Collision attack on NaSHA-512

Li Ji[1], Xu Liangyu[1], and Guan Xu[2]

[1] Sony China Research Laboratory
[2] Mathematics department, Nankai university
{Ji.Li, Liangyu.Xu}@sony.com.cn
guanxu1984@mail.nankai.edu.cn

Abstract. The hash function NaSHA [1] is a new algorithm proposed for SHA-3. It follows the wide-pipe structure and compression function adopts quasigroup transformations. These properties of operation in quasigroup raise obstacles to analysis. However, the high probability difference to cause inner collision can be found in the quasigroup transformations. We propose a collision attack to NaSHA-512 with the complexity is 2^{192}, which is lower than the complexity of birthday attack to NaSHA-512. Using the similar method, we can find free-start collision on all versions with negligible complexity.

1 Description of NaSHA

NaSHA [1] is a hash functions family, defined as NaSHA-(m,k,r). It adopts linear transformations $LinTr_{2^s}$ and quasigroup transformations $\mathcal{MT}$. The parameters m denotes the length of hash value. The parameters k denotes the complexity of $\mathcal{MT}$ and the order 2^{2^r} of used quasigroup.

The main transformations of $\mathcal{MT}$ is defined by three transformations $\mathcal{A}_l$, ρ and $\mathcal{RA}_l$.

Definition 1 (The operation of quasigroup $*$).
The operation of quasigroup $*$ is build from the Extended Feistel Networks $F_{A,B,C}(L,R) = (r + A, L + B + f_{a_1,b_1,c1,a_2,b_2,c_2,a_3,b_3,c_3,\alpha,\beta,\gamma}(R + C))$, which is illustrated in Fig 1. The operation $*_{a_1,b_1,c_1,a_2,b_2,c_2,a_3,b_3,c_3,\alpha,\beta,\gamma,A,B,C}$ denoted by

$$x *_{a_1,b_1,c1,a_2,b_2,c_2,a_3,b_3,c_3,\alpha,\beta,\gamma,A,B,C} y = F_{A,B,C}(x \oplus y) \oplus y$$

is the quasigroup operation in $\mathbb{Z}_2^{64}$.

Definition 2 (Quasigroup additive string transformations $\mathcal{A}_l : Q^t \to Q^t$ with leader l). *. Let t be a positive integer, let $(Q, *)$ be quasigroup, $Q = (z)_{2^n}$, and $l, x_j, z_j \in Q$.*

$$\mathcal{A}_l(x_1, \dots x_t = (z_1, \dots z_t) \Leftrightarrow z_j = \begin{cases} (l + x_1) * x_1, & j = 1 \\ (z_{j-1} + x_j) * x_j, & 2 \leq j \leq t \end{cases}$$

where $+$ is addition modulo 2^n. The element l is said to be a leader of $\mathcal{A}$. The transformation is illustrated in Fig 2.

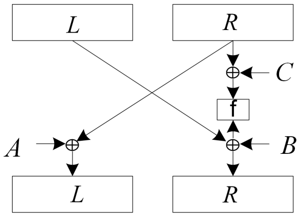


Fig. 1. The extended Feistel networks

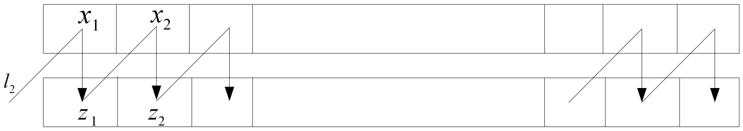


Fig. 2. The transformations $\mathcal{A}_l$

The definition of ρ and $\mathcal{RA}_l$ can be refer to the specification of NaSHA [1]. We ignore them because them have no relation with the attack.

We give a short description of NaSHA$(512, 2, 6)$. The compression of NaSHA$(512, 2, 6)$ adopts 2048-bit (32 words) state and output 512-bit hash value.

Firstly, the 512-bits message block M and the 512-bits initial value H form the state S alternately:

$$S = M_1||H_1||M_2||H_2||M_3||H_3||...||M_{16}||H_{16}$$

Secondly, update state words 32 times by the transformations of $LinTr_{512}$, which is defined by:

$$LinTr_{512}(S_1||S_2||...||S_{31}||S_{32}) = (S_7 \oplus S_{15} \oplus S_{25} \oplus S_{32})||S_1||S_2||...||S_{31})$$

Then choose parameters for the quasigroup transformations $\mathcal{MT}$ according the values of S_1 to S_{16}. And update the state one time by quasigroup transformations $\mathcal{MT}$.

After all message blocks have been processed, NaSHA(512,2,6) output:

$$NaSHA(512, 2, 6)(M) = S_4||S_8||...||S_{28}||S_{32}$$

2 Observations of NaSHA

We observed some properties, which help us to find collision in NaSHA-512.

Observation 1 (Differential of basic calculation) $(a + x) * x$ *is the basic calculation in the transformations* $\mathcal{A}_l$*, which is defined by the Extended Feistel Network.*

Given the input difference $\Delta x = \mathtt{0x00000000FFFFFFFF}$, the output difference of $(a+x)*x$ will always equate to zero when a and x satisfy the conditions $(a)_{64\ldots32} = \neg(x)_{64\ldots32}$, $(a)_{32} = 1$ and $(a)_{31\ldots1} = 0$. ($(x)_i$ denotes the i-th bit of x)

For example, given $x = \mathtt{0xAAAAAAAA00000000}$, $x' = \mathtt{0xAAAAAAAAFFFFFFFF}$ and $a = \mathtt{0x5555555580000000}$, $(a+x)*x = (a+x')*x'$ always holds no matter what parameters are set for the quasigroup operation $*$. The differential property attributes to the structure of Extended Feistel Network. The details are explained as follows.

$$\begin{aligned}
(a+x)*x &= F_{A,B,C}((a+x)\oplus x)\oplus x \\
&= F_{A,B,C}(\mathtt{0x5555555580000000})\oplus \mathtt{0xAAAAAAAA00000000} \\
&= ((\mathtt{0x80000000}\oplus A)\oplus \mathtt{0xAAAAAAAA}) \\
&\quad ||(f(\mathtt{0x80000000}\oplus C)\oplus B\oplus \mathtt{0x55555555}) \\
&= \\
(a+x')*x' &= F_{A,B,C}((a+x')\oplus x')\oplus x' \\
&= F_{A,B,C}(\mathtt{0xAAAAAAAA80000000})\oplus \mathtt{0xAAAAAAAAFFFFFFFF} \\
&= ((\mathtt{0x80000000}\oplus A)\oplus \mathtt{0xAAAAAAAA}) \\
&\quad ||(f(\mathtt{0x80000000}\oplus C)\oplus B\oplus \mathtt{0x55555555})
\end{aligned}$$

The calculations of $F_{A,B,C}$ are illustrated in Fig 3.

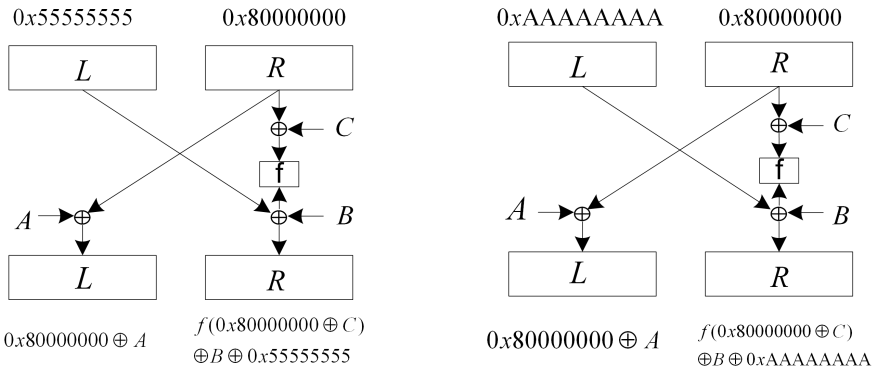


Fig. 3. The calculation of $F_{A,B,C}$

Observation 2 (The output of basic calculation) *According to the definition of $(a+x)*x$, the output value of $(a+x)*x$ can be changed by modifying the parameters A, B and C.*

Especially, given a and x, we can choose the parameters of A, B and C to make $(a+x)*x = a$. For the same parameters$(a_1, b_1, c_1, a_2, b_2, c_2, a_3, b_3, c_3, \alpha, \beta, \gamma, A, B, C)$, $(a+x')*x' = a$ always holds if the difference $\Delta x = x \oplus x' = \mathtt{0x00000000FFFFFFFF}$.

Observation 3 (Continuous collisions in $\mathcal{A}_l$) *According the observation 1 and the observation 2, difference sequence to generate continuous collisions in full transformation of $\mathcal{A}_l$ can be constructed easily.*

Firstly, select the triple x, x', a to make $(a+x)*x = (a+x')*x'$ for any quasigroup operation $*$. Secondly, for the basic basic calculation of $(z_{j-1}+x_j)*x_j$, if z_{j-1} and $x_j, x_{j+1}, \ldots, x_{j+k}$ (k denotes the length of the differential sequence) equate to a and x, we can select the parameters of the operation $*$ to make $(a+x)*x = a$ hold. Finally, after the transformation $\mathcal{A}_l$, all differences on the difference sequence will be absorbed.

We can control the state words before the transformation $\mathcal{A}_l$ freely due to the message input scheme. It is not easy to control the state words directly after$\mathcal{A}_l$, such as z_{j-1}. The continuous collision requires one word conditions (64 bits) on the first leader(z_{j-1}).

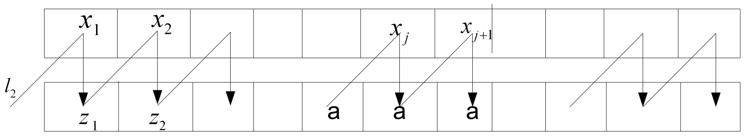


Fig. 4. Continuous collision in $\mathcal{A}_l$

Observation 4 (Difference absorption for parameters) *The first 16-words of state will be used as parameters of the quasigroup operations. However, it is easy to select differences on state words to make no difference on these parameters.*

For example: $\alpha_1||\beta_1||\gamma_1||\alpha_2 = S_7 + S_8$. If $\Delta S_7 = \Delta S_8 = \Delta x$ and $S_7 = x, S_7' = x', S_8 = x', S_8' = x$, parameters $\alpha_1, \beta_1, \gamma_1, \alpha_2$ have no differences.

Observation 5 (Freedom on state words) *For NaSHA-512, only 16-word out of 32-word are used, some state words can be changed freely while parameters of quasigroup transformation keeps.*

First 16-word of state is chose as parameters of quasigroup transformation $\mathcal{A}_l$ and $\mathcal{RA}_l$. Eight state words are selected as parameters of quasigroup transformation

$\mathcal{A}_l$ as follows:

$$\begin{aligned}
S_3 + S_4 &= l_2, \\
S_5 + S_6 &= a_1||b_1||c_1||a_2||b_2||c_2||a_3||b_3,\ c_3 = a_1, \\
S_7 + S_8 &= \alpha_1||\beta_1||\gamma_1||-, \\
S_{11} + S_{12} &= A||B, \\
S_{13} + S_{14} &= C|| - .
\end{aligned}$$

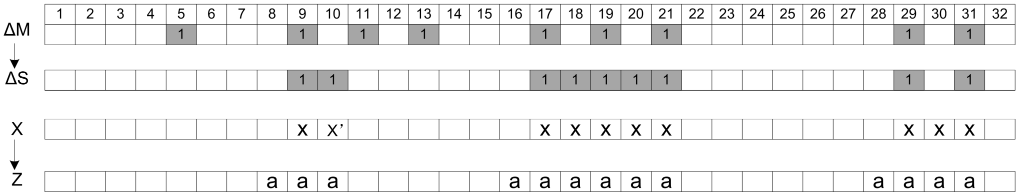
l_2 is the 64-bit leader of $\mathcal{A}_l$, the 8-bit words $a_1, b_1, c_1, a_2, b_2, c_2, a_3, b_3$, the 16-bit words $\alpha_1, \beta_1, \gamma_1$ and the 32-bit words A, B, C are parameters of the operation $*$. (The two $-$ denotes the values do not used in $\mathcal{A}_l$).

These observations can be used to construct collision in full transformation $\mathcal{A}_l$.

3 Collision attack of NaSHA-512

According to these Observations in section 2, we can choose differences on state words to find collision. The differential pattern is illustrated in Fig 5. We set three continuous differentials on state words, which results in the complexity of 2^{3*64} because three words conditions need to be fulfilled. We have enough free words to satisfied all conditions.

Following we explain the details.



	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	32
ΔM					1				1		1		1				1		1		1								1		1	
ΔS									1	1							1	1	1	1	1								1		1	
X									x	x’							x	x	x	x	x								x	x	x	
Z								a	a	a						a	a	a	a	a	a							a	a	a	a	

Fig. 5. The differential pattern

Step 1: Fix differences and values of state words.

We set differences on the state words after $LinTr_{512}$: $\Delta S_9 = \Delta S_{10} = \Delta S_{17} = \Delta S_{18} = \Delta S_{19} = \Delta S_{20} = \Delta S_{21} = \Delta S_{29} = \Delta S_{31} = \Delta x = \mathtt{0x00000000FFFFFFFF}$. No difference exists on other state words. Set the value of state words $S_9 = x$, $S_{10} = x'$ and set $S_{17}, S_{18}, S_{19}, S_{20}, S_{21}, S_{29}, S_{30}, S_{31}$ as x or x'.

Step 2: Find free state words for parameters of $(a + x) * x = a$.

We need right parameters A, B and C to make $(a+x)*x=a$. Denote H as initial value, $M^{32\times 16}_{LinTr_{512}}$ as the transformation matrix from the state S to H.

$$H=\begin{bmatrix} H_1 \\ H_2 \\ \cdots \\ \\ H_{16} \end{bmatrix} = M^{16\times 32}_{LinTr_{512}} \times \begin{bmatrix} S_1 \\ S_2 \\ \cdots \\ \\ S_{31} \\ S_{32} \end{bmatrix}$$

According the linear transformation of $LinTr_{512}$, we can get the algebraic equations among state words as follows.

$$S' = \begin{bmatrix} S_1 \\ S_2 \\ S_3 \\ S_4 \\ S_{12} \\ S_{13} \\ S_{15} \\ S_{16} \\ S_{22} \\ S_{23} \\ S_{24} \\ S_{25} \\ S_{26} \\ S_{27} \\ S_{28} \\ S_{32} \end{bmatrix} = H' \oplus S_{fix} \oplus \left[\begin{array}{llllll} & & S_7 \oplus & & S_{11} & \\ S_5 & & & & & \\ S_5 \oplus & S_6 & & & & \\ & & & & S_{11} & \\ & & S_7 \oplus & S_8 & & \\ & S_6 & & & & \\ S_5 \oplus & & S_7 \oplus & S_8 & & \\ S_5 \oplus & & S_7 & & & \\ & & S_7 \oplus & & & S_{14} \\ S_5 \oplus & S_7 & & & & \\ S_5 \oplus & S_7 & & & & \\ S_5 \oplus & & & S_8 & & \\ S_5 \oplus & & & S_8 \oplus & S_{11} & \\ S_5 & & & & & \\ S_6 \oplus & S_8 & & & & \\ & & & & & S_{14} \end{array}\right] \qquad (1)$$

Where H' and S_{fix} are constants vectors. H' denotes the linear relationship of initial value words, and S_{fix} denotes the linear relationship of these state words need to fix for the differential pattern, refer to Appendix A.

The values of 10 words need to be fixed to generated the differential pattern, and 16 state words in S' are limited by the 16 equations in (1). There are still 6 free words($S_5, S_6, S_7, S_8, S_{11}, S_{14}$)left. The right parameters of A, B and C can be calculated by:

$$S_{11} + S_{12} = S_{11} + (S_7 \oplus S_8 \oplus C_1), \tag{2}$$

$$S_{13} + S_{14} = S_{14} + (S_6 \oplus C_2). \tag{3}$$

Where C_1 and C_2 denote the fixed values in H' and S_{fix}. The 16 equations in (1) cost 16 state words, this two equations (2) and (3) need to be fulfilled and will cost 2 words out of 6 free words.

As a result, we can find 4 free state words. (For example, we can select S_5, S_6, S_7, S_8 as free state words, use S_{11} and S_{14} for the calculation of parameters.

Step 3: Find right state words to generate continuous collision of $\mathcal{A}_l$.

At this time, if after the transformation $\mathcal{A}_l$, these values of $S_8 = S_{16} = S_{28} = a$ hold, we always can make all differences on the state words absorbed by the transformation of $\mathcal{A}_l$. We can change these values of free state word randomly. For example, change S_5, S_6, S_7, S_8, and calculate the two values of S_{11} and S_{14}. The conditions(3 words, 192 bits) will cost 3 words out of 4 left free state words. Generally we need to try 2^{192} times.

Step 4: Calculated all 16 message words according selected state words by the inverse of $LinTr_{512}$.

Complexity analysis: The main complexity comes from the step 3, which need 2^{192} times call of $\mathcal{A}_l$. Step 4 needs one time calculation of the inverse of $LinTr_{512}$, which can be ignored. Finally, we can find collision to NaSHA-512 with the complexity of 2^{192}.

4 Free-start collision of NaSHA

Considering of free-start collision attack, we can find more differentials patterns. Fig 6 shows a free-start differential pattern of NaSHA-256. Fig 7 shows a free-start differential pattern of NaSHA-512.

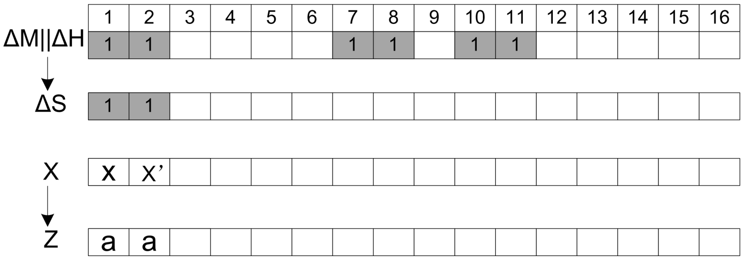
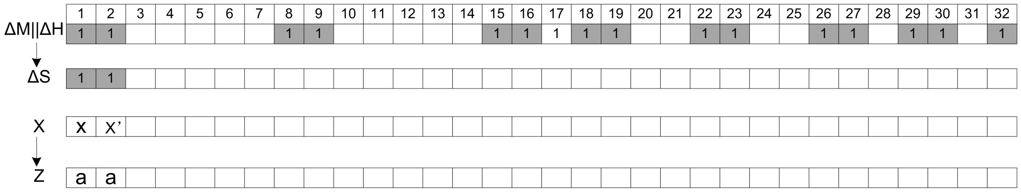


Fig. 6. The free-start differential pattern of NaSHA-256

In the two free-start differential patterns, differences only are deposited on S_1 and S_2. We select their values as: $S_1 = x$, $S_2 = x'$. Choose the value of S_3



	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	32
ΔM||ΔH	1	1						1	1						1	1	1	1	1			1	1			1	1		1	1		1
ΔS	1	1																														
X	x	x’																														
Z	a	a																														

Fig. 7. The free-start differential pattern of NaSHA-512

and S_4 to make: $S_3 + S_4 = a$. Using the similar steps in 3, we can get free-start collisions for all version of NaSHA-(m,k,r). The complexity is trivial. Appendix B gives examples of a message pair and initial values to make free-start collision on NaSHA.

5 Conclusion

NaSHA adopts quasigroup transformations, which raises an obstacle to analysis. However, we can find the differential with the high probability in quasigroup transformations. For NaSHA-512, only 16 words out of 32 words are used as parameters of quasigroup transformations. By analysis the algebraic structure of linear transformation, we can find a collision with the complexity of 2^{192}. The similar differential can be used to find free-start collision for all version with the negligible complexity.

References

1. Smile Markovski, Aleksandra Mileva, Algorithm Specifications of NaSHA, 2008. http://inf.ugd.edu.mk/images/stories/file/Mileva/Nasha.htm

A The linear relationships

H' denotes the linear relationship of initial value words(H_i) as follows.

$$H' = \begin{bmatrix} H_1 \oplus H_2 \oplus H_4 \oplus H_5 \oplus H_6 \oplus H_7 \oplus H_8 \oplus H_{12} \oplus H_{13} \oplus H_{16} \\ H_1 \oplus H_6 \\ H_6 \oplus H_{10} \\ H_2 \\ H_2 \oplus H_3 \oplus H_4 \oplus H_5 \oplus H_6 \oplus H_8 \oplus H_{10} \oplus H_{12} \oplus H_{14} \oplus H_{15} \oplus H_{16} \\ H_3 \\ H_2 \oplus H_3 \oplus H_5 \oplus H_6 \oplus H_8 \oplus H_{10} \oplus H_{11} \oplus H_{12} \oplus H_{14} \oplus H_{15} \oplus H_{16} \\ H_1 \oplus H_2 \oplus H_3 \oplus H_4 \oplus H_5 \oplus H_{10} \oplus H_{11} \oplus H_{12} \oplus H_{13} \oplus H_{14} \oplus H_{15} \oplus H_{16} \\ H_5 \oplus H_{12} \\ H_3 \oplus H_4 \oplus H_6 \oplus H_7 \oplus H_8 \oplus H_9 \oplus H_{10} \oplus H_{11} \oplus H_{12} \oplus H_{15} \oplus H_{16} \\ H_1 \oplus H_3 \oplus H_4 \oplus H_6 \oplus H_7 \oplus H_9 \oplus H_{10} \oplus H_{11} \oplus H_{12} \oplus H_{15} \oplus H_{16} \\ H_2 \oplus H_3 \oplus H_6 \oplus H_7 \oplus H_8 \oplus H_9 \oplus H_{10} \oplus H_{11} \oplus H_{14} \oplus H_{15} \\ H_2 \oplus H_3 \oplus H_5 \oplus H_6 \oplus H_8 \oplus H_9 \oplus H_{10} \oplus H_{11} \oplus H_{15} \\ H_6 \\ H_2 \oplus H_3 \oplus H_5 \oplus H_7 \oplus H_8 \oplus H_9 \oplus H_{11} \oplus H_{14} \oplus H_{15} \\ H_5 \oplus H_7 \oplus H_{12} \end{bmatrix}$$

S_{fix} denotes the linear relationship of these words need to fix for the differential pattern.

$$S_{fix} = \left[\begin{array}{cccccccccc}
 & & S_{17} \oplus & & S_{19} \oplus & & S_{21} \oplus & S_{29} \oplus & S_{30} & \\
S_9 \oplus & & S_{17} \oplus & & S_{19} \oplus & S_{20} \oplus & & & S_{30} & \\
 & S_{10} \oplus & & S_{18} \oplus & S_{19} \oplus & S_{20} \oplus & S_{21} \oplus & & S_{30} \oplus & S_{31} \\
 & & & & S_{19} \oplus & & & S_{29} & & \\
 & & S_{17} \oplus & S_{18} \oplus & S_{19} \oplus & S_{20} \oplus & & S_{29} \oplus & S_{30} \oplus & S_{31} \\
 & & & & & & S_{21} \oplus & & & S_{31} \\
 & & S_{17} \oplus & S_{18} \oplus & S_{19} \oplus & & & & S_{30} \oplus & S_{31} \\
 & & S_{17} \oplus & S_{18} \oplus & & & & & x_{30} & \\
 & & S_{17} \oplus & & & & & & & S_{31} \\
 & S_{10} \oplus & & & S_{19} \oplus & & S_{21} \oplus & & S_{30} \oplus & S_{31} \\
S_9 \oplus & S_{10} \oplus & & & S_{19} \oplus & & S_{21} \oplus & & S_{30} & \\
 & & & S_{18} \oplus & S_{19} \oplus & & x_{21} \oplus & S_{29} & & \\
 & S_{10} \oplus & S_{17} \oplus & & S_{19} \oplus & & & S_{29} & & \\
 & & & & S_{19} \oplus & S_{20} \oplus & & & S_{30} & \\
 & S_{10} \oplus & S_{17} \oplus & S_{18} \oplus & & S_{20} \oplus & & S_{29} \oplus & S_{30} \oplus & S_{31} \\
 & & S_{17} \oplus & & & & S_{21} \oplus & S_{29} \oplus & & S_{31}
\end{array}\right]$$

B Message pairs for free-start collision of NaSHA

B.1 Message pairs and initial values for NaSHA-224 and NaSHA-256

$M0$: (length: 512 bits)

```
ffffffff0000000000000080ffffffff0514ff7fffffff7fffffffff00000000
00000080ffffffff000000000000000000000000000000000000000000000000
```

$H0$:

```
0x7fffffff7fff1405, 0x0000000000000000,
0x0000000000000000, 0x0000000000000000,
0x00000000ffffffff, 0x80000000ffff1405,
0x0000000000000000, 0x0000000000000000
```

$M1$:(length:512 bits)

```
0000000000000000000000080ffffffff0514ff7fffffff7f0000000000000000
00000080ffffffffffffffff00000000000000000000000000000000000000000
```

$H1$:

```
0x7fffffff8000ebfa, 0x0000000000000000,
0x0000000000000000, 0x00000000ffffffff,
0x0000000000000000, 0x80000000ffff1405,
0x0000000000000000, 0x0000000000000000
```

The message digest of NaSHA-256 is:

```
d96e238f061ced9ab4fc687c33875efd29ec5def0dc7173e61c852b21967f58b
```

The message digest of NaSHA-224 is:

```
d96e238f061ced9ab4fc687c33875efd29ec5def0dc7173e61c852b2
```

B.2 Message pair and initial values for NaSHA-384 and NaSHA-512

$M0$: (length: 1024 bits)

```
000000000000000000000080ffffffff0514ff7fffffff7f0000000000000000
ffffffff00000000000000000000000000000000000000000000000000000000
00000080ffffffffffffffff000000000000000000000000ffffffff00000000
0000000000000000ffffffff000000000514ff7fffffff7f00000080ffffffff
```

$H0$:

```
0x0000000000000000, 0x0000000000000000,
0x0000000000000000, 0x00000000ffffffff,
0xffffffff80000000, 0x0000000000000000,
0x0000000000000000, 0x0000000000000000,
0x00000000ffffffff, 0xffffffff80000000,
0x7fffffff8000ebfa, 0xffffffff80000000,
0x00000000ffffffff, 0xffffffff80000000,
0x7fffffff7fff1405, 0x00000000ffffffff
```

$M1$: (length: 1024 bits)

```
ffffffff0000000000000080ffffffff0514ff7fffffff7f0000000000000000
000000000000000000000000000000000000000000000000ffffffff00000000
00000080ffffffff000000000000000000000000000000000000000000000000
00000000000000000000000000000000faeb0080ffffff7f00000080ffffffff
```

$H1$:

```
0x00000000ffffffff, 0x0000000000000000,
0x0000000000000000, 0x0000000000000000,
0xffffffff80000000, 0x0000000000000000,
0x0000000000000000, 0x00000000ffffffff,
0x0000000000000000, 0xffffffff80000000,
0x7fffffff7fff1405, 0xffffffff80000000,
0x0000000000000000, 0xffffffff80000000,
0x7fffffff8000ebfa, 0x0000000000000000
```

The message digest of NaSHA-512 is:

```
9401156aaa365b353fb7b3fd8a7d4ca944f4ba788c7fcfadbe1411e4adcbebd9
ecb7ecf86528134a30c639fb083ec658782d9fbfe730051e15458227e96c3dcf
```

The message digest of NaSHA-384 is:

```
9401156aaa365b353fb7b3fd8a7d4ca944f4ba788c7fcfadbe1411e4adcbebd9
ecb7ecf86528134a30c639fb083ec658
```