Tabelle 1

	DESFire TDES decryption in SEND mode																				
	Note on naming: to send encrypted data to the PICC we technically need to decrypt the data																				
start the decryption with data from DES authentication in step 10																					
triple the key for a TDES key (24 bytes)	D0	00	22	44	66	88	AA	CC	D0	00	22	..	66	88	AA	CC		(total 24 bytes)			
The iv for TDES decryption is 8 bytes 0x00																					
Concatenated rndA with left rotated rndB	45	CC	39	92	87	13	E1	C0	A3	46	8F	B0	D8	7E	74	C4					
Note: this value is the ciphertext for decryption																					
01 start with an empty 'cipheredBlock' of 8 bytes length	00	00	00	00	00	00	00	00		(8 bytes = DES block length)											
02 split the ciphertext into blocks of 8 bytes	45	CC	39	92	87	13	E1	C0	A3	46	8F	B0	D8	7E	74	C4					
ciphertextBlock1 (ctBlock1)	45	CC	39	92	87	13	E1	C0													
ciphertextBlock2 (ctBlock2)									A3	46	8F	B0	D8	7E	74	C4					
03 XORing ctBlock1 with cipheredBlock (step 01)	45	CC	39	92	87	13	E1	C0													
note: no surprise, XORing with 0x00 results in an unchanged value																					
04 decrypt ct1Xored (step 03) using TripleDES.decrypt	88	E1	99	B0	2D	A8	33	67													
05 copy ct1XoredDecrypted (step 04) to cipheredBlock	88	E1	99	B0	2D	A8	33	67													

06 XORing ctBlock2 with cipheredBlock (step 05)	2B	A7	16	00	F5	D6	47	A3														
07 decrypt ct2Xored (step 06) using TripleDES.decrypt	55	72	08	D9	62	AE	4B	4F														
08 note: for more data the steps 05 to 07 are replicated																						
09 concatenate decrypted ct1Xored (step 04) and decrypted ct2Xored	88	E1	99	B0	2D	A8	33	67	55	72	08	D9	62	AE	4B	4F						
Result of TDES.decrypt 0 ‚plaintext‘	88	E1	99	B0	2D	A8	33	67	55	72	08	D9	62	AE	4B	4F						
Use the decryption result in DES authentication step 11 as ‚encryption result‘																						