Impossible Differential on 8-Round MARS' Core

Eli Biham* Vladimir Furman†

March 15, 2000

Abstract

MARS is one of the AES finalists. The up-to-date analysis of MARS includes the discovery of weak keys, and Biham's estimation that a 12-round variant of MARS is breakable. This estimation was partly founded based on a 7-round impossible differential of the core of MARS. However, no such attack was presented to-date. In this paper we present two new longer impossible differentials of 8 rounds.

1 Introduction

MARS[5] is a block cipher designed by IBM as a candidate for the Advanced Encryption Standard selection process, and was accepted as one of the five finalists.

The up-to-date analysis of MARS includes weak keys, and Biham's estimation that MARS reduced to 12 rounds can be attacked[2]. This estimate was partially based on the existence of a 7-round impossible differential of MARS[1] (see [3, 4, 6] for more details on attacks using impossible differential). In this paper we introduce two 8-round impossible differentials of MARS' core.

2 An 8-Round Impossible Differential

We denote binary numbers with a subscript b, and a 32-bit binary numbers whose all bits except of bit i are all zero, and only bit i is one by $\delta_i = 0^{31-i}1^1 0^i_b$ (i.e., `1<<i` in C). We also denote a string of 0's (and 1's) of variable lengths (including zero length) by 0^*_b (and 1^*_b) and the complement of a bit-value x by $\bar{x}$ ($\bar{x} = 1 - x$).

*Computer Science Department, Technion - Israel Institute of Technology, Haifa 32000, Israel. biham@cs.technion.ac.il, http://www.cs.technion.ac.il/~biham/.

†Computer Science Department, Technion - Israel Institute of Technology, Haifa 32000, Israel. vfurman@cs.technion.ac.il.

The 7-round impossible wordwise (truncated) differential of MARS is of the form

$$(0,0,0,X) \overset{3\ rounds}{\rightarrow} (Y,0,0,0) \overset{1\ round}{\not\rightarrow} (0,0,0,W) \overset{3\ rounds}{\rightarrow} (Z,0,0,0)$$

where W, X, Y, and Z are non-zero, all pairs with differences of the form $(0,0,0,X)$ must have differences of the form $(Y,0,0,0)$ after 3 rounds, and similarly the differences $(0,0,0,W)$ always cause differences $(Z,0,0,0)$ after 3 rounds. However, there are no pairs with differences $(Y,0,0,0)$ such that the differences become $(0,0,0,W)$ after one round.

We observe that an extension of this impossible differential shows that when $W=\delta_{31}$ the intermediate one-round impossible differential can be replaced by a two-round impossible differential $(Y,0,0,0) \overset{2\ rounds}{\not\rightarrow} (0,0,0,\delta_{31})$, for some values of Y, leading to the following 8-round impossible differential for some values of X

$$(0,0,0,X) \overset{3\ rounds}{\rightarrow} (Y,0,0,0) \overset{2\ rounds}{\not\rightarrow} (0,0,0,\delta_{31}) \overset{3\ rounds}{\rightarrow} (\delta_{31},0,0,0).$$

In the following we describe the 3-round differentials with probability 1. Then, we describe why the 2-round intermediate differential is impossible, and for which values of Y. The conjunction of the various differentials to the 8-round impossible differentials is described at the end of this section.

2.1 The 3-Round Differentials with Probability 1

We denote additive difference by Δ, and XOR-differences by Δ_{xor}. In every round of MARS' core, every single 32-bit input word B, C and D influences only one 32-bit output word (on A, B and C respectively). Thus if we take the input difference of one of the foregoing to be non-zero (e.g., $\Delta B \neq 0$) and all others including ΔA to be 0 (e.g., $\Delta A = \Delta C = \Delta D = 0$), then we receive the output difference with only one non-zero difference. In particular, if we take some input difference $(0,0,0,X)$ where X is non-zero, we get the difference $(0,0,X_1,0)$ for some non-zero X_1 after one round, then the difference becomes $(0,X_2,0,0)$ for some non-zero X_2 after the next round. Finally, the difference becomes $(Y,0,0,0)$ for some non-zero Y after the third round. In total we get a 3-round truncated differential $(0,0,0,X) \rightarrow (Y,0,0,0)$ with probability 1.

Note that, if the least significant bits of X have the form $1\underbrace{0..0}_{i}$ $(i \geq 0)$, then the least significant bits of Y have the same form. It follows from the fact that the least significant bits of such form are preserved in both additive and XOR differences.

In the particular case $X=\delta_{31}$ we always get $Y=\delta_{31}$: We start with the following difference $(0,0,0,\delta_{31})$, i.e., $\Delta A_0 = \Delta B_0 = \Delta C_0 = 0, \Delta D_0 = \delta_{31}$. Since $\Delta A_0 = 0$, the mixings to B, C, and D have zero differences. Since the difference

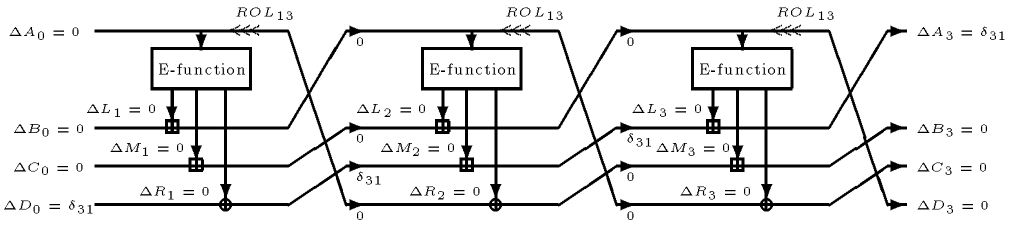

Figure 1: 3-round differential

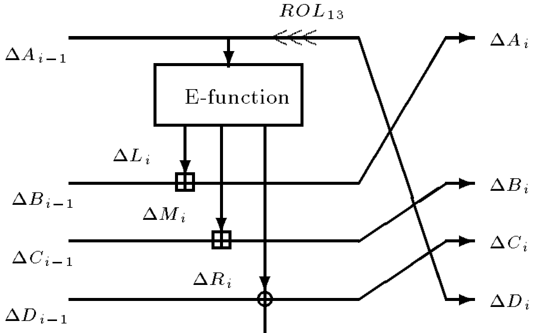

Figure 2: Round i in forward mode on MARS core

in ΔD is only in the most significant bit, this difference remains only in the most significant bit independently of whether the mixing operation is performed by addition or by XOR. Therefore, we get the difference $(\Delta A_1, \Delta B_1, \Delta C_1, \Delta D_1) = (0, 0, \delta_{31}, 0)$ after one round with probability 1. This can be repeated three times, and we get the difference $(\delta_{31}, 0, 0, 0)$ with probability one after 3 rounds, as shown in Figure 1. Notice, that this differential holds in all the rounds of the core including the forward mode, the backward mode and even on the boundary of both.

2.2 The 2-Round Impossible Differential

In this section we describe the 2-round impossible differential of MARS core.

Let $(\Delta A_0, \Delta B_0, \Delta C_0, \Delta D_0) = (Y, 0, 0, 0)$, where Y is an unknown value and $(\Delta A_2, \Delta B_2, \Delta C_2, \Delta D_2) = (0, 0, 0, \delta_{31})$. We want to find the values of Y that give impossible differential on a 2-round MARS core. We look for these values separately in the cases of forward and backward modes.

2.2.1 Forward Mode

Figure 2 outlines one round of the forward mode.

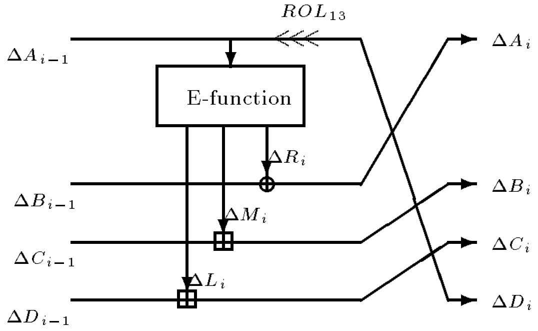

Figure 3: Round i backward mode on MARS core

- We know that $R_i = ((A_{i-1} <<< 13) \cdot K) <<< 10 = (D_i \cdot K) <<< 10$, where K is an unknown subkey. Because, the key used in this stage is odd and $\Delta D_2 = \delta_{31}$, we have that $\Delta_{xor} R_2 = \delta_9$.

- We have $\Delta_{xor} R_2 = \delta_9$ and $\Delta C_2 = \Delta_{xor} C_2 = 0$, so $\Delta_{xor} D_1 = \delta_9$. Thus, we receive $\Delta_{xor} A_0 = \delta_{28}$.

- $\Delta_{xor} A_0 = \delta_{28} \Rightarrow \Delta A_0 = aaa1\underbrace{0..0}_{28}{}_b$, where a is either 0 or 1 (i.e., $\Delta A_0 = \pm\delta_{28}$).

In total, we get that all values of Y, with possible exception of $\pm\delta_{28}$, give impossible differentials on a 2-round MARS core in the forward mode.

2.2.2 Backward Mode

The Figure 3 outlines the backward mode round.

- $\Delta D_2 = \delta_{31} \Rightarrow \Delta_{xor} D_2 = \delta_{31} \Rightarrow \Delta_{xor} A_1 = \delta_{18}$.

- $\Delta B_0 = 0 \Rightarrow \Delta_{xor} B_0 = 0$; Together with $\Delta_{xor} A_1 = \delta_{18}$ we get that $\Delta_{xor} R_1 = \delta_{18}$.

- $R_i = ((A_{i-1} <<< 13) \cdot K) <<< 10 = (D_i \cdot K) <<< 10$, so $\Delta_{xor}(D_i \cdot K) = \Delta_{xor} R_i >>> 10$. So $\Delta_{xor}(D_1 \cdot K) = \delta_{18} >>> 10 = \delta_8$, and $\Delta(D_1 \cdot K) = \Delta D_1 \cdot K = \pm\delta_8$. Because, the key used in this stage is odd, we have two important conclusions:

  1. ΔD_1 has $\underbrace{10..0}_{9}{}_b$ as a 9 least significant bits.
  2. We may look at this as $(\Delta D_1/2^8) \cdot (K \bmod 2^{24}) = \pm 1$. So the 24 least significant bits of the key are equal to the inverse of $\pm(\Delta D_1/2^8) \bmod 2^{24}$.

- On the other hand:

  $L_2 = (S[9 \text{ least significant bits of } (A_1 + K^+)] \oplus (R_2 >>> 5) \oplus R_2) <<< (5$ least significant bits of $R_2)$,

  where K^+ is an unknown subkey.

  - $\Delta_{xor} A_1 = \delta_{18}$ so the 9 least significant bits of ΔA_1 are 0, then $\Delta($9 least significant bits of $(A_1 + K^+)) = 0$, so $\Delta S = 0$ and thus $\Delta_{xor} S = 0$.
  - As in forward mode, we get $\Delta_{xor} R_2 = \delta_9$, so $\Delta_{xor}(R_2 >>> 5) = \delta_4$.
  - $\Delta_{xor}(S \oplus (R_2 >>> 5) \oplus R_2) = \underbrace{0..0}_{22}1000010000_b$.
  - A variable rotation is performed on L_2 by a number of bits derived from the 5 least significant bits of R_2. Because $\Delta_{xor} R_2 = \delta_9$ both rotations are by the same number of bits (denoted by r_l), so we have:

    $$\Delta_{xor} L_2 = \underbrace{0..0}_{22}1000010000_b <<< r_l.$$

  - After the rotation, the result is always of the form:

    $$\Delta_{xor} L_2 = \underbrace{0..0}_{30-i-j}\ 1\,\underbrace{0..0}_{j}\,1\,\underbrace{0..0}_{i}{}_b,$$

    where $j = 4$ or 26, and $i = 0..30 - j$.
  - Thus we have $\Delta L_2 = \underbrace{b..b}_{30-i-j}\ \bar{a}\,\underbrace{a..a}_{j}\,1\,\underbrace{0..0}_{i}{}_b$, where a,b are unknown bit values.

- Because $\Delta L_2 + \Delta D_1 = \Delta C_2 = 0$, we have that $\Delta D_1 = \underbrace{\bar{b}..\bar{b}}_{30-i-j}\ a\,\underbrace{\bar{a}..\bar{a}}_{j}\,1\,\underbrace{0..0}_{i}{}_b$. But we know that ΔD_1 has $\underbrace{10..0}_{9}{}_b$ as the 9 least significant bits, so only a single possibility remains:

  $$\Delta D_1 = \underbrace{\bar{b}..\bar{b}}_{18}\,a\,\underbrace{\bar{a}..\bar{a}}_{4}\,1\,\underbrace{0..0}_{8}{}_b.$$

  Observation: ΔD_1 may have four possible values:

  - $\underbrace{0..0}_{18}1\underbrace{0..0}_{4}1\underbrace{0..0}_{8}{}_b$ and $\underbrace{1..1}_{18}0\underbrace{1..1}_{4}1\underbrace{0..0}_{8}{}_b$ (i.e., $\pm\underbrace{0..0}_{18}1\underbrace{0..0}_{4}1\underbrace{0..0}_{8}{}_b$).
  - $\underbrace{0..0}_{18}0\underbrace{1..1}_{4}1\underbrace{0..0}_{8}{}_b$ and $\underbrace{1..1}_{18}1\underbrace{0..0}_{4}1\underbrace{0..0}_{8}{}_b$ (i.e., $\pm\underbrace{0..0}_{19}\underbrace{1..1}_{4}1\underbrace{0..0}_{8}{}_b$).

We have two pairs of possible values for ΔD_1, and thus there are only two possible values (one for each pair) for the 24 least significant bits of the key used in first round for multiplication (according to the conclusion in the beginning of this section(2.2.2)). These key values in hexadecimal form are $f07c1f_x$ (for $\Delta D_1 = \pm \underbrace{0..0}_{18} 1 \underbrace{0..0}_{4} 1 \underbrace{0..0}_{8}{}_b$) and $ef7bdf_x$ (for $\Delta D_1 = \pm \underbrace{0..0}_{19} \underbrace{1..1}_{4} 1 \underbrace{0..0}_{8}{}_b$).

- It is known that sequences of the form 01^*1_b or of the form 10^*1_b in the additive difference (Δ) are translated to the sequence of the form either $100^*1^*1_b$ or 01^*1_b in the corresponding XOR difference $(\Delta_{xor})^1$. Thus we have two options:

    1. $\Delta_{xor} D_1 = \underbrace{0^*1^*}_{18} \underbrace{100^*1^*}_{5} 1 \underbrace{0..0}_{8}{}_b$
    2. $\Delta_{xor} D_1 = \underbrace{0^*1^*}_{18} \underbrace{01..1}_{5} 1 \underbrace{0..0}_{8}{}_b$

- $\Delta_{xor} A_0 = \Delta_{xor} D_1 >>> 13$, so there are two possible values for $\Delta_{xor} A_0$:

    1. $\Delta_{xor} A_0 = \underbrace{00^*1^*}_{4} 1 \underbrace{0..0}_{8} \underbrace{0^*1^*}_{18} 1_b$
    2. $\Delta_{xor} A_0 = \underbrace{1..1}_{4} 1 \underbrace{0..0}_{8} \underbrace{0^*1^*}_{18} 0_b$

- In the first case, $\Delta_{xor} A_0$ is odd, so the ΔA_0 is odd too, and we cannot show that this case is impossible. In the second case, $\Delta_{xor} A_0$ is even so the ΔA_0 is even too, and therefore we can divide this case in two sub-cases:

    1. There is at least one 1 in $\underbrace{0^*1^*}_{18}{}_b$, so we have 10_b as two least significant bits in $\Delta_{xor} A_0$ and ΔA_0. This sub-case is impossible (see Appendix A for a detailed proof).

[1] For checking this fact, look at different cases of such sequence with and without carry from previous bits. For example, we take $\Delta I = 10..01_b$, i.e., $I^1 - I^2 = 10..01_b$. Then either:

1. The least significant bit of I^1 is 1: then the least significant bit of I^2 must be 0, and thus there is no carry to the next bit. On the other hand, the next bit in the difference is 0. Combining these together we conclude that the next bit in I^1 and the next bit in I^2 must be equal. Continuing in this way we get that $\Delta_{xor} I = 10..01_b$.
2. The least significant bit of I^1 is 0: then the least significant bit of I^2 must be 1, and thus there is a carry to the next bit. On the other hand, the next bit in the difference is 0. Combining these together we conclude that the next bit in I^1 and the next bit in I^2 have different values. Continuing in this way we get that the corresponding bits in I^1 and in I^2 are different till either: 1) in some bit I^1 has 1 and in I^2 has 0, or 2) we reach the most significant bits with difference 1 and, due to existence of a carry from the previous bits, this bit in I^1 and I^2 must have the same value. So $\Delta_{xor} I$ is equal either to $100^*1^*11_b$ or to $01..11_b$.

2. There are no 1's in $\underbrace{0^*1^*}_{18}{}_b$, so $\Delta_{xor}A_0 = \underbrace{1..1}_{4}1\underbrace{0..0}_{27}{}_b$, and ΔA_0 has $1\underbrace{0..0}_{27}{}_b$ as 28 least significant bits. For this sub-case, we cannot show that it is impossible.

Thus, we have a 2-round impossible differential for any **even** Y whose 28 least significant bits are not $\underbrace{10..0}_{28}{}_b$. For other Y's we cannot say anything whether there exist impossible differentials. However, if the differentials are not impossible for some Y, then the 24 least significant bits of the multiplication key used in the first round of the differential are either $f07c1f_x$ or $ef7bdf_x$.

2.3 Conjunction to the 8-Round Impossible Differentials

We want now to check what values of X give the 8-round impossible differentials. We describe the two cases in which the two middle rounds work in forward mode and in backward mode.

For forward mode, we have a 2-round impossible differential for any value of Y, except of $\pm\delta_{28}$. Because in $(0,0,0,X) \overset{3\ rounds}{\rightarrow} (Y,0,0,0)$ the relation between X to Y passes through two additions and one exclusive-or operation, the 29 rightmost bits remains $1\underbrace{0..0}_{28}{}_b$ and the 3 most significant bits may get any value. So, we have the 8-round impossible differentials $(0,0,0,X) \overset{8\ rounds}{\not\rightarrow} (\delta_{31},0,0,0)$ for all X, except of those with $\underbrace{10..0}_{29}{}_b$ as the 29 least significant bits.

For backward mode, we have a 2-round impossible differential for any even Y, except of those with $\underbrace{10..0}_{28}{}_b$ as 28 least significant bits. As in forward mode, in $(0,0,0,X) \overset{3\ rounds}{\rightarrow} (Y,0,0,0)$ the 28 least significant bits remains $\underbrace{10..0}_{28}{}_b$ and the 4 most significant bits may get any value. So we have the 8-round impossible differentials $(0,0,0,X) \overset{8\ rounds}{\not\rightarrow} (\delta_{31},0,0,0)$ for any even X, except of those with $\underbrace{10..0}_{28}{}_b$ as the 28 least significant bits.

3 Another 8-Round Impossible Differential

There is another 8-round impossible differential on MARS' core:

$$(0,0,0,\delta_{31}) \overset{3\ rounds}{\rightarrow} (\delta_{31},0,0,0) \overset{3\ rounds}{\not\rightarrow} (0,0,X,\delta_{31}) \overset{2\ rounds}{\rightarrow} (Y,\delta_{31},0,0),$$

where the 3 middle round are in backward mode, and X,Y are non-zero values such that X must have $\underbrace{0..0}_{24}{}_b$ as the least significant bits, and the 8 most significant bits of X may have any value (except of all zeroes). Thus, as was shown in the previous section, Y must have $\underbrace{0..0}_{24}{}_b$ as the least significant bits, and the 8 most significant bits may have any value (except of all zeroes). The explanation for this differential is similar to the explanation described earlier.

Acknowledgments

We would like to acknowledge the work of Alon Becker and Eran Richardson who made the initial observations in this direction.

References

[1] A. Becker, E. Richardson, Course Project, 1998.

[2] E. Biham, *A note on Comparing the AES Candidates*, Second AES Conference, March 1999.

[3] E. Biham, A. Biryukov, A. Shamir, *Cryptanalysis of Skipjack Reduced o 31 Rounds Using Impossible Differentials*, LNCS, Advanced in Cryptology - Proceeding of EUROCRYPT'99, Springer-Verlag 1999.

[4] E. Biham, A. Biryukov, A. Shamir, *Miss in the Middle Attacks on IDEA, Khufu, and Khafre*, LNCS 1636, Fast Software Encryption, pp. 124-138, March 1999.

[5] C. Burwick, D. Coppersmith, E. C'Avignon, R. Gennaro, S. Halevi, C. Jutla, S.M. Matyas, L. O'Connor, M. Peyravian, D. Safford, and N. Zunic, *"MARS - A Candidate Cipher for AES "*, NIST AES Proposal, June 1998.

[6] L. Knudsen, *DEAL - A 128-bit Block Cipher*, NIST AES Proposal, June 1998.

A Impossible differential for Y, with 10_b as least significant bits, in backward mode on MARS core.

In this appendix we show that the sub-case of backward mode where $\Delta_{xor}A_0 = \underbrace{1..1}_{4}1\underbrace{0..0}_{8}\underbrace{0^*1^*}_{17}10_b$, mentioned in section 2.2.2, is impossible.

- As in forward mode $\Delta D_2 = \delta_{31} \Rightarrow \Delta_{xor}R_2 = \delta_9$.
- $\Delta_{xor}R_2 \oplus \Delta_{xor}B_1 = \Delta_{xor}A_2 = 0$, so $\Delta_{xor}B_1 = \Delta_{xor}R_2 = \delta_9$.
- $\Delta_{xor}B_1 = \delta_9 \Rightarrow \Delta B_1 = \underbrace{a..a}_{22}1\underbrace{0..0}_{9}{}_b$, where a is unknown bit value.
- $\Delta C_0 + \Delta M_1 = \Delta B_1 = \underbrace{a..a}_{22}1\underbrace{0..0}_{9}{}_b$. Because $\Delta C_0 = 0$, $\Delta M_1 = \underbrace{a..a}_{22}1\underbrace{0..0}_{9}{}_b$.
- $\Delta M_1 = \underbrace{a..a}_{22}1\underbrace{0..0}_{9}{}_b \Rightarrow \Delta_{xor}M_1 = \underbrace{0^*1^*}_{22}1\underbrace{0..0}_{9}{}_b$.
- We know that $M_i = (A_{i-1} + K) <<<$ (low 5 bits of $(R_i >>> 5)$). However, because $\Delta_{xor}R_1 = \delta_{18}$, both rotations are by the same number of bits (denoted r_m), and because $\Delta K = 0$ we have

  $$\Delta M_1 = \Delta A_0 <<< r_m$$

  or

  $$\Delta A_0 = \Delta M_1 >>> r_m.$$
- We know that $\Delta_{xor}A_0 = \underbrace{1..1}_{4}1\underbrace{0..0}_{8}\underbrace{0^*1^*}_{17}10_b$. It gives us that $\Delta A_0 = \underbrace{x}_{4}\bar{a}\underbrace{a..a}_{9}\underbrace{z}_{17}10_b$, where x, z are unknown binary word and a is unknown bit value.
- The ΔA_0 has 10_b as 2 least significant bits, so the only one possibility for r_m to be 8. Thus $\Delta_{xor}(M_1 >>> 8) = \underbrace{0..0}_{8}\underbrace{0^*1^*}_{22}10_b$, and therefore, $\Delta(M_1 >>> 8) = \underbrace{b..b}_{8}\underbrace{y}_{22}10_b$, where b is an unknown bit value and y is unknown binary word.
- Now we have $\Delta(M_1 >>> 8) = \underbrace{b..b}_{8}\underbrace{y}_{22}10_b$ and $\Delta A_0 = \underbrace{x}_{4}\bar{a}\underbrace{a..a}_{9}\underbrace{z}_{17}10_b$. These must be equal. However, the bit 26th of the later differ than bit 27th, while bits 26th and 27th of the former are equal. This contradicts the fact that both values must be equal.