations made on the key sensitivity performed on LAO-3D lightweight block cipher against other existing algorithms are presented in Table 8.9 (Biswas et al., 2020). The 50.00% key sensitivity result obtained by LAO-3D lightweight block cipher denotes that the entire key bits have an impact on every ciphertext bit of the algorithm. Based on the findings, LAO-3D block cipher performed better than the existing works on the avalanche effect of the key modifications.

Table 8.9: Key Sensitivity Results

Algorithm	Average Avalanche Effect
LAO-3D	50.00%
LED	50.37%
LRBC	55.75%
PRINCE	49.06%
PRINT	46.42%
QTL	50.31%
SIMECK	51.25%
TEA	47.12%

8.3.4 Randomness Tests

Comparison of randomness tests between LAO-3D lightweight block cipher and other algorithms is presented in Table 8.10 (Chew et al., 2015; Shah & Ismail, 2020; Zakaria et al., 2020). The results show that only LAO-3D passed all of the randomness tests based on the 9 data categories and 15 statistical tests which indicates that the lightweight block cipher able to behave as a pseudorandom bit generator.

Table 8.10: Randomness Tests Results

Algorithm	Results	Data Category	Statistical Test
LAO-3D	Pass	100%	100%
	Fail	0	0
PRESENT	Pass	44.44%	86.67%
	Fail	55.56%	13.33%
SPECK	Pass	77.78%	86.67%
	Fail	22.22%	13.33%
3D RECTANGLE	Pass	88.89%	66.67%
	Fail	11.11%	33.33%

8.3.5 Differential Cryptanalysis

Comparison of differential cryptanalysis results of LAO-3D algorithm and other existing algorithms are shown in Table 8.11 (Zhou et al., 2019). Apart from that, the number of active S-boxes of the differential cryptanalysis are presented in Table 8.12 (Banik et al., 2017). The results show that LAO-3D achieved the lowest allowable probabilities of differential trails in the fifth round. In comparison with GIFT and PRESENT, the algorithms require 12 and 14 rounds respectively to achieve the lowest allowable probabilities.

Table 8.11: Probabilities of Differential Trails

Rounds	Probability		
	LAO-3D	GIFT	PRESENT
1	2^{-2}	2^{-6}	2^{-2}
2	2^{-8}	2^{-10}	2^{-4}
3	2^{-15}	2^{-16}	2^{-8}
4	2^{-27}	2^{-20}	2^{-12}
5	2^{-44}	2^{-26}	2^{-20}
6	$*2^{-64}$	2^{-30}	2^{-24}
7		2^{-36}	2^{-28}
8		2^{-40}	2^{-32}
9		2^{-46}	2^{-36}
10		2^{-50}	2^{-41}
11		2^{-56}	2^{-46}
12		2^{-60}	2^{-52}
13		$*2^{-64}$	2^{-56}
14			2^{-62}
15			$* 2^{-66}$

* indicates no effective trail from the encryption round onwards

Table 8.12: Active S-Boxes of Differential Cryptanalysis

Rounds	Active S-Boxes		
	LAO-3D	GIFT	PRESENT
1	1	1	1
2	3	2	2
3	6	3	4
4	11	5	6
5	17	7	10
6	*25	10	12
7		13	14
8		16	16
9		18	18
10		21	20
11		24	22
12		27	25
13		*30	27
14			30
15			*32

* indicates no effective trail from the encryption round onwards

8.3.6 Linear Cryptanalysis

The comparison results of the linear cryptanalysis of LAO-3D against the other existing algorithms is shown in Table 8.13 (Zhu et al., 2019) to present the advantage of the new algorithm. In addition, the number of active S-boxes of the linear cryptanalysis are presented in Table 8.14 (Banik et al., 2017). The results indicate that LAO-3D achieved the lowest allowable correlation potentials of linear trails in the sixth round. From the observation, LAO-3D block cipher performs better than GIFT and PRESENT algorithms.

Table 8.13: Correlation Potentials of Linear Trails

Rounds	Correlation Potential		
	LAO-3D	GIFT	PRESENT
1	2^{-2}	2^{-1}	2^{-1}
2	2^{-8}	2^{-2}	2^{-2}
3	2^{-14}	2^{-3}	2^{-4}
4	2^{-20}	2^{-5}	2^{-6}
5	2^{-26}	2^{-7}	2^{-8}
6	2^{-32}	2^{-10}	2^{-10}
7	$*2^{-38}$	2^{-13}	2^{-12}
8	-	2^{-16}	2^{-14}
9	-	2^{-20}	2^{-16}
10	-	2^{-25}	2^{-18}
11	-	2^{-29}	2^{-20}
12	-	2^{-31}	2^{-22}
13	-	$*2^{-34}$	2^{-24}
14	-	-	2^{-26}
15	-	-	2^{-28}
16	-	-	2^{-30}
17	-	-	2^{-32}
18	-	-	$*2^{-34}$

* indicates no effective trail from the encryption round onwards

Table 8.14: Active S-Boxes of Linear Cryptanalysis

Rounds	Active S-Boxes		
	LAO-3D	GIFT	PRESENT
1	1	1	1
2	3	2	2
3	5	3	3
4	7	5	4
5	9	7	5
6	11	9	6
7	*13	12	7
8		15	8
9		18	9
10		22	10
11		25	11
12		27	12
13		*30	13
14			14
15			15
16			16
17			17
18			*18

* indicates no effective trail from the encryption round onwards

8.4 Software Performance Tests

This section highlights the execution of software performance tests on the new lightweight block cipher to answer *Research Question 8*, as well as to fulfil part of *Research Objective 4*, thus producing part of *Research Contribution 3*. LAO-3D algorithm was implemented using Microsoft Visual Studio 2008 on an Intel(R) Core(TM) i7 2.70 GHz CPU with 8 GB RAM on Windows 10. Speed tests were conducted on the execution process of LAO-3D and other existing algorithms. The speed tests were carried out on the full rounds of each encryption algorithm to observe the time required to process a ciphertext block that consists of 64-bit data. In addition, the throughput tests evaluated the impact of cipher design such as the key size, block size, number of rounds, and encryption components on the algorithm throughput.

The performance of a cryptographic algorithm is determined by evaluating the running speed that can be measured by the average encryption time, encryption throughput, and the required number of cycles to encrypt one byte or block plaintext which permits researchers to compare the running speed of different algorithms working on different platforms. The encryption throughput and the number of cycles are defined in equations (15) and (16).

$$Encryption\ Throughput = \frac{\text{Message Size}}{\text{Encryption Speed}} \tag{15}$$

$$Cycles\ per\ Byte = \frac{\text{CPU Clock Speed}}{\text{Encryption Throughput}} \tag{16}$$

8.4.1 LAO-3D vs RECTANGLE

Results presented in Table 8.15 shows that LAO-3D achieved better performance against the original RECTANGLE lightweight block cipher in terms of execution speed and throughput. LAO-3D algorithm recorded 10.85% faster execution speed and produced 12.18% more throughput than the RECTANGLE. Factors that contributed to the result are the number of encryption rounds and the algorithm components. LAO-3D has low encryption rounds with optimum component operations. Substitution (*SubColumn*) and permutation (*Double3DRotation*) functions of the new block cipher ensure security without the need of implementing a high number of encryption rounds.

Table 8.15: Performance Tests Results

	LAO-3D	RECTANGLE
Block Size (bit)	64	64
Key Size (bit)	128	128
Rounds	20	25
Encryption Algorithm Component	1. Add Round Key 2. Sub Column 3. Double 3D Rotation	1. Add Round Key 2. Sub Column 3. ShiftRow
Encryption Speed (millisecond)	1.4569	1.6342
Encryption Throughput (byte per second)	5,491	4,895
Encryption Throughput (block per second)	686	611
Cycles per Byte	491,691	551,583
Cycles per Block	3,933,531	4,412,666

8.4.2 LAO-3D vs Other Block Ciphers

Table 8.16 shows that LAO-3D performed better than the other existing lightweight block ciphers in terms of execution speed and throughput (Singh et al., 2019). Existing high-performance algorithms such as KATAN (De Cannière et al., 2009), KLEIN (Gong et al., 2011), PRESENT (Bogdanov et al., 2007), and SPECK (Beaulieu et al., 2013) are included in this comparison as the benchmark for the proposed lightweight block cipher. LAO-3D lightweight block cipher recorded 29.66% faster execution speed and produced 42.18% more throughput than the closest competitor which is the KLEIN algorithm. Although LAO-3D implements the same block size as the other algorithms, the optimized encryption algorithm components along with its round number have contributed to the performance of the block cipher. Therefore, these observations justified that the performance of LAO-3D is competitive and suitable to be applied in mobile applications.

Table 8.16: Performance Tests Results

	LAO-3D	KATAN	KLEIN	PRESENT	SPECK
Block Size (bit)	64	64	64	64	64
Key Size (bit)	128	80	64	128	128
Rounds	20	254	12	31	27
Encryption Algorithm Component	1. Add Round Key 2. Sub Column 3. Double 3D Rotation	1. LFSR	1. Sub Nibble 2. Rotate Nibble 3. Mix Nibble	1. Add Round Key 2. Substitution 3. Permutation	1. XOR 2. Modulo Addition 3. Rotation
Encryption Speed (millisecond)	1.4569	2.5498	2.0712	5.5895	4.6377
Encryption Throughput (byte per second)	5,491	3,137	3,862	1,431	1,725
Encryption Throughput (block per second)	686	392	482	178	215
Cycles per Byte	491,691	860,558	699,029	1,886,463	1,565,217
Cycles per Block	3,933,531	6,884,462	5,592,233	15,091,703	12,521,739

8.5 Discussion

Analysis of secure cryptographic components of a lightweight block cipher is required before developing a new cryptographic algorithm. Besides that, conducting studies and experiments on the original algorithm is important to identify the weaknesses of the existing cryptographic design. In this section, LAO-3D block cipher is compared with RECTANGLE algorithm to justify the modification made to the original block cipher in producing a new algorithm. The encryption algorithm and key schedule algorithm were modified to enhance the security of the lightweight block cipher which were verified through cryptanalysis results presented in this chapter.

8.5.1 Encryption Algorithm

The aim of designing a lightweight encryption algorithm is to provide a smaller block size, less number of rounds, and simple cryptographic operations. Taking RECTANGLE as the base algorithm reference, three modifications were made to the original block cipher that includes reducing the number of encryption rounds, improving the substitution component, and enhancing the permutation component as listed in Table 8.17.

Table 8.17: Modifications of RECTANGLE Encryption Algorithm

Criteria	RECTANGLE (Original Algorithm)	Modification	LAO-3D (New Algorithm)
Block Size	64 bits	None	64 bits
No. of Encryption Rounds	25	Reduced encryption rounds	20
Structure	SPN	None	SPN
Functions	*AddRoundKey*	None	*AddRoundKey*
	SubColumn	Improved the substitution component using other S-box	*SubColumn*
	ShiftRow	Improved the permutation component using 3D bit rotation method	*Double3DRotation*

Firstly, lowering the number of encryption rounds from 25 to 20 rounds can increase the efficiency of the algorithm, takes lesser execution time, and consumes lower energy implementation. Determining the round number of an encryption algorithm heavily depends on the result of cryptanalysis attacks (Chen et al., 2020). This is sufficient to avoid successful attacks on the full rounds of the cipher that can lead to security problems and secret key exposure. From the cryptanalysis conducted on LAO-3D, the maximum number of rounds that can be attacked is in the 6^{th} round. Therefore, 20-round LAO-3D is enough to resist the differential and linear cryptanalysis that is discussed in Section 8.6 and 8.7.

Secondly, PRESENT S-box is implemented to replace the original RECTANGLE S-box due to its secure cryptographic properties as well as its small hardware footprint (Sherine et al., 2021). The new substitution method can improve the confusion property in the algorithm as proven through the avalanche effect tests in Section 8.2, 8.3, and 8.4.

Thirdly, the *Double3DRotation* function was introduced due to its capability to enhance the security strength of the block cipher with improvements to the diffusion property. The rotation method implemented in LAO-3D has optimized the randomization of the ciphertext as presented in Section 8.5.

8.5.2 Key Schedule Algorithm

Similar to the objective of designing a lightweight encryption algorithm, the key schedule algorithm prefers simple cryptographic operations to keep the simplicity of the block cipher structure (McKay et al., 2017). Three modifications were made to the original RECTANGLE key schedule algorithm that consists of reducing the number of round keys, increasing the length of constants, and improving the substitution component as shown in Table 8.18.

Table 8.18: Modifications of RECTANGLE Key Schedule Algorithm

Criteria	RECTANGLE (Original Algorithm)	Modification	LAO-3D (New Algorithm)
Key Size	64 bits	None	64 bits
No. of Round Keys	25	Reduced round keys	20
Functions	*RoundConstantsXOR*	Increased the length of constants/ nonce from 5 bits to 128 bits	*NonceXOR*
		Implemented modifiable nonce	
	SubkeyExtraction	None	*RoundKeyExtraction*
	SubColumn	Improved the substitution method using other S-box	*KeySubColumn*
	FeistelTransformation	None	*RowTransformation*

First of all, decreasing the number of rounds in the key schedule algorithm can lessen the burden of generating more round keys (Alassaf et al., 2019). Therefore, the modification can reduce the memory consumption to store the pre-generated round keys that are used during the encryption and decryption processes.

Other than that, expanding the bit length of the round constants or nonce can add extra security to the lightweight block cipher. On top of that, the implementation of modifiable nonce can avoid the algorithm from generating the same set of round keys when applying a unique secret key (Sehrawat & Gill, 2018).

Lastly, the implementation of PRESENT S-box as in the encryption algorithm provides confusion property to the key schedule algorithm (Modi et al., 2021). The confusion property can improve the generation of round keys that contribute to the randomization of the output ciphertext.

8.6 Chapter Summary

This chapter presented the cryptanalysis and software performance tests on the proposed LAO-3D block cipher. The experimental results of LAO-3D are compared with the existing block ciphers. Overall, LAO-3D algorithm has shown its security strength from the cryptanalysis tests results. Moreover, the performance tests conducted on the software implementation of LAO-3D show that the new algorithm is competitive among the existing block ciphers. Therefore, it is justified that LAO-3D algorithm fulfils the criteria of a lightweight block cipher and is suitable for security products such as mobile applications given its security and efficiency. In the following Chapter 9, the implementation of the LAO-3D algorithm is presented to show its functionality.

CHAPTER 9

IMPLEMENTATION OF LAO-3D LIGHTWEIGHT BLOCK CIPHER

9.1 Introduction

This chapter presents the implementation of LAO-3D lightweight block cipher on software applications. Two types of software development were carried out on LAO-3D algorithm that includes desktop and mobile applications to observe the functionality of the lightweight block cipher.

9.2 Software Implementation of LAO-3D Lightweight Block Cipher

In order to demonstrate the applicability of LAO-3D lightweight block cipher, two software implementations were conducted to observe the functionality of the algorithm. Due to limited resources, the implementations were built on a desktop application using C++ programming on Microsoft Visual Studio 2008 and a mobile application using Android Studio development software.

9.2.1 Desktop Application

Desktop application is a software program that can be executed on a personal computer to perform a specific task by an end-user. Microsoft Visual Studio is selected as the programming software since it is a very convenient and powerful application development software that works on most operating systems such as Windows, Linux, Android, and iOS. Therefore, an encryption desktop application was developed by implementing LAO-3D lightweight block cipher.

The desktop application can be used to secure sensitive data before sending it through email or storing it in a database. Data is encrypted within the application and does not depend on the underlying transport. When data is stored or transferred over the network, it remains encrypted until it reaches the destination of the application user that holds the encryption key, therefore, the security of the data is guaranteed.

9.2.1.1 Encryption

Encryption is a method for securing digital data using a cryptographic algorithm, along with a password. The encryption process converts the information into unreadable data to protect it from the unintended receiver. In order to provide data security through encryption, LAO-3D algorithm was implemented on Microsoft Visual Studio using the source code from APPENDIX D as displayed in Figure 9.1. Three steps are required to perform the encryption application that includes entering the encryption key, entering the message, and executing the data encryption.

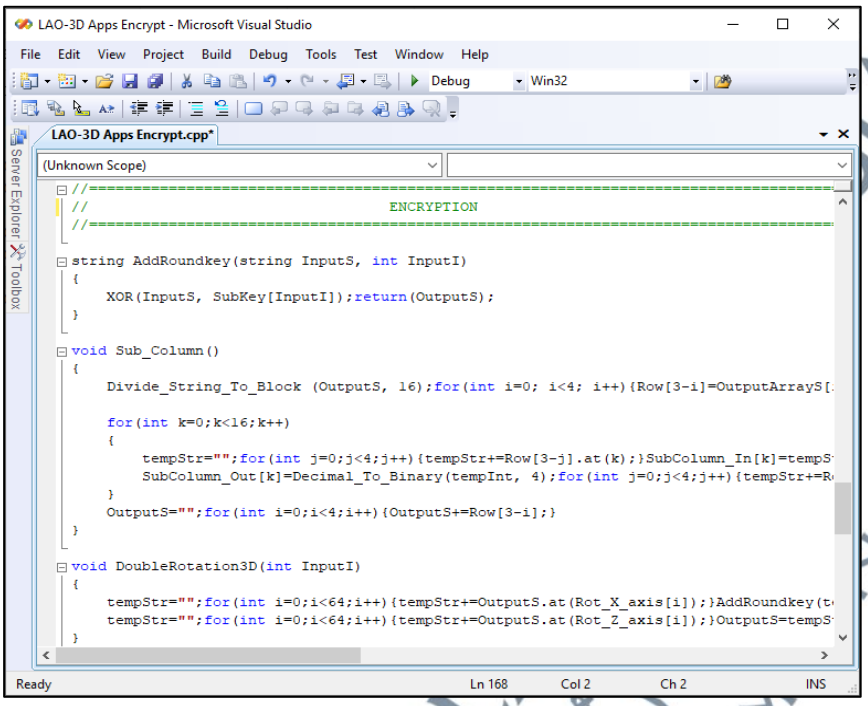
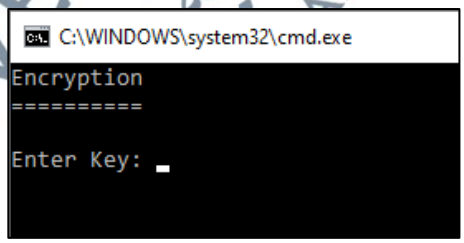


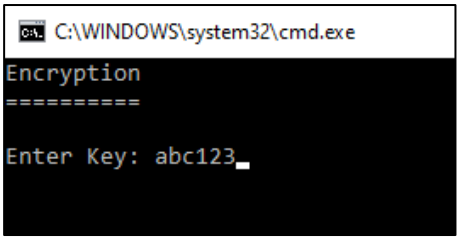
Figure 9.1: Encryption

i) **Step 1**

User is required to enter the encryption key as displayed in Figure 9.2. Encryption key is a secret word or phrase with a maximum of 16 characters long which is equal to the 128-bit key size of LAO-3D algorithm. The encryption key can be represented in form of letters, numbers, or special characters.



(i) Before entering the key

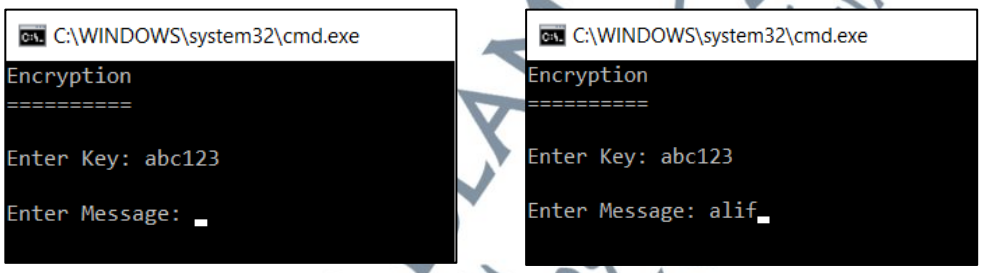


(ii) After entering the key

Figure 9.2: Input Encryption Key

ii) **Step 2**

User is required to enter the message as shown in Figure 9.3. Message is the information that needs to be protected through an encryption process. Similar to the encryption key, the message can be represented in form of letters, numbers, or special characters. The length of the message must not exceed 2,040 characters long or equal to 16,320 bits due to the limitation of the C++ programming of the Microsoft Visual Studio 2008.



(i) Before entering the message (ii) After entering the message

Figure 9.3: Input Message

iii) **Step 3**

After entering the encryption key and message, the application will execute an encryption process to generate the ciphertext as presented in Figure 9.4. The ciphertext is transformed into hexadecimal characters.

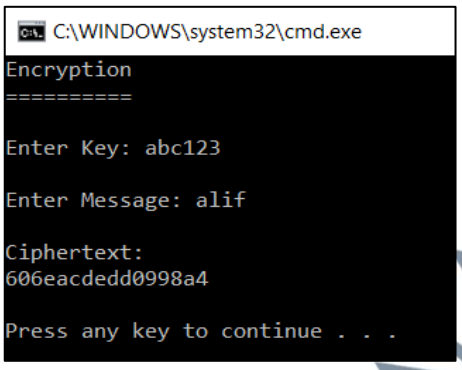


Figure 9.4: Output Ciphertext

9.2.1.2 Decryption

Decryption is a method of converting encrypted data into its original form using a cryptographic algorithm and a password. In general, decryption is a reverse process of an encryption method. The encrypted data is decoded to allow the authorized receiver to read the information. Similar to the encryption process implemented in the desktop application, LAO-3D algorithm was applied to Microsoft Visual Studio using the source code provided in APPENDIX E as displayed in Figure 9.5. Three steps are required to perform the decryption process which include entering the decryption key, entering the ciphertext, and executing the data decryption.

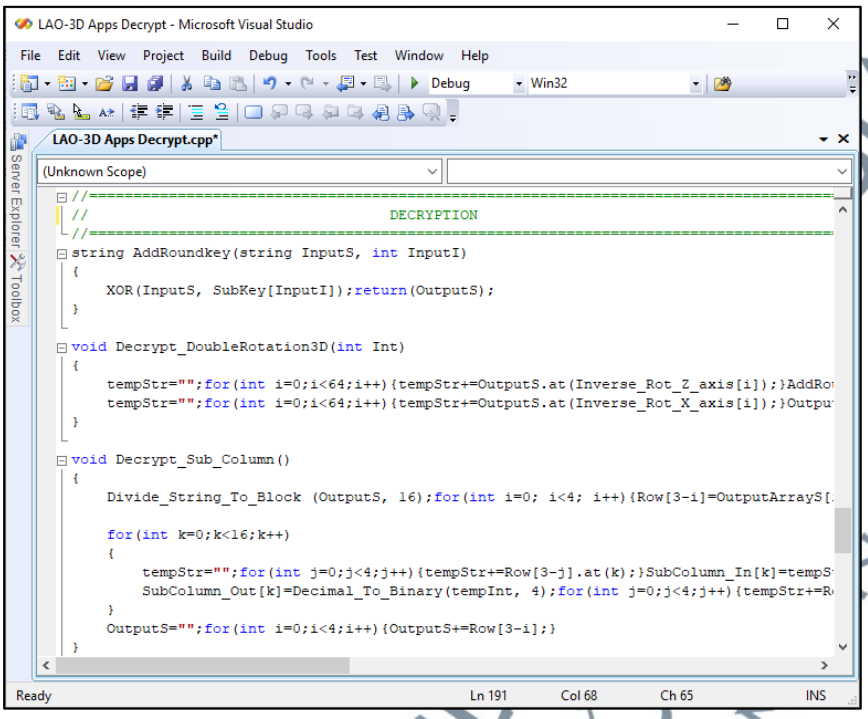
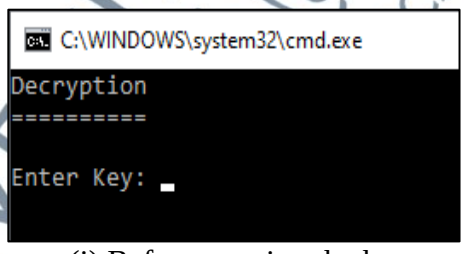


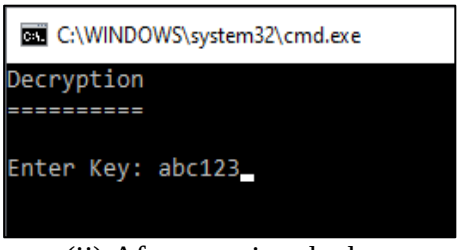
Figure 9.5: Decryption

i) **Step 1**

User is required to enter the decryption key as displayed in Figure 9.6. Decryption key is a secret word or phrase with a maximum of 16 characters long that can be represented in form of letters, numbers, or special characters.



(i) Before entering the key

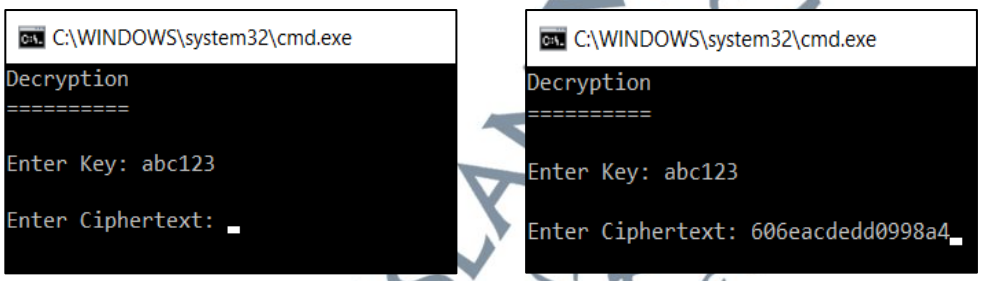


(ii) After entering the key

Figure 9.6: Input Decryption Key

ii) **Step 2**

User is required to enter the ciphertext as shown in Figure 9.7. Ciphertext is the data that needs to be recovered to reveal the message through a decryption process which is represented in form of hexadecimal characters. The length of the ciphertext must not exceed 4,080 hexadecimal characters long which equal 16,320 bits of data.



(i) Before entering the ciphertext (ii) After entering the ciphertext

Figure 9.7: Input Ciphertext

iii) **Step 3**

After entering the decryption key and ciphertext, the application will execute a decryption process to generate the plaintext as presented in Figure 9.8.

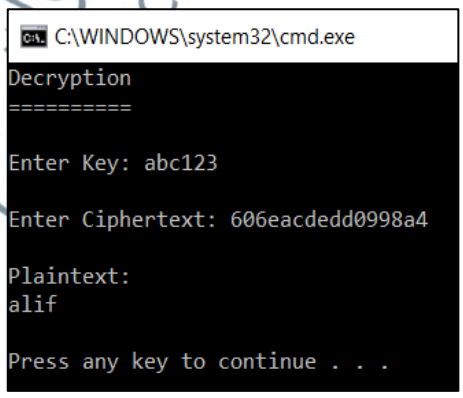
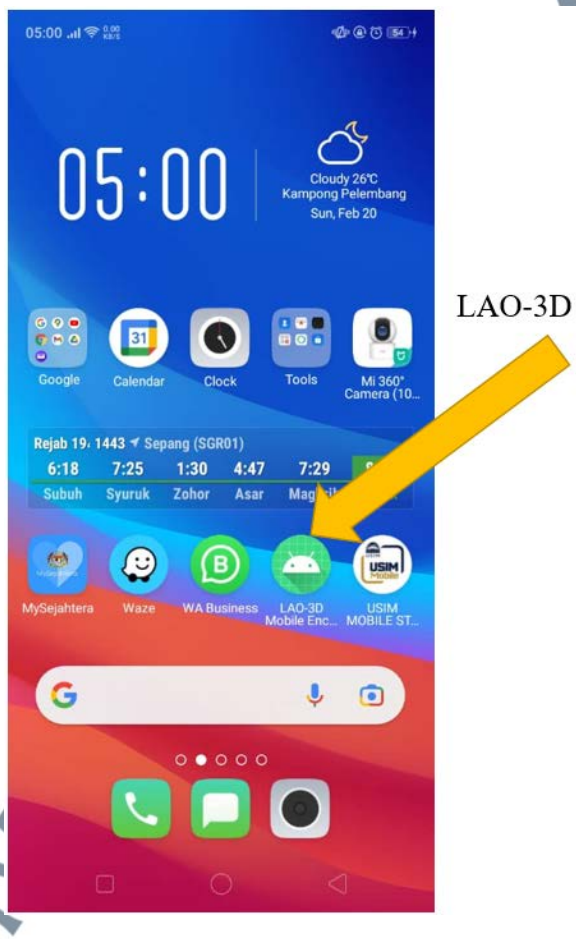


Figure 9.8: Output Plaintext

9.2.2 Mobile Application

Mobile application is a computer program or software application designed to run on a mobile device such as a smartphone, tablet, or watch. As the world's leading mobile operating system, a mobile encryption application was developed on Android by implementing LAO-3D lightweight block cipher as shown in Figure 9.9. Android Studio is selected as the development software due to its capability in building market-leading apps on every type of Android device.



LAO-3D Mobile Encryption App

Figure 9.9: Mobile Encryption Application

The mobile application was developed for data at rest encryption in which the encryption of the data (plaintext) is stored in the smartphone and is not moving through the internet. Only the encrypted data (ciphertext) will be transferred through the internet using any available instant messaging application (WhatsApp, Telegram, etc.) as shown in Figure 9.10. The application was designed specifically for offline data encryption to protect data from unauthorized access that can occur in online data encryption applications.

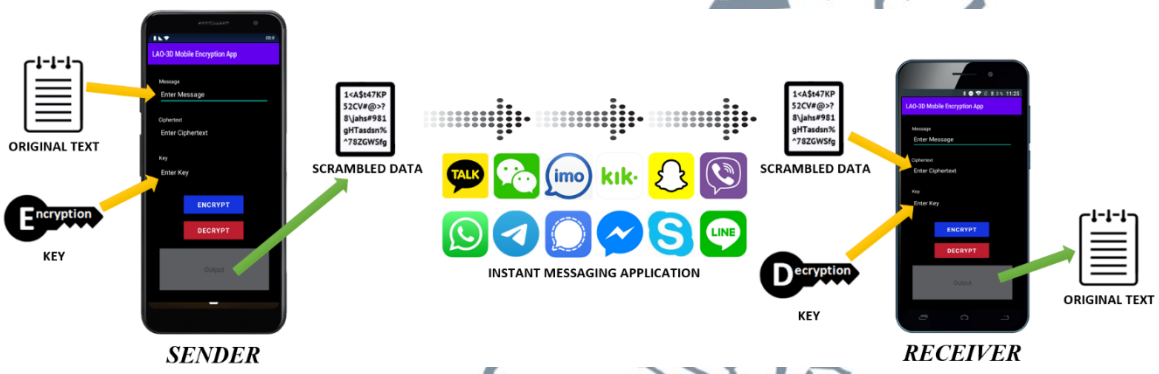


Figure 9.10: Encryption and Decryption Process

9.2.2.1 Encryption

In order to provide data security in mobile applications, LAO-3D lightweight block cipher was implemented on Android Studio (Bumblebee 2021.1.1) using the source code from APPENDIX D as displayed in Figure 9.11. Three steps are required to use the application that includes entering the encryption key, entering the message, and executing the data encryption in the mobile encryption application.

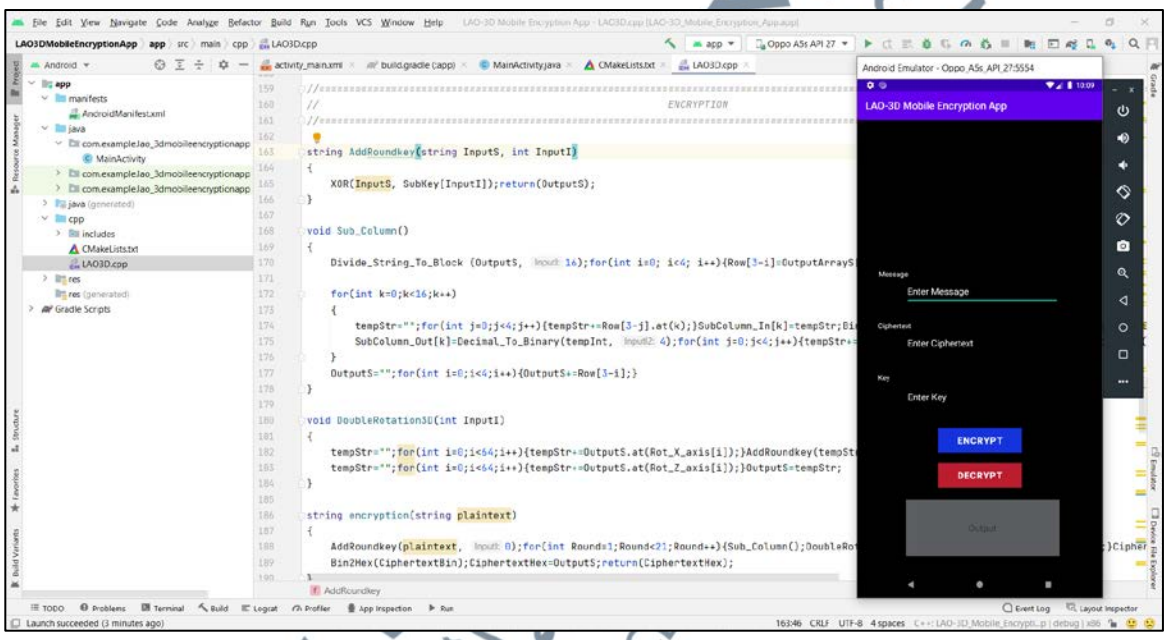
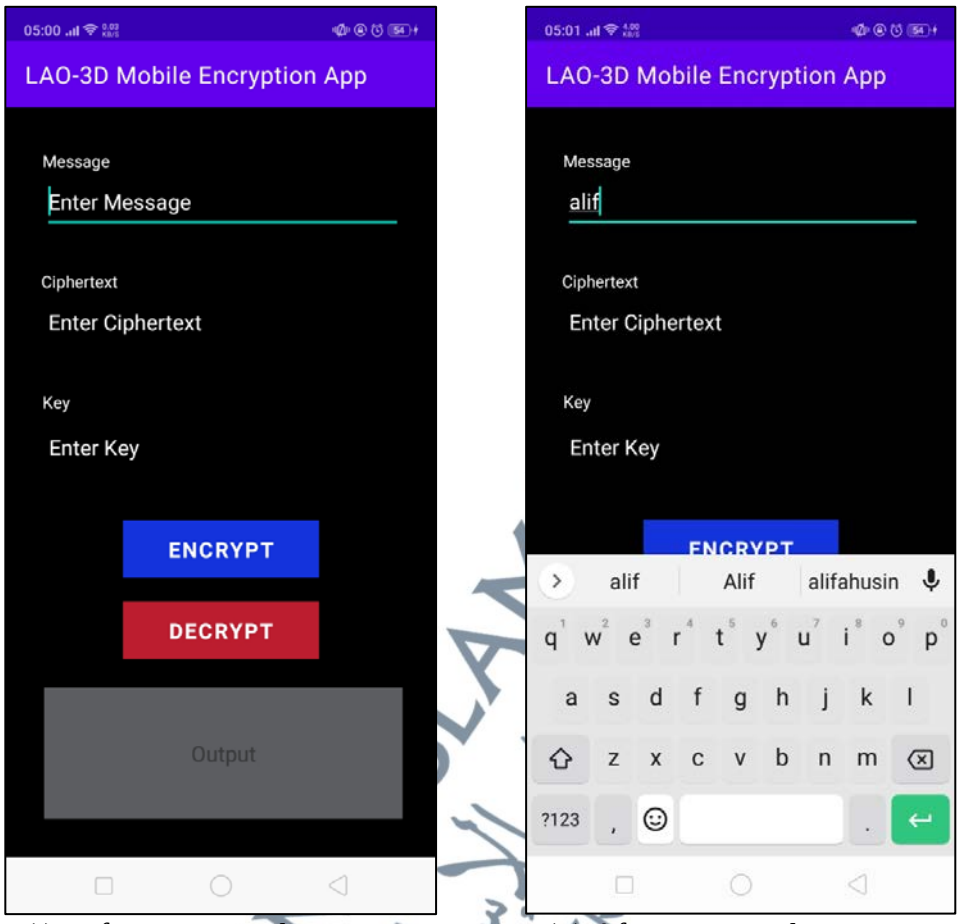


Figure 9.11: Encryption

i) **Step 1**

User is required to enter the message in the text box as shown in Figure 9.12. The message can be represented in form of letters, numbers, or special characters.

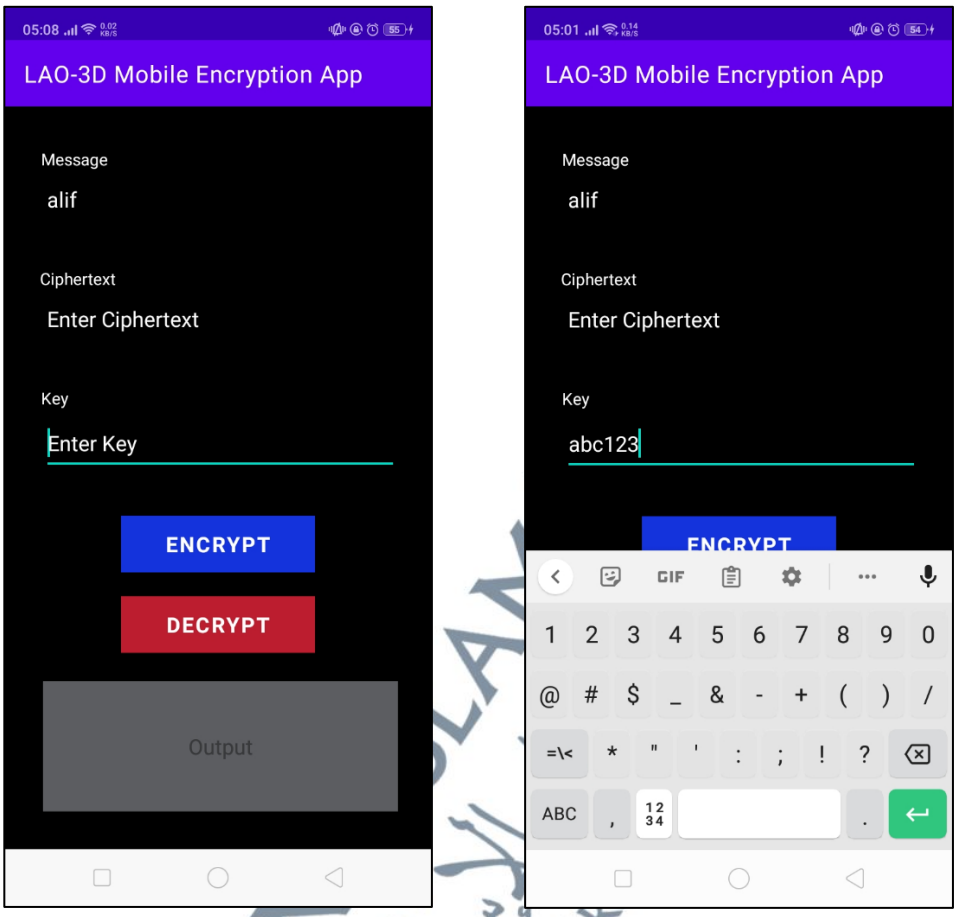


(i) Before entering the message (ii) After entering the message

Figure 9.12: Input Message

ii) **Step 2**

User is required to enter the encryption key in the text box as displayed in Figure 9.13. Similar to the message, the encryption key can be represented in form of letters, numbers, or special characters.

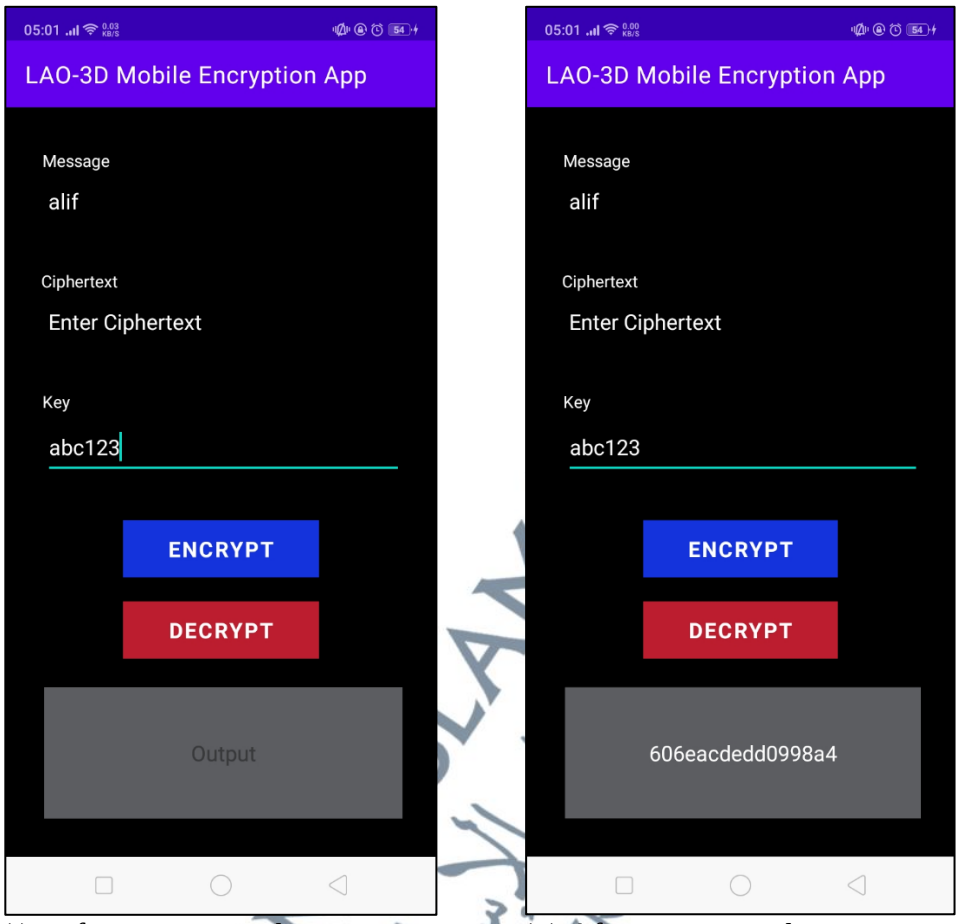


(i) Before entering the key (ii) After entering the key

Figure 9.13: Input Encryption Key

iii) **Step 3**

After entering the message and encryption key, the user is required to press the "Encrypt" button to generate the ciphertext. The ciphertext is transformed into hexadecimal characters as displayed in Figure 9.14.



(i) Before executing the encryption (ii) After executing the encryption

Figure 9.14: Output Ciphertext

9.2.2.2 Decryption

Similar to the encryption process implemented in the mobile application, LAO-3D lightweight block cipher was applied on Android Studio using the source code provided in APPENDIX E as displayed in Figure 9.15. Three steps are required to perform the decryption process which include entering the decryption key, entering the ciphertext, and executing the data decryption in the mobile encryption application.

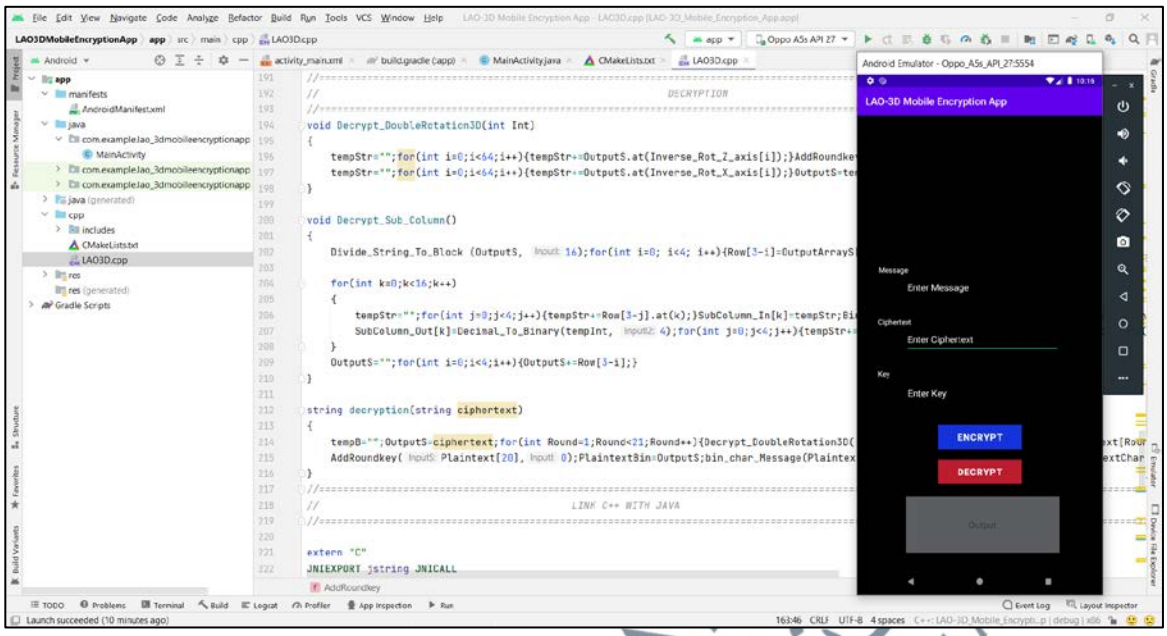
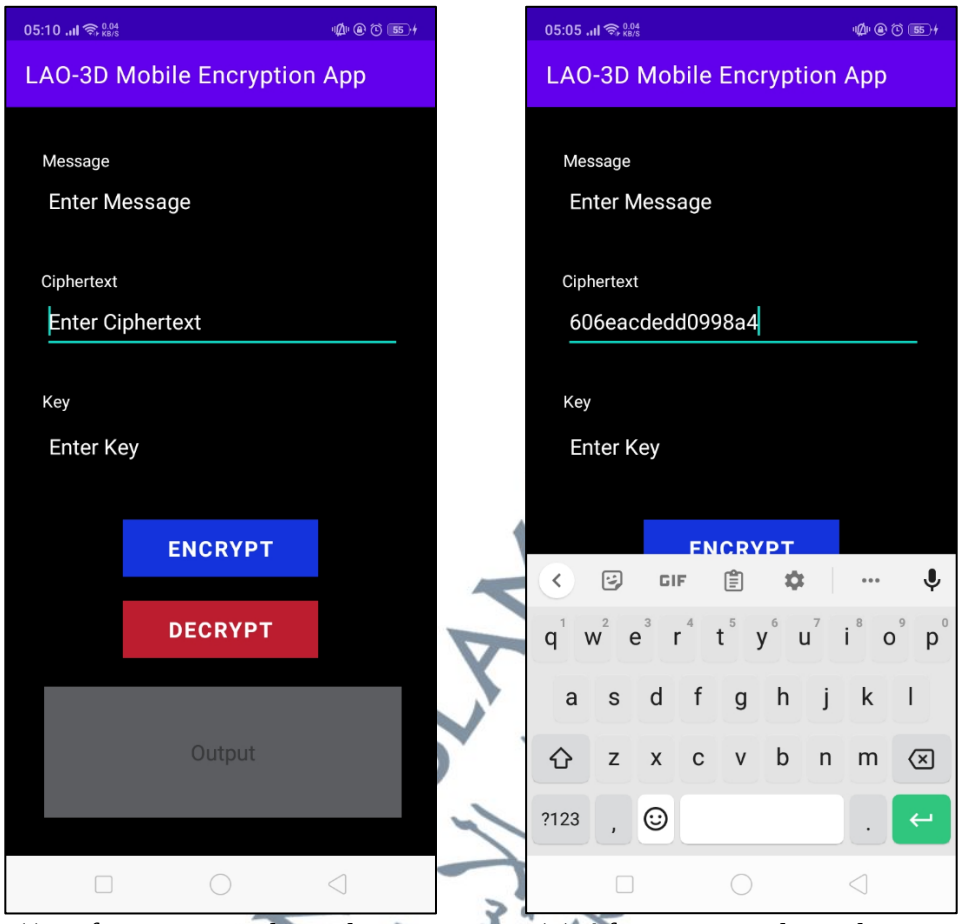


Figure 9.15: Decryption

i) Step 1

User is required to enter the ciphertext in the text box as shown in Figure 9.16.

The ciphertext is represented in the form of hexadecimal characters.

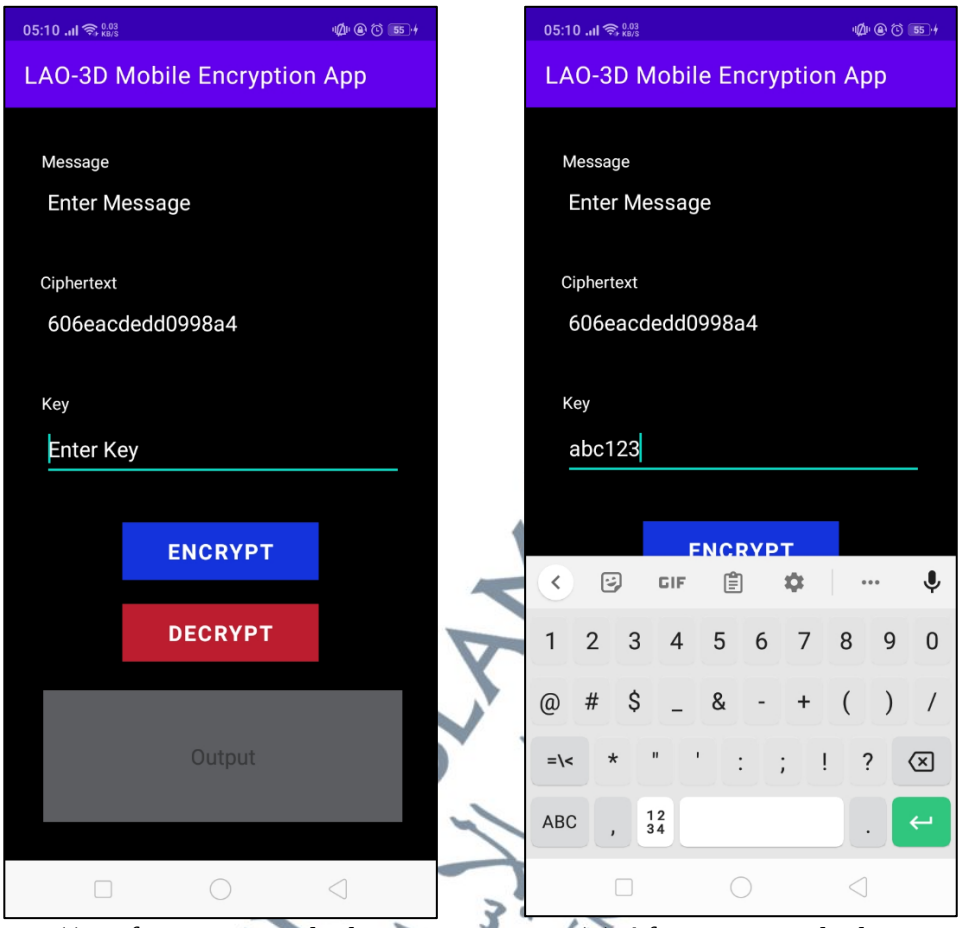


(i) Before entering the ciphertext (ii) After entering the ciphertext

Figure 9.16: Input Ciphertext

ii) **Step 2**

User is required to enter the decryption key in the text box as displayed in Figure 9.17. The decryption key can be represented in the form of letters, numbers, or special characters.

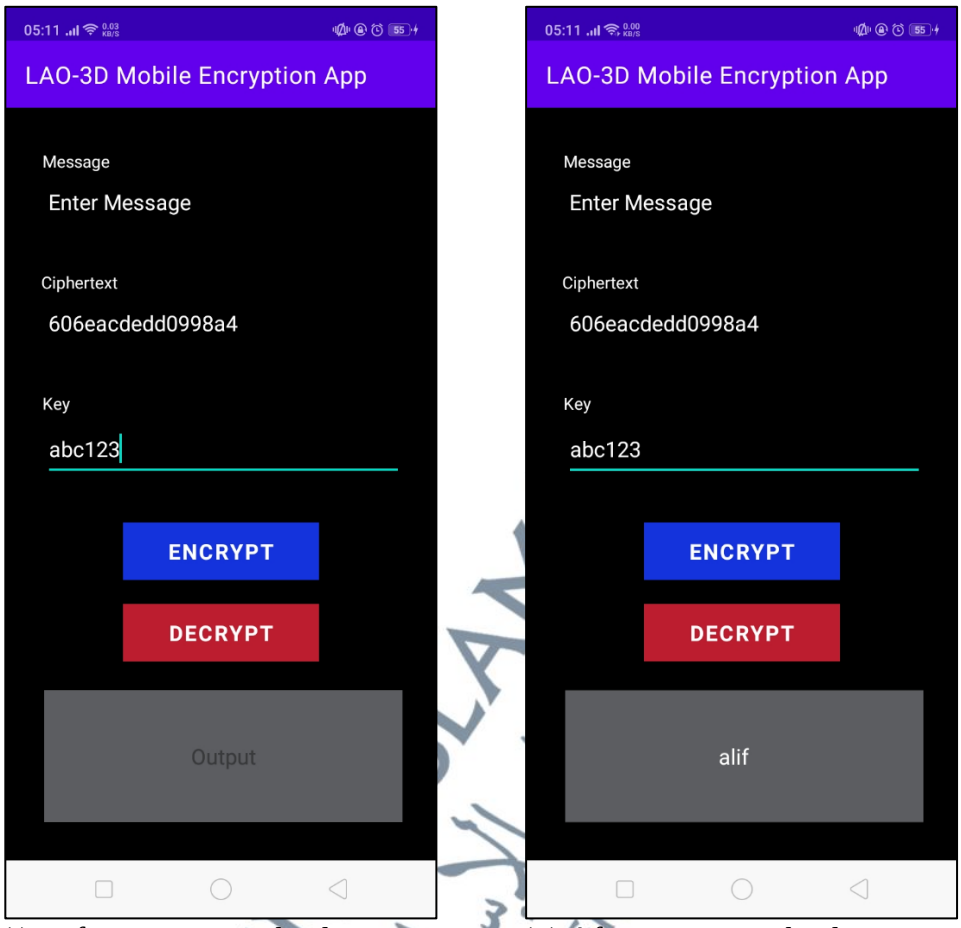


(i) Before entering the key (ii) After entering the key

Figure 9.17: Input Decryption Key

iii) **Step 3**

After entering the ciphertext and decryption key, the user is required to press the “Decrypt” button to generate the plaintext through a decryption process as displayed in Figure 9.18.



(i) Before executing the decryption (ii) After executing the decryption

Figure 9.18: Output Plaintext

9.3 Chapter Summary

Apart from the theoretical representation of the new algorithm design, two software implementations of LAO-3D lightweight block cipher were presented in the chapter. Firstly, a desktop application was developed by implementing the new lightweight block cipher to show the usefulness of the algorithm in a real application. The output from the application shows that the design, source code, and functionality of the LAO-3D works well as claimed in the thesis.

On the other hand, a mobile encryption application using Android Studio development software was established. This application gives users hands-on experience in using the data encryption application on their smartphones. For users with zero knowledge of cryptography, this mobile application can increase users' interest in knowing how actually encryption works. In addition, this approach can increase users' awareness of information security especially in protecting confidential data.