On Blue Midnight Wish Decomposition

Vlastimil Klima

v.klima@volny.cz

Independent Cryptologist - Consultant
Prague, Czech Republic

Danilo Gligoroski

danilog@item.ntnu.no

Department of Telematics
Norwegian University of Science and Technology
Trondheim, Norway

Abstract

Blue Midnight Wish is one of the 14 candidates in the second round of the NIST SHA-3 competition [1]. In this paper we present a decomposition of the Blue Midnight Wish core functions, what gives deeper look at the Blue Midnight Wish family of hash functions and a tool for their cryptanalysis. We used this decomposition for better understanding the insights of Blue Midnight Wish functions and to propose the tweak for the second round. We would like to encourage further cryptanalysis of Blue Midnight Wish, as the quickest candidate in the second round.

Keywords: hash, SHA-3, Blue Midnight Wish.

1 Introduction

In this paper we describe a decomposition of the functions used in tweaked version of Blue Midnight Wish, which was sent to the second round of the NIST SHA-3 competition [2]. It is better for the reader to start reading the basic description in [2], because this paper in some sense extends it. Thus, here we just repeat the necessary notations from the basic description and add some new notations of the new variables of decomposed blocks. Then we describe complete decomposition of Blue Midnight Wish into simple blocks. The notation and decomposition is written generally for all variants of Blue Midnight Wish, because BMW224/384 are based on BMW256/512 and BMW256/512 differ only in the length of the word $w = 32/64$, in some constants and shifts in the definitions of logical functions. Therefore for the simplicity it suffices to talk simply of Blue Midnight Wish. When necessary, we will use concrete values of constants and logical functions for BMW256. Otherwise the description and the decomposition holds for all Blue Midnight Wish variants. The aim is to give the symbolic and simple description, which would unveil and show the basic relations and thus help cryptanalysis. Moreover we note several basic properties of used transformations.

2 Notations

The basic units in the description are mainly words of w bits. Note that for BMW256/512 we have $w = 32/64$. The message block and the chaining values and other variables are usually vectors of 16 words. We will use capitals for these vectors and indexes for their words, for instance $X = (X_0, \ldots, X_{15})$.

Similarly for variables A, S, W, G, H, K, D. The small exception is notation $Q_a = (Q_0, \ldots, Q_{15})$ and $Q_b = (Q_{16}, \ldots, Q_{31})$ for the first and second part of the quadruple pipe Q.

We denote by $ROTL^1(H)$ the rotation on words of the variable $H = (H_0, \ldots, H_{15})$ such that the words as a whole are rotated one position to the left. Concretely, $ROTL^1(H) = (H_1, H_2, \ldots, H_{15}, H_0)$. Analogously we define $ROTL^7(H) = (H_7, H_8, \ldots, H_{15}, H_0, H_1, \ldots, H_6)$.

We also use $rotM$ to express another kind of rotation. It is defined as an ensemble of rotations of separate words of M such that the i-th word of M (as a w-bit string) is rotated by $i+1$ positions to the left. For instance the word M_0 is rotated left by 1 position, the word M_1 is rotated left by 2 positions, and so forth up to the the word M_{15} which is rotated to the left by 16 positions.

The width of the message block is the same as the length of chaining value and it is $m = 16w$ bits.

3 Blue Midnight Wish description and decomposition

3.1 General design principles

Blue Midnight Wish is an iterative hash function based on a compression function. Padding and preprocessing of the message is similar to that of SHA-1/2. The first noticeable difference from SHA-1/2 is in the width of the chaining value, which is twice as long as the final hash value. The size of the message block is same as the size of the chaining value (double length). We can say that this is one of the design principles that contribute to the Blue Midnight Wish speed. The second difference from SHA-1/2 consists of an additional processing of the last chaining value once more by the compression function. Finally, the output of the final invocation of the compression function is truncated to give the hash value.

The generic description of the Blue Midnight Wish hash algorithm is given in Table 1.

Algorithm: Blue Midnight Wish

Input: Message M of length l bits, and the message digest size n.
Output: A message digest $Hash$, that is n bits long.

1. Preprocessing
   (a) Pad the message M.
   (b) Parse the padded message into N, m-bit message blocks, $M^{(1)}$, $M^{(2)}$, ..., $M^{(N)}$.
   (c) Set the initial value of the double pipe $H^{(0)}$.
2. Hash computation

   For $i = 1$ to N
   {
   $Q_a^{(i)} = f_0(M^{(i)}, H^{(i-1)})$;
   $Q_b^{(i)} = f_1(M^{(i)}, H^{(i-1)}, Q_a^{(i)})$;
   $H^{(i)} = f_2(M^{(i)}, Q_a^{(i)}, Q_b^{(i)})$;
   }
3. Finalization

   $Q_a^{final} = f_0(H^{(N)}, CONST^{final})$;
   $Q_b^{final} = f_1(H^{(N)}, CONST^{final}, Q_a^{final})$;
   $H^{final} = f_2(H^{(N)}, Q_a^{final}, Q_b^{final})$;
4. $Hash$ =Take_n_Least_Significant_Bits(H^{final}).

Table 1: A generic description of the Blue Midnight Wish hash algorithm

A graphic representation of the Blue Midnight Wish hash algorithm and its complete decomposition is given in the Figure 1.

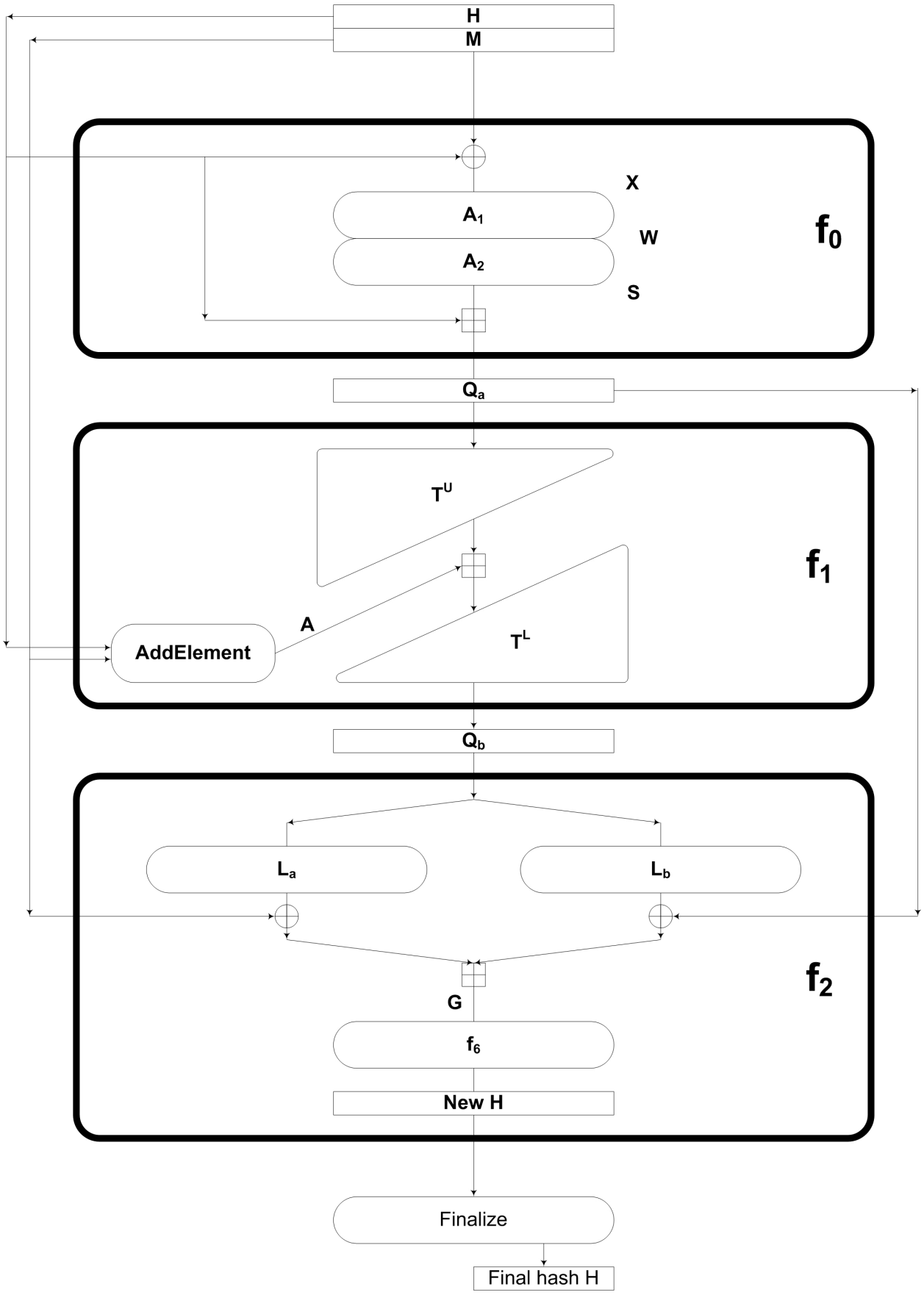

Figure 1: A graphic representation of the fully decomposed Blue Midnight Wish hash algorithm.

3.2 The basic structure

The basic structure consists of three functions f_0, f_1 and f_2. If we denote by f the compression function, then we can symbolically represent f as $f = f_2 \circ f_1 \circ f_0$. The compression function have two inputs, the old chaining value H (which is set to a constant IV in the beginning) and a message block M. Both enter the function f_0, which creates the value Q_a. Then H, Q_a and M enter the function f_1 which produces Q_b. At the end, Q_b, Q_a and M enter the function f_2, which creates the new chaining value H. For the security of the whole hash function it is very important that the values M, Q_a, Q_b, and H are twice as long as the final hash value. It makes creating multi-collisions from collisions of the chaining values ineffective. Definitions of the functions f_0, f_1, f_2 and the logic functions are showed in the appendix.

3.3 Decomposition of the function f_0

We can write symbolically $f_0 = A_3 \circ A_2 \circ A_1 \circ A_0$ as a composition of alternating binary ($\oplus$) and arithmetical (additions/subtractions $\texttt{mod}\ 2^w$) transformations (bijections or multipermutations).

$Q_a = f_0(M, H) = A_3(A_2(A_1(A_0(M, H))), H)$, $X = A_0(M, H) = M \oplus H$, where A_0 only xors appropriate (particular) input words together, $W = A_1(X)$, where A_1 creates output words by applying additions and subtractions on 5 summands modulo 2^w, $S = A_2(W)$, where A_2 uses binary operations xors, shifts and rotations on input words, $Q_a = A_3(S, H) = S + ROTL^1(H)$, where A_3 adds S and rotated vector H component wise, also modulo 2^w.

The useful expression is $Q_a = A_2(A_1(M \oplus H)) + ROTL^1(H)$, where

$W = A_1(X)$:

$$
\begin{array}{rcrcrcrcrcr}
W_0 &=& X_5 &-& X_7 &+& X_{10} &+& X_{13} &+& X_{14} \\
W_1 &=& X_6 &-& X_8 &+& X_{11} &+& X_{14} &-& X_{15} \\
W_2 &=& X_0 &+& X_7 &+& X_9 &-& X_{12} &+& X_{15} \\
W_3 &=& X_0 &-& X_1 &+& X_8 &-& X_{10} &+& X_{13} \\
W_4 &=& X_1 &+& X_2 &+& X_9 &-& X_{11} &-& X_{14} \\
W_5 &=& X_3 &-& X_2 &+& X_{10} &-& X_{12} &+& X_{15} \\
W_6 &=& X_4 &-& X_0 &-& X_3 &-& X_{11} &+& X_{13} \\
W_7 &=& X_1 &-& X_4 &-& X_5 &-& X_{12} &-& X_{14} \\
W_8 &=& X_2 &-& X_5 &-& X_6 &+& X_{13} &-& X_{15} \\
W_9 &=& X_0 &-& X_3 &+& X_6 &-& X_7 &+& X_{14} \\
W_{10} &=& X_8 &-& X_1 &-& X_4 &-& X_7 &+& X_{15} \\
W_{11} &=& X_8 &-& X_0 &-& X_2 &-& X_5 &+& X_9 \\
W_{12} &=& X_1 &+& X_3 &-& X_6 &-& X_9 &+& X_{10} \\
W_{13} &=& X_2 &+& X_4 &+& X_7 &+& X_{10} &+& X_{11} \\
W_{14} &=& X_3 &-& X_5 &+& X_8 &-& X_{11} &-& X_{12} \\
W_{15} &=& X_{12} &-& X_4 &-& X_6 &-& X_9 &+& X_{13}
\end{array}
$$

$S = A_2(W)$:

$$
\begin{array}{cccc}
S_0 = s_0(W_0) & S_1 = s_1(W_1) & S_2 = s_2(W_2) & S_3 = s_3(W_3) \\
S_4 = s_4(W_4) & S_5 = s_0(W_5) & S_6 = s_1(W_6) & S_7 = s_2(W_7) \\
S_8 = s_3(W_8) & S_9 = s_4(W_9) & S_{10} = s_0(W_{10}) & S_{11} = s_1(W_{11}) \\
S_{12} = s_2(W_{12}) & S_{13} = s_3(W_{13}) & S_{14} = s_4(W_{14}) & S_{15} = s_0(W_{15})
\end{array}
$$

$Q_a = A_3(S, H)$:

$$
\begin{array}{cccc}
Q_0 = S_0 + H_1; & Q_1 = S_1 + H_2; & Q_2 = S_2 + H_3; & Q_3 = S_3 + H_4; \\
Q_4 = S_4 + H_5; & Q_5 = S_5 + H_6; & Q_6 = S_6 + H_7; & Q_7 = S_7 + H_8; \\
Q_8 = S_8 + H_9; & Q_9 = S_9 + H_{10}; & Q_{10} = S_{10} + H_{11}; & Q_{11} = S_{11} + H_{12}; \\
Q_{12} = S_{12} + H_{13}; & Q_{13} = S_{13} + H_{14}; & Q_{14} = S_{14} + H_{15}; & Q_{15} = S_{15} + H_0;
\end{array}
$$

3.4 Decomposition of the function f_1

The function f_1 can be viewed as a (weak) block cipher with the default value of $NR = 16$ rounds. NR is divided into two parts, $NR = ExpandRounds_1 + ExpandRounds_2$. At first the transformation $expand_1()$ is used with number of rounds $ExpandRounds_1$. Then the the second transformation $expand_2()$ follows with the remaining number of rounds $ExpandRounds_2 = NR - ExpandRounds_1$. Note that the first transformation is more complex and is assumed as tunable parameter. Increasing it (at the expense of the second one) it is possible to tune the complexity of Blue Midnight Wish.

If we denote by E the underlying block cipher, then we have the ciphertext $Q_b = f_1(M, Q_a) = E_A(Q_a)$, where the key A is created by the transformation $AddElement()$ and Q_a is the plaintext.

Here we denote the key as A, because the letter K is used to denote the constants used in $AddElement()$ transformation.

Now we have this simple decomposition:

$A = AddElement(M, H) = (B(rotM) + K) \oplus ROTL^7(H)$, where K is a constant $K = (16 * 0x05555555, ..., 31 * 0x05555555)$.

The transformation B is similar to A_1 and it creates output words adding and subtracting 3 summands modulo 2^w. It is given by the nonsingular matrix over Z_{2^w}. If we denote by $D = B(M)$, we have

$$\begin{array}{lcccccc}
D_0 & = & M_0 & + & M_3 & - & M_{10} \\
D_1 & = & M_1 & + & M_4 & - & M_{11} \\
D_2 & = & M_2 & + & M_5 & - & M_{12} \\
D_3 & = & M_3 & + & M_6 & - & M_{13} \\
D_4 & = & M_4 & + & M_7 & - & M_{14} \\
D_5 & = & M_5 & + & M_8 & - & M_{15} \\
D_6 & = & M_6 & + & M_9 & - & M_0 \\
D_7 & = & M_7 & + & M_{10} & - & M_1 \\
D_8 & = & M_8 & + & M_{11} & - & M_2 \\
D_9 & = & M_9 & + & M_{12} & - & M_3 \\
D_{10} & = & M_{10} & + & M_{13} & - & M_4 \\
D_{11} & = & M_{11} & + & M_{14} & - & M_5 \\
D_{12} & = & M_{12} & + & M_{15} & - & M_6 \\
D_{13} & = & M_{13} & + & M_0 & - & M_7 \\
D_{14} & = & M_{14} & + & M_1 & - & M_8 \\
D_{15} & = & M_{15} & + & M_2 & - & M_9
\end{array}$$

The function f_1 expands $Q_a = (Q_0, ..., Q_{15})$ to $Q_b = (Q_{16}, ..., Q_{31})$ according to the tunable parameters $ExpandRounds_1$ and $ExpandRounds_2$:

1.1 For $ii = 0$ to $ExpandRounds_1 - 1$
$Q^{(i)}_{ii+16} = expand_1(ii + 16)$

1.2 For $ii = ExpandRounds_1$ to $ExpandRounds_1 + ExpandRounds_2 - 1$
$Q^{(i)}_{ii+16} = expand_2(ii + 16)$

where the functions $expand_1()$ and $expand_2()$ are defined as:

$$\begin{aligned}
expand_1(j) = {} & s_1(Q_{j-16}) + s_2(Q_{j-15}) + s_3(Q_{j-14}) + s_0(Q_{j-13}) \\
& + s_1(Q_{j-12}) + s_2(Q_{j-11}) + s_3(Q_{j-10}) + s_0(Q_{j-9}) \\
& + s_1(Q_{j-8}) + s_2(Q_{j-7}) + s_3(Q_{j-6}) + s_0(Q_{j-5}) \\
& + s_1(Q_{j-4}) + s_2(Q_{j-3}) + s_3(Q_{j-2}) + s_0(Q_{j-1}) \\
& + A_{j-16}
\end{aligned}$$

$$\begin{aligned}
expand_2(j) = {} & Q_{j-16} + r_1(Q_{j-15}) + Q_{j-14} + r_2(Q_{j-13}) \\
& + Q_{j-12} + r_3(Q_{j-11}) + Q_{j-10} + r_4(Q_{j-9}) \\
& + Q_{j-8} + r_5(Q_{j-7}) + Q_{j-6} + r_6(Q_{j-5}) \\
& + Q_{j-4} + r_7(Q_{j-3}) + s_4(Q_{j-2}) + s_5(Q_{j-1}) \\
& + A_{j-16}.
\end{aligned}$$

In order to get an insight of the transformations that compute Q_b we will introduce two triangular (upper and lower) transformations. Those transformations will be separated by a part that can be seen as a addition of the key A.

Let us first define an upper triangular transformation of $P = T^U(Q_a)$:

$$\begin{aligned}
P_0 &= s1(Q_0) + s2(Q_1) + s3(Q_2) + s0(Q_3) + s1(Q_4) + s2(Q_5) + s3(Q_6) + s0(Q_7) + s1(Q_8) + s2(Q_9) + s3(Q_{10}) + s0(Q_{11}) + s1(Q_{12}) + \\
&\quad + s2(Q_{13}) + s3(Q_{14}) + s0(Q_{15}) \\
P_1 &= s1(Q_1) + s2(Q_2) + s3(Q_3) + s0(Q_4) + s1(Q_5) + s2(Q_6) + s3(Q_7) + s0(Q_8) + s1(Q_9) + s2(Q_{10}) + s3(Q_{11}) + s0(Q_{12}) + \\
&\quad + s1(Q_{13}) + s2(Q_{14}) + s3(Q_{15}) \\
P_2 &= Q_2 + r1(Q_3) + Q_4 + r2(Q_5) + Q_6 + r3(Q_7) + Q_8 + r4(Q_9) + Q_{10} + r5(Q_{11}) + Q_{12} + r6(Q_{13}) + Q_{14} + r7(Q_{15}) \\
P_3 &= Q_3 + r1(Q_4) + Q_5 + r2(Q_6) + Q_7 + r3(Q_8) + Q_9 + r4(Q_{10}) + Q_{11} + r5(Q_{12}) + Q_{13} + r6(Q_{14}) + Q_{15} \\
P_4 &= Q_4 + r1(Q_5) + Q_6 + r2(Q_7) + Q_8 + r3(Q_9) + Q_{10} + r4(Q_{11}) + Q_{12} + r5(Q_{13}) + Q_{14} + r6(Q_{15}) \\
P_5 &= Q_5 + r1(Q_6) + Q_7 + r2(Q_8) + Q_9 + r3(Q_{10}) + Q_{11} + r4(Q_{12}) + Q_{13} + r5(Q_{14}) + Q_{15} \\
P_6 &= Q_6 + r1(Q_7) + Q_8 + r2(Q_9) + Q_{10} + r3(Q_{11}) + Q_{12} + r4(Q_{13}) + Q_{14} + r5(Q_{15}) \\
P_7 &= Q_7 + r1(Q_8) + Q_9 + r2(Q_{10}) + Q_{11} + r3(Q_{12}) + Q_{13} + r4(Q_{14}) + Q_{15} \\
P_8 &= Q_8 + r1(Q_9) + Q_{10} + r2(Q_{11}) + Q_{12} + r3(Q_{13}) + Q_{14} + r4(Q_{15}) \\
P_9 &= Q_9 + r1(Q_{10}) + Q_{11} + r2(Q_{12}) + Q_{13} + r3(Q_{14}) + Q_{15} \\
P_{10} &= Q_{10} + r1(Q_{11}) + Q_{12} + r2(Q_{13}) + Q_{14} + r3(Q_{15}) \\
P_{11} &= Q_{11} + r1(Q_{12}) + Q_{13} + r2(Q_{14}) + Q_{15} \\
P_{12} &= Q_{12} + r1(Q_{13}) + Q_{14} + r2(Q_{15}) \\
P_{13} &= Q_{13} + r1(Q_{14}) + Q_{15} \\
P_{14} &= Q_{14} + r1(Q_{15}) \\
P_{15} &= Q_{15}
\end{aligned}$$

Analyzing $expand_{1,2}()$ transformations we can locate parts that can be considered as expressions with variables which are known in advance before the expansion. As we can see, these equations creates a triangle. We call it upper triangle T^U.

Let us denote the transformation K^A of the key addition to the vector P i.e. $R = K^A(P, K) = P + A = (R_0, \ldots, R_{15})$, where $R_i = A_i + P_i$, $i = 0, \ldots, 15$.

Finally, we define lower triangle transformation $Q_b = T^L(R)$ as:

$$
\begin{aligned}
Q_{16} &= R_0 \\
Q_{17} &= R_1 + s0(Q_{16}) \\
Q_{18} &= R_2 + s4(Q_{16}) + s5(Q_{17}) \\
Q_{19} &= R_3 + r7(Q_{16}) + s4(Q_{17}) + s5(Q_{18}) \\
Q_{20} &= R_4 + Q_{16} + r7(Q_{17}) + s4(Q_{18}) + s5(Q_{19}) \\
Q_{21} &= R_5 + r6(Q_{16}) + Q_{17} + r7(Q_{18}) + s4(Q_{19}) + s5(Q_{20}) \\
Q_{22} &= R_6 + Q_{16} + r6(Q_{17}) + Q_{18} + r7(Q_{19}) + s4(Q_{20}) + s5(Q_{21}) \\
Q_{23} &= R_7 + r5(Q_{16}) + Q_{17} + r6(Q_{18}) + Q_{19} + r7(Q_{20}) + s4(Q_{21}) + s5(Q_{22}) \\
Q_{24} &= R_8 + Q_{16} + r5(Q_{17}) + Q_{18} + r6(Q_{19}) + Q_{20} + r7(Q_{21}) + s4(Q_{22}) + s5(Q_{23}) \\
Q_{25} &= R_9 + r4(Q_{16}) + Q_{17} + r5(Q_{18}) + Q_{19} + r6(Q_{20}) + Q_{21} + r7(Q_{22}) + s4(Q_{23}) + s5(Q_{24}) \\
Q_{26} &= R_{10} + Q_{16} + r4(Q_{17}) + Q_{18} + r5(Q_{19}) + Q_{20} + r6(Q_{21}) + Q_{22} + r7(Q_{23}) + s4(Q_{24}) + s5(Q_{25}) \\
Q_{27} &= R_{11} + r3(Q_{16}) + Q_{17} + r4(Q_{18}) + Q_{19} + r5(Q_{20}) + Q_{21} + r6(Q_{22}) + Q_{23} + r7(Q_{24}) + s4(Q_{25}) + s5(Q_{26}) \\
Q_{28} &= R_{12} + Q_{16} + r3(Q_{17}) + Q_{18} + r4(Q_{19}) + Q_{20} + r5(Q_{21}) + Q_{22} + r6(Q_{23}) + Q_{24} + r7(Q_{25}) + s4(Q_{26}) + s5(Q_{27}) \\
Q_{29} &= R_{13} + r2(Q_{16}) + Q_{17} + r3(Q_{18}) + Q_{19} + r4(Q_{20}) + Q_{21} + r5(Q_{22}) + Q_{23} + r6(Q_{24}) + Q_{25} + r7(Q_{26}) + s4(Q_{27}) + s5(Q_{28}) \\
Q_{30} &= R_{14} + Q_{16} + r2(Q_{17}) + Q_{18} + r3(Q_{19}) + Q_{20} + r4(Q_{21}) + Q_{22} + r5(Q_{23}) + Q_{24} + r6(Q_{25}) + Q_{26} + r7(Q_{27}) + s4(Q_{28}) + s5(Q_{29}) \\
Q_{31} &= R_{15} + r1(Q_{16}) + Q_{17} + r2(Q_{18}) + Q_{19} + r3(Q_{20}) + Q_{21} + r4(Q_{22}) + Q_{23} + r5(Q_{24}) + Q_{25} + r6(Q_{26}) + Q_{27} + r7(Q_{28}) + s4(Q_{29}) + s5(Q_{30})
\end{aligned}
$$

We can see that the upper triangle transformation could be computed independently first, then we can apply key addition to the result and then we can apply the lower triangle transformation on the result. In other words the upper triangular computation is parallelizable, while the lower triangular computation uses variables just computed, i.e. it uses a feedback. Note that the decomposition $f_1 = T^L \circ K^A \circ T^U$ doesn't depend on the choice of tunable parameter. The choice has influence to the internal definition of the transformations T^U and T^L, not to the decomposition and character of T^U and T^L themself.

Having this decomposition $f_1 = T^L \circ K^A \circ T^U$ where T^U, K^A and T^L are bijective and/or multipermutation transformations, we can get the following decomposition:

$$
\begin{aligned}
P &= T^U(Q_a), \\
A &= AddElement(M, H) = (B(rotM) + K) \oplus ROTL^7(H), \\
R &= K^A(P, A) = P + A, \\
Q_b &= T^L(R)
\end{aligned}
$$

or

$$Q_b = f_1(M, H, Q_a) = T^L(T^U(Q_a) + A),$$

or

$$Q_b = f_1(M, H, Q_a) = T^L(T^U(Q_a) + ((B(rotM) + K) \oplus ROTL^7(H))).$$

This decomposition shows very simply how the variables are processed. It separates inputs in a clear and understandable way and shows how they are mixed using different bijections and multipermutations.

3.5 Decomposition of the function f_2

The folding function f_2 compresses the three inputs M, Q_a and Q_b into $H = f_2(M, Q_a, Q_b)$. It uses bijective binary linear transformation (matrix) L, which is divided into two matrices L_a and L_b, where $L = L_a \oplus L_b$.

$$
L_a(Q_b) = \begin{pmatrix}
SHL^5(XH) \oplus & SHR^5(Q_{16}^{(i)}) \\
SHR^7(XH) \oplus & SHL^8(Q_{17}^{(i)}) \\
SHR^5(XH) \oplus & SHL^5(Q_{18}^{(i)}) \\
SHR^1(XH) \oplus & SHL^5(Q_{19}^{(i)}) \\
SHR^3(XH) \oplus & Q_{20}^{(i)} \\
SHL^6(XH) \oplus & SHR^6(Q_{21}^{(i)}) \\
SHR^4(XH) \oplus & SHL^6(Q_{22}^{(i)}) \\
SHR^{11}(XH) \oplus & SHL^2(Q_{23}^{(i)}) \\
XH \oplus & Q_{24}^{(i)} \\
XH \oplus & Q_{25}^{(i)} \\
XH \oplus & Q_{26}^{(i)} \\
XH \oplus & Q_{27}^{(i)} \\
XH \oplus & Q_{28}^{(i)} \\
XH \oplus & Q_{29}^{(i)} \\
XH \oplus & Q_{30}^{(i)} \\
XH \oplus & Q_{31}^{(i)}
\end{pmatrix}, \qquad
L_b(Q_b) = \begin{pmatrix}
XL \oplus & Q_{24}^{(i)} \\
XL \oplus & Q_{25}^{(i)} \\
XL \oplus & Q_{26}^{(i)} \\
XL \oplus & Q_{27}^{(i)} \\
XL \oplus & Q_{28}^{(i)} \\
XL \oplus & Q_{29}^{(i)} \\
XL \oplus & Q_{30}^{(i)} \\
XL \oplus & Q_{31}^{(i)} \\
SHL^8(XL) \oplus & Q_{23}^{(i)} \\
SHR^6(XL) \oplus & Q_{16}^{(i)} \\
SHL^6(XL) \oplus & Q_{17}^{(i)} \\
SHL^4(XL) \oplus & Q_{18}^{(i)} \\
SHR^3(XL) \oplus & Q_{19}^{(i)} \\
SHR^4(XL) \oplus & Q_{20}^{(i)} \\
SHR^7(XL) \oplus & Q_{21}^{(i)} \\
SHR^2(XL) \oplus & Q_{22}^{(i)}
\end{pmatrix}
$$

The matrices L_a and L_b are not bijections, but have the ranks near to the full rank value. They were selected to provide fast binary mixing separately on their variables, and giving a bijection together.

The function f_2 can be decomposed into several functions. We define:

$$\begin{aligned} f_3(M, Q_b) &= L_a(Q_b) \oplus M, \\ f_4(Q_a, Q_b) &= L_b(Q_b) \oplus Q_a, \\ G &= f_3(M, Q_b) + f_4(Q_a, Q_b) = (M \oplus L_a(Q_b)) + (Q_a \oplus L_b(Q_b)), \\ H &= f_6(G) = G + f_5(G), \end{aligned}$$

where f_5 is Feistel-like transformation and where f_6 is one Feistel-like round.

$$f_5(X) = \begin{pmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ ROTL^{9}(X_4) \\ ROTL^{10}(X_5) \\ ROTL^{11}(X_6) \\ ROTL^{12}(X_7) \\ ROTL^{13}(X_0) \\ ROTL^{14}(X_1) \\ ROTL^{15}(X_2) \\ ROTL^{16}(X_3) \end{pmatrix}.$$

Finally we can write decomposition of f_2 as:

$$\begin{aligned} f_2 &= f_6(f_3 + f_4), \\ H &= f_2(M, Q_a, Q_b) = f_6((M \oplus L_a(Q_b)) + (Q_a \oplus L_b(Q_b))). \end{aligned}$$

From cryptographic point of view, transformation f_6 is very useful for statistical randomization of the result. On the other hand for cryptanalytic purposes (for instance when studying collisions or preimages), it is possible to explore at first the value G, because H is a bijective image just only of G, without input of any other variable. When we want to find collisions or pseudo-collisions, it suffices to find out them on G-values. Note however that in the case of (pseudo) preimages, the final invocation of the compression function is very restrictive and is there as a security measure against those pseudo-attacks.

In any case we have this simple decomposition:

$$G = (M \oplus L_a(Q_b)) + (Q_a \oplus L_b(Q_b)).$$

3.6 Decomposition of the compression function f

Now we can write the complete decomposition of the compression function f. The value $newH$ is computed as follows, what is the complete decomposition of the compression function f:

$$\begin{aligned} Q_a &= f_0(M, H) = A_2(A_1(M \oplus H)) + ROTL^1(H), \\ Q_b &= f_1(M, H, Q_a) = T^L(T^U(Q_a) + ((B(rotM) + K) \oplus ROTL^7(H))), \\ G &= f_3(M, Q_b) + f_4(Q_a, Q_b) = (M \oplus L_a(Q_b)) + (Q_a \oplus L_b(Q_b)), \\ newH &= f(M, H) = f_6(G). \end{aligned}$$

The transformation $L(Q_b) = L_a(Q_b) \oplus L_a(Q_b)$ is a dominant part in the G-value $(M \oplus L_a(Q_b)) + (Q_a \oplus L_b(Q_b))$. As we already mentioned, the partial linear transformations L_a and L_b are not bijective, but they have a very high rank (so they are near to bijective transformations). Additionally both parts L_a and L_b are summed together arithmetically, and thus when we don't assume carry bits, they are summed linearly together and creates the linear bijective image $L(Q_b)$ of Q_b.

From cryptanalytic point of view we can approximate H by the G value and the G value $((M \oplus L_a(Q_b)) + (Q_a \oplus L_b(Q_b)))$ by the expression $M \oplus L(Q_b) \oplus Q_a$ or even more simply by $M \oplus Q_b \oplus Q_a$. Then we have

$$G_{simplified} \approx M \oplus Q_a \oplus Q_b = M \oplus Q_a \oplus T^L(T^U(Q_a) + ((B(rotM) + K) \oplus ROTL^7(H))).$$

By this simplification where we omitted the function f_6 we can analyze the value G instead of $newH$. Thus someone can first start to analyze the following simplified and decomposed compression function f:

$$\begin{aligned} Q_a &= A_2A_1(M \oplus H) + ROTL^1(H), \\ Q_b &= T^L(T^U(Q_a) + ((B(rotM) + K) \oplus ROTL^7(H))), \\ G &= (M \oplus L_a(Q_b)) + (Q_a \oplus L_b(Q_b)). \end{aligned}$$

We hope that this simple symbolic expression would help cryptanalysts to analyze the hash function and its parts more effectively, because the relations between variables are now visible more clearly.

4 Bijections and multipermutations in the decompositions

In this section we will number all bijective properties that posses all components of Blue Midnight Wish. All those properties can be formally and mathematically proved as different Lemmas or Propositions, but in this paper we just give them without formal proofs.

1. Function f_0
   - $A_0(M, H)$ is a multipermutation
   - $A_1(X)$ is a bijection
   - All $s_i()$ (xorshifts) are bijections
   - $A_2(W)$ is a bijection
   - $A_3(S, H)$ is a multipermutation
   - When H is fixed, $f_0(M, H)$ is a bijection
2. Function f_1
   - $T^U(Q_a)$ is a bijection
   - B is a bijection
   - $AddElement(M, H)$ is a multipermutation
   - $K^A(P, K)$ is a multipermutation
   - $T^L(R)$ is a bijection
   - When A is fixed, f_1 is a bijection between Q_a and Q_b
   - When Q_a is fixed, f_1 is a bijection between A and Q_b
   - When Q_b is fixed, f_1 is a bijection between A and Q_a
3. Function f_2
   - L is a bijection
   - f_3 is a multipermutation
   - f_4 is a multipermutation
   - $f_5(G)$ as a function of the first half of the variable G is a bijection
   - f_6 is a bijection
   - When Q_b and M are fixed, $f_2(Q_a)$ is a bijection
   - When Q_b and Q_a are fixed, $f_2(M)$ is a bijection

4.1 Some design rationales for Blue Midnight Wish

Additionally to the design rationales given in the official Blue Midnight Wish documentation here we give several more.

- For different versions of Blue Midnight Wish we use the abbreviations BMW224, BMW256, BMW384 and BMW512. The abbreviation BMW can be interpreted also as "Bijections Mounted Widely" because Blue Midnight Wish uses many bijections and multipermutations entangled together in a complex way.
- Guaranteed change. Bijections and multipermutations guarantee propagation of changes. When we fix some value inside, due to many bijections, it has large consequences on fixing more complex values elsewhere. And this property diffuses through the scheme.
- Bijections and multipermutations are made by arithmetical or binary transformations, which are always alternating. It assures non-linearity and mixes variables from two essentially different algebraic structures together.
- Most of bijections and multipermutations provide fast diffusion.
- The use of basic instructions ADD, XOR, Shift Left and Shift Right do not create weaknesses for side channel attacks.

5 Conclusions

In this paper we gave simple symbolic description and decomposition of Blue Midnight Wish family. This gave us a tool for understanding the strength and role of internal design parts and variables.

We hope that this decomposition can serve other cryptanalysts in their attempts for finding weaknesses or developing new attacks. On the other hand, by expressing all the main principles behind the Blue Midnight Wish design in this transparent way, we think that the security of Blue Midnight Wish can be understand even better.

References

[1] Announcing Request for Candidate Algorithm Nominations for a New Cryptographic Hash Algorithm (SHA-3) Family, 2007, NIST, `http://csrc.nist.gov/groups/ST/hash/index.html`

[2] Danilo Gligoroski, Vlastimil Klima, Svein Johan Knapskog, Mohamed El-Hadedy, Jørn Amundsen, Stig Frode Mjølsnes: Cryptographic Hash Function Blue Midnight Wish, September 2009, `http://people.item.ntnu.no/~danilog/Hash/BMW-SecondRound/Supporting_Documentation/BlueMidnightWishDocumentation.pdf`

Appendix: Definitions of the functions f_0, f_1, f_2 and the logic functions

Here we show necessary definitions. It is possible to find them also in [2]. The function $f_0 : \{0,1\}^{2m} \rightarrow \{0,1\}^m$ is defined in the Table 2.

$f_0 : \{0,1\}^{2m} \rightarrow \{0,1\}^m$

Input: Message block $M^{(i)} = (M_0^{(i)}, M_1^{(i)}, \ldots, M_{15}^{(i)})$, and the previous double pipe $H^{(i-1)} = (H_0^{(i-1)}, H_1^{(i-1)}, \ldots, H_{15}^{(i-1)})$.

Output: First part of the quadruple pipe $Q_a^{(i)} = (Q_0^{(i)}, Q_1^{(i)}, \ldots, Q_{15}^{(i)})$.

1. Bijective transform of $M^{(i)} \oplus H^{(i-1)}$:

$$
\begin{array}{lcccccccccc}
W_0^{(i)} &=& (M_5^{(i)} \oplus H_5^{(i-1)}) &-& (M_7^{(i)} \oplus H_7^{(i-1)}) &+& (M_{10}^{(i)} \oplus H_{10}^{(i-1)}) &+& (M_{13}^{(i)} \oplus H_{13}^{(i-1)}) &+& (M_{14}^{(i)} \oplus H_{14}^{(i-1)}) \\
W_1^{(i)} &=& (M_6^{(i)} \oplus H_6^{(i-1)}) &-& (M_8^{(i)} \oplus H_8^{(i-1)}) &+& (M_{11}^{(i)} \oplus H_{11}^{(i-1)}) &+& (M_{14}^{(i)} \oplus H_{14}^{(i-1)}) &-& (M_{15}^{(i)} \oplus H_{15}^{(i-1)}) \\
W_2^{(i)} &=& (M_0^{(i)} \oplus H_0^{(i-1)}) &+& (M_7^{(i)} \oplus H_7^{(i-1)}) &+& (M_9^{(i)} \oplus H_9^{(i-1)}) &-& (M_{12}^{(i)} \oplus H_{12}^{(i-1)}) &+& (M_{15}^{(i)} \oplus H_{15}^{(i-1)}) \\
W_3^{(i)} &=& (M_0^{(i)} \oplus H_0^{(i-1)}) &-& (M_1^{(i)} \oplus H_1^{(i-1)}) &+& (M_8^{(i)} \oplus H_8^{(i-1)}) &-& (M_{10}^{(i)} \oplus H_{10}^{(i-1)}) &+& (M_{13}^{(i)} \oplus H_{13}^{(i-1)}) \\
W_4^{(i)} &=& (M_1^{(i)} \oplus H_1^{(i-1)}) &+& (M_2^{(i)} \oplus H_2^{(i-1)}) &+& (M_9^{(i)} \oplus H_9^{(i-1)}) &-& (M_{11}^{(i)} \oplus H_{11}^{(i-1)}) &-& (M_{14}^{(i)} \oplus H_{14}^{(i-1)}) \\
W_5^{(i)} &=& (M_3^{(i)} \oplus H_3^{(i-1)}) &-& (M_2^{(i)} \oplus H_2^{(i-1)}) &+& (M_{10}^{(i)} \oplus H_{10}^{(i-1)}) &-& (M_{12}^{(i)} \oplus H_{12}^{(i-1)}) &+& (M_{15}^{(i)} \oplus H_{15}^{(i-1)}) \\
W_6^{(i)} &=& (M_4^{(i)} \oplus H_4^{(i-1)}) &-& (M_0^{(i)} \oplus H_0^{(i-1)}) &-& (M_3^{(i)} \oplus H_3^{(i-1)}) &-& (M_{11}^{(i)} \oplus H_{11}^{(i-1)}) &+& (M_{13}^{(i)} \oplus H_{13}^{(i-1)}) \\
W_7^{(i)} &=& (M_1^{(i)} \oplus H_1^{(i-1)}) &-& (M_4^{(i)} \oplus H_4^{(i-1)}) &-& (M_5^{(i)} \oplus H_5^{(i-1)}) &-& (M_{12}^{(i)} \oplus H_{12}^{(i-1)}) &-& (M_{14}^{(i)} \oplus H_{14}^{(i-1)}) \\
W_8^{(i)} &=& (M_2^{(i)} \oplus H_2^{(i-1)}) &-& (M_5^{(i)} \oplus H_5^{(i-1)}) &-& (M_6^{(i)} \oplus H_6^{(i-1)}) &+& (M_{13}^{(i)} \oplus H_{13}^{(i-1)}) &-& (M_{15}^{(i)} \oplus H_{15}^{(i-1)}) \\
W_9^{(i)} &=& (M_0^{(i)} \oplus H_0^{(i-1)}) &-& (M_3^{(i)} \oplus H_3^{(i-1)}) &+& (M_6^{(i)} \oplus H_6^{(i-1)}) &-& (M_7^{(i)} \oplus H_7^{(i-1)}) &+& (M_{14}^{(i)} \oplus H_{14}^{(i-1)}) \\
W_{10}^{(i)} &=& (M_8^{(i)} \oplus H_8^{(i-1)}) &-& (M_1^{(i)} \oplus H_1^{(i-1)}) &-& (M_4^{(i)} \oplus H_4^{(i-1)}) &-& (M_7^{(i)} \oplus H_7^{(i-1)}) &+& (M_{15}^{(i)} \oplus H_{15}^{(i-1)}) \\
W_{11}^{(i)} &=& (M_8^{(i)} \oplus H_8^{(i-1)}) &-& (M_0^{(i)} \oplus H_0^{(i-1)}) &-& (M_2^{(i)} \oplus H_2^{(i-1)}) &-& (M_5^{(i)} \oplus H_5^{(i-1)}) &+& (M_9^{(i)} \oplus H_9^{(i-1)}) \\
W_{12}^{(i)} &=& (M_1^{(i)} \oplus H_1^{(i-1)}) &+& (M_3^{(i)} \oplus H_3^{(i-1)}) &-& (M_6^{(i)} \oplus H_6^{(i-1)}) &-& (M_9^{(i)} \oplus H_9^{(i-1)}) &+& (M_{10}^{(i)} \oplus H_{10}^{(i-1)}) \\
W_{13}^{(i)} &=& (M_2^{(i)} \oplus H_2^{(i-1)}) &+& (M_4^{(i)} \oplus H_4^{(i-1)}) &+& (M_7^{(i)} \oplus H_7^{(i-1)}) &+& (M_{10}^{(i)} \oplus H_{10}^{(i-1)}) &+& (M_{11}^{(i)} \oplus H_{11}^{(i-1)}) \\
W_{14}^{(i)} &=& (M_3^{(i)} \oplus H_3^{(i-1)}) &-& (M_5^{(i)} \oplus H_5^{(i-1)}) &+& (M_8^{(i)} \oplus H_8^{(i-1)}) &-& (M_{11}^{(i)} \oplus H_{11}^{(i-1)}) &-& (M_{12}^{(i)} \oplus H_{12}^{(i-1)}) \\
W_{15}^{(i)} &=& (M_{12}^{(i)} \oplus H_{12}^{(i-1)}) &-& (M_4^{(i)} \oplus H_4^{(i-1)}) &-& (M_6^{(i)} \oplus H_6^{(i-1)}) &-& (M_9^{(i)} \oplus H_9^{(i-1)}) &+& (M_{13}^{(i)} \oplus H_{13}^{(i-1)})
\end{array}
$$

2. Further bijective transform of $W_j^{(i)}, j = 0, \ldots, 15$:

$$
\begin{array}{llll}
Q_0^{(i)} = s_0(W_0^{(i)}) + H_1^{(i-1)}; & Q_1^{(i)} = s_1(W_1^{(i)}) + H_2^{(i-1)}; & Q_2^{(i)} = s_2(W_2^{(i)}) + H_3^{(i-1)}; & Q_3^{(i)} = s_3(W_3^{(i)}) + H_4^{(i-1)}; \\
Q_4^{(i)} = s_4(W_4^{(i)}) + H_5^{(i-1)}; & Q_5^{(i)} = s_0(W_5^{(i)}) + H_6^{(i-1)}; & Q_6^{(i)} = s_1(W_6^{(i)}) + H_7^{(i-1)}; & Q_7^{(i)} = s_2(W_7^{(i)}) + H_8^{(i-1)}; \\
Q_8^{(i)} = s_3(W_8^{(i)}) + H_9^{(i-1)}; & Q_9^{(i)} = s_4(W_9^{(i)}) + H_{10}^{(i-1)}; & Q_{10}^{(i)} = s_0(W_{10}^{(i)}) + H_{11}^{(i-1)}; & Q_{11}^{(i)} = s_1(W_{11}^{(i)}) + H_{12}^{(i-1)}; \\
Q_{12}^{(i)} = s_2(W_{12}^{(i)}) + H_{13}^{(i-1)}; & Q_{13}^{(i)} = s_3(W_{13}^{(i)}) + H_{14}^{(i-1)}; & Q_{14}^{(i)} = s_4(W_{14}^{(i)}) + H_{15}^{(i-1)}; & Q_{15}^{(i)} = s_0(W_{15}^{(i)}) + H_0^{(i-1)};
\end{array}
$$

Table 2: Definition of the function f_0 of Blue Midnight Wish

The function $f_1 : \{0,1\}^{3m} \rightarrow \{0,1\}^m$ is defined in the Table 3.

$f_1 : \{0,1\}^{3m} \rightarrow \{0,1\}^m$
Input: Message block $M^{(i)} = (M_0^{(i)}, M_1^{(i)}, \ldots, M_{15}^{(i)})$, the previous double pipe $H^{(i-1)} = (H_0^{(i-1)}, H_1^{(i-1)}, \ldots, H_{15}^{(i-1)})$ and the first part of the quadruple pipe $Q_a^{(i)} = (Q_0^{(i)}, Q_1^{(i)}, \ldots, Q_{15}^{(i)})$. **Output:** Second part of the quadruple pipe $Q_b^{(i)} = (Q_{16}^{(i)}, Q_{17}^{(i)}, \ldots, Q_{31}^{(i)})$.
1. Double pipe expansion according to the tunable parameters $ExpandRounds_1$ and $ExpandRounds_2$. **1.1** For $ii = 0$ to $ExpandRounds_1 - 1$ $Q_{ii+16}^{(i)} = expand_1(ii + 16)$ **1.2** For $ii = ExpandRounds_1$ to $ExpandRounds_1 + ExpandRounds_2 - 1$ $Q_{ii+16}^{(i)} = expand_2(ii + 16)$

Table 3: Definition of the function f_1 of Blue Midnight Wish

The function $f_2 : \{0,1\}^{3m} \rightarrow \{0,1\}^m$ is defined in the Table 4.

Folding $f_2 : \{0,1\}^{3m} \rightarrow \{0,1\}^m$

Input: Message block $M^{(i)} = (M_0^{(i)}, M_1^{(i)}, \ldots, M_{15}^{(i)})$,
quadruple pipe $Q^{(i)} = (Q_0^{(i)}, Q_1^{(i)}, \ldots, Q_{15}^{(i)}, Q_{16}^{(i)}, \ldots, Q_{31}^{(i)})$.
Output: New double pipe $H^{(i)} = (H_0^{(i)}, H_1^{(i)}, \ldots, H_{15}^{(i)})$.

1. Compute the cumulative temporary variables XL and XH.

$$\begin{aligned} XL &= Q_{16}^{(i)} \oplus Q_{17}^{(i)} \oplus \ldots \oplus Q_{23}^{(i)} \\ XH &= XL \oplus Q_{24}^{(i)} \oplus Q_{25}^{(i)} \oplus \ldots \oplus Q_{31}^{(i)} \end{aligned}$$

2. Compute the new double pipe $H^{(i)}$:

$$\begin{aligned}
H_0^{(i)} &= \left(SHL^5(XH) \oplus SHR^5(Q_{16}^{(i)}) \oplus M_0^{(i)}\right) + \left(XL \oplus Q_{24}^{(i)} \oplus Q_0^{(i)}\right) \\
H_1^{(i)} &= \left(SHR^7(XH) \oplus SHL^8(Q_{17}^{(i)}) \oplus M_1^{(i)}\right) + \left(XL \oplus Q_{25}^{(i)} \oplus Q_1^{(i)}\right) \\
H_2^{(i)} &= \left(SHR^5(XH) \oplus SHL^5(Q_{18}^{(i)}) \oplus M_2^{(i)}\right) + \left(XL \oplus Q_{26}^{(i)} \oplus Q_2^{(i)}\right) \\
H_3^{(i)} &= \left(SHR^1(XH) \oplus SHL^5(Q_{19}^{(i)}) \oplus M_3^{(i)}\right) + \left(XL \oplus Q_{27}^{(i)} \oplus Q_3^{(i)}\right) \\
H_4^{(i)} &= \left(SHR^3(XH) \oplus Q_{20}^{(i)} \oplus M_4^{(i)}\right) + \left(XL \oplus Q_{28}^{(i)} \oplus Q_4^{(i)}\right) \\
H_5^{(i)} &= \left(SHL^6(XH) \oplus SHR^6(Q_{21}^{(i)}) \oplus M_5^{(i)}\right) + \left(XL \oplus Q_{29}^{(i)} \oplus Q_5^{(i)}\right) \\
H_6^{(i)} &= \left(SHR^4(XH) \oplus SHL^6(Q_{22}^{(i)}) \oplus M_6^{(i)}\right) + \left(XL \oplus Q_{30}^{(i)} \oplus Q_6^{(i)}\right) \\
H_7^{(i)} &= \left(SHR^{11}(XH) \oplus SHL^2(Q_{23}^{(i)}) \oplus M_7^{(i)}\right) + \left(XL \oplus Q_{31}^{(i)} \oplus Q_7^{(i)}\right) \\
H_8^{(i)} &= ROTL^9(H_4^{(i)}) + \left(XH \oplus Q_{24}^{(i)} \oplus M_8^{(i)}\right) + \left(SHL^8(XL) \oplus Q_{23}^{(i)} \oplus Q_8^{(i)}\right) \\
H_9^{(i)} &= ROTL^{10}(H_5^{(i)}) + \left(XH \oplus Q_{25}^{(i)} \oplus M_9^{(i)}\right) + \left(SHR^6(XL) \oplus Q_{16}^{(i)} \oplus Q_9^{(i)}\right) \\
H_{10}^{(i)} &= ROTL^{11}(H_6^{(i)}) + \left(XH \oplus Q_{26}^{(i)} \oplus M_{10}^{(i)}\right) + \left(SHL^6(XL) \oplus Q_{17}^{(i)} \oplus Q_{10}^{(i)}\right) \\
H_{11}^{(i)} &= ROTL^{12}(H_7^{(i)}) + \left(XH \oplus Q_{27}^{(i)} \oplus M_{11}^{(i)}\right) + \left(SHL^4(XL) \oplus Q_{18}^{(i)} \oplus Q_{11}^{(i)}\right) \\
H_{12}^{(i)} &= ROTL^{13}(H_0^{(i)}) + \left(XH \oplus Q_{28}^{(i)} \oplus M_{12}^{(i)}\right) + \left(SHR^3(XL) \oplus Q_{19}^{(i)} \oplus Q_{12}^{(i)}\right) \\
H_{13}^{(i)} &= ROTL^{14}(H_1^{(i)}) + \left(XH \oplus Q_{29}^{(i)} \oplus M_{13}^{(i)}\right) + \left(SHR^4(XL) \oplus Q_{20}^{(i)} \oplus Q_{13}^{(i)}\right) \\
H_{14}^{(i)} &= ROTL^{15}(H_2^{(i)}) + \left(XH \oplus Q_{30}^{(i)} \oplus M_{14}^{(i)}\right) + \left(SHR^7(XL) \oplus Q_{21}^{(i)} \oplus Q_{14}^{(i)}\right) \\
H_{15}^{(i)} &= ROTL^{16}(H_3^{(i)}) + \left(XH \oplus Q_{31}^{(i)} \oplus M_{15}^{(i)}\right) + \left(SHR^2(XL) \oplus Q_{22}^{(i)} \oplus Q_{15}^{(i)}\right)
\end{aligned}$$

Table 4: Definition of the folding function f_2 of Blue Midnight Wish

Blue Midnight Wish uses the logic functions, summarized in Table 5.

BMW224/BMW256

$$
\begin{aligned}
s_0(x) &= SHR^1(x) \oplus SHL^3(x) \oplus ROTL^4(x) \quad \oplus ROTL^{19}(x)\\
s_1(x) &= SHR^1(x) \oplus SHL^2(x) \oplus ROTL^8(x) \oplus ROTL^{23}(x)\\
s_2(x) &= SHR^2(x) \oplus SHL^1(x) \oplus ROTL^{12}(x) \oplus ROTL^{25}(x)\\
s_3(x) &= SHR^2(x) \oplus SHL^2(x) \oplus ROTL^{15}(x) \oplus ROTL^{29}(x)\\
s_4(x) &= SHR^1(x) \oplus x\\
s_5(x) &= SHR^2(x) \oplus x\\
r_1(x) &= ROTL^3(x)\\
r_2(x) &= ROTL^7(x)\\
r_3(x) &= ROTL^{13}(x)\\
r_4(x) &= ROTL^{16}(x)\\
r_5(x) &= ROTL^{19}(x)\\
r_6(x) &= ROTL^{23}(x)\\
r_7(x) &= ROTL^{27}(x)
\end{aligned}
$$

$$
\begin{aligned}
AddElement(j) = \Big(& ROTL^{((j \bmod 16)+1)}(M_j^{(i)}) + \\
& ROTL^{((j+3 \bmod 16)+1)}(M_{j+3}^{(i)}) - ROTL^{((j+10 \bmod 16)+1)}(M_{j+10}^{(i)}) + \\
& K_{j+16} \Big) \oplus H_{j+7}^{(i)}
\end{aligned}
$$

$$
\begin{aligned}
expand_1(j) = \quad & s_1(Q_{j-16}^{(i)}) + s_2(Q_{j-15}^{(i)}) + s_3(Q_{j-14}^{(i)}) + s_0(Q_{j-13}^{(i)}) \\
+ \quad & s_1(Q_{j-12}^{(i)}) + s_2(Q_{j-11}^{(i)}) + s_3(Q_{j-10}^{(i)}) + s_0(Q_{j-9}^{(i)}) \\
+ \quad & s_1(Q_{j-8}^{(i)}) + s_2(Q_{j-7}^{(i)}) + s_3(Q_{j-6}^{(i)}) + s_0(Q_{j-5}^{(i)}) \\
+ \quad & s_1(Q_{j-4}^{(i)}) + s_2(Q_{j-3}^{(i)}) + s_3(Q_{j-2}^{(i)}) + s_0(Q_{j-1}^{(i)}) \\
& + AddElement(j-16)
\end{aligned}
$$

$$
\begin{aligned}
expand_2(j) = \quad & Q_{j-16}^{(i)} + r_1(Q_{j-15}^{(i)}) + Q_{j-14}^{(i)} + r_2(Q_{j-13}^{(i)}) \\
+ \quad & Q_{j-12}^{(i)} + r_3(Q_{j-11}^{(i)}) + Q_{j-10}^{(i)} + r_4(Q_{j-9}^{(i)}) \\
+ \quad & Q_{j-8}^{(i)} + r_5(Q_{j-7}^{(i)}) + Q_{j-6}^{(i)} + r_6(Q_{j-5}^{(i)}) \\
+ \quad & Q_{j-4}^{(i)} + r_7(Q_{j-3}^{(i)}) + s_4(Q_{j-2}^{(i)}) + s_5(Q_{j-1}^{(i)}) \\
& + AddElement(j-16)
\end{aligned}
$$

BMW384/BMW512

$$
\begin{aligned}
s_0(x) &= SHR^1(x) \oplus SHL^3(x) \oplus ROTL^4(x) \oplus ROTL^{37}(x)\\
s_1(x) &= SHR^1(x) \oplus SHL^2(x) \oplus ROTL^{13}(x) \oplus ROTL^{43}(x)\\
s_2(x) &= SHR^2(x) \oplus SHL^1(x) \oplus ROTL^{19}(x) \oplus ROTL^{53}(x)\\
s_3(x) &= SHR^2(x) \oplus SHL^2(x) \oplus ROTL^{28}(x) \oplus ROTL^{59}(x)\\
s_4(x) &= SHR^1(x) \oplus x\\
s_5(x) &= SHR^2(x) \oplus x\\
r_1(x) &= ROTL^5(x)\\
r_2(x) &= ROTL^{11}(x)\\
r_3(x) &= ROTL^{27}(x)\\
r_4(x) &= ROTL^{32}(x)\\
r_5(x) &= ROTL^{37}(x)\\
r_6(x) &= ROTL^{43}(x)\\
r_7(x) &= ROTL^{53}(x)
\end{aligned}
$$

$$
\begin{aligned}
AddElement(j) = \Big(& ROTL^{((j \bmod 16)+1)}(M_j^{(i)}) + \\
& ROTL^{((j+3 \bmod 16)+1)}(M_{j+3}^{(i)}) - ROTL^{((j+10 \bmod 16)+1)}(M_{j+10}^{(i)}) + \\
& K_{j+16} \Big) \oplus H_{j+7}^{(i)}
\end{aligned}
$$

$$
\begin{aligned}
expand_1(j) = \quad & s_1(Q_{j-16}^{(i)}) + s_2(Q_{j-15}^{(i)}) + s_3(Q_{j-14}^{(i)}) + s_0(Q_{j-13}^{(i)}) \\
+ \quad & s_1(Q_{j-12}^{(i)}) + s_2(Q_{j-11}^{(i)}) + s_3(Q_{j-10}^{(i)}) + s_0(Q_{j-9}^{(i)}) \\
+ \quad & s_1(Q_{j-8}^{(i)}) + s_2(Q_{j-7}^{(i)}) + s_3(Q_{j-6}^{(i)}) + s_0(Q_{j-5}^{(i)}) \\
+ \quad & s_1(Q_{j-4}^{(i)}) + s_2(Q_{j-3}^{(i)}) + s_3(Q_{j-2}^{(i)}) + s_0(Q_{j-1}^{(i)}) \\
& + AddElement(j-16)
\end{aligned}
$$

$$
\begin{aligned}
expand_2(j) = \quad & Q_{j-16}^{(i)} + r_1(Q_{j-15}^{(i)}) + Q_{j-14}^{(i)} + r_2(Q_{j-13}^{(i)}) \\
+ \quad & Q_{j-12}^{(i)} + r_3(Q_{j-11}^{(i)}) + Q_{j-10}^{(i)} + r_4(Q_{j-9}^{(i)}) \\
+ \quad & Q_{j-8}^{(i)} + r_5(Q_{j-7}^{(i)}) + Q_{j-6}^{(i)} + r_6(Q_{j-5}^{(i)}) \\
+ \quad & Q_{j-4}^{(i)} + r_7(Q_{j-3}^{(i)}) + s_4(Q_{j-2}^{(i)}) + s_5(Q_{j-1}^{(i)}) \\
& + AddElement(j-16)
\end{aligned}
$$

Table 5: Logic functions used in Blue Midnight Wish. Note that for the function $AddElement(j)$ index expressions involving the variable j for M and H are computed modulo 16.