>CypherMatrix< Encryptions

The **CypherMatrix** procedure performs encryptions with the combination of following operations:

XOR concatenation	**XOR**	D, 4, b
substitution dyn24	**dyn24**	D, 4, c
bit conversion (8 bit → 7 bit)	**BC(8/7)**	D, 4, d (2)
bit conversion (8 bit → 9 bit)	**BC(8/9)**	D, 4, d (4)
bit conversion (8 bit → 6 bit)	**coding base 64**	D, 4, d (1)
structure changing	**SC**	D, 4, d (3)
number system base 4	**BC4**	D, 4, d (5)
regression Reg	**Reg**	D, 4, d (6)
bit exchange	**BE**	D, 4, e (1,2)
extended bit series	**SBE**	D, 4, e (3)
bit crossing	**BCR**	D, 4, f
number system base 256	**B256**	D, 4, g
byte transposition	**BT**	D, 4, h
multi-time-pad	**MtP**	D, 4, j

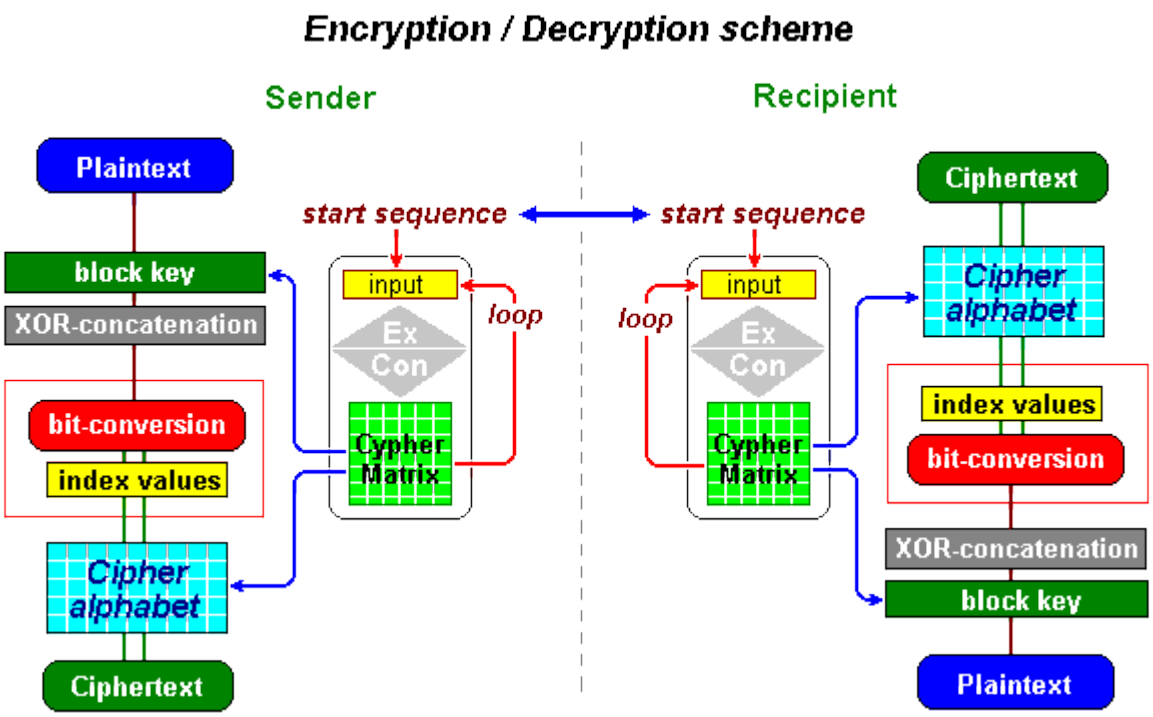
Right column points to the corresponding text passage in WEB-article:

CORECYPH.HTM Performance of digital Encryption

The encryption as a symetric procedure is performed by the aforesaid operations separate or in combinations.
A >start sequence< of optimal 42 bytes controls the entire process at sender and at recipient as well. For each program the following parameters are needed:

Length of >matrix key< :	36 – 64 bytes
Length of block key :	35 – 96 bytes
Number system for expansion function:	35 – 96 basic digits
individal user code:	1 – 99 number

Following scheme illustrates the functional connections:



Further details you will find in internet at:

http://www.telecypher.net/CORECYPH.HTM

Following cypher programs are realized by the obove mentioned operations:

main programs

CYPHER	**XOR – dyn24 - BC(8/7)**
CYPHERXT	**XOR - BC(8/7)**

single operations

CYPHER11	**XOR**
CYPHER12	**dyn24**
CYPHER13	**BC(8/7)**
CYPHER14	**BE**
CYPHER15	**CR**
CYPHER16	**BC4**
CYPHER17	**SC**
CYPHER18	**BCR**
CYPHER 51	**BT**
CYPHER55	**SBE**
CYPHER64	**coding base 64**

CYPHER89	**BC(8/9)**
CYPHER97	**B256**
CYPHER98	**B256**
CYPHER99	**B256**

double combinations

CYPHER19	**BC(8/7) – Reg**
CYPHER20	**dyn24 – BE**
CYPHER21	**XOR - dyn24**
CYPHER22	**XOR - BC(8/7)**
CYPHER23	**dyn24 - XOR**
CYPHER24	**dyn24 - BC(8/7)**
CYPHER25	**BC(8/7) - XOR**
CYPHER26	**BC(8/7) - dyn24**
CYPHER27	**BE - XOR**
CYPHER28	**BE - BC(8/7)**
CYPHER29	**XOR - BCR**
CYPHER44	**dyn24 - BC(8/7)**
CYPHER52	**BT - dyn24**
CYPHER56	**SBE - BC(8/7)**
CYPHER59	**BE - SBE**
CYPHER61	**XOR - BC4**
CYPHER71	**XOR - SC**
CYPHER81	**SC - BC(8/7)**
CYPHER90	**BC(8/7) - Reg**
CYPHER91	**BC(8/7) - Reg**
CYPHER94	**BC(8/7) - Reg**

threefold combinations

CYPHER31	**XOR – dyn24 - BC(8/7)**
CYPHER32	**XOR – BC(8/7) - dyn24**
CYPHER33	**dyn24 – XOR - BC(8/7)**
CYPHER34	**dyn24 – BC(8/7) - XOR**
CYPHER35	**BC(8/7) – XOR - dyn24**
CYPHER36	**BC(8/7) – dyn24 - XOR**
CYPHER37	**BE – XOR - BC(8/7)**
CYPHER38	**BE – BC(8/7) - XOR**
CYPHER39	**BE – dyn24 - BCR**
CYPHER53	**BT – dyn24 - BC(8/7)**
CYPHER57	**SBE – dyn24 - BC(8/7)**
CYPHER5B	**SBE – BCR - Reg**

CYPHER5C	**BT – dyn24 - XOR**
CYPHER62	**BE – dyn24 - BC4**
CYPHER63	**SBE – dyn24 - BC4**
CYPHER72	**XOR – dyn24 - SC**
CYPHER82	**MtP (SC – dyn24 - XOR)**
CYPHER83	**SC – dyn24 - BC(8/7)**
CYPHER92	**XOR – BC(8/7) - Reg**
CYPHER93	**dyn24 – BC(8/7) - Reg**

fourfold combinations

CYPHER41	**BE – XOR – dyn24 - BC(8/7)**
CYPHER42	**BE – dyn24 – XOR - BC(8/7)**
CYPHER43	**BE – dyn24 – XOR - SC**
CYPHER54	**BT – dyn24 – BC(8/7) - Reg**
CYPHER58	**SBE – XOR – BC(8/7) - Reg**
CYPHER5A	**BE – dyn24 – XOR - SBE**

download versions

CYPHER-SC	**XOR - SC**	D, 4, d (3)
RECYPHER	**XOR – BC(8/7) – Reg**	D, 4, d (6)
BECYPHER	**BE – BC(8/7)**	D, 4, e (2)
SBCYPHER	**SBE – dyn24 – BC(8/7)**	D, 4 e (3)
BTCYPHER	**BT – dyn24 - BC(8/7)**	D, 4, h
MTCYPHER	**MtP**	D, 4, j

Due to **bit conversion** >**BC(8/7)**< and thereby as a result of „congruence of length“ all current attacks on CypherMatrix procedures will have no success and even the only remaining attack „brute force“ will be beyond all hope:
(see: telecypher.net/CORECYPH.HTM / Cryptanalysis).

Attacks on Side Areas: „Side Channel Attacks“

Attacks on side areas of the procedure Article: "Side Channel Attacks" are aligned to win conclusions on individual bits from species-characteristic technical accompaniments - particularly on used keys - at time of the decoding excution. As typically processing related side areas are known:

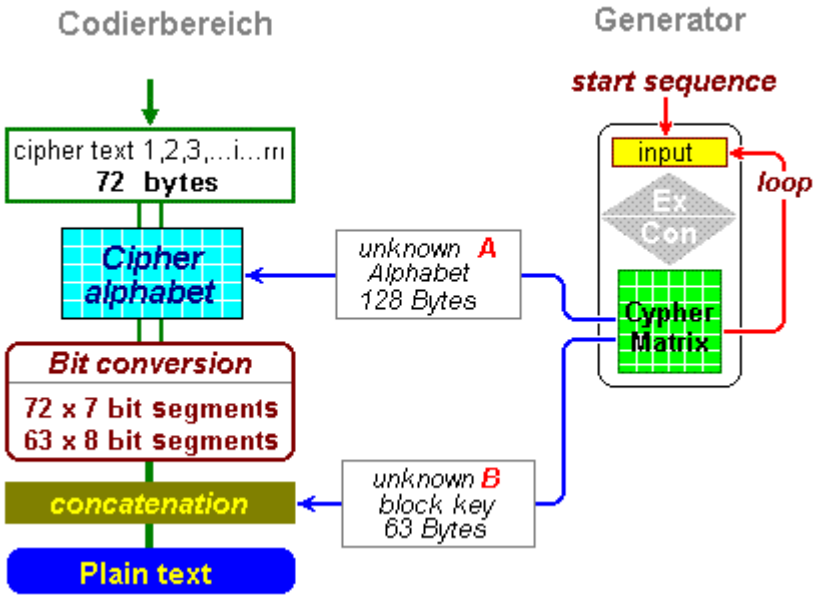
1. Analysing the time it takes for different units to perform operations (timming attack),
2. Power consumption of units while performing calculations

in order to gain conclusions on the used keys (power consumption analysis),
3. Generating faults during operations (differential fault analysis) and
4. elektromagnetic radiations possibly caused while processing operations (TEMPEST).

An attacker only knows the cipher text and the CypherMatrix procedure. The respective aplied program and the single parameters - start sequence included - are not known to him.

Side channel attacks are dedicated especially to find single bits which may lead to further conclusions and knowledges. But the CypherMatrix procedure does almost exclusively work with bytes (Byte Technology). Processing related bits are not in use, at all.

At each cycle decryption process works as follows:



The cipher text - handled in cycles of e.g. 72 signs – contains neither any bit nor bytes of the original plaintext. The signs are sole **pointers** to positions in the cipher alphabet. Moreover, the cipher text does not include any data of the start sequence or other references to the program and its control parameters.
In the generator working area there are used no data or characteristic features of the plaintext. Only the start sequence at the beginning of the procedure, controls the entire process (processing generator).

The unknown quantities **A** and **B** necessary for decoding are occasionally created anew at each cycle. Hence, after a cycle is done no performing Data (excepting cipher text) will be stored neither in cache nor elsewhere which could be a target for attacks on side areas. Last of all we have to state that there will be no chance for "side channel attacks" on CypherMatrix decoding procedures.

If you are interested or for further engagement with the matter you may request single programs per e-mail from the author or download from the marked text passage in the mentioned WEB article.

mailto:eschnoor@multi-matrix.de

Munich, in April 2009

Ernst Erich Schnoor
München
