Impossible Differential Cryptanalysis of Reduced-Round LBlock

Ferhat Karakoç, Hüseyin Demirci, A. Harmancı

Impossible Differential Cryptanalysis of Reduced-Round LBlock

Ferhat Karakoç[1,2], Hüseyin Demirci[1] and A. Emre Harmancı[2]

[1] Tübitak BILGEM UEKAE, 41470, Gebze, Kocaeli, Turkey
{ferhatk, huseyind}@uekae.tubitak.gov.tr
[2] Istanbul Technical University, Computer Engineering Department, 34469, Maslak, Istanbul, Turkey
harmanci@itu.edu.tr

Abstract. In this paper, we improve the impossible differential attack on 20-round LBlock given in the design paper of the LBlock cipher. Using relations between the round keys we attack on 21-round and 22-round LBlock with a complexity of $2^{69.5}$ and $2^{79.28}$ encryptions respectively. We use the same 14-round impossible differential characteristic observed by the designers to attack on 21 rounds and another 14-round impossible differential characteristic to attack on 22 rounds of LBlock.

Key words: LBlock, differential cryptanalysis, impossible differential cryptanalysis, miss-in-the-middle attack.

1 Introduction

In recent years, lightweight cryptography has been getting prominent because of the growing computation research area which uses resource constraint devices such as RFID tags and sensor nodes. For this reason, many lightweight cryptographic algorithms have been designed such as PRESENT [3], PRINTcipher [5], and LED [4].

LBlock is a lightweight block cipher introduced at ACNS 2011 [7]. The number of rounds is 32 and the block and key lengths are 64 and 80 bits respectively. The designers of the algorithm give a 14-round impossible differential characteristic and attack on 20-round LBlock. To the best of our knowledge, there is only one cryptanalytic study on the algorithm [6]. The analysis includes differential attacks on 12 and 13 rounds and a related key impossible differential attack on 22-round LBlock.

In the impossible differential attack [1], the attacker tries to find a differential characteristic with a probability of 0 while in the differential cryptanalysis [2] a differential characteristic with a high probability is used.

In this paper, we improve the impossible differential attack given by the designers, using relations between rounds keys. We attack on 21-round and 22-round LBlock in a single key model with a complexity of $2^{69.5}$ and $2^{79.28}$ encryptions. In the attack on the 21-round cipher we use only the relations in the first

4 rounds while in the 22-round attack we use all relations between the round keys.

This paper is organized as follows. Section 2 includes the notation we use, a short description of LBlock and an overview of the 20-round attack done by the designers. We explain the impossible differential attack technique in Section 3. In Section 4, we attack on 21-round LBlock. An attack on 22-round LBlock is presented in Section 5. Finally, we conclude the paper in Section 6.

2 A Short Description of LBlock

Notation. Throughout this paper the following notations are used.

A	: a bit string
$A(i)$	: i-th nibble of A. The right most nibble is $A(0)$.
$A(i, j, ..., k)$	: concatenation of i, j, ... , k-th nibbles of A.
$A(i - j)$	: concatenation of i, $(i - 1)$, ..., j-th nibbles of A where $i \geq j$.
$A[i]$	: i-th bit of A. The right most bit of A is $A[0]$.
$A[i, j, ..., k]$	: concatenation of i, j, ..., k-th bits of A.
$A[i - j]$	: concatenation of i, $(i - 1)$, ..., j-th bits of A where $i \geq j$.
$A <<< i$	: i-bit cyclic shift of A.
$A\|\|B$	: concatenation of A and B.
K_i	: round key used in the i-th round.
K^i	: 80-bit value calculated in the key schedule.
X_i	: the leftmost 32-bit of the input of i-th round where X_0 is the rightmost 32-bit of the input of the first round.

LBlock. LBlock is a block cipher with 64-bit block and 80-bit key length. It consists of 32 rounds and one round is shown on the left in Figure 1. In the F function depicted on the right in Figure 1, firstly round key K_i is exored to the input of the function. After that, 4-bit S-Boxes and finally the permutation are applied.

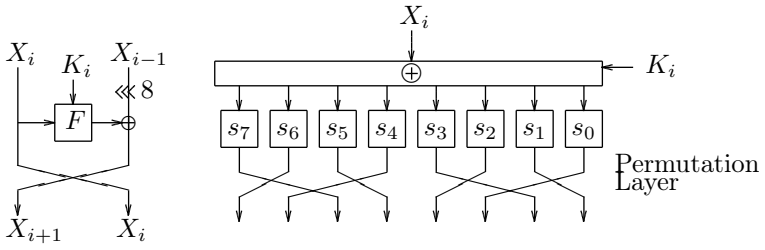

Fig. 1. i-th round of LBlock

The Encryption Process is as follows.

1. $(X_1||X_0) = P$

2. for $i = 1, 2, ..., 32$ do the following calculation
   $X_{i+1} = F(X_i, K_i) \oplus (X_{i-1} <<< 8)$
3. $C = (X_{32}||X_{33})$

The Key Schedule Process is as follows.

1. $K^0 = K$
2. $K_1 = K^0[79-48]$
3. for $i = 1, 2, ..., 31$ do the following calculations
   -$K^i = K^{i-1} <<< 29$
   -$K^i[79-76] = s_9[K^i[79-76]]$
   -$K^i[75-72] = s_8[K^i[75-72]]$
   -$K^i[50-46] = K^i[50-46] \oplus [i]_2$ where $[i]_2$ is the binary representation of the round index
   -$K_{i+1} = K^i[79-48]$

The S-boxes used in the encryption process and key schedule are given in Appendix B. For a complete description of the algorithm one can refer to [7].

Previous Work. In the design paper of LBlock, the designers attack on 20-round LBlock using the 14-round impossible differential characteristic $(00000000, 00 * 00000) \overset{14}{\nrightarrow} (0 * 000000, 00000000)$. They add 2 rounds at the top and 4 rounds at the bottom of the characteristic and assume that the round keys guessed in their attack are independent. This assumption doesn't change the correctness of the result but affects the complexity of the attack. The number of guessed key bits is 68 and there is a 36-bit sieving. They use 2^{63} chosen plaintexts and the complexity of the attack is $2^{72.7}$ encryptions. Using the relations between the round keys we improve the attack on 21 and 22 rounds.

3 The Impossible Differential Attack Technique

In this method, first an impossible differential characteristic is found. After finding a characteristic, several rounds are added at the top and at the bottom of the characteristic. Let E_1, E_0, and E_2 denote the encryption part which has the impossible differential characteristic, the added part before, and the added part after the characteristic respectively and the cipher $E = E_2 \circ E_1 \circ E_0$. Also, let the impossible differential be $\Delta\alpha \nrightarrow \Delta\beta$. The attack can be done in two ways. The first way is fallows. One plaintext pair P, P' is taken and guessing the keys in E_0 and E_2 the input and output differences of E_1 is calculated. In the case the input and output differences is the impossible differential characteristic, the guessed key is removed from the candidate key space. These steps are repeated using different plaintext pairs P, P' until a unique key is remained in the candidate key space. The complexity of the attack can be reduced if there is independence between guessed keys in E_0 and E_2 with the help of tables. One plaintext pair P, P' is taken and the keys which lead to the difference $\Delta\alpha$ at the end of E_0 are stored in a table whose name is A guessing the key bits used in E_0. Similarly,

the keys which lead to the difference $\Delta\beta$ at the top of E_2 are stored in another table whose name is B guessing the key bits used in E_2 and partially decrypting corresponding ciphertexts. Then, the keys in $A \times B$ are removed from the candidate key space.

The complexity of the attack can be calculated as follows. Let k and l denote the bit lengths guessed for E_0 and E_2 respectively and we have n-bit elimination in total. The number of pairs m required to eliminate all wrong candidates can be calculated as

$$(1 - 2^{-n})^m \times 2^{k+l} \leq 1 \Rightarrow m \geq (k + l) \times \ln 2 \times 2^n.$$

The time complexity of the attack will be $max(2^k \times m, 2^l \times m)$.

4 An Attack on 21-Round LBlock

In this attack we use the impossible differential characteristic given by the designers of the algorithm. This characteristic is $(00000000, 00 * 00000) \stackrel{14}{\nrightarrow} (0 * 000000, 00000000)$ which means that if there is a difference only in the 5-th nibble it is not possible having a difference only in the 14-th nibble after 14 rounds. We add 4 rounds at the top and 3 rounds at the bottom of this characteristic. Let 21-round LBlock starts with the 1-st round and ends with the 21-th round of LBlock. Attack on this 21-round LBlock can be executed using Algorithm 1.

Note that, in Algorithm 1 in Step 2, 3, 4, and 5 instead of guessing round keys we guess the bits of K^0 to use the relations between the round keys (see Table 1).

Table 1. Guessed master key bits in Step 2, 3, 4, and 5 in Algorithm 1

Step	the bits of K^0 which affect the round keys	# of guessed bits
2	$K^0[79-72, 51-48]$	12-bit
3	$K^0[63-56, 34-31, 26-23]$	16-bit
4	$K^0[79-78, 55-52, 22-19, 1-0]$	10-bit
5	$K^0[67-61, 46-43, 21-18]$	9-bit

In the attack, using 2^{50} chosen plaintext pairs we try to find 75 bits guessed and there exists 44-bit sieving. For a random key the probability of being remained in the candidate key set using only one P, P' pair is $(1 - 2^{-44})$. This probability will be $(1 - 2^{-44})^{2^{50}} \approx 2^{-92}$ when 2^{50} plaintext pairs are used which have the difference $(**00000*, *0*0*0**)$ in the plaintexts and the difference $(000*0**0, 00*0000*)$ in the ciphertexts. We guess 75 bits in the attack so $2^{-92} \times 2^{75} = 2^{-17}$ keys will remain in the candidate key set that means we can find the correct key with a high probability.

To have 2^{50} plaintext-ciphertext pairs which have the differences in the input and output we need $2^{50} \times 2^{44} = 2^{94}$ pairs having the input difference

Algorithm 1 Attack on 21-Round LBlock.

1: **for all** 2^{50} plaintext pairs which have the difference $\Delta(X_1, X0) = (**00000*, *0*0*0**)$ in the plaintexts and the difference $\Delta(X_{22}, X21) = (000*0**0, 00*0000*)$ in the ciphertexts for the reduced 21-round cipher **do**
2: **for all** $K_1(7, 6, 0)$, if $\Delta(X_2, X_1) = (0000*0*0, **00000*)$ then **do**
3: **for all** $K_1(3, 2)$ and $K_2(3, 1)$, if $\Delta(X_3, X_2) = (00000*00, 0000*0*0)$ then **do**
4: **for all** $K_1(1)$, $K_2(0)$ and $K_3(2)$, if $\Delta(X_4, X_3) = (00*00000, 00000*00)$ then **do**
5: **for all** $K_1(4)$, $K_2(6)$, $K_3(7)$ and $K_4(5)$, if $\Delta(X_5, X_4) = (00000000, 00*00000)$ then **do**
6: Insert the guessed key into the table A.
7: **end for**
8: **end for**
9: **end for**
10: **end for**
11: **for all** $K_{21}(5, 0)$, if $\Delta(X_{21}, X_{20}) = (00*0000*, *0000000)$ then **do**
12: **for all** $K_{21}(3)$ and $K_{20}(7)$, if $\Delta(X_{20}, X_{19}) = (*0000000, 0*000000)$ then **do**
13: **for all** $K_{21}(2)$, $K_{20}(1)$ and $K_{19}(6)$, if $\Delta(X_{19}, X_{18}) = (0*000000, 00000000)$ then **do**
14: Insert the guessed key into the table B.
15: **end for**
16: **end for**
17: **end for**
18: Remove the keys in the $A \times B$ from the candidate key set.
19: **end for**

$(**00000*, *0*0*0**)$. Using 2^{32} pairs having the same structure we can have $2^{32} \times 2^{31} = 2^{63}$ pairs. So we need to use $2^{94}/2^{63} = 2^{31}$ structures. Thus $2^{31} \times 2^{32} = 2^{63}$ plaintexts are required to apply the attack in Algorithm 1.

The time complexity of the attack can be calculated as follows. In Step 2 in the algorithm, we make $2^{51} \times 2^{12} = 2^{63}$ partial encryptions using 2^{51} data and guessing 12-bit key values. In Step 3, the number of operations is $2^{63} \times 2^{-12} \times 2^{16} = 2^{67}$ partial encryptions because of the sieving in Step 2 and 16-bit key guessing in Step 3. In Step 4, the number of partial encryptions is $2^{67} \times 2^{-8} \times 2^{10} = 2^{69}$ due to the 8-bit sieving and 10-bit key guessing. In Step 5, we perform $2^{69} \times 2^{-4} \times 2^9 \times 7 = 2^{74} \times 7$ s-box operations. The dominant number of operations is in Step 5. As a result, the complexity will be $\frac{2^{74} \times 7}{21 \times 8} \approx 2^{69.5}$ 21-round encryptions.

5 An Attack on 22-Round LBlock

In this attack, we use the impossible differential characteristic $(00000000, 000*0000) \overset{14}{\nrightarrow} (000000*0, 00000000)$. We add 4 rounds at the top and 4 rounds at the bottom of this characteristic. In this section, we use the relations between all round keys guessed in the attack. Also, we recover K^{19} instead of the master key

$K = K^0$. It is trivial to recover the master key using K^{19}. Algorithm 2 describes the attack on 22-round LBlock. In the attack, we recover 76 bits of K^{19} and there is 56-bit sieving.

Algorithm 2 Attack on 22-Round LBlock.

1: **for all** 2^{57} plaintext pairs which have the difference $\Delta(X_1, X_0) = (000*0**0, 0*0*0***)$ in the plaintexts and the difference $\Delta(X_{23}, X_{22}) = (00***0**, 0000***0)$ in the ciphertexts **do**
2: Run Algorithm 3.
3: Run Algorithm 4.
4: Remove the $K^{19}[79-39, 37-10, 6-0]$'s which lead to the round keys returned by Algorithm 3 and 4 from the candidate keys. Table 4 in Appendix A depicts the bits of K^{19} which determine the round keys.
5: **end for**

The complexity of Algorithm 3 can be calculated as follows. In Step 2, we guess 4 bits and perform two s-box operations for one pair in each guess and on the average one guess passes the condition. Thus we perform 2×2^4 s-box operations in Step 2. Similarly the total number of operations in Step 2-15 will be $2\times(2^4+2^4+2^4+2^4+2^8+2^8+2^{12}+2^{12}+2^{16}+2^{20}+2^{20}+2^{21}+2^{25}+2^{26}) \approx 2^{27.65}$ s-box look-up's which is approximately equivalent to $2^{20.19}$ 22-round LBlock encryptions. Note that the number of guessed key bits in Step 13 and 15 is 1 because of the relations between the round keys (see Table 2).

Table 2. The number of bits of K^0 guessed in Step 2-15 in Algorithm 3

Step	round keys	the bits of K^0	# of bits	Step	round keys	the bits of K^0	# of bits
2	$K_1(4)$	$K^0[67-64]$	4	3	$K_1(2)$	$K^0[59-56]$	4
4	$K_1(1)$	$K^0[55-52]$	4	5	$K_1(5)$	$K^0[71-68]$	4
6	$K_2(4)$	$K^0[38-35]$	4	7	$K_1(0)$	$K^0[51-48]$	4
8	$K_2(2)$	$K^0[30-27]$	4	9	$K_1(7)$	$K^0[79-76]$	4
10	$K_2(5)$	$K^0[42-39]$	4	11	$K_3(4)$	$K^0[9-6]$	4
12	$K_1(6)$	$K^0[75-72]$	4	13	$K_2(7)$	$K^0[50-47]$	1
14	$K_3(5)$	$K^0[13-10]$	4	15	$K_4(4)$	$K^0[60-57]$	1

The complexity of Algorithm 4 is approximately equivalent to $2^{19.59}$ 22-round encryptions (see Table 3 for the relations between round keys guessed in Algorithm 4). Thus the complexity of Algorithm 2 is $2^{57} \times (2^{20.19} + 2^{19.59}) \approx 2^{77.92}$ encryptions. When we use 2^{57} different pairs the number of 76-bit keys in the candidate space will be $(1-2^{-56})^{2^{57}} \times 2^{76} \approx 2^{74,56}$ because of the 56-bit sieving. Thus the total complexity to recover 80-bit K^{19} is $2^{77.92} + 2^{74.56+4} \approx 2^{79.28}$ encryptions.

Algorithm 3 Finding the keys which lead to the difference $\Delta\alpha$.

A plaintext pair which has the difference $\Delta(X_1, X_0) = (000 * 0 * *0, 0 * 0 * 0 * **)$ is given.
for all $K_1(4)$, if $\Delta S_4[X_1(4) \oplus K_1(4)] = \Delta X_0(4)$ then **do**
  for all $K_1(2)$, if $\Delta S_2[X_1(2) \oplus K_1(2)] = \Delta X_0(1)$ then **do**
    for all $K_1(1)$, if $\Delta S_1[X_1(1) \oplus K_1(1)] = \Delta X_0(6)$ then **do**
      for all $K_1(5)$ calculate $X_2(4)$ and **do**
        for all $K_2(4)$, if $\Delta S_4[X_2(4) \oplus K_2(4)] = \Delta X_1(4)$ then **do**
          for all $K_1(0)$ calculate $X_2(2)$ and **do**
            for all $K_2(2)$, if $\Delta S_2[X_2(2) \oplus K_2(2)] = \Delta X_1(1)$ then **do**
              for all $K_1(7)$ calculate $X_2(5)$ and **do**
                for all $K_2(5)$ calculate $X_3(4)$ and **do**
                  for all $K_3(4)$, if $\Delta S_4[X_3(4) \oplus K_3(4)] = \Delta X_2(4)$ then **do**
                    for all $K_1(6)$ calculate $X_2(7)$ and **do**
                      for all $K_2(7)$ calculate $X_3(5)$ and **do**
                        for all $K_3(5)$ calculate $X_4(4)$ and **do**
                          for all $K_4(4)$ check if $\Delta S_4[X_4(4) \oplus K_4(4)] = \Delta X_3(4)$ then **do**
                            Store the round keys in Table A.
                          end for
                        end for
                      end for
                    end for
                  end for
                end for
              end for
            end for
          end for
        end for
      end for
    end for
  end for
end for
Return Table A.

To have 2^{57} pairs having the input difference $(000*0**0, 0*0*0***)$ and the output difference $(00***0**, 0000***0)$ $2^{57} \times 2^{32} = 2^{89}$ pairs having the input difference $(000*0**0, 0*0*0***)$ are required. Using 2^{32} pairs having the same structure we can have $2^{32} \times 2^{31} = 2^{63}$ pairs. Thus $2^{89}/2^{63} = 2^{26}$ structures are needed. As a result, $2^{26} \times 2^{32} = 2^{58}$ plaintexts are required to apply the attack.

6 Conclusion

In this work, we have improved the attack done by the designers and attacked on 21-round and 22-round LBLock having a complexity of $2^{69.5}$ and $2^{79.28}$ respectively. In the designer's attack, it is assumed that all round keys are independent. In the proposed 21-round attack we use the relation between the rounds keys in

Algorithm 4 Finding the keys which lead to the difference $\Delta\beta$.

A ciphertext pair which has the difference $\Delta(X_{23}, X_{22}) = (00***0**, 0000***0)$ is given.
for all $K_{22}(3)$, if $\Delta S_3[X_{22}(3) \oplus K_{22}(3)] = \Delta X_{23}(1)$ then **do**
  for all $K_{22}(2)$, if $\Delta S_2[X_{22}(2) \oplus K_{22}(2)] = \Delta X_{23}(2)$ then **do**
    for all $K_{22}(1)$, if $\Delta S_1[X_{22}(1) \oplus K_{22}(1)] = \Delta X_{23}(0)$ then **do**
      for all $K_{22}(7)$ calculate $X_{21}(3)$ and **do**
        for all $K_{21}(3)$, if $\Delta S_3[X_{21}(3) \oplus K_{21}(3)] = \Delta X_{22}(1)$ then **do**
          for all $K_{22}(5)$ calculate $X_{21}(2)$ and **do**
            for all $K_{21}(2)$, if $\Delta S_2[X_{21}(2) \oplus K_{21}(2)] = \Delta X_{22}(3)$ then **do**
              for all $K_{22}(0)$ calculate $X_{21}(0)$ and **do**
                for all $K_{21}(0)$ calculate $X_{20}(0)$ and **do**
                  for all $K_{20}(0)$, if $\Delta S_0[X_{20}(0) \oplus K_{20}(0)] = \Delta X_{21}(2)$ then **do**
                    for all $K_{22}(6)$ calculate $X_{21}(5)$ and **do**
                      for all $K_{21}(5)$ calculate $X_{20}(2)$ and **do**
                        for all $K_{20}(2)$ calculate $X_{19}(1)$ **do**
                          for all $K_{19}(1)$, if $\Delta S_1[X_{19}(1) \oplus K_{19}(1)] = \Delta X_{20}(0)$ then **do**
                            Store the round keys in Table B.
                          end for
                        end for
                      end for
                    end for
                  end for
                end for
              end for
            end for
          end for
        end for
      end for
    end for
  end for
end for
Return Table B.

the first 4 rounds. Also, we use all of the relations between the round keys in the first 4 rounds and the last 4 rounds to attack on 22 round-LBlock.

Acknowledgments. This work was supported by the project COGSA.

References

1. Eli Biham, Alex Biryukov, and Adi Shamir. Cryptanalysis of skipjack reduced to 31 rounds using impossible differentials. In Jacques Stern, editor, *EUROCRYPT*, volume 1592 of *Lecture Notes in Computer Science*, pages 12–23. Springer, 1999.
2. Eli Biham and Adi Shamir. Differential cryptanalysis of des-like cryptosystems. *J. Cryptology*, 4(1):3–72, 1991.

Table 3. The number of bits of K^{19} guessed in Step 2-15 in Algorithm 4

Step	round keys	K^{19}	# of bits	Step	round keys	K^{19}	# of bits
2	$K_{22}(3)$	$K^{19}[5-2]$	4	3	$K_{22}(2)$	$K^{19}[79,78,1,0]$	4
4	$K_{22}(1)$	$K^{19}[77-74]$	4	5	$K_{22}(7)$	$K^{19}[21-18]$	4
6	$K_{21}(3)$	$K^{19}[34-31]$	4	7	$K_{22}(5)$	$K^{19}[13-10]$	4
8	$K_{21}(2)$	$K^{19}[30-27]$	4	9	$K_{22}(0)$	$K^{19}[73-70]$	4
10	$K_{21}(0)$	$K^{19}[22-19]$	1	11	$K_{20}(0)$	$K^{19}[51-48]$	4
12	$K_{22}(6)$	$K^{19}[17-14]$	4	13	$K_{21}(5)$	$K^{19}[42-39]$	4
14	$K_{20}(2)$	$K^{19}[59-56]$	4	15	$K_{19}(1)$	$K^{19}[4-1]$	0

3. Andrey Bogdanov, Lars R. Knudsen, Gregor Leander, Christof Paar, Axel Poschmann, Matthew J. B. Robshaw, Yannick Seurin, and C. Vikkelsoe. Present: An ultra-lightweight block cipher. In Pascal Paillier and Ingrid Verbauwhede, editors, *CHES*, volume 4727 of *Lecture Notes in Computer Science*, pages 450–466. Springer, 2007.
4. Jian Guo, Thomas Peyrin, Axel Poschmann, and Matthew J. B. Robshaw. The led block cipher. In Bart Preneel and Tsuyoshi Takagi, editors, *CHES*, volume 6917 of *Lecture Notes in Computer Science*, pages 326–341. Springer, 2011.
5. Lars R. Knudsen, Gregor Leander, Axel Poschmann, and Matthew J. B. Robshaw. Printcipher: A block cipher for ic-printing. In Stefan Mangard and François-Xavier Standaert, editors, *CHES*, volume 6225 of *Lecture Notes in Computer Science*, pages 16–32. Springer, 2010.
6. Marine Minier and María Naya-Plasencia. Some preliminary studies on the differential behavior of the lightweight block cipher lblock. In Proceedings of ECRYPT Workshop on Lightweight Cryptography, 2011. http://www.uclouvain.be/.
7. Wenling Wu and Lei Zhang. Lblock: A lightweight block cipher. In Javier Lopez and Gene Tsudik, editors, *ACNS*, volume 6715 of *Lecture Notes in Computer Science*, pages 327–344, 2011.

A The Bits of K^{19} which Determines the Round Keys Guessed in Algorithm 3 and 4

Table 4. The bits of K^{19} which determines the round keys guessed in Algorithm 3 and 4

Round keys	The bits of K^{19}	Round keys	The bits of K^{19}
$K_1(7)$	[71-68,66-63,61]	$K_{19}(1)$	[4-1]
$K_1(6)$	[67-61,59-56,54]	$K_{20}(2)$	[59-56]
$K_1(5)$	[64-54,52-50]	$K_{20}(0)$	[51-48]
$K_1(4)$	[60-54,52-50]	$K_{21}(5)$	[42-39]
$K_1(2)$	[53-46]	$K_{21}(3)$	[34-31]
$K_1(1)$	[49-41,39]	$K_{21}(2)$	[30-27]
$K_1(0)$	[45-39,37-34,32]	$K_{21}(0)$	[22-19]
$K_2(7)$	[42-39,37-34,32]	$K_{22}(7)$	[21-18]
$K_2(5)$	[35-25,23-21]	$K_{22}(6)$	[17-14]
$K_2(4)$	[31-25,23-21]	$K_{22}(5)$	[13-10]
$K_2(2)$	[24-17]	$K_{22}(3)$	[5-2]
$K_3(5)$	[79-76,74-72,6-0]	$K_{22}(2)$	[79,78,1,0]
$K_3(4)$	[79-76,74-72,2-0]	$K_{22}(1)$	[77-74]
$K_4(4)$	[53-46]	$K_{22}(0)$	[73-70]

B The S-Boxes Used in LBlock

x	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
$s_0[x]$	14	9	15	0	13	4	10	11	1	2	8	3	7	6	12	5
$s_1[x]$	4	11	14	9	15	13	0	10	7	12	5	6	2	8	1	3
$s_2[x]$	1	14	7	12	15	13	0	6	11	5	9	3	2	4	8	10
$s_3[x]$	7	6	8	11	0	15	3	14	9	10	12	13	5	2	4	1
$s_4[x]$	14	5	15	0	7	2	12	13	1	8	4	9	11	10	6	3
$s_5[x]$	2	13	11	12	15	14	0	9	7	10	6	3	1	8	4	5
$s_6[x]$	11	9	4	14	0	15	10	13	6	12	5	7	3	8	1	2
$s_7[x]$	13	10	15	0	14	4	9	11	2	1	8	3	7	5	12	6
$s_8[x]$	8	7	14	5	15	13	0	6	11	12	9	10	2	4	1	3
$s_9[x]$	11	5	15	0	7	2	9	13	4	8	1	12	14	10	3	6