More Differential Paths of TIB3

Harry Wiggins, Philip Hawkes, Gregory G. Rose, Cameron McDonald

Qualcomm Incorporated
{hwiggins,phawkes,ggr,cameronm}@qualcomm.com

Abstract. The TIB3-256 hashing algorithm [3] is a first round candidate in the SHA-3 competition [2]. Properties of the message expansion and the PHTX function are observed, and then exploited to create new high-probability differential paths through the compression function. Examples conforming to the differential paths are presented. Only one of these differential paths can be applied to the tweaked version of TIB3v2 [4]. Due to the dual-block input mode used in TIB3 and TIB3v2, these differential paths do not seem extensible to the full hash functions. *Note: In the time between when this paper was written and when the paper was made public, the SHA-3 Round 2 Candidates were announced, and TIB3 had been eliminated from the competition.*
Keywords: hash functions, TIB3

1 Introduction

TIB3 [3] is a candidate for the SHA-3 hash function competition organized by NIST [2]. A minor tweak [4] to the original TIB3 algorithm has been proposed by the submitters: we denote the original version by TIB3 and we denote the tweaked version by TIB3v2. This paper primarily examines TIB3. Some results are applicable to TIB3v2: we explicitly note where such results apply.

This analysis searches for high probability differential paths of TIB3. A differential path of a cryptographic algorithm is a description of the differences between internal values that result when processing two related external inputs. A differential path is obtained by analyzing how the differences in the inputs to a component propagate to differences in the outputs of a component. If the two inputs to a component have identical values, then the outputs have identical value with probability one, and the differential is thought of as having "avoided" that component. If the inputs are not identical, then the component must be analyzed, and the differential path is thought of as "going through" that component.

Previous Work. Florian Mendel and Martin Schläffer published a paper [1] presenting a set of differential paths for TIB3 and pseudo-collisions based on those paths. The differential paths of Mendel and Schläffer [1] cleverly avoided the message expansions and PHTX function: these components provide diffusion between bit positions and so the differential paths were restricted to a single bit position of the internal values. Mendel and Schläffer used a linear model for the modular addition operation: that is, the modular addition operation is modeled using an XOR operation for predicting the propagation of bit differences, and appropriate probability factors are introduced as required. The probability of the resulting differential paths was quite high, and consequently pairs of differential paths could be considered concurrently. This large set of differential paths could be exploited to obtain a collision (for more details see [1]).

New Results. The current paper extends the work in [1] by examining differential paths that go through the message expansion procedure and the PHTX function. Our work continues to employ the linear model for the modular addition operation. The analysis of the message expansion procedure finds *message patterns*, corresponding to differences in the message block for which the differences in each roundkey can be predicting with probability one. New high-probability differential paths are found using these message patterns.[1] These differential paths are difficult to describe simply, so the reader is referred to the relevant sections of the paper rather than repeating the differential paths here. Practical examples for some of these paths are provided in the Appendix.

The research found new differential paths with a single active bit position (that is, where differences occur only in one bit position of internal state words): these are listed in Table 19 of Section 6.2. The probabilities of these paths are identical to the probability of the Mendel and Schläffer differential.

The highest probability differential path found using this technique has probability 2^{-4}. This differential path has differences in bit positions 63 and 31 only (where 63 is the most significant bit position). This differential path is shown in Table 24 in Section 7.3, where the differential is discussed in depth. This differential path also applies for TIB3v2 (the analysis was unable to find any other paths for TIB3v2).

The next highest probability is 2^{-10}. Differential paths using only bit positions 63 and 31 are listed in Section 7.4. Section 8.2 provides additional differential paths using bit positions 63, 31 and 20.

Implications of the New Results. This analysis reveals new differential paths with higher probabilities than previously reported. A compression function should not have differential paths of such high probabilities. Only one of these differential paths applies to TIB3v2, indicating that the tweak is a step in the right direction. However the remaining differential path is of very high probability, and the TIB3 designers are encouraged to explore options that eliminate this differential path.

The compression function of TIB3 (and TIB3v2) takes the current message block and the previous message block as input. All the differential paths we have found require a difference in the current message block. As a result, none of these differential paths can be used as the last pair of message blocks containing a difference. Consequently, the differential paths we have found are not sufficient to find a collision in TIB3. The technique of Mendel and Schläffer [1] continues to be the most efficient way to find a collision.

[1] In this research, the probability of the differential path is taken to be the probability from rounds 5 through to round 16.

The analysis in the current paper focusses on TIB3-256, but the TIB3-512 message expansion and the PHTXD have other similar properties that are worth investigating.

1.1 Outline of This Paper

Section 2 contains a brief description of TIB3. An introduction to differential analysis of TIB3, including the result of Mendel and Schläffer, is provided in Section 3. Section 4 describes the properties of the message expansion and the PHTX function used in this analysis. Section 6, Section 7 and Section 8 use these observations to construct paths affecting only one, two and three bit positions (of the internal state variables) respectively. An examination of TIB3v2 is provided in Section 9, and this is followed by the conclusion.

2 TIB3 description

TIB3-256 is an iterated hashing function based on the Merkle-Damgård design principle. It processes message blocks of 512 bits and produces a hash value of 224 or 256 bits. If the message length is not a multiple of 512, a padding procedure is applied. Suppose $m = M_1||M_2||\ldots||M_t$ is the t-block message after padding. The hash value h_{t+1} is calculated as follows :

$$\begin{aligned} h_i &= f(h_{i-1}, M_i||M_{i-1}) \oplus h_{i-1}, \text{for } 1 \leq i \leq t; \\ h_{t+1} &= f(h_t, 0||h_t||M_t) \oplus h_t, \end{aligned}$$

where h_0 and M_0 are predefined initial values. The compression function f consists of two parts: the key schedule, and the state update transformation.

2.1 Key schedule

The key for each compression function has a left part and a right part: $K = (LK, RK)$, each of 512 bits. $LK \oplus RK$ is then expanded to 2048 by means of a modified LFSR via function ψ which takes four 64-bit words W, X, Y, Z as input and outputs one 64-bit word $V = \psi(W, X, Y, Z)$ as defined below:

$$\begin{aligned} V^1 &:= (Y + (Z << 32)) \oplus W \oplus X \oplus (Z >> 32); \\ V^2 &:= V^1 + (V^1 << 32) + (V^1 << 43); \\ V &:= V^2 \oplus (V^2 >> 39). \end{aligned}$$

Firstly $LK \oplus RK$ is loaded into eight 64-bit words $D_0, D_1, \ldots, D_7$, and D_8 and D_9 are computed as follows:

$$\begin{aligned} D_8 &= \psi(D_3 \oplus RK_0, D_4 \oplus RK_1, D_5 \oplus RK_2, D_1 \oplus RK_3); \\ D_9 &= \psi(D_2 \oplus RK_4 \oplus \mathit{const}, D_7 \oplus RK_5 \oplus \mathit{salt}, D_6 \oplus RK_7, D_0 \oplus RK_6); \end{aligned}$$

where *salt* is a salt value and *const*=`0x428a2f98d728ae22`. Next, the remaining D_i, $10 \leq i \leq 31$, are generated recursively as:

$$D_i = \psi(D_{i-10}, D_{i-8}, D_{i-3}, D_{i-2}).$$

These LK_i, RK_j, D_k values are then used to form the 16 roundkeys (each roundkey consists of four 64-bit words):

$$\begin{array}{llll}
\text{Round1}: & D_0, LK_0, D_1, LK_0, & \text{Round2}: & D_2, LK_1, D_3, LK_1, \\
\text{Round3}: & D_4, LK_2, D_5, LK_2, & \text{Round4}: & D_6, LK_3, D_7, LK_3, \\
\text{Round5}: & D_8, LK_4, D_9, LK_4, & \text{Round6}: & D_{10}, LK_5, D_{11}, LK_5, \\
\text{Round7}: & D_{12}, LK_6, D_{13}, LK_6, & \text{Round8}: & D_{14}, LK_7, D_{15}, LK_7, \\
\text{Round9}: & RK_0, D_{16}, RK_1, D_{16}, & \text{Round10}: & RK_2, D_{17}, RK_3, D_{17}, \\
\text{Round11}: & RK_4, D_{18}, RK_5, D_{18}, & \text{Round12}: & RK_6, D_{19}, RK_7, D_{19}, \\
\text{Round13}: & D_{20}, D_{21}, D_{22}, D_{21}, & \text{Round14}: & D_{23}, D_{24}, D_{25}, D_{24}, \\
\text{Round15}: & D_{26}, D_{27}, D_{28}, D_{27}, & \text{Round16}: & D_{29}, D_{30}, D_{31}, D_{30}.
\end{array}$$

For example, in Round 1, roundkeys $= (D_0, LK_0, D_1, LK_0)$.

2.2 State update transformation

The state update transformation of TIB3-256 initializes four 64-bit internal state variables A, C, E, G from h_{i-1}, and the state variables are then updated over 16 rounds. For the purposes of this analysis, we have labeled all the internal values. The inputs to a round are labeled $A_{i-1}, C_{i-1}, E_{i-1}, G_{i-1}$, and the roundkeys are labeled $K = (KA_i, KC_i, KE_i, KG_i)$. For this analysis, the internal values to the round are labeled as follows:

$$\begin{aligned}
(A_a,\ C_a,\ E_a,\ G_a) &:= (A_{i-1},\ C_{i-1},\ E_{i-1},\ G_{i-1} \oplus C_{i-1}); \\
(A_b,\ C_b,\ E_b,\ G_b) &:= (A_a,\ C_a,\ E_{a1},\ G_a) \oplus (KA_i,\ KC_i,\ KE_i,\ KG_i); \\
(A_c,\ C_c,\ E_c,\ G_c) &:= (\mathit{Sbox}(A_b,\ C_b,\ E_b),\ G_b); \\
(A_d,\ C_d,\ E_d,\ G_d) &:= (A_c,\ PHTX(C_c),\ E_c,\ PHTX(G_c)); \\
(A_e,\ C_e,\ E_e,\ G_e) &:= (A_d \tilde{+} G_d,\ C_d,\ E_d,\ G_d \tilde{+} E_d); \\
(A_i,\ C_i,\ E_i,\ G_i) &:= (C_e,\ E_e,\ G_e,\ A_e);
\end{aligned}$$

where the operations are defined as follows:

- For the $\tilde{+}$ operation: the 32 most significant bits (MSBs) of the output correspond to applying addition modulo 2^{32} to the 32 MSBs of the inputs; the 32 least significant bits (MSBs) of the output correspond to applying addition modulo 2^{32} to the 32 LSBs of the inputs.
- The S-box applies a nonlinear mapping from three input bits to three output bits in a bit-wise manner: that is,

  $$(A_c[j], C_c[j], E_c[j]) = \mathit{Sbox}(A_b[j], C_b[j], E_b[j]),$$

  for all j, $0 \le j \le 63$. The S-box is specified in [3].

- The operation $D^* = PHTX(D)$ is computed as:

$$\tilde{D} = D + (D << 32) + (D << 47),$$
$$D^* = \tilde{D} \oplus (\tilde{D} >> 32) \oplus (\tilde{D} >> 43).$$

  In this paper, the *PHTX* to the C_c variables is labeled the *C-PHTX*, while the application of the *PHTX* to the G_c variables is labeled the *G-PHTX*.

We adopt the notation that 63 is the index of the most significant bit and 0 is the index of the least significant bit in a 64-bit word.

3 Introduction to Differential Attacks

Recall that a differential path of a cryptographic algorithm is a description of the difference between internal values that result when processing two sets of external inputs. For this introduction, we indicate values of the internal parameters resulting when processing the first (second) set of external input by adding one prime $'$ (two primes $''$). For example, A_0' is the value of A_0 when processing the first set of external inputs and A_0'' is the initial value of A_0 when processing the second set of external inputs. In a differential analysis, we typically assume that the two values of the parameters (e.g. A_0' and A_0'') are not important, but the difference between those values is important.

The attacker can choose what notion of difference best suits the analysis. Typically, when analyzing an algorithm that has extensive use of a particular group operation $\star$ (with group inverse x denoted using x^{-1}), then the differences between the two values of a parameter X are defined as $\Delta_\star X = X'' \star (X')^{-1}$. This definition is useful since if $Z = X \star Y$ then $\Delta_\star Z = \Delta_\star X \star \Delta_\star Y$. Note that if e denotes the identity of the group, then $\Delta X = e$ implies that $X' = X''$: that is, the two values are identical.

In TIB3, the 64-bit XOR operation is used extensively, so the notion of difference used is $\Delta_\oplus X := X'' \oplus (X')^{-1} := X'' \oplus X'$, since X' is the group inverse of itself. Since only one notion of difference is used in this paper, we usually ignore the subscript of $\oplus$ and simply write ΔX. The group identity is the all zeroes value 0, so a difference of $\Delta X = 0$ implies that the two values X' and X'' are identical. This notion of difference is identical to examining which bits of X' differ from the bits of X'': the j-th bit of ΔX can be determined directly from $X''[j] \oplus X'[j]$ We typically use a "*" to indicate the presence of a bit difference, while a "-" indicates that there is no bit difference. In many cases, is easier to simply specify which bits of X' differ from the bits of X''.

In analyzing TIB3, the differences in the internal state is of particular importance and we use the notation:

$$\begin{aligned}
\Delta_{i-1} &:= (\Delta A_{i-1}, \Delta C_{i-1}, \Delta E_{i-1}, \Delta G_{i-1});\\
\Delta_a &:= (\Delta A_a, \Delta C_a, \Delta E_a, \Delta G_a);\\
\Delta_b &:= (\Delta A_b, \Delta C_b, \Delta E_b, \Delta G_b);\\
\Delta_c &:= (\Delta A_c, \Delta C_c, \Delta E_c, \Delta G_c);\\
\Delta_d &:= (\Delta A_d, \Delta C_d, \Delta E_d, \Delta G_d);\\
\Delta_e &:= (\Delta A_e, \Delta C_e, \Delta E_e, \Delta G_e);\\
\Delta_i &:= (\Delta A_i, \Delta C_i, \Delta E_i, \Delta G_i).
\end{aligned}$$

We are often interested in representing the differences at a single bit position of the internal state. For example, the differences in bit j of the internal state A_i, C_i, E_i, G_i (the inputs to the i-th round) form a 4-entry vector

$$\Delta_i[j] := (\Delta A_{i-1}[j], \Delta C_{i-1}[j], \Delta E_{i-1}[j], \Delta G_{i-1}[j]).$$

To save space, we often convert "*" and "-" to bit values 1 and 0 respectively, and transform the 4-bit vector into the corresponding integer - the use of the integer representation is indicated using $\hat{\Delta}$. For example, $\Delta_i[j] = (-, *, *, -)$ is equivalent to writing $\hat{\Delta}_i[j] = 6$. We indicate the equivalence between the two notations using the "$\sim$" symbol.

3.1 Complexity of a Differential Attack

For a differential path of probability p, we would expect to try p^{-1} pairs of inputs before finding a pair such that the internal differences conform to the path. This is an over simplification particularly in the case of hash functions where the attacker has complete control over the messages being input to the hash function. This typically means that the attacker can control the values in the first few rounds, so the probability of satisfying the differential in these first few rounds is not relevant: the relevant metric is the probability of the differential path in the remaining rounds. For this analysis, we shall assume that the attacker can control the inputs to the first four rounds, so the relevant metric is the probability of the differential path through rounds 5 to 16.

Additionally, we assume for any input difference to round 5 (in particular, the input difference leading to the highest probability), the attacker can find input pairs such that (a) the input difference to round 5 conforms to the differential path (b) the roundkey differences to each round are as required. Furthermore, we assume that the complexity of finding such input pairs is a constant value independent of the differential path being used.

3.2 Example: Mendel and Schläffer Differential Path

The key observation in the Mendel and Schläffer [2] paper was a differential "fixed point" of the round function: that is, the output difference is identical to the input difference for this one-round differential path. The one-round differential path is traced in Table 1.

The one-round differential path can be applied to any bit position j. The inputs are assumed have bit differences at bit position j in C, E, G and no difference in bit position j of A. That is, $\Delta_i[j] = (-, *, *, *) \sim \hat{\Delta}_i[j] = 7$. There are assumed to be no differences in bit position j of the roundkeys. If these conditions hold, then the outputs have bit differences at bit position j in C, E, G and no differences in bit position j of A. The differential path proceeds as follows:

1. The value of C is first XORed with G, so the difference in G is eliminated resulting in $\Delta_a[j] = (-, *, *, -) \sim \hat{\Delta}_a[j] = 6$.
2. The roundkey is XORed with the state, which has no effect on the internal differences and $\Delta_b[j] = (-, *, *, -) \sim \hat{\Delta}_b[j] = 6$.
3. The S-box is applied to the values in $A_b[j], C_b[j], E_b[j]$. With probability 2^{-2}, the differences in $A_b[j], C_b[j], E_b[j]$ result in bit differences in $A_c[j]$ and $C_[j]$ only. This is a known differential property of the TIB3 S-box. The difference in G is not affected, and thus $\Delta_c[j] = (*, -, *, -) \sim \hat{\Delta}_c[j] = 10$.

Difference	Note	$\Delta A[j]$	$\Delta C[j]$	$\Delta E[j]$	$\Delta G[j]$	$\hat{\Delta}[j]$	Prob (log_2)	
							31,63	rest
$\Delta_{i-1}[j]$	(round input)	–	*	*	*	7		
$\Delta_a[j]$	(after XOR C with G)	–	*	*	–	6		
$\Delta_b[j]$	(after XOR with roundkey)	–	*	*	–	10	-2	-2
$\Delta_c[j]$	(after S-box)	*	–	*	–	10		
$\Delta_d[j]$	(after PHTX)	*	–	*	–	10		
$\Delta_e[j]$	(after addition)	*	–	*	*	11		-2
$\Delta_i[j]$	(after rotate = round output)	–	*	*	*	7		
Total Probability factor ($\log_2$)							-2	-4

Table 1. The Mendel and Schläffer one-round differential path, with output differences at bit position j and probabilities shown in log_2. The roundkeys are assumed to have no bit differences at bit position j.

4. The PHTX operation is applied to the values in C_c and G_c. There are no differences input to C_c and G_c, so there will be no differences in the output C_d and G_d. Hence, $\Delta_d[j] = \Delta_c[j]$.
5. The value of G_d is added to A_d and the value of C_d is added to the value of G_d. Recall that the linear model being applied to the $\tilde{+}$ operation. There are bit differences in $A_d[j]$ and $C_d[j]$ which will result in $\Delta_e[j] = (*, -, *, *) \sim \hat{\Delta}_e[j] = 11$. If the bit difference occurs at either the most significant bit of the upper or lower 32-bit halves (i.e. at bit positions 63 or 31), then there is no carry effect and the probability factor is $2^0 = 1$. Otherwise, there is a carry effect, and there is a probability factor of 2^{-1} for each modular addition, with a total probability factor of $(2^{-1})^2 = 2^{-2}$.
6. Finally, after rotating the position of the words, the resulting output difference is $\Delta_e[j] = (-, *, *, *)$.

Note there were no differences in the PHTX functions and the message blocks.

Mendel and Schläffer construct multi-round differential path by iterating the one-round differential path. The differential path from rounds 5 to 16 has probability 2^{-24} for bit positions 63 and 31. Using this characteristic Mendel and Schläffer constructed a pseudo-collision for the compression function with a complexity of about 2^{24}, which is significantly lower than of 2^{128} as expected for a compression function of 256 bits.

4 Differential Properties of the PHTX and Message Expansion

This section explains the differential properties of the PHTX and message expansion that are exploited in this analysis to find new differential paths with probability even higher than the differential path of Mendel and Schläffer [2]. Section 4.1 notes differential properties of the PHTX that hold with probability 1. Section 4.2 describes a differential property of the ψ function, while Section 4.3 explains how differences in the message blocks can be selected to prevent differences being introduced to the D_i parameters. Finally, Section 4.4 lists the possible sequence of round key differences resulting from the combination of these two observations.

4.1 PHTX Observations

The goal of the PHTX function in TIB3 is twofold: to create some non-linearity in the mixing process; and to diffuse information to other bit positions. However interesting things happen when differences are introduced to bit positions 63 or/and 31, that is, the most-significant bits of a 64-bit word broken into two 32-bit words. Table 2 shows the differential paths through the PHTX that hold with probability 1. For example, the first line means a bit change in the 31^{st} bit will result in a bit change in bit positions 20 and 63. *Note: this result can be easily extended to the PHTXD used in TIB3-512.*

ΔD			$\Delta\tilde{D}$			$\Delta PHTX(D)$		
63	31	20	63	31	20	63	31	20
–	–	–	–	–	–	–	–	–
–	*	–	*	*	–	*	–	*
*	–	–	*	–	–	*	*	*
*	*	–	–	*	–	–	*	–

Table 2. The differential paths through the PHTX that hold with probability 1.

Since these differential paths through the PHTX are restricted to bit positions 20, 31 and 63, the analysis hereafter will focus on differential paths in bit positions 20, 31 and 63.[2]

[2] We have recently noticed that if the input to the *PHTX* has differences in both bit positions 52 and 20, then with probability $\frac{1}{2}$, there is an output difference only in bit 20. A quick examination showed that this dramatically increased the number of possible one-round differential paths. We have not had time to explore this avenue further, but hope to examine the impact on TIB3v2 in the near future.

4.2 A Differential Property of the ψ Function

The inputs to the ψ function are denoted W, X, Y, Z. If W, X, Y, Z contain a bit difference only in the 31^{st} bit, that is, if:

$$\Delta W[j] = \Delta X[j] = \Delta Y[j] = \Delta Z[j] = \begin{cases} * & j = 31; \\ - & \text{otherwise}; \end{cases}$$

then the bit differences propagate in the following manner:

$$\begin{aligned}
\Delta V^1[63] &:= \Delta Y[63] \oplus Z[31] \oplus W[63] \oplus X[63] &&= - \oplus * \oplus - \oplus - &&= *;\\
\Delta V^1[31] &:= \Delta Y[31] \oplus \oplus W[31] \oplus X[31] \oplus Z[63] &&= * \oplus * \oplus * \oplus - &&= *;\\
\Delta V^2[63] &:= \Delta V^1[63] \oplus \Delta V^1[31] \oplus \Delta V^1[20] &&= * \oplus * \oplus - &&= -;\\
\Delta V^2[31] &:= \Delta V^1[31] &&&&= *;\\
\Delta V[63] &:= \Delta V^2[63] &&&&= -;\\
\Delta V[31] &:= \Delta V^1[31] &&&&= *.
\end{aligned}$$

That is, if we modify the 31^{st} bit of W, X, Y, Z then it causes $V = \psi(W, X, Y, Z)$ to have a bit difference in the 31^{st} bit. Thus, if we can create a bit difference in the 31^{st} bit of $D_0, \ldots, D_9$ we get this result for all D_i values. This can be accomplished by introducing differences in bit position 31 of all the LK_i's and having no bit differences in the RK_j's. This guarantees the sequence of roundkey differences shown in Table 3; we call this sequence a *message pattern*, in this case the message pattern applies for bit 31. This table uses the same notation for $\Delta K_i[j]$ as used to denote differences in bit position j of the internal state (e.g. $\Delta_{i-1}[j]$): if a zero replaces "−" (no difference at bit position j) and a 1 replaces a "*" (difference at bit position j), then $\hat{\Delta} K_i[j]$ is the decimal integer corresponding to the resulting binary value. The message pattern in Table 3 corresponds to message pattern 32 in Table 7.[3]

[3] Note for further research: if there is a bit difference in $W[j], X[j], Y[j]$ and $Z[j]$ for some j, $21 \leq j \leq 30$, then V might have a bit difference in the same index. The carry effect in the additions impose additional probability factors, so this avenue has not been explored any further in this paper. However, this could be an interesting avenue of future research, particularly since the message expansion in TIB3v2 is identical to the message expansion for TIB3. We also note that the message expansion for TIB3-512 has a similar property.

Rounds i	$\Delta K_i[31]$	$\hat{\Delta} K_i[31]$
1-8	$(*, *, *, *)$	15
9-12	$(-, *, -, *)$	5
13-16	$(*, *, *, *)$	15

Table 3. The message pattern resulting from changes in $LK_j[31]$, $0 \leq j \leq 7$, and no changes to RK.

4.3 Cancelation effect

For $0 \leq j \leq 31$, note that

$$V^1[j] = Y[j] \oplus W[j] \oplus X[j] \oplus Z[j + 32].$$

If $\Delta W[j] \oplus \Delta X[j] \oplus \Delta Y[j] \oplus \Delta Z[j + 32] = 0$, then the bit differences always cancel out, leaving no resulting output difference in $V = \psi(W, X, Y, Z)$.

For bit positions $32 \leq j \leq 63$,

$$V^1 = (Y + Z >> 32) \oplus W \oplus X.$$

and thus if $\Delta W[j] \oplus \Delta X[j] \oplus \Delta Y[j] \oplus \Delta Z[j - 32] = 0$, then it is possible that the bit differences will cancel and have no net difference in bits of V^1. It is also possible that the bit difference in $Y[j]$ and/or $Z[j - 32]$ can result in bit differences in position greater than j (the obvious exception is when $j = 63$). However, since the attacker has direct control over the messages, it is easy for the attacker to ensure that the differences in $Y[j]$ and/or $Z[j-32]$ always result in no difference after the modular addition, so this is not a concern. Hence, bit positions $32 \leq j \leq 63$ can be treated like bit positions $0 \leq j \leq 31$.

The possible combinations of differences are shown below.

	$\Delta W[j]$	$\Delta X[j]$	$\Delta Y[j]$	$\Delta Z[j \pm 32]$	Notes
1	$-$	$-$	$-$	$-$	The trivial case: no differences.
2	$*$	$*$	$-$	$-$	Bit differences in a single bit position.
3	$*$	$-$	$*$	$-$	Bit differences in a single bit position.
4	$-$	$*$	$*$	$-$	Bit differences in a single bit position.
5	$*$	$-$	$-$	$*$	Bit differences in two bit positions.
6	$-$	$*$	$-$	$*$	Bit differences in two bit positions.
7	$-$	$-$	$*$	$*$	Bit differences in two bit positions.
8	$*$	$*$	$*$	$*$	Bit differences in four bit positions.

This suggests that if bit differences are introduced to appropriate pairs of words in LK and RK, then the differences might cancel out when forming the values $D_0, \ldots, D_9$, and no differences will be introduced to message expansion values D_i. Some care must be taken in choosing the words in which to introduce differences: appropriate pairs of words of RK must be chosen to prevent introducing differences in D_8 and D_9.

Example 1. Table 4 shows the effect of modifying bits j of LK_0, LK_1, LK_4, LK_7, RK_0, RK_1, RK_4, RK_7. Note that these differences may occur in any bit position. This message pattern corresponds to message pattern 15 in Table 7. □

Round i	$\Delta K_i[j]$	$\hat{\Delta} K_i[j]$
1	$(-,*,-,*)$	5
2	$(-,*,-,*)$	5
3	$(-,-,-,-)$	0
4	$(-,-,-,-)$	0
5	$(-,*,-,*)$	5
6	$(-,-,-,-)$	0
7	$(-,-,-,-)$	0
8	$(-,*,-,*)$	5
9	$(*,-,-,*)$	10
10	$(-,-,-,-)$	0
11	$(*,-,-,-)$	8
12	$(-,-,*,-)$	2
13-16	$(-,-,-,-)$	0

Table 4. The message pattern resulting when the message blocks have differences at $LK_0[j]$, $LK_1[j]$, $LK_4[j]$, $LK_7[j]$, $RK_0[j]$, $RK_1[j]$, $RK_4[j]$ and $RK_7[j]$. These message patterns hold for all bit positions j.

Message blocks conforming to this type of message pattern must be a linear combination of the following *message pattern basis* shown in the table below. These message patterns hold for all j. Hence, the total dimension of this linear space is $6 \times 64 = 384$: that is, there are 2^{384} message patterns available using the cancelation effect.

Block	RK_0	RK_1	RK_2	RK_3	RK_4	RK_5	RK_6	RK_7
Bit	j	j	j	$j \pm 32$	j	j	$j \pm 32$	j
Differences	$*$	$*$	$-$	$-$	$-$	$-$	$-$	$-$
	$-$	$-$	$-$	$-$	$*$	$*$	$-$	$-$
	$*$	$-$	$*$	$-$	$-$	$-$	$-$	$-$
	$-$	$-$	$-$	$-$	$*$	$-$	$-$	$*$
	$*$	$-$	$-$	$*$	$-$	$-$	$-$	$-$
	$-$	$-$	$-$	$-$	$*$	$-$	$*$	$-$

Recall that this analysis is interested only in differential paths restricted to bit positions 20, 31 and 63. Some care must be taken in creating message patterns including bit position $j = 20$: only the first four basis vectors above can be used since the last two basis vectors will also involve bit differences in position $j + 32 = 52$. On the other hand, all basis vectors may be used for bit positions 31 and 63. The resulting basis for the message patterns under consideration are shown in Table 5. The total dimension of this restricted linear space is 16: that is, there are 2^{16} message patterns available using the cancelation effect.[4]

[4] Note: the message expansion for TIB3-512 has a similar set of message patterns.

Block		RK_0	RK_1	RK_2	RK_3	RK_4	RK_5	RK_6	RK_7
Bit		j	j	j	$j \pm 32$	j	j	$j \pm 32$	j
	20	$*$	$*$	$-$	$-$	$-$	$-$	$-$	$-$
	20	$-$	$-$	$-$	$-$	$*$	$*$	$-$	$-$
	20	$*$	$-$	$*$	$-$	$-$	$-$	$-$	$-$
	20	$-$	$-$	$-$	$-$	$*$	$-$	$-$	$*$
	31	$*$	$*$	$-$	$-$	$-$	$-$	$-$	$-$
	31	$-$	$-$	$-$	$-$	$*$	$*$	$-$	$-$
	31	$*$	$-$	$*$	$-$	$-$	$-$	$-$	$-$
j	31	$-$	$-$	$-$	$-$	$*$	$-$	$-$	$*$
	31	$*$	$-$	$-$	$*$	$-$	$-$	$-$	$-$
	31	$-$	$-$	$-$	$-$	$*$	$-$	$*$	$-$
	63	$*$	$*$	$-$	$-$	$-$	$-$	$-$	$-$
	63	$-$	$-$	$-$	$-$	$*$	$*$	$-$	$-$
	63	$*$	$-$	$*$	$-$	$-$	$-$	$-$	$-$
	63	$-$	$-$	$-$	$-$	$*$	$-$	$-$	$*$
	63	$*$	$-$	$-$	$*$	$-$	$-$	$-$	$-$
	63	$-$	$-$	$-$	$-$	$*$	$-$	$*$	$-$

Table 5. The basis for the message patterns possible, using the cancelation effect, that are under consideration in this paper. Note that these message patterns require $\Delta LK_i[j] = \Delta RK_i[j]$ for $0 \leq i \leq 7$.

4.4 List of Message Patterns

The ψ observation (Section 4.2) and the cancelation effect (Section 4.3) can be combined.

Example 2. Table 6 shows the message pattern when there are bit differences in bit position 13 of RK_0, LK_1, RK_2, LK_3, RK_4, LK_5, LK_6 and RK_7. The message pattern in Table 6 corresponds to message pattern 22 in Table 7. □

Round i	ΔK_i	$\hat{\Delta} K_i$
1	$(*, -, *, -)$	10
2	$(*, *, *, *)$	15
3	$(*, -, *, -)$	10
4	$(*, *, *, *)$	15
5	$(*, -, *, -)$	10
6	$(*, *, *, *)$	15
7	$(*, *, *, *)$	15
8	$(*, -, *, -)$	10
9	$(*, *, -, *)$	13
10	$(*, *, -, *)$	13
11	$(*, *, -, *)$	13
12	$(-, *, *, *)$	7
13-16	$(*, *, *, *)$	15

Table 6. The message pattern resulting when the message blocks have differences at $RK_0[31]$, $LK_1[31]$, $RK_2[31]$, $LK_3[31]$, $RK_4[31]$, $LK_5[31]$, $LK_6[31]$ and $RK_7[31]$.

The message pattern in Section 4.2 can added to the message pattern basis in Section 4.3. This results in a total of $2^{17} = 131072$ message patterns to consider. Due to the size of this set, it is not practical to list all possible message patterns. However it is possible to show some interesting sub-spaces.

These message patterns involving a single bit position are listed in Table 7. Message patterns $1 - 16$ hold with probability one for all bit positions, as they induce no bit differences in the D_i's so the first differential property of ψ does not need to be exploited. Message patterns $17 - 32$ hold with probability one only for bit position 31. [5]

[5] Note: Message patterns $17 - 32$ also applies to other bit positions where the ψ observation holds (bit positions 21-30). However, a probability factor must then be applied. These options have not been explored.

	Patterns for all bit positions		Extra patterns for bit positions $21-31$
1	0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0	17	10,10,15,15,10,10,15,15,15,5,15,5,15,15,15,15
2	0,0,0,0,0,5,0,5,0,0,2,2,0,0,0,0	18	10,10,15,15,10,15,15,10,15,5,13,7,15,15,15,15
3	0,0,0,0,5,0,0,5,0,0,8,2,0,0,0,0	19	10,10,15,15,15,10,15,10,15,5,7,7,15,15,15,15
4	0,0,0,0,5,5,0,0,0,0,10,0,0,0,0,0	20	10,10,15,15,15,15,15,15,15,5,5,5,15,15,15,15
5	0,5,5,0,0,0,0,0,2,8,0,0,0,0,0,0	21	10,15,10,15,10,10,15,15,13,13,15,5,15,15,15,15
6	0,5,5,0,0,5,0,5,2,8,2,2,0,0,0,0	22	10,15,10,15,10,15,15,10,13,13,13,7,15,15,15,15
7	0,5,5,0,5,0,0,5,2,8,8,2,0,0,0,0	23	10,15,10,15,15,10,15,10,13,13,7,7,15,15,15,15
8	0,5,5,0,5,5,0,0,2,8,10,0,0,0,0,0	24	10,15,10,15,15,15,15,15,13,13,5,5,15,15,15,15
9	5,0,5,0,0,0,0,0,8,8,0,0,0,0,0,0	25	15,10,10,15,10,10,15,15,7,13,15,5,15,15,15,15
10	5,0,5,0,0,5,0,5,8,8,2,2,0,0,0,0	26	15,10,10,15,10,15,15,10,7,13,13,7,15,15,15,15
11	5,0,5,0,5,0,0,5,8,8,8,2,0,0,0,0	27	15,10,10,15,15,10,15,10,7,13,7,7,15,15,15,15
12	5,0,5,0,5,5,0,0,8,8,10,0,0,0,0,0	28	15,10,10,15,15,15,15,15,7,13,5,5,15,15,15,15
13	5,5,0,0,0,0,0,0,10,0,0,0,0,0,0,0	29	15,15,15,15,10,10,15,15,5,5,15,5,15,15,15,15
14	5,5,0,0,0,5,0,5,10,0,2,2,0,0,0,0	30	15,15,15,15,10,15,15,10,5,5,13,7,15,15,15,15
15	5,5,0,0,5,0,0,5,10,0,8,2,0,0,0,0	31	15,15,15,15,15,10,15,10,5,5,7,7,15,15,15,15
16	5,5,0,0,5,5,0,0,10,0,10,0,0,0,0,0	32	15,15,15,15,15,15,15,15,5,5,5,5,15,15,15,15

Table 7. List of possible message patterns involving a single bit position. Message patterns 1 to 16 all bit positions. Message patterns 17 to 32 hold only for bit positions 21-31, and hold with probability one only for bit position 31.

Table 8 shows the set of message patterns orthogonal to the message patterns shown in Table 7. Those message patterns require differences in bit positions 31 and 63 to cancel in the ψ function. The set of possible message patterns involving only bit positions 63 and 31 can be obtained from linear combinations of

- the 16 message patterns 1-16 in Table 7 for bit position 63;
- the 32 message patterns 1-32 in Table 7 for bit position 31; and
- the 16 message patterns 33-48 in Table 8 involving both bit positions 63 and 31.

Aside from message pattern 1 (which was the pattern used by Mendel and Schläffer [1])), every other message pattern investigated in this analysis requires a difference in the input LK. The input LK corresponds to message block M_i: which is the current message block (while RK is the previous message block). This means that none of the new message patterns can be used as the last pair of messages block containing a difference. Consequently, the differential paths we have found are not sufficient to find a collision in TIB3.

Pattern	Round															
Number	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
33	$\frac{\cdot}{\cdot}$	$\frac{\cdot}{\cdot}$	$\frac{\cdot}{\cdot}$	$\frac{\cdot}{\cdot}$	$\frac{\cdot}{5}$	$\frac{\cdot}{\cdot}$	$\frac{5}{\cdot}$	$\frac{\cdot}{\cdot}$	$\frac{\cdot}{\cdot}$	$\frac{\cdot}{\cdot}$	$\frac{\cdot}{8}$	$\frac{8}{\cdot}$	$\frac{\cdot}{\cdot}$	$\frac{\cdot}{\cdot}$	$\frac{\cdot}{\cdot}$	$\frac{\cdot}{\cdot}$
34	$\frac{\cdot}{5}$	$\frac{\cdot}{\cdot}$	$\frac{\cdot}{\cdot}$	$\frac{5}{\cdot}$	$\frac{\cdot}{\cdot}$	$\frac{\cdot}{\cdot}$	$\frac{\cdot}{\cdot}$	$\frac{\cdot}{\cdot}$	$\frac{\cdot}{8}$	$\frac{2}{\cdot}$	$\frac{\cdot}{\cdot}$	$\frac{\cdot}{\cdot}$	$\frac{\cdot}{\cdot}$	$\frac{\cdot}{\cdot}$	$\frac{\cdot}{\cdot}$	$\frac{\cdot}{\cdot}$
35	$\frac{\cdot}{5}$	$\frac{\cdot}{\cdot}$	$\frac{\cdot}{\cdot}$	$\frac{5}{\cdot}$	$\frac{\cdot}{5}$	$\frac{\cdot}{\cdot}$	$\frac{5}{\cdot}$	$\frac{\cdot}{\cdot}$	$\frac{\cdot}{8}$	$\frac{2}{\cdot}$	$\frac{\cdot}{8}$	$\frac{8}{\cdot}$	$\frac{\cdot}{\cdot}$	$\frac{\cdot}{\cdot}$	$\frac{\cdot}{\cdot}$	$\frac{\cdot}{\cdot}$
36	$\frac{\cdot}{\cdot}$	$\frac{\cdot}{\cdot}$	$\frac{\cdot}{\cdot}$	$\frac{\cdot}{\cdot}$	$\frac{5}{\cdot}$	$\frac{\cdot}{\cdot}$	$\frac{\cdot}{5}$	$\frac{\cdot}{\cdot}$	$\frac{\cdot}{\cdot}$	$\frac{\cdot}{\cdot}$	$\frac{8}{\cdot}$	$\frac{\cdot}{8}$	$\frac{\cdot}{\cdot}$	$\frac{\cdot}{\cdot}$	$\frac{\cdot}{\cdot}$	$\frac{\cdot}{\cdot}$
37	$\frac{\cdot}{\cdot}$	$\frac{\cdot}{\cdot}$	$\frac{\cdot}{\cdot}$	$\frac{\cdot}{\cdot}$	$\frac{5}{5}$	$\frac{\cdot}{\cdot}$	$\frac{5}{5}$	$\frac{\cdot}{\cdot}$	$\frac{\cdot}{\cdot}$	$\frac{\cdot}{\cdot}$	$\frac{8}{8}$	$\frac{8}{8}$	$\frac{\cdot}{\cdot}$	$\frac{\cdot}{\cdot}$	$\frac{\cdot}{\cdot}$	$\frac{\cdot}{\cdot}$
38	$\frac{\cdot}{5}$	$\frac{\cdot}{\cdot}$	$\frac{\cdot}{\cdot}$	$\frac{5}{\cdot}$	$\frac{5}{\cdot}$	$\frac{\cdot}{\cdot}$	$\frac{\cdot}{5}$	$\frac{\cdot}{\cdot}$	$\frac{\cdot}{8}$	$\frac{2}{\cdot}$	$\frac{8}{\cdot}$	$\frac{\cdot}{8}$	$\frac{\cdot}{\cdot}$	$\frac{\cdot}{\cdot}$	$\frac{\cdot}{\cdot}$	$\frac{\cdot}{\cdot}$
39	$\frac{\cdot}{5}$	$\frac{\cdot}{\cdot}$	$\frac{\cdot}{\cdot}$	$\frac{5}{\cdot}$	$\frac{5}{5}$	$\frac{\cdot}{\cdot}$	$\frac{5}{5}$	$\frac{\cdot}{\cdot}$	$\frac{\cdot}{8}$	$\frac{2}{\cdot}$	$\frac{8}{8}$	$\frac{8}{8}$	$\frac{\cdot}{\cdot}$	$\frac{\cdot}{\cdot}$	$\frac{\cdot}{\cdot}$	$\frac{\cdot}{\cdot}$
40	$\frac{5}{\cdot}$	$\frac{\cdot}{\cdot}$	$\frac{\cdot}{\cdot}$	$\frac{\cdot}{5}$	$\frac{\cdot}{\cdot}$	$\frac{\cdot}{\cdot}$	$\frac{\cdot}{\cdot}$	$\frac{\cdot}{\cdot}$	$\frac{8}{\cdot}$	$\frac{\cdot}{2}$	$\frac{\cdot}{\cdot}$	$\frac{\cdot}{\cdot}$	$\frac{\cdot}{\cdot}$	$\frac{\cdot}{\cdot}$	$\frac{\cdot}{\cdot}$	$\frac{\cdot}{\cdot}$
41	$\frac{5}{\cdot}$	$\frac{\cdot}{\cdot}$	$\frac{\cdot}{\cdot}$	$\frac{\cdot}{5}$	$\frac{\cdot}{5}$	$\frac{\cdot}{\cdot}$	$\frac{5}{\cdot}$	$\frac{\cdot}{\cdot}$	$\frac{8}{\cdot}$	$\frac{\cdot}{2}$	$\frac{\cdot}{8}$	$\frac{8}{\cdot}$	$\frac{\cdot}{\cdot}$	$\frac{\cdot}{\cdot}$	$\frac{\cdot}{\cdot}$	$\frac{\cdot}{\cdot}$
42	$\frac{5}{5}$	$\frac{\cdot}{\cdot}$	$\frac{\cdot}{\cdot}$	$\frac{5}{5}$	$\frac{\cdot}{\cdot}$	$\frac{\cdot}{\cdot}$	$\frac{\cdot}{\cdot}$	$\frac{\cdot}{\cdot}$	$\frac{8}{8}$	$\frac{2}{2}$	$\frac{\cdot}{\cdot}$	$\frac{\cdot}{\cdot}$	$\frac{\cdot}{\cdot}$	$\frac{\cdot}{\cdot}$	$\frac{\cdot}{\cdot}$	$\frac{\cdot}{\cdot}$
43	$\frac{5}{5}$	$\frac{\cdot}{\cdot}$	$\frac{\cdot}{\cdot}$	$\frac{5}{5}$	$\frac{\cdot}{5}$	$\frac{\cdot}{\cdot}$	$\frac{5}{\cdot}$	$\frac{\cdot}{\cdot}$	$\frac{8}{8}$	$\frac{2}{2}$	$\frac{\cdot}{8}$	$\frac{8}{\cdot}$	$\frac{\cdot}{\cdot}$	$\frac{\cdot}{\cdot}$	$\frac{\cdot}{\cdot}$	$\frac{\cdot}{\cdot}$
44	$\frac{5}{\cdot}$	$\frac{\cdot}{\cdot}$	$\frac{\cdot}{\cdot}$	$\frac{\cdot}{5}$	$\frac{5}{\cdot}$	$\frac{\cdot}{\cdot}$	$\frac{\cdot}{5}$	$\frac{\cdot}{\cdot}$	$\frac{8}{\cdot}$	$\frac{\cdot}{2}$	$\frac{8}{\cdot}$	$\frac{\cdot}{8}$	$\frac{\cdot}{\cdot}$	$\frac{\cdot}{\cdot}$	$\frac{\cdot}{\cdot}$	$\frac{\cdot}{\cdot}$
45	$\frac{5}{\cdot}$	$\frac{\cdot}{\cdot}$	$\frac{\cdot}{\cdot}$	$\frac{\cdot}{5}$	$\frac{5}{5}$	$\frac{\cdot}{\cdot}$	$\frac{5}{5}$	$\frac{\cdot}{\cdot}$	$\frac{8}{\cdot}$	$\frac{\cdot}{2}$	$\frac{8}{8}$	$\frac{8}{8}$	$\frac{\cdot}{\cdot}$	$\frac{\cdot}{\cdot}$	$\frac{\cdot}{\cdot}$	$\frac{\cdot}{\cdot}$
46	$\frac{5}{5}$	$\frac{\cdot}{\cdot}$	$\frac{\cdot}{\cdot}$	$\frac{5}{5}$	$\frac{5}{\cdot}$	$\frac{\cdot}{\cdot}$	$\frac{\cdot}{5}$	$\frac{\cdot}{\cdot}$	$\frac{8}{8}$	$\frac{2}{2}$	$\frac{8}{\cdot}$	$\frac{\cdot}{8}$	$\frac{\cdot}{\cdot}$	$\frac{\cdot}{\cdot}$	$\frac{\cdot}{\cdot}$	$\frac{\cdot}{\cdot}$
47	$\frac{5}{5}$	$\frac{\cdot}{\cdot}$	$\frac{\cdot}{\cdot}$	$\frac{5}{5}$	$\frac{5}{5}$	$\frac{\cdot}{\cdot}$	$\frac{5}{5}$	$\frac{\cdot}{\cdot}$	$\frac{8}{8}$	$\frac{2}{2}$	$\frac{8}{8}$	$\frac{8}{8}$	$\frac{\cdot}{\cdot}$	$\frac{\cdot}{\cdot}$	$\frac{\cdot}{\cdot}$	$\frac{\cdot}{\cdot}$

Table 8. The orthogonal set of message patterns to the message patterns in Table 7. These message patterns require differences in both bit positions 31 and 63 in order for the differences to cancel in the ψ function. To save space, the differences in round i are shown as a fraction $\frac{\Delta K_i[63]}{\Delta K_i[31]}$, with the value "0" replaced by "·" to highlight the non-zero differences.

5 Constructing One-Round Differential Paths

This section explains the chosen algorithm for determining the set of one-round differential paths in a class.

5.1 Notation

The differential paths in this paper only considers cases where there are differences in bit positions 63, 31 and 20.[6] To make differential paths easier to read, the difference in these bit positions in a variable X is shown using a triple in square brackets: [$\delta X[63], \delta X[31], \delta X[20]$] where

$$\delta X[j] := \begin{cases} - & \text{if there is no difference in bit position } i \text{ of } X; \\ i & \text{if there is a difference in bit position } i \text{ of } X. \end{cases}$$

For example, $\Delta C_c = [63, 31, -]$ indicates that there are differences in bit positions 63 and 31, but no difference in bit position 20.

We define four cases for differential paths through the *C-PHTX*, corresponding to the four differential paths in Table 2:

Case 0**:** $\Delta C_c = [-, -, -] \rightarrow \Delta C_d = [-, -, -]$;
Case α**:** $\Delta C_c = [-, 31, -] \rightarrow \Delta C_d = [63, -, 20]$;
Case β**:** $\Delta C_c = [63, -, -] \rightarrow \Delta C_d = [63, 31, 20]$;
Case γ**:** $\Delta C_c = [63, 31, -] \rightarrow \Delta C_d = [-, 31, -]$;

The same four cases apply for differential paths through the *G-PHTX*.

5.2 Framework

The one-round differential paths are partitioned into *classes* according to which of the cases $0, \alpha, \beta, \delta$ is used for each of the *C-PHTX* operation and the *G-PHTX* operation. Since there are two PHTX operations, each with four cases considered, there are a total of $4 \times 4 = 16$ classes considered. The class using case x for the *C-PHTX* and case y for the *G-PHTX* is labeled $\langle x, y \rangle$. For example, Class $\langle 0, \gamma \rangle$ has $\Delta C_c = [-, -, -] \rightarrow \Delta C_d = [-, -, -]$ and $\Delta G_c = [63, 31, -] \rightarrow \Delta G_d = [-, 31, -]$. Note that when we specify a class, then we are specifying the values of ΔC_c, ΔC_d, ΔG_c and ΔG_d for all differential paths within that class.

Once a class is specified, and the corresponding values of ΔC_c, ΔC_d, ΔG_c and ΔG_d determined, there are additional internal differences that can be immediately inferred by tracing forwards or backwards for as long as no operations are

[6] Differential paths with bit differences in bits $a + 31$,a, $a - 12$ can be used for some other values of a, but these one-round differential paths lower probability and have not been examined.

applied to that variable. As shown in Table 9, the following internal differences can be inferred:

$$\Delta C_e := \Delta C_d;$$
$$\Delta A_i := \Delta C_e := \Delta C_d;$$
$$\Delta G_e := \Delta G_d;$$
$$\Delta G_b := \Delta G_c.$$

	A	C	E	G
Δ_{i-1}	?	?	?	?
Δ_a	?	?	?	?
Δ_b	?	?	?	ΔG_c
Δ_c	?	ΔC_c	?	ΔG_c
Δ_d	?	ΔC_d	?	ΔG_d
Δ_e	?	ΔC_d	?	ΔG_d
Δ_i	ΔC_d	?	?	?

Table 9. The internal differences that are specified for a given class. The "?" symbol represents internal differences that are not specified for a given class.

There are a range of differences possible for the other internal variables, each corresponding to a unique one-round differential path. This can be achieved by tracing the differences forwards from $\hat{\Delta}_d$ or backwards from $\hat{\Delta}_c$ We desire an efficient algorithm to search through the set of possible one-round differential paths. At first glance this may appear daunting, due to possible interactions between the bit positions. Fortunately for the cryptanalyst, the situation is improved by the use of the linear model which ignore the carries in the addition operation. A side effect is that, aside from interactions in the *PHTX*, the differences in the bit positions 63, 31 and 20 will not interact within a single round. Consequently, each of the bits can be analyzed independently when tracing the differences forwards from $\hat{\Delta}_d$ or backwards from $\hat{\Delta}_c$. Furthermore, the differences at all bit positions behave the same when tracing backwards or forwards, and it suffices to describe how to trace forwards and backwards for one bit position, and then that description applies for all bit positions. The exception is that bit differences in position 20 only interact linearly in the modular addition operation with probability 2^{-1}, so this probability factor must be accounted for in determining the final probability.

This motivates the following algorithm for determining the set of differential paths in a class:

One-Round Path Construction Algorithm
Inputs: ΔC_c, ΔC_d, ΔG_c, ΔG_d.

Step 1: First, choose values for $\Delta A_c = \Delta A_d$ and $\Delta E_c = \Delta E_d$. Each choice will result in fully specifying the internal state differences Δ_c and Δ_d. Steps 2 and 3 are then performed for each active bit position j.

Step 2: Trace the differences $\hat{\Delta}_d[j]$ forwards to derive the corresponding value for the output difference $\hat{\Delta}_i[j]$. This is addressed in more detail in Section 5.3.

Step 3: Trace the difference $\hat{\Delta}_c[j]$ backwards to derive possible values for the input difference $\hat{\Delta}_{i-1}[j]$.

- **Step 3a:** For a given $\hat{\Delta}_c[j]$, the choice of values for$\hat{\Delta}_b[j]$ is provided in Table 10. Table 10 is constructed from the TIB3 submission's difference table [3, p. 15].
- **Step 3b:** For a given $\hat{\Delta}K_i[j]$, and choice of $\hat{\Delta}_b[j]$, the value of $\hat{\Delta}_a[j]$ can be computed as $\hat{\Delta}_a[j] = \hat{\Delta}_c[j] \oplus \hat{\Delta}K_i[j]$. This follows from the specification.
- **Step 3c:** For a given $\hat{\Delta}_a[j]$, obtain $\hat{\Delta}_{i-1}[j]$ from Table 11. This table is easily derived from the specification. □

$\hat{\Delta}_c[j]$	$\hat{\Delta}_b[j]$	$\hat{\Delta}_c[j]$	$\hat{\Delta}_b[j]$	Pr. ($\log_2$)
0	0	1	1	0
2	4,6,12,14	3	3,7,11,15	-2
4	2,6,10,14	5	3,7,11,15	-2
6	2,4,10,12	7	3,5,11,13	-2
8	8,10,12,14	9	9,11,13,15	-2
10	4,6,8,10	11	5,7,9,11	-2
12	2,6,8,12	13	3,7,9,13	-2
14	2,4,8,14	15	3,5,9,15	-2

Table 10. The possible values of $\hat{\Delta}_b[j]$ that can generate the corresponding value of $\hat{\Delta}_c[j]$. The last column lists the logarithm (base 2) of the probability factor. This table is constructed from the TIB3 submission's difference table [3, p. 15].

A drawback of this approach is the number of combinations to consider. For each class, that there are 2^6 combinations of bit differences for $\hat{\Delta}A_c = \hat{\Delta}A_d$ and $\hat{\Delta}E_c = \hat{\Delta}E_d$ (since there are three bit positions: 63, 31 and 20). For each bit position, this number of combinations is (typically) multiplied by a factor of 4 in Step 3a, and a further factor corresponding to the number of possible key differences in Step 3b. The total number of combinations in all classes becomes

$\hat{\Delta}_a$	0	1	2	3	5	4	7	6	8	9	10	11	13	12	15	11
$\hat{\Delta}_{i-1}$	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15

Table 11. Tracing the difference backwards from $\hat{\Delta}_a[j]$ to $\hat{\Delta}_{i-1}$. This relationship holds with probability one for differences in all bit positions.

unmanageable. Rather than construct each path individually, it is possible to describe the construction of sets of paths simultaneously. Recall that in a given class, the values of $\hat{\Delta}C_c$, $\hat{\Delta}C_d$, $\hat{\Delta}G_c$ and $\hat{\Delta}G_d$ do not change for all differential paths within that class. Suppose we partition the set of internal state differences at a particular bit position into the following sets:

$$
\begin{aligned}
S^0 &:= \{(x, -, y, -) : x, y \in \{-, *\}\} \sim \{0, 2, 8, 10\};\\
S^1 &:= \{(x, -, y, *) : x, y \in \{-, *\}\} \sim \{1, 3, 9, 11\};\\
S^2 &:= \{(x, *, y, -) : x, y \in \{-, *\}\} \sim \{4, 6, 12, 14\};\\
S^3 &:= \{(x, *, y, *) : x, y \in \{-, *\}\} \sim \{5, 7, 13, 15\}.
\end{aligned}
$$

Notice that the set of differences $\hat{\Delta}_c[j]$ such that $\hat{\Delta}C_c[j] = 0$ and $\hat{\Delta}G_c[j] = 0$ is exactly the set S^0.

With this partitioning, we know that for a given class, there is a choice of set $S_{c,j} \in \{S^0, S^1, S^2, S^3\}$ and a choice of $S_{d,j} \in \{S^0, S^1, S^2, S^3\}$ for each $j \in \{63, 31, 20\}$ such that $\hat{\Delta}_c[j] \in S_{c,j}$ and $\hat{\Delta}_d[j] \in S_{d,j}$ for all paths in that class. Table 12 shows the indices for the sets that apply to each bit 63, 31 and 20 for each class.

C-PHTX Case	0				α				β				γ			
G-PHTX Case	0	α	β	γ	0	α	β	γ	0	α	β	γ	0	α	β	γ
$\hat{\Delta}_c[63]$	0	0	1	1	0	0	1	1	2	2	3	3	2	2	3	3
$\hat{\Delta}_d[63]$	0	1	1	0	2	3	3	2	2	3	3	2	0	1	1	0
$\hat{\Delta}_c[31]$	0	1	0	1	2	3	2	3	0	1	0	1	2	3	2	3
$\hat{\Delta}_d[31]$	0	0	1	1	0	0	1	1	2	2	3	3	2	2	3	3
$\hat{\Delta}_c[20]$	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
$\hat{\Delta}_d[20]$	0	1	1	0	2	3	3	2	2	3	3	2	0	1	1	0

Table 12. The index of the sets of values for $\hat{\Delta}_c$ and $\hat{\Delta}_d$ for each of the classes.

Using Table 12: For a particular class and a particular bit position j, suppose that Table 12 indicates that the set index for $\hat{\Delta}_c[j]$ is 2 and the set index for $\hat{\Delta}_d[j]$ is 3: this indicates that $\hat{\Delta}_c[j] \in S^2$ and $\hat{\Delta}_d[j] \in S^3$. This means that $\hat{\Delta}_c[j] = (w, *, x, -)$ for some choice of w and x, and $\hat{\Delta}_d[j] \sim (y, *, z, *)$ for some choice of y and z. Note that $\Delta A_c = \Delta A_d$ and $\Delta E_c = \Delta E_d$, since the $PHTX$ is only applied to the values of C and G. This implies that for all differential paths, $w = y$ and $x = z$.

This gives another restriction to use in constructing one-round differential paths: Suppose S^x_z represent the z-th value in the set S^x, starting with index 0; for example, $S^1_3 = 11$. For a particular class and a particular bit position j, suppose that Table 12 indicates that the set index for $\hat{\Delta}_c[j]$ is 2 and the set index for $\hat{\Delta}_d[j]$ is 3, Then, whenever our analysis decides to use $\hat{\Delta}_c[j] = S^x_z$, then the corresponding value of $\hat{\Delta}_d[j]$ must be assigned to $\hat{\Delta}_d[j] = S^y_z$: that is, $\hat{\Delta}_d[j]$ corresponds to the value in the same position of S^y as the position of the input difference in the set S^x.

Example 3. Suppose the analysis is investigating paths in the class $\langle \alpha, \beta \rangle$. For this class, Table 12 indicates that

$$\begin{aligned}
&\hat{\Delta}_c[63] \in S^1 = \{1, 3, 9, 11\}, \;\; \hat{\Delta}_d[63] \in S^3 = \{5, 7, 13, 15\};\\
&\hat{\Delta}_c[31] \in S^2 = \{4, 6, 12, 14\}, \hat{\Delta}_d[31] \in S^1 = \{1, 3, 9, 11\};\\
&\hat{\Delta}_c[20] \in S^0 = \{0, 2, 8, 10\}, \;\; \hat{\Delta}_d[20] \in S^3 = \{5, 7, 13, 15\}.
\end{aligned}$$

Suppose a differential path chosen from this set has

$$\begin{aligned}
\hat{\Delta}_c[63] &= 3 \;\;\, \sim (-, -, *, *);\\
\hat{\Delta}_c[31] &= 12 \sim (*, *, -, -);\\
\hat{\Delta}_c[20] &= 0 \;\;\, \sim (-, -, -, -);
\end{aligned}$$

corresponding to the 2nd, 3rd and 1st value in the sets for $\hat{\Delta}_c[63]$, $\hat{\Delta}_c[31]$ and $\hat{\Delta}_c[20]$ respectively. The corresponding difference $\hat{\Delta}_d$ must use the 2nd, 3rd and 1st value in the sets for $\hat{\Delta}_d[63]$, $\hat{\Delta}_d[31]$ and $\hat{\Delta}_d[20]$ (that is, the 2nd, 3rd and 1st value in the sets S^3, S^1 and S^3 respectively). That is,

$$\begin{aligned}
\hat{\Delta}_c[63] &= 7 \sim (-, *, *, *);\\
\hat{\Delta}_c[31] &= 9 \sim (*, -, -, *);\\
\hat{\Delta}_c[20] &= 5 \sim (-, *, -, *). \qquad \square
\end{aligned}$$

5.3 Tracing Forward

Recall that, as part of our analysis, we use the linear model which assumes that the differences pass through the addition operation as though it were an XOR operation and ensure that we apply a suitable probability factor. Table 13 traces the differences forward from each of the possible differences in $\hat{\Delta}_d[j]$ to the corresponding difference in $\hat{\Delta}_i[j]$. When $j \in \{63, 31\}$, then the bit differences propagate as predicted with probability 1. This is because there is no carry bit up from bit positions 63 and 31 in the $\tilde{+}$ operation. However, when $j \notin \{63, 31\}$, then there is the chance of the bit difference propagating to more significant bits, resulting in the differential not behaving as specified. To account for this, a probability factors are introduced for bit position $j \notin \{63, 31\}$:

- Whenever there is a bit difference in $\hat{\Delta}A_d[j]$ and/or $\hat{\Delta}C_d[j]$, then an additional probability factor of 2^{-1} is incurred due to the addition $A_e := A_d \tilde{+} G_d$.
- Whenever there is a bit difference in $\hat{\Delta}C_d[j]$ and/or $\hat{\Delta}G_d[j]$, then an additional probability factor of 2^{-1} is incurred due to the addition $G_e := G_d \tilde{+} C_d$.

These probabilities are reflected in Table 13. Using this data, Table 14 generates the possible outputs (and corresponding probability factors for each class.

Sets	S^0				S^1				S^2				S^3			
$\hat{\Delta}_d[j]$	0	2	8	10	1	3	9	11	4	6	12	14	5	7	13	15
$\hat{\Delta}_i[j]$	0	6	1	7	3	5	2	4	8	14	9	15	11	13	10	12
Pr. ($\log_2$)	0	-1	-1	-2	-2	-2	-2	-2	0	-1	-1	-2	-2	-2	-2	-2

Table 13. Tracing the differences forward from differences in $\hat{\Delta}_d[j]$ to differences in $\hat{\Delta}_i[j]$. The probability factor (provided logarithm base 2) applies only for bit positions $j \notin \{63, 31\}$.

5.4 Tracing Backward

The first two steps in the TIB3 round function are:

$$(A_a,\ C_a,\ E_a,\ G_a) := (A_{i-1},\ C_{i-1},\ E_{i-1},\ G_{i-1} \oplus C_{i-1});$$
$$(A_b,\ C_b,\ E_b,\ G_b) := (A_a,\ C_a,\ E_{a1},\ G_a) \oplus (KA_i,\ KC_i,\ KE_i,\ KG_i).$$

An alternative description of the TIB3 round function can be obtained by reversing the order of these operations - provided an appropriate modification to the roundkey is performed:

$$K_i^L = (KA_i^L,\ KC_i^L,\ KE_i^L,\ KG_i^L) := (KA_i,\ KC_i,\ KE_i,\ KG_i \oplus\ KC_i);$$
$$(A_b^L,\ C_b^L,\ E_b^L,\ G_b^L) := (A_{i-1},\ C_{i-1},\ E_{i-1},\ G_{i-1}) \oplus K_i^L;$$
$$(A_b,\ C_b,\ E_b,\ G_b) := (A_b^L,\ C_b^L,\ E_b^L,\ G_b^L \oplus C_b^L).$$

Similarly, we can reverse steps 3b and 3c of the initial algorithm (Section 5.2) Recall that steps 3b and 3c are:

Step 3b: For a given $\hat{\Delta}K_i[j]$, and choice of $\hat{\Delta}_b[j]$, the value of $\hat{\Delta}_a[j]$ can be computed as $\hat{\Delta}_a[j] = \hat{\Delta}_c[j] \oplus \hat{\Delta}K_i[j]$. This follows from the specification.

Class	$\langle 0,0\rangle$				$\langle 0,\alpha\rangle$				$\langle 0,\beta\rangle$				$\langle 0,\gamma\rangle$			
$\Delta_i[63,20]$	0	6	1	7	3	5	2	4	3	5	2	4	0	6	1	7
$\Delta_i[31]$	0	6	1	7	0	6	1	7	3	5	2	4	3	5	2	4
Class	$\langle \alpha,0\rangle$				$\langle \alpha,\alpha\rangle$				$\langle \alpha,\beta\rangle$				$\langle \alpha,\gamma\rangle$			
$\Delta_i[63,20]$	8	14	9	15	11	13	10	12	11	13	10	12	8	14	9	15
$\Delta_i[31]$	0	6	1	7	0	6	1	7	3	5	2	4	3	5	2	4
Class	$\langle \beta,0\rangle$				$\langle \beta,\alpha\rangle$				$\langle \beta,\beta\rangle$				$\langle \beta,\gamma\rangle$			
$\Delta_i[63,20]$	8	14	9	15	11	13	10	12	11	13	10	12	8	14	9	15
$\Delta_i[31]$	8	14	9	15	8	14	9	15	11	13	10	12	11	13	10	12
Class	$\langle \gamma,0\rangle$				$\langle \gamma,\alpha\rangle$				$\langle \gamma,\beta\rangle$				$\langle \gamma,\gamma\rangle$			
$\Delta_i[63,20]$	0	6	1	7	3	5	2	4	3	5	2	4	0	6	1	7
$\Delta_i[31]$	8	14	9	15	8	14	9	15	11	13	10	12	11	13	10	12
Pr. ($\log_2$)	0	-1	-1	-2	-2	-2	-2	-2	-2	-2	-2	-2	0	-1	-1	-2

Table 14. The possible outputs Δ_i for each of the classes. The probability factors are provided (logarithm base 2) in the final row. Note that the probability is related to $\Delta_d[20]$ and is the same for all values in a column.

Step 3c: For a given $\hat{\Delta}_a[j]$, obtain $\hat{\Delta}_{i-1}[j]$ from Table 11. This table is easily derived from the specification.

The alternative description of TIB3 round function allows us to replace steps 3b and 3c with alternative steps 3b', 3b' and 3c'.

Step 3b': For a choice of $\hat{\Delta}_b[j]$, compute $\hat{\Delta}^L_b[j]$ using Table 11.
Step 3c': For a choice of $\hat{\Delta}K_i[j]$, compute $\hat{\Delta}K^L_i[j]$ using Table 11.
Step 3d': For a choice of $\hat{\Delta}^L_b[j]$ and $\hat{\Delta}K^L_i[j]$, compute $\hat{\Delta}_i = \hat{\Delta}^L_b[j] \oplus \hat{\Delta}K^L_i[j]$.

Steps 3b', 3c' and 3d' can be applied for each bit position independently. These alternative steps have been chosen over Steps 3b and 3c because it is possible to show more paths in less space.

Table 15 (Table 16) uses these steps to trace the differences backwards from $\hat{\Delta}_c[j]$ in S^0 and S^1 ($\hat{\Delta}_c[j]$ in S^2 and S^3 respectively). The corresponding probability factors are also shown in this table. Note that the differential paths that incur a probability factor of $2^0 = 1$ when $\hat{\Delta}_c[j] = 0$ or $\hat{\Delta}_c[j] = 1$, corresponding to differential paths that avoid the S-box. Otherwise, all other differential paths incur an equal probability factor of 2^{-2}. This suggests that the highest probability differential paths will be those differential paths that maximize the number of bit positions $j \in \{63, 31, 20\}$ for which $\hat{\Delta}_c[j] = 0$, or $\hat{\Delta}_c[j] = 1$.

This concludes the description of the tools used to construct one-round differential paths. The next section investigates the differential paths with only one *active* bit: that is, differential paths with non-zero differences in only one bit

1. $\hat{\Delta}_c[j]$		S^0				S^1			
		0	2	8	10	1	3	9	11
3a. $\hat{\Delta}_b[j]$		0	4,6,12,14	8,10,12,14	4,6,8,10	1	3,7,11,15	9,11,13,15	5,7,9,11
3b'. $\hat{\Delta}_b^L[j]$		0	5,7,13,15	8,10,13,15	5,7,8,10	1	5,6,11,14	9,11,12,14	4,6,9,11
$\hat{\Delta}K_i[j]$	$\hat{\Delta}K_i^L[j]$	$\hat{\Delta}_{i-1}[j] = \hat{\Delta}_b^L[j] \oplus \hat{\Delta}K_i^L[j]$							
	3c'	3d'							
0	0	0	5,7,13,15	8,10,13,15	5,7,8,10	1	5,6,11,14	9,11,12,14	4,6,9,11
2	2	2	7,5,15,13	10,8,15,13	7,5,10,8	3	7,4,9,12	11,9,14,12	6,4,11,9
5	4	4	1,3,9,11	12,14,9,11	1,3,12,14	5	1,2,15,10	13,15,8,10	0,2,13,15
7	6	6	3,1,11,9	14,12,11,9	3,1,14,12	7	3,0,5,8	15,13,10,8	2,0,15,13
8	8	8	13,15,5,7	0,2,5,7	13,15,0,2	9	13,14,3,6	1,3,4,6	12,14,1,3
10	10	10	15,13,7,5	2,0,7,5	15,13,2,0	11	11,12,1,4	3,1,6,4	14,12,3,1
13	12	12	9,11,1,3	4,6,1,3	9,11,4,6	13	9,10,7,2	5,7,0,2	8,10,5,7
15	14	14	11,9,3,1	6,4,3,1	11,9,6,4	15	11,8,5,0	7,5,2,0	10,8,7,5
Pr. ($\log_2$)		0	-2	-2	-2	0	-2	-2	-2

Table 15. Steps 3a, 3b', 3c' and 3d' for tracing differences in $\hat{\Delta}_c \in S^0 \cup S^1$ backwards to the input differences $\hat{\Delta}_{i-1}$ and the corresponding probability factors. Note that the entries in the last the heading 3d' correspond the possible input differences $\hat{\Delta}_{i-1}$.

position. Section 7 considers differential paths with two active bits and Section 8 considers differential paths with three active bits.

1. $\hat{\Delta}_c$		S^2				S^3			
		4	6	12	14	5	7	13	15
3a. $\hat{\Delta}_b$		2,6,10,14	2,4,10,12	2,6,8,12	2,4,8,14	3,7,11,15	3,5,11,13	3,7,9,13	3,5,9,15
3b'. $\hat{\Delta}_b^L$		2,7,10,15	2,5,10,13	2,7,8,13	2,5,8,15	3,6,11,14	3,4,11,12	3,6,9,12	3,4,9,14
$\hat{\Delta}K_i$	$\hat{\Delta}K_i^L$ 3c'	$\hat{\Delta}_{i-1} = \hat{\Delta}_b^L \oplus \hat{\Delta}K_i^L$ 3d'							
0	0	2,7,10,15	2,5,10,13	2,7,8,13	2,5,8,15	3,6,11,14	3,4,11,12	3,6,9,12	3,4,9,14
2	2	0,5,8,13	0,7,8,15	0,5,10,15	0,7,10,13	0,5,8,13	0,7,8,15	0,5,10,15	0,7,10,13
5	4	6,3,14,11	6,1,14,9	6,3,12,9	6,1,12,11	6,3,14,11	6,1,14,9	6,3,12,9	6,1,12,11
7	6	4,1,12,9	4,3,12,11	4,1,14,11	4,3,14,9	4,1,12,9	4,3,12,11	4,1,14,11	4,3,14,9
8	8	10,15,2,7	10,13,2,5	10,15,0,5	10,13,0,7	10,15,2,7	10,13,2,5	10,15,0,5	10,13,0,7
10	10	8,13,0,5	8,15,0,7	8,13,2,7	8,15,2,5	8,13,0,5	8,15,0,7	8,13,2,7	8,15,2,5
13	12	14,11,6,3	14,9,6,1	14,11,4,1	14,9,4,3	14,11,6,3	14,9,6,1	14,11,4,1	14,9,4,3
15	14	12,9,4,1	12,11,4,3	12,9,6,3	12,11,6,1	12,9,4,1	12,11,4,3	12,9,6,3	12,11,6,1
Pr. ($\log_2$)		-2	-2	-2	-2	-2	-2	-2	-2

Table 16. Steps 3a, 3b', 3c' and 3d' for tracing differences in $\hat{\Delta}_c$ backwards.

6 Differential Paths with One Active Bit Position

This section investigates differential paths with one active bit position. Note that differential paths through the *PHTX* always introduce interactions between multiple bit positions, so differential paths with one active bit position must avoid the *PHTX*. Hence, only class $\langle 0, 0\rangle$ allows differential paths with one active bit. This class only allows internal differences $\hat{\Delta}_c[j] = \hat{\Delta}_d[j]$ to be chosen from the set $S^0 = \{0, 2, 8, 10\}$.

6.1 One-round Differential Paths with One Active Bit Position

Table 17 shows all the steps in tracing the differential paths forwards and backwards from the restricted set of state differences with $\hat{\Delta}_c[j] = \hat{\Delta}_d[j] \in S^0$.

1. $\hat{\Delta}_c[j]$		S^0			
		0	2	8	10
3a. $\hat{\Delta}_b[j]$		0	4,6,12,14	8,10,12,14	4,6,8,10
3b'. $\hat{\Delta}_b^L[j]$		0	5,7,13,15	8,10,13,15	5,7,8,10
$\hat{\Delta}K_i[j]$	$\hat{\Delta}K_i^L[j]$	$\hat{\Delta}_{i-1}[j] = \hat{\Delta}_b^L[j] \oplus \hat{\Delta}K_i^L$[j]			
	3c'	3d'			
0	0	0	5,7,13,15	8,10,13,15	5,7,8,10
2	2	2	7,5,15,13	10,8,15,13	7,5,10,8
5	4	4	1,3,9,11	12,14,9,11	1,3,12,14
7	6	6	3,1,11,9	14,12,11,9	3,1,14,12
8	8	8	13,15,5,7	0,2,5,7	13,15,0,2
10	10	10	15,13,7,5	2,0,7,5	15,13,2,0
13	12	12	9,11,1,3	4,6,1,3	9,11,4,6
15	14	14	11,9,3,1	6,4,3,1	11,9,6,4
$\hat{\Delta}_d[j] \in S^0$		0	2	8	10
$\hat{\Delta}_i[j]$		0	6	1	7
Pr. ($\log_2$)		0	-2	-2	-2

Table 17. Tracing differences in $\hat{\Delta}_c[j] \in S^0$ for Class $\langle 0, 0\rangle$ and the corresponding probability factors.

Multi-round differential paths are constructed by chaining one-round differential paths and ensuring that the output difference of one round is equal to the input difference for the next round. Table 17 is useful for seeing how the differential paths are constructed, but it is difficult to use this representation for constructing multi-round paths. A more useful representation of the one-round differential paths is for a row to indicate a choice of input difference and a column to indicate a choice of output differences. The entry where this row and column meet will indicate which round key differences may be used to allow the input difference to result in the output difference. We call this a "input-output-roundkey" representation. The "input-output-roundkey" representation for the class $\langle 0, 0\rangle$ is provided in Table 18. Note that the only output differences possible in this class are 0, 1, 6 and 7. In a multi-round differential path, the input differences (to all but the first round) are restricted to the set of possible output differences. This means that the input differences 0, 1, 6 and 7 are particularly important. In light of this, these input differences are shown at the top of Table 18.

	$\hat{\Delta}_i[j]$ possible for classes $\langle 0, 0\rangle$			
	0	1	6	7
$\hat{\Delta}_{i-1[j]}$	$\hat{\Delta}K_i[j]$ such that $\hat{\Delta}_{i-1}[j] \overset{\hat{\Delta}K_i[j]}{\rightarrow} \hat{\Delta}_i[j]$			
0	0	8,10		8,10
1 or 3		13,15	5,7,13,15	5,7
6	7	13,15		13,15
5 or 7		8,10	0,2,8,10	0,2
2	2	8,10		8,10
4	5	13,15		13,15
8	8	0,2		0,2
9 or 11		5,7	5,7,13,15	13,15
10	10	0,2		0,2
12	13	5,7		5,7
13 or 15		0,2	0,2,8,10	8,10
14	15	5,7		5,7
Pr. ($\log_2$) $j = 63, 31$	0	-2	-2	-2
Pr. ($\log_2$) $j \neq 63, 31$	0	-3	-3	-4

Table 18. One-round differential paths with one active bit. The rows correspond to input differences (at the active bit position), the columns correspond to output differences and the table entries list the key differences that allow this differential to occur. The probability factor (provided logarithm base 2) is divided into the case $j = 63, 31$ and the cases $j \neq 63, 31$. The input differences corresponding to possible output differences are put at the top of the table.

6.2 Multi-Round Differential Paths with One Active Bit Position

The focus now shifts to the construction of multi-round differential paths with one active bit position. The possible one-round differential paths are represented in Table 18. These differential paths can now be chained using the possible message patterns in Table 7. A process of elimination shows that 16-round differential paths can be constructed for only 4 of these message patterns.

Firstly note the output of a given single round is an element of the following set $N = \{0, 1, 6, 7\}$. Hence, for round 2 onwards, we can ignore all the other input differences in Table 18. Note that input difference 1 goes to output difference 6 or 7 using roundkey difference 5 and this is the only input difference from the set N that can use a roundkey difference of 5. Therefore no path can have two adjacent "middle" rounds with roundkey difference 5. This immediately shows that message patterns $4, 5, 6, 7, 8, 12, 16, 19, 20, 23, 24, 27, 28, 29, 30, 31$ and 32 are impossible with one active bit position.

Making a similar observation with roundkey difference 8, we see input difference 0 goes to output difference 1 or 7 using a roundkey difference 8. Also input difference 7 goes to output difference 1 or 6 using a roundkey difference 8. So the only possibility for two adjacent "middle" rounds with roundkey difference 8 is $0 \xrightarrow{8} 7 \xrightarrow{8} 1/6$. However neither of the input differences 1 nor 6 can use a roundkey difference of 2. Thus the roundkey difference pattern $8 - 8 - 2$ is not possible. This eliminates message patterns $9, 10, 11$.

Similarly, the only possible output difference from two "middle" rounds with roundkey difference 2 is either a 6 or a 7. However neither of the input differences of 6 nor 7 can use a roundkey difference of 0. Therefore the roundkey difference pattern $2 - 2 - 0$ is not possible and this eliminates message patterns $2, 14$.

Note that an input difference of $0, 1, 6$ cannot use a roundkey difference of 2. Thus, a roundkey difference of 2 must use an input difference of 7. If a one-round differential path uses a roundkey difference of 8 and produces an output difference of 7, then the input difference must have been 0. Hence, the only way to have a $8 - 2$ section of the roundkey difference is to start with a 0. Now, the only one-round differential paths using a roundkey difference of 0 and ending with an output difference of 0 have an input difference of 0. However, there is no input difference in the set N that uses a roundkey difference of 5 or 10 and ends with a roundkey difference of 0. This shows that roundkey difference pattern $5/10 - 0 - \ldots - 0 - 8 - 2$ is not possible, eliminating message patterns 3 and 15.

Considering all the input/output pairs using roundkey difference 5, we see that a two consecutive roundkey differences of 5 in the beginning must end on either a 6/7. Now, with a string of 0 roundkey differences, the possible output differences are again either 6 or 7. After using a roundkey difference of 10 the output differences are either 1 or 6. Neither input difference 1 nor 6 can use a message key of 0, which proves why message pattern 13 isn't possible.

Looking at the sequence of roundkey differences $15 - 10$ (that is 15 followed by a 10) we see that this portion of the differential path must start with an input difference of 6 going to output difference of 7 then the input difference of 7 going to output difference of either 1 or 6. Neither 1 nor 6 can use a roundkey difference of 10. Thus the pattern $15 - 10 - 10$ is not possible, getting rid of message patterns $17, 21, 25$.

After this process of elimination, only four possible message patterns remain: 1, 18, 22 and 26. Of these message patterns, pattern 1 can be applied for any bit position, while message patterns 18, 22 and 26 can be applied only for bit position 31. Example differential paths (there are other) are shown in Table 19.

Message	Rounds								Prob ($\log_2$)	
Pattern	1/9	2/10	3/11	4/12	5/13	6/14	7/15	8/16	63,31	Other
1	$7 \xrightarrow{\frac{0,-2}{\langle 0,0\rangle}} 7$	$\xrightarrow{\frac{0,-2}{\langle 0,0\rangle}} 7$	$\xrightarrow{\frac{0,-2}{\langle 0,0\rangle}} 7$	$\xrightarrow{\frac{0,-2}{\langle 0,0\rangle}} 7$	$\xrightarrow{\frac{0,-2}{\langle 0,0\rangle}} 7$	$\xrightarrow{\frac{0,-2}{\langle 0,0\rangle}} 7$	$\xrightarrow{\frac{0,-2}{\langle 0,0\rangle}} 7$	$\xrightarrow{\frac{0,-2}{\langle 0,0\rangle}} 7$	-24	-48
	$7 \xrightarrow{\frac{0,-2}{\langle 0,0\rangle}} 7$	$\xrightarrow{\frac{0,-2}{\langle 0,0\rangle}} 7$	$\xrightarrow{\frac{0,-2}{\langle 0,0\rangle}} 7$	$\xrightarrow{\frac{0,-2}{\langle 0,0\rangle}} 7$	$\xrightarrow{\frac{0,-2}{\langle 0,0\rangle}} 7$	$\xrightarrow{\frac{0,-2}{\langle 0,0\rangle}} 7$	$\xrightarrow{\frac{0,-2}{\langle 0,0\rangle}} 7$	$\xrightarrow{\frac{0,-2}{\langle 0,0\rangle}} 7$		
18	$0 \xrightarrow{\frac{10,-2}{\langle 0,0\rangle}} 1$	$\xrightarrow{\frac{10,-2}{\langle 0,0\rangle}} 1$	$\xrightarrow{\frac{15,-2}{\langle 0,0\rangle}} 6$	$\xrightarrow{\frac{15,-2}{\langle 0,0\rangle}} 7$	$\xrightarrow{\frac{10,-2}{\langle 0,0\rangle}} 1$	$\xrightarrow{\frac{15,-2}{\langle 0,0\rangle}} 6$	$\xrightarrow{\frac{15,-2}{\langle 0,0\rangle}} 7$	$\xrightarrow{\frac{15,-2}{\langle 0,0\rangle}} 6$	-24	n/a
	$6 \xrightarrow{\frac{15,-2}{\langle 0,0\rangle}} 1$	$\xrightarrow{\frac{5,-2}{\langle 0,0\rangle}} 6$	$\xrightarrow{\frac{13,-2}{\langle 0,0\rangle}} 1$	$\xrightarrow{\frac{7,-2}{\langle 0,0\rangle}} 6$	$\xrightarrow{\frac{15,-2}{\langle 0,0\rangle}} 1$	$\xrightarrow{\frac{15,-2}{\langle 0,0\rangle}} 1$	$\xrightarrow{\frac{15,-2}{\langle 0,0\rangle}} 1$	$\xrightarrow{\frac{15,-2}{\langle 0,0\rangle}} 1$		
22	$7 \xrightarrow{\frac{10,-2}{\langle 0,0\rangle}} 6$	$\xrightarrow{\frac{15,-2}{\langle 0,0\rangle}} 7$	$\xrightarrow{\frac{10,-2}{\langle 0,0\rangle}} 6$	$\xrightarrow{\frac{15,-2}{\langle 0,0\rangle}} 7$	$\xrightarrow{\frac{10,-2}{\langle 0,0\rangle}} 1$	$\xrightarrow{\frac{15,-2}{\langle 0,0\rangle}} 6$	$\xrightarrow{\frac{15,-2}{\langle 0,0\rangle}} 7$	$\xrightarrow{\frac{10,-2}{\langle 0,0\rangle}} 1$	-24	n/a
	$1 \xrightarrow{\frac{13,-2}{\langle 0,0\rangle}} 1$	$\xrightarrow{\frac{13,-2}{\langle 0,0\rangle}} 6$	$\xrightarrow{\frac{13,-2}{\langle 0,0\rangle}} 1$	$\xrightarrow{\frac{7,-2}{\langle 0,0\rangle}} 6$	$\xrightarrow{\frac{15,-2}{\langle 0,0\rangle}} 1$	$\xrightarrow{\frac{15,-2}{\langle 0,0\rangle}} 1$	$\xrightarrow{\frac{15,-2}{\langle 0,0\rangle}} 1$	$\xrightarrow{\frac{15,-2}{\langle 0,0\rangle}} 1$		
26	$14 \xrightarrow{\frac{15,-2}{\langle 0,0\rangle}} 0$	$\xrightarrow{\frac{10,-2}{\langle 0,0\rangle}} 7$	$\xrightarrow{\frac{10,-2}{\langle 0,0\rangle}} 6$	$\xrightarrow{\frac{15,-2}{\langle 0,0\rangle}} 7$	$\xrightarrow{\frac{10,-2}{\langle 0,0\rangle}} 1$	$\xrightarrow{\frac{15,-2}{\langle 0,0\rangle}} 6$	$\xrightarrow{\frac{15,-2}{\langle 0,0\rangle}} 7$	$\xrightarrow{\frac{10,-2}{\langle 0,0\rangle}} 1$	-24	n/a
	$1 \xrightarrow{\frac{13,-2}{\langle 0,0\rangle}} 1$	$\xrightarrow{\frac{13,-2}{\langle 0,0\rangle}} 6$	$\xrightarrow{\frac{13,-2}{\langle 0,0\rangle}} 1$	$\xrightarrow{\frac{7,-2}{\langle 0,0\rangle}} 6$	$\xrightarrow{\frac{15,-2}{\langle 0,0\rangle}} 1$	$\xrightarrow{\frac{15,-2}{\langle 0,0\rangle}} 1$	$\xrightarrow{\frac{15,-2}{\langle 0,0\rangle}} 1$	$\xrightarrow{\frac{15,-2}{\langle 0,0\rangle}} 1$		

Table 19. Differential paths through 16 rounds with one active bit. The first differential path (message pattern 1) applies for all bit positions j, while the remaining three paths apply only for bit position 31. The first columns lists the message pattern numbers for the bit position. The 16-round differential paths are split over two consecutive rows. The values above the arrows are the roundkey difference and probability (above the line) and the class (below the line). The values at the same level at the arrows list the bit differences in the state. The final two columns list the probability of the differential path when $j \in \{63, 31\}$ and when $j \notin \{63, 31\}$.

Note that the differential path for message pattern 1 is the differential path observed by Mendel and Schläffer [1]. With this message pattern, there are no differences in any of the roundkeys.

7 Differential Paths with Two Active Bit Positions

The only differential paths through the *C-PHTX* involving two or less active bits are of the form:

$$\hat{\Delta}C_c = [-,-,-] \rightarrow \hat{\Delta}C_d = [-,-,-]: \text{ Case 0;}$$
$$\hat{\Delta}C_c = [63,31,-] \rightarrow \hat{\Delta}C_d = [-,31,-]: \text{ Case } \gamma.$$

Similarly, the only differential through the *G-PHTX* involving only two or less active bits are cases 0 and γ. Hence, only classes $\langle 0,0\rangle$ $\langle 0,\gamma\rangle$, $\langle \gamma,0\rangle$ and $\langle 0,\gamma\rangle$. use two or less active bits. Note that these classes have no differences in bit position 20.

7.1 One-round Differential Paths with Two Active Bit Positions

Table 12 indicates that:

- For class $\langle 0,0\rangle$: $\hat{\Delta}_c[31] = \hat{\Delta}_d[31] \in S^0$, $\hat{\Delta}_c[63] = \hat{\Delta}_d[63] \in S^0$.
- For class $\langle 0,\gamma\rangle$: $\hat{\Delta}_c[31] = \hat{\Delta}_d[31] \in S^1$, $\hat{\Delta}_c[63] \in S^1$ and $\hat{\Delta}_d[63] \in S^0$.
- For class $\langle \gamma,0\rangle$: $\hat{\Delta}_c[31] = \hat{\Delta}_d[31] \in S^2$, $\hat{\Delta}_c[63] \in S^2$ and $\hat{\Delta}_d[63] \in S^0$.
- For class $\langle \gamma,\gamma\rangle$: $\hat{\Delta}_c[31] = \hat{\Delta}_d[31] \in S^3$, $\hat{\Delta}_c[63] \in S^3$ and $\hat{\Delta}_d[63] \in S^0$.

The process for tracing internal differences is similar for all three classes, so we trace the internal differences of only once of the classes, simply as an example. The internal differences for class $\langle 0,\gamma\rangle$ are traced in Table 20.

The one-round differential paths in all three classes are combined in the more-useful "input-output-roundkey" representations: Table 21 and Table 22 apply for bit positions 63 and 31 respectively. Note that the output differences for bit position 63 are restricted to the set $\hat{\Delta}_i[63] \in \{0,1,6,7\}$, while the output differences for $\hat{\Delta}_i[63]$ depend on the class.

1. $\hat{\Delta}_c[63]$		1	3	9	11
3a. $\hat{\Delta}_b[63]$		1	3,7,11,15	9,11,13,15	5,7,9,11
3b'. $\hat{\Delta}_b^L[63]$		1	5,6,11,14	9,11,12,14	4,6,9,11
$\hat{\Delta}K_i[63]$	$\hat{\Delta}K_i^L[63]$	$\hat{\Delta}_{i-1} = \hat{\Delta}_b^L \oplus \hat{\Delta}K_i^L$			
0	0	1	5,6,11,14	9,11,12,14	4,6,9,11
2	2	3	7,4,9,12	11,9,14,12	6,4,11,9
5	4	5	1,2,15,10	13,15,8,10	0,2,13,15
8	8	9	13,14,3,6	1,3,4,6	12,14,1,3
10	10	11	11,12,1,4	3,1,6,4	14,12,3,1
$\hat{\Delta}_d[63] \in S^0$		0	2	8	10
$\hat{\Delta}_i[63]$		0	6	1	7
		S^1			
1. $\hat{\Delta}_c[31]$		1	3	9	11
3a. $\hat{\Delta}_b[31]$		1	3,7,11,15	9,11,13,15	5,7,9,11
3b'. $\hat{\Delta}_b^L[31]$		1	5,6,11,14	9,11,12,14	4,6,9,11
$\hat{\Delta}K_i[31]$	$\hat{\Delta}K_i^L[31]$	$\hat{\Delta}_{i-1} = \hat{\Delta}_b^L \oplus \hat{\Delta}K_i^L$			
0	0	1	5,6,11,14	9,11,12,14	4,6,9,11
2	2	3	7,4,9,12	11,9,14,12	6,4,11,9
5	4	5	1,2,15,10	13,15,8,10	0,2,13,15
7	6	7	3,0,5,8	15,13,10,8	2,0,15,13
8	8	9	13,14,3,6	1,3,4,6	12,14,1,3
10	10	11	11,12,1,4	3,1,6,4	14,12,3,1
13	12	13	9,10,7,2	5,7,0,2	8,10,5,7
15	14	15	11,8,5,0	7,5,2,0	10,8,7,5
$\hat{\Delta}_d[31] \in S^1$		1	3	9	11
$\hat{\Delta}_i[31]$		3	5	2	4
Pr. ($\log_2$)		0	-2	-2	-2

Table 20. Tracing differences in bit position 63 and 31 for Class $\langle 0, \gamma \rangle$ and the corresponding probability factors.

$\hat{\Delta}_c$	0	1	4	5	8	9	12	13	2	3	6	7	10	11	14	15
$\hat{\Delta}_i$	$0_{0,0,-2,-2}$				$1_{all\ -2}$				$6_{all\ -2}$				$7_{all\ -2}$			
Class	0,0	$0,\gamma$	$\gamma,0$	γ,γ	0,0	$0,\gamma$	$\gamma,0$	γ,γ	0,0	$0,\gamma$	$\gamma,0$	γ,γ	0,0	$0,\gamma$	$\gamma,0$	γ,γ
$\hat{\Delta}_{i-1}$	$\hat{\Delta}K_i$ such that $\hat{\Delta}_{i-1} \overset{\hat{\Delta}K_i}{\rightarrow} \hat{\Delta}_c$															
0	0		2,10		8,10		2,8				2,10	5	8,10	5	2,8	5
1		0		2,10		8,10		2,8	5	5,10	5	2,10	5	8,10	5	2,8
2	2		0,8	5	8,10		0,10	5		5	0,8		8,10	5	0,10	
3		2	5	0,8		8,10	5	0,10	5	8		0,8	5	8,10		0,10
4	5			2,10		8,10		2,8		2,10		0,8		0,2		0,10
5		5	2,10		8,10		2,8		0,2,8,10	0	0,8		0,2		0,10	
6			5	0,8		8,10	5	0,10		0,8	5	2,10		0,2	5	2,8
7			0,8	5	8,10		0,10	5	0,2,8,10	2	2,10	5	0,2		2,8	5
8	8		2,10		0,2	5	0,10	5			2,10	5	0,2		0,10	
9		8		2,10	5	0,2	5	0,10	5	2	5	2,10		0,2		0,10
10	10		0,8	5	0,2	5	2,8			5	0,8		0,2		2,8	5
11		10	5	0,8	5	0,2		2,8	5	0		0,8		0,2	5	2,8
12				2,10	5	0,2	5	0,10		2,10		0,8	5	8,10	5	2,8
13			2,10		0,2	5	0,10	5	0,2,8,10	8	0,8		8,10	5	2,8	5
14			5	0,8	5	0,2		2,8		0,8	5	2,10	5	8,10		0,10
15			0,8	5	0,2	5	2,8		0,2,8,10	10,5	2,10	5	8,10	5	0,10	

Table 21. The "input-output-roundkey" representation of the one-round differential paths of bit position 63 for classes $\langle 0,0\rangle$, $\langle 0,\gamma\rangle$ $\langle \gamma,0\rangle$ $\langle \gamma,\gamma\rangle$. See the accompanying text for an explanation of this table.

Interpreting Table 21. The first row indicates the possible values for the intermediate difference $\hat{\Delta}_c[63]$. Note that for each $\hat{\Delta}_c[63]$ there is one possible resulting output difference $\hat{\Delta}_i[63]$ (since $\hat{\Delta}_d[63] \in S^0$ for these classes). The corresponding output differences $\hat{\Delta}_i[63]$ are listed in the next row. The class that allows $\hat{\Delta}_c[63]$ (in a particular column) to result in the corresponding $\hat{\Delta}_i[63]$ is indicated in the next row by following down the column. Each of the last 16 rows corresponds to the 16 possible input differences $\hat{\Delta}_{i-1}[63]$: the input difference is listed in the first entry of the row. The remaining entries in these rows correspond to the roundkey differences that allow the input difference $\hat{\Delta}_{i-1}[63]$ to result in the intermediate difference $\hat{\Delta}_c[63]$ at the top of that column. The probability of the differential path for the four possible intermediate differences $\hat{\Delta}_c[63]$ is provided in the subscripts of the output difference.

For a choice of an input difference (from the bottom 16 rows) and an output difference (from the top of the table) then this table can be used to determine the combinations of roundkey differences and classes such that the input difference will result in the output difference. The following technique can be used:

1. Choose any of the classes $\langle 0,0\rangle$, $\langle 0,\gamma\rangle$ $\langle \gamma,0\rangle$ $\langle \gamma,\gamma\rangle$. Choosing a class will determine the intermediate difference $\hat{\Delta}_c[63]$ that results in the output dif-

ference when using the chosen class. Determining the intermediate difference will also dictate a column to be used.

2. In the row corresponding to the input difference, examine the entry in the column determined by the choice of a class. This entry contains the set of roundkey differences that allow the input difference to result in the output difference when using the chosen class.

$\hat{\Delta}_c$	0	1	4	5	8	9	12	13	2	3	6	7	10	11	14	15
	Output Difference															
$\hat{\Delta}_i$	0_0	3_0	8_{-2}	11_{-2}	1_{-2}	2_{-2}	9_{-2}	10_{-2}	6_{-2}	2_{-2}	9_{-2}	10_{-2}	7_{-2}	4_{-2}	15_{-2}	12_{-2}
Class	$0,0$	$0,\gamma$	$\gamma,0$	γ,γ	$0,0$	$0,\gamma$	$\gamma,0$	γ,γ	$0,0$	$0,\gamma$	$\gamma,0$	γ,γ	$0,0$	$0,\gamma$	$\gamma,0$	γ,γ
$\hat{\Delta}_{i-1}$	$\hat{\Delta}K_i$ such that $\hat{\Delta}_{i-1} \xrightarrow{\hat{\Delta}K_i} \hat{\Delta}_i$															
0	0		2,10	7,15	8,10	13,15	2,8	7,13		7,15	2,10	5,13	8,10	5,7	2,8	5,15
1		0	7,15	2,10	13,15	8,10	7,13	2,8	5,7,13,15	5,10	5,13	2,10	5,7	8,10	5,15	2,8
2	2		0,8	5,13	10,8	15,13	0,10	5,15		5,13	0,8	7,15	10,8	7,5	0,10	7,13
3		2	5,13	0,8	15,13	10,8	5,15	0,10	7,5,15,13	7,8	7,15	0,8	7,5	10,8	7,13	0,10
4	5		7,15	2,10	13,15	8,10	7,13	2,8		2,10	7,15	0,8	13,15	0,2	7,13	0,10
5		5	2,10	7,15	8,10	13,15	2,8	7,13	0,2,8,10	0,15	0,8	7,15	0,2	13,15	0,10	7,13
6	7		5,13	0,8	15,13	10,8	5,15	0,10		0,8	5,13	2,10	15,13	2,0	5,15	2,8
7		7	0,8	5,13	10,8	15,13	0,10	5,15	2,0,10,8	2,13	2,10	5,13	2,0	15,13	2,8	5,15
8	8		10,2	15,7	0,2	5,7	10,0	15,5		15,7	10,2	13,5	0,2	13,15	10,0	13,7
9		8	15,7	10,2	5,7	0,2	15,5	10,0	13,15,5,7	13,2	13,5	10,2	13,15	0,2	13,7	10,0
10	10		8,0	13,5	2,0	7,5	8,2	13,7		13,5	8,0	15,7	2,0	15,13	8,2	15,5
11		10	13,5	8,0	7,5	2,0	13,7	8,2	15,13,7,5	15,0	15,7	8,0	15,13	2,0	15,5	8,2
12	13		15,7	10,2	5,7	0,2	15,5	10,0		10,2	15,7	8,0	5,7	8,10	15,5	8,2
13		13	10,2	15,7	0,2	5,7	10,0	15,5	8,10,0,2	8,7	8,0	15,7	8,10	5,7	8,2	15,5
14	15		13,5	8,0	7,5	2,0	13,7	8,2		8,0	13,5	10,2	7,5	10,8	13,7	10,0
15		15	8,0	13,5	2,0	7,5	8,2	13,7	10,8,2,0	10,5	10,2	13,5	10,8	7,5	10,0	13,7

Table 22. The "input-output-roundkey" representation of the one-round differential paths of bit position 31 for classes that allow two or less active bits. See the text accompanying Table 21 for an explanation of this table.

7.2 Multi-Round Differential Paths with Two Active Bit Positions

We now consider multi-round differential paths where two bits are active: that is, bit positions 31 and 63 are active. The set of message patterns for this case is spanned by the linear combinations of

- the 16 message patterns 1-16 in Table 7 for bit position 63;
- the 32 message patterns 1-32 in Table 7 for bit position 31; and
- the 16 message patterns 33-48 in Table 8 involving both bit positions 63 and 31.

This is a total of 8192 possible message patterns. There are too many paths to search by hand, so an automated searching program was written. The probability was evaluated over rounds 5 to 16.

The best differential path has probability 2^{-4}, and is discussed in Section 7.3. The next best differential paths have probability 2^{-10}: these are discussed in Section 7.4. There are other differential paths of probability 2^{-12}, 2^{-14} and so forth, but there are too many to report.

7.3 An Exceptional Differential Path

The best differential path has probability 2^{-4}. This is a very high probability, so the differential path warrants further investigation. The message blocks for this differential path have the following differences:

$$\begin{aligned}
\hat{\Delta}RK_0 &= \hat{\Delta}LK_0 = \texttt{0x80000000};\\
\hat{\Delta}RK_1 &= \hat{\Delta}LK_1 = \texttt{0x80008000};\\
\hat{\Delta}RK_2 &= \hat{\Delta}LK_2 = \texttt{0x00008000};\\
\hat{\Delta}RK_i &= \hat{\Delta}LK_i = \texttt{0x00000000},\ 3 \leq i \leq 7.
\end{aligned}$$

That is, the only differences are in bit 63 of the words RK_0, RK_1, LK_0, LK_1 and in bit 31 of the words RK_1, RK_2, LK_1, LK_2. This results in message pattern 13 for bit position 63 and message pattern 5 for bit position 31, as shown in Table 23. These message patterns are good because there are only a few rounds with nonzero roundkey differences. For most rounds with zero roundkey difference, we can use the "trivial" one-round differential:

$$\hat{\Delta}_{i-1}[63,31] = \mathbf{0} \xrightarrow[\langle 0,0\rangle]{\mathbf{0}} \hat{\Delta}_i[63,31] = \mathbf{0},$$

which holds with probability one, hence the high probability of the differential.

i	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
$\hat{\Delta}K_i[63]$	5	5	–	–	–	–	–	10	–	–	–	–	–	–	–	–
$\hat{\Delta}K_i[31]$	–	5	5	–	–	–	–	2	8	–	–	–	–	–	–	–

Table 23. Roundkey differences for a high-probability differential paths with two active bit positions for Rounds 3-16. Zero differences have been represented using a period "-" to aid readability.

The differential path starts at round 3 with $\hat{\Delta}_2[63,31] = (0,4)$ and ends with $\hat{\Delta}_{16}[63,31] = \mathbf{0}$. The sequence of one-round differential paths is shown in Table 24. The probability of this differential is extremely high: all rounds have probability one except round 9 which has probability $(2^{-2})^2 = 2^{-4}$. That is, if $\hat{\Delta}_3[63,31] = (0,4)$, then $\hat{\Delta}_{16}[63,31] = \mathbf{0}$, with probability $\frac{1}{16}$. None of the paths that we looked at could be used for Rounds 1 and 2. However, for many input hash differences, message modification may be used to result in the correct difference $\hat{\Delta}_3[63, 31] = (0; 4)$. We have not investigated the possibilities for Round 1 and 2 in any detail for TIB3.

Round				
3	4-8	9	10	11-16
$(0,4) \xrightarrow{\frac{[0,5],-2}{\langle 0,\gamma\rangle}} \mathbf{0}$	$\xrightarrow{\frac{\mathbf{0},0}{\langle 0,0\rangle}} \mathbf{0}$	$\xrightarrow{\frac{[10,2],-4}{\langle \gamma,0\rangle}} (0,8)$	$\xrightarrow{\frac{[0,8],0}{\langle 0,0\rangle}} \mathbf{0}$	$\xrightarrow{\frac{\mathbf{0},0}{\langle 0,0\rangle}} \mathbf{0}$

Table 24. The differential path of probability 2^{-4}. In the table entries, the values above the arrows are: (above the line) the roundkey differences for bit positions 63 and 31 and the probability factor for that round; and (below the line) the class. The values at the same level at the arrows list the bit differences in the state for bit positions 63 and 31 respectively. Where all roundkey differences or state differences are zero, a boldface zero "**0**" is used to save space.

The final output difference from the TIB3 block cipher is $\hat{\Delta}_{16} = 0$ for this differential. It follows that $\hat{\Delta}h_{i+1} = \hat{\Delta}h_i$ for this differential. This might be exploited to create message pairs for which $\hat{\Delta}h_i = \hat{\Delta}h_{i+1} = \hat{\Delta}h_{i+1} = \cdots$. It is unclear how this differential can be used to compromise TIB3.

Interestingly, this path also applies for TIB3v2 (see Section 9).

7.4 Paths of probability 2^{-10}

There are several message patterns for which there are differential paths of probability 2^{-10}. For rounds 5 to 8 and rounds 11 to 16, these differential paths follow the sequence of one-round differential paths shown in Table 25. The one-round differential paths for rounds 9 and 10 for these message patterns are listed in Table 26.

Round							
5	6	7	8	9	10	11	12-16
$(1,0) \xrightarrow{\frac{[0,5],-2}{\langle 0,\gamma\rangle}} (0,4)$	$\xrightarrow{\frac{[0,5],-2}{\langle 0,0\rangle}} \mathbf{0}$	$\xrightarrow{\frac{\mathbf{0},0}{\langle 0,0\rangle}} \mathbf{0}$	$\xrightarrow{\frac{\mathbf{0},0}{\langle 0,0\rangle}} \mathbf{0}$	$\xrightarrow{\frac{[?,?],-4}{\langle ?,0\rangle}} (?,?)$	$\xrightarrow{\frac{[?,?],-2}{\langle \gamma,\gamma\rangle}} (0,10)$	$\xrightarrow{\frac{[0,10],-2}{\langle 0,0\rangle}} \mathbf{0}$	$\xrightarrow{\frac{\mathbf{0},0}{\langle 0,0\rangle}} \mathbf{0}$

Table 25. The one-round differential paths that are common to all differential paths of probability 2^{-10} for two active bits. The unassigned differences (indicated by "?") in rounds 9 and 10 of the differential paths are shown in Table 26. See the caption to Table 24 for an explanation of the notation.

Message Patterns			Round	
63+31	63	31	9	10
-	13	8	$\mathbf{0} \xrightarrow{\overset{[10;2]}{\langle\gamma,0\rangle}} (6;14)$	$\xrightarrow{\overset{[0;8]}{\langle\gamma,\gamma\rangle}} (0;10)$
-	5	8	$\mathbf{0} \xrightarrow{\overset{[2;2]}{\langle\gamma,0\rangle}} (6;14)$	$\xrightarrow{\overset{[8;8]}{\langle\gamma,\gamma\rangle}} (0;10)$
34	9	4	$\mathbf{0} \xrightarrow{\overset{[8;8]}{\langle\gamma,0\rangle}} (1;9)$	$\xrightarrow{\overset{[10;0]}{\langle\gamma,\gamma\rangle}} (0;10)$
34	5	4	$\mathbf{0} \xrightarrow{\overset{[2;8]}{\langle\gamma,0\rangle}} (1;9)$	$\xrightarrow{\overset{[10;0]}{\langle\gamma,\gamma\rangle}} (0;10)$
34	9	14	$\mathbf{0} \xrightarrow{\overset{[8;2]}{\langle\gamma,0\rangle}} (1;9)$	$\xrightarrow{\overset{[10;0]}{\langle\gamma,\gamma\rangle}} (0;10)$
34	5	14	$\mathbf{0} \xrightarrow{\overset{[2;2]}{\langle\gamma,0\rangle}} (1;9)$	$\xrightarrow{\overset{[10;0]}{\langle\gamma,\gamma\rangle}} (0;10)$
34	13	8	$\mathbf{0} \xrightarrow{\overset{[10;10]}{\langle 0,0\rangle}} (1;1)$	$\xrightarrow{\overset{[2;8]}{\langle\gamma,\gamma\rangle}} (0;10)$
34	9	8	$\mathbf{0} \xrightarrow{\overset{[8;10]}{\langle 0,0\rangle}} (1;1)$	$\xrightarrow{\overset{[10;8]}{\langle\gamma,\gamma\rangle}} (0;10)$
34	5	8	$\mathbf{0} \xrightarrow{\overset{[2;10]}{\langle\gamma,0\rangle}} (1;14)$	$\xrightarrow{\overset{[10;8]}{\langle\gamma,\gamma\rangle}} (0;10)$
40	13	14	$\mathbf{0} \xrightarrow{\overset{[2;10]}{\langle\gamma,0\rangle}} (6;14)$	$\xrightarrow{\overset{[0;2]}{\langle\gamma,\gamma\rangle}} (0;10)$
40	5	14	$\mathbf{0} \xrightarrow{\overset{[10;10]}{\langle\gamma,0\rangle}} (6;14)$	$\xrightarrow{\overset{[8;2]}{\langle\gamma,\gamma\rangle}} (0;10)$
40	13	12	$\mathbf{0} \xrightarrow{\overset{[2;8]}{\langle\gamma,0\rangle}} (6;9)$	$\xrightarrow{\overset{[0;10]}{\langle\gamma,\gamma\rangle}} (0;10)$
40	13	8	$\mathbf{0} \xrightarrow{\overset{[2;2]}{\langle\gamma,0\rangle}} (6;9)$	$\xrightarrow{\overset{[0;10]}{\langle\gamma,\gamma\rangle}} (0;10)$
40	5	12	$\mathbf{0} \xrightarrow{\overset{[10;8]}{\langle\gamma,0\rangle}} (6;9)$	$\xrightarrow{\overset{[8;10]}{\langle\gamma,\gamma\rangle}} (0;10)$
40	5	8	$\mathbf{0} \xrightarrow{\overset{[10;2]}{\langle\gamma,0\rangle}} (6;9)$	$\xrightarrow{\overset{[8;10]}{\langle\gamma,\gamma\rangle}} (0;10)$
42	1	4	$\mathbf{0} \xrightarrow{\overset{[8;8]}{\langle 0,0\rangle}} (1;1)$	$\xrightarrow{\overset{[2;2]}{\langle\gamma,\gamma\rangle}} (0;10)$
42	1	14	$\mathbf{0} \xrightarrow{\overset{[8;2]}{\langle\gamma,0\rangle}} (1;14)$	$\xrightarrow{\overset{[2;2]}{\langle\gamma,\gamma\rangle}} (0;10)$
42	13	14	$\mathbf{0} \xrightarrow{\overset{[2;2]}{\langle\gamma,0\rangle}} (1;14)$	$\xrightarrow{\overset{[2;2]}{\langle\gamma,\gamma\rangle}} (0;10)$
42	5	4	$\mathbf{0} \xrightarrow{\overset{[10;8]}{\langle 0,0\rangle}} (1;1)$	$\xrightarrow{\overset{[10;2]}{\langle\gamma,\gamma\rangle}} (0;10)$

Table 26. The message patterns with differential paths of probability 2^{-10} for rounds 5-16. The first column lists a message pattern from Table 8 (which applies for both bit positions 63 and 31). The second and third columns list the message pattern from Table 7 for bit positions 63 and 31 (respectively) which must be combined with the message pattern from the first column. For rounds 5-8 and rounds 11-16, all these differential paths follow the differential path shown Table 25. Only the differential path for rounds 9 and 10 are shown in the current table. See the caption to Table 24 for an explanation of the notation.

8 Differential Paths with Three Active Bit Positions

These differential paths can have differences in bit positions 63, 31 and 20. All possible classes are considered in this section. A significant challenge is representing the set of one-round differential paths in a small space.

This is achieved by partitioning the classes according to the set of values allowed for $\hat{\Delta}_d[63]$: the set of classes for which $\hat{\Delta}_d[63] \in S^k$ is called the *class group* T^k. Note that whenever $\hat{\Delta}_d[63] \in S^k$ then it is also true that $\hat{\Delta}_d[20] \in S^k$. The classes with two or less active bits corresponds to group T^0. Then, within each group T^k, the classes are ordered according to the set S^j such that $\hat{\Delta}_c[63] \in S^j$. Table 27 shows the class groups $T^0, \ldots, T^3$, and shows the set of possible values $\hat{\Delta}_c[j]$ and $\hat{\Delta}_d[j]$ allowed for the classes in those class groups

Group	T^0				T^1				T^2				T^3			
C-PHTX	0	0	γ	γ	0	0	γ	γ	α	α	β	β	α	α	β	β
G-PHTX	0	γ	0	γ	α	β	α	β	0	γ	0	γ	α	β	α	β
Bit Diff.	Index k s.t. such that Bit Diff $\in S^k$															
$\hat{\Delta}_c[63]$	0	1	2	3	0	1	2	3	0	1	2	3	0	1	2	3
$\hat{\Delta}_c[31]$	0	1	2	3	1	0	3	2	2	3	0	1	3	2	1	0
$\hat{\Delta}_c[20]$	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
$\hat{\Delta}_d[63]$	0	0	0	0	1	1	1	1	2	2	2	2	3	3	3	3
$\hat{\Delta}_d[31]$	0	1	2	3	0	1	2	3	0	1	2	3	0	1	2	3
$\hat{\Delta}_d[20]$	0	0	0	0	1	1	1	1	2	2	2	2	3	3	3	3

Table 27. Partitioning the classes into groups $T^0, \ldots, T^3$. Each column corresponds to a class in the group, and each of the last 6 rows corresponds to a particular bit difference. For the class (in a given column), the bit difference (in given row) must be always in the set S^j where k is the value in the entry where the row and column meet.

8.1 One-round Differential paths with Three Active Bit Positions

Bit Position 20: The difference $\hat{\Delta}_c$ at position 20 is always in set S^0. Consequently, there are only a few options for the internal differences at position 20 and the differences are easily traced. The resulting “input-output-roundkey” table is shown in Table 28.

$\hat{\Delta}_c[20]$	0	8	2	10
Classes such that $\hat{\Delta}_{i-1} \to \hat{\Delta}_c$.	Output Difference $\hat{\Delta}_i[20]$			
$T^0 = \{\langle 0,0\rangle,\ \langle 0,\gamma\rangle,\ \langle\gamma,0\rangle,\ \langle\gamma,\gamma\rangle\}$	0_0	1_{-3}	6_{-3}	7_{-4}
$T^1 = \{\langle 0,\alpha\rangle,\ \langle 0,\beta\rangle,\ \langle\gamma,\alpha\rangle,\ \langle\gamma,\beta\rangle\}$	3_{-2}	2_{-4}	5_{-4}	4_{-4}
$T^2 = \{\langle\alpha,0\rangle,\ \langle\alpha,\gamma\rangle,\ \langle\beta,0\rangle,\ \langle\beta,\gamma\rangle\}$	8_0	9_{-3}	14_{-3}	15_{-4}
$T^3 = \{\langle\alpha,\alpha\rangle,\ \langle\alpha,\beta\rangle,\langle\beta,\alpha\rangle,\ \langle\beta,\beta\rangle\}$	11_{-2}	10_{-4}	13_{-4}	12_{-4}
$\hat{\Delta}_{i-1}[20]$	$\hat{\Delta}K_i[20]$ s.t. $\hat{\Delta}_{i-1}[20] \overset{\hat{\Delta}K_i[20]}{\to} \hat{\Delta}_c[20]$			
0	0	8,10		8,10
1 or 3			5	5
2	2	8,10		8,10
4	5			
5 or 7		8,10	0,2,8,10	0,2
6				
8	8	0,2		0,2
9 or 11		5	5	
10	10	0,2		0,2
12		5		5
13 or 15		0,2	0,2,8,10	8,10
14		5		5

Table 28. The "input-output-roundkey" representation of the one-round differential paths for bit position 20 in all classes. See the accompanying text for an explanation of this table.

Interpreting Table 28. The first row indicates the four possible values for the intermediate difference $\hat{\Delta}_c[20] \in S^0$. Recall that for each $\hat{\Delta}_c[20]$ there are four possible resulting output differences $\hat{\Delta}_i[20]$. These possible output differences $\hat{\Delta}_i[20]$ are listed in the next four rows (excluding the explanatory row). The first entry in these rows list the class group that allows $\hat{\Delta}_c[20]$ (in a particular column) to result in $\hat{\Delta}_i[20]$.

Each of the last 16 rows corresponds to the 16 possible input differences $\hat{\Delta}_{i-1}[20]$: the input difference is listed in the first entry of the row. The remaining entries in these rows correspond to the roundkey differences that allow the input difference $\hat{\Delta}_{i-1}[20]$ to result in the intermediate difference $\hat{\Delta}_c[20]$ at the top of that column. The probability of the differential path for the four possible intermediate differences $\hat{\Delta}_c[20]$ is provided in the subscript of the output differences.

For a choice of an input difference (from the bottom 16 rows) and an output difference (from the top of the table) then this table can be used to determine the class group and the choices of roundkey differences such that the input difference will result in the output difference. The following technique can be used:

1. The first entry of the row containing the output difference will indicate the applicable class group.
2. In the row corresponding to the input difference, examine the entry in the column containing the output difference. This entry contains the set of roundkey differences that allow the input difference to result in the output difference when using the indicated class.

Example 4. Suppose we choose input difference $\hat{\Delta}_{i-1}[20] = 3$ and output difference $\hat{\Delta}_i[20] = 5$. The class group in the first entry of that row containing the output difference 5 is $T^2 = \{\langle\alpha, 0\rangle, \langle\alpha, \gamma\rangle, \langle\beta, 0\rangle, \langle\beta, \gamma\rangle\}$. The entry where the row with input difference $\hat{\Delta}_{i-1}[20] = 3$, meets the column for output difference $\hat{\Delta}_i[20] = 5$, contains the roundkey difference 5. Hence, only roundkey difference $\hat{\Delta}_i[20] = 5$ can be used with input difference $\hat{\Delta}_{i-1}[20] = 3$ and output difference $\hat{\Delta}_i[20] = 5$. The subscript of the output difference is -4, and thus the probability factor for this differential path is 2^{-4}. $\square$

Bit Position 63: The differential paths for position 63 are more complicated to explain, since the differences $\hat{\Delta}_c[63]$ and $\hat{\Delta}_d[63]$ may range independently over the four sets $S^0, \ldots, S^3$. Table 29 shows the set of one-round differential paths. Notice that Table 29 contains the differential paths for all classes, so this table may also be used as a summary of the possible differential paths at bit position 63 for differential paths with two active bits.

0	$\hat{\Delta}_i$	0				1				6				7			
	Cls	0,0	$0,\gamma$	$\gamma,0$	γ,γ	0,0	$0,\gamma$	$\gamma,0$	γ,γ	0,0	$0,\gamma$	$\gamma,0$	γ,γ	0,0	$0,\gamma$	$\gamma,0$	γ,γ
1	$\hat{\Delta}_i$	3				2				5				4			
	Cls	$0,\alpha$	$0,\beta$	γ,α	γ,β	$0,\alpha$	$0,\beta$	γ,α	γ,β	$0,\alpha$	$0,\beta$	γ,α	γ,β	$0,\alpha$	$0,\beta$	γ,α	γ,β
2	$\hat{\Delta}_i$	8				9				14				15			
	Cls	$\alpha,0$	α,γ	$\beta,0$	β,γ	$\alpha,0$	α,γ	$\beta,0$	β,γ	$\alpha,0$	α,γ	$\beta,0$	β,γ	$\alpha,0$	α,γ	$\beta,0$	β,γ
3	$\hat{\Delta}_i$	11				10				13				12			
	Cls	α,α	α,β	β,α	β,β	α,α	α,β	β,α	β,β	α,α	α,β	β,α	β,β	α,α	α,β	β,α	β,β
$\hat{\Delta}_c$		0	1	4	5	8	9	12	13	2	3	6	7	10	11	14	15
$\hat{\Delta}_{i-1}$		$\hat{\Delta}K_i$ such that $\hat{\Delta}_{i-1} \overset{\hat{\Delta}K_i}{\rightarrow} \hat{\Delta}_c$															
0		0		2,10		8,10		2,8				2,10	5	8,10	5	2,8	5
1			0		2,10		8,10		2,8	5	5,10	5	2,10	5	8,10	5	2,8
2		2		0,8	5	8,10		0,10	5		5	0,8		8,10	5	0,10	
3			2	5	0,8		8,10	5	0,10	5	8		0,8	5	8,10		0,10
4		5			2,10		8,10		2,8		2,10		0,8		0,2		0,10
5			5	2,10		8,10		2,8		♦	0	0,8		0,2		0,10	
6				5	0,8		8,10	5	0,10		0,8	5	2,10		0,2	5	2,8
7				0,8	5	8,10		0,10	5	♦	2	2,10	5	0,2		2,8	5
8		8		2,10		0,2	5	0,10	5			2,10	5	0,2		0,10	
9			8		2,10	5	0,2	5	0,10	5	2	5	2,10		0,2		0,10
10		10		0,8	5	0,2	5	2,8			5	0,8		0,2		2,8	5
11			10	5	0,8	5	0,2		2,8	5	0		0,8		0,2	5	2,8
12					2,10	5	0,2	5	0,10		2,10		0,8	5	8,10	5	2,8
13				2,10		0,2	5	0,10	5	♦	8	0,8		8,10	5	2,8	5
14				5	0,8	5	0,2		2,8		0,8	5	2,10	5	8,10		0,10
15				0,8	5	0,2	5	2,8		♦	10,5	2,10	5	8,10	5	0,10	
Pr.		0	0	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2

Table 29. The "input-output-roundkey" representation of the one-round differential paths for bit position 63 in all classes. See the accompanying text for an explanation of this table. The symbol "♦" represents the set 0,2,8,10.

Interpreting Table 29.

The first eight rows consists of four sets of pairs of rows indexed 0 through to 3 according to the class group being used. In each pair of rows, the first row lists a value of the output difference $\hat{\Delta}_i[63]$ and the second row lists the classes that may be used with that output difference: one " "sub-column" is used for each class. The ninth row shows the intermediate difference $\hat{\Delta}_c[63]$ that applies when using a combination of output difference $\hat{\Delta}_i[63]$ and class in any of the previous pairs of rows. The probability (logarithm base 2) of the differential path from intermediate difference $\hat{\Delta}_c[63]$ to output difference $\hat{\Delta}_i[63]$ is provided in the last row of the table.. For example, if using output difference $\hat{\Delta}_i[63]$ for differentials in class $\langle \gamma, 0, rangle$, then corresponding intermediate difference is $\hat{\Delta}_c[63] = 2$, and the differential path holds with probability 2^{-2}.

Each of the remaining 16 rows corresponds to the 16 possible input differences $\hat{\Delta}_{i-1}[63]$: the input difference is listed in the first entry of the row. The remaining entries in these rows correspond to the roundkey differences that allow the input difference $\hat{\Delta}_{i-1}[63]$ to result in the intermediate difference $\hat{\Delta}_c[63]$ where that column meets the ninth row.

The four rows are indexed $k = 1$ through to $k = 4$.

For a choice of an input difference (from the bottom 16 rows) and an output difference (from the top of the table) then this table can be used to determine the combinations of roundkey differences and classes such that the input difference will result in the output difference. The following technique can be used:

1. The output difference is in a one of the pairs of rows in the first eight rows.
2. For that row pair, choose any of the classes from the four classes in the second rows of the pair. Choosing a class will determine the intermediate difference $\hat{\Delta}_c[63]$ that results in the output difference when using the chosen class. Determining the intermediate difference will also dictate a column to be used.
3. In the row corresponding to the input difference, examine the entry in the column determined by the choice of a class. This entry contains the set of roundkey differences that allow the input difference to result in the output difference when using the chosen class.

Example 5. Suppose we choose input difference $\hat{\Delta}_{i-1}[63] = 3$ and output difference $\hat{\Delta}_i[63] = 8$. The possible classes for that output difference are $\langle \alpha, 0 \rangle$, $\langle \alpha, \gamma \rangle$, $\langle \beta, 0 \rangle$ and $\langle \beta, \gamma \rangle$. Now look at the row with input difference $\hat{\Delta}_{i-1}[63] = 3$:

1. The column for class $\langle \alpha, 0 \rangle$ and $\hat{\Delta}_i[63] = 8$, indicates that no roundkey differences are possible, hence this class cannot be used in this case.
2. The column for class $\langle \alpha, \gamma \rangle$ and $\hat{\Delta}_i[63] = 8$, indicates that roundkey difference $\hat{\Delta} K_i[63] = 2$ can be used.
3. The column for class $\langle \beta, 0 \rangle$ and $\hat{\Delta}_i[63] = 8$, indicates that roundkey difference $\hat{\Delta} K_i[63] = 5$ can be used.
4. The column for class $\langle \beta, \gamma \rangle$ and $\hat{\Delta}_i[63] = 8$, indicates that roundkey differences $\hat{\Delta} K_i[63] = 0$, $\hat{\Delta} K_i[63] = 8$ can be used.

For the last three possibilities, the probability factor is 2^{-2}. $\square$

Bit Position 31: The differential paths for position 31 are treated in a similar manner to the differential paths for position 63, since the differences $\hat{\Delta}_c[31]$ and $\hat{\Delta}_d[31]$ may range independently over the four sets $S^0, \dots, S^3$. Bit position 31 differs from bit position 63 in that more roundkey differences are applicable for bit position 31, and the bit position 31 re-orders the classes for which $\hat{\Delta}_c \rightarrow \Delta_{i-1}$. Table 30 represents the differential paths at bit position 31. This table is interpreted using in the same way as the Table 29. Notice that Table 30 contains the differential paths for all classes, so this table may also be used as a summary of the possible differential paths at bit position 63 for differential paths with one active bit or two active bits.

0 $\hat{\Delta}_i$	0				1				6				7			
Cls	$0,0$	$0,\alpha$	$\alpha,0$	α,α	$0,0$	$0,\alpha$	$\alpha,0$	α,α	$0,0$	$0,\alpha$	$\alpha,0$	α,α	$0,0$	$0,\alpha$	$\alpha,0$	α,α
1 $\hat{\Delta}_i$	3				2				5				4			
Cls	$0,\beta$	$0,\gamma$	α,β	α,γ	$0,\beta$	$0,\gamma$	α,β	α,γ	$0,\beta$	$0,\gamma$	α,β	α,γ	$0,\beta$	$0,\gamma$	α,β	α,γ
2 $\hat{\Delta}_i$	8				9				14				15			
Cls	$\beta,0$	β,α	$\gamma,0$	γ,α	$\beta,0$	β,α	$\gamma,0$	γ,α	$\beta,0$	β,α	$\gamma,0$	γ,α	$\beta,0$	β,α	$\gamma,0$	γ,α
3 $\hat{\Delta}_i$	11				10				13				12			
Cls	β,β	β,γ	γ,β	γ,γ	β,β	β,γ	γ,β	γ,γ	β,β	β,γ	γ,β	γ,γ	β,β	β,γ	γ,β	γ,γ
$\hat{\Delta}_c$	0	1	4	5	8	9	12	13	2	3	6	7	10	11	14	15
$\hat{\Delta}_{i-1}$	$\hat{\Delta}K_i$ such that $\hat{\Delta}_{i-1} \overset{\hat{\Delta}K_i}{\rightarrow} \hat{\Delta}_c$															
0	0		2,10	7,15	8,10	13,15	2,8	7,13		7,15	2,10	5,13	8,10	5,7	2,8	5,15
1		0	7,15	2,10	13,15	8,10	7,13	2,8	◊	5,10	5,13	2,10	5,7	8,10	5,15	2,8
2	2		0,8	5,13	10,8	15,13	0,10	5,15		5,13	0,8	7,15	10,8	7,5	0,10	7,13
3		2	5,13	0,8	15,13	10,8	5,15	0,10	◊	7,8	7,15	0,8	7,5	10,8	7,13	0,10
4	5		7,15	2,10	13,15	8,10	7,13	2,8		2,10	7,15	0,8	13,15	0,2	7,13	0,10
5		5	2,10	7,15	8,10	13,15	2,8	7,13	♦	0,15	0,8	7,15	0,2	13,15	0,10	7,13
6	7		5,13	0,8	15,13	10,8	5,15	0,10		0,8	5,13	2,10	15,13	2,0	5,15	2,8
7		7	0,8	5,13	10,8	15,13	0,10	5,15	♦	2,13	2,10	5,13	2,0	15,13	2,8	5,15
8	8		10,2	15,7	0,2	5,7	10,0	15,5		15,7	10,2	13,5	0,2	13,15	10,0	13,7
9		8	15,7	10,2	5,7	0,2	15,5	10,0	◊	13,2	13,5	10,2	13,15	0,2	13,7	10,0
10	10		8,0	13,5	2,0	7,5	8,2	13,7		13,5	8,0	15,7	2,0	15,13	8,2	15,5
11		10	13,5	8,0	7,5	2,0	13,7	8,2	◊	15,0	15,7	8,0	15,13	2,0	15,5	8,2
12	13		15,7	10,2	5,7	0,2	15,5	10,0		10,2	15,7	8,0	5,7	8,10	15,5	8,2
13		13	10,2	15,7	0,2	5,7	10,0	15,5	♦	8,7	8,0	15,7	8,10	5,7	8,2	15,5
14	15		13,5	8,0	7,5	2,0	13,7	8,2		8,0	13,5	10,2	7,5	10,8	13,7	10,0
15		15	8,0	13,5	2,0	7,5	8,2	13,7	♦	10,5	10,2	13,5	10,8	7,5	10,0	13,7
Pr.	0	0	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2

Table 30. The "input-output-roundkey" representation of the one-round differential paths for bit position 31 for all classes. See the text accompanying Table 29 for an explanation. The symbol "♦" represents the set 0,2,8,10. The symbol "◊" represents the set 5,7,13,15.

Example 6. Suppose we choose input difference $\hat{\Delta}_{i-1}[31] = 3$ and output difference $\hat{\Delta}_i[31] = 8$. The possible classes for that output difference are $\langle\beta, 0\rangle$, $\langle\beta, \alpha\rangle$, $\langle\gamma, 0\rangle$ and $\langle\gamma, \alpha\rangle$. Now look at the row with input difference $\hat{\Delta}_{i-1}[31] = 3$:

1. The column for class $\langle\beta, 0\rangle$ and $\hat{\Delta}_i[31] = 8$, indicates that no roundkey differences roundkey differences are possible, hence this class cannot be used in this case.
2. The column for class $\langle\beta, \alpha\rangle$ and $\hat{\Delta}_i[31] = 8$, indicates that roundkey difference $\hat{\Delta}K_i[31] = 2$ can be used.
3. The column for class $\langle\gamma, 0\rangle$, indicates that roundkey differences $\hat{\Delta}K_i[31] = 5$ and $\hat{\Delta}K_i[31] = 13$ can be used.
4. The column for class $\langle\gamma, \alpha\rangle$, indicates that roundkey differences $\hat{\Delta}K_i[31] = 0$, $\hat{\Delta}K_i[31] = 8$ can be used.

For the last three possibilities, the probability factor is 2^{-2}. □

8.2 Multi-Round Differential paths with Three Active Bit Positions

The message patterns for this case are spanned by combinations of

- the 16 message patterns 1-16 in Table 7 for bit position 20;
- the 16 message patterns 1-16 in Table 7 for bit position 63;
- the 32 message patterns 1-32 in Table 7 for bit position 31; and
- the 16 message patterns 33-48 in Table 8 involving both bit positions 63 and 31.

This is a total of $2^{17} = 131072$ possible message patterns.

The following algorithm is used to find high probability differential paths for three active bits. The input to the program was a probability limit that allowed us to quickly search for low probability differential paths. An iterative tree searching algorithm is used: the **Walk** algorithm searches through the possible one-round differentials at a given round, and calls the **Walk** algorithm on the next round if the cumulative probability is still below the probability limit.

Search Algorithm for Three Active Bit Positions

Input: probability limit *Limit*.

1. **Branch on Message Patterns**. Choose message pattern spanned by the above set, and determine the message pattern π_j for each bit positions $j \in \{20, 31, 63\}$. This will specify $\Delta K_i[j]$ for $j \in \{20, 31, 63\}$ for each round i
2. **Branch on Initial Differences**. Choose an initial difference Δ_4 for each bit position 20, 31 and 63. Set $i = 5$ and cumulative probability $p = 1$.
3. Apply $\mathbf{Walk}(5, \pi_{63}, \pi_{31}, \pi_{20}, \Delta_4[63], \Delta_4[31], \Delta_4[20], p, \mathit{Limit})$.

Walk Algorithm
Inputs:

- Round i,
- Message patterns $\pi_{63}, \pi_{31}, \pi_{20}$,
- Input differences $\Delta_4[63], \Delta_4[31], \Delta_4[20]$,
- Cumulative probability p.
- Probability limit $Limit$.

1. **Branch for** $\Delta_i[20]$. Set $p_{new} = p$. From input difference $\Delta_{i-1}[20]$ and $\Delta K_i[20]$ choose a possible output difference $\Delta_i[20]$ and note the corresponding class group and probability factor. *Update Cumulative Probability.*
2. **Branch for** $\Delta_i[63]$. From input difference $\Delta_{i-1}[63]$, $\Delta K_i[63]$ and the class group chosen in branching for bit position 20, choose a possible output difference $\Delta_i[63]$ and note the corresponding class and probability factor. *Update Cumulative Probability.*
3. **Branch for** $\Delta_i[31]$. From input difference $\Delta_{i-1}[31]$, $\Delta K_i[63]$ and the class chosen in branching for bit position 63, choose a possible output difference $\Delta_i[31]$ and note the probability factor. *Update Cumulative Probability.*
4. If $i == 16$, then output the differential path, otherwise apply **Walk**$(i+1, \pi_{63}, \pi_{31}, \pi_{20}, \Delta_i[63], \Delta_i[31], \Delta_i[20], p_{new}, Limit)$, which looks for paths in the next round.

At every *Update Cumulative Probability* in the algorithm, the cumulative probability p_{new} is updated and if the cumulative probability exceeded the limit, then the search chooses another value at that branching point or (if the choices were exhausted) then the tree reverted to the previous branching point.

There are too many paths to search by hand, so an automated searching program was written. The best differential paths with three active bits have probability 2^{-10}: these are shown in Table 31. There are other paths for message pattern $(9, 13, 9)$ of probability 2^{-10} that are not shown here.

Message	Rounds				
Patterns	5-7	8	9	10	11-16
$\begin{bmatrix}5\\9\\9\end{bmatrix}$	$\begin{bmatrix}7\\0\\0\end{bmatrix} \xrightarrow[\langle 0,\gamma\rangle]{\mathbf{0},-2} \begin{bmatrix}7\\0\\0\end{bmatrix}$	$\xrightarrow[\langle \beta,0\rangle]{\mathbf{0},-2} \begin{bmatrix}8\\8\\8\end{bmatrix}$	$\xrightarrow[\langle \beta,0\rangle]{\begin{bmatrix}2\\8\\8\end{bmatrix},-2} \begin{bmatrix}8\\8\\8\end{bmatrix}$	$\xrightarrow[\langle 0,0\rangle]{\begin{bmatrix}8\\8\\8\end{bmatrix},0} \mathbf{0}$	$\xrightarrow[\langle 0,0\rangle]{\mathbf{0},0} \mathbf{0}$
$\begin{bmatrix}9\\13\\9\end{bmatrix}$	$\begin{bmatrix}7\\0\\0\end{bmatrix} \xrightarrow[\langle 0,\gamma\rangle]{\mathbf{0},-2} \begin{bmatrix}7\\0\\0\end{bmatrix}$	$\xrightarrow[\langle \beta,0\rangle]{\mathbf{0},-2} \begin{bmatrix}8\\8\\8\end{bmatrix}$	$\xrightarrow[\langle \alpha,0\rangle]{\begin{bmatrix}8\\10\\8\end{bmatrix},-2} \begin{bmatrix}8\\0\\8\end{bmatrix}$	$\xrightarrow[\langle 0,0\rangle]{\begin{bmatrix}8\\0\\8\end{bmatrix},0} \mathbf{0}$	$\xrightarrow[\langle 0,0\rangle]{\mathbf{0}} \mathbf{0}$

Table 31. The best differential paths through rounds 5 to 16 using three active bits. The probability for these paths is 2^{-10}. The first column lists the message pattern numbers (from Table 7) for bit positions 63, 31 and 20 respectively. In the table entries, the values above the arrows are: (above the line) the roundkey differences for bit positions 63, 31 and 20 and the probability factor for that round; and (below the line) the class. The values at the same level at the arrows list the bit differences in the state for bit positions 63, 31 and 20 respectively. Where all roundkey differences or state are zero, a boldface zero "**0**" is used to save space.

9 Differential Analysis of TIB3v2

TIB3v2 differs from TIB3 only in the computation of the variable A_e. For TIB3v2, the variable A_e is computed as

$$A_e := ((G_d \lll 37) \oplus (A_d \lll 12) \oplus (A \gg 1).$$

This change provides additional diffusion between bit positions, and is a positive improvement over TIB3.[7]

However, if we can find paths that avoid changes in A_d and G_d, then this tweak has no effect on these differential paths - that is, we can take such paths from our analysis of TIB3, and apply these paths directly to TIB3v2. This restricts the set of differential paths to those with $\hat{\Delta}_d$ having differences in C_d and E_d only. The corresponding output differences $\hat{\Delta}_i$ are shown in Table 32.

$\hat{\Delta}_d$	$\hat{\Delta}_i$
$0 \sim (-,-,-,-)$	$0 \sim (-,-,-,-)$
$2 \sim (-,-,*,-)$	$6 \sim (-,*,*,-)$
$4 \sim (-,*,-,-)$	$8 \sim (*,-,-,-)$
$6 \sim (-,*,*,-)$	$14 \sim (*,*,*,-)$

Table 32. The values of $\hat{\Delta}_d$ and $\hat{\Delta}_i$ of this analysis that are applicable to TIB3v2.

Our existing analysis can be applied where we restrict ourselves to one-round differential paths with output differences $\hat{\Delta}_i \in \{0, 6, 8, 14\}$. Since we want to construct multi-round differential paths, we will also restrict ourselves to one-round differential paths with input differences $\hat{\Delta}_{i-1} \in \{0, 6, 8, 14\}$ and output differences $\hat{\Delta}_i \in \{0, 6, 8, 14\}$.

Note: There will be many other one-round differential paths of TIB3v2 that we could examine if we do allow differences in A_d and G_d, but we have not investigated these here since the diffusion provided by the rotations is not compatible with our current analysis technique.

9.1 Differential paths for TIB3v2 with One Active Bit

The set of one-round differential paths with one active bit that we considered for TIB3 allowed only class $\langle 0, 0\rangle$. Table 33 shows the one-round differential paths using this class. Only the 'trivial" one-round differential path can be applied, so

[7] The original specification of the tweak used $(A_d \lll 13)$, but we have been informed by the TIB3 designers that the rotation is now changed to $(A_d \lll 12)$ as shown in this document.

there is nothing to gain from investigating multi-round differential paths with one active bit. The tweak has succeeded in removing all possible 16-round differential paths with one active bit.

$\hat{\Delta}_i$	0	6
Class	$\langle 0,0 \rangle$	$\langle 0,0 \rangle$
$\hat{\Delta}_{i-1}$	$\hat{\Delta}K_i$ s.t. $\hat{\Delta}_{i-1} \stackrel{\hat{\Delta}K_i}{\rightarrow} \hat{\Delta}_i$	
0	0	-
6	-	-
Pr.	0	-

Table 33. The "input-output-roundkey" representation of the one-round differential paths (applicable to TIB3v2) when considering one active bit position.

9.2 Differential paths for TIB3v2 with Two Active Bits

Differential paths using two active bits may only use the classes $\langle 0,0 \rangle$, $\langle 0,\gamma \rangle$, $\langle \gamma,0 \rangle$, $\langle \gamma,\gamma \rangle$. Recall that only output differences $\hat{\Delta}_i \in \{0, 6, 8, 14\}$ are of interest. The classes $\langle 0,\gamma \rangle$ and $\langle \gamma,\gamma \rangle$ never output these values in bit position 31, so these classes can be ignored. Hence, only classes $\langle 0,0 \rangle$ and $\langle \gamma,0 \rangle$ are applicable. Table 34 and Table 35 shows the one-round differential paths that are applicable for TIB3v2 at bit position 63 and 31.

$\hat{\Delta}_i[63]$	0		6	
Class	$\langle 0,0 \rangle$	$\langle \gamma,0 \rangle$	$\langle 0,0 \rangle$	$\langle \gamma,0 \rangle$
$\hat{\Delta}_{i-1}[63]$	$\hat{\Delta}K_i[63]$ s.t. $\hat{\Delta}_{i-1}[63] \stackrel{\hat{\Delta}K_i[63]}{\rightarrow} \hat{\Delta}_i[63]$			
0	0	2,10	-	2,10
6	-	5	-	5
Pr.	0	-2	-2	-2

Table 34. The "input-output-roundkey" representation of the one-round differential paths (applicable to TIB3v2) for bit position 63 when considering two active bit positions.

$\hat{\Delta}_i[31]$	0	8	6	14
Class	$\langle 0,0\rangle$	$\langle\gamma,0\rangle$	$\langle 0,0\rangle$	$\langle\gamma,0\rangle$
$\hat{\Delta}_{i-1}[31]$	$\hat{\Delta}K_i[31]$ s.t. $\hat{\Delta}_{i-1}[31] \overset{\hat{\Delta}K_i[31]}{\rightarrow} \hat{\Delta}_i[31]$			
0	0	2,10	-	2,10
6	7	5,13	-	5,13
8	8	10,2	-	10,2
14	15	13,5	-	13,5
Pr.	0	-2	-2	-2

Table 35. The "input-output-roundkey" representation of the one-round differential paths (applicable to TIB3v2) for bit position 31 when considering two active bit positions.

9.3 Multi-round differential paths of TIB3v2 with Two Active Bit Positions.

There are 8192 message patterns for this case, as discussed at the beginning of Section 7.2. An automated search examined all these message patterns using the one-round differential paths allowed for TIB3v2, and only one differential path was found from round 5 to 16. This differential path uses message pattern 13 for bit position 63 and message pattern 5 for bit position 31: this corresponds to the differential from rounds 3 to 16 noted in Section 7.2, which has probability $2^{-4} = \frac{1}{16}$.

It is unfortunate (for the designers) that this differential we found for TIB3v2 is the best differential we have found thus far. This is this is the only differential (in the set of differential paths we consider) that can be applied to TIB3v2. As mentioned above, it is unclear if this differential can be used in an attack.

Continuing the Differential Path Through Rounds 1 and 2 of TIB3v2. None of the one-round differential paths available for TIB3v2 allow output difference $\hat{\Delta}_2[63,31] = (0;4)$. However, we can try to trace the differential path backwards. We begin by tracing the differences backwards through Round 2 from $\hat{\Delta}_2$ to $\hat{\Delta}_d$:

$$\begin{aligned} \hat{\Delta}_2 &= (\quad - \ , \ 31, - \ , \ - \ ), \\ \Rightarrow \hat{\Delta}_e &= (\quad - \ , - \ , \ 31, \ - \ ), \\ \Rightarrow \hat{\Delta}_d &= (\ \hat{\Delta}A_d^2, - \ , \ 31, 31 \ ), \end{aligned}$$

where $\hat{\Delta}A_d^2$ is the differences such that $(\hat{\Delta}A_d^2 \lll 13) \oplus (\hat{\Delta}A_d^2 \gg 1) = (\hat{\Delta}G_d \lll 37)$ = a single bit differences in bit position 3. The difference $\hat{\Delta}A_d^2$ has differences at 14 positions: 62,56,54,48,42,40,34,28,26,20,14,12,6,0.

Continuing to trace backwards through Round 2:

– $\hat{\Delta}_c = (\hat{\Delta}A_d^2, -, 31, (63, 31))$, where the difference $\hat{\Delta}G_c = (63, 31)$ is the input difference to the *PHTX* to get $\hat{\Delta}G_d = 31$.
– There are $4^{15} = 2^{30}$ options for $\hat{\Delta}_b$ according to the choice of differential paths through the S-Box at each of the 15 bit positions with differences. Each option holds with probability $(2^{-2})^{15} = 2^{-30}$. Each of these options must have $\hat{\Delta}A_b$ including the bit differences from $\hat{\Delta}A_d^2$, and $\hat{\Delta}C_b^2$ including difference 31 (due to differential properties of the S-box).

It is already clear that there are many potential paths through Round 2, and the number of paths back through Round 1 will be very large. The difference $\hat{\Delta}A_d^2$ must pass through the *PHTX* in Round 1 with (currently unknown) effect which may result in many additional differences that must go through the S-box. A new difference $\hat{\Delta}A_d^1$ in Round 1 (analogous to $\hat{\Delta}A_d^2$) is also generated, and those bit difference must pass through the S-boxes. A quick guess is that there are at least 2^{64} possible input hash state differences $\hat{\Delta}h_i$ that result in the correct value of $\hat{\Delta}_2[63, 31] = (0; 4)$.

9.4 One-Round Differential paths for TIB3v2 with Three Active Bits

Recall that we only consider differential paths with input differences $\hat{\Delta}_{i-1} \in \{0, 6, 8, 14\}$, and output differences $\hat{\Delta}_i \in \{0, 6, 8, 14\}$. Table 36 shows the one-round differential paths at bit position 20 that are applicable for TIB3v2. This table shows that the only roundkey differences 0 and 8 can be used at bit position 20 when applying the analysis to TIB3v2.

k	$\hat{\Delta}_i$ possible in T^k	
0	0	6
2	8	14
$\hat{\Delta}_{i-1}$	$\hat{\Delta}K_i$ s.t. $\hat{\Delta}_{i-1} \xrightarrow{\hat{\Delta}K_i} \hat{\Delta}_i$	
0	0	-
6	-	-
8	8	-
14	-	-
Pr.	0	-

Table 36. The "input-output-roundkey" representation of the one-round differential paths (applicable to TIB3v2) for bit position 20. The rows correspond to input differences (at the active bit position), the columns correspond to output differences, and the table entries list the key differences that allow this differential to occur. The probability factor (provided logarithm base 2) is shown in the last row.

When considering one-round differential paths at bit position 31 that are applicable for TIB3v2, we see that the restriction to the output difference $\hat{\Delta}_i \in \{0, 6, 8, 14\}$, invokes the additional restrictions that only classes in group U^0 or group U^2 allow differential paths for TIB3v2. This leaves only four possible classes: $\langle 0, 0\rangle$, $\langle \alpha, 0\rangle$, $\langle \beta, 0\rangle$, $\langle \gamma, 0\rangle$. Table 37 and Table 38 show the resulting one-round differential paths that are applicable for TIB3v2 at bit positions 63 and 31 respectively.

0	$\hat{\Delta}_i$	$0_{0,-2}$		$6_{all\ -2}$	
	Cls	0,0	$\gamma, 0$	0,0	$\gamma, 0$
2	$\hat{\Delta}_i$	$8_{all\ -2}$		$14_{all\ -2}$	
	Cls	$\alpha, 0$	$\beta, 0$	$\alpha, 0$	$\beta, 0$
$\hat{\Delta}_{i-1}$		$\hat{\Delta}K_i$ s.t. $\hat{\Delta}_{i-1} \xrightarrow{\hat{\Delta}K_i} \hat{\Delta}_i$			
0		0	2,10	-	2,10
6		-	5	-	5
8		8	10,2	-	10,2
14		-	5	-	5
Pr.		0	-2	-2	-2

Table 37. The "input-output-roundkey" representation of the one-round differential paths (applicable to TIB3v2) for bit position 63. See the text accompanying Table 29 for an explanation of this table.

0	$\hat{\Delta}_i$	0		6	
	Cls	$0, 0$	$\alpha, 0$	$0, 0$	$\alpha, 0$
2	$\hat{\Delta}_i$	8		14	
	Cls	$\beta, 0$	$\gamma, 0$	$\beta, 0$	$\gamma, 0$
$\hat{\Delta}_{i-1}$		$\hat{\Delta}K_i$ s.t. $\hat{\Delta}_{i-1} \stackrel{\hat{\Delta}K_i}{\rightarrow} \hat{\Delta}_i$			
0		0	2,10	-	2,10
6		7	5,13	-	5,13
8		8	10,2	-	10,2
14		15	13,5	-	13,5
Pr.		0	-2	-2	-2

Table 38. The "input-output-roundkey" representation of the one-round differential paths (applicable to TIB3v2) for bit position 31. See the text accompanying Table 29 for an explanation of this table.

9.5 Multi-Round Differential paths of TIB3v2 with Three Active Bit Positions

There are $2^{17} = 131072$ possible message patterns for this case, as discussed at the beginning of Section 8.2. An automated search examined all these message patterns using the one-round differential paths allowed for TIB3v2. No additional differential paths were found: only the differential path found was the differential path already mentioned in Section 9.3.

10 Conclusion

New differential properties of the TIB3 hashing function have been illustrated, resulting in new differential paths with higher probabilities than previously reported. The results were extended to TIB3v2 - the tweaked version of TIB3.

The best path for rounds 5 to 16 of both TIB3 and TIB3v2 has probability $2^{-4} = \frac{1}{16}$. For TIB3, there are several other paths for rounds 5 to 16 with probability 2^{-10}. No other paths for TIB3v2 were found in this analysis, so the tweak can be considered quite effective.

The differential paths found in this analysis are not sufficient to mount a collision attack. Hence, while these properties demonstrate weaknesses we do not claim that these observations lead to a direct attack on either TIB3 or TIB3v2.

This analysis (and the analysis of Mendel and Schläffer [1]) indicate that the message expansion and the PHTX do not provide sufficient diffusion between bit positions. We would like to mention that the choice of S-box and the choice of the XOR and addition operations between internal values frustrated our search significantly by preventing many desirable differential paths. We would like to commend the designers on this feature of the TIB3 design.

Areas for Further Research. In the text, some areas of further research were noted:

- We have recently noticed that if the input to the *PHTX* has differences in both bit positions 52 and 20, then there is an output differences only in bit 20 with probability $\frac{1}{2}$. A quick examination showed that this dramatically increased the number of possible one-round differential paths. We have not had time to explore this avenue further, but hope to examine the impact on TIB3v2 in the near future.
- Message patterns $17-32$ also apply to other bit positions where the ψ observation holds (bit positions 21-30). However, a probability factor must then be applied when propagating this difference trough the message expansion. These options have not been explored.
- This paper focusses on TIB3-256. The message expansion and the PHTXD of TIB3-512 have similar differential properties that could be exploited to find differential paths using the same techniques. However, more emphasis should be placed on analysis TIB3v2. The current techniques have proven inadequate for finding differential paths of TIB3v2, so new avenues must be explored.

References

1. Florian Mendel, Martin Schläffer, On Pseudo-Collisions and Collisions for TIB3, `http://ehash.iaik.tugraz.at/uploads/2/2b/Tib3-pseudo.pdf`.
2. National Institute of Standards and Technology. Announcing Request for Candidate Algorithm Nominations for a New Cryptographic Hash Algorithm (SHA-3) Family. *Federal Register*, 27(212) : 62212 – 62220, November 2007. Available

at http://csrc.nist.gov/groups/ST/hash/documents/FR_Notice_Nov07.pdf (2008/10/17).
3. Daniel Penazzi, Miguel Montes, “The TIB3 Hash”, 2008. See http://ehash.iaik.tugraz.at/wiki/TIB3. Submissions package available from http://csrc.nist.gov/groups/ST/hash/sha-3/Round1/documents/TIB3.zip.
4. Daniel Penazzi, Miguel Montes, “Tweak of TIB3”, 2009. http://www.famaf.unc.edu.ar/~penazzi/tib3/TweakofTIB3.

A Example Input Pairs Following Differential Paths

This appendix provides example values for pairs of inputs such that the internal differences conform to the differential paths for rounds 5-16 listed in this document. The practical examples are presented in the following format:

- `In1` (`In2`) is the input to Round 5 for the first (second) set of inputs.
- `LK1` (`LK2`) is the value of LK for the first (second) set of inputs.
- `RK1` (`RK2`) is the value of RA for the first (second) set of inputs.
- `Out1` (`Out2`) is the output from Round 16 for the first (second) set of inputs.

The values are provided in columns in hexadecimal format.

A.1 Message Patterns (13,5) for Bit Positions (63,31) for TIB3 and TIB3v2

Pr.	Bit j	Message Pattern	$\Delta_4[j]$ $A\ C\ E\ G$	$\Delta_{16}[j]$ $A\ C\ E\ G$
2^{-4}	63	$\begin{bmatrix} 13 \\ 5 \end{bmatrix}$	$-\ -\ -\ -$	$-\ -\ -\ -$
	31		$-\ -\ -\ -$	$-\ -\ -\ -$

The first input pair conforms when using TIB3:

```
In1 = In2
af5b36783ec71ddf
fd4cf5035f307f87
1a04a18424388e02
30658004cab21af4
LK1              LK2
658f1880827333cf e58f1880827333cf
b0628114a0a29e0d 3062811420a29e0d
0fee4074d16a903b 0fee4074516a903b
da8b8b2038d22dc1 da8b8b2038d22dc1
d1ffc29b947476fe d1ffc29b947476fe
0902002206a07ec7 0902002206a07ec7
847627dc967df7b9 847627dc967df7b9
cca1bde64073556e cca1bde64073556e
RK1              RK2
fe4f837c4038cd05 7e4f837c4038cd05
ffbfb28d487e5c4d 7fbfb28dc87e5c4d
facebbe8e84ce2dc facebbe8684ce2dc
ed54b99607296366 ed54b99607296366
a36402f24e1d138e a36402f24e1d138e
eb3b1a0423917433 eb3b1a0423917433
c3b799ac3e073723 c3b799ac3e073723
8821c71a59e52203 8821c71a59e52203
Out1=Out2
f7fe2fa10ec4d267
7ecd4d1151e86bfc
993f9edfd9b0a465
8634bf9308268f6a
```

The next input pair conforms when using TIB3v2:

```
In1 = In2
cb755a095c635f21
e148462be30abd34
97bff330a352dfd6
21e43ebd4ef9dd16
LK1              LK2
4b4bc3384335c702 cb4bc3384335c702
4aeea198d80164b5 caeea198580164b5
a011b82e626d7c7c a011b82ee26d7c7c
9eecc1558b2057aa 9eecc1558b2057aa
bc884210f866ad20 bc884210f866ad20
2466d05b4c4f1dd0 2466d05b4c4f1dd0
ff03f4ca93b84947 ff03f4ca93b84947
762bb99771598e27 762bb99771598e27
RK1              RK2
5a8a7609e7e24388 da8a7609e7e24388
259b54033b297273 a59b5403bb297273
498bce9693eb3e74 498bce9613eb3e74
a6ed1cf24f29791d a6ed1cf24f29791d
d311400cf1aaa22f d311400cf1aaa22f
593225a407883273 593225a407883273
5420826087411b53 5420826087411b53
a9c121d8c284208e a9c121d8c284208e
Out1 = Out2
a87817c9c777506c
55d42a11f1509059
575cab161f0ea45c
4a64133eb31e9208
```

A.2 Message Patterns (5,8) and (13,8) for Bit Positions (63,31)

Pr.	Bit	Message Pattern	$\Delta_4[j]$	$\Delta_{16}[j]$
	j	Options	$A\ C\ E\ G$	$A\ C\ E\ G$
2^{-10}	63 31	$\begin{bmatrix} 5 \\ 8 \end{bmatrix}, \begin{bmatrix} 13 \\ 8 \end{bmatrix}$	$-\ -\ -\ *$ $-\ -\ -\ -$	$-\ -\ -\ -$ $-\ -\ -\ -$

A practical example with message patterns $(5, 8)$ for bit positions (63,31).

```
In1=In2
87e0c109899f44f8
2326edc8820afc08
6894e07ff0e01143
34cd4a4b5f6eff79
LK1              LK2
d42cf548ece5e091 d42cf548ece5e091
4ec1addd565e571a cec1adddd65e571a
5b4368988d8355b2 db4368980d8355b2
5aa736a13a64d483 5aa736a13a64d483
7660ed82d6515534 7660ed8256515534
f09eede20db75a5e f09eede28db75a5e
16a88f7edc13bab1 16a88f7edc13bab1
f3b82bc2be996306 f3b82bc2be996306
RK1              RK2
47fbdaed98216280 47fbdaed98216280
62475b69b095873b e2475b693095873b
c3a402c2b750a3df 43a402c23750a3df
c6372e1ef07817a8 c6372e1ef07817a8
17368ad1f2020e99 17368ad172020e99
f02e372cce1ed241 f02e372c4e1ed241
3fa263e2fdc51ddb 3fa263e2fdc51ddb
acaf329208ed8eac acaf329208ed8eac
Out1=Out2
9a73d47b064994ba
19fe80e7c76c7b99
75732424dee51fb2
ca8427980e2c8b97
```

A practical example with message pattern $(13, 8)$ for bit positions (63,31).

```
In1=In2
9e8affbb014c4412
51be84fcff64050d
41167523d9d9fc79
9936ca4433c6549a
LK1              LK2
ee2561c1436f56ec 6e2561c1436f56ec
5687055576ef5809 d6870555f6ef5809
c3db2f1232608f71 c3db2f12b2608f71
ca2ac90e9a93070d ca2ac90e9a93070d
da10af5cc6ea0383 da10af5c46ea0383
30384b8cea788a9b 30384b8c6a788a9b
0eec50e3018d6ff1 0eec50e3018d6ff1
b4861b86898f1548 b4861b86898f1548
RK1              RK2
bd2d75bada7720a8 3d2d75bada7720a8
d29b5f24a74dfd18 529b5f24274dfd18
f3fe10bb434c327d f3fe10bbc34c327d
e9d4917a5ff437a7 e9d4917a5ff437a7
e65da68781fe09d5 e65da68701fe09d5
8f9a04dfa28c901e 8f9a04df228c901e
d2ca15a13020f463 d2ca15a13020f463
0784604be75ac8fb 0784604be75ac8fb
Out1=Out2
365529c0ed782585
d51fe1b15da2d5b0
3f7593f2a55761ae
054270e9b48f08c2
```

A.3 Message Patterns (5,9,9) and (9,13,9) for Bit Positions (63,31,20)

Pr.	Bit	Message Pattern	$\Delta_4[j]$	$\Delta_{16}[j]$
	j	Options	$A\ C\ E\ G$	$A\ C\ E\ G$
2^{-18}	63	5	$-\ *\ *\ *$	$-\ -\ -\ -$
	31	9 ,	$-\ -\ -\ -$	$-\ -\ -\ -$
	20	9	$-\ -\ -\ -$	$-\ -\ -\ -$

A practical example with message pattern $(9, 13, 9)$ for bit positions (63,31,20).

```
In1              In2
8541b2c613041ca9 8541b2c613041ca9
766ffa68c5a33ea5 d66ffa68c5a33ea5
2b8ae88589efc28f ab8ae88589efc28f
165b27841b6b529a 965b27841b6b529a
LK1              LK2
0fb07b541b862b83 8fb07b549b962b83
320c45f5f0aea35a 320c45f570aea35a
5ce9846705d4f4ab dce9846705c4f4ab
7e2c499b3c5c636f 7e2c499b3c5c636f
bbd6ef0d5e11c718 bbd6ef0d5e11c718
0ac47693e5b57bf3 0ac47693e5b57bf3
6092d2a0aa20d9b2 6092d2a0aa20d9b2
d2f9b727978465a7 d2f9b727978465a7
RK1             RK2
932322defb6322c9 132322de7b7322c9
4e100f97fcecfcf2 4e100f977cecfcf2
486a7961c224d3fe c86a7961c234d3fe
d74ad8078d4b6883 d74ad8078d4b6883
0d14082b585a7f0f 0d14082b585a7f0f
8b1a6ea01e7d7698 8b1a6ea01e7d7698
432aca61c0016ec8 432aca61c0016ec8
d0c95fb020cd1dca d0c95fb020cd1dca
Out1 = Out2
e62ec03a60168d45
49fae1a93bfb4420
2c19beea7b92ee6e
7792812037bda2f3
```

A practical example with message pattern $(5, 9, 9)$ for bit positions (63,31,20).

```
In1              In2
59240d89b1a41aba 59240d89b1a41aba
0464d0b28740e47c 8464d0b28740e47c
a507ae797dd8780a 2507ae797dd8780a
a753cf06a6ef0200 2753cf06a6ef0200
LK1              LK2
40b1faa12bbb63a8 40b1faa1abab63a8
5df47f8ab965b77c ddf47f8ab965b77c
f28ae98aef28068f 728ae98a6f38068f
e5b92df17904dd41 e5b92df17904dd41
b4b33b3f895034f8 b4b33b3f895034f8
22cb51218a9c584c 22cb51218a9c584c
45985da211538b3c 45985da211538b3c
a6023e20733435be a6023e20733435be
RK1              RK2
8401fc33541ffec2 8401fc33d40ffec2
88fbcd6dc2a5cd74 08fbcd6dc2a5cd74
d7b8b1c02481cd15 57b8b1c0a491cd15
bfc25d8ca7619ca0 bfc25d8ca7619ca0
605ef4a9b3d60cf8 605ef4a9b3d60cf8
78ccedb5bd87933e 78ccedb5bd87933e
5a908ce336c7f7b9 5a908ce336c7f7b9
477cc1324958f7bf 477cc1324958f7bf
Out1 = Out2
3fc480280ee64384
31a51945522750e9
7d8e5c01e78b144d
c9591c9db84e0fe1
```

A.4 Message Patterns Requiring Differences in bits 63 and 31 to Cancel

Pr.	Message Pattern Options for (63+31, 63,31)	Bit j	$\Delta_4[j]$ A C E G	$\Delta_{16}[j]$ A C E G
2^{-10}	(34,9,4), (34,5,4), (34,9,14), (43,13,14), (43,5,4)	63	$-$ $-$ $-$ $*$	$-$ $-$ $-$ $-$
		31	$-$ $-$ $-$ $-$	$-$ $-$ $-$ $-$

A practical example using message pattern 34 (with differences in bit positions 63 and 31) and message patterns $(9,4)$ for bit positions $(63,31)$.

```
In1              In2
71707c30c007b4e3 71707c30c007b4e3
d912d5a8d7e02911 d912d5a8d7e02911
5dec738722935c4b 5dec738722935c4b
a0908c0e39979baa 20908c0e39979baa
LK1              LK2
7606643552a44bbd f6066435d2a44bbd
28649425d06c4026 28649425d06c4026
e8d5b9aa062a72cb 68d5b9aa062a72cb
75ea72c6d15a4a54 f5ea72c6d15a4a54
3479e19c563f9d22 3479e19cd63f9d22
ec9155f9f7e8b66f ec9155f977e8b66f
b9174b0b9e2cb3f3 b9174b0b9e2cb3f3
af83114736ab0c03 af83114736ab0c03
RK1              RK2
3fe76e00e032a4be bfe76e006032a4be
13681f7f746dd12e 13681f7f746dd12e
e7a5697db45f1e51 67a5697db45f1e51
feee50a28761349c 7eee50a28761349c
02968839e837464e 029688396837464e
ce1f7b9e314a34d6 ce1f7b9eb14a34d6
b315e2abd10147e0 b315e2abd10147e0
b12681a6f8b7e2d7 b12681a6f8b7e2d7
Out1 = Out2
e0250f145ca64e2e
285edabcbbce5c12
69b75bde7b7f30d3
56c1917c9685fb07
```

A practical example using message pattern 34 (with differences in bit positions 63 and 31) and message patterns $(5, 4)$ for bit positions $(63, 31)$.

```
In1              In2
2ca4185361e1d998 2ca4185361e1d998
ba5c29f5091148a0 ba5c29f5091148a0
df0e5bd4a97471b3 df0e5bd4a97471b3
76f95163c7220a2b f6f95163c7220a2b
LK1              LK2
c7a0c3ff88869226 c7a0c3ff08869226
c776f67b7c9ec0ca 4776f67b7c9ec0ca
53443e15519391f9 d3443e15519391f9
c7ef7efc7f70dd36 47ef7efc7f70dd36
5b616d2aa8f094de 5b616d2a28f094de
ac35053466cba6d0 ac350534e6cba6d0
b239477af2af0066 b239477af2af0066
e5885792782a19f3 e5885792782a19f3
RK1              RK2
4ac46cae43bc5c3e 4ac46caec3bc5c3e
65325d64d3e89c19 e5325d64d3e89c19
3d340c84547b4f1a bd340c84547b4f1a
4304b6a3c34c90eb c304b6a3c34c90eb
1eb84447be6d9a8c 1eb844473e6d9a8c
5aa77ef0290c1ff3 5aa77ef0a90c1ff3
1429c5ccdf8e77a4 1429c5ccdf8e77a4
482168dd9357fc95 482168dd9357fc95
Out1 = Out2
fa9df8c2ae3cb844
5dde0fdb9f439779
a27a5195c139eaa9
9cda80ac2402b63e
```

A practical example using message pattern 34 (with differences in bit positions 63 and 31) and message patterns $(5, 14)$ for bit positions $(63, 31)$.

```
In1              In2
8ddd75a46296c655 8ddd75a46296c655
91cbd6e50f339bee 91cbd6e50f339bee
3ceef3c760267a23 3ceef3c760267a23
34c46f5a60c56edb b4c46f5a60c56edb
LK1              LK2
058b9ed1c6928ae0 858b9ed1c6928ae0
071c6729d6e73698 071c672956e73698
82f65c65dd971c14 02f65c65dd971c14
6fdbab75338b70b8 efdbab75338b70b8
5d0a9d1601b85cfe 5d0a9d1681b85cfe
5bf0bd098f0754a8 5bf0bd090f0754a8
f9921c51037ab84f f9921c51037ab84f
b720f2d9261f2af1 b720f2d9261f2af1
RK1              RK2
3a96c07edea25111 ba96c07edea25111
503e3c6caa2232ea 503e3c6c2a2232ea
2c712b525e111535 ac712b525e111535
afa1135d8b4fcfca 2fa1135d8b4fcfca
cfd1401d02d4788e cfd1401d82d4788e
8da0cbe517991c5e 8da0cbe597991c5e
d598fd06b9775af2 d598fd06b9775af2
ab4d766541fdac56 ab4d766541fdac56
Out1 = Out2
602b211a921a487e
da47e75c2d72dc14
f758df3b0c7ec63b
024830b920a79946
```

A practical example using message pattern 42 (with differences in bit positions 63 and 31) and message patterns $(13, 14)$ for bit positions $(63, 31)$.

```
In1              In2
e95814f8a8640366 e95814f8a8640366
389707fc2c916044 389707fc2c916044
41ad1993cd82e442 41ad1993cd82e442
a88a256159378feb 288a256159378feb
LK1              LK2
96617e2d3d1b4acf 96617e2d3d1b4acf
57b0448fac6eea11 d7b0448f2c6eea11
cf9c1c0e67914063 cf9c1c0e67914063
4747a6ed42a21f2f c747a6edc2a21f2f
ece66b2773d1e845 ece66b27f3d1e845
6f493df9c535c718 6f493df94535c718
400d8ac773ce02cb 400d8ac773ce02cb
7618dff630da4940 7618dff630da4940
RK1              RK2
fdf6174c20d5ed4b fdf6174c20d5ed4b
fb6e9d86dfa8dfd4 7b6e9d865fa8dfd4
61a9d23ec7d65f11 61a9d23ec7d65f11
617d5c517a8d928b e17d5c51fa8d928b
e92c486deff41ac5 e92c486d6ff41ac5
d633a62988713da5 d633a62908713da5
fe37457898c8685d fe37457898c8685d
b81a9fcab7816514 b81a9fcab7816514
Out1 = Out2
ce6f55bf8716d3e5
c935bff32acad954
516905da456c08c4
0647f0f60cc2bfec
```

A practical example using message pattern 42 (with differences in bit positions 63 and 31) and message patterns $(5, 4)$ for bit positions $(63, 31)$.

```
In1              In2
09f2181aae88cf4c 09f2181aae88cf4c
15cfaf6bca0ca1c2 15cfaf6bca0ca1c2
4401f9a1d5f307cb 4401f9a1d5f307cb
4e9ece4f15cd9ed2 ce9ece4f15cd9ed2
LK1              LK2
c09f62aedf0ad141 409f62ae5f0ad141
226eeff6e7842a49 a26eeff6e7842a49
2e35047d970e5d08 ae35047d970e5d08
05e2865ee7e5b9a8 85e2865e67e5b9a8
b5ef971b68309604 b5ef971be8309604
0abfeb0d1dc558ab 0abfeb0d9dc558ab
03204b7771428a93 03204b7771428a93
fde9ab5e38516f3f fde9ab5e38516f3f
RK1              RK2
d6f2d0602bebf2b9 56f2d060abebf2b9
23d532aeac2bd90b a3d532aeac2bd90b
dd996612c07435b8 5d996612c07435b8
e86188f97e63172e 686188f9fe63172e
0926207508365a41 0926207588365a41
2bffe3c95838124c 2bffe3c9d838124c
05a2337b0c672fd3 05a2337b0c672fd3
caebaede4a3cd40f caebaede4a3cd40f
Out1 = Out2
d28173ad495838b3
cf24ed056bd0e465
a672ce5cc4c82892
7beff8e3f88f4ab2
```