NDS FAQ by Colibri

2003-06-08: Updated Channel ID Tiers table. Bugfix in nano 33.

2003-01-04: I got new info by emails. Thanks! New nanos (27, 34, 37, 3B, 44, 6F, 7A and 90) and **IRD**-EMM (EMM triggered firmware download)

2002-06-09: New Channel ID Tiers table

Important note: All informations in this document are strictly for Educational purpose only. The author is not to be made liable for abuses. No warranty for the correctness of all data and procedures.

Chapter 1: EMM (Entitlement Management Message)

The EMMs can be logged e.g. on Astra 2 (28.2 degrees east) from the BBC transponder 11758 H SR 27.5 MS/s FEC 2/3. The EMM PID is C0 hex (192 dec.).

Each TS packed contains 188 bytes, but the most loggingprograms stores only the last 184 bytes.

These EMMs (with size of 184 bytes) always starts with the byte sequence 00 82 70. The following byte contains the length of the payload. After the payload itself, bytes with the value FF are used to fill the rest of the section.

Since the possible payload of 180 bytes is not always used in an EMM, an EMM is divided in up to four blocks. The first byte of the EMM is the address-type-byte, which describes how many sub-EMMs are contained in that EMM. It describes also the destination of the sub-EMMs:

- Supposed to all cards
- Supposed to a cardgroup
- Supposed to one card.

Address-type-byte:

b8	b7	b6	b5	b4	b3	b2	b1	Remark
0	0	x	x	0	0	0	0	All sub-EMMs contained in this EMM are purposed for all cards. No filterblock is used.
0	1	x	x	0	0	0	0	All sub-EMMs contained in this EMM are purposed for one card. A filterblock with n * 4 bytes follows.
1	0	x	x	0	0	0	0	All sub-EMMs contained in this EMM are purposed for a cardgroup. A filterblock with n * 4 bytes follows.
1	1	x	x	x	x	x	x	Not used

x	x	0	0	0	0	0	0	This EMM contains one sub-EMM. If the b8 or b7 is set then the filterblock is 1 * 4 bytes long.
x	x	0	1	0	0	0	0	This EMM contains two sub-EMMs. If the b8 or b7 is set then the filterblock is 2 * 4 bytes long.
x	x	1	0	0	0	0	0	This EMM contains tree sub-EMMs. If the b8 or b7 is set then the filterblock is 3 * 4 bytes long.
x	x	1	1	0	0	0	0	This EMM contains four sub-EMMs. If the b8 or b7 is set then the filterblock is 4 * 4 bytes long.
x	x	x	x	1	x	x	x	Not used
x	x	x	x	x	1	x	x	Not used
x	x	x	x	x	x	1	x	Not used
x	x	x	x	x	x	x	1	Not used

If filters are used (bit 7 or 8 of the address-type-byte is set) then this filters follows the address-type-byte in groups of 4 bytes each. If the sub-EMM is purposed for one card then the 4 bytes represents the cardaddress. If the sub-EMM is purposed for a group then the first 3 bytes represents the cardgroup and the last byte is a padding byte with the value 1.

The next two bytes can be 00 00 or 00 01 and are independently of the value of the address-type-byte always present.
If these bytes are 00 00 then a Block-length-byte follows.
If these bytes are 00 01 then a Block-length-word follows.

Now the sub-EMM blocks follows. A block starts with the block-length-byte or block-length-word.

Block-length-byte:

b8	b7	b6	b5	b4	b3	b2	b1	Remark
0	0	x						Not used
0	1	x						Not used
1	0	x						Not used
1	1	Length						Shows the length of the following block in WORDs (a WORD are two bytes). If a block contains an odd number of bytes, then one padding byte with the value 0 is added at the end of the block, so the block-length-byte is valid again.

Block-length-word:

b8	b7	b6	b5	b4	b3	b2	b1	b8	b7	b6	b5	b4	b3	b2	b1	Remark
0	0	x														Not used
0	1	x														Not used
1	0	x														Not used
1	1	Length														Shows the length of the following block in WORDs (a WORD are two bytes). If a block contains an odd number of bytes, then one padding byte with the value 0 is added at the end of the block, so the block-length-byte is valid again.

The end of a block is followed by the block-length-byte/word of the next block (only if the address-type-byte shows that more than one sub-EMM is contained (min. bit 5 or 6 is set)).

If filter are used, then the first 4 filterbytes belongs to the first sub-EMM block, the second 4 filterbytes belongs to the second sub-EMM block and so on.

What is the structure of a sub-EMM?

A sub-EMM can contain an IRD-EMM (processed locally by the digibox) and/or a CARD-EMM (sent to and processed by the smartcard).

Formatbyte (can be 2 or 7)	IRD-EMM lengthbyte	IRD-EMM data	Card-EMM lengthbyte	Card-EMM nano block

- **IRD-EMM block**
    - If the Formatbyte is 2:

      The IRD-EMM block does not contain filternanos. The only nano that I have seen is nano 02. The last byte in the IRD-EMM block is the checksum byte. The checksum is calculated by adding all bytes from incl. formatbyte to the byte before the checksum byte.

      Nanos without a lengthbyte:

Content of the first Byte	Number of remaining bytes	Remark

02	04	Set the date (the first two bytes) and the time (the last two bytes) in the digibox.
		Examples:
		`38 19 A4 9B (2001-09-25 20:37 GMT)`
		`38 19 BF 7C (2001-09-25 23:59 GMT)`
		`38 1A 00 01 (2001-09-26 00:00 GMT)`
		`38 1A 52 BC (2001-09-26 10:22 GMT)`

Here are examples:

```
02 (Formatbyte)
06 (IRD-EMM lengthbyte)
02 38 16 AA 3D (Set Date/Time nano)
3F (Checksum)
16 (Card-EMM lengthbyte)
09 10 00 00
01 38 16 AA 3D
10 29 58
67 08 03 AB B4 6C 7F 21 BE 21

02 (Formatbyte)
06 (IRD-EMM lengthbyte)
02 38 17 4D D7 (Set Date/Time nano)
7D (Checksum)
16 (Card-EMM lengthbyte)
09 10 00 00
01 38 17 4D D7
10 35 52
67 08 81 44 EF F2 D6 CB 14 EE



00
82 (Table-ID = EMM)
70
85 (Length)
00 (Address-type-byte = One sub-emm incl. no filterbocks)
00 01 (0000 => Lengthbyte follows / 0001 => Lengthword follows)
C0 40 (Length = 40 Words = 80 Bytes)
02 (Formatbyte)
4B (IRD-EMM lengthbyte)
10 48 01 1E 06 FD 00 00

50 6C 65 61 73 65 20 55 73 65 20 59 6F 75 72 20 4E 65 77 20 56
69 65 77 69 6E 67 20 43 61 72 64 0D 0A
= “Please Use Your New Viewing Card”

46 6F 72 20 41 73 73 69 73 74 61 6E 63 65 20 43 61 6C 6C 20 30
38 37 30 36 20 30 39 30 39 34 30
= “For Assistance Call 08706 090940”

9D (checksum)
31 (Card-EMM lengthbyte)
09 10 00 00
10 16 35
38 03 08 00 90
6F 02 09 60 (CAID 0960 = old Sky cards)
58 03 08 00 96
58 03 08 00 97
58 03 08 00 94
58 03 08 00 95
10 7C 66
67 08 32 38 2E 78 84 89 C2 A0
```

If it is an old sky card (with CAID 0960) in the box then to box shows the message (new cards have CAID 0961):

```
Please Use Your New Viewing Card
For Assistance Call 08706 090940
```

- If the Formatbyte is 7:

  The IRD-EMM block does contain a filtersection. Only if min. one of the filternanos is true the following data are executed.

  The first byte after the filtersection is the IRD-EMM-Type. There are 16 hex. types of IRD-EMM-Types numbered from 0 to 16 hex..

  IRD-EMM-Type 07 and/or 08 seems to be a download instruction for a new firmware. The data describes the manufacturer/modelcode (for a list look at the firmware section of my homepage) and the new versionnumber.

  The IRD-EMM (with formatbyte 7) is signed with a different signature algorithm than the card uses (the algorithm seems to be based on a HMAC concept and uses SHA-1 as a one-way function).

  Nanos without a lengthbyte:

Content of the first Byte	Number of remaining bytes	Remark
30	00	Forces the Filter to be valid.
31	04	The four bytes representing the card address. Only if this address is identical with the real card then the following data is executed.

  Here are a few Examples of IRD-EMM-Type 7:

```
07 (Formatbyte)
65 (IRD-EMM lengthbyte)
01 (length of filtersection)
30 (nano 30 force filter to be true)
07 (IRD-EMM-Type)
14
4F (Manufacturer = Amstrad)
03 (Modelcode = Black/Silver DRX200)
FF FF FF FF 00 1A 00 00 00 00 (1A = firmware version)
00 00 00 03 (Modelcode = Black/Silver DRX200)
00 00 00 4F (Manufacturer = Amstrad)
01 00 01 08 5D DC 3D 19 27 D5 83 26 DC C8 47 E6 BC E9 DE B4 83
C9 AB FA 30 D5 F1 0B ED D8 0E 38 07 C5 0E 60 F7 F0 42 3A C7 4C
E5 F5 68 2D 96 64 60 BA 11 C1 82 A4 EF 57 2B B4 2C 33 65 16 41
84 5D A2 1F E5 2C A9 61 DA 15 11 9F 00 7C
00 (Card-EMM lengthbyte)

07 (Formatbyte)
65 (IRD-EMM lengthbyte)
01 (length of filtersection)
30 (nano 30 force filter to be true)
07 (IRD-EMM-Type)
14
9F (Manufacturer = Pace)
20 (Modelcode = Sky+)
FF FF FF FF 00 1D 00 00 01 00 (1D = firmware version)
00 00 00 20 (Modelcode = Sky+)
00 00 00 9F (Manufacturer = Pace)
```

```
01 00 01 08 DB CC 82 AA 96 65 7E 53 D0 31 03 AC EA F8 8D 2E 2C
28 F8 4E 28 11 AB 9E 23 FD 66 AC 7A 7F 20 4F F3 56 BD 3D 90 AF
A3 BC DE 0A CD DF CE 33 E7 E5 46 FD 73 34 4E 64 5A 3F 4A A7 1C
67 A4 0E CC FC 3E 5E AE C5 96 13 AE 00 C7
00 (Card-EMM lengthbyte)

07 (Formatbyte)
65 (IRD-EMM lengthbyte)
01 (length of filtersection)
30 (nano 30 force filter to be true)
07 (IRD-EMM-Type)
14
4F (Manufacturer = Amstrad)
01 (Modelcode = Black DRX100)
FF FF FF FF 00 3B 00 00 00 00 (3B = firmware version)
00 00 00 01 (Modelcode = Black DRX100)
00 00 00 4F (Manufacturer = Amstrad)
01 00 01 08 08 D0 64 9B F1 16 6C E1 3D 6F BC 2F 3F E7 A1 C6 4E
00 FA E3 B5 29 F3 53 7B 91 79 3B B3 C1 BE 5E 7C B3 64 F5 A6 3A
0B D9 41 6A 50 86 4D 4C EB 28 8D A6 03 12 CC C0 1D 86 A0 AE AA
0A D7 60 6D CF BE 78 1C 0E 74 87 04 00 4B
00 (Card-EMM lengthbyte)

07 (Formatbyte)
65 (IRD-EMM lengthbyte)
01 (length of filtersection)
30 (nano 30 force filter to be true)
07 (IRD-EMM-Type)
14
4F (Manufacturer = Amstrad)
02 (Modelcode = Black/Silver DRX100)
FF FF FF FF 00 32 00 00 00 00 (32 = firmware version)
00 00 00 02 (Modelcode = Black/Silver DRX100)
00 00 00 4F (Manufacturer = Amstrad)
01 00 01 08 0F 8F 48 0A 9E AF C2 C6 C6 11 60 AE 76 4B 21 FD 71
9C 14 C1 23 04 23 68 64 D6 9A 9C E0 1A 90 E4 58 48 DF 67 A1 27
2D 71 77 B5 CF 76 AB 9E 29 E0 6C 2E C9 CB B4 12 6E CC B5 11 6B
5E 2C 16 B5 26 D3 1D 39 93 0D 89 39 00 01
00 (Card-EMM lengthbyte)
```

Here are a few Examples of IRD-EMM-Type 8:

```
07 (Formatbyte)
66 (IRD-EMM lengthbyte)
05 (length of filtersection)
31 00 09 87 3D (nano 31 with card address)
08 (IRD-EMM-Type)
11
6F (Manufacturer = Sony)
01 (Modelcode = Silver VTX-S750U)
FF FF FF FF 00 25 01
00 00 00 01 (Modelcode = Silver VTX-S750U)
00 00 00 6F (Manufacturer = Sony)
01 00 01 08 27 0C BC F6 14 BB 67 B3 F9 3D 2C 5C 79 FF 7C CB E7
49 F7 C6 2F 19 6E 10 56 AD 68 35 CE 1D 29 FB 3D 23 5F 2C 92 6E
76 44 00 2A 70 EC 50 02 C4 AC C6 5B DF 8F 52 EA 74 C5 11 7C 31
2A 84 34 80 AE CB C6 80 1B 26 3D B7 00 47
00 (Card-EMM lengthbyte)

07 (Formatbyte)
66 (IRD-EMM lengthbyte)
```

```
05 (length of filtersection)
31 00 2F CE D8 (nano 31 with card address)
08 (IRD-EMM-Type)
11
6F (Manufacturer = Sony)
01 (Modelcode = Silver VTX-S750U)
FF FF FF FF 00 25 01
00 00 00 01 (Modelcode = Silver VTX-S750U)
00 00 00 6F (Manufacturer = Sony)
01 00 01 08 DB EC F1 EE CB B2 41 6A AA DA BC 1A 1E AB 4D 41 F5
A3 8E C2 D5 C3 02 13 C9 6E CD 21 B2 7D 51 CF EA 82 89 7A 27 7B
13 3E 0D 2A 6D 6B B9 05 1A 35 7A B2 30 30 73 FF 51 C0 F8 03 14
B6 56 40 F8 42 6C F0 FA 93 EC 07 0D 00 30
00 (Card-EMM lengthbyte)

07 (Formatbyte)
66 (IRD-EMM lengthbyte)
05 (length of filtersection)
31 00 5D 00 6E (nano 31 with card address)
08 (IRD-EMM-Type)
11
6F (Manufacturer = Sony)
01 (Modelcode = Silver VTX-S750U)
FF FF FF FF 00 25 01
00 00 00 01 (Modelcode = Silver VTX-S750U)
00 00 00 6F (Manufacturer = Sony)
01 00 01 08 C2 FC 64 89 AC 23 45 D4 F8 F8 C2 CC D8 8D 26 0A DC
95 67 26 08 50 BF EC 78 5E 7B C1 77 05 78 E3 C8 72 AB 08 CD 0E
9B 14 1D C3 6D 3D AD D9 BB 66 5C 05 CF 23 7F 98 6D 2B 4C 63 4C
80 B4 6D 21 44 16 02 56 09 EB E3 DF 00 F2
00 (Card-EMM lengthbyte)
```

- **Card-EMM block**

  I saw only a Card-EMM block if the formatbyte is 2. If the formatbyte is 7 the Card-EMM lengthbyte was always 0 on my loggings, but there is code in the digibox to process them also.

  Two different types of nanos exist, nanos with a lengthbyte and nanos without a lengthbyte.

  Nanos with a lengthbyte:

Content of the first Byte	Typical value of the lengthbyte	Remark
07	01	The content of the payload was always 00. Seen in combination with the following EMM filter: one card.
34	LL	Filternano Not seen yet, but present in the digibox code. Format: 34 LL NN {Sl1 Sl2 Sl3 Sl4 Sh1 Sh2 Sh3 Sh4}. Match card serial in range Sl1..Sl4 to Sh1..Sh4, there might be NN repetition of the ranges.

37	LL	Filternano Not seen yet, but present in the digibox code. Format: 37 LL NN {S1 S2 Sa3 Sa4 Sb3 Sb4 Sc3 Sc4 Sd3 Sd4}. Match cards with serials S1 S2 S[a-d]3 S[a-d]4, the {..} repeated NN times; LL is lenght of the whole filter, including NN, but excluding the 37 and itself.

38	09	Filternano
		The first byte contains 06.
		The remaining eight bytes contain only numbers, capital letters or blanks (postcode).
		Here are three examples:
		`06 4C 4E 34 20 20 33 45 46 = 06 "LN4  3EF"`
		`06 4F 58 33 33 20 20 20 20 = 06 "OX33    "`
		`06 53 54 36 20 20 20 20 20 = 06 "ST6     "`
		Only if this postcode is identical with that stored on the card then the following nanos in the block are executed. The postcode from the card can be queried with the command 48 48 00 00 36. After the byte 1E in the answer of the card the eight character long postcode is contained.
		Nano 38 (postcode) and nano 31 (card address) do never occur in the same block.
		
		Seen in combination with the following EMM filter: one card or all cards.

3B	LL	Filternano Not seen yet, but present in the digibox code. Format: 3B LL NN {S1 S2 S3 S4}. Match any of card serial #S1..S4, NN different cards.

5B	0B	Here are three examples:
		`E0 00 86 AE 00 64 00 00 3F 00 00`
		`E0 00 87 4F FF FF 00 00 FF FF F1`
		`E0 00 9A 3F FF FF 00 00 C1 C1 F1`
		
		Seen in combination with the following EMM filter: one card.

5C	05	Here are three examples:
		`00 6F FE 37 1B`
		`0A 6A C7 37 0E`
		`17 71 F0 38 07`
		This nano can occur more then once in a block.
		
		Seen in combination with the following EMM filter: one card.

67	08	Signature (crypto checksum of the bytes between nano 09 and this nano) This nano is always present in a block. Seen in combination with the following EMM filter: one card or all cards or cardgroup.

6C	04	Here is the only bytes sequence I have found: `00 91 18 01` Seen in combination with the following EMM filter: one card.

6D	09	Here are three examples:
		`01 08 00 73 17 92 9F FF FF`
		`02 03 45 57 60 00 FF FF FF`
		`03 01 25 42 11 11 1F FF FF`
		This nano can occur more then once in a block.
		
		Seen in combination with the following EMM filter: one card or all cards.

6E	07	Here are three examples:
		`06 01 01 01 01 20 08`
		`07 01 01 01 01 8F 00`
		`08 01 01 01 01 20 00`
		This nano can occur more then once in a block.
		
		Seen in combination with the following EMM filter: one card or all cards.

6F	02	Filternano Format: 6F 02 CA1 CA2. Match CA System ID (typically 0960 for old cards or 0961 for new cards).

72	02	Here are the only two bytes sequences I have found:
		`00 01`
		`00 02`
		
		Seen in combination with the following EMM filter: one card.

73	0C	Here are three examples:
		`00 00 65 00 01 00 01 00 00 00 00 1A`
		`00 00 65 00 1A FF FF 0B B9 00 00 1A`
		`03 00 6B 00 04 FF FF 00 00 00 00 1A`
		This nano can occur more then once in a block.
		
		Seen in combination with the following EMM filter: one card.

75	0F	Here are three examples from an EMM filter for one card only (seems to use only length 0F):
		`12 00 02 00 00 00 00 00 00 00 00 08 00`
		`12 01 82 00 01 00 00 00 E0 37 00 10 00`
		`12 11 18 80 04 00 00 00 00 00 00 08 00`

	13	Here are three examples from an EMM filter for all card (seems to use only length 13):
		`00 47 42 52 01 00 00 00 00 00 00 00 00 00 00 00 00 00 00`
		`00 47 42 52 02 00 00 00 00 00 00 00 00 00 00 00 00 00 00`
		`00 47 42 52 08 00 00 00 00 00 00 00 00 00 00 00 00 00 00`
		The meaning of the bytes `47 42 52` is GBR
		
		Seen in combination with the following EMM filter: one card or all cards.

76	02	Here are three examples:
		`80 90`
		`80 CD`
		`80 FF`
		This nano can occur more then once in a block.
		
		Seen in combination with the following EMM filter: one card.

90	15, 1E, 25, 2F, 32, 3B, 3C, 50, 5A, 5C or 64	This nano never appears with other nanos incl. the key selection and the signatur nano.
		It seems this is the encrypted nano that is used by the new SKY card.
		
		Here are three examples from an EMM filter for one card only:
		`42 02 05 2B 33 39 5C 1F 1F 01 D5 B1 8F 6D 75 88 0E 29 96 37 58`
		
		`42 02 00 4E DA 1F 7C EE 60 65 46 B7 64 4D 95 FC 99 9F 66 7F 99 7E B7 08 FA D3 B8 7E 3B BE`
		
		`42 02 9B FC D4 73 87 92 28 EA 65 47 D3 7A 7C 92 90 3D CA BF 20 69 6C C2 AA 26 28 20 39 32 4D 67 2F 58 E2 76 26 5F B9 F7 86 DB E5 53 7B 62 55 1E A3 EA 9E FD 8D 4C 65 15 A8 B3 70 D2 37 3F BD D3 DD 43 F6 71 95 DB 16 B1 32 47 53 0F E2 B4 DA A0 8E 02 2F D2 3B 9C 72 D8 70 9D 57 DE 7F F1 3C 24 15 85 E9 47`

	16 or 3F	Here are three examples from an EMM cardgroup filter (seems to use only length 16 or 3F):
		`B4 A0 4A 2F A9 18 CC BB 31 B8 0B 09 B4 81 7E 54 A5 D3 43 4D`
		
		`42 02 0C AC EA 5F 66 00 2B CF C2 6B 19 04 50 59 65 7F 7E 9E 84 E9 C3 71 42 5C 45 D0 C2 37 78 4E D8 86 20 CA 15 ED 60 79 27 5A 6E 6B 07 5A 6E D2 2D 1B F9 95 4F 2F B2 99 AA 0C E9 D1 8B EF 18`
		
		`42 02 6B 9F 2D 48 84 9F 3A 81 56 0B 76 7E DC 4B 57 13 8C AB 54 C1 E2 D2 A9 B8 40 83 DB 5A 17 C2 23 D9 3E E6 9F 30 83 9C 11 83 71 C7 90 34 F7 C4 2F 12 15 77 87 2E 2B E2 23 B9 79 F2 EA 00 C4`
		
		Seen in combination with the following EMM filter: one card or cardgroup.

Nanos without a lengthbyte:

Content of the first Byte	Number of remaining bytes	Remark

01	04	Set the date (the first two bytes) and the time (the last two bytes).
		Examples:
		`38 19 A4 9B (2001-09-25 20:37 GMT)`
		`38 19 BF 7C (2001-09-25 23:59 GMT)`
		`38 1A 00 01 (2001-09-26 00:00 GMT)`
		`38 1A 52 BC (2001-09-26 10:22 GMT)`
		
		Seen in combination with the following EMM filter: all cards.

09	03	The first byte shows the keynumber of the key that was used to calculate the signature (nano 67).
		But what is the meaning of the last two bytes?
		Here are the only examples I have found:
		`10 00 00`
		`12 00 00`
		`14 00 00`
		`14 80 00`
		Key 10 is identical on all cards.
		Key 12 is identical within one card group and is always used in combination with nano 32 or 33.
		Key 14 is individual on each card and is always used in combination with nano 31.
		This nano is always present in a block.
		
		Seen in combination with the following EMM filter: one card or all cards or cardgroup.

10	02	The two bytes show an increasing sequence.
		But what is the meaning of this sequence?
		Here are a few examples:
		`29 FE`
		`29 FF`
		`2A 00`
		`2A 01`
		`2A 02`
		`2A 03`
		
		Seen in combination with the following EMM filter: all cards.

19	01	Set the Timezone. The only value I have found is 0. Seen in combination with the following EMM filter: one card or all cards.

1E	08	The eight bytes contain only numbers, capital letters or blanks (postcode).
		Here are three examples:
		`41 42 31 30 20 20 20 20 -- "AB10    "`
		`43 48 34 20 20 37 45 4A -- "CH4  7EJ"`
		`5A 45 32 20 20 20 20 20 -- "ZE2     "`
		Nano 1E is always used in combination with nano 31. Nano 1E and nano 38 never occurs in the same block.
		
		Seen in combination with the following EMM filter: one card.

24	00	Because this nano don’t have a payload, it’s only a flag that a specific action should be executed. But which one? Seen in combination with the following EMM filter: one card or all cards.

25	00	Filternano This nano inverts the filter result. Example: Nano 09 (KeyNr) => nano 33 (Cardgroup & Matrix) => nano 41 (Add ChID) => nano 25 (Invert filter) => 42 (Delete ChID) => 67 (Signature). By all cards matching the matrix the new ChID was added, but through nano 25 the ChID was deleted by all cards not in the matrix. Seen in combination with the following EMM filter: one card or cardgroup.

27	0D	Here are two examples:
		`07 C4 1F 6B A6 E5 81 A9 00 EC D8 A1 96`
		`74 49 8D F5 6A 7A 9E 75 00 EC D8 B5 96`
		
		Seen in combination with the following EMM filter: one card.

2B	02	Here are three examples:
		`03 EB`
		`03 F5`
		`04 0B`
		
		Seen in combination with the following EMM filter: one card.

2D	04	Here are three examples:
		`00 40 00 00`
		`38 99 4C 23`
		`39 96 A0 24`
		
		Seen in combination with the following EMM filter: one card.

30	00	Filternano Forces the Filter to be valid. Seen in combination with the following EMM filter: all cards.
31	04	Filternano The four bytes representing the card address. Only if this address is identical with the real card then the following nanos are executed. Nano 31 (card address) and nano 38 (postcode) do never occurs in the same block. Seen in combination with the following EMM filter: one card.

32	03	Filternano The three bytes representing the card group. Only if these bytes are identical with the first three bytes of the card address then the following nanos are executed. Nano 32 (whole group) and 33 (part of a group) do never occur in the same block. Seen in combination with the following EMM filter: cardgroup.
33	23	Filternano The three bytes representing the card group. The remaining 20 (32 dec.) bytes shows the addressmatrix (like CB20 by Irdeto). This addressmatrix contains exact 256 dec. bits. The positions of set bits shows the 4th bytes of the card addresses there are addressed. The counting starts from the leftmost byte and from the least significant bit (lsb). I will show you two examples to make it clear: `33 00 12 34 01 00 00...` means that the card `00 12 34 00` is addressed (bitstring: `00000001000000000000000...`). `33 00 12 34 0A 40 00...` means that the cards `00 12 34 01, 00 12 34 03` and `00 12 34 0E` are addressed (bitstring: `00001010010000000000000...`). A good demonstration program called BitmapCalc from the Skybusters team is available on the internet. Seen in combination with the following EMM filter: cardgroup.

3D	02	Here are three examples:
		`03 EA`
		`04 06`
		`04 39`
		
		Seen in combination with the following EMM filter: one card.

3E	00	Because this nano don't have a payload, it's only a flag that a specific action should be executed. But which one? Seen in combination with the following EMM filter: one card.

41	05	Add a ChID.
		The first two bytes represent the channel ID.
		The next two bytes represent the last date on which this channel ID is valid.
		But what is the meaning of the last byte?
		Here are three examples:
		`00 10 39 1F 00`
		`00 1D 39 1F C0`
		`B8 5A 39 1F C0`
		This nano can occur more then once in a block.
		
		Seen in combination with the following EMM filter: one card or cardgroup.

42	02	Delete a ChID.
		The two bytes represent the channel ID.
		Here are three examples:
		`00 0C`
		`60 FE`
		`B8 5A`
		This nano can occur more then once in a block.
		
		Seen in combination with the following EMM filter: one card or all cards or cardgroup.

44	04	Filternano Not seen yet, but present in the digibox code. Format: 44 I1 I2 I3 I4. Match IRD # I1..I4 only.

4E	04	Here are three examples:
		`00 00 01 DC`
		`00 45 81 BA`
		`00 6A 70 70`
		
		Seen in combination with the following EMM filter: one card.

7A	02	Filternano Not seen yet, but present in the digibox code. Format: 7A PM NM. Match bits in the card fuse byte, 1 byte mask for positive and 1 byte mask for negative matches.

Chapter 2: ECM (Entitlement Control Message)

The ECMs can be logged e.g. on Astra 2 (28.2 degrees east) from the BBC transponder 11758 H SR 27.5 MS/s FEC 2/3. One of this ECM PIDs is 500 hex. (1280 dec.).

Each TS packed contains 188 bytes, but the most loggingprograms stores only the last 184 bytes.

These ECMs (with size of 184 bytes) always starts with the byte sequence 00 80 70 or 00 81 70. The following byte contains the length of the payload. After the payload itself, bytes with the value FF are used to fill the rest of the section.

Here are three examples:

```
00 80 70 87 00 00 01 0E 38 19 A4 9D 13 1E F9 67 11 20 01 00 00 64 74 7E 12
00 00 00 00 00 00 00 00 20 0F B8 F5 54 0C D4 5A 00 00 7F 12 8D A5 E6 66 DE
05 5B A8 B7 43 C2 57 13 C7 EF C4 92 E7 09 10 10 00 01 38 19 A4 9D CB 02 FF
FF 02 53 03 74 BB C0 03 B7 DE C0 03 B8 28 C0 38 03 08 00 CC 25 04 4D 0E 74
BB 39 01 00 00 00 00 00 A3 E3 00 D9 E4 4D 0E 74 BB 39 01 00 00 00 00 01 45
E3 01 B1 E4 67 08 80 54 61 36 84 34 D3 33 FF FF FF FF FF FF FF FF FF FF FF
FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF
FF FF FF FF FF FF FF FF FF

00 81 70 89 00 00 01 0E 38 19 A4 AA FF FF E1 A2 11 20 01 00 00 61 76 7E 12
00 00 00 00 00 00 00 00 82 5C ED EE B1 FF 8E 1A 00 00 7F 12 5B 36 8A B0 9A
97 7B E9 2C D6 85 E6 68 16 AE 40 E2 D7 09 10 10 00 01 38 19 A4 AA CB 02 FF
FF 02 40 38 05 08 00 1A 00 1D 04 03 00 1D 00 24 38 03 08 00 1A 25 38 03 08
00 91 04 03 00 10 00 24 38 03 08 00 1A 58 03 08 00 91 25 04 24 38 03 08 00
E0 04 03 B7 C8 00 67 08 81 3F D9 E2 71 BE F5 6E FF FF FF FF FF FF FF FF FF
FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF
FF FF FF FF FF FF FF FF FF

00 80 70 54 00 00 01 0E 38 1A 23 C5 FF FF 32 50 01 20 01 00 00 EB 41 7E 12
00 00 00 00 00 00 00 00 7E CC A1 C2 5E B6 85 92 00 00 7F 12 76 68 2C CF 57
B6 41 F6 80 E4 86 8F 7B C5 C3 0D 92 44 09 10 10 00 01 38 1A 23 C5 CB 02 FF
FF 02 80 67 08 70 7F 04 34 82 D6 A1 D8 FF FF FF FF FF FF FF FF FF FF FF FF
FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF
FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF
FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF
FF FF FF FF FF FF FF FF FF
```

The values of the first three bytes are always 00 00 01.

An ECM contain an IRD-ECM (processed locally by the digibox) and a CARD-ECM (sent to and processed by the smartcard).

IRD-ECM lengthbyte	IRD-ECM data	Card-ECM lengthbyte	Card-ECM nano block

- **IRD-ECM block**

  The only IRD-ECM lengthbyte I have seen is 0E.

  IRD-ECM data:
  - The first two bytes shows the date and the next two bytes the time. Examples:
    ```
    38 19 A4 9B = 2001-09-25 20:37 GMT
    38 19 BF 7C = 2001-09-25 23:59 GMT
    38 1A 00 01 = 2001-09-26 00:00 GMT
    38 1A 52 BC = 2001-09-26 10:22 GMT
    ```
  - The next four bytes can vary.
  - The following byte is always 01 or 11.
  - The next four bytes are always 20 01 00 00.
  - The following byte can vary. This was the last content byte of the header. Perhaps it's a checksum byte.

- **Card-ECM block**

  After the Card-ECM lengthbyte the Card-ECM nano block follows.

  Two different types of nanos exist, nanos with a lengthbyte and nanos without a lengthbyte.

  Nanos with a lengthbyte:

Content of the first Byte	Typical value of the lengthbyte	Remark

38	03/05/07	Here are three examples:
		`08 00 18`
		`08 00 1A 00 1D`
		`08 00 93 00 91 00 1D`
		This nano can occur more then once in a block.

4D	0E	Here are three examples:
		`01 6C 39 01 00 00 00 00 00 A3 E3 00 D9 E4`
		`25 9F 39 01 00 65 06 00 00 A3 E3 00 D9 E4`
		`7E 36 39 02 00 65 06 00 01 45 E3 01 B1 E4`
		This nano can occur more then once in a block.

58	03	Here the only two examples I have found:
		`08 00 1D`
		`08 00 91`
		The content looks similar to nano 38.

67	08	Signature (crypto checksum of the bytes between nano 09 and this nano) This nano is always present in a block.

7E	12	This nano have something to do with controllwords:
		Here are three examples:
		`00 00 00 00 00 00 00 00`
		`B9 43 44 F4 76 52 D5 D8`
		`00 00`
		
		`00 00 00 00 00 00 00 00`
		`E6 8F 31 76 76 12 DD D2`
		`00 00`
		
		`00 00 00 00 00 00 00 00`
		`28 4E 14 FE CA 4F 41 9F`
		`00 00`

7F	12	This nano also have something to do with controllwords. It seems that some IRD's use the nano 7E and others nano 7F.
		Here are three examples:
		`24 39 4F 4E 1B 5A 66 AA`
		`BC 61 EE 38 5E 96 B8 44`
		`F4 3D`
		
		`B6 39 13 FB D0 F6 76 96`
		`D3 45 A1 3A 5A 90 D2 3A`
		`94 75`
		
		`95 06 2C AB 5F 9B 2F CA`
		`E1 FD 21 3B 6E 42 F8 DC`
		`65 B5`

CB	02	Here are the only byte sequence I have found: `FF FF` This nano is always present in a block.

Nanos without a lengthbyte:

Content of the first Byte	Number of remaining bytes	Remark

01	04	Set the date (the first two bytes) and the time (the last two bytes).
		Examples:
		`38 19 A4 9B (2001-09-25 20:37 GMT)`
		`38 19 BF 7C (2001-09-25 23:59 GMT)`
		`38 1A 00 01 (2001-09-26 00:00 GMT)`
		`38 1A 52 BC (2001-09-26 10:22 GMT)`

02	01	Here are three examples:
		`00`
		`43`
		`80`
		This nano is always present in a block.

03	03	The first two bytes represent the channel ID.
		But what is the meaning of the last byte?
		Here are three examples:
		`00 10 00`
		`B7 21 00`
		`B8 5B 00`
		This nano can occur more then once in a block.
		Only if min. one of these channel IDs is present on the card, the card gives us the secret controllwords (necessary for decrypting the picture).

04	00	Because this nano don't have a payload, it's only a flag that a specific action should be executed. But which one? This nano can occur more then once in a block.
09	03	The first byte shows the keynumber of the key that was used to calculate the signature (nano 67). But what is the meaning of the last two bytes? Here is the only example I have found: `10 10 00` Key 10 is identical on all cards. This nano is always present in a block.
24	00	Because this nano don't have a payload, it's only a flag that a specific action should be executed. But which one? This nano can occur more then once in a block.
25	00	Because this nano don't have a payload, it's only a flag that a specific action should be executed. But which one?
48	00	Because this nano don't have a payload, it's only a flag that a specific action should be executed. But which one?

Chapter 3: Channel ID Tiers

The following table shows which channel IDs are used on which channel.
The first two bytes of nano 03 represents the CHID.

Channels send over transponder beam 2D are not shown, because I'm unable to receive it in Germany.

Channel ID Tiers that I can not log any more (changed from FTV to FTA or to Beam 2D) are shown in parenthesis.

Channel	**Channel ID Tiers**
Sky World	
Movies	
Sky Movies Premier 1	30E2 B74D
Sky Movies Premier 2	30E2 B74E
Sky Movies Premier 3	30E2 B74F
Sky Movies Premier 4	30E2 B750
Sky Movies Premier Wide	30E2 B754
Sky Movies Max 1	30E1 B749
Sky Movies Max 2	30E1 B74A
Sky Movies Max 3	30E1 B74B
Sky Movies Max 4	30E1 B74C
Sky Movies Max 5	30E1 B78A
Sky Movies Cinema 1	B732 B747
Sky Movies Cinema 2	B732 B748
Sports	
Sky Sports 1	30E3
Sky Sports 2	30E4
Sky Sports 3	30E5
Sky Sports Extra	30E3 30E4
Family Pack	
Sports	
Sky Sports News	0010 B752
British Eurosport	0010 B776
attheraces	0010 1531
Entertainment	
Sky One	0010 B758 B759 B75A B75B B757 B723
Sky One Mix	0010 B889
UK Gold	0010 B758 B743
UK Gold +1	0010 B87B
UK Gold 2	0010 B745
Living TV	0010 B758 B73C
Living TV +1	0010 B87F
Granada Plus	0010 B75B B727
Challenge?	0010 B75B B73D
Bravo	0010 B75B B73F
Bravo +1	0010 B890
Paramount	(0010 B758 B762)
Sci-Fi	0010 B75B B729

Discovery Home & Leisure	0010 B757 B758 B759 B75A B75B B753
Discovery Home & Leisure + 1	0010 B84F
Men and Motors	0010 B75B B761
UK Style	0010 B75B B740
UK Style+1	0010 B869
UK Food	0010 B85C
UK Drama	0010 B75B B742
Discovery Health	0010 B833
E4	0010 B847
CNX	0010 B884
E!	0010 5B6B
UK Bright Ideas	0010 3C4D
Ftn	0010 33C0
Sky Travel Shop	0010 B738
News and Documentaries	
Bloomberg	0010 B757 B758 B759 B75A B75B B725
CNBC	0010 B759 B735
Fox News	0010 B867
Discovery Channel	0010 B758 B759 B733
Discovery Channel + 1hr	0010 B758 B759 B763
Discovery Travel and Adventure	0010 B75D
Discovery Civilisation	0010 B75E
Discovery Sci-Trek	(0010 B75F)
Discovery Wings	0010 B831
National Geographic	0010 B758 B759 B728
National Geographic Channel + 1	0010 B758 B759 B7C9
Adventure One	0010 B7CB
The History Channel	0010 B759 30DD
History + 1 hr	0010 B759 B839
Biography	0010 B759 B83A
UK Horizons	0010 B759 B741
UK Horizons + 1	0010 B759 B84D
Discovery Animal Planet	0010 B759 B734
Discovery Animal Planet + 1	0010 B84E
UK History	0010 B880
Music and Radio	
MTV	0010 B75A B72E
MTV Hits	0010 B77B
VH1	0010 B75A B72F
VH1 Classic	0010 B77D
MTV 2	0010 B75A B760
MTV Base	0010 B77C
MTV Dance	0010 B84B
TMF	0010 223A
The Box	0010 B75A B730
KISS	0010 B82B
Smash Hits	0010 B84C
Magic	0010 B853
Q	0010 B82C
Kerrang!	0010 B849
The Hits	0010 255F
Music Choice (10 channels)	0010 B757 B758 B759 B75A B75B B764
Big Blue The Saint The Villan	0010 B858 B85A B85B B859
Kids	

Cartoon Network	0010 B758 B75A B774
Cartoon Network + 1	0010 B758 B75A B774
Boomerang	0010 B88F
Nickelodeon	0010 B758 B75A B72C
Nick Replay	0010 B758 B75A B72C
Nick Toons TV	0010 B87D
Trouble	0010 B75A B75B B73E
Fox Kids	0010 B758 B75A B72A
Fox Kids + 1	0010 B758 B75A B72A
Nick Jr.	0010 B7CA
Non-subscription channels	
Channel 4	B766
five	B765
BBC ONE	(001D)
BBC TWO	(001D)
BBC THREE	(001D)
BBC FOUR	(001D)
S4C ~ Digitol	(B777)
Various channels	
Music Choice Extra (30 channels)	B721
Bangla TV	B878
Premiership Plus Chelsea TV	0010 B757 B758 B759 B75A B75B B739
LBC ART Europe ART Movies ART Music Al Jazeera	B85F
FilmFour FilmFour +1 Film4 Weekly	Night: B770 only. Day: B770 0010 B757 B758 B759 B75A B75B
redhot uktalent redhot uktalent PH	B857
SouthForYou	B879
NASN	2590
Ekushey TV	B865
Red Hot Wives Red Hot Films RH 40+ Wives redhot amateur RedHot All Girl	B857 (20.00 – 03.00) B773 (03.30 – 20.00)
PTV Prime	B837
STREAM-0 STREAM-1 STREAM-2 STREAM-3 STREAM-4 STREAM-5 STREAM-6	0010 001D B7C8
Pub Channel	B78B B78C B78D B78E B78F B790 B791 B792 B793 B794
Playboy / Adult	B71E
ISM Test1 v5 ISM Sports 1 v6 ISM Sky One v6	B778

ISM Sky News v6 STT 101-5	
Spice Xtreme	B87E
Snr	B7D9
Tantalise TV	B874
STAR Plus STAR News	B838
Talk Sport	0010 B757 B758 B759 B75A B75B B7D3
Zee TV / Zee Music	B79A
Zee Cinema Sony TV Asia	B7CC
Alpha EtcPun	B7D5
ARY Digital	B7D2
B4U Movies	B799
TV5	0010 B86D
Television X	B76F
Artsworld	B82F
MUTV	B73B

Please read also the other NDS Faqs (e.g. “NDS Command Structure v7.77” by TheSkyLog Team updated by Powermouse).

For corrections and additional infos that you can send to the following email address, many thanks in advance. Many thanks to all people which have already done this: colibri_dvb@lycos.com

You can find the newest document at the following page: http://colibri.move.to/

On my homepage you can find also firmware for 28 different digibox models and tools for extracting and decompressing logged OverTheAir firmware updates and a tool to transfer firmware/EEPROM config from/to a PACE2500N digibox.