SM3 Cryptographic Hash Algorithm

CONTENTS

SM3 Cryptographic Hash Algorithm

1 Scope

This document specifies the SM3 cryptographic hash algorithm, including the details of the computation process. Furthermore, this document also gives examples for the computation of SM3 algorithm.

This document applies to digital signing and verification, the generation and verification of message authentication code, and the generation of random number under commercial cryptographic applications, and can meet security requirements of multiple cryptographic applications. Meanwhile, this document can also be used as the standard reference by the manufacturers of security products, and improve the credibility and interoperability of security products.

2 Terms and definitions

The following terms and definitions apply to this document.

2.1
Bit String

A binary sequence of 0's and 1's.

2.2
Big-endian

A format that describes the order in which a sequence of bytes is stored in computer memory. It defines that the most significant byte is stored at the lowest storage address and the least significant byte are stored at the highest address.

2.3
Message

A bit string with arbitrary finite length.

Note 1 to entry: In this document, a message is used as the input data of the hash algorithm

2.4

Hash Value

The output returned by a hash function.

Note 2 to entry: The length of the hash value in this document is 256 bits.

2.5

Word

A bit string of length 32.

3 Symbols

The following symbols apply to the document.

-- $ABCDEFGH$: 32-bit registers or the concatenation of the values.

-- $B^{(i)}$: the i-th message block.

-- CF : a compression function.

-- FF_j: a boolean function, the expression varies with j.

-- GG_j: a boolean function, the expression varies with j.

-- IV : the initial value, used to initialize the register of the compression function.

-- P_0 : the permutation involved in the compression function.

-- P_1 : the permutation involved in the message function.

-- T_j : a constant, which varies with j.

-- m : a message.

-- m': a message after padding.

-- mod: the modular operation.

-- $\wedge$: bitwise AND operation.

-- $\vee$: bitwise OR operation.

-- $\oplus$: bitwise XOR operation.

-- $\neg$: bitwise NOT operation.

-- $+$: addition modulo 2^{32} operation

-- $<<< k$: the circular left shift by k bit.

-- $\leftarrow$: the left assignment operation.

4 Constants and functions

4.1 Initial Value

IV = 7380166f 4914b2b9 172442d7 da8a0600 a96f30bc 163138aa e38dee4d b0fb0e4e

4.2 Constants

$$T_j = \begin{cases} 79cc4519, \text{ for } 0 \le j \le 15 \\ 7a879d8a, \text{ for } 16 \le j \le 63 \end{cases}$$

4.3 Boolean Functions

$$FF_j(X,Y,Z) = \begin{cases} X \oplus Y \oplus Z, \text{ for } 0 \le j \le 15, \\ (X \wedge Y) \vee (X \wedge Z) \vee (Y \wedge Z), \text{ for } 16 \le j \le 63, \end{cases}$$
$$GG_j(X,Y,Z) = \begin{cases} X \oplus Y \oplus Z, \text{ for } 0 \le j \le 15, \\ (X \wedge Y) \vee (\neg X \wedge Z), \text{ for } 16 \le j \le 63, \end{cases}$$

where *X, Y, Z* are 32-bit words.

4.4 Permutations

$$P_0(X) = X \oplus (X <<< 9) \oplus (X <<< 17),$$
$$P_1(X) = X \oplus (X <<< 15) \oplus (X <<< 23),$$

where *X* is 32-bit word.

5 Algorithm description

5.1 Overview

For a message *m* with length l ($l < 2^{64}$), SM3 hash algorithm generates a 256-bit hash value after padding and iterative compression .See Annex A for example of operation.

5.2 Padding

Assume a message has l bits. Firstly add the bit '1' to the end of this message, then add k bits of '0', such that k is the smallest non-negative integer satisfying $l + 1 + k \equiv 448 \bmod 512$. Then add a 64-bit bit string, which is the binary expression of length l. After padding, the length of the new message m' is a multiple of 512.

EXAMPLE For the message 01100001011000100110001 1, with length l =24, the bit string after padding is: $01100001011000100110001\overbrace{11\underbrace{00\ldots00}_{423\text{ bits}}}^{}\overbrace{\underbrace{00\ldots011000}_{\text{binary expression of } l}}^{64\text{ bits}}$.

5.3 Iterative Compression

5.3.1 Iteration Procedure

The padded message m' is split into 512-bit blocks, and denoted as $m' = B^{(0)}B^{(1)}\cdots B^{(n-1)}$, where $n = (l+k+65)/512$. The iteration procedure for m' is as follows:

FOR $i = 0$ TO $n-1$

$V(i+1) = CF(V(i), B(i))$

ENDFOR

Here , CF is the compression function , $V^{(0)}$ is the 256-bit IV , and $B^{(i)}$ is the *i*-th message block after padding. The result after iterative procedure is $V^{(n)}$.

5.3.2 Message Expansion

The message block $B^{(i)}$ is expanded to 132 words $W_0, W_1, \cdots, W_{67}, W_1', \cdots W_{63}'$, which are applied to compression function CF :

a. Split message block $B^{(i)}$ into 16 words $W_0, W_1, \cdots, W_{15}$.

FOR $j = 16$ TO 67

b. $W_j \leftarrow P_1(W_{j-16} \oplus W_{j-9} \oplus (W_{j-3} <<< 15)) \oplus (W_{j-13} <<< 7) \oplus W_{j-6}$

ENDFOR

FOR $j = 0$ TO 63

c. $W_j' = W_j \oplus W_{j+4}$

ENDFOR

5.3.3 Compression Function

Let A, B, C, D, E, F, G, H be eight word registers , $SS1, SS2, TT1, TT2$ be four intermediate variables , and the compression function $V^{(i+1)} = CF(V^{(i)}, B^{(i)})$ $(0 \le i \le n-1)$

The computation procedure is described as following:

$ABCDEFGH \leftarrow V(\mathrm{i})$

FOR $j = 0$ TO 63

$SS1 \leftarrow \left((\mathrm{A} <<< 12) + E + \left(T_j <<< (j \bmod 32) \right) \right) <<< 7$

$SS2 \leftarrow SS1 \oplus (A <<< 12)$

$TT1 \leftarrow FF_j(A,B,C) + D + SS2 + W_j^{'}$

$TT2 \leftarrow GG_j(E,F,G) + H + SS1 + W_j$

$D \leftarrow C$

$C \leftarrow B <<< 9$

$B \leftarrow A$

$A \leftarrow TT1$

$H \leftarrow G$

$G \leftarrow F <<< 19$

$F \leftarrow E$

$E \leftarrow P_0(TT2)$

ENDFOR

$V(i+1) \leftarrow ABCDEFGH \oplus V(i)$

Here, a word is stored in big-endian format.

5.4 Hash Value

$ABCDEFGH \leftarrow V^{(n)}$

Output a 256-bit hash value : $y = ABCDEFGH$

Annex A

(informative)

Examples

A.1 Example 1

The input message is "abc", and its ASCII-coded version is:

```
616263
```

The message after padding process is:

```
61626380 00000000 00000000 00000000 00000000 00000000 00000000 00000000
00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000018
```

The message after message expansion:

$W_0W_1...W_{67}$:

```
61626380 00000000 00000000 00000000 00000000 00000000 00000000 00000000
00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000018
9092e200 00000000 000c0606 719c70ed 00000000 8001801f 939f7da9 00000000
2c6fa1f9 adaaef14 00000000 0001801e 9a965f89 49710048 23ce86a1 b2d12f1b
e1dae338 f8061807 055d68be 86cfd481 1f447d83 d9023dbf 185898e0 e0061807
050df55c cde0104c a5b9c955 a7df0184 6e46cd08 e3babdf8 70caa422 0353af50
a92dbca1 5f33cfd2 e16f6e89 f70fe941 ca5462dc 85a90152 76af6296 c922bdb2
68378cf5 97585344 09008723 86faee74 2ab908b0 4a64bc50 864e6e08 f07e6590
325c8f78 accb8011 e11db9dd b99c0545
```

$W_0'W_1'\cdots W_{63}'$:

```
61626380 00000000 00000000 00000000 00000000 00000000 00000000 00000000
00000000 00000000 00000000 00000018 9092e200 00000000 000c0606 719c70f5
9092e200 8001801f 93937baf 719c70ed 2c6fa1f9 2dab6f0b 939f7da9 0001801e
b6f9fe70 e4dbef5c 23ce86a1 b2d0af05 7b4cbcb1 b177184f 2693ee1f 341efb9a
fe9e9ebb 210425b8 1d05f05e 66c9cc86 1a4988df 14e22df3 bde151b5 47d91983
6b4b3854 2e5aadb4 d5736d77 a48caed4 c76b71a9 bc89722a 91a5caab f45c4611
6379de7d da9ace80 97c00c1f 3e2d54f3 a263ee29 12f15216 7fafe5b5 4fd853c6
428e8445 dd3cef14 8f4ee92b 76848be4 18e587c8 e6af3c41 6753d7d5 49e260d5
```

The intermediate values during the iterative compression are:

```
 j A        B        C        D        E        F        G        H
   7380166f 4914b2b9 172442d7 da8a0600 a96f30bc 163138aa e38dee4d b0fb0e4e
 0 b9edc12b 7380166f 29657292 172442d7 b2ad29f4 a96f30bc c550b189 e38dee4d
```

```
 1 ea52428c b9edc12b 002cdee7 29657292 ac353a23 b2ad29f4 85e54b79 c550b189
 2 609f2850 ea52428c db825773 002cdee7 d33ad5fb ac353a23 4fa59569 85e54b79
 3 35037e59 609f2850 a48519d4 db825773 b8204b5f d33ad5fb d11d61a9 4fa59569
 4 1f995766 35037e59 3e50a0c1 a48519d4 8ad212ea b8204b5f afde99d6 d11d61a9
 5 374a0ca7 1f995766 06fcb26a 3e50a0c1 acf0f639 8ad212ea 5afdc102 afde99d6
 6 33130100 374a0ca7 32aecc3f 06fcb26a 3391ec8a acf0f639 97545690 5afdc102
 7 1022ac97 33130100 94194e6e 32aecc3f 367250a1 3391ec8a b1cd6787 97545690
 8 d47caf4c 1022ac97 26020066 94194e6e 6ad473a4 367250a1 64519c8f b1cd6787
 9 59c2744b d47caf4c 45592e20 26020066 c6a3ceae 6ad473a4 8509b392 64519c8f
10 481ba2a0 59c2744b f95e99a8 45592e20 02afb727 c6a3ceae 9d2356a3 8509b392
11 694a3d09 481ba2a0 84e896b3 f95e99a8 9dd1b58c 02afb727 7576351e 9d2356a3
12 89cbcd58 694a3d09 37454090 84e896b3 6370db62 9dd1b58c b938157d 7576351e
13 24c95abc 89cbcd58 947a12d2 37454090 1a4a2554 6370db62 ac64ee8d b938157d
14 7c529778 24c95abc 979ab113 947a12d2 3ee95933 1a4a2554 db131b86 ac64ee8d
15 34d1691e 7c529778 92b57849 979ab113 61f99646 3ee95933 2aa0d251 db131b86
16 796afab1 34d1691e a52ef0f8 92b57849 067550f5 61f99646 c999f74a 2aa0d251
17 7d27cc0e 796afab1 a2d23c69 a52ef0f8 b3c8669b 067550f5 b2330fcc c999f74a
18 d7820ad1 7d27cc0e d5f562f2 a2d23c69 575c37d8 b3c8669b 87a833aa b2330fcc
19 f84fd372 d7820ad1 4f981cfa d5f562f2 a5dceaf1 575c37d8 34dd9e43 87a833aa
20 02c57896 f84fd372 0415a3af 4f981cfa 74576681 a5dceaf1 bec2bae1 34dd9e43
21 4d0c2fcd 02c57896 9fa6e5f0 0415a3af 576f1d09 74576681 578d2ee7 bec2bae1
22 eeeec41a 4d0c2fcd 8af12c05 9fa6e5f0 b5523911 576f1d09 340ba2bb 578d2ee7
23 f368da78 eeeec41a 185f9a9a 8af12c05 6a879032 b5523911 e84abb78 340ba2bb
24 15ce1286 f368da78 dd8835dd 185f9a9a 62063354 6a879032 c88daa91 e84abb78
25 c3fd31c2 15ce1286 d1b4f1e6 dd8835dd 4db58f43 62063354 8193543c c88daa91
26 6243be5e c3fd31c2 9c250c2b d1b4f1e6 131152fe 4db58f43 9aa31031 8193543c
27 a549beaa 6243be5e fa638587 9c250c2b cf65e309 131152fe 7a1a6dac 9aa31031
28 e11eb847 a549beaa 877cbcc4 fa638587 e5b64e96 cf65e309 97f0988a 7a1a6dac
29 ff9bac9d e11eb847 937d554a 877cbcc4 9811b46d e5b64e96 184e7b2f 97f0988a
30 a5a4a2b3 ff9bac9d 3d708fc2 937d554a e92df4ea 9811b46d 74b72db2 184e7b2f
31 89a13e59 a5a4a2b3 37593bff 3d708fc2 0a1ff572 e92df4ea a36cc08d 74b72db2
32 3720bd4e 89a13e59 4945674b 37593bff cf7d1683 0a1ff572 a757496f a36cc08d
33 9ccd089c 3720bd4e 427cb313 4945674b da8c835f cf7d1683 ab9050ff a757496f
34 c7a0744d 9ccd089c 417a9c6e 427cb313 0958ff1b da8c835f b41e7be8 ab9050ff
35 d955c3ed c7a0744d 9a113939 417a9c6e c533f0ff 0958ff1b 1afed464 b41e7be8
36 e142d72b d955c3ed 40e89b8f 9a113939 d4509586 c533f0ff f8d84ac7 1afed464
37 e7250598 e142d72b ab87dbb2 40e89b8f c7f93fd3 d4509586 87fe299f f8d84ac7
38 2f13c4ad e7250598 85ae57c2 ab87dbb2 1a6cabc9 c7f93fd3 ac36a284 87fe299f
39 19f363f9 2f13c4ad 4a0b31ce 85ae57c2 c302badb 1a6cabc9 fe9e3fc9 ac36a284
40 55e1dde2 19f363f9 27895a5e 4a0b31ce 459daccf c302badb 5e48d365 fe9e3fc9
41 d4f4efe3 55e1dde2 e6c7f233 27895a5e 5cfba85a 459daccf d6de1815 5e48d365
42 48dcbc62 d4f4efe3 c3bbc4ab e6c7f233 6f49c7bb 5cfba85a 667a2ced d6de1815
43 8237b8a0 48dcbc62 e9dfc7a9 c3bbc4ab d89d2711 6f49c7bb 42d2e7dd 667a2ced
44 d8685939 8237b8a0 b978c491 e9dfc7a9 8ee87df5 d89d2711 3ddb7a4e 42d2e7dd
45 d2090a86 d8685939 6f714104 b978c491 2e533625 8ee87df5 388ec4e9 3ddb7a4e
46 e51076b3 d2090a86 d0b273b0 6f714104 d9f89e61 2e533625 efac7743 388ec4e9
47 47c5be50 e51076b3 12150da4 d0b273b0 3567734e d9f89e61 b1297299 efac7743
48 abddbdc8 47c5be50 20ed67ca 12150da4 3dfcdd11 3567734e f30ecfc4 b1297299
49 bd708003 abddbdc8 8b7ca08f 20ed67ca 93494bc0 3dfcdd11 9a71ab3b f30ecfc4
50 15e2f5d3 bd708003 bb7b9157 8b7ca08f c3956c3f 93494bc0 e889efe6 9a71ab3b
51 13826486 15e2f5d3 e100077a bb7b9157 cd09a51c c3956c3f 5e049a4a e889efe6
52 4a00ed2f 13826486 c5eba62b e100077a 0741f675 cd09a51c 61fe1cab 5e049a4a
```

```
53 f4412e82 4a00ed2f 04c90c27 c5eba62b 7429807c 0741f675 28e6684d 61fe1cab
54 549db4b7 f4412e82 01da5e94 04c90c27 f6bc15ed 7429807c b3a83a0f 28e6684d
55 22a79585 549db4b7 825d05e8 01da5e94 9d4db19a f6bc15ed 03e3a14c b3a83a0f
56 30245b78 22a79585 3b696ea9 825d05e8 f6804c82 9d4db19a af6fb5e0 03e3a14c
57 6598314f 30245b78 4f2b0a45 3b696ea9 f522adb2 f6804c82 8cd4ea6d af6fb5e0
58 c3d629a9 6598314f 48b6f060 4f2b0a45 14fb0764 f522adb2 6417b402 8cd4ea6d
59 ddb0a26a c3d629a9 30629ecb 48b6f060 589f7d5c 14fb0764 6d97a915 6417b402
60 71034d71 ddb0a26a ac535387 30629ecb 14d5c7f6 589f7d5c 3b20a7d8 6d97a915
61 5e636b4b 71034d71 6144d5bb ac535387 09ccd95e 14d5c7f6 eae2c4fb 3b20a7d8
62 2bfa5f60 5e636b4b 069ae2e2 6144d5bb 4ac3cf08 09ccd95e 3fb0a6ae eae2c4fb
63 1547e69b 2bfa5f60 c6d696bc 069ae2e2 e808f43b 4ac3cf08 caf04e66 3fb0a6ae
```

The hash value is:

```
66c7f0f4 62eeedd9 d1f2d46b dc10e4e2 4167c487 5cf2f7a2 297da02b 8f4ba8e0
```

A.2 Example 2

A message of 512 bits:

```
61626364 61626364 61626364 61626364 61626364 61626364 61626364 61626364
61626364 61626364 61626364 61626364 61626364 61626364 61626364 61626364
```

The message after padding process is:

```
61626364 61626364 61626364 61626364 61626364 61626364 61626364 61626364
61626364 61626364 61626364 61626364 61626364 61626364 61626364 61626364
80000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000
00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000200
```

The first message block:

The message after message expansion:

$W_0W_1...W_{67}$:

```
61626364 61626364 61626364 61626364 61626364 61626364 61626364 61626364
61626364 61626364 61626364 61626364 61626364 61626364 61626364 61626364
a121a024 a121a024 a121a024 6061e0e5 6061e0e5 6061e0e5 a002e345 a002e345
a002e345 49c969ed 49c969ed 49c969ed 85ae5679 a44ff619 a44ff619 694b6244
e8c8e0c4 e8c8e0c4 240e103e 346e603e 346e603e 9a517ab5 8a01aa25 8a01aa25
0607191c 25f8a37a d528936a 89fbd8ae 00606206 10501256 7cff7ef9 3c78b9f9
cc2b8a69 9f03f169 df45be20 9ec5bee1 0a212906 49ff72c0 46717241 67e09a19
6efaa333 2ebae676 3475c386 201dcff6 2f18fccf 2c5f2b5c a80b9f38 bc139f34
c47f18a7 a25ce71d 42743705 51baf619
```

$W_0'W_1'\cdots W_{63}'$:

```
00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000
00000000 00000000 00000000 00000000 c043c340 c043c340 c043c340 01038381
c14040c1 c14040c1 01234361 c06303a0 c06303a0 29a88908 e9cb8aa8 e9cb8aa8
25acb53c ed869ff4 ed869ff4 20820ba9 6d66b6bd 4c8716dd 8041e627 5d25027a
dca680fa 72999a71 ae0fba1b be6fca1b 32697922 bfa9d9cf 5f29394f 03fa728b
06677b1a 35a8b12c a9d7ed93 b5836157 cc4be86f 8f53e33f a3bac0d9 a2bd0718
c60aa36f d6fc83a9 9934cc61 f92524f8 64db8a35 674594b6 7204b1c7 47fd55ef
41e25ffc 02e5cd2a 9c7e5cbe 9c0e50c2 eb67e468 8e03cc41 ea7fa83d eda9692d
```

The intermediate values during iterative compression are:

```
 j A        B        C        D        E        F        G        H
   7380166f 4914b2b9 172442d7 da8a0600 a96f30bc 163138aa e38dee4d b0fb0e4e
 0 588b5dab 7380166f 29657292 172442d7 b2e561d0 a96f30bc c550b189 e38dee4d
 1 b31cecd3 588b5dab 002cdee7 29657292 887cdf53 b2e561d0 85e54b79 c550b189
 2 087b31df b31cecd3 16bb56b1 002cdee7 5234344f 887cdf53 0e85972b 85e54b79
 3 17448b12 087b31df 39d9a766 16bb56b1 16372ca6 5234344f fa9c43e6 0e85972b
 4 dca06de5 17448b12 f663be10 39d9a766 f7bc113c 16372ca6 a27a91a1 fa9c43e6
 5 8eb847a3 dca06de5 8916242e f663be10 9fe64fb1 f7bc113c 6530b1b9 a27a91a1
 6 0e0f1218 8eb847a3 40dbcbb9 8916242e 57e5fc4e 9fe64fb1 89e7bde0 6530b1b9
 7 ada83827 0e0f1218 708f471d 40dbcbb9 55eb8591 57e5fc4e 7d8cff32 89e7bde0
 8 6e12c163 ada83827 1e24301c 708f471d c26a14b8 55eb8591 e272bf2f 7d8cff32
 9 f7578117 6e12c163 50704f5b 1e24301c 3433dd28 c26a14b8 2c8aaf5c e272bf2f
10 bc497c66 f7578117 2582c6dc 50704f5b 4f85c749 3433dd28 a5c61350 2c8aaf5c
11 ecc59168 bc497c66 af022fee 2582c6dc 8ce5ee61 4f85c749 e941a19e a5c61350
12 63723715 ecc59168 92f8cd78 af022fee 38e2aa27 8ce5ee61 3a4a7c2e e941a19e
13 e57bfbf8 63723715 8b22d1d9 92f8cd78 542318e7 38e2aa27 730c672f 3a4a7c2e
14 8ba504b1 e57bfbf8 e46e2ac6 8b22d1d9 a8c73777 542318e7 5139c715 730c672f
15 b6a4be20 8ba504b1 f7f7f1ca e46e2ac6 8ae4d7a0 a8c73777 c73aa118 5139c715
16 c0a0e3f7 b6a4be20 4a096317 f7f7f1ca f671e12a 8ae4d7a0 bbbd4639 c73aa118
17 68ef7357 c0a0e3f7 497c416d 4a096317 673f9d46 f671e12a bd045726 bbbd4639
18 4c6499d3 68ef7357 41c7ef81 497c416d f01924a3 673f9d46 0957b38f bd045726
19 9f532735 4c6499d3 dee6aed1 41c7ef81 71c6ef02 f01924a3 ea3339fc 0957b38f
20 231d84bd 9f532735 c933a698 dee6aed1 108149de 71c6ef02 251f80c9 ea3339fc
21 6a203212 231d84bd a64e6b3e c933a698 90c31af9 108149de 78138e37 251f80c9
22 175c3b57 6a203212 3b097a46 a64e6b3e 508f82d2 90c31af9 4ef0840a 78138e37
23 cdcbabd5 175c3b57 406424d4 3b097a46 b5a2f2fb 508f82d2 d7cc8618 4ef0840a
24 7dd941f8 cdcbabd5 b876ae2e 406424d4 a541cb9b b5a2f2fb 1692847c d7cc8618
25 eaf54f3e 7dd941f8 9757ab9b b876ae2e 912d4e17 a541cb9b 97ddad17 1692847c
26 f7310a83 eaf54f3e b283f0fb 9757ab9b b43da5e9 912d4e17 5cdd2a0e 97ddad17
27 f8441d7e f7310a83 ea9e7dd5 b283f0fb cf194872 b43da5e9 70bc896a 5cdd2a0e
28 270dce67 f8441d7e 621507ee ea9e7dd5 7564b6c0 cf194872 2f4da1ed 70bc896a
29 ac12a6c0 270dce67 883afdf0 621507ee 964015e3 7564b6c0 439678ca 2f4da1ed
30 1bd9e6e3 ac12a6c0 1b9cce4e 883afdf0 0fac4cad 964015e3 b603ab25 439678ca
31 32418d74 1bd9e6e3 254d8158 1b9cce4e 3f717698 0fac4cad af1cb200 b603ab25
32 9c89b505 32418d74 b3cdc637 254d8158 38766abf 3f717698 65687d62 af1cb200
33 3c60352a 9c89b505 831ae864 b3cdc637 8aedd93b 38766abf b4c1fb8b 65687d62
34 2a116c70 3c60352a 136a0b39 831ae864 476048d4 8aedd93b 55f9c3b3 b4c1fb8b
35 a0c7c66f 2a116c70 c06a5478 136a0b39 b47a7dc5 476048d4 c9dc576e 55f9c3b3
36 b7e58f33 a0c7c66f 22d8e054 c06a5478 3a3537a9 b47a7dc5 46a23b02 c9dc576e
37 79baf4ca b7e58f33 8f8cdf41 22d8e054 9455b731 3a3537a9 ee2da3d3 46a23b02
```

```
38 ad5b0bcf 79baf4ca cb1e676f 8f8cdf41 289d35e0 9455b731 bd49d1a9 ee2da3d3
39 a167bd76 ad5b0bcf 75e994f3 cb1e676f da27276b 289d35e0 b98ca2ad bd49d1a9
40 2ccc1878 a167bd76 b6179f5a 75e994f3 7eded43b da27276b af0144e9 b98ca2ad
41 610c6084 2ccc1878 cf7aed42 b6179f5a 9da32cab 7eded43b 3b5ed139 af0144e9
42 a40209fe 610c6084 9830f059 cf7aed42 7d483846 9da32cab a1dbf6f6 3b5ed139
43 6fa376a2 a40209fe 18c108c2 9830f059 12a851cf 7d483846 655ced19 a1dbf6f6
44 53f9ffc5 6fa376a2 0413fd48 18c108c2 c3d3327b 12a851cf c233ea41 655ced19
45 4f60bbd5 53f9ffc5 46ed44df 0413fd48 f3cae7e6 c3d3327b 8e789542 c233ea41
46 6e89a7fb 4f60bbd5 f3ff8aa7 46ed44df 17394ca0 f3cae7e6 93de1e99 8e789542
47 fef3cb16 6e89a7fb c177aa9e f3ff8aa7 4a9e594f 17394ca0 3f379e57 93de1e99
48 fa8e6731 fef3cb16 134ff6dd c177aa9e 7d9e1966 4a9e594f 6500b9ca 3f379e57
49 08a826c3 fa8e6731 e7962dfd 134ff6dd ebfa90cc 7d9e1966 ca7a54f2 6500b9ca
50 614c7627 08a826c3 1cce63f5 e7962dfd 969ecf53 ebfa90cc cb33ecf0 ca7a54f2
51 d776618d 614c7627 504d8611 1cce63f5 423489f6 969ecf53 86675fd4 cb33ecf0
52 ef958266 d776618d 98ec4ec2 504d8611 6ef4554d 423489f6 7a9cb4f6 86675fd4
53 04b44fd2 ef958266 ecc31bae 98ec4ec2 290032b5 6ef4554d 4fb211a4 7a9cb4f6
54 008d6012 04b44fd2 2b04cddf ecc31bae 50aa1faa 290032b5 aa6b77a2 4fb211a4
55 57859fec 008d6012 689fa409 2b04cddf c00cd655 50aa1faa 95a94801 aa6b77a2
56 c864420d 57859fec 1ac02401 689fa409 2fb3c502 c00cd655 fd528550 95a94801
57 e7423482 c864420d 0b3fd8af 1ac02401 aac3b183 2fb3c502 b2ae0066 fd528550
58 5c5be9dd e7423482 c8841b90 0b3fd8af 8b1ba117 aac3b183 28117d9e b2ae0066
59 ebd4948c 5c5be9dd 846905ce c8841b90 74a75fe1 8b1ba117 8c1d561d 28117d9e
60 05627b53 ebd4948c b7d3bab8 846905ce f58d98d8 74a75fe1 08bc58dd 8c1d561d
61 28aaec87 05627b53 a92919d7 b7d3bab8 cc6b5f2a f58d98d8 ff0ba53a 08bc58dd
62 0f92d652 28aaec87 c4f6a60a a92919d7 b8ab6d40 cc6b5f2a c6c7ac6c ff0ba53a
63 2ad0c8ee 0f92d652 55d90e51 c4f6a60a 69caa1b7 b8ab6d40 f956635a c6c7ac6c
```

The second message block:

The message after message expansion:

$W_0W_1...W_{67}$:

```
80000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000
00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000200
80404000 00000000 01008080 10005000 00000000 002002a0 ac545c04 00000000
09582a39 a0003000 00000000 00200280 a4515804 20200040 51609838 30005701
a0002000 008200aa 6ad525d0 0a0e0216 b0f52042 fa7073b0 20000000 008200a8
7a542590 22a20044 d5d6ebd2 82005771 8a202240 b42826aa eaf84e59 4898eaf9
8207283d ee6775fa a3e0e0a0 8828488a 23b45a5d 628a22c4 8d6d0615 38300a7e
e96260e5 2b60c020 502ed531 9e878cb9 218c38f8 dcae3cb7 2a3e0e0a e9e0c461
8c3e3831 44aaa228 dc60a38b 518300f7
```

$W_0'W_1'\cdots W_{63}'$:

```
80000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000
00000000 00000000 00000000 00000200 80404000 00000000 01008080 10005200
80404000 002002a0 ad54dc84 10005000 09582a39 a02032a0 ac545c04 00200280
ad09723d 80203040 51609838 30205581 04517804 20a200ea 3bb5bde8 3a0e5517
10f50042 faf2731a 4ad525d0 0a8c02be caa105d2 d8d273f4 f5d6ebd2 828257d9
```

```
f07407d0 968a26ee 3f2ea58b ca98bd88 08270a7d 5a4f5350 4918aef9 c0b0a273
a1b37260 8ced573e 2e8de6b5 b01842f4 cad63ab8 49eae2e4 dd43d324 a6b786c7
c8ee581d f7cefc97 7a10db3b 776748d8 adb200c9 98049e9f f65ead81 b863c496
```

The intermediate values during iterative compression are:

```
 j A        B        C        D        E        F        G        H
   5950de81 468664eb 42fd4c86 1e7ca00a c0a5910b ae9a55ea 1adb8d17 763ca222
 0 1cc66027 5950de81 0cc9d68d 42fd4c86 24fe81a1 c0a5910b af5574d2 1adb8d17
 1 b7197324 1cc66027 a1bd02b2 0cc9d68d 61b7397a 24fe81a1 885e052c af5574d2
 2 b1aacb3f b7197324 8cc04e39 a1bd02b2 4c7cbb59 61b7397a 0d0927f4 885e052c
 3 920d5d4d b1aacb3f 32e6496e 8cc04e39 c6c863a3 4c7cbb59 cbd30db9 0d0927f4
 4 03162191 920d5d4d 55967f63 32e6496e dbcb73dd c6c863a3 daca63e5 cbd30db9
 5 cbfddbb7 03162191 1aba9b24 55967f63 6a6eaafb dbcb73dd 1d1e3643 daca63e5
 6 67f45147 cbfddbb7 2c432206 1aba9b24 e0cc5b97 6a6eaafb 9eeede5b 1d1e3643
 7 dfc06393 67f45147 fbb76f97 2c432206 9d84a8d5 e0cc5b97 57db5375 9eeede5b
 8 777f980d dfc06393 e8a28ecf fbb76f97 89d0a059 9d84a8d5 dcbf0662 57db5375
 9 502a9be2 777f980d 80c727bf e8a28ecf befc3eda 89d0a059 46acec25 dcbf0662
10 df0f77ed 502a9be2 ff301aee 80c727bf c8b999f7 befc3eda 02cc4e85 46acec25
11 b8bc2801 df0f77ed 5537c4a0 ff301aee 3a05da38 c8b999f7 f6d5f7e1 02cc4e85
12 5b3baaa5 b8bc2801 1eefdbbe 5537c4a0 eebf718f 3a05da38 cfbe45cc f6d5f7e1
13 0f7185e4 5b3baaa5 78500371 1eefdbbe f3fbf969 eebf718f d1c1d02e cfbe45cc
14 141cb1e7 0f7185e4 77554ab6 78500371 5cc495db f3fbf969 8c7f75fb d1c1d02e
15 f185448a 141cb1e7 e30bc81e 77554ab6 32028d02 5cc495db cb4f9fdf 8c7f75fb
16 a7374acd f185448a 3963ce28 e30bc81e 3d03e81b 32028d02 aedae624 cb4f9fdf
17 aaca2dcb a7374acd 0a8915e3 3963ce28 130bc932 3d03e81b 68119014 aedae624
18 3d2dfd31 aaca2dcb 6e959b4e 0a8915e3 07fff8f8 130bc932 40d9e81f 68119014
19 15bab3e6 3d2dfd31 945b9755 6e959b4e 85b2dd34 07fff8f8 4990985e 40d9e81f
20 f477625b 15bab3e6 5bfa627a 945b9755 d2b3c82b 85b2dd34 c7c03fff 4990985e
21 ecbfba29 f477625b 7567cc2b 5bfa627a 604bda38 d2b3c82b e9a42d96 c7c03fff
22 b9f6943d ecbfba29 eec4b7e8 7567cc2b e996d68b 604bda38 415e959e e9a42d96
23 c537ac67 b9f6943d 7f7453d9 eec4b7e8 7f6c2bc6 e996d68b d1c3025e 415e959e
24 c59665b3 c537ac67 ed287b73 7f7453d9 1a89ef0d 7f6c2bc6 b45f4cb6 d1c3025e
25 50115e1f c59665b3 6f58cf8a ed287b73 3ddf2899 1a89ef0d 5e33fb61 b45f4cb6
26 44196085 50115e1f 2ccb678b 6f58cf8a 0abc22da 3ddf2899 7868d44f 5e33fb61
27 bde4e355 44196085 22bc3ea0 2ccb678b da96412a 0abc22da 44c9eef9 7868d44f
28 ca176dca bde4e355 32c10a88 22bc3ea0 b418ac1b da96412a 16d055e1 44c9eef9
29 541e456e ca176dca c9c6ab7b 32c10a88 35cf8215 b418ac1b 0956d4b2 16d055e1
30 b6feeef7 541e456e 2edb9594 c9c6ab7b d41f5fda 35cf8215 60dda0c5 0956d4b2
31 026e42f7 b6feeef7 3c8adca8 2edb9594 c9436b11 d41f5fda 10a9ae7c 60dda0c5
32 8fd27582 026e42f7 fdddef6d 3c8adca8 a48dc4c2 c9436b11 fed6a0fa 10a9ae7c
33 2527f8c6 8fd27582 dc85ee04 fdddef6d b29dc9d4 a48dc4c2 588e4a1b fed6a0fa
34 3218579f 2527f8c6 a4eb051f dc85ee04 0da81ad7 b29dc9d4 2615246e 588e4a1b
35 35421cf3 3218579f 4ff18c4a a4eb051f 644b37e4 0da81ad7 4ea594ee 2615246e
36 12cb048f 35421cf3 30af3e64 4ff18c4a 107cb2fb 644b37e4 d6b86d40 4ea594ee
37 c6716749 12cb048f 8439e66a 30af3e64 7903974d 107cb2fb bf232259 d6b86d40
38 66bf4600 c6716749 96091e25 8439e66a e5575380 7903974d 97d883e5 bf232259
39 046516a9 66bf4600 e2ce938c 96091e25 e23d4f18 e5575380 ba6bc81c 97d883e5
40 e14ab898 046516a9 7e8c00cd e2ce938c 6e25affe e23d4f18 9c072aba ba6bc81c
41 bc44d883 e14ab898 ca2d5208 7e8c00cd 4ef0cb38 6e25affe 78c711ea 9c072aba
42 e017c779 bc44d883 957131c2 ca2d5208 10132c10 4ef0cb38 7ff3712d 78c711ea
```

```
43 11154e38 e017c779 89b10778 957131c2 c1d401bd 10132c10 59c27786 7ff3712d
44 3ba43e10 11154e38 2f8ef3c0 89b10778 953c1e65 c1d401bd 60808099 59c27786
45 445e8d34 3ba43e10 2a9c7022 2f8ef3c0 94bcdd11 953c1e65 0dee0ea0 60808099
46 34d09ee0 445e8d34 487c2077 2a9c7022 1d0ea72c 94bcdd11 f32ca9e0 0dee0ea0
47 18c77c40 34d09ee0 bd1a6888 487c2077 a8ca98c6 1d0ea72c e88ca5e6 f32ca9e0
48 a2507cea 18c77c40 a13dc069 bd1a6888 9845362a a8ca98c6 3960e875 e88ca5e6
49 7e014176 a2507cea 8ef88031 a13dc069 2cb0c2f2 9845362a c6354654 3960e875
50 eb39074b 7e014176 a0f9d544 8ef88031 0df22b74 2cb0c2f2 b154c229 c6354654
51 f67597e1 eb39074b 0282ecfc a0f9d544 8d4f6b2f 0df22b74 17916586 b154c229
52 31e9309d f67597e1 720e97d6 0282ecfc eecf99be 8d4f6b2f 5ba06f91 17916586
53 c6329c3c 31e9309d eb2fc3ec 720e97d6 c672ad96 eecf99be 597c6a7b 5ba06f91
54 75cc3800 c6329c3c d2613a63 eb2fc3ec 8515c87f c672ad96 cdf7767c 597c6a7b
55 925156ad 75cc3800 6538798c d2613a63 150cbd57 8515c87f 6cb63395 cdf7767c
56 7d0de10b 925156ad 987000eb 6538798c 7ee47610 150cbd57 43fc28ae 6cb63395
57 2066f136 7d0de10b a2ad5b24 987000eb 7d7aadcc 7ee47610 eab8a865 43fc28ae
58 85b31359 2066f136 1bc216fa a2ad5b24 07b9cfd1 7d7aadcc b083f723 eab8a865
59 6cddcb93 85b31359 cde26c40 1bc216fa c43eb29c 07b9cfd1 6e63ebd5 b083f723
60 23eff97d 6cddcb93 6626b30b cde26c40 1ea21d46 c43eb29c 7e883dce 6e63ebd5
61 07bd4e82 23eff97d bb9726d9 6626b30b c8d6867c 1ea21d46 94e621f5 7e883dce
62 64f3dc4a 07bd4e82 dff2fa47 bb9726d9 96e4028f c8d6867c ea30f510 94e621f5
63 87ee4178 64f3dc4a 7a9d040f dff2fa47 af7ee1ee 96e4028f 33e646b4 ea30f510
```

The hash value is:

```
debe9ff9 2275b8a1 38604889 c18e5a4d 6fdb70e5 387e5765 293dcba3 9c0c5732
```