Cryptanalysis of Hummingbird-1

Markku-Juhani O. Saarinen
mjos@reveresecurity.com

16 February 2011

Fast Software Encryption 2011

Hummingbird-1

Hummingbird-1 is an encryption and message authentication primitive that has a 256-bit secret key, uses a 64-bit nonce and optionally produces a 64-bit authenticator for the message.

The algorithm is intended for use in extremely resource-constrained devices. The algorithm has been patented and extensively cryptanalyzed prior to publication by CACR and ISSI.

Hummingbird is similar to ciphers such as Helix and Phelix in that it is a word-based stream cipher that can also be used for authentication.

Publication info:

D. ENGELS, X. FAN, G. GONG, H. HU AND E. M. SMITH. "Ultra-Lightweight Cryptography for Low-Cost RFID Tags: Hummingbird Algorithm and Protocol." Centre for Applied Cryptographic Research (CACR) Technical Reports, CACR-2009-29.

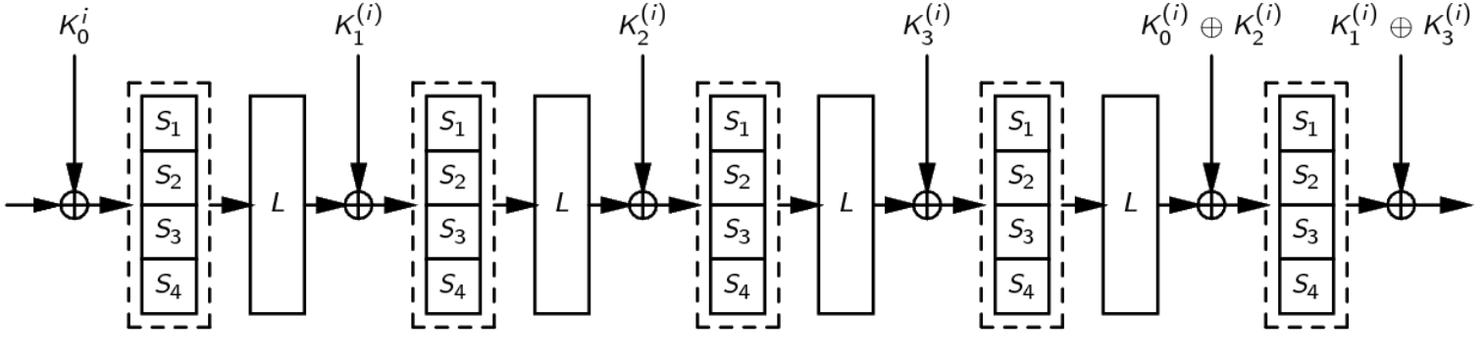
X. FAN, H. HU, G. GONG, E. M. SMITH AND D. ENGELS. "Lightweight Implementation of Hummingbird Cryptographic Algorithm on 4-Bit Microcontroller." The 1st International Workshop on RFID Security and Cryptography 2009 (RISC'09), pp. 838 – 844, 2009.

D. ENGELS, X. FAN, G. GONG, H. HU AND E. M. SMITH. "Hummingbird: Ultra-Lightweight Cryptography for Resource-Constrained Devices." 1st International Workshop on Lightweight Cryptography for Resource-Constrained Devices (WLC'2010). Tenerife, Canary Islands, Spain, January 2010

Building blocks

- Hummingbird-1 has a 64 + 16 - bit state consisting of four 16-bit registers $R1$, $R2$, $R3$, $R4$ and a 16-bit LFSR L.
- The cipher is initialized by setting the 64-bit nonce in the registers and running an initialization function for four rounds.
- Each round updates the four registers and the LFSR and processes one 16-bit word of plaintext into ciphertext.
- Nonlinearity is derived the “E Box” and from mixing the XOR operation and modular addition.

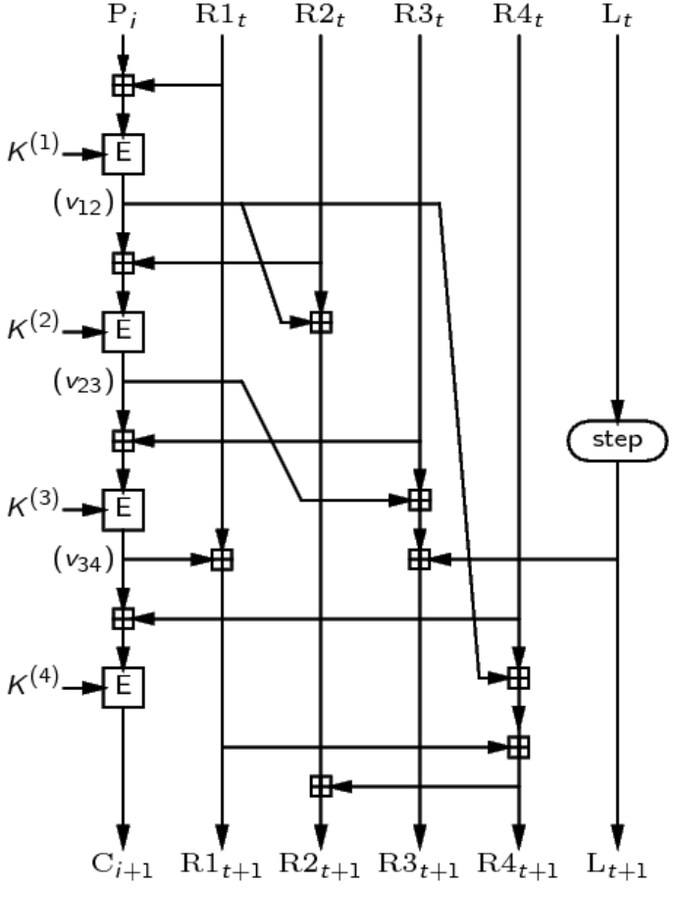
The E Box



- The cipher has a 16-bit “E-Box” that utilizes a 64-bit subkey. The design of the E-Box is irrelevant to the attack presented here (as long as it does not use more than 64 bits of keying material).

- The E-Box is built from five invocations of 4x4 S-Boxes and a linear mixing function L.

Hummingbird-1 Round



The Key

The 256-bit secret key K is split into four 64-bit subkeys $K^{(1)}$, $K^{(2)}$, $K^{(3)}$ and $K^{(4)}$ without any mixing.

We index each one of the 64-bit subkeys as 16-bit words $K_j^{(i)}$ as follows:

$$
\begin{aligned}
K &= (K^{(1)}, K^{(2)}, K^{(3)}, K^{(4)}) \\
K^{(1)} &= (K_1^{(1)}, K_2^{(1)}, K_3^{(1)}, K_4^{(1)}) \\
K^{(2)} &= (K_1^{(2)}, K_2^{(2)}, K_3^{(2)}, K_4^{(2)}) \\
K^{(3)} &= (K_1^{(3)}, K_2^{(3)}, K_3^{(3)}, K_4^{(3)}) \\
K^{(4)} &= (K_1^{(4)}, K_2^{(4)}, K_3^{(4)}, K_4^{(4)}).
\end{aligned}
$$

Attack outline

We will describe the following attack (which can be improved!):

- A chosen plaintext and ciphertext attack that requires about 2^{20} queries using two distinct IVs.
- The attack is made possible by a flaw in the initialization function.
- Uses high-bit additional differentials only, the structure of the E box is not relevant.
- Uses a divide-and-conquer strategy to attack each 64-bit subkey individually. The attack complexity is therefore bound by 2^{66} but can be improved by differential attacks on E.

Flaw in the IV setup

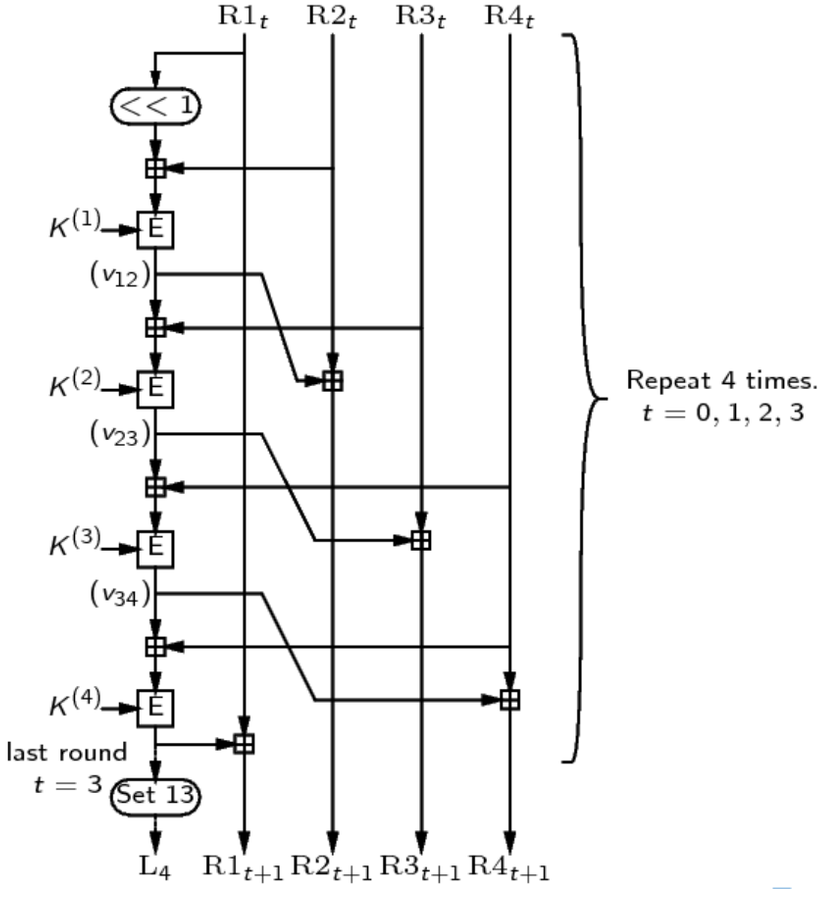
Observation 1. *The Hummingbird-1 initialization function has a high-bit XOR differential that holds with probability 1:*

$$\Delta(\mathrm{IV}_1, \mathrm{IV}_2, \mathrm{IV}_3, \mathrm{IV}_4) = (8000, 0000, 0000, 0000)$$

$$\Downarrow$$

$$\Delta(\mathrm{RS1}_0, \mathrm{RS2}_0, \mathrm{RS3}_0, \mathrm{RS4}_0, \mathrm{LFSR}_0) = (8000, 0000, 0000, 0000, 0000).$$

Hummingbird-1 Initialization



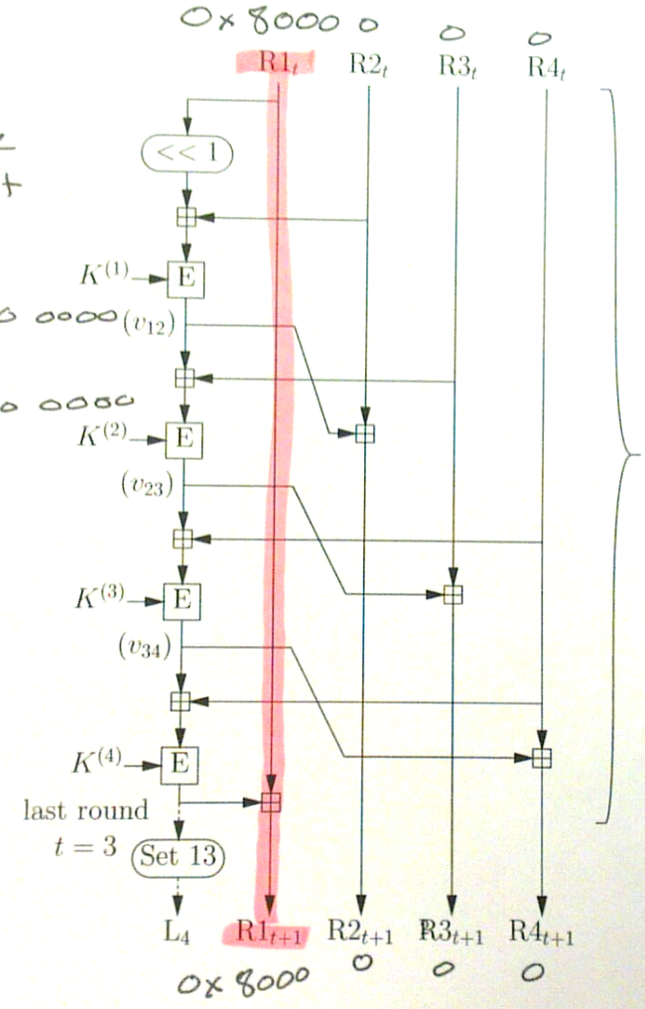
Attacking IWIT

We only use two distinct IVs:

0000 0000 0000 0000 0000 0000 0000 0000

and

8000 0000 0000 0000



$\Delta = 0x8000$

It is easy to see that the high bit differential passes through 4 rounds of Init with $P = 1$

First Round

Observation 2. *There is a Chosen-IV distinguisher for Hummingbird that works with probability* $P = 65535/65536$ *and has data complexity of 1 word. One can use the high-bit differential of Observation 1 and the following differential for the first round:*

$$\Delta(\mathrm{P}_0, \mathrm{RS1}_0, \mathrm{RS2}_0, \mathrm{RS3}_0, \mathrm{RS4}_0, \mathrm{LFSR}_0) = (8000, 8000, 0000, 0000, 0000, 0000)$$

$$\Updownarrow$$

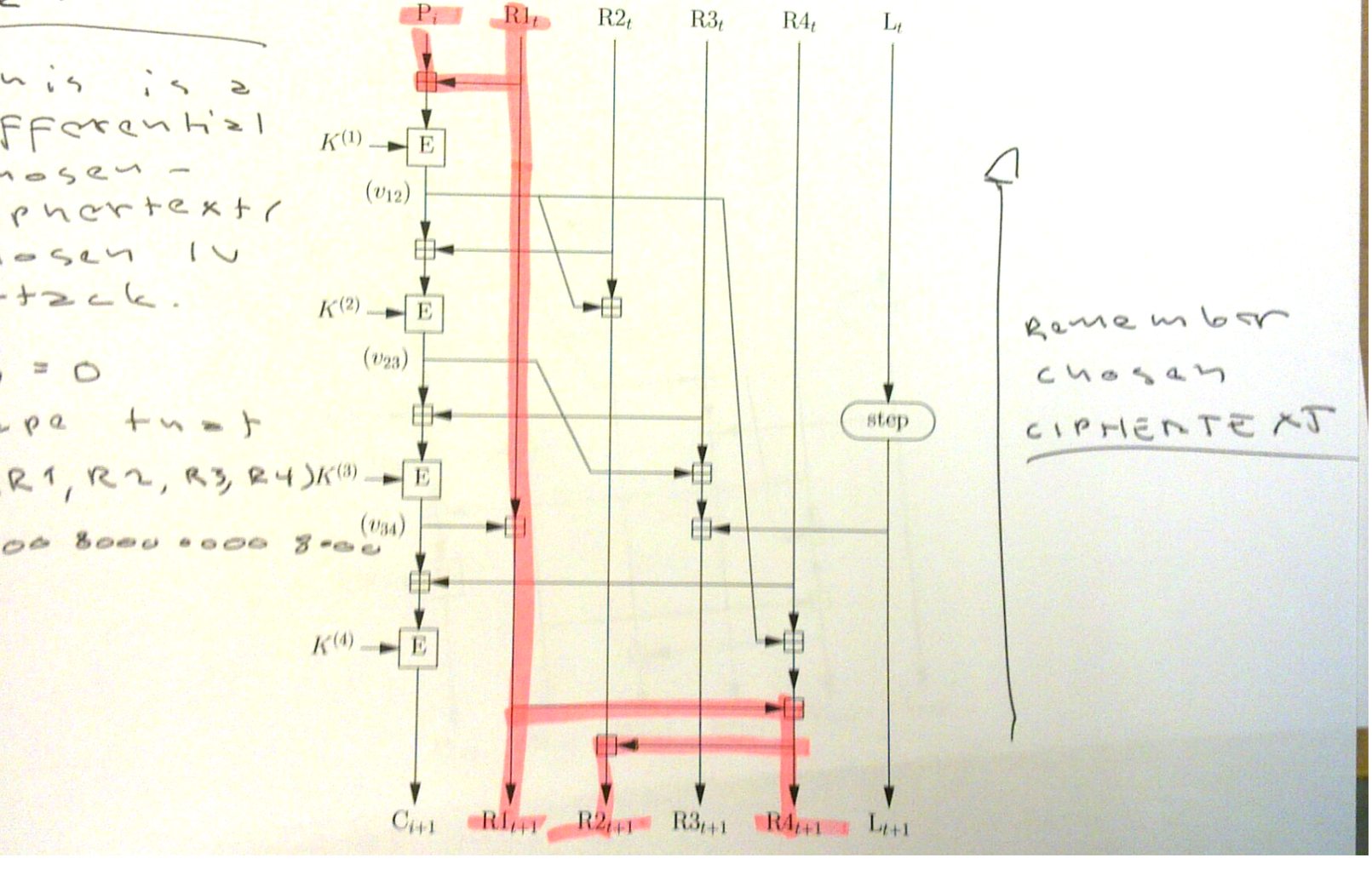
$$\Delta(\mathrm{C}_0, \mathrm{RS1}_1, \mathrm{RS2}_1, \mathrm{RS3}_1, \mathrm{RS4}_1, \mathrm{LFSR}_1) = (0000, 8000, 8000, 0000, 8000, 0000)$$

Attacking $k^{(1)}$

This is a differential chosen-ciphertext/ chosen IV attack.

$L_1 = 0$

Hope that $\Delta(R1, R2, R3, R4)K^{(3)}$ = 8000 8000 0000 8000



Remember chosen CIPHERTEXT

An Iterated Differential

Observation 3. *There is a one-round iterated differential that works if a collision occurs inside the cipher as follows:*

$$\Delta v_{12_t} = 8000\ ,\ \Delta v_{23_t} = 0000\ ,\ \Delta v_{34_t} = 0000$$

$$\Delta(\mathrm{RS1_t}, \mathrm{RS2_t}, \mathrm{RS3_t}, \mathrm{RS4_t}, \mathrm{LFSR_t}) = (8000, 8000, 0000, 8000, 0000)$$

$$\Updownarrow$$

$$\Delta(\mathrm{RS1_{t+1}}, \mathrm{RS2_{t+1}}, \mathrm{RS3_{t+1}}, \mathrm{RS4_{t+1}}, \mathrm{LFSR_{t+1}}) = (8000, 8000, 0000, 8000, 0000).$$

The initial condition for $t = 5$ can be satisfied using the initialization and first-round encryption differentials given in Observations 1 and 2.

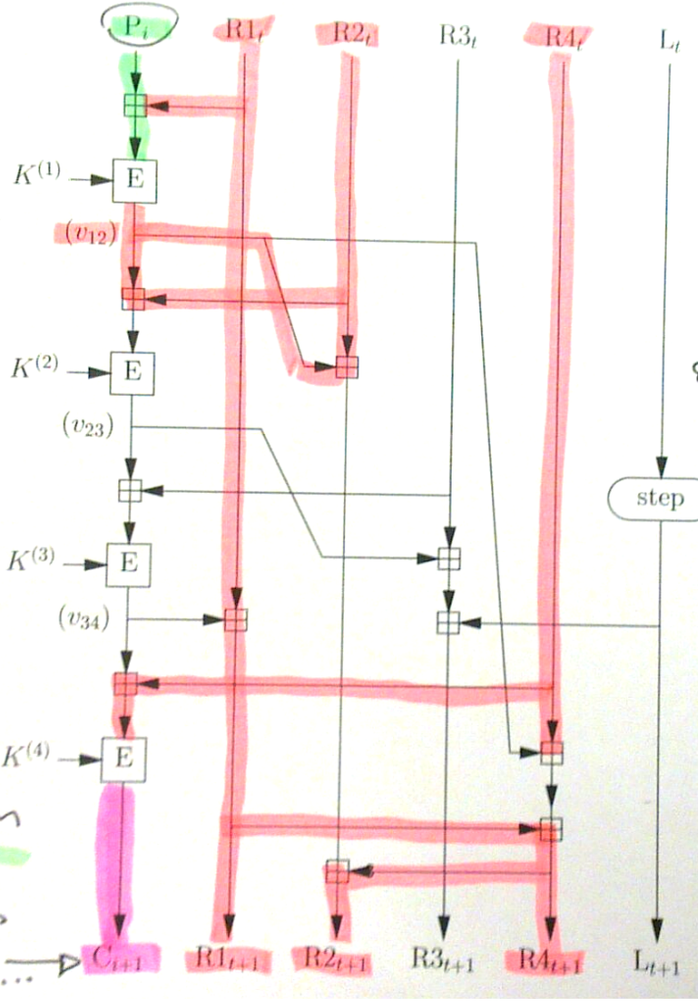
Attack on K1

- Work on two IVs, `0000 0000 0000 0000` and `8000 0000 0000 0000`.
- Try to find a pair of ciphertexts `0000 aaaa aaaa ..` and `0000 bbbb bbbb ..` so that the range of the absolute difference of plaintext words is around $2^{15}(1 - \frac{1}{e}) \approx 20713.3$ rather than the random $2^{15} = 32768$.
- When such a “right pair” is found, we may do a search on the first 64-bit subkey by eliminating impossible keys.
- Note that we don’t care about various weaknesses of the E box. This step may be sped up significantly.

Attacking $k^{(1)}$

A chosen ciphertext attack

If right pair 2/6 is found $\Delta V_{12} = 8000$ and range of $|\Delta P_i|$ is $2^{15}(1-\frac{1}{e}) \approx 20713$ rather than $2^{15} = 32768$



chosen 2/6 values for C_2...

$\Delta = 8000$

The first stage is to find ciphertext vectors

0 a a a a a a

and

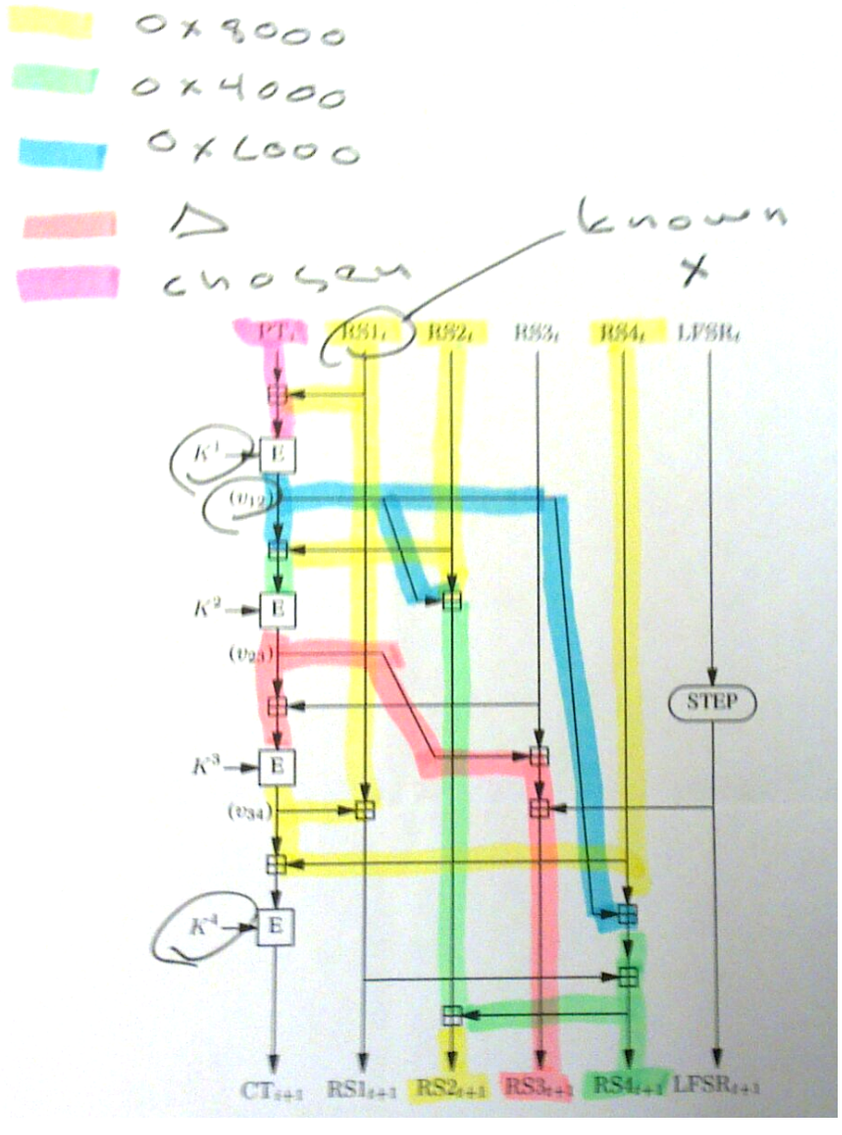
0 6 6 6 6 6 6

such that $\Delta V_{12} = 8000$

Decrypt vectors of size 4k words and look at range of PT differentials.

Attack on K2-K4 (abridged.. details in the paper)

- Attack proceeds by attacking K4, then K3 and finally K2.
- These attacks use a bit more complicated math to discard impossible subkeys.
- A four-round differential is used. Each sub-attack requires knowledge previously gathered key bits.
- The additive differentials use 2 highest bits (bit 14 and 15).
- The data complexity is smaller than in the first step.


0x8000
0x4000
0x1000
Δ
chosen
known
x
PT$_t$
RS1$_t$
RS2$_t$
RS3$_t$
RS4$_t$
LFSR$_t$
K^1
E
(v_{12})
K^2
E
(v_{23})
STEP
K^3
E
(v_{34})
K^4
E
CT$_{t+1}$
RS1$_{t+1}$
RS2$_{t+1}$
RS3$_{t+1}$
RS4$_{t+1}$
LFSR$_{t+1}$

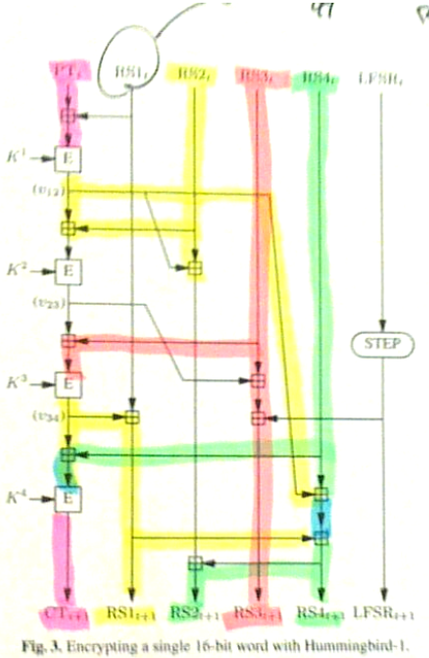


Fig. 3. Encrypting a single 16-bit word with Hummingbird-1.

Round 3

$v34_2 = RS1_3 \boxplus RS1_2$

$RS4_2 = D_{EK3}(CT_2) \boxplus v34_2$

$RS4_3 = RS4_2 \boxplus E_{EK1}(PT_2 + RS1_2) \boxplus RS1_3$

$v34_3 = D_{EK3}(CT_3) \boxplus RS4_3$

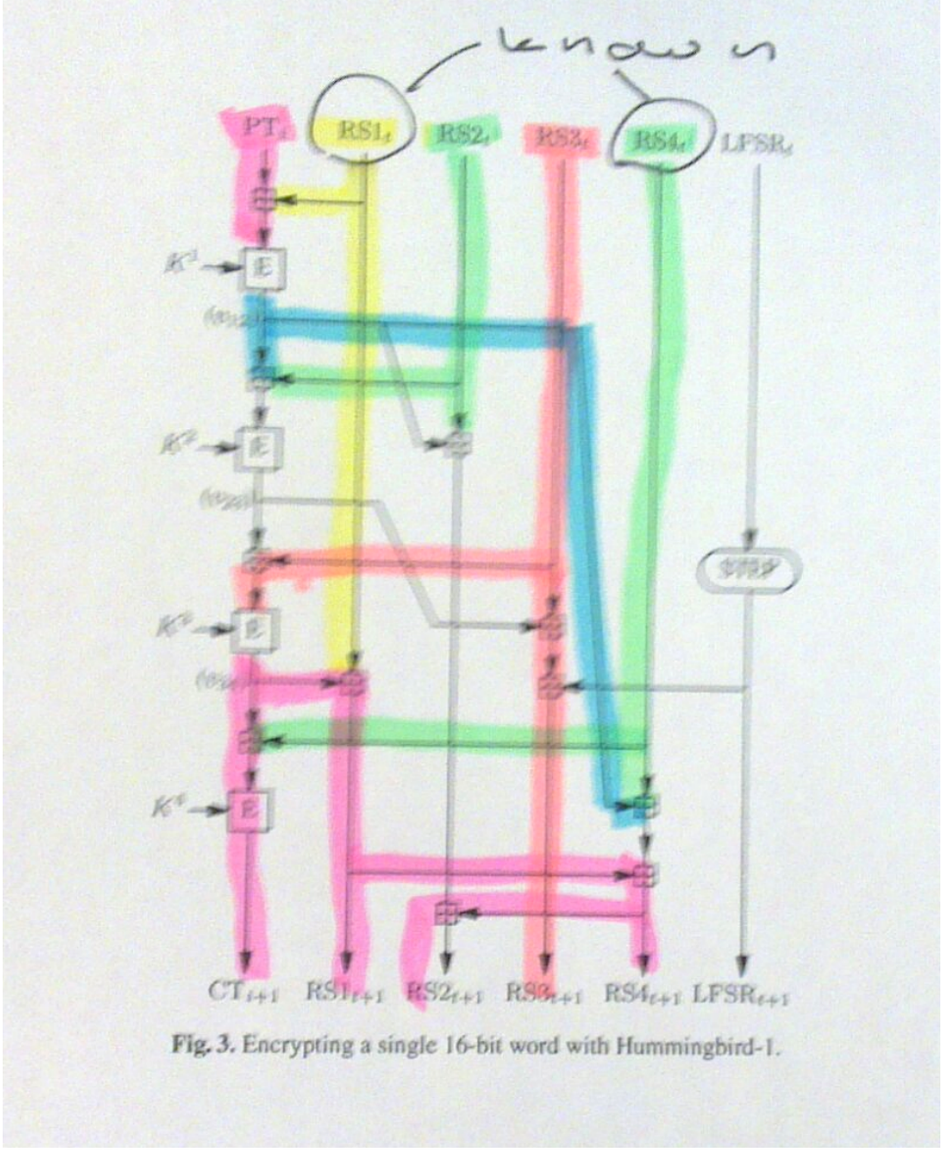


Fig. 3. Encrypting a single 16-bit word with Hummingbird-1.

Demo attacking a 4 * 24 = 96 bit key

Source code is available: **http://www.mjos.fi/dist/hb1an.tgz**

```
~/hb1an$ ./hb1an
rand seed = 1297763753
self test — passed.
tru_key[] = 0000000000EA178D0000000000AAB48A00000000009387CD0000000000676B51

hb1_break() started on Tue Feb 15 11:55:53 2011
decrypting  ..........................................................
right pair  ..........................................................
paired a/b  .. 00D1 / 0138 .. c = 20757
EK1 search  ............................ 0000000000EA178D
tabulating  923D D79C D6D3 A86D 9D60 09B0 7FF6 DAD2 07C8 34E6 BB2D 407B 91CD
EK4 search  ........... 0000000000676B51
tabulating  .. max slot = 8 .. quartets = 32
EK3 search  ................. 00000000009387CD (d = 6)
EK2 search  ................... 0000000000AAB48A
hb1_break() finished on Tue Feb 15 11:56:20 2011
running time: 27 wall—clock seconds

crk_key[] = 0000000000EA178D0000000000AAB48A00000000009387CD0000000000676B51
~/hb1an$
```

Hummingbird-2

- The key size has been set to 128 bits to be commensurable with the actual security of the cipher.
- The state size of the cipher has been increased from 80 bits to 128 bits and the LFSR has been eliminated.
- The keyed “E Box” now only has four invocations of the S-Boxes, compared to five in Hummingbird-1. This increases the encryption speed of the cipher.
- The authentication mechanism has been improved due to thwart a message extension attack (unpublished but trivial).

Conclusions

- We describe a very effective attack found that will break full Hummingbird-1 in reasonable time.
- The attack code is about 500 lines without the actual Hummingbird-1 implementation.
- The presented attack depends on a flaw in the key setup procedure, but can be adopted to slight modifications in the cipher structure (this became apparent during the design of Hummingbird-2).
- Colored highlighting pens can be very useful in cryptanalysis!