A Differential Fault Attack on MICKEY 2.0

Subhadeep Banik and Subhamoy Maitra

Applied Statistics Unit, Indian Statistical Institute,
203 B T Road, Kolkata 700 108, India,
{s.banik_r, subho}@isical.ac.in

Abstract. In this paper we present a differential fault attack on the stream cipher MICKEY 2.0 which is in eStream's hardware portfolio. While fault attacks have already been reported against the other two eStream hardware candidates Trivium and Grain, no such analysis is known for MICKEY. Using the standard assumptions for fault attacks, we show that by injecting around $2^{16.7}$ faults and performing $2^{32.5}$ computations on an average, it is possible to recover the entire internal state of MICKEY at the beginning of the key-stream generation phase.

Keywords: eStream, Fault attacks, MICKEY 2.0, Stream Cipher.

1 Introduction

The stream cipher MICKEY 2.0 [4] was designed by Steve Babbage and Matthew Dodd as a submission to the eStream project. The cipher has been selected as a part of eStream's final hardware portfolio. MICKEY is a synchronous, bit-oriented stream cipher designed for low hardware complexity and high speed. After a TMD tradeoff attack [15] against the initial version of MICKEY (version 1), the designers responded by tweaking the design by increasing the state size from 160 to 200 bits and altering the values of some control bit tap locations. These changes were incorporated in MICKEY 2.0 and these are the only differences between MICKEY version 1 and MICKEY 2.0. While MICKEY 2.0 uses an 80-bit key and a variable length IV, a modified version of the cipher, MICKEY-128 2.0 that uses a 128-bit key [5] was also proposed by the designers.

The name MICKEY is derived from "Mutual Irregular Clocking Key-stream generator" which describes the behavior of the cipher. The state consists of two 100-bit shift registers named R and S, each of which is irregularly clocked and controlled by the other. The cipher specification underlines that each key can be used with up to 2^{40} different IVs of the same length, and that 2^{40} key-stream bits can be generated from each key-IV pair. Very little cryptanalysis of MICKEY 2.0 is available in literature. In fact it has been noted in [3, Section 3.2] that other than the observation related to time or power analysis attacks [11] on straightforward implementations of the MICKEY family, there have been no known cryptanalytic advances on these ciphers. To the best our knowledge, the work in this paper presents the first cryptanalytic result of MICKEY 2.0 in terms of differential fault attack.

Since the work of [6, 7], fault attacks have been employed to test the strengths/weaknesses of cryptographic primitives. Such attacks on stream ciphers was first described by Hoch and Shamir [12]. A typical fault attack [12] involves the random injection of faults (using laser shots/clock glitches [17, 18]) in a device (typically initialized by a secret key) which changes one or more bits of its internal state. The adversary then attempts to deduce information about the internal state/secret key using the output stream from this faulty device. In order to perform the attack, certain privileges are required like the ability to re-key the device, control the timing of the fault etc. The attack becomes impractical and unrealistic if the adversary is granted too many privileges. In this work we assume the following privileges of the adversary which are generally acceptable in cryptanalytic literature:

1. She can re-key the cipher with the original key-IV and restart cipher operations multiple times.
2. She has full control over the timing of fault injection.
3. She can inject a fault that alters the bit value of one randomly chosen register location in either the R or the S register.
4. She is however unable to fix the exact location of the R or S register where she wants to inject the fault. Obtaining the fault location by comparison of the fault-free and the faulty key-streams is one of the challenges while mounting the fault attack.

As has been previously mentioned, these assumptions do not ask for more privileges than the existing works [9,13]. In fact, there are some published works where the assumptions made are quite strong, e.g., the works [8,10,16] considers that the attacker can reproduce multiple faults in the same (but unknown) locations, that we do not need to assume.

Differential fault attack is a special class of fault attack in which the attacker uses the difference between fault-free and faultless key-streams to deduce the internal state or the secret key of the cipher. In case of MICKEY 2.0, the differential attack is possible due to the rather simplistic nature of the output function $(r_0 + s_0)$ used to produce key-stream bits. Additionally, there are some interesting properties of the state update function in MICKEY that help facilitate the attack that we shall describe. The organization of the paper is as follows. In Section 2, we present a description of the cipher which is suitable for our analysis, where we also present some notations that will be henceforth used in the paper. The complete attack is described in Section 3. Section 4 concludes the paper.

2 An alternate description of the MICKEY 2.0 PRGA and some notations

A detailed description of MICKEY 2.0 is available in [4]. For convenience of the reader we also describe it in Appendix A. MICKEY 2.0 uses an 80-bit key and a variable length IV, the length of which may be between 0 and 80 bits. The physical structure of the cipher consists of two 100 bit registers R and S. Both registers are initially initialized to the all-zero state, and the three stages of register update **1.** IV loading, **2.** Key Loading, and **3.** Pre Clock are executed sequentially before the production of the first key-stream bit. Thereafter in the PRGA (Pseudo Random bitstream Generation Algorithm) key-stream bits are produced. We will try to give an alternate description of this stage of operation of MICKEY 2.0. Consider a_0, a_1, a_2, a_3 to be variables over GF(2). Let a_0 be defined as follows

$$a_0 = \begin{cases} a_2, \text{ if } a_1 = 0 \\ a_3, \text{ if } a_1 = 1. \end{cases}$$

Then it is straightforward to see that a_0 can be expressed as a multivariate polynomial over GF(2), i.e., $a_0 = (1 + a_1) \cdot a_2 + a_1 \cdot a_3$. The state registers R and S, during the PRGA are updated by a call to the $CLOCK\_KG$ routine, which in turn calls the $CLOCK\_R$ and the $CLOCK\_S$ routine. In both these routines state update is done via a number of If-Else constructs. As a result of this the state update may be equivalently expressed as a series of multi-variate polynomials over GF(2). Let $r_0, r_1, \ldots, r_{99}, s_0, s_1, \ldots, s_{99}$ denote the internal state at a certain round during the MICKEY PRGA and let $r'_0, r'_1, \ldots, r'_{99}, s'_0, s'_1, \ldots, s'_{99}$ denote the internal state at the next round. Then it is possible to write

$$r'_i = \rho_i(r_0, r_1, \ldots, r_{99}, s_0, s_1, \ldots, s_{99}),\ s'_i = \beta_i(r_0, r_1, \ldots, r_{99}, s_0, s_1, \ldots, s_{99}),\ \forall i \in [0, 99]$$

where ρ_i, β_i are polynomial functions over GF(2). The exact forms of ρ_i, β_i are described in Appendix B. Before describing the attack we will describe certain notations that will be used henceforth.

1. $R_t = [r^t_0, r^t_1, \ldots, r^t_{99}], S_t = [s^t_0, s^t_1, \ldots, s^t_{99}]$ is used to denote the internal states of the R, S registers at the beginning of the round t of the PRGA. That is, $r^t_i,\ s^t_i$ respectively denotes the i^{th} bit of the registers $R,\ S$ at the beginning of round t of the PRGA. Note that $r^{t+1}_i = \rho_i(R_t, S_t)$ and $s^{t+1}_i = \beta_i(R_t, S_t)$.
2. The value of the variables $CONTROL\_BIT\_R,\ CONTROL\_BIT\_S$ at PRGA round t are denoted by the variables $CR_t,\ CS_t$ respectively. These bits are used by the R, S registers to exercise mutual self control over each other. Note that $CR_t = r^t_{67} + s^t_{34}$ and $CS_t = r^t_{33} + s^t_{67}$.
3. $R_{t,\Delta r_\phi}(t_0), S_{t,\Delta r_\phi}(t_0)$ (resp. $R_{t,\Delta s_\phi}(t_0), S_{t,\Delta s_\phi}(t_0)$) are used to denote the internal states of the cipher at the beginning of round t of the PRGA, when a fault has been injected in location ϕ of R (resp. S) at the beginning of round t_0 of the PRGA.
4. $z_{i,\Delta r_\phi}(t_0)$ or $z_{i,\Delta s_\phi}(t_0)$ denotes the key-stream bit produced in the i^{th} PRGA round, after a fault has been injected in location ϕ of R or S at the beginning of round t_0 of the PRGA. By z_i, we refer to the fault-free key-stream bit produced in the i^{th} PRGA round.

3 Complete description of the Attack

We will start with a few algorithmic tools that will be used later to mount the attack.

Lemma 1. *Consider the first* 100 *states of the MICKEY 2.0 PRGA. If* r^t_{99} *and* CR_t *are known* $\forall t \in [0, 99]$*, then the initial state* R_0 *of the register* R *can be determined efficiently.*

Proof. Let the values of r^t_{99} and CR_t be known $\forall t \in [0, 99]$. We will begin by noticing that the functions ρ_i for all values of $i \in [1, 99]$ are of the form

$$\rho_i(\cdot) = r_{i-1} + (s_{34} + r_{67}) \cdot \hat{\rho}_i(r_i, r_{i+1}, \ldots, r_{99}),$$

where $s_{34} + r_{67}$ is the value of $CONTROL\_BIT\_R$ and $\hat{\rho}_i$ is a function that depends on $r_i, r_{i+1}, \ldots, r_{99}$ but not any of $r_0, r_1, \ldots, r_{i-1}$. Now consider the following equation governing r^{99}_{99} :

$$r^{99}_{99} = \rho_{99}(R_{98}, S_{98}) = r^{98}_{98} + CR_{98} \cdot \hat{\rho}_{99}(r^{98}_{99}).$$

In the above equation, r^{98}_{98} is the only unknown and it appears as a linear term, and so its value can be calculated immediately. We therefore know the values of 2 state bits of R_{98}: $r^{98}_{99},\ r^{98}_{98}$. Similarly look at the equations governing $r^{98}_{99},\ r^{98}_{98}$.

$$r^{98}_{99} = r^{97}_{98} + CR_{97} \cdot \hat{\rho}_{99}(r^{97}_{99}),$$

$$r^{98}_{98} = r^{97}_{97} + CR_{97} \cdot \hat{\rho}_{98}(r^{97}_{98}, r^{97}_{99}).$$

As before, r_{98}^{97} is the lone unknown term in the first equation whose value is determined immediately. After this r_{97}^{97} becomes the only unknown linear term in the next equation whose value too is determined easily. Thus we know 3 bits of R_{97}: r_{97+i}^{97}, $i = 0, 1, 2$. Continuing in such a bottom up manner we can successively determine 4 bits of R_{96}, 5 bits of R_{95} and eventually all the 100 bits of R_0. The process is explained pictorially in Figure 1. $\square$

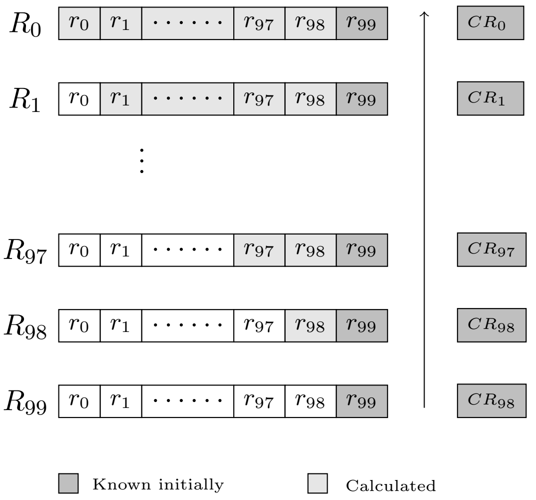

Fig. 1: Constructing the state R_0. Starting from PRGA round 99, any bit calculated at PRGA round i is used to determine state bits of round $i-1$.

Lemma 2. *Consider the first* 100 *states of the MICKEY 2.0 PRGA. If* R_0 *is known and* s_{99}^t, CS_t, CR_t *are known* $\forall t \in [0, 99]$, *then the initial state* S_0 *of the register* S *can be determined efficiently.*

Proof. Since R_0 is known and so is CR_t for each $t \in [0, 99]$ we can construct all the bits of R_1 by calculating

$$r_i^1 = r_{i-1}^0 + CR_0 \cdot \hat{\rho}_i(r_i^0, \ldots, r_{99}^0), \ \forall i \in [0, 99].$$

Once all the bits of R_1 are known, all the bits of R_2 may be determined by calculating

$$r_i^2 = r_{i-1}^1 + CR_1 \cdot \hat{\rho}_i(r_i^1, \ldots, r_{99}^1), \ \forall i \in [0, 99].$$

Similarly all the bits of $R_3, R_4, \ldots, R_{99}$ can be calculated successively. As before, we begin by observing that the functions β_i for all values of $i \in [1, 99]$ are of the form

$$\beta_i(\cdot) = s_{i-1} + (s_{67} + r_{33}) \cdot \hat{\beta}_i(r_0, r_1, \ldots, r_{99}, s_i, s_{i+1}, \ldots, s_{99}),$$

where $s_{67} + r_{33}$ is the value of *CONTROL_BIT_S* and $\hat{\beta}_i$ is a function that depends on $r_0, r_1, \ldots, r_{99}$, $s_i, s_{i+1}, \ldots, s_{99}$ but not any of $s_0, s_1, \ldots, s_{i-1}$. Now consider the following equation governing s_{99}^{99}:

$$s_{99}^{99} = \beta_{99}(R_{98}, S_{98}) = s_{98}^{98} + CS_{98} \cdot \hat{\beta}_{99}(R_{98}, s_{99}^{98}).$$

In the above equation s_{98}^{98} is the only unknown and it appears as a linear term, and so its value can be calculated immediately. We therefore know the values of the 2 state bits of S_{98}: $s_{99}^{98},\ s_{98}^{98}$. Similarly look at the equations governing $s_{99}^{98},\ s_{98}^{98}$:

$$s_{99}^{98} = s_{98}^{97} + CS_{97} \cdot \hat{\beta}_{99}(R_{97}, s_{99}^{97}),$$

$$s_{98}^{98} = s_{97}^{97} + CS_{97} \cdot \hat{\beta}_{98}(R_{97}, s_{98}^{97}, s_{99}^{97}).$$

As before, s_{98}^{97} is the lone unknown term in the first equation whose value is determined immediately. After this s_{97}^{97} becomes the only unknown linear term in the next equation whose value too is determined easily. Thus we know 3 bits of S_{97}: $s_{97+i}^{97},\ i = 0, 1, 2$. Continuing in such a bottom up manner we can successively determine 4 bits of S_{96}, 5 bits of S_{95} and eventually all the 100 bits of S_0. The process is explained pictorially in Figure 2. □

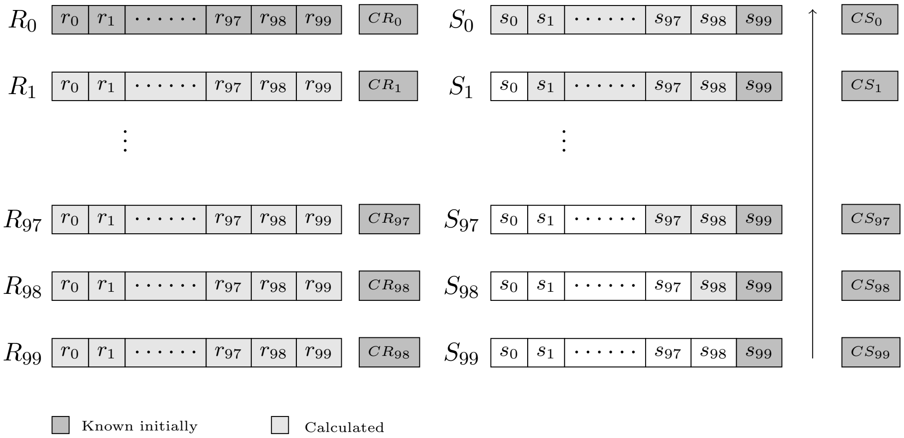

Fig. 2: Constructing the state S_0. Starting from PRGA round 99, any bit calculated at PRGA round i is used to determine state bits of round $i - 1$.

3.1 Faulting specific bits of R, S

Before getting into the details of the attack, we further note that the output key-stream bits $z_t, z_{t+1}, \ldots$ can also be expressed as polynomial functions over R_t, S_t. We have

$$\begin{aligned} z_t &= r_0^t + s_0^t = \theta_0(R_t, S_t), \\ z_{t+1} &= r_0^{t+1} + s_0^{t+1} = \rho_0(R_t, S_t) + \beta_0(R_t, S_t) = \theta_1(R_t, S_t), \\ z_{t+2} &= r_0^{t+2} + s_0^{t+2} = \rho_0(R_{t+1}, S_{t+1}) + \beta_0(R_{t+1}, S_{t+1}) = \theta_2(R_t, S_t) \end{aligned}$$

The exact forms of $\theta_0, \theta_1, \theta_2$ are given in Table 1.

In the rest of this section we will assume that the adversary is able to **(a)** re-key the device containing the cipher with the original key-IV, **(b)** apply faults to specific bit locations in the R, S registers and **(c)** exercise control over the timing of fault injection. Note that **(b)** is a stronger assumption, but we do not need it in our attack. We are using this assumption here to build a sub-routine. In the next subsection we shall demonstrate how the adversary can partially identify the location of any fault injected at a random position by comparing the faulty and fault-free key-streams.

We begin by observing the following differential properties of the functions $\theta_0, \theta_1, \theta_2$.

Table 1: The functions θ_i

i	$\theta_i(\cdot)$
0	$r_0 + s_0$
1	$r_0 \cdot r_{67} + r_0 \cdot s_{34} + r_{99} + s_{99}$
2	$r_0 \cdot r_{66} \cdot r_{67} + r_0 \cdot r_{66} \cdot s_{34} + r_0 \cdot r_{67} \cdot r_{99} + r_0 \cdot r_{67} \cdot s_{33} + r_0 \cdot r_{67} \cdot s_{34} \cdot s_{35} + r_0 \cdot r_{67} \cdot s_{34} + r_0 \cdot r_{67} + r_0 \cdot r_{99} \cdot s_{34} + r_0 \cdot s_{33} \cdot s_{34} + r_0 \cdot s_{34} \cdot s_{35} + r_{33} \cdot s_{99} + r_{66} \cdot r_{99} + r_{67} \cdot r_{99} \cdot s_{34} + r_{98} + r_{99} \cdot s_{33} + r_{99} \cdot s_{34} \cdot s_{35} + r_{99} \cdot s_{34} + r_{99} + s_{67} \cdot s_{99} + s_{98}$

(1) $\theta_1(\ldots, r_{67}, \ldots) + \theta_1(\ldots, 1 + r_{67}, \ldots) = r_0$
(2) $\theta_1(r_0, \ldots) + \theta_1(1 + r_0, \ldots) = s_{34} + r_{67}$
(3) $\theta_2(\ldots, s_{99}) + \theta_2(\ldots, 1 + s_{99}) = s_{67} + r_{33}$

These differential properties have the following immediate implications.

$$
\begin{aligned}
z_{t+1} + z_{t+1,\Delta r_{67}}(t) &= \theta_1(R_t, S_t) + \theta_1(R_{t,\Delta r_{67}}(t), S_{t,\Delta r_{67}}(t)) = r_0^t \ldots (1) \\
z_{t+1} + z_{t+1,\Delta r_0}(t) &= \theta_1(R_t, S_t) + \theta_1(R_{t,\Delta r_0}(t), S_{t,\Delta r_0}(t)) \\
&= s_{34}^t + r_{67}^t = CR_t \ \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots (2) \\
z_{t+2} + z_{t+2,\Delta s_{99}}(t) &= \theta_2(R_t, S_t) + \theta_2(R_{t,\Delta s_{99}}(t), S_{t,\Delta s_{99}}(t)) \\
&= s_{67}^t + r_{33}^t = CS_t \ \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots (3)
\end{aligned}
$$

The above equations hold for all the values of $t = 0, 1, 2, \ldots$. This implies that if the adversary is able to re-key the device with the original key-IV pair multiple times and apply faults at PRGA rounds $t = 0, 1, 2, 3, \ldots, 100$ at precisely[1] the R register locations $0, 67$ and the S register location 99, then by observing the difference between the fault-less and faulty key-stream bits, she would be able to recover the values of r_0^t, CR_t, CS_t for all values of $t = 0, 1, 2, \ldots, 100$. The fault at each register location must be preceded by re-keying.

Determining the other bits Hereafter, the values s_0^t for all $t = 0, 1, 2, 3, 4, \ldots, 100$ may be found by solving: $s_0^t = z_t + r_0^t$. Since $\beta_0(\cdot) = s_{99}$, this implies that $s_0^{t+1} = s_{99}^t,\ \forall t = 0, 1, 2, \ldots$ Therefore calculating the values of $s_0^t,\ \forall t \in [1, 100]$ is the same as calculating $s_{99}^t,\ \forall t \in [0, 99]$. The values of $r_{99}^t,\ \forall t \in [0, 99]$ may be obtained as follows. Consider the equation for z_{t+1}:

$$
\begin{aligned}
z_{t+1} = \ \theta_1(R_t, S_t) &= r_0^t \cdot r_{67}^t + r_0^t \cdot s_{34}^t + r_{99}^t + s_{99}^t \\
&= CR_t \cdot r_0^t + r_{99}^t + s_{99}^t,\ \forall t \in [0, 99].
\end{aligned}
$$

Note that r_{99}^t is the only unknown linear term in these equations and hence its value too can be determined immediately. At this point, we have the following state bits with us:

$$[r_0^t,\ r_{99}^t,\ CR_t,\ s_0^t,\ s_{99}^t,\ CS_t],\quad \forall t \in [0, 99].$$

Now by using the techniques outlined in Lemma 1 we can determine all the bits of the state R_0. Thereafter using Lemma 2, one can determine all the bits of S_0. Thus we have recovered the entire internal state at the beginning of the PRGA.

---

[1] We would like to point out that our actual attack does not need precise fault injection at all locations of R, S. This will be explained in the next sub-section.

3.2 How to identify the random locations where faults are injected

In this subsection we will show how the adversary can identify the locations of randomly applied faults to the registers R and S. Although it will not be possible to conclusively determine the location of faults applied to each and every location of R and the S registers, we will show that the adversary can, with some probability, identify faulty streams corresponding to locations $0, 67$ of R and 99 of S. The adversary will then use the techniques described in Subsection 3.1 to complete the attack.

To help with the process of fault location identification, we define the first and second Signature vectors for the location ϕ of R as

$$\Psi^1_{r_\phi}[i] = \begin{cases} 1, \text{ if } z_{t+i} = z_{t+i,\Delta r_\phi}(t) \text{ for all choices of } R_t, S_t, \\ 0, \text{ otherwise.} \end{cases}$$
$$\Psi^2_{r_\phi}[i] = \begin{cases} 1, \text{ if } z_{t+i} \neq z_{t+i,\Delta r_\phi}(t) \text{ for all choices of } R_t, S_t, \\ 0, \text{ otherwise.} \end{cases}$$

for $i = 0, 1, 2, \ldots, l-1$. Here $l \approx 40$ is a suitably chosen constant.

Remark 1. The value of l should be large enough so that one can differentiate 100 randomly generated bit sequences over GF(2) by comparing the first l bits of each sequence. By Birthday paradox, this requires the value of l to be atleast $2 \cdot \log_2 100 \approx 14$. We take $l = 40$ as computer simulations show that this value of l is sufficient to make a successful distinction with high probability.

Similarly one can define Signature vectors for any location ϕ the register S.

$$\Psi^1_{s_\phi}[i] = \begin{cases} 1, \text{ if } z_{t+i} = z_{t+i,\Delta s_\phi}(t) \text{ for all choices of } R_t, S_t, \\ 0, \text{ otherwise.} \end{cases}$$
$$\Psi^2_{s_\phi}[i] = \begin{cases} 1, \text{ if } z_{t+i} \neq z_{t+i,\Delta s_\phi}(t) \text{ for all choices of } R_t, S_t, \\ 0, \text{ otherwise.} \end{cases}$$

The task for the fault location identification routine is to determine the fault location ϕ of R (or S) by analyzing the difference between $z_t, z_{t+1}, \ldots$ and $z_{t,\Delta r_\phi}(t), z_{t+1,\Delta r_\phi}(t), \ldots$ (or $z_{t,\Delta s_\phi}(t), z_{t+1,\Delta s_\phi}(t), \ldots$) by using the Signature vectors $\Psi^1_{r_\phi}, \Psi^2_{r_\phi}$ (or $\Psi^1_{s_\phi}, \Psi^2_{s_\phi}$).

Note that the i^{th} bit of $\Psi^1_{r_\phi}$ is 1 if and only if the $(t+i)^{th}$ key-stream bits produced by R_t, S_t and $R_{t,\Delta r_\phi}(t), S_{t,\Delta r_\phi}(t)$ are the same for all choices of the internal state R_t, S_t and that i^{th} bit of $\Psi^2_{r_\phi}$ is 1 if the above key-stream bits are different for all choices of the internal state. Using this fact, one can devise the heuristic given in Algorithm 1 for the calculation of the Signature vectors.

Remark 2. Note that the value of N used in this algorithm should be large enough so that if the $(t+i)^{th}$ bits ($0 \leq i \leq 100$) generated by two randomly chosen states R_t, S_t and $R_{t,\Delta r_\phi}(t), S_{t,\Delta r_\phi}(t)$ are not equal for all R_t, S_t then for at least 1 of N randomly chosen R_t, S_t the $(t+i)^{th}$ keystream bits generated by them should be actually unequal. By computer simulations $N = 2^{20}$ has been found to be sufficient for this purpose.

One can compute the Signature vectors for all the fault locations in S in a similar manner. The complete list of Signature vectors for all the bit locations in R, S can be found in Appendix C. The concept of Signature vectors to deduce the location of a randomly applied fault was introduced in [8]. However the analysis of [8] can not be reproduced for MICKEY 2.0, since a lot of different register locations have the same Signature vector. However one can observe the following which are important to mount the attack.

```
Input: $N$: Any large integer $\approx 2^{20}$, $l$: A suitable vector length $\approx 40$;
Output: The Signature vectors $\Psi^1_{r_\phi}, \Psi^2_{r_\phi}\ \forall \phi \in [0, 99]$;

$\phi \leftarrow 0$;
while $\phi < 100$ do
    $t \leftarrow 0$;
    $Count[i] \leftarrow 0,\ \forall i \in [0, l-1]$;
    while $t < N$ do
        Choose $R \in_R \{0,1\}^{100},\ S \in_R \{0,1\}^{100}$ ;
        Set $R' \leftarrow R,\ S' \leftarrow S$ ;
        Set $R'(\phi) = 1 + R(\phi)$ /* Flip the $\phi^{th}$ bit of $R$ */;
        Set $[z_0, z_1, \ldots, z_{l-1}] = MICKEY(R, S)$;
        /* The first $l$ keystream bits generated by the state $R, S$ */
        Set $[\hat{z}_0, \hat{z}_1, \ldots, \hat{z}_{l-1}] = MICKEY(R', S')$;
        /* The first $l$ keystream bits generated by the state $R', S'$ */
        for $i = 0$ to $l - 1$ do
            if $z_i = \hat{z}_i$ then
                $Count[i] \leftarrow Count[i] + 1$;
            end
        end
        $t \leftarrow t + 1$;
    end
    for $i = 0$ to $l - 1$ do
        if $Count[i] = N$ then
            $\Psi^1_{r_\phi}[i] = 1\ \Psi^2_{r_\phi}[i] = 0$;
        end
        else if $Count[i] = 0$ then
            $\Psi^1_{r_\phi}[i] = 0\ \Psi^2_{r_\phi}[i] = 1$;
        end
        else
            $\Psi^1_{r_\phi}[i] = 0,\ \Psi^2_{r_\phi}[i] = 0$;
        end
    end
    $\phi \leftarrow \phi + 1$;
end
Return $\Psi^1_{r_\phi}, \Psi^2_{r_\phi}\ \forall \phi \in [0, 99]$;
```

Algorithm 1: Algorithm to calculate Signature vectors in MICKEY

Theorem 1. *The following statements hold for the Signature vectors*

$$\Psi^1_{r_\phi}, \Psi^2_{r_\phi}, \Psi^1_{s_\phi}, \Psi^2_{s_\phi}$$

of MICKEY 2.0.

A. $\Psi^1_{r_\phi}[0] = 1, \forall \phi \in [1, 99]$ *but* $\Psi^2_{r_0}[0] = 1$.
B. $\Psi^1_{r_\phi}[0] = \Psi^1_{r_\phi}[1] = 1, \forall \phi \in [1, 99] \setminus \{67, 99\}$.
C. $\Psi^2_{r_{99}}[1] = 1$, *and* $\Psi^2_{r_{67}}[1] = 0$.
D. $\Psi^1_{s_\phi}[0] = 1, \forall \phi \in [1, 99]$ *but* $\Psi^2_{s_0}[0] = 1$.
E. $\Psi^1_{s_\phi}[0] = \Psi^1_{s_\phi}[1] = 1, \forall \phi \in [1, 99] \setminus \{34, 99\}$.
F. $\Psi^2_{s_{99}}[1] = 1$, *and* $\Psi^2_{s_{34}}[1] = 0$.

Proof. Though some of the cases are similar, we present each of the cases for clarity. Proofs for the cases **A** to **F** are described separately below.

A. We have

$$\begin{aligned} z_t + z_{t,\Delta r_0}(t) &= \theta_0(R_t, S_t) + \theta_0(R_{t,\Delta r_0}(t), S_{t,\Delta r_0}(t)) \\ &= (r^t_0 + s^t_0) + (1 + r^t_0 + s^t_0) = 1,\ \forall R_t, S_t \in \{0,1\}^{100}. \end{aligned}$$

So, $\Psi^2_{r_0}[0] = 1$. Also θ_0 is not a function of any r_i, s_i for $i \in [1, 99]$ and so

$$\theta_0(R_{t,\Delta r_\phi}(t), S_{t,\Delta r_\phi}(t)) = \theta_0(R_t, S_t)$$

for all $\phi \in [1, 99]$ and so we have

$$\begin{aligned} z_t + z_{t,\Delta r_\phi}(t) &= \theta_0(R_t, S_t) + \theta_0(R_{t,\Delta r_\phi}(t), S_{t,\Delta r_\phi}(t)) \\ &= 0,\ \forall \phi \in [1, 99],\ \forall R_t, S_t \in \{0, 1\}^{100}. \end{aligned}$$

So, $\Psi^1_{r_\phi}[0] = 1$ for all $\phi \in [1, 99]$.

B. Since θ_1 is a function of $r_0, r_{67}, s_{34}, r_{99}, s_{99}$ only, for any $\phi \in [1, 99] \setminus \{67, 99\}$ we have

$$\theta_1(R_{t,\Delta r_\phi}(t), S_{t,\Delta r_\phi}(t)) = \theta_1(R_t, S_t).$$

Therefore

$$\begin{aligned} z_{t+1} + z_{t+1,\Delta r_\phi}(t) &= \theta_1(R_t, S_t) + \theta_1(R_{t,\Delta r_\phi}(t), S_{t,\Delta r_\phi}(t)) \\ &= 0,\ \forall \phi \in [1, 99] \setminus \{67, 99\},\ \forall R_t, S_t \in \{0, 1\}^{100}. \end{aligned}$$

So, $\Psi^1_{r_\phi}[1] = 1$ for all $\phi \in [1, 99] \setminus \{67, 99\}$.

C. We have

$$\begin{aligned} z_{t+1} + z_{t+1,\Delta r_{99}}(t) &= \theta_1(R_t, S_t) + \theta_1(R_{t,\Delta r_{99}}(t), S_{t,\Delta r_{99}}(t)) \\ &= (r_0^t \cdot r_{67}^t + r_0^t \cdot s_{34}^t + r_{99}^t + s_{99}^t) + (r_0^t \cdot r_{67}^t + r_0^t \cdot s_{34}^t + 1 + r_{99}^t + s_{99}^t) \\ &= 1,\ \forall R_t, S_t \in \{0, 1\}^{100}. \end{aligned}$$

So, $\Psi^2_{r_{99}}[1] = 1$. Also

$$\begin{aligned} z_{t+1} + z_{t+1,\Delta r_{67}}(t) &= \theta_1(R_t, S_t) + \theta_1(R_{t,\Delta r_{67}}(t), S_{t,\Delta r_{67}}(t)) \\ &= (r_0^t \cdot r_{67}^t + r_0^t \cdot s_{34}^t + r_{99}^t + s_{99}^t) + (r_0^t \cdot (1 + r_{67}^t) + r_0^t \cdot s_{34}^t + r_{99}^t + s_{99}^t) \\ &= r_0^t \neq 0 \text{ or } 1,\ \forall R_t, S_t \in \{0, 1\}^{100}. \end{aligned}$$

So, $\Psi^2_{r_{67}}[1] = 0$.

D. We have

$$\begin{aligned} z_t + z_{t,\Delta s_0}(t) &= \theta_0(R_t, S_t) + \theta_0(R_{t,\Delta s_0}(t), S_{t,\Delta s_0}(t)) \\ &= (r_0^t + s_0^t) + (r_0^t + 1 + s_0^t) = 1,\ \forall R_t, S_t \in \{0, 1\}^{100}. \end{aligned}$$

So, $\Psi^2_{s_0}[0] = 1$. Also θ_0 is not a function of any r_i, s_i for $i \in [1, 99]$ and so

$$\theta_0(R_{t,\Delta s_\phi}(t), S_{t,\Delta s_\phi}(t)) = \theta_0(R_t, S_t)$$

for all $\phi \in [1, 99]$ and so we have

$$\begin{aligned} z_t + z_{t,\Delta s_\phi}(t) &= \theta_0(R_t, S_t) + \theta_0(R_{t,\Delta s_\phi}(t), S_{t,\Delta s_\phi}(t)) \\ &= 0,\ \forall \phi \in [1, 99],\ \forall R_t, S_t \in \{0, 1\}^{100}. \end{aligned}$$

So, $\Psi^1_{s_\phi}[0] = 1$ for all $\phi \in [1, 99]$.

E. Since θ_1 is a function of $r_0, r_{67}, s_{34}, r_{99}, s_{99}$ only, for any $\phi \in [1, 99] \setminus \{34, 99\}$ we have

$$\theta_1(R_{t,\Delta s_\phi}(t), S_{t,\Delta s_\phi}(t)) = \theta_1(R_t, S_t).$$

Therefore

$$\begin{aligned} z_{t+1} + z_{t+1,\Delta s_\phi}(t) &= \theta_1(R_t, S_t) + \theta_1(R_{t,\Delta s_\phi}(t), S_{t,\Delta s_\phi}(t)) \\ &= 0,\ \forall \phi \in [1, 99] \setminus \{34, 99\},\ \forall R_t, S_t \in \{0,1\}^{100}. \end{aligned}$$

So, $\Psi^1_{s_\phi}[1] = 1$ for all $\phi \in [1, 99] \setminus \{34, 99\}$.

F. We have

$$\begin{aligned} z_{t+1} + z_{t+1,\Delta s_{99}}(t) &= \theta_1(R_t, S_t) + \theta_1(R_{t,\Delta s_{99}}(t), S_{t,\Delta s_{99}}(t)) \\ &= (r^t_0 \cdot r^t_{67} + r^t_0 \cdot s^t_{34} + r^t_{99} + s^t_{99}) + (r^t_0 \cdot r^t_{67} + r^t_0 \cdot s^t_{34} + r^t_{99} + 1 + s^t_{99}) \\ &= 1,\ \forall R_t, S_t \in \{0,1\}^{100}. \end{aligned}$$

So, $\Psi^2_{s_{99}}[1] = 1$. Also

$$\begin{aligned} z_{t+1} + z_{t+1,\Delta s_{34}}(t) &= \theta_1(R_t, S_t) + \theta_1(R_{t,\Delta s_{34}}(t), S_{t,\Delta s_{34}}(t)) \\ &= (r^t_0 \cdot r^t_{67} + r^t_0 \cdot s^t_{34} + r^t_{99} + s^t_{99}) + (r^t_0 \cdot r^t_{67} + r^t_0 \cdot (1 + s^t_{34}) + r^t_{99} + s^t_{99}) \\ &= r^t_0 \neq 0 \text{ or } 1,\ \forall R_t, S_t \in \{0,1\}^{100}. \end{aligned}$$

So, $\Psi^2_{s_{34}}[1] = 0$.

Thus the proof. □

Now, consider the attack scenario in which the adversary is able to re-key the device with the same key-IV multiple number of times and inject a single fault at a random location of register R at the beginning of any particular PRGA round $t \in [0, 100]$ and obtain faulty key-streams. She continues the process until she obtains 100 different faulty key-streams corresponding to 100 different fault locations in R and for each $t \in [0, 100]$ (as mentioned earlier this is done by comparing the first l bits of each faulty key-stream sequence). Assuming that every location has equal probability of getting injected by fault, the above process on an average takes around $100 \cdot \sum_{i=1}^{100} \frac{1}{i} \approx 2^{9.02}$ faults [2] and hence re-keyings for each value of $t \in [0, 100]$ and hence a total of $101 \cdot 2^{9.02} \approx 2^{15.68}$ faults. The process has to be repeated for the S register, and so the expected number of faults is $2 \cdot 2^{15.68} = 2^{16.68}$.

Mathematically speaking, if we define

$$Z_t = [z_t, z_{t+1}, \ldots, z_{t+l-1}], \text{ and } \Delta_{r_\phi} Z_t = [z_{t,\Delta r_\phi}(t), z_{t+1,\Delta r_\phi}(t), \ldots, z_{t+l-1,\Delta r_\phi}(t)],$$

then the adversary at this point has knowledge of the 100 differential key-streams $\eta_{t,r_\phi} = Z_t + \Delta_{r_\phi} Z_t$ for each value of $t \in [0, 100]$. The adversary however does not know the exact fault location corresponding to any differential stream i.e. she has been unable to assign fault location labels to any of the differential streams. With this information in hand we shall study the implications of the observations **A** to **F**.

Implication of A: For any $t \in [0, 100]$, $\Psi^2_{r_0}[0] = 1$ guarantees that there is at least one differential stream with $\eta_{t,r_\phi}[0] = 1$ whereas $\Psi^1_{r_\phi}[0] = 1, \forall \phi \in [1, 99]$ guarantees that that there is exactly one differential stream with this property. This implies that out of the 100 differential streams for any PRGA round t the one and only differential stream with this property must have been produced due to a fault on the 0^{th} location in R. Note that labelling of this stream helps us determine the values of CR_t for all $t \in [0, 100]$ from Eqn. (2).

Implication of B, C: Once the differential stream corresponding to the 0^{th} location has been labelled we now turn our attention to the remaining 99 streams. Statement **B** guarantees that of the remaining 99 streams at least 97 have the property

$$\text{(P1)}\ \eta_{t,r_\phi}[0] = \eta_{t,r_\phi}[1] = 0.$$

Statement **C** guarantees that the number of streams with the property

$$\text{(P2)}\ \eta_{t,r_\phi}[0] = 0, \eta_{t,r_\phi}[1] = 1.$$

is at most 2 and at least 1. If the number of streams that satisfy (P1) is 98 then the lone stream satisfying (P2) must have been produced due to fault on location 99 of R. This immediately implies that $\eta_{t,r_{67}}[1] = 0$ which by Eqn. (1) in turn implies that $r_0^t = 0$. Else if the number of streams satisfying (P2) is 2 then it implies that these streams were produced due to faults in location $67, 99$ of R. This implies $\eta_{t,r_{67}}[1] = r_0^t = 1$.

Repeating the entire process on Register S one can similarly obtain the vectors $\varDelta_{s_\phi} Z_t$ and the differential streams $\eta_{t,s_\phi} = Z_t + \varDelta_{s_\phi} Z_t$ for all values of $t \in [0, 100]$. As before the streams η_{t,s_ϕ} are unlabeled. Let us now study the implications of **D, E, F**.

Implication of D: For any $t \in [0, 100]$, $\varPsi^2_{s_0}[0] = 1$ guarantees that there is at least one differential stream with $\eta_{t,s_\phi}[0] = 1$ whereas $\varPsi^1_{s_\phi}[0] = 1, \forall \phi \in [1, 99]$ guarantees that that there is exactly one differential stream with this property. This implies that out of the 100 differential streams for any PRGA round t the one and only differential stream with this property must have been produced due to a fault on the 0^{th} location in S.

Implication of E, F: Once the differential stream corresponding to the 0^{th} location has been labelled we now turn our attention to the remaining 99 streams. The statement **E** guarantees that of the remaining 99 streams at least 97 have the property

$$\text{(P3)}\ \eta_{t,s_\phi}[0] = \eta_{t,s_\phi}[1] = 0.$$

Statement **F** guarantees that the number of streams with the property

$$\text{(P4)}\ \eta_{t,s_\phi}[0] = 0, \eta_{t,s_\phi}[1] = 1,$$

is at most 2 and at least 1.

Case 1 If the number of streams that satisfy (P3) is 98 then the lone stream satisfying (P4) must have been produced due to fault on location 99 of S. Once the stream corresponding to location 99 of S has been labelled, we can use Eqn (3) to determine $CS_t = \eta_{t,s_{99}}[2]$.

Case 2 If the number of streams satisfying (P4) is 2 then it implies that these streams were produced due to faults in location $34, 99$ of S.

- **(i)** Now if the bit indexed 2 of both these vectors are equal then we can safely assume $CS_t = \eta_{t,s_{99}}[2] = \eta_{t,s_{34}}[2]$.

(ii) A confusion occurs when $\eta_{t,s_{99}}[2] \neq \eta_{t,s_{34}}[2]$. In such a situation we would be unable to conclusively able to determine the value of CS_t.

Assuming independence, we assume that **Cases 1, 2** have equal probability of occurring. Given the occurrence of **Case 2**, we can also assume that **2(i), 2(ii)** occurs with equal probability. Therefore the probability of confusion, i.e., the probability that we are unable to determine the value of CS_t for any t is approximately equal to $\frac{1}{2} \cdot \frac{1}{2} = \frac{1}{4}$. Let γ denote the number of $t \in [0, 100]$ such that CS_t can not be conclusively determined then γ is distributed according to $\gamma \sim Binomial(101, \frac{1}{4})$. Therefore the expected value of γ is $E(\gamma) = 101 \cdot \frac{1}{4} = 25.25$. Also the probability that

$$P(\gamma > 35) = \sum_{k=36}^{101} \binom{101}{k} \left(\frac{1}{4}\right)^k \left(\frac{3}{4}\right)^{101-k} \approx 0.01.$$

In such a situation the adversary must guess the γ values of CS_t to perform the attack, which implies that the adversary must perform the calculations in Section 3.1 and Lemma 1, Lemma 2 a total of 2^γ times to complete the attack. For the correct value of the guesses, the calculated state R_0, S_0 will produce the given fault-free key-stream sequence.

We present a complete description of the attack in Algorithm 2.

```
Generate and record the fault-free keystream $z_0, z_1, z_2, \dots$ for some key-IV $K, IV$
$t \leftarrow 0$;
while $t \leq 100$ do
    while 100 different faulty key-stream sequences $\Delta_{r_\phi} Z_t$ have not been obtained do
        Re-key the cipher with key-IV $K, IV$;
        Inject a fault at a random unknown location $\phi \in [0, 99]$ in $R$ at PRGA round $t$;
        Record the faulty key-stream sequence $\Delta_{r_\phi} Z_t$;
    end
    $t \leftarrow t + 1$;
end
Calculate $r_0^t, CR_t, \ \forall t \in [0, 100]$ using A, B, C;
$t \leftarrow 0$;
while $t \leq 100$ do
    while 100 different faulty key-stream sequences $\Delta_{s_\phi} Z_t$ have not been obtained do
        Re-key the cipher with key-IV $K, IV$;
        Inject a fault at a random unknown location $\phi \in [0, 99]$ in $S$ at PRGA round $t$;
        Record the faulty key-stream sequence $\Delta_{s_\phi} Z_t$;
    end
    $t \leftarrow t + 1$;
end
Using D, E, F calculate $CS_t$,  for all such $t \in [0, 100]$ for which there is no confusion;
Let the number of undecided bits $CS_t = \gamma$;
for Each of the $2^\gamma$ guesses of the undecided $CS_t$'s do
    Use techniques of Subsection 3.1 compute $r_0^t, \ r_{99}^t, \ CR_t, \ s_0^t, \ s_{99}^t, \ CS_t, \ \ \forall t \in [0, 99]$;
    Use Lemma 1, Lemma 2 try to compute $R_0, S_0$. ;
    if $R_0, S_0$ produce the sequence $z_0, z_1, z_2, \dots$ then
        Output the required state $R_0, S_0$;
    end
end
```

Algorithm 2: Fault Attack against MICKEY 2.0

3.3 Issues related to the length of the IV

It is known that MICKEY 2.0 employs a variable length IV of length at most 80. So if v is the length of the IV then the cipher will run for $v + 80$ (Key loading) $+ 100$ (Preclock) clock intervals before entering the PRGA phase. Our attack requires that the first faults are to be injected at the beginning of the PRGA. In order to do that the adversary must know the value of v. This not a strong assumption as IVs are assumed to be known. However even if the adversary does not know the IV or its length the attack can be performed. Since $0 \leq v \leq 80$ must be satisfied, the strategy of the adversary who does not know the value of v will be as follows. She will inject the first set of faults at clock round 260 which corresponds to the PRGA round $p = 260 - 180 - v = 80 - v$. After performing the attack, the adversary will end up constructing the internal state R_p, S_p instead of R_0, S_0. Finding the value of p by looking at the faultless key-stream sequence is straightforward.

3.4 Complexity of the Attack

As mentioned in Section 3.2, the attack requires the adversary to obtain 100 different faulty key-streams corresponding to all fault locations in R for PRGA rounds $t \in [0, 100]$. This requires $101 \cdot 100 \cdot \sum_{i=1}^{100} \frac{1}{k} \approx 2^{15.68}$ faults on an average. The same process must be repeated for the register S and hence the expected number of total faults is $2^{16.68}$. The computational overload comes from guessing the γ values of CS_t which can not be found out by observing the differential key-streams. This requires a computational effort proportional to 2^{γ}. Since γ is distributed according to $Binomial(101, \frac{1}{4})$, the expected value of γ is 25.25. The expected value of the computation complexity is therefore given by $E(2^{\gamma}) = \sum_{k=0}^{101} \binom{101}{k} \left(\frac{1}{4}\right)^k \left(\frac{3}{4}\right)^{101-k} 2^k \approx 2^{32.5}$.

4 Conclusion

A differential fault attack against the stream cipher MICKEY 2.0 is presented. The work is one of the first cryptanalytic attempts against this cipher and requires reasonable computational effort. The attack is somewhat made possible due to the simplicity of the output function and certain register update operations of MICKEY 2.0 and would have been thwarted had these been of a more complex nature. It would be interesting to study efficient counter-measures with minimum tweak in the design.

Given our work in this paper, differential fault attacks are now known against all of the three ciphers in the hardware portfolio of eStream. The attacks on all the 3 ciphers use exactly the same fault model that is similar to what described in this paper. Table 2 summarizes the fault requirements.

Table 2: Summary of fault attacks against eStream's hardware candidates

Cipher	State size	Average # of Faults
Trivium [14]	288	3.2
Grain v1 [9]	160	$\approx 2^{8.5}$
MICKEY 2.0	200	$\approx 2^{16.7}$

To the best of our knowledge, there was no published fault attack on MICKEY 2.0. prior to our work. We believe that one of the reasons this remained open for such a long time could be that the cipher uses irregular clocking to update its state registers. Hence it becomes difficult to determine the location of a randomly applied fault injected in either the R or S register by simply comparing the faulty and fault-free key-streams. The idea explained in Theorem 1 and its implications are instrumental in mounting the attack. The total number of faults is indeed much higher when we compare it with the other two eStream hardware candidates. However, this seems natural as MICKEY 2.0 has more complex structure than Trivium or Grain v1.

References

1. The ECRYPT Stream Cipher Project. eSTREAM Portfolio of Stream Ciphers. Revised on September 8, 2008.
2. Coupon Collector's Problem. Available at `http://en.wikipedia.org/wiki/Coupon_collector's_problem`.
3. C. Cid and M. Robshaw (Editors), S. Babbage, J. Borghoff and V. Velichkov (Contributors). The eSTREAM Portfolio in 2012, 16 January 2012, Version 1.0. Available at `http://www.ecrypt.eu.org/documents/D.SYM.10-v1.pdf`.
4. S. Babbage and M. Dodd. The stream cipher MICKEY 2.0. ECRYPT Stream Cipher Project Report. Available at `http://www.ecrypt.eu.org/stream/p3ciphers/mickey/mickey_p3.pdf`.
5. S. Babbage and M. Dodd. The stream cipher MICKEY-128 2.0. ECRYPT Stream Cipher Project Report. Available at `http://www.ecrypt.eu.org/stream/p3ciphers/mickey/mickey128_p3.pdf`.
6. E. Biham and A. Shamir. Differential Fault Analysis of Secret Key Cryptosystems. In CRYPTO 1997, LNCS Vol. 1294, pp. 513–525.
7. D. Boneh, R. A. DeMillo and R. J. Lipton. On the Importance of Checking Cryptographic Protocols for Faults. In EUROCRYPT 1997, LNCS Vol. 1233, pp. 37–51.
8. S. Banik, S. Maitra and S. Sarkar. A Differential Fault Attack on the Grain Family of Stream Ciphers. In CHES 2012, LNCS Vol. 7428, pp. 122–139, 2012.
9. S. Banik, S. Maitra and S. Sarkar. A Differential Fault Attack on Grain Family under Reasonable Assumptions. In INDOCRYPT 2012, to be published in LNCS.
10. A. Berzati, C. Canovas, G. Castagnos, B. Debraize, L. Goubin, A. Gouget, P. Paillier and S. Salgado. Fault Analysis of Grain-128. In IEEE International Workshop on Hardware-Oriented Security and Trust, pp. 7–14, 2009.
11. B. Gierlichs, L. Batina, C. Clavier, T. Eisenbarth, A. Gouget, H. Handschuh, T. Kasper, K. Lemke-Rust, S. Mangard, A. Moradi, and E. Oswald. Susceptibility of eSTREAM Candidates towards Side Channel Analysis. In Proceedings of SASC 2008, available via `http://www.ecrypt.eu.org/stvl/sasc2008/`.
12. J. J. Hoch and A. Shamir. Fault Analysis of Stream Ciphers. In CHES 2004, LNCS, Vol. 3156, pp. 1–20, 2004.
13. M. Hojsík and B. Rudolf. Differential Fault Analysis of Trivium. In FSE 2008, LNCS, Vol. 5086, pp. 158–172, 2008.
14. M. Hojsík and B. Rudolf. Floating Fault Analysis of Trivium. In INDOCRYPT 2008, LNCS, Vol. 5365, pp. 239–250, 2008.
15. J. Hong and W. Kim. TMD-Tradeoff and State Entropy Loss Considerations of stream cipher MICKEY. In INDOCRYPT 2005, LNCS Vol. 3797, pp. 169–182, 2005.
16. S. Karmakar and D. Roy Chowdhury. Fault analysis of Grain-128 by targeting NFSR. In AFRICACRYPT 2011, LNCS, Vol. 6737, pp. 298–315, 2011.
17. S. P. Skorobogatov. Optically Enhanced Position-Locked Power Analysis. In CHES 2006, LNCS, Vol. 4249, pp. 61–75, 2006.
18. S. P. Skorobogatov and R. J. Anderson. Optical Fault Induction Attacks. In CHES 2002, LNCS, Vol. 2523, pp. 2–12, 2003.

Appendix A: Brief Description of MICKEY 2.0

MICKEY 2.0 uses an 80-bit key and a variable length IV, the length of which may be between 0 and 80 bits. The physical structure of the cipher consists of two 100 bit registers R and S that exercise mutual control over each other's evolution. Let $r_0, r_1, r_2, \ldots, r_{99}$ denote the contents of the register R and $s_0, s_1, s_2, \ldots, s_{99}$ denote the contents of the register S. In order to describe the structure of the cipher and its working let us first define the following routines. Note that the description given here is based on [4].

Clocking register R Let $r_0, r_1, \ldots, r_{99}$ be the state of the register R before clocking, and let $r'_0, r'_1, \ldots, r'_{99}$ be the state of the register R after clocking. Define the integer array $RTAPS$ as

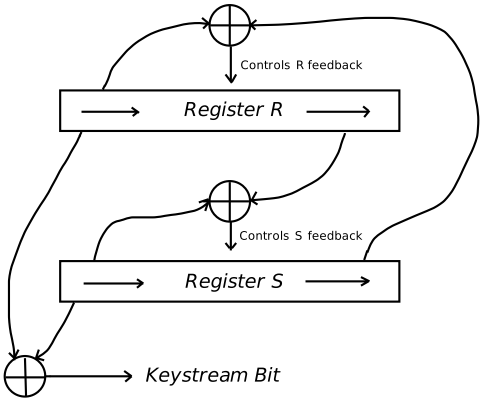

Fig. 3: The variable clocking architecture of MICKEY

follows

$$\begin{aligned} RTAPS = \{\ & 0,1,3,4,5,6,9,12,13,16,19,20,21,22,25,28,37,38,41,42, \\ & 45,46,50,52,54,56,58,60,61,63,64,65,66,67,71,72,79,80, \\ & 81,82,87,88,89,90,91,92,94,95,96,97\} \end{aligned}$$

Now define an operation

$$CLOCK\_R(R, INPUT\_BIT\_R, CONTROL\_BIT\_R)$$

```
1. Define $FEEDBACK\_BIT = r_{99} + INPUT\_BIT\_R$
2. For $1 \leq i \leq 99 : r'_i = r_{i-1}$. $r'_0 = 0$.
3. For $0 \leq i \leq 99$ : if $i \in RTAPS$, $r'_i = r'_i + FEEDBACK\_BIT$.
4. If $CONTROL\_BIT\_R = 1$:
     For $0 \leq i \leq 99 : r'_i = r'_i + r_i$
```

Clocking register *S* Let $s_0, s_1, \ldots, s_{99}$ be the state of the register S before clocking, and let $s'_0, s'_1, \ldots, s'_{99}$ be the state of the register S after clocking. Let $\hat{s}_0, \hat{s}_1, \ldots,\ \hat{s}_{99}$ be intermediate variables. Define the four sequences $COMP0_i,\ 1 \leq i \leq 98$; $COMP1_i,\ 1 \leq i \leq 98$; $FB0_i,\ 0 \leq i \leq 99$ and $FB1_i,\ 0 \leq i \leq 99$ over GF(2) as in Table 3: Now define an operation

$$CLOCK\_S(S, INPUT\_BIT\_S, CONTROL\_BIT\_S)$$

```
1. Define $FEEDBACK\_BIT = s_{99} + INPUT\_BIT\_S$
2. For $1 \leq i \leq 98 : \hat{s}_i = s_{i-1} + \big((s_i + COMP0_i) \cdot (s_{i+1} + COMP1_i)\big)$. $\hat{s}_0 = 0,\ \hat{s}_{99} = s_{98}$.
3. If $CONTROL\_BIT\_S = 0$:
     For $0 \leq i \leq 99$: $s'_i = \hat{s}_i + (FB0_i \cdot FEEDBACK\_BIT)$
   Else If $CONTROL\_BIT\_S = 1$:
     For $0 \leq i \leq 99$: $s'_i = \hat{s}_i + (FB1_i \cdot FEEDBACK\_BIT)$
```

The *CLOCK_KG* routine We define another operation

$$CLOCK\_KG(R, S, MIXING, INPUT\_BIT)$$

Table 3: The sequences $COMP0, COMP1, FB0, FB1$

i	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24
$COMP0_i$		0	0	0	1	1	0	0	0	1	0	1	1	1	1	0	1	0	0	1	0	1	0	1	0
$COMP1_i$		1	0	1	1	0	0	1	0	1	1	1	1	0	0	1	0	1	0	0	0	1	1	0	1
$FB0_i$	1	1	1	1	0	1	0	1	1	1	1	1	1	1	1	0	0	1	0	1	1	1	1	1	1
$FB1_i$	1	1	1	0	1	1	1	0	0	0	0	1	1	1	0	1	0	0	1	1	0	0	0	1	0
i	25	26	27	28	29	30	31	32	33	34	35	36	37	38	39	40	41	42	43	44	45	46	47	48	49
$COMP0_i$	1	0	1	0	1	1	0	1	0	0	1	0	0	0	0	0	0	0	1	0	1	0	1	0	1
$COMP1_i$	0	1	1	1	0	1	1	1	1	0	0	0	1	1	0	1	0	1	1	1	0	0	0	0	1
$FB0_i$	1	1	1	1	0	0	1	1	0	0	0	0	0	0	1	1	1	0	0	1	0	0	1	0	1
$FB1_i$	0	1	1	0	0	1	0	1	1	0	0	0	1	1	0	0	0	0	0	1	1	0	1	1	0
i	50	51	52	53	54	55	56	57	58	59	60	61	62	63	64	65	66	67	68	69	70	71	72	73	74
$COMP0_i$	0	0	0	0	1	0	1	0	0	1	1	1	1	0	0	1	0	1	0	1	1	1	1	1	1
$COMP1_i$	0	0	0	1	0	1	1	1	0	0	0	1	1	1	1	1	1	0	1	0	1	1	1	0	1
$FB0_i$	0	1	0	0	1	0	1	1	1	1	0	1	0	1	0	1	0	0	0	0	0	0	0	0	0
$FB1_i$	0	0	1	0	0	0	1	0	0	1	0	0	1	0	1	1	0	1	0	1	0	0	1	0	1
i	75	76	77	78	79	80	81	82	83	84	85	86	87	88	89	90	91	92	93	94	95	96	97	98	99
$COMP0_i$	1	1	1	0	1	0	1	1	1	1	1	1	0	1	0	1	0	0	0	0	0	0	1	1	
$COMP1_i$	1	1	1	0	0	0	1	0	0	0	0	1	1	1	0	0	0	1	0	0	1	1	0	0	
$FB0_i$	1	1	0	1	0	0	0	1	1	0	1	1	1	0	0	1	1	1	0	0	1	1	0	0	0
$FB1_i$	0	0	0	1	1	1	1	0	1	1	1	1	1	0	0	0	0	0	0	1	0	0	0	0	1

```
1. $CONTROL\_BIT\_R = s_{34} + r_{67}$, $CONTROL\_BIT\_S = s_{67} + r_{33}$
2. If $MIXING = 1$ :
     $INPUT\_BIT\_R = INPUT\_BIT + s_{50}$
   Else If $MIXING = 0$ :
     $INPUT\_BIT\_R = INPUT\_BIT$
3. $INPUT\_BIT\_S = INPUT\_BIT$
4. $CLOCK\_R(R, INPUT\_BIT\_R, CONTROL\_BIT\_R)$
5. $CLOCK\_S(S, INPUT\_BIT\_S, CONTROL\_BIT\_S)$
```

Working of the Cipher We will now describe the algorithm governing the functioning of the cipher. Let $K = k_0, k_1, \ldots, k_{79}$ be the 80 bit key used by the cipher. Let $IV = iv_0, iv_1, \ldots, iv_{v-1}$ be the v-bit IV ($0 \leq v \leq 80$). Then the cipher operates in the 4 stages as described below.

STAGE 1. IV loading

```
Initialize both $R$ and $S$ to the all-zero state.
For $0 \leq i \leq v - 1$ : $CLOCK\_KG(R, S, 1, iv_i)$
```

STAGE 2. Key loading

```
For $0 \leq i \leq 79$ : $CLOCK\_KG(R, S, 1, k_i)$
```

STAGE 3. Preclock Stage

```
For $0 \leq i \leq 99$ : $CLOCK\_KG(R, S, 1, 0)$
```

STAGE 4. PRGA(Pseudo-Random stream generation algorithm)

```
$i \leftarrow 0$
While key-stream is required
    $z_i = r_0 + s_0$
    $CLOCK\_KG(R, S, 0, 0)$
    $i \leftarrow i + 1$
```

Appendix B: The functions $\rho_i, \beta_i \ \forall i \in [0, 99]$

i	ρ_i	β_i
0	$r_0 \cdot r_{67} + r_0 \cdot s_{34} + r_{99}$	s_{99}
1	$r_0 + r_1 \cdot r_{67} + r_1 \cdot s_{34} + r_{99}$	$s_0 + s_1 \cdot s_2 + s_1 + s_{99}$
2	$r_1 + r_2 \cdot r_{67} + r_2 \cdot s_{34}$	$s_1 + s_2 \cdot s_3 + s_{99}$
3	$r_2 + r_3 \cdot r_{67} + r_3 \cdot s_{34} + r_{99}$	$r_{33} \cdot s_{99} + s_2 + s_3 \cdot s_4 + s_3 + s_{67} \cdot s_{99} + s_{99}$
4	$r_3 + r_4 \cdot r_{67} + r_4 \cdot s_{34} + r_{99}$	$r_{33} \cdot s_{99} + s_3 + s_4 \cdot s_5 + s_4 + s_5 + s_{67} \cdot s_{99} + 1$
5	$r_4 + r_5 \cdot r_{67} + r_5 \cdot s_{34} + r_{99}$	$s_4 + s_5 \cdot s_6 + s_6 + s_{99}$
6	$r_5 + r_6 \cdot r_{67} + r_6 \cdot s_{34} + r_{99}$	$r_{33} \cdot s_{99} + s_5 + s_6 \cdot s_7 + s_{67} \cdot s_{99}$
7	$r_6 + r_7 \cdot r_{67} + r_7 \cdot s_{34}$	$r_{33} \cdot s_{99} + s_6 + s_7 \cdot s_8 + s_7 + s_{67} \cdot s_{99} + s_{99}$
8	$r_7 + r_8 \cdot r_{67} + r_8 \cdot s_{34}$	$r_{33} \cdot s_{99} + s_7 + s_8 \cdot s_9 + s_{67} \cdot s_{99} + s_{99}$
9	$r_8 + r_9 \cdot r_{67} + r_9 \cdot s_{34} + r_{99}$	$r_{33} \cdot s_{99} + s_8 + s_9 \cdot s_{10} + s_9 + s_{10} + s_{67} \cdot s_{99} + s_{99} + 1$
10	$r_9 + r_{10} \cdot r_{67} + r_{10} \cdot s_{34}$	$r_{33} \cdot s_{99} + s_9 + s_{10} \cdot s_{11} + s_{10} + s_{67} \cdot s_{99} + s_{99}$
11	$r_{10} + r_{11} \cdot r_{67} + r_{11} \cdot s_{34}$	$s_{10} + s_{11} \cdot s_{12} + s_{11} + s_{12} + s_{99} + 1$
12	$r_{11} + r_{12} \cdot r_{67} + r_{12} \cdot s_{34} + r_{99}$	$s_{11} + s_{12} \cdot s_{13} + s_{12} + s_{13} + s_{99} + 1$
13	$r_{12} + r_{13} \cdot r_{67} + r_{13} \cdot s_{34} + r_{99}$	$s_{12} + s_{13} \cdot s_{14} + s_{14} + s_{99}$
14	$r_{13} + r_{14} \cdot r_{67} + r_{14} \cdot s_{34}$	$r_{33} \cdot s_{99} + s_{13} + s_{14} \cdot s_{15} + s_{15} + s_{67} \cdot s_{99} + s_{99}$
15	$r_{14} + r_{15} \cdot r_{67} + r_{15} \cdot s_{34}$	$r_{33} \cdot s_{99} + s_{14} + s_{15} \cdot s_{16} + s_{15} + s_{67} \cdot s_{99}$
16	$r_{15} + r_{16} \cdot r_{67} + r_{16} \cdot s_{34} + r_{99}$	$s_{15} + s_{16} \cdot s_{17} + s_{17}$
17	$r_{16} + r_{17} \cdot r_{67} + r_{17} \cdot s_{34}$	$r_{33} \cdot s_{99} + s_{16} + s_{17} \cdot s_{18} + s_{17} + s_{67} \cdot s_{99} + s_{99}$
18	$r_{17} + r_{18} \cdot r_{67} + r_{18} \cdot s_{34}$	$r_{33} \cdot s_{99} + s_{17} + s_{18} \cdot s_{19} + s_{67} \cdot s_{99}$
19	$r_{18} + r_{19} \cdot r_{67} + r_{19} \cdot s_{34} + r_{99}$	$s_{18} + s_{19} \cdot s_{20} + s_{20} + s_{99}$
20	$r_{19} + r_{20} \cdot r_{67} + r_{20} \cdot s_{34} + r_{99}$	$r_{33} \cdot s_{99} + s_{19} + s_{20} \cdot s_{21} + s_{67} \cdot s_{99} + s_{99}$
21	$r_{20} + r_{21} \cdot r_{67} + r_{21} \cdot s_{34} + r_{99}$	$r_{33} \cdot s_{99} + s_{20} + s_{21} \cdot s_{22} + s_{21} + s_{22} + s_{67} \cdot s_{99} + s_{99} + 1$
22	$r_{21} + r_{22} \cdot r_{67} + r_{22} \cdot s_{34} + r_{99}$	$r_{33} \cdot s_{99} + s_{21} + s_{22} \cdot s_{23} + s_{22} + s_{67} \cdot s_{99} + s_{99}$
23	$r_{22} + r_{23} \cdot r_{67} + r_{23} \cdot s_{34}$	$s_{22} + s_{23} \cdot s_{24} + s_{24} + s_{99}$
24	$r_{23} + r_{24} \cdot r_{67} + r_{24} \cdot s_{34}$	$r_{33} \cdot s_{99} + s_{23} + s_{24} \cdot s_{25} + s_{24} + s_{67} \cdot s_{99} + s_{99}$
25	$r_{24} + r_{25} \cdot r_{67} + r_{25} \cdot s_{34} + r_{99}$	$r_{33} \cdot s_{99} + s_{24} + s_{25} \cdot s_{26} + s_{26} + s_{67} \cdot s_{99} + s_{99}$
26	$r_{25} + r_{26} \cdot r_{67} + r_{26} \cdot s_{34}$	$s_{25} + s_{26} \cdot s_{27} + s_{26} + s_{99}$
27	$r_{26} + r_{27} \cdot r_{67} + r_{27} \cdot s_{34}$	$s_{26} + s_{27} \cdot s_{28} + s_{27} + s_{28} + s_{99} + 1$
28	$r_{27} + r_{28} \cdot r_{67} + r_{28} \cdot s_{34} + r_{99}$	$r_{33} \cdot s_{99} + s_{27} + s_{28} \cdot s_{29} + s_{28} + s_{67} \cdot s_{99} + s_{99}$
29	$r_{28} + r_{29} \cdot r_{67} + r_{29} \cdot s_{34}$	$s_{28} + s_{29} \cdot s_{30} + s_{30}$
30	$r_{29} + r_{30} \cdot r_{67} + r_{30} \cdot s_{34}$	$r_{33} \cdot s_{99} + s_{29} + s_{30} \cdot s_{31} + s_{30} + s_{31} + s_{67} \cdot s_{99} + 1$
31	$r_{30} + r_{31} \cdot r_{67} + r_{31} \cdot s_{34}$	$r_{33} \cdot s_{99} + s_{30} + s_{31} \cdot s_{32} + s_{31} + s_{67} \cdot s_{99} + s_{99}$
32	$r_{31} + r_{32} \cdot r_{67} + r_{32} \cdot s_{34}$	$s_{31} + s_{32} \cdot s_{33} + s_{32} + s_{33} + s_{99} + 1$
33	$r_{32} + r_{33} \cdot r_{67} + r_{33} \cdot s_{34}$	$r_{33} \cdot s_{99} + s_{32} + s_{33} \cdot s_{34} + s_{33} + s_{67} \cdot s_{99}$
34	$r_{33} + r_{34} \cdot r_{67} + r_{34} \cdot s_{34}$	$s_{33} + s_{34} \cdot s_{35}$
35	$r_{34} + r_{35} \cdot r_{67} + r_{35} \cdot s_{34}$	$s_{34} + s_{35} \cdot s_{36} + s_{36}$
36	$r_{35} + r_{36} \cdot r_{67} + r_{36} \cdot s_{34}$	$s_{35} + s_{36} \cdot s_{37}$
37	$r_{36} + r_{37} \cdot r_{67} + r_{37} \cdot s_{34} + r_{99}$	$r_{33} \cdot s_{99} + s_{36} + s_{37} \cdot s_{38} + s_{37} + s_{67} \cdot s_{99}$
38	$r_{37} + r_{38} \cdot r_{67} + r_{38} \cdot s_{34} + r_{99}$	$r_{33} \cdot s_{99} + s_{37} + s_{38} \cdot s_{39} + s_{38} + s_{67} \cdot s_{99}$
39	$r_{38} + r_{39} \cdot r_{67} + r_{39} \cdot s_{34}$	$r_{33} \cdot s_{99} + s_{38} + s_{39} \cdot s_{40} + s_{67} \cdot s_{99} + s_{99}$
40	$r_{39} + r_{40} \cdot r_{67} + r_{40} \cdot s_{34}$	$r_{33} \cdot s_{99} + s_{39} + s_{40} \cdot s_{41} + s_{40} + s_{67} \cdot s_{99} + s_{99}$
41	$r_{40} + r_{41} \cdot r_{67} + r_{41} \cdot s_{34} + r_{99}$	$r_{33} \cdot s_{99} + s_{40} + s_{41} \cdot s_{42} + s_{67} \cdot s_{99} + s_{99}$
42	$r_{41} + r_{42} \cdot r_{67} + r_{42} \cdot s_{34} + r_{99}$	$s_{41} + s_{42} \cdot s_{43} + s_{42}$
43	$r_{42} + r_{43} \cdot r_{67} + r_{43} \cdot s_{34}$	$s_{42} + s_{43} \cdot s_{44} + s_{43} + s_{44} + 1$
44	$r_{43} + r_{44} \cdot r_{67} + r_{44} \cdot s_{34}$	$s_{43} + s_{44} \cdot s_{45} + s_{44} + s_{99}$
45	$r_{44} + r_{45} \cdot r_{67} + r_{45} \cdot s_{34} + r_{99}$	$r_{33} \cdot s_{99} + s_{44} + s_{45} \cdot s_{46} + s_{46} + s_{67} \cdot s_{99}$
46	$r_{45} + r_{46} \cdot r_{67} + r_{46} \cdot s_{34} + r_{99}$	$s_{45} + s_{46} \cdot s_{47}$
47	$r_{46} + r_{47} \cdot r_{67} + r_{47} \cdot s_{34}$	$s_{46} + s_{47} \cdot s_{48} + s_{48} + s_{99}$
48	$r_{47} + r_{48} \cdot r_{67} + r_{48} \cdot s_{34}$	$r_{33} \cdot s_{99} + s_{47} + s_{48} \cdot s_{49} + s_{67} \cdot s_{99}$
49	$r_{48} + r_{49} \cdot r_{67} + r_{49} \cdot s_{34}$	$r_{33} \cdot s_{99} + s_{48} + s_{49} \cdot s_{50} + s_{49} + s_{50} + s_{67} \cdot s_{99} + s_{99} + 1$

i	ρ_i	β_i
50	$r_{49} + r_{50} \cdot r_{67} + r_{50} \cdot s_{34} + r_{99}$	$s_{49} + s_{50} \cdot s_{51}$
51	$r_{50} + r_{51} \cdot r_{67} + r_{51} \cdot s_{34}$	$r_{33} \cdot s_{99} + s_{50} + s_{51} \cdot s_{52} + s_{67} \cdot s_{99} + s_{99}$
52	$r_{51} + r_{52} \cdot r_{67} + r_{52} \cdot s_{34} + r_{99}$	$r_{33} \cdot s_{99} + s_{51} + s_{52} \cdot s_{53} + s_{67} \cdot s_{99}$
53	$r_{52} + r_{53} \cdot r_{67} + r_{53} \cdot s_{34}$	$s_{52} + s_{53} \cdot s_{54} + s_{53}$
54	$r_{53} + r_{54} \cdot r_{67} + r_{54} \cdot s_{34} + r_{99}$	$r_{33} \cdot s_{99} + s_{53} + s_{54} \cdot s_{55} + s_{55} + s_{67} \cdot s_{99} + s_{99}$
55	$r_{54} + r_{55} \cdot r_{67} + r_{55} \cdot s_{34}$	$s_{54} + s_{55} \cdot s_{56} + s_{55}$
56	$r_{55} + r_{56} \cdot r_{67} + r_{56} \cdot s_{34} + r_{99}$	$s_{55} + s_{56} \cdot s_{57} + s_{56} + s_{57} + s_{99} + 1$
57	$r_{56} + r_{57} \cdot r_{67} + r_{57} \cdot s_{34}$	$r_{33} \cdot s_{99} + s_{56} + s_{57} \cdot s_{58} + s_{57} + s_{67} \cdot s_{99} + s_{99}$
58	$r_{57} + r_{58} \cdot r_{67} + r_{58} \cdot s_{34} + r_{99}$	$r_{33} \cdot s_{99} + s_{57} + s_{58} \cdot s_{59} + s_{67} \cdot s_{99} + s_{99}$
59	$r_{58} + r_{59} \cdot r_{67} + r_{59} \cdot s_{34}$	$s_{58} + s_{59} \cdot s_{60} + s_{60} + s_{99}$
60	$r_{59} + r_{60} \cdot r_{67} + r_{60} \cdot s_{34} + r_{99}$	$s_{59} + s_{60} \cdot s_{61} + s_{61}$
61	$r_{60} + r_{61} \cdot r_{67} + r_{61} \cdot s_{34} + r_{99}$	$r_{33} \cdot s_{99} + s_{60} + s_{61} \cdot s_{62} + s_{61} + s_{62} + s_{67} \cdot s_{99} + s_{99} + 1$
62	$r_{61} + r_{62} \cdot r_{67} + r_{62} \cdot s_{34}$	$r_{33} \cdot s_{99} + s_{61} + s_{62} \cdot s_{63} + s_{62} + s_{63} + s_{67} \cdot s_{99} + 1$
63	$r_{62} + r_{63} \cdot r_{67} + r_{63} \cdot s_{34} + r_{99}$	$r_{33} \cdot s_{99} + s_{62} + s_{63} \cdot s_{64} + s_{63} + s_{67} \cdot s_{99} + s_{99}$
64	$r_{63} + r_{64} \cdot r_{67} + r_{64} \cdot s_{34} + r_{99}$	$r_{33} \cdot s_{99} + s_{63} + s_{64} \cdot s_{65} + s_{64} + s_{67} \cdot s_{99}$
65	$r_{64} + r_{65} \cdot r_{67} + r_{65} \cdot s_{34} + r_{99}$	$s_{64} + s_{65} \cdot s_{66} + s_{65} + s_{66} + s_{99} + 1$
66	$r_{65} + r_{66} \cdot r_{67} + r_{66} \cdot s_{34} + r_{99}$	$s_{65} + s_{66} \cdot s_{67} + s_{66}$
67	$r_{66} + r_{67} \cdot s_{34} + r_{67} + r_{99}$	$r_{33} \cdot s_{99} + s_{66} + s_{67} \cdot s_{68} + s_{67} \cdot s_{99} + s_{68}$
68	$r_{67} \cdot r_{68} + r_{67} + r_{68} \cdot s_{34}$	$s_{67} + s_{68} \cdot s_{69} + s_{68}$
69	$r_{67} \cdot r_{69} + r_{68} + r_{69} \cdot s_{34}$	$r_{33} \cdot s_{99} + s_{67} \cdot s_{99} + s_{68} + s_{69} \cdot s_{70} + s_{70}$
70	$r_{67} \cdot r_{70} + r_{69} + r_{70} \cdot s_{34}$	$s_{69} + s_{70} \cdot s_{71} + s_{70} + s_{71} + 1$
71	$r_{67} \cdot r_{71} + r_{70} + r_{71} \cdot s_{34} + r_{99}$	$s_{70} + s_{71} \cdot s_{72} + s_{71} + s_{72} + 1$
72	$r_{67} \cdot r_{72} + r_{71} + r_{72} \cdot s_{34} + r_{99}$	$r_{33} \cdot s_{99} + s_{67} \cdot s_{99} + s_{71} + s_{72} \cdot s_{73} + s_{72} + s_{73} + 1$
73	$r_{67} \cdot r_{73} + r_{72} + r_{73} \cdot s_{34}$	$s_{72} + s_{73} \cdot s_{74} + s_{74}$
74	$r_{67} \cdot r_{74} + r_{73} + r_{74} \cdot s_{34}$	$r_{33} \cdot s_{99} + s_{67} \cdot s_{99} + s_{73} + s_{74} \cdot s_{75} + s_{74} + s_{75} + 1$
75	$r_{67} \cdot r_{75} + r_{74} + r_{75} \cdot s_{34}$	$r_{33} \cdot s_{99} + s_{67} \cdot s_{99} + s_{74} + s_{75} \cdot s_{76} + s_{75} + s_{76} + s_{99} + 1$
76	$r_{67} \cdot r_{76} + r_{75} + r_{76} \cdot s_{34}$	$r_{33} \cdot s_{99} + s_{67} \cdot s_{99} + s_{75} + s_{76} \cdot s_{77} + s_{76} + s_{77} + s_{99} + 1$
77	$r_{67} \cdot r_{77} + r_{76} + r_{77} \cdot s_{34}$	$s_{76} + s_{77} \cdot s_{78} + s_{77} + s_{78} + 1$
78	$r_{67} \cdot r_{78} + r_{77} + r_{78} \cdot s_{34}$	$s_{77} + s_{78} \cdot s_{79} + s_{99}$
79	$r_{67} \cdot r_{79} + r_{78} + r_{79} \cdot s_{34} + r_{99}$	$r_{33} \cdot s_{99} + s_{67} \cdot s_{99} + s_{78} + s_{79} \cdot s_{80} + s_{80}$
80	$r_{67} \cdot r_{80} + r_{79} + r_{80} \cdot s_{34} + r_{99}$	$r_{33} \cdot s_{99} + s_{67} \cdot s_{99} + s_{79} + s_{80} \cdot s_{81}$
81	$r_{67} \cdot r_{81} + r_{80} + r_{81} \cdot s_{34} + r_{99}$	$r_{33} \cdot s_{99} + s_{67} \cdot s_{99} + s_{80} + s_{81} \cdot s_{82} + s_{81} + s_{82} + 1$
82	$r_{67} \cdot r_{82} + r_{81} + r_{82} \cdot s_{34} + r_{99}$	$r_{33} \cdot s_{99} + s_{67} \cdot s_{99} + s_{81} + s_{82} \cdot s_{83} + s_{83} + s_{99}$
83	$r_{67} \cdot r_{83} + r_{82} + r_{83} \cdot s_{34}$	$s_{82} + s_{83} \cdot s_{84} + s_{84} + s_{99}$
84	$r_{67} \cdot r_{84} + r_{83} + r_{84} \cdot s_{34}$	$r_{33} \cdot s_{99} + s_{67} \cdot s_{99} + s_{83} + s_{84} \cdot s_{85} + s_{85}$
85	$r_{67} \cdot r_{85} + r_{84} + r_{85} \cdot s_{34}$	$s_{84} + s_{85} \cdot s_{86} + s_{86} + s_{99}$
86	$r_{67} \cdot r_{86} + r_{85} + r_{86} \cdot s_{34}$	$s_{85} + s_{86} \cdot s_{87} + s_{86} + s_{87} + s_{99} + 1$
87	$r_{67} \cdot r_{87} + r_{86} + r_{87} \cdot s_{34} + r_{99}$	$s_{86} + s_{87} \cdot s_{88} + s_{87} + s_{99}$
88	$r_{67} \cdot r_{88} + r_{87} + r_{88} \cdot s_{34} + r_{99}$	$s_{87} + s_{88} \cdot s_{89} + s_{88} + s_{89} + 1$
89	$r_{67} \cdot r_{89} + r_{88} + r_{89} \cdot s_{34} + r_{99}$	$s_{88} + s_{89} \cdot s_{90}$
90	$r_{67} \cdot r_{90} + r_{89} + r_{90} \cdot s_{34} + r_{99}$	$r_{33} \cdot s_{99} + s_{67} \cdot s_{99} + s_{89} + s_{90} \cdot s_{91} + s_{91} + s_{99}$
91	$r_{67} \cdot r_{91} + r_{90} + r_{91} \cdot s_{34} + r_{99}$	$r_{33} \cdot s_{99} + s_{67} \cdot s_{99} + s_{90} + s_{91} \cdot s_{92} + s_{99}$
92	$r_{67} \cdot r_{92} + r_{91} + r_{92} \cdot s_{34} + r_{99}$	$r_{33} \cdot s_{99} + s_{67} \cdot s_{99} + s_{91} + s_{92} \cdot s_{93} + s_{92} + s_{99}$
93	$r_{67} \cdot r_{93} + r_{92} + r_{93} \cdot s_{34}$	$s_{92} + s_{93} \cdot s_{94}$
94	$r_{67} \cdot r_{94} + r_{93} + r_{94} \cdot s_{34} + r_{99}$	$r_{33} \cdot s_{99} + s_{67} \cdot s_{99} + s_{93} + s_{94} \cdot s_{95}$
95	$r_{67} \cdot r_{95} + r_{94} + r_{95} \cdot s_{34} + r_{99}$	$r_{33} \cdot s_{99} + s_{67} \cdot s_{99} + s_{94} + s_{95} \cdot s_{96} + s_{95} + s_{99}$
96	$r_{67} \cdot r_{96} + r_{95} + r_{96} \cdot s_{34} + r_{99}$	$r_{33} \cdot s_{99} + s_{67} \cdot s_{99} + s_{95} + s_{96} \cdot s_{97} + s_{96} + s_{99}$
97	$r_{67} \cdot r_{97} + r_{96} + r_{97} \cdot s_{34} + r_{99}$	$s_{96} + s_{97} \cdot s_{98} + s_{98}$
98	$r_{67} \cdot r_{98} + r_{97} + r_{98} \cdot s_{34}$	$s_{97} + s_{98} \cdot s_{99} + s_{99}$
99	$r_{67} \cdot r_{99} + r_{98} + r_{99} \cdot s_{34}$	$r_{33} \cdot s_{99} + s_{67} \cdot s_{99} + s_{98}$

Appendix C: Fault Location Signatures (In Hexadecimal)

I. Signatures for fault locations in $\boldsymbol{R}$

In the following table we list the vectors $\Psi^1_{r_\phi}$ as hexadecimal constants.

ϕ	$\Psi^1_{r_\phi}$	ϕ	$\Psi^1_{r_\phi}$	ϕ	$\Psi^1_{r_\phi}$	ϕ	$\Psi^1_{r_\phi}$	ϕ	$\Psi^1_{r_\phi}$
0	000FFFFFE0	20	FFFE000000	40	FFFFFFF000	60	FF00000000	80	FFFFF00000
1	FFFFFFFFC0	21	FFFC000000	41	FFFFFFE000	61	FE00000000	81	FFFFE00000
2	FFFFFFFF80	22	FFF8000000	42	FFFFFFC000	62	FC00000000	82	FFFFC00000
3	FFFFFFFF00	23	FFF0000000	43	FFFFFF8000	63	F800000000	83	FFFF800000
4	FFFFFFFE00	24	FFE0000000	44	FFFFFF0000	64	F000000000	84	FFFF000000
5	FFFFFFFC00	25	FFC0000000	45	FFFFFE0000	65	E000000000	85	FFFE000000
6	FFFFFFF800	26	FF80000000	46	FFFFFC0000	66	C000000000	86	FFFC000000
7	FFFFFFF000	27	FF00000000	47	FFFFF80000	67	8000000000	87	FFF8000000
8	FFFFFFE000	28	FE00000000	48	FFFFF00000	68	FFFFFFFF00	88	FFF0000000
9	FFFFFFC000	29	FC00000000	49	FFFFE00000	69	FFFFFFFE00	89	FFE0000000
10	FFFFFF8000	30	F800000000	50	FFFFC00000	70	FFFFFFFC00	90	FFC0000000
11	FFFFFF0000	31	F000000000	51	FFFF800000	71	FFFFFFF800	91	FF80000000
12	FFFFFE0000	32	E000000000	52	FFFF000000	72	FFFFFFF000	92	FF00000000
13	FFFFFC0000	33	C000000000	53	FFFE000000	73	FFFFFFE000	93	FE00000000
14	FFFFF80000	34	FFFFFFFFC0	54	FFFC000000	74	FFFFFFC000	94	FC00000000
15	FFFFF00000	35	FFFFFFFF80	55	FFF8000000	75	FFFFFF8000	95	F800000000
16	FFFFE00000	36	FFFFFFFF00	56	FFF0000000	76	FFFFFF0000	96	F000000000
17	FFFFC00000	37	FFFFFFFE00	57	FFE0000000	77	FFFFFE0000	97	E000000000
18	FFFF800000	38	FFFFFFFC00	58	FFC0000000	78	FFFFFC0000	98	C000000000
19	FFFF000000	39	FFFFFFF800	59	FF80000000	79	FFFFF80000	99	8000000000

We now list the vectors $\Psi^2_{r_\phi}$. Note that $\Psi^2_{r_0} = \texttt{8000000000}$, $\Psi^2_{r_\phi} = \texttt{0000000000}\ \forall \phi \in [1, 67]$.

ϕ	$\Psi^2_{r_\phi}$	ϕ	$\Psi^2_{r_\phi}$	ϕ	$\Psi^2_{r_\phi}$	ϕ	$\Psi^2_{r_\phi}$
68	0000000080	76	0000008000	84	0000800000	92	0080000000
69	0000000100	77	0000010000	85	0001000000	93	0100000000
70	0000000200	78	0000020000	86	0002000000	94	0200000000
71	0000000400	79	0000040000	87	0004000000	95	0400000000
72	0000000800	80	0000080000	88	0008000000	96	0800000000
73	0000001000	81	0000100000	89	0010000000	97	1000000000
74	0000002000	82	0000200000	90	0020000000	98	2000000000
75	0000004000	83	0000400000	91	0040000000	99	4000000000

II. Signatures for fault locations in $\boldsymbol{S}$

Now we list the vectors $\Psi^1_{s_\phi}$ as hexadecimal constants.

ϕ	$\Psi^1_{s_\phi}$	ϕ	$\Psi^1_{s_\phi}$	ϕ	$\Psi^1_{s_\phi}$	ϕ	$\Psi^1_{s_\phi}$	ϕ	$\Psi^1_{s_\phi}$
0	7FFFFFFFE0	20	FFFE000000	40	FE00000000	60	FF80000000	80	FFFFC00000
1	FFFFFFFFC0	21	FFFC000000	41	FF00000000	61	FF00000000	81	FFFFE00000
2	FFFFFFFF80	22	FFF8000000	42	FF80000000	62	FE00000000	82	FFFFC00000
3	FFFFFFFF00	23	FFF0000000	43	FFE0000000	63	FC00000000	83	FFFF800000
4	FFFFFFFE00	24	FFE0000000	44	FFF0000000	64	F800000000	84	FFFF000000
5	FFFFFFFC00	25	FFC0000000	45	FFFC000000	65	F000000000	85	FFFE000000
6	FFFFFFF800	26	FF80000000	46	FFFF800000	66	E000000000	86	FFFC000000
7	FFFFFFF000	27	FF00000000	47	FFFF800000	67	C000000000	87	FFF8000000
8	FFFFFFE000	28	FE00000000	48	FFFFF80000	68	E000000000	88	FFF0000000
9	FFFFFFC000	29	FC00000000	49	FFFFF00000	69	F000000000	89	FFE0000000
10	FFFFFF8000	30	F800000000	50	FFFFE00000	70	F800000000	90	FFC0000000
11	FFFFFF0000	31	F000000000	51	FFFFC00000	71	FC00000000	91	FF80000000
12	FFFFFE0000	32	E000000000	52	FFFF800000	72	FE00000000	92	FF00000000
13	FFFFFC0000	33	C000000000	53	FFFF000000	73	FF00000000	93	FE00000000
14	FFFFF80000	34	8000000000	54	FFFE000000	74	FF80000000	94	FC00000000
15	FFFFF00000	35	C000000000	55	FFFC000000	75	FFC0000000	95	F800000000
16	FFFFE00000	36	E000000000	56	FFF8000000	76	FFE0000000	96	F000000000
17	FFFFC00000	37	F000000000	57	FFF0000000	77	FFF8000000	97	E000000000
18	FFFF800000	38	F800000000	58	FFE0000000	78	FFF8000000	98	C000000000
19	FFFF000000	39	FC00000000	59	FFC0000000	79	FFFD000000	99	8000000000

We now list the vectors $\Psi^2_{s_\phi}$. Note that $\Psi^2_{s_0} = \mathsf{8000000000}$, $\Psi^2_{s_\phi} = \mathsf{0000000000}\ \forall \phi \in [1, 81]$.

ϕ	$\Psi^2_{s_\phi}$	ϕ	$\Psi^2_{s_\phi}$	ϕ	$\Psi^2_{s_\phi}$	ϕ	$\Psi^2_{s_\phi}$	ϕ	$\Psi^2_{s_\phi}$
82	0000200000	86	0002000000	90	0020000000	94	0200000000	98	2000000000
83	0000400000	87	0004000000	91	0040000000	95	0400000000	99	4000000000
84	0000800000	88	0008000000	92	0080000000	96	0800000000		
85	0001000000	89	0010000000	93	0100000000	97	1000000000		