A Cryptographic Review of CIPHERUNICORN-A

M.J.B. Robshaw
16d Stowe Rd.
London
W12 8BN
mrobshaw@supanet.com

December 14, 2001

Executive summary

This report describes a brief cryptographic review of CIPHERUNICORN-A. While a broad range of cryptanalytic attacks were considered during this work, our attention was particularly focused on the differential and linear cryptanalysis of the cipher as requested. CIPHERUNICORN-A is a complicated cipher which hinders accurate analysis. This should be contrasted with other ciphers that permit a reasonably close assessment. However we draw the following conclusions.

The techniques of the designers used to estimate the effectiveness of differential and linear cryptanalysis appear to be reasonable given the complexity of the cipher. There remain some open questions with regards to this analysis and it is unknown how closely it might compare to actuality. However, there is currently little evidence to believe that existing analysis is unreasonable in predicting the general resistance of the cipher to advanced cryptanalytic attack.

While slight improvements to bounds on the probability of a linear approximation for CIPHERUNICORN-A can be made, with our current understanding practical differential and linear cryptanalytic attacks on CIPHERUNICORN-A seem unlikely. There is a certificational weakness in the key schedule of CIPHERUNICORN-A. However any practical impact on security was not immediately clear.

This review took place over a limited time and with limited resources. It should be anticipated that additional analysis may well yield improved results for the cryptanalysis of this cipher and provide a greater understanding of the true security offered.

1 Introduction

In this report we present the results of a brief cryptographic review of the block cipher CIPHERUNICORN-A. This cipher has been submitted to the *Cryptrec Evaluation* process and has already received considerable study by the designers of the algorithm. CIPHERUNICORN-A is a companion to CIPHERUNICORN-E and they share some functional components. However the specific details of the ciphers are sufficiently different that surprisingly little of the analysis from one cipher is of immediate relevance to the other.

While some effort was been made to consider a broad range of attacks on the cipher, most effort was concentrated on considering the effectiveness of differential and linear cryptanalysis. The materials provided for this evaluation were

- Cryptographic techniques specifications: CIPHERUNICORN-A, FY 2000 submission, NEC Corporation. (Undated.)
- Notice of updates to the above report. NEC Corporation. (Undated.)
- Cryptographic techniques specifications: CIPHERUNICORN-A, Version 2, NEC Corporation [17].
- Self Evaluation Report: CIPHERUNICORN-A, Version 2, FY 2000 submission, NEC Corporation. (Undated.)
- Notice of updates to the above report. NEC Corporation. (Undated.)
- Self Evaluation Report: CIPHERUNICORN-A, Version 3, NEC Corporation [18].
- Copy of overhead slides: "128-bit Block Cipher CIPHERUNICORN-A (UNI-A)", NEC Corporation. (Undated.)
- An independent cryptographic review of CIPHERUNICORN-A [1].

2 Terminology, definitions, and notation

Throughout this report we assume that the reader is familiar with many different aspects of block cipher design and analysis, particularly differential [2] and linear [13] cryptanalysis.

When we consider differential cryptanalysis, we will use a notion of difference given by bitwise exclusive-or. While other notions of difference might be considered, the design of CIPHERUNICORN-A is such that this particular measure is likely to be the most useful. General differences will be denoted by Δ. For linear cryptanalysis we will require the use of so-called *parity masks* denoted by

Γ. When explicit values to either differences or parity masks are required they will be represented in hexadecimal notation prefixed with `0x`.

CIPHERUNICORN-A relies on several structural components. These include integer addition modulo 2^{32} which we will denote by $+$ and the bitwise exclusive-or of 8, 32, and 64-bit data units which we will denote by $\oplus$. CIPHERUNICORN-A requires integer multiplication modulo 2^{32} represented by $\times$ and we will refer to the four different 8-bit to 8-bit substitution boxes S_0–S_3 as S-boxes. The cipher requires the use of a bitwise rotate to the left as well as the bitwise `and` of words. The rotation of a to the left by r bit positions will be denoted by $a \lll r$ while the bitwise `and` of a and b will be denoted by $a \wedge b$. The Hamming weight of a word a is defined as the number of ones in the binary representation of the word.

3 Existing analysis of CIPHERUNICORN-A

The designers of CIPHERUNICORN-A have provided the results of their own evaluation of the cipher [18]. The bulk of this analysis appears to be concentrated on the results of extensive statistical testing. While this approach is not entirely without some merit, it is very unlikely that such testing, no matter how extensive, will uncover problems with the cipher. While a cipher must certainly pass such tests, a cipher that passes is not necessarily secure. The designers also consider the resistance of the cipher to a wide-range of sophisticated cryptanalytic attacks. In particular, bounds on the effectiveness of differential and linear cryptanalysis have been derived.

In addition to the self-evaluation report [18] an independent cryptographic review [1] of CIPHERUNICORN-A has observed several interesting structural features in the cipher. With regards to differential and linear cryptanalysis, this report concluded that practical attacks were unlikely.

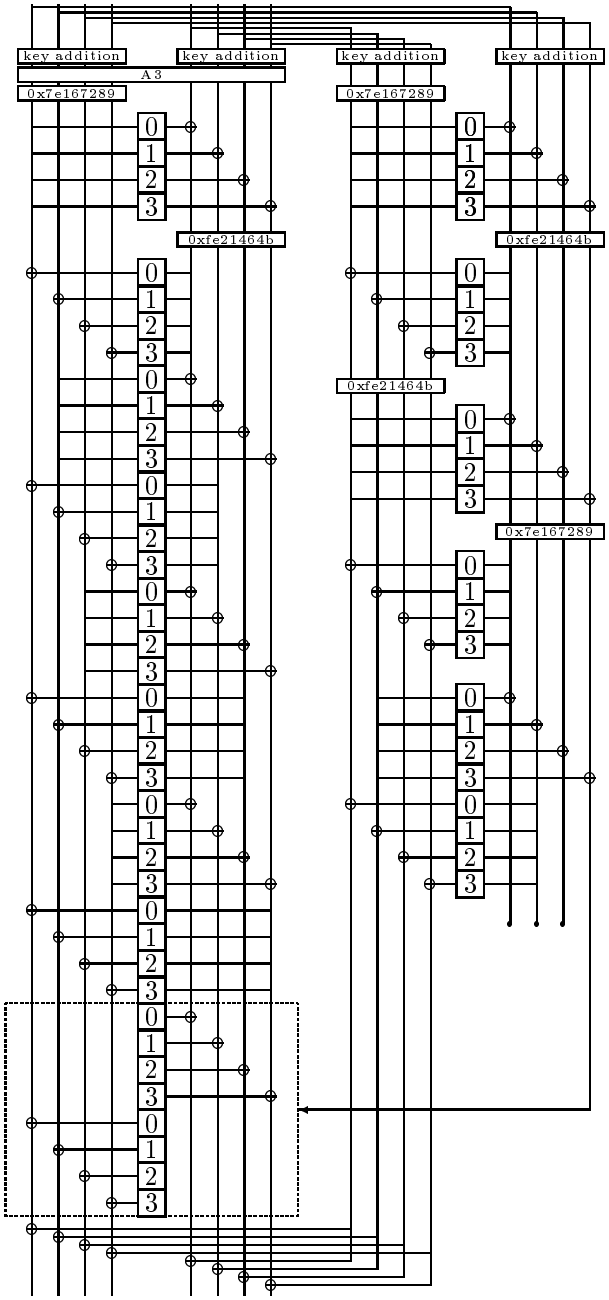
4 Description of CIPHERUNICORN-A

CIPHERUNICORN-A has a Feistel-like structure and uses pre-whitening, and post-whitening of the encryption routine with key dependent material. The round function takes as input two 32-bit words and provides two 32-bit words as output. Given the block size of 128 bits and the allowed key lengths of 128, 192, and 256 bits, CIPHERUNICORN-A might well be intended as a drop-in replacement to the AES [5, 16].

4.1 The round function

The round function for CIPHERUNICORN-A is complicated. There are two parallel computations that take place which we will refer to as Computation I and Computation II (see Figure 1).

Figure 1: The round function for CIPHERUNICORN-A.



Computation I is the heavier computation and involves a Feistel-like structure on two 32-bit words with ten mini-rounds. Each round takes as input one of the four bytes from one of the 32-bit strands. Each byte of the 32-bit target strand is then exclusive-ored with the output from the four S-boxes for which the same source byte was used. The input bytes are taken in a regular and natural order, and after eight mini-rounds have been completed two more mini-rounds are computed. The source bytes for these last two mini-rounds is determined as a result of Computation II. In Figure 1 this is indicated with a dashed box and S-boxes without a source byte. As well as providing one of sixteen possible combinations for the final two mini-rounds, Computation II also provides a 32-bit quantity that is exclusive-ored with both 32-bit outputs from the round function.

Key material is combined with the input to both Computation I and Computation II using integer addition. Within Computation I and Computation II, integer multiplication by a fixed constant is used immediately prior to the use of the most significant byte as a source byte for a mini-round.

4.2 The function A3

The A3 function is used within Computation I. It seems intended to provide improved mixing within the round function. Two 32-bit words are input to A3 which provides two 32-bit words as output. We will write $A3(a, b) = (x, y)$ where

$$(x||y) = (a||b) \oplus ((a||b) \lll 23) \oplus ((a||b) \lll 41).$$

The role and effectiveness of this function will be explored in Section 5.

4.3 The S-boxes

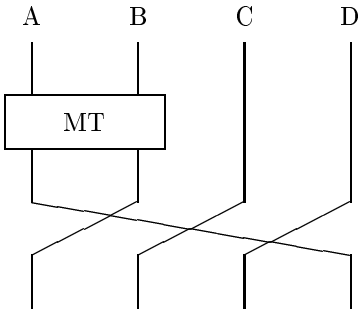
There are four different S-boxes S_0–S_3 used in CIPHERUNICORN-A. They are designed along similar principles to those used in the design of the new AES [5, 16]. The construction and the properties of the S-boxes have not been checked during this work, and it is assumed that they have the properties claimed.

4.4 The key schedule

The key schedule is simple (though computationally intensive) and it is based around the repeated application of a function denoted MT. MT takes as input two 32-bit quantities and provides as output two 32-bit quantities according to the following equations. We will write $\mathrm{MT}(a, b) = (x, y)$ where

$$\begin{aligned} x &= a \times \texttt{0x01010101} \\ t &= (x \ggg 24) \wedge \texttt{0x000000ff} \\ y &= b \oplus (S_0[t]||S_1[t]||S_2[t]||S_3[t]) \end{aligned}$$

The function MT is used within a network of $\frac{k}{32}$ strands where k is one of 128, 192, or 256 bits in length. For the case of $k = 128$, the network has four strands A–D as shown here.



We will make some simple observations on this process in Section 8.

4.5 Initial comments

The round function for CIPHERUNICORN-A is complicated. Certainly it makes analysis difficult, but the fact that there are two strands of computation running concurrently with only limited interaction raises several questions.

It seems to be the case that the bulk of any cryptographic strength derives from Computation I. The results of Computation II are used to vary the operation flow in Computation I in modest ways. First, four bits from Computation II are used to determine which particular mini-rounds are used in the last two mini-rounds of Computation I. Second, 32 bits of key- and text-dependent material are exclusive-ored to both outputs in Computation I. Given that there are only 16 possibilities for the last two mini-rounds anyway, and that the second effect is of questionable value with regards to a differential attack, the additional benefit of Computation II is not so clear.

During both Computation I and Computation II integer multiplication by a fixed odd constant is used as a means of complicating analysis and providing diffusion. A similar technique is used in the cipher MARS [3]. While a single-bit difference in the most significant bit (Δ = `0x80000000`) and the parity of the least significant bit (Γ = `0x00000001`) are preserved across the integer multiplication, it seems that multiplication can generally be an effective way of increasing the avalanche of change in a cipher. This is particularly the case when the higher-order bits of the output from the multiplication operation are used in immediately subsequent operations (since multiplication tends to propagate bitwise differences to the higher-order bits of the product).

4.6 Some simplifications to CIPHERUNICORN-A

It is unfortunate that the round function of CIPHERUNICORN-A is too complicated to be readily analyzed. In fact, it seems to have been a design aim that security be derived through having a round function that is difficult to analyze. This runs counter to the design philosophy of many other cipher designers.

To help gain a greater intuition into the true security that might be offered by CIPHERUNICORN-A we will consider two simplified variants of the round function.

1. The designers consider a variant of CIPHERUNICORN-A in which the integer addition of key material is ignored and each integer multiplication $a \times c$ is approximated by

$$a \oplus ((a \wedge \texttt{0xff}) \ll 24) \oplus ((a \wedge \texttt{0xff00}) \ll 16) \oplus ((a \wedge \texttt{0xff0000}) \ll 8).$$

   While the designers of CIPHERUNICORN-A call this modified round function mF, we will denote this variant by UNI-A-REP-MULT to show that the multiplication step has been replaced. It is illustrated in Figure 2.

2. Since the function A3 is too complicated to consider naturally we will define a simpler variant of CIPHERUNICORN-A. It is unlikely that this variant without the function A3 will be stronger than one with A3, so we will consider a version of CIPHERUNICORN-A with the A3 function removed and all multiplication constants fixed to 1. We will label this variant of the cipher UNI-A-NO-A3.

5 The Function A3

The A3 function takes as input two 32-bit words (viewed as a 64-bit word by simple concatenation) and provides two 32-bit words (derived from a 64-bit word by truncation) as output. It appears to be intended to provide mixing across the whole of the input to the round function and we have $A3(a, b) = (x, y)$ where

$$(x||y) = (a||b) \oplus ((a||b) \lll 23) \oplus ((a||b) \lll 41).$$

An equivalent way of looking at A3 is shown here.

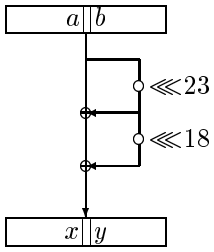
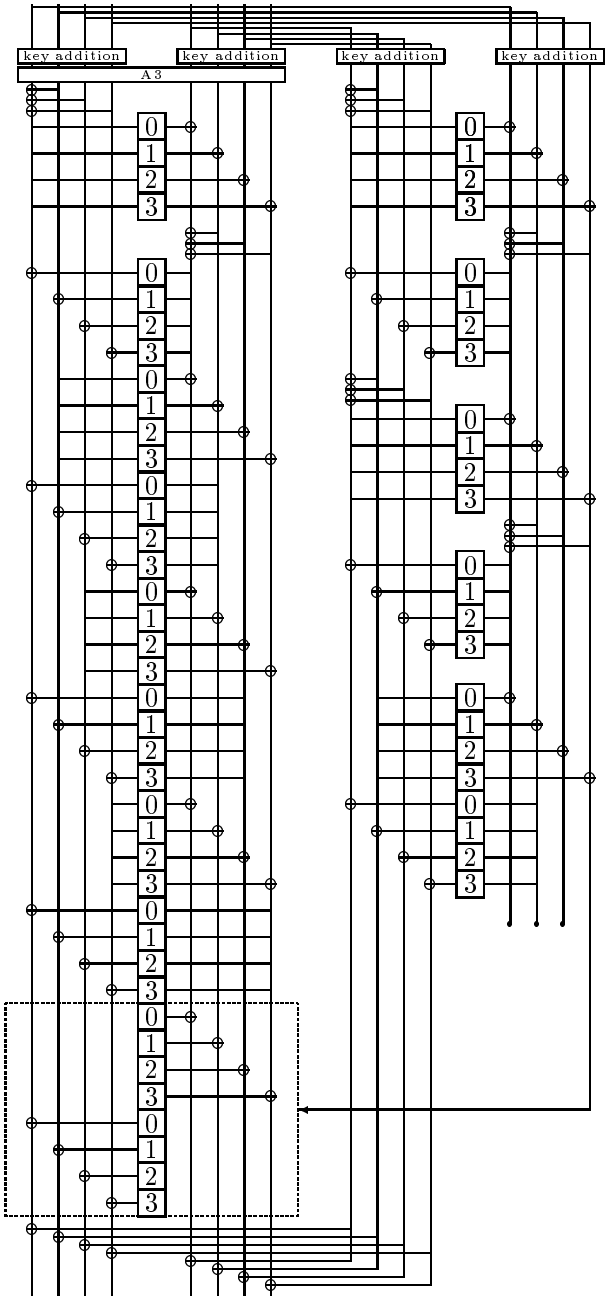


Figure 2: A modified round function for CIPHERUNICORN-A.



At one level it might be reasoned that A3 could be irrelevant with regards to differential cryptanalysis since $\mathrm{A3}(r \oplus s) = \mathrm{A3}(r) \oplus \mathrm{A3}(s)$. Indeed, in establishing conservative bounds for a characteristic or differential it is reasonable to assume that A3 offers no resistance to the cryptanalyst. However when it comes to any particularly detailed analysis, the bit positions involved in a specified characteristic are likely to be important and A3 is still likely to play a role in hindering the development of differential and linear attacks.

With regards to differential cryptanalysis there are some obvious degenerate examples where A3 actually reduces the Hamming weight of a characteristic. For example $\mathrm{A3}(\texttt{0xeeeeeeeeeeeeeeee}) = \texttt{0x4444444444444444}$ and the Hamming weight has dropped from 48 to 16. However these and other degenerate examples are unlikely to be of much significance to the overall security of the cipher. Certainly if our concern is input differences of low Hamming weight, then it seems that the function A3 is likely to produce higher Hamming weight output differences thereby helping the avalanche of change in the cipher.

With regards to linear cryptanalysis the situation is slightly different. It is fairly easy to identify linear approximations across the function A3 (holding with probability one) for which the Hamming weight of the input parity relation is low, yet the Hamming weight of the output parity relation is even lower. For instance if we set

$$\begin{aligned} \Gamma_1 &= \texttt{0x0000020000800001} \text{ and} \\ \Gamma_2 &= \texttt{0x0000000000000001} \end{aligned}$$

then we have that

$$\mathrm{A3}(a) \cdot \Gamma_2 = a \cdot \Gamma_1.$$

While such instances are interesting, it is difficult to see how they might be of specific use to the cryptanalyst. Particularly so when we consider the likely resistance of CIPHERUNICORN-A to linear cryptanalysis in general (see Section 7).

6 Differential Cryptanalysis

Differential cryptanalysis was invented by Biham and Shamir [2]. While some advanced variants have been proposed [8, 9, 12] these will not be our concern in this report. In differential cryptanalysis, the cryptanalyst attempts to predict (with some probability) the evolution of a difference between two inputs as they pass through the encryption process. The notion of difference can vary depending on the cipher, but in this case it seems that bitwise exclusive-or would be most appropriate.

The evolution of the difference can be expressed in different ways. It is typical to trace this evolution in an exact manner, defining an input and output for a given operation in the encryption process. Under certain assumptions, the

probability of this path (which is called a *characteristic*) is estimated by the product of the probabilities at each step in the process.

It is typical to assume that a cryptanalyst aims to find a 14-round characteristic when attacking a 16-round cipher. (We assume that the two outer rounds of the cipher can be removed in what is frequently referred to as a 2R-attack.) The success of such an attack is dependent on the probability of the identified characteristic. Actually it is more accurate to say that the success of the attack is dependent on the accumulated probability of all possible characteristics with the same starting and ending difference. Thus accumulation of relevant characteristics is typically termed a *differential* [11]. Throughout this section we will switch our attention between characteristics and differentials as the need arises.

In the self-evaluation report [18] the designers of CIPHERUNICORN-A provided some conservative estimates for the resistance of the cipher to differential cryptanalysis. In this section we will look at their technique, consider our own independent approach and provide our own conclusions on the resistance of CIPHERUNICORN-A to differential cryptanalysis.

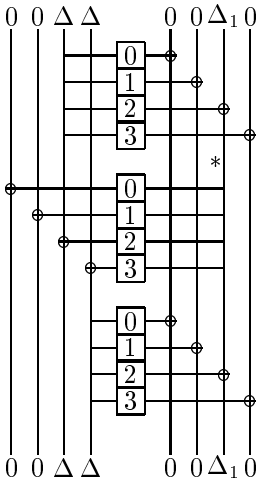
6.1 Differential cryptanalysis of UNI-A-REP-MULT

In this section we will consider the round function shown in Figure 2 which was used by the designers in evaluating the resistance of CIPHERUNICORN-A to differential cryptanalysis.

In their self-evaluation report [18] the designers make conservative estimates on the resistance of the cipher to differential cryptanalysis. In particular they consider the resistance of UNI-A-REP-MULT to differential cryptanalysis. The designers claim that an upper bound on the probability of a differential for a round of UNI-A-REP-MULT is around 2^{-12}. At the level of detail provided in the self-evaluation report [18] and with the limited time available for this review, no improvements over the basic differential attack for one round were observed. The designers then took the reasonable step of assuming that an upper bound for the probability of a differential over one round of UNI-A-REP-MULT would likely provide a conservative estimate for the probability of a differential over one round of CIPHERUNICORN-A. From this, estimates on bounds for the probability of a differential for CIPHERUNICORN-A can be derived.

6.2 Differential cryptanalysis of UNI-A-NO-A3

To pursue our own independent analysis of CIPHERUNICORN-A we consider the simplified variant UNI-A-NO-A3. We start by observing the following differential feature in CIPHERUNICORN-A.



Suppose we find some Δ and Δ_1 such that $\Delta \xrightarrow{S_2} \Delta_1$ with probability $p = 2^{-6}$. Then with probability 2^{-6} the byte difference at the position marked (*) will be zero. This means that the same inputs will be used to the second set of S-boxes for both texts in the pair. This results in the same outputs. Then since the difference in the input to the third set of S-boxes is the same as the difference to the first set of S-boxes, with probability 2^{-7} the actual inputs themselves will be the same. In such a case, we have exactly the same set of outputs in both the first and third set of S-boxes which cancel out. Thus the differential

$$(0, 0, \Delta, \Delta, 0, 0, \Delta_1, 0) \rightarrow (0, 0, \Delta, \Delta, 0, 0, \Delta_1, 0)$$

will hold with probability 2^{-13} across three sets of S-boxes. This is closely analogous to observations made independently in the external review [1].

Since $\Delta \xrightarrow{S_2} \Delta_1$ holds with probability 2^{-6} for $\Delta = \texttt{0x40}$ and $\Delta_1 = \texttt{0x20}$, for example, we can immediately use this differential structure in an attack on the full round function of Uni-A-no-A3 as follows. This is illustrated in Figure 3 and we can specify the following 64-bit differential

$$(0, 0, \texttt{0x40}, \texttt{0x40}, 0, 0, \texttt{0x20}, 0) \rightarrow (0, 0, \texttt{0x60}, \texttt{0x40}, 0, 0, 0, 0, 0)$$

that holds over the entire round of Cipherunicorn-A with probability $2^{-13} \times \frac{3}{4} \times \frac{1}{2} \approx 2^{-14.4}$. The factors of $\frac{3}{4}$ and $\frac{1}{2}$ are necessary since the mini-rounds at the end of the round need to take inputs with a zero difference. Note that we have assumed that the integer addition of key material in the round has had no effect on the evolution of the differential. This particular differential has low Hamming weight input and output differences. As a consequence the interaction between the integer addition of any key material and the exclusive-or difference may well be quite limited. Indeed, there are key values (namely `0x00000000`)

for which there is no interaction at all, so we might adopt a conservative position and use such cases in our analysis.

If we were to take a very conservative approach then we might say that the multiplication and A3 operations have no effect on the evolution of this differential. This would then give us an estimate of $2^{-14.4}$ for the probability of a one round differential in CIPHERUNICORN-A. Clearly this does not contradict the upper-bound of 2^{-12} claimed in the self-evaluation report [18]. However this differential is more detailed in that it is fully specified over all 64 bits of input and output. It has also been experimentally verified (see Section 6.3). Thus it gives some confirmation to the upper-bound presented in the self evaluation report, but it also hints at the fact that earlier bounds [18] might be tighter than expected.

Indeed the independent review [1] provides confirmation for this intuition. There a differential for part of the round function of CIPHERUNICORN-A is outlined that holds with an estimated probability of 2^{-13}. This differential incorporated the multiplication and A3 function (at least in an undetailed way). However it did not extend well when considering the action of Computation II. Instead, it was necessary to identify a truncated differential yielding 32 bits of information with a probability of 2^{-13} [1].

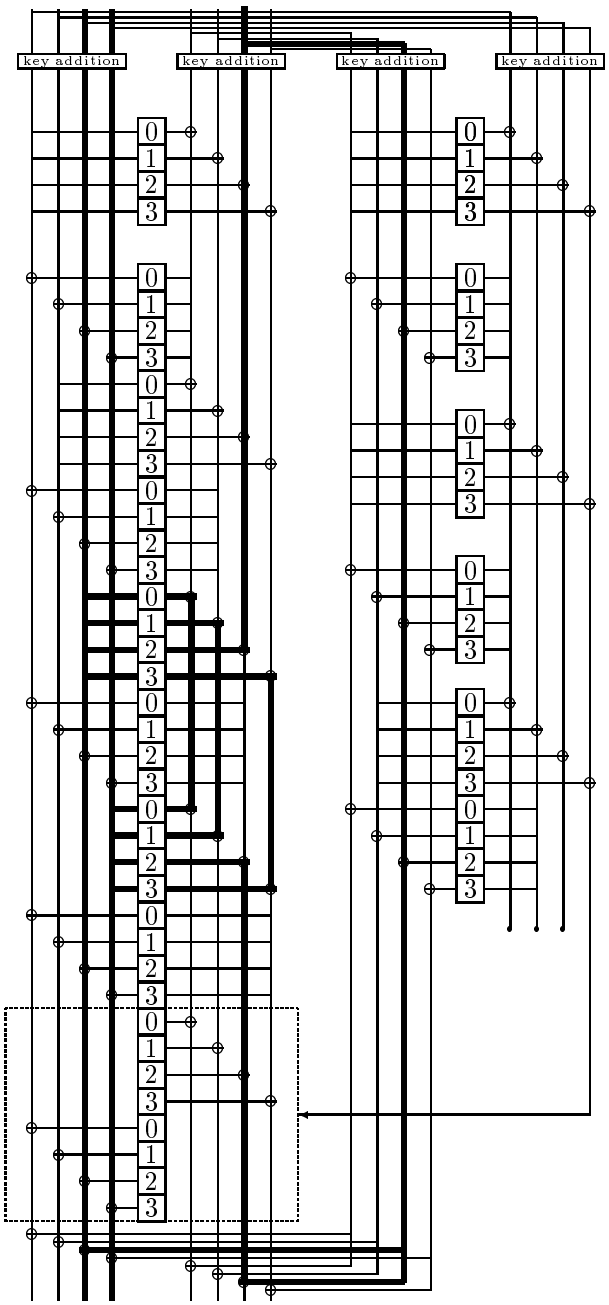
The differential presented here passes directly through both Computation I and Computation II of the modified round function. Further we obtain a fully specified 64-bit approximation that holds with probability $2^{-14.4}$. While it does not take account of the multiplications that are used, nor the presence of the A3 function (which would very likely hinder the joining of different round differentials), it does fall in line with the estimates provided in [1] and also (surprisingly) with the very conservative figures provided in the self-evaluation report [18].

6.3 Unanticipated effects

We implemented the round function of UNI-A-NO-A3 and verified the claimed probability of the differential in Figure 3. With a trial over 2^{24} randomly chosen input texts and with the keys taking the values specified in Section 6.2, the measured probability of the differential was $2^{-14.40}$. This matched the theoretical prediction.

While only very limited experimentation could be employed, there was little sign of any unexpected or degenerate differential behavior in the cipher. Thus, with current understanding, there seems little reason to question the normal technique of multiplying the probabilities of constituent characteristics and differentials in estimating the probability of a characteristic or differential for the cipher in its entirety.

Figure 3: A simple differential for Uni-A-no-A3.



6.4 Implications for the full cipher

Given the exceptional complexity of CIPHERUNICORN-A we are left with little alternative but to study much simplified versions of the cipher. However if we make too many changes then we are in no real position to assess how close to the true behavior of the cipher such variants might be.

Estimates given in the self-evaluation report [18] are very crude and conservative. While some specific improvements to the accuracy of the figures might be possible, significant improvements at a practical level for the cryptanalyst have not been apparent in this short review. In addition, limited experimentation found little evidence to question the typical approach of multiplying the probability of differentials and characteristics when they are joined. We might therefore conclude the following.

For CIPHERUNICORN-A it seems unlikely that an active differential for a single round could be readily identified that would hold with a probability much greater than 2^{-12} for even a small proportion of the keyspace. The function $A3$ and the integer multiplications are expected to make a tangible contribution to the security of CIPHERUNICORN-A.

Provided some care is taken, there is currently little evidence to question the typical approach of multiplying the probabilities of component differentials in estimating the probability of the complete differential. With the current state of knowledge, it would be reasonable to view CIPHERUNICORN-A as being practically resistant to differential cryptanalysis.

7 Linear Cryptanalysis

Linear cryptanalysis [13] has been very effective in the analysis of DES [14, 15] but less so against other ciphers. While there are some advanced variants to this basic attack [7, 10, 19], the practical significance of these methods is likely to be slight.

In linear cryptanalysis we are concerned with predicting the value of a single bit of information. This bit is typically formed as the exclusive-or combination of different bits in some word. The bits from a word a, say, forming the bit of information can be indicated by a (0,1)-vector Γ. The value of the bit we are interested in can be conveniently represented by the dot product $a \cdot \Gamma$. This single bit value will have the values zero and one with a certain probability p. The effectiveness of a linear cryptanalytic attack can be measured in terms of two closely related concepts: the bias ϵ and what we'll call the correlation coefficient LP. In the self-evaluation report [18] the correlation coefficient LP is used. In this report we will use the bias ϵ where $\epsilon = |1/2 - p|$.

It is an interesting consequence of the design of CIPHERUNICORN-A that the cryptanalyst will probably aim to use linear approximations involving the output of many S-boxes. Since the same input byte is used for four S-boxes at

a time, the combination of the output from the S-boxes can be readily approximated while avoiding the proliferation of additional bits at the input side of the approximation. Thus it is likely that better linear approximations will involve multiple S-boxes. Further, approximations simultaneously involving several S-boxes allow for much larger biases than we would expect from the approximation of single S-boxes individually (such as might be the case if we were using lighter Hamming weight linear approximations). This runs counter to typical techniques when applying linear cryptanalysis in other environments. Often we aim to minimize the number of S-boxes involved in an approximation and hence to reduce the Hamming weight of the approximations involved.

7.1 Linear cryptanalysis of simplified round functions

In this section we will consider the round function in the simplified variant UNI-A-REP-MULT shown in Figure 2. This was used by the designers to estimate the resistance of CIPHERUNICORN-A to linear cryptanalysis.

The designers identify a linear approximation for UNI-A-REP-MULT holding with an estimated correlation coefficient of $\mathrm{LP} = 2^{-22.47}$. However, using exactly the same methodology we can identify a linear approximation holding with correlation coefficient $\mathrm{LP} = 2^{-21.68}$. This improved linear approximation is illustrated in Figure 4. It can be seen that it is a trivially straightforward linear approximation. It is unknown whether or not a specific linear characteristic can be identified that follows this outlined trail. The reader is referred to the self evaluation report [18] for more details of the notation used, but by following the example given there we might estimate that

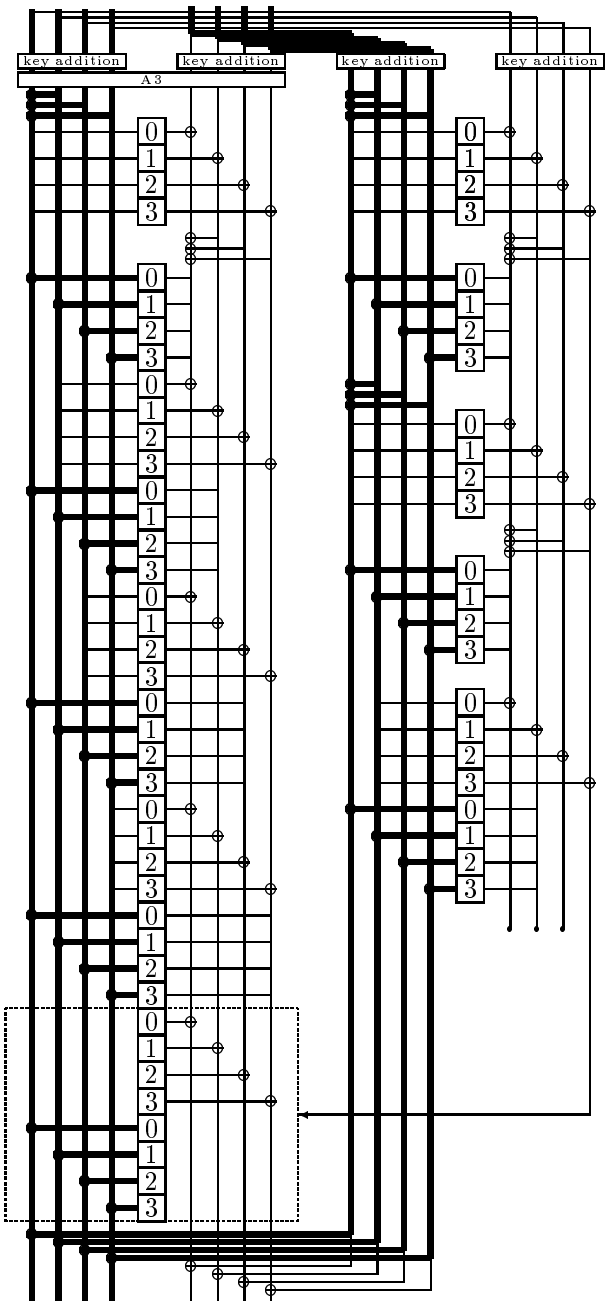
$$\begin{aligned} \mathrm{LP} &= \{\text{input mask } = 0 \text{ for } (S_0||S_1||S_2||S_3)\}^8 \\ &\approx (2^{-2.71})^8 = 2^{-21.68} \end{aligned}$$

Thus, using the same techniques that were used in the self-evaluation report [18], we observe that the correlation coefficient for a single round of UNI-A-REP-MULT might be better bounded by $2^{-21.68}$ instead of $2^{-22.47}$.

With regards to other simplified variants of the cipher, it seems that linear approximations with similar paths will likely offer similar results. The presence or absence of either the A3 function or the integer multiplication is essentially irrelevant at this level of detail.

Of course the practical significance of such improvements are very slight. Indeed, work by Chabaud and Vaudenay [4] and Selçuk [20] suggests that the low correlation values for the cipher as a whole implied by these per round figures, are not very useful. The important thing, practically speaking, is that there appears to be little opportunity to use linear cryptanalytic techniques against the cipher.

Figure 4: An improved linear approximation for UNI-A-REP-MULT.



7.2 Unanticipated effects

It is an interesting question whether the biases and correlation coefficients for sub-components of a round of CIPHERUNICORN-A can be reasonably combined using techniques such as the piling-up lemma [13] for biases or the multiplication of correlation coefficients. Only in such a way were the designers able to come up with bounds on the effectiveness of a linear approximation for the whole round function.

In the absence of well-developed analytical techniques, we might turn to direct empirical evidence. Such testing cannot be exhaustive. In fact one almost has to know the effect one is looking for before it can be demonstrated. Certainly, detecting any significant divergence from the expected behavior with such a limited set of experiments would be highly surprising. So while the results of this section might provide no strong evidence of unexpected behavior, it remains unknown whether or not there might be hidden problems in the cipher.

Our experiments are based around the networks given in Figures 5 and 6. For experiments involving the network in Figure 5 we used the 64-bit input mask `0x00eeb12e 0x00000000`. This input mask offers a linear approximation across the ouputs of S-boxes S_1, S_2, S_3 with bias $44/256 \approx 2^{-2.54}$. The input mask to the S-boxes is set to zero. For the network in Figure 6 we use the 64-bit input mask `0x00eeb12e 0x00eeb12e`. The purpose of this network is to allow for additional interactions between Computation I and Computation II.

Other approximations might well have a more direct interaction with either the pseudo-multiplication used in UNI-A-REP-MULT or the real multiplication in CIPHERUNICORN-A. However such approximations seemed likely to be practically unmeasurable. Instead the concern of these experiments was to consider any hint of a hidden interaction between different strands of text leading to an unanticipated increase or decrease in the bias of the linear approximation. We measured the bias of the linear approximation formed with the input masks `0x00eeb12e 0x00000000` for the network in Figure 5 and `0x00eeb12e 0x00eeb12e` for the network in Figure 6 together with the output mask and conditions indicated here.

1. The network in Figure 5 with output mask `0x00eeb12e 0x00000000` at position A and the multiplication M set to 1.
2. The network in Figure 5 with output mask `0x00eeb12e 0x00000000` at position B and the multiplication M set to 1.
3. The network in Figure 5 with output mask `0x00eeb12e 0x00000000` at position B with the multiplication M replaced by the pseudo-multiplication used in UNI-A-REP-MULT.
4. The network in Figure 5 with output mask `0x00eeb12e 0x00000000` at position B with the multiplication M replaced by the multiplication used in CIPHERUNICORN-A.

Figure 5: Network used to assess linear approximations in CIPHERUNICORN-A.

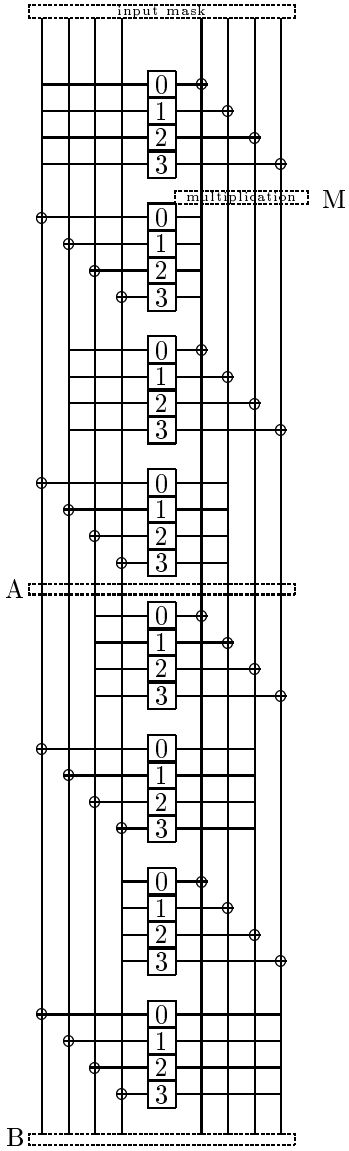
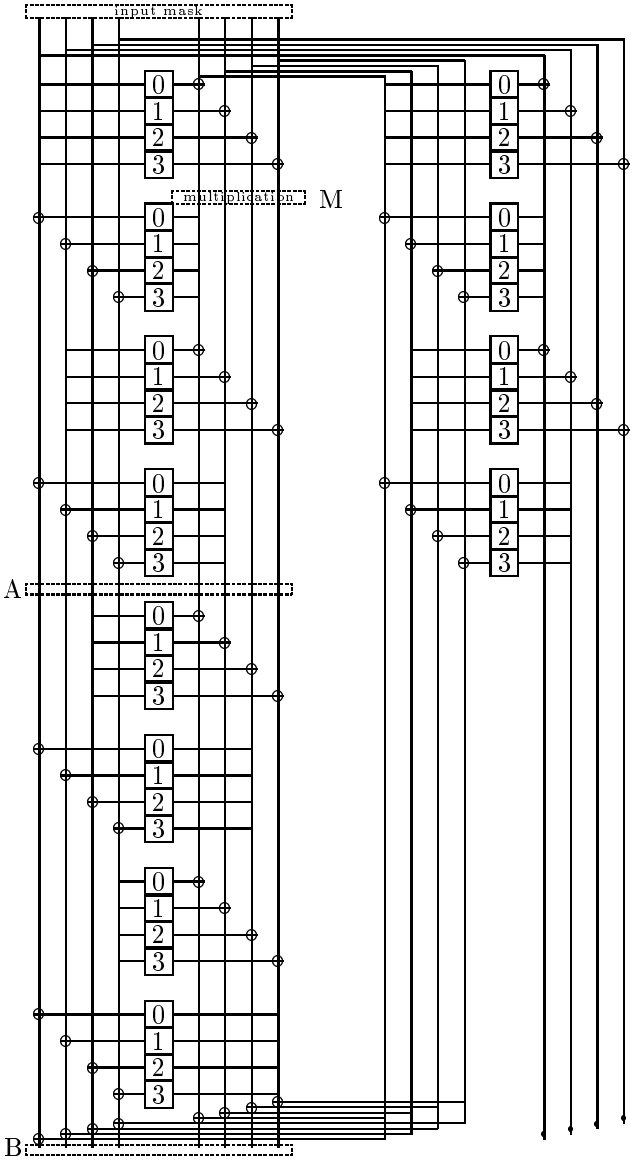


Figure 6: Network used to assess linear approximations in CIPHERUNICORN-A.



5. The network in Figure 6 with output mask `0x00eeb12e 0x00000000` at position B and the multiplication M set to 1.

6. The network in Figure 6 with output mask `0x00eeb12e 0x00000000` at position B with multiplication M replaced by the pseudo-multiplication used in UNI-A-REP-MULT.

7. The network in Figure 6 with output mask `0x00eeb12e 0x00000000` at position B with the multiplication M replaced by the multiplication used in CIPHERUNICORN-A.

For each experiment, we computed the expected bias using the so-called piling-up lemma [13]. We also measured the bias over a large randomly chosen set of inputs. The results are given in the following table.

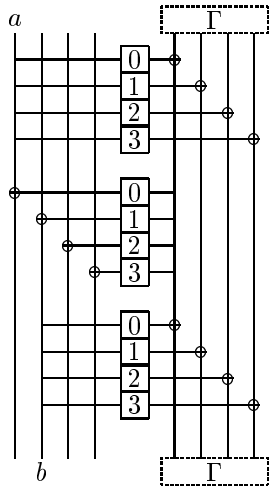
Experiment	*Expected bias*	*Measured bias*	*Number of texts*
1	$2^{-4.08}$	$2^{-4.06}$	2^{20}
2	$2^{-7.16}$	$2^{-7.15}$	2^{20}
3	$2^{-7.16}$	$2^{-7.15}$	2^{20}
4	$2^{-7.16}$	$2^{-7.04}$	2^{20}
5	$2^{-10.24}$	$2^{-10.13}$	2^{24}
6	$2^{-10.24}$	$2^{-9.84}$	2^{24}
7	$2^{-10.24}$	$2^{-10.04}$	2^{24}

While the biases might be very slightly larger than expected there is little here to cause comment. Indeed, it would be a surprise if such a simple network could be identified that would clearly demonstrate any unusual behavior. Other approximations and more extensive testing might provide different results and lead to different conclusions.

We are in the unfortunate position that using the piling-up lemma to combine biases or multiplying linear correlation coefficients is, despite its known and acknowledged faults, the only tool available to the cryptanalyst. However, from the limited work carried out here, there is little evidence to suggest that using these techniques would lead to particularly misleading results.

7.3 A cautionary observation

In the independent review [1] of CIPHERUNICORN-A it was observed that the same input to two different sets of S-boxes would yield the same outputs which would cancel out. This was used to derive an observation about fixed points within the round function for CIPHERUNICORN-A [1]. Earlier in this report (see Section 6.2) we used a related observation differentially. Here we might consider using the observation as part of a linear cryptanalytic attack. Consider the network illustrated here.



Suppose that $a = b$. Then the inputs to the first and third S-box combinations must be the same, as must the outputs. When this happens, the outputs cancel out and it appears that any linear approximation across the three mini-rounds defined by any input and output mask Γ must be perfect. This is troublesome since there seems to be nothing to prevent us setting, say, $\Gamma =$ `0x00000001`. This would therefore imply a linear approximation across two independent S-boxes with a cumulative bias of 2^{-9}.

The mistake is that we overlook compensating biases that result from the cases when the inputs a and b are not the same. Consider the case of $\Gamma =$ `0x00000001`. If $a \neq b$ the probability of the approximation holding is not $1/2$ it is $127/255$ and so instead of a probability for the linear approximation of

$$\frac{1}{256} + \frac{255}{256} \times \frac{1}{2} = \frac{1}{2} + \frac{1}{512}$$

we would have a probability of

$$\frac{1}{256} + \frac{255}{256} \times \frac{127}{255} = \frac{1}{2}.$$

An interesting, yet uneventful, observation.

7.4 Implications for the full cipher

Estimates given in the self-evaluation report [18] for the linear cryptanalysis of CIPHERUNICORN-A are very crude. But they should also be conservative since they ignore the full effects of the integer multiplication and the function A3.

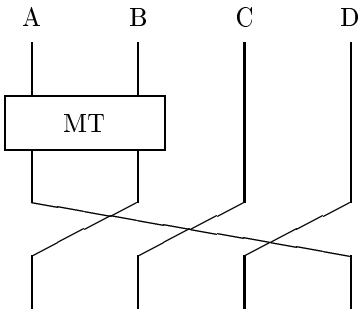
While we have made some improvements to the estimates given in that self-evaluation report, this marginal improvement is unlikely to be of any practical

significance. For CIPHERUNICORN-A it seems unlikely that an active linear approximation for a single round with a practically significant bias could be readily identified for even a small proportion of the keyspace.

Provided some care is taken, limited experimentation has found no particular reason to question the typical methodology of multiplying the correlation coefficients in an assessment of the effectiveness of linear cryptanalysis. It seems that integer multiplication and the use of the function A3 are likely to have a tangible effect in hindering the development of a practical linear cryptanalytic attack. Thus, with the current state of knowledge, it would be reasonable to view CIPHERUNICORN-A as being practically resistant to linear cryptanalysis.

8 Key Schedule Analysis

In Section 4.4 we introduced the key schedule function for CIPHERUNICORN-A. It is based around the recursive use of a simple network that is dependent on the length of key. Here we have illustrated the network for a key length of 128 bits.



We have not looked at the key schedule in any great detail, but we can make the following observation.

Suppose that $\mathrm{MT}(a, b) = (b, a)$. If we were to start the computation with the value (a, b, b, b) then at each stage of the key schedule calculation the four words would always have the value (a, b, b, b). Since key material is always extracted from the left-most strand, every 32-bit word of key material would then have the value a. To see whether there are some a and b such that $\mathrm{MT}(a, b) = (b, a)$ we need to find solutions to

$$\begin{aligned} b &= a \times \texttt{0x01010101}, \\ t &= (b \gg 24) \wedge \texttt{0x000000ff}, \text{ and} \\ a &= b \oplus (S_0[t]||S_1[t]||S_2[t]||S_3[t]). \end{aligned}$$

There is a solution with $a = \texttt{0x61db99c8}$ and $b = \texttt{0x9f3d61c8}$. This means that there are equivalent keys for CIPHERUNICORN-A. The 128-bit key K128,

the 192-bit key K192, and the 256-bit key K256 all provide exactly the same set of subkeys and hence provide exactly the same encryption transformation on 128-bit blocks. These keys are given here in hexadecimal notation.

$$
\begin{aligned}
K128 &= \texttt{0x61db99c89f3d61c89f3d61c89f3d61c8} \\
K192 &= \texttt{0x61db99c89f3d61c89f3d61c89f3d61c89f3d61c89f3d61c8} \\
K256 &= \texttt{0x61db99c89f3d61c89f3d61c89f3d61c89f3d61c89f3d61c8} \\
&\quad\ \texttt{9f3d61c89f3d61c8}
\end{aligned}
$$

While it is not immediately clear what impact this property has in practice, it might be viewed as a certificational weakness.

9 Conclusions

This report describes a brief cryptographic review of CIPHERUNICORN-A. While a broad range of cryptanalytic attacks were considered during this work, our attention was particularly focused on differential and linear cryptanalysis.

In general, the techniques of the designers to quantify the effectiveness of differential and linear cryptanalysis appear to be reasonable. There remain some open questions with regards to this analysis and it is unknown how closely it compares to actuality. While slight improvements to bounds on the probability of a linear approximation for CIPHERUNICORN-A can be made, with our current understanding, CIPHERUNICORN-A should be viewed as being practically resistant to both differential and linear cryptanalysis. There is also a certificational weakness in the key schedule of CIPHERUNICORN-A. Nevertheless, current analysis appears to confirm the continued practical resistance of CIPHERUNICORN-A to advanced cryptanalytic attack.

This review took place over a limited time and with limited resources. It should be anticipated that additional analysis may well yield improved results in the cryptanalysis of this cipher and provide a greater understanding of the true security offered.

References

[1] Anonymous. *Analysis of Cipherunicorn-A.* January 12, 2001.

[2] E. Biham and A. Shamir. *Differential Cryptanalysis of the Data Encryption Standard.* Springer-Verlag, New York, 1993.

[3] C. Burwick, D. Coppersmith, E. D'Avignon, R. Gennaro, S. Halevi, C. Jutla, S. Matyas, L. O'Conner, M. Peyravian, D. Safford, and N. Zunic. *MARS - a candidate cipher for AES.* IBM Corporation. June 10, 1998.

[4] F. Chabaud and S. Vaudenay. Links between differential and linear cryptanalysis. In A.D. Santis, editor, *Advances in Cryptology — Eurocrypt '94*, volume 950 of *Lecture Notes in Computer Science*, pages 356–365, 1994. Springer Verlag.

[5] J. Daemen and V. Rijmen. *AES Proposal: Rijndael.* June 11, 1998.

[6] T. Jakobsen and L.R. Knudsen. The interpolation attacks on block ciphers. In E. Biham, editor, *Fast Software Encryption*, volume 1267 of *Lecture Notes in Computer Science*, pages 28–40, 1997. Springer Verlag.

[7] B.S. Kaliski and M.J.B. Robshaw. Linear cryptanalysis using multiple approximations. In Y.G. Desmedt, editor, *Advances in Cryptology — Crypto '94*, volume 839 of *Lecture Notes in Computer Science*, pages 26–39, New York, 1994. Springer Verlag.

[8] L.R. Knudsen. Applications of higher order differentials and partial differentials. In B. Preneel, editor, *Fast Software Encryption*, volume 1008 of *Lecture Notes in Computer Science*, pages 196–211, 1995. Springer Verlag.

[9] L.R. Knudsen and T. Berson. Truncated differentials of SAFER. In D. Gollmann, editor, *Fast Software Encryption*, volume 1039 of *Lecture Notes in Computer Science*, pages 15–25, 1996. Springer Verlag.

[10] L.R. Knudsen and M.J.B. Robshaw. Non-linear approximations in linear cryptanalysis. In U. Maurer, editor, *Advances in Cryptology — Eurocrypt '96*, volume 1070 of *Lecture Notes in Computer Science*, pages 224-236, 1996. Springer Verlag.

[11] X. Lai, J.L. Massey, and S. Murphy. Markov ciphers and differential cryptanalysis. In D.W. Davies, editor, *Advances in Cryptology — Eurocrypt '91*, volume 547 of *Lecture Notes in Computer Science*, pages 17–38, Berlin, 1992. Springer-Verlag.

[12] S.K. Langford and M.E. Hellman. Differential-linear cryptanalysis. In Y.G. Desmedt, editor, *Advances in Cryptology — Crypto '94*, volume 839 of *Lecture Notes in Computer Science*, pages 17–25, 1994. Springer Verlag.

[13] M. Matsui. Linear cryptanalysis method for DES cipher. In T. Helleseth, editor, *Advances in Cryptology — Eurocrypt '93*, volume 765 of *Lecture Notes in Computer Science*, pages 386–397, 1994. Springer-Verlag.

[14] M. Matsui. The first experimental cryptanalysis of the Data Encryption Standard. In Y.G. Desmedt, editor, *Advances in Cryptology — Crypto '94*, volume 839 of *Lecture Notes in Computer Science*, pages 1–11, New York, 1994. Springer-Verlag.

[15] National Institute of Standards and Technology (NIST). *FIPS Publication 46-2: Data Encryption Standard*, December 30, 1993.

[16] National Institute of Standards and Technology (NIST). *FIPS Publication 197: Advanced Encryption Standard*, November 26, 2001.

[17] NEC Corporation. *Cryptographic techniques specifications: Cipherunicorn-A*, Version 2. Undated.

[18] NEC Corporation. *Self Evaluation Report: Cipherunicorn-A*, Version 3. Undated.

[19] K. Nyberg. Linear approximation of block ciphers. In A.D. Santis, editor, *Advances in Cryptology — Eurocrypt '94*, volume 950 of *Lecture Notes in Computer Science*, pages 439–444, 1994. Springer-Verlag.

[20] A. Selçuk. On bias estimation in linear cryptanalysis. In *Proceedings of Indocrypt 2000*, volume 1977 of *Lecture Notes in Computer Science*, pages 52–66, 2000. Springer-Verlag.