

User Guide Transaction Logs For Generation 6 Reader Systems

Rev. 2.06

Last modification: 05. Mrz. 2013

Impressum:

MICROTRONIC AG
Duennernstrasse 32
CH-4702 Oensingen

Tel +41 62 388 45 45
Fax +41 62 388 45 55
e-mail info@microtronic.ch
 <http://www.microtronic.ch>

History:

Date:	Version:	Modification
06.09.2010	01.01	Release/DG
20.09.2010	01.02	Added on C-protocol - 'T' par.3 & 'U' par.1. Pricelist Tarif mod / DG
25.10.2010	01.03	Added S-Protocol: 't' for FTP upload / DG
03.01.2011	01.04	Format type changed from ,0' to ,n' (signed) at the following protocols: A-Protocol: f,L,l,n,S,T,V E-Protocol: n
02.02.2011	01.05	Protocols A- 'P', A-p, C-?, C-&, C-! modified for BV / DG
08.03.2011	01.06	Modifications for BV: S-H new defined. Added: E-M and E-m. A-Q modified / DG
11.03.2011	01.07	A-S modified. Now the technology will be recorded /DG
11.03.2011	01.08	A-g modified. Added PCA payback lever / DG
11.03.2011	01.09	A-A modified. VMC-type now logged in ASCII-format / DG
01.04.2011	01.10	C-& modified for new MDB/RD implementation / DG
06.04.2011	01.11	S-+, C-?, A-P, A-p modified for implementation of MDB/CG
12.04.2011	01.12	E-M for MDBRD modified/DG
11.09.2012	02.00	Completely revised, new transactions for peripheral / DG
01.10.2012	02.01	S-t added DTS event files and new error codes for FTP client / DG
19.10.2012	02.02	S-t modified, download and file names / DG
13.12.2012	02.03	Added protocols for Happy-Meals: Class A- 'H' and A- 't'. A- 'j' (token), A- 'L' and A- 'I' (Cash) modified for Debit Master. Class E- 'R' modified
06.02.2013	02.04	New entry at „7-Pricelist-Information” for reduction per category: „Reduction Amount Category“ (Source Group „Vmc“)
07.02.2013	02.05	New protocol for G5 Statistic Tag: 6S ,A' (Activate IrDA) New source for S-'I', Configuration-update over FTP / MDD / File DL Tag)
05.03.2013	02.06	Protocol S- 't' modified / DG

Table of Contents

1. Introduction	6
1.1. Basics	6
1.2. Structure	6
1.3. Output visualisation	6
1.4. Data Encoding	7
1.5. Data Format of Transaction Logs	8
2. Transaction Classes	8
3. Transaction log Class 'C' (Configuration)	9
3.1. Transaction details	10
3.1.1. Transaction log Type 'F' (Class 6C)	10
3.1.2. Transaction log Type 'N' (Class 6C)	10
3.1.3. Transaction log Type 'P' (Class 6C)	11
3.1.4. Transaction log Type 'R' (Class 6C)	11
3.1.5. Transaction log Type 'S' (Class 6C)	12
3.1.6. Transaction log Type 'T' (Class 6C)	12
3.1.7. Transaction log Type 'U' (Class 6C)	13
3.1.8. Transaction log Type 'W' (Class 6C)	13
3.1.9. Transaction log Type 'I' (Class 6C)	13
3.1.10. Transaction log Type '#' (Class 6C)	13
3.1.11. Transaction log Type '%' (Class 6C)	13
3.1.12. Transaction log Type '&' (Class 6C)	14
3.1.13. Transaction log Type '*' (Class 6C)	15
3.1.14. Transaction log Type '~' (Class 6C)	15
3.1.15. Transaction log Type '?' (Class 6C)	16
4. Transaction log Class 'S' (System)	17
4.1. Transaction Log Details	18
4.1.1. Transaction log Type 'A' (Class 6S)	18
4.1.2. Transaction log Type 'C' (Class 6S)	18
4.1.3. Transaction log Type 'D' (Class 6S)	18
4.1.4. Transaction log Type 'd' (Class 6S)	19
4.1.5. Transaction log Type 'E' (Class 6S)	19
4.1.6. Transaction log Type 'F' (Class 6S)	20
4.1.7. Transaction log Type 'f' (Class 6S)	21
4.1.8. Transaction log Type 'G' (Class 6S)	21
4.1.9. Transaction log Type 'H' (Class 6S)	22
4.1.10. Transaction log Type 'I' (Class 6S)	23
4.1.11. Transaction log Type 'L' (Class 6S)	23
4.1.12. Transaction log Type 'M' (Class 6S)	23
4.1.13. Transaction log Type 'n' (Class 6S)	24
4.1.14. Transaction log Type 'o' (Class 6S)	24
4.1.15. Transaction log Type 'P' (Class 6S)	25
4.1.16. Transaction log Type 'r' (Class 6S)	26
4.1.17. Transaction log Type 'S' (Class 6S)	27
4.1.18. Transaction log Type 'T' (Class 6S)	28
4.1.19. Transaction log Type 't' (Class 6S)	29
4.1.20. Transaction log Type 'V' (Class 6S)	31
4.1.21. Transaction log Type 'X' (Class 6S)	31
4.1.22. Transaction log Type '%' (Class 6S)	31
5. Transaction log Class 'A' (Application)	32
5.1. Transaction Log Details	33
5.1.1. Transaction log Type 'A' (Class 6A)	33
5.1.2. Transaction log Type 'a' (Class 6A)	33
5.1.3. Transaction log Type 'B' (Class 6A)	34
5.1.4. Transaction log Type 'C' (Class 6A)	35
5.1.5. Transaction log Type 'D' (Class 6A)	36
5.1.6. Transaction log Type 'd' (Class 6A)	37
5.1.7. Transaction log Type 'e' (Class 6A)	38
5.1.8. Transaction log Type 'F' (Class 6A)	38
5.1.9. Transaction log Type 'f' (Class 6A)	39
5.1.10. Transaction log Type 'G' (Class 6A)	39
5.1.11. Transaction log Type 'g' (Class 6A)	40



5.1.12.	Transaction log Type `H` (Class 6A)	40
5.1.13.	Transaction log Type `h` (Class 6A)	41
5.1.14.	Transaction log Type `I` (Class 6A)	41
5.1.15.	Transaction log Type `i` (Class 6A)	41
5.1.16.	Transaction log Type `J` (Class 6A)	42
5.1.17.	Transaction log Type `j` (Class 6A)	42
5.1.18.	Transaction log Type `K` (Class 6A)	42
5.1.19.	Transaction log Type `k` (Class 6A)	42
5.1.20.	Transaction log Type `L` (Class 6A)	43
5.1.21.	Transaction log Type `l` (Class 6A)	44
5.1.22.	Transaction log Type `M` (Class 6A)	45
5.1.23.	Transaction log Type `m` (Class 6A)	46
5.1.24.	Transaction log Type `n` (Class 6A)	46
5.1.25.	Transaction log Type `O` (Class 6A)	47
5.1.26.	Transaction log Type `P` (Class 6A)	48
5.1.27.	Transaction log Type `p` (Class 6A)	50
5.1.28.	Transaction log Type `Q` (Class 6A)	51
5.1.29.	Transaction log Type `q` (Class 6A)	51
5.1.30.	Transaction log Type `R` (Class 6A)	51
5.1.31.	Transaction log Type `r` (Class 6A)	51
5.1.32.	Transaction log Type `S` (Class 6A)	53
5.1.33.	Transaction log Type `T` (Class 6A)	54
5.1.34.	Transaction log Type `t` (Class 6A)	55
5.1.35.	Transaction log Type `V` (Class 6A)	56
5.1.36.	Transaction log Type `v` (Class 6A)	56
5.1.37.	Transaction log Type `W` (Class 6A)	56
5.1.38.	Transaction log Type `X` (Class 6A)	56
5.1.39.	Transaction log Type `Y` (Class 6A)	57
5.1.40.	Transaction log Type `y` (Class 6A)	57
5.1.41.	Transaction log Type `~` (Class 6A)	58
6.	Transaction Types of Class `E` (Error)	59
6.1.	Transaction Log Details	60
6.1.1.	Transaction log Type `A` (Class 6E)	60
6.1.2.	Transaction log Type `C` (Class 6E)	61
6.1.3.	Transaction log Type `D` (Class 6E)	62
6.1.4.	Transaction log Type `E` (Class 6E)	62
6.1.5.	Transaction log Type `e` (Class 6E)	63
6.1.6.	Transaction log Type `F` (Class 6E)	63
6.1.7.	Transaction log Type `G` (Class 6E)	63
6.1.8.	Transaction log Type `H` (Class 6E)	63
6.1.9.	Transaction log Type `L` (Class 6E)	64
6.1.10.	Transaction log Type `l` (Class 6E)	65
6.1.11.	Transaction log Type `M` (Class 6E)	66
6.1.12.	Transaction log Type `m` (Class 6E)	67
6.1.13.	Transaction log Type `n` (Class 6E)	67
6.1.14.	Transaction log Type `P` (Class 6E)	68
6.1.15.	Transaction log Type `R` (Class 6E)	69
6.1.16.	Transaction log Type `r` (Class 6E)	69
6.1.17.	Transaction log Type `T` (Class 6E)	69
6.1.18.	Transaction log Type `U` (Class 6E)	70
6.1.19.	Transaction log Type `W` (Class 6E)	70
6.1.20.	Transaction log Type `y` (Class 6E)	72
7.	Pricelist Information	73
8.	Transaction Log Examples	74
8.1.	Badge vend (VMC prices)	74
8.1.1.	Badge vend listing of transaction log:	74
8.2.	Token vend (VMC prices)	75
8.2.1.	Token vend listing of transaction log:	75
8.3.	Bonus vend (VMC prices)	76
8.3.1.	Bonus vend listing of transaction log:	76
8.4.	Cash vend (VMC prices)	77
8.4.1.	Cash vend listing of transaction log:	78
8.5.	Loading to Media	79
8.5.1.	Loading listing of transaction log:	79

1. Introduction

This documentation describes the transactions which are logged in generation 6 (G6) reader systems.

1.1. Basics

All basic events are logged persistently in the reader system (in non-volatile memory). Any manipulation on the reader is also logged. Transaction logs can also be used to verify sequential history of all user actions.

The readers can be set up to store standard or detailed transaction logs which means that even an MDB protocol analysis could be stored in the memory. The detailed transaction logs are marked with , or .

1.2. Structure

DATE	TIME	TYPE	PARAM0	PARAM1	PARAM2	PARAM3	FLAG
------	------	------	--------	--------	--------	--------	------

1.3. Output visualisation

Nummer	FLAG	DATE	TIME	TYPE	PARAM0	PARAM1	PARAM2	PARAM3
--------	------	------	------	------	--------	--------	--------	--------

Number	Class	Date	Time	Type	Par 0	Par 1	Par 2	Par 3
--------	-------	------	------	------	-------	-------	-------	-------

1.4. Data Encoding

Multibyte fields are stored in *little-endian* format i.e. the least significant byte (LSB) is stored first.

Data types used in this document:

NIBBLE	4-bit field
BYTE	1-byte field
ASCII	1-byte field, interpreted as ASCII-character
WORD	2-byte field, B0 = LSB, B1 = MSB
INT3	3-byte field, B0 = LSB, B2 = MSB
DWORD	4-byte field, B0 = LSB, B3 = MSB
N <Typ>	Array, beginning with element index 0.
	Bit fields are stored bit 0 = 0x01, bit 7 = 0x80, bit 15 = 0x8000,
RFU	Reserved for future use, must be 0 to guarantee compatibility with future extensions.

The following aggregate types are used:

DATE	WORD	Dates are in MS-DOS Directory Format, i.e.: Bit 00-04 Day (1-31) Bit 05-08 Month (1-12) Bit 09-15 Year % 100 (0-99) All fields 0 denote a non applicable date (has been reset or reader without RTC)
TIME	WORD	Time is stored in MS-DOS Directory Format, i.e.: Bit 00-04 Second / 2 (0-31) Bit 05-10 Minute (0-63) Bit 11-15 Hour (1-23)
TYPE	ASCII	Protocol type, which differentiates the various kinds of transaction logs, each type is described below. Only bits 0 to 6 are considered, bit 7 is RFU.
PARAM0	INT3	Parameter 0. The meaning is dependant from the type and is described below.
PARAM1	WORD	Parameter 1. The meaning is dependant from the type and is described below.
PARAM2	INT3	Parameter 2. The meaning is dependant from the type and is described below.
PARAM3	WORD	Parameter 3. The meaning is dependant from the type and is described below..
FLAG	BYTE	Bit 0 – 2 classification 0 → configuration, class `C` 1 → system, class `S` 2 → application, class `A` 3 → error, class `E` 4-6 → RFU 7 → must be 0, RFU Bit 3 - 7 RFU

1.5. Data Format of Transaction Logs

The following field formats are used in the transaction log:

Typ	Previous Typ	Format	Description
0	Space		Parameter not used
1	'U'		User number with 8 digits, leading zeros „0“ will added
2	'0'		Decimal number without decimal point
3	'n'		Decimal number with decimal point and 2 decimal digits after decimal point
4	'3'	XX.YY.ZZ	3 bytes will be represented with 2 digits per part and with decimal point in between.
5	'4'		Decimal 4-digit, with leading zeros „0“.
6	'6'		Decimal 6-digit, with leading zeros „0“.
7	'B'	HH.LL	High.Low = High Byte, Low Byte from one word (e.g.: 01.56)
8	'P'	PRICELIST	Pricelist information in the format: X.Y.ZZ. Details are described at chapter „Pricelist Information“ on page 73.
9	'T'		Interpretation from two or three bytes as ASCII character set
10	'X'		Hexadecimal, without leading zeros.
11			RFU
12			RFU
13			RFU
14			RFU
15			RFU

2. Transaction Classes

The transactions (or “internal protocols” which is the internal code name for the transactions) of generation 6 readers are separated in transaction classes. The transaction class will be stored in the field “FLAG”.

The following transaction classes are available:

- Class `C` [Configuration](#)
- Class `S` [System](#)
- Class `A` [Application](#)
- Class `E` [Error](#)

Each transaction class uses the entire ASCII character set.

3. Transaction log Class 'C' (Configuration)

Type	Transaction log-Class 6C
A	
B	
C	
D	
E	
F	File download (9/2)
G	
H	
I	
J	
K	
L	
M	
N	VMC initializing sequence
O	
P	Initializing pricelist
Q	
R	Hardware reset
S	Session open/closed
T	Time/date set
U	Initializing U-KEY pricelist
V	
W	Selection-Category-Setup
X	
Y	
Z	
!	MDB peripheral data
#	MDB VMC Data
%	MDB Cashless Options
&	MDB peripheral config data
*	MDB change giver tube status
~	MDB command „Setup Max/Min Price“.
?	Accepted cash type (coins, bills) of MDB Peripheral (CG, BV)

3.1. Transaction details

3.1.1. Transaction log Type `F` (Class 6C)

FDTAG (Sys 9/1): File read out from archive on MEDIA.

Field	Format	Description
PARAM0	2	Type of the TAG: 1: Complete configuration without text file 2: Complete configuration with text file 3: only text file 4: Product-/Price configuration 5: Blocked/Allowed Users 0: in case of specific errors (PARAM3)
PARAM1	2	Configuration-version of the config on the Tag according MDSmanager 0, if PARAM0 = 5 or in case of specific errors (PARAM3)
PARAM2	RFU	RFU
PARAM3	A	Error-Code. See document "Error-Coding G6.doc" 0 = No error.

3.1.2. Transaction log Type `N` (Class 6C)

The information of MDB command "setup config" will be logged. The transaction log is written after receiving of every setup config – command.

Field	Format	Description
PARAM0	4 (XX.YY.ZZ)	XX: Y3 Byte of setup config. command: Number of columns on the display. YY: Y4 Byte of setup config. command: Number of lines on the display. ZZ: Y5 Byte of setup config. command: Display information 0: Numbers, capital letters. Spaces and decimal points. 1: The entire ASCII character set can be displayed. (siehe auch MDB Spezifikation, Setup Config Response)
PARAM1	2	MDB level of VMC
PARAM2	2	Init Sequence: Bit 0: ‚Setup Config Data‘ Bit 1: ‚Setup Max / Min Price‘ Bit 2: ‚Request ID‘ VMC Level 3 expended mode: Bit 3: ‚Enable Options‘ Bit 4: ‚Setup Max / Min Price‘ - 32 bit for Level 03+
PARAM3	RFU	RFU

3.1.3. Transaction log Type `P` (Class 6C)

Price list settings.

Field	Format	Description
PARAM0	2	Aktion: 1: Datei gelöscht 2: Ausschliessende Dateien vorhanden -> beide Löschen 3: RFU
PARAM1	8 PRICELIST (X.Y.ZZ)	PARAM0 = 1: Price list information of the file which caused "erase" PARAM0 = 2: Price list information of the 1st erased file.
PARAM2	8 PRICELIST (X.Y.ZZ)	PARAM0 = 1: Price list information of the erased file(s) PARAM0 = 2: Price list information of the 2nd erased file(s)
PARAM3	RFU	RFU

3.1.4. Transaction log Type `R` (Class 6C)

Reader startup

Field	Format	Description
PARAM0	XX.YY.ZZ	XX: Reset-type of the controller 0: Power Up Reset 1: Wake Up Rest 2: Watchdog Reset 3: Software Reset 4: User Reset (external reset at pin NRST) 5: Brownout Reset YY: HW-Address ZZ: 0 = Debug / 1 = Release
PARAM1	RFU	RFU
PARAM2	RFU	RFU
PARAM3	RFU	RFU

3.1.5. Transaction log Type `S` (Class 6C)

This transaction will be written, if a configuration session has been opened or closed.

Field	Format	Description
PARAM0	2	0: Session closed 1: Session opened
PARAM1	2	If PARAM0 = 0: 0 Restart Flag not set 1 Restart Flag set If PARAM0 = 1: „Transport-Layer“ 1 UHC 2 MDS 3 CAN 4 ETH 5 UKEY (FileDownload-Tag / Stat-Tag)
PARAM2	2	If PARAM0 = 0 0 ‚Configuration loaded but not initialized‘ Flag not set 1 ‚Configuration loaded but not initialized‘ Flag set If PARAM0 = 1: Authorization Bit 0: Maintenance (CMS) Bit 1: read file Bit 2: write file
PARAM3	RFU	RFU

3.1.6. Transaction log Type `T` (Class 6C)

Time/date has been set, respectively show if summer or wintertime.

Field	Format	Description
PARAM0	2	0: before modification 1: after successful modification of time/date. 2: Date/ time could not be set, cause outside of valid range (min. link-date / max. 31.12.2030) 3: Date/time could not be set, call cannot be earlier as actual date. 4: Date/time could not be set (RTC-error or RTC not available). 100: Just for information (used at startup) 101: released on start-up, if data invalid
PARAM1	7 HH.LL	HH: 0 = Date/time valid / 1 = Date/time/ invalid LL: 0 = Wintertime / 1 = Summertime
PARAM2	2	Call: (e_ModuleId – check common.h) 5: (MOD_ID_U-KEY): RTC set with SystemMEDIA 8: (MOD_ID_CCCC): time set over credit card server 9: (MOD_ID_VMC): RTC set from VMC 12: (MOD_ID_REMC): RTC set over DDCMP 13: (MOD_ID_CSES): RTC set with MDSmanager 15: (MOD_ID_SHELL): RTC modified over Shell 16: (MOD_ID_MSF): Summer/Wintertime automatically set
PARAM3	2	Synchronisation counter (starts at 1 on each power up)

3.1.7. Transaction log Type `U` (Class 6C)

Initialized U-KEY-pricelist

Field	Format	Description
PARAM0	8 PRICELIST (X.Y.ZZ)	Pricelist information from the initialized U-KEY-pricelist
PARAM1	8 PRICELIST (X.Y.ZZ)	Pricelist information from the initialized U-KEY-tariff-pricelist
PARAM2	RFU	RFU
PARAM3	RFU	RFU

3.1.8. Transaction log Type `W` (Class 6C)

Selection-category-pricelist-settings at reader startup.

Field	Format	Description
PARAM0	0	Value of parameter GEN_SelectionsCategories
PARAM1	0	Number of selections
PARAM2	0	Number of categories
PARAM3	0	Number of pricelists

3.1.9. Transaction log Type `I` (Class 6C)

MDB peripheral device data.

Field	Format	Description
PARAM0	9	MDB peripheral device serial # (Byte 1-3)
PARAM1	2	SW version
PARAM2	9	MDB peripheral device manufacturer code, e.g. MIC, TET
PARAM3	9	MDB Slave (RD, CG, BV)

3.1.10. Transaction log Type `#` (Class 6C)

MDB VMC data. The information of MDB Request ID is logged. This is written after receiving of a Request ID – command. (Visible under EVA-DTS at IC7)

Field	Format	Description
PARAM0	9	VMC serial # (Byte 6-8)
PARAM1	2	SW version
PARAM2	9	VMC manufacturer code, e.g. MIC, DIX, SIE, CAI, FAS, RHV
PARAM3	RFU	RFU

3.1.11. Transaction log Type `%` (Class 6C)

MDB cashless-device options.

Field	Format	Description
PARAM0	RFU	RFU
PARAM1	RFU	RFU
PARAM2	2	Option bits: Bit 0: Ready to pay back Bit 1: Ready for multivend Bit 2: Display available Bit 3: Ready for cash vend
PARAM3	9	MDB Slave (RD, CG, BV)

3.1.12. Transaction log Type `&` (Class 6C)

MDB slave-device ,Config info' data or other start-up information.

CG:

BV: Always two protocols in a row

RD: If parameter PER_MdbRdMaxNoRespTime not 0, two protocols in a row will be written:

1. Value of the reader
2. Value of parameter PER_MdbRdMaxNoRespTime.

CC: Beginning and end of the start-up

Field	Format	Description
PARAM0	2	CG: Optional Features BV: Stacker Capacity (1. Protocol) Stacker Level (2. Protocol in a row) Decimal 1-7 = Level / Decimal 8: 1 = Stacker Full RD: RFU CC: RFU
PARAM1	7 (HH.LL)	CG: HH: Max No Response Time LL: Level MDB Peripherie Changer BV: 1. Protocol in a row: HH: Max No Response Time LL: Level MDB peripheral Bill Validator BV: 2. Protocol in a row: HH: 0 LL: Escrow Capable RD: HH: Max No Response Time LL: Level MDB peripheral reader CC: LL: 1 = Beginn Startup LL: 2 = Startup complete
PARAM2	7 (HH.LL)	RD, CG, BV: HH: Scaling factor LL: Decimal point CC: RFU
PARAM3	9	Peripheral (RD, CG, BV, CC)

3.1.13. Transaction log Type `*` (Class 6C)

Informationen over tube fill level / tube configuration.

This transaction log is written at start up for each tube.

- After each tube coin this transaction log is written, in order that the fill level is immediately visible.
- If MdbCgTubeLevelSource = 1, the transaction log is also written during operation mode, if the fill level changes (the interval of request "tube status" is 10 seconds).
- Also written in „Tube is bad“ situation for the concerned channel. In this case only PARAM3.HH will be set; the rest is = 0.
- (Since version R02.00.01 no more fatal error. Tube will be inactivated and logged under 6C-?)

Field	Format	Description
PARAM0	2	Coin value (allocated value at the .chg file)
PARAM1	8 PRICELIST (X.Y.ZZ)	X: Image of PER_MdbCgTubeLevelSource 0: Fill level from reader 1: Fill level from changer Y: Manual Dispense (Image from .chg-File) Z: Actual fill level, which the reader is using [1]
PARAM2	7 (HH.LL)	HH: Min-Level LL: Max-Level
PARAM3	7 (HH.LL)	HH: Channel number LL: Fill level which the changer reported. This level will be taken from the reader, if PER_MdbCgTubeLevelSource = 1

3.1.14. Transaction log Type `~` (Class 6C)

Information over MDB command „Setup Max/Min Price“. After every command Setup Max/Min Prices this transaction log is written. At MDB level 3 and expanded currency code two transaction logs are written during the initialization. The first after receiving of „Setup Max/Min Price“ (Format Level 1-3, 16-bit max/min value). The 2nd after receiving of „Setup Max/Min Price“ (Format Level 3 Expanded Currency Code, 32-bit max/min value).

Field	Format	Description
PARAM0	2	Min Price
PARAM1	2	Init Sequence: Bit 0: ‚Setup Config Data‘ Bit 1: ‚Setup Max / Min Price‘ Bit 2: ‚Request ID‘ VMC Level 3 Expanded Currency Mode: Bit 3: ‚Enable Options‘ Bit 4: ‚Setup Max / Min Price‘ - 32 bit for Level 03+ As PARAM2 in transaction log 'N', class C
PARAM2	2	Max Price
PARAM3	2	Currency Code if MDB Level 3, else 0

3.1.15. Transaction log Type `?' (Class 6C)

The accepted types of cash (coins, notes) which are reported in the startup sequence.

Field	Format	Description
PARAM0	0	CG: Allocated value from .chg-file CA: Binary input at coin acceptor BV: Allocated value from bil-file
PARAM1	PRICELIST (X.Y.ZZ)	<p>CG</p> <p>X:</p> <ul style="list-style-type: none"> 1 = Coin applied Coin in tube 2 = Coin applied Coin not in tube 3 = Coin not applied [1] Coin in tube 4 = Coin not applied [1] Coin not in tube 5 = Coin not applied [2] Coin in tube 6 = Coin not applied [2] Coin not in tube [1]: Not applied, because not in .chg-file [2]: Not applied, tube configuration incorrect: <ul style="list-style-type: none"> If coin in tube min-/max level must be set. If coin not in tube min-/max level must be = 0. <p>Y:</p> <ul style="list-style-type: none"> 1 = Cash vend 2 = Revalue to TAG 3 = Cash vend & revalue to TAG <p>Z:</p> <p>Logic channel of coin</p> <p>CA</p> <p>X:</p> <ul style="list-style-type: none"> 0 = Coin 1 = Bill <p>Y:</p> <ul style="list-style-type: none"> 1 = Cash vend 2 = Revalue to TAG 3 = Cash vend & revalue to TAG <p>Z:</p> <p>Index at Matrix file (cam)</p> <p>BV</p> <p>X:</p> <ul style="list-style-type: none"> 1 = Bill applied 2 = Bill not applied, because not .bil file! <p>Y:</p> <ul style="list-style-type: none"> 1 = Cash vend 2 = Revalue to TAG 3 = Cash vend & revalue to TAG <p>Z:</p> <p>Logic channel of bill</p>
PARAM2	0	Value of cash-type in base unit
PARAM3	T	MDB slave (CG, BV) or parallel cash acceptor (CA)

4. Transaction log Class 'S' (System)

Type	Transaction log-Class 6S
A	New protocol for G5 Statistic Tag: 6S „A‘ (Activate IrDA)
B	
C	System media
D	Defragmentation of File System
d	MDS communication session open/close
E	
F	File system status / action
f	Flash: autopage rewrite (only Mmaster6)
G	Service media / fill function
H	MDB peripherals: activate / deactivate / parameter change
I	Initialization done
J	
K	
L **	Incomming coins during refill mode.
M	VMC change state
N	
n	Missing VMC ACK to MDB reader data
O	
o	No response from VMC over the transaction (timeout)
P	Power watch, PIN status changed
Q	
R	
r	External devices activated / started
S	Statistics audited
T	MICROtrans status
t	Results of FTP upload/download
U	
V	Application-version, Application-number
W	
X	Cash-box removed / inserted / statistics generated
Y	
Z	
%	Price changed

** This transaction is only written if parameter **MISC_ProtocolLevel is = 2** (extended transaction logs).

4.1. Transaction Log Details

4.1.1. Transaction log Type `A` (Class 6S)

If a Statistic-MEDIA has been presented (Header 2, function 0x80), the country code corresponds to the country code in the reader (Code / Ranges are stored at File „A“ in Archive on ext. Partition) and the MEDIA is not locked (serial number), the following transaction will be written.

Field	Format	Description
PARAM0	1	Serial number of MEDIA
PARAM1	RFU	RFU
PARAM2	RFU	RFU
PARAM3	RFU	RFU

4.1.2. Transaction log Type `C` (Class 6S)

If a diagnose-MEDIA has been presented (Header 2, function 0x01), the country code corresponds to the country code in the reader and the MEDIA is not locked (serial number), the following transaction will be written.

Field	Format	Description
PARAM0	1	Serial number of MEDIA
PARAM1	2	Diagnose function group (interpretation of the block type of log. block 0)
PARAM2	2	Diagnose function of the log. block 0
PARAM3	RFU	RFU

4.1.3. Transaction log Type `D` (Class 6S)

Defragmentation of the File System.

Always two transaction logs will be written. One before (1) and one after (2) the defragmentation.

Field	Format	Description
PARAM0	2	Transaction log 1: Free space before defragmentation. Transaction log 2: Free space after defragmentation.
PARAM1	2	Transaction log 1: Fragmentation-factor 0-255 Transaction log 2: Result: 0 No error. Function completed successfully 1 Files are open for reading or writing. Defragmentation was canceled 2 No Flash pages were available to use for defragmentation. Defragmentation was canceled
PARAM2	2	Transaction log 1: Procedur 1 Shell 2 FTS 3 Initialize 4 RFM – at OpenFileWrite, if no space 5 RFM – after CloseFile, if opened with "w" or "a" 6 RFM – after DeleteFile 7 RFM – But during saving of transaction log file. 8 CCCC – After writing of CreditCard.log 9 QRC – After writing of LockedCodes.2Dc Transaction log 2: RFU
PARAM3	9	Device (F: for Smart, S: for Master / MICROpay)

4.1.4. Transaction log Type `d` (Class 6S)

MDS-Communication-Session Open/Close

Field	Format	Description
PARAM0	2	0 Session closed 1 Session opened 2 Session Open refused, since the driver version of slave is newer than the driver version of master.
PARAM1	2	Transport-Medium: (If PARAM0 = 1, else 0) 4 On-Board Infrared 5 MDS-COM3, Cable 6 Infrared-Interface 7 Bluetooth-Interface
PARAM2	2	Baudrate used
PARAM3	RFU	RFU

4.1.5. Transaction log Type `E` (Class 6S)

Ethernet

Field	Format	Description
PARAM0	2	Action 1: State change
PARAM1	2	If PARAM0 = 1: 1: Cable connected, Wait DHCP 2: Cable connected, static config applied. 3: DHCP: Configuration applied. 4: DHCP timeout: static config applied. 5: Cable disconnected
PARAM2	RFU	
PARAM3	RFU	

4.1.6. Transaction log Type `F` (Class 6S)

File System status or action

Field	Format	Description
PARAM0	2	FileSystem- Status / Action 1 File System defragmentation 2 Format File System 3 A prior incomplete defragmentation has been recognized, during start-up! 4 A parameterfile (Application.par / System.par) has been created.
PARAM1	2	If PARAM0 = 1: 3 No error. Function completed successfully 4 Files are open for reading or writing. Defragmentation was canceled 5 No Flash pages were available to use for defragmentation. Defragmentation was canceled If PARAM0 = 2: 0 No error. 1 Error. If PARAM0 = 4: 1 File new written, since not existing 2 File new written, since at least one parameter was outside of the valid range. -> Default value set and file restored.
PARAM2	2	Procedur 10 Shell 11 FTS 12 Initialize 13 RFM 14 PROT (after rotation of the log file, if free space on AT71 < 80k / on AT70 < 20k). 15 PROT (after rotation of the log file, if fragmentation >100)
PARAM3	9	Device (F: for Smart, S: for Master / MICROPay) If PARAM0 = 4: „Ap“ = Application.par Sy“ = System.par

4.1.7. Transaction log Type `f` (Class 6S)

External flash (on MMaster6): After a several number of write cycles into the flash a „Auto-Page-Rewrite“ needs to be released.

Field	Format	Description
PARAM0	2	0: Auto page rewrite started Not 0: Auto page rewrite completed. PARAM0 = Flash Page number, from where it has been recently written
PARAM1	RFU	RFU
PARAM2	RFU	RFU
PARAM3	RFU	RFU

4.1.8. Transaction log Type `G` (Class 6S)

If a service-MEDIA has been presented (Header 2, function 0x10 and the MEDIA is not locked (serial number), the following transaction will be written.

Field	Format	Description
PARAM0	1	The serial number of the MEDIA
PARAM1	2	Diagnose function group (interpretation of the block type of log. block 0)
PARAM2	2	Service MEDIA function of the log. block 0
PARAM3	9	empty

4.1.9. Transaction log Type `H` (Class 6S)

Activating / Deactivating / Parameter changes MDB Peripheral devices.

Field	Format	Description
PARAM0	2	BV: 1: Stacker CG: 1: Exact-Change 2: Fill mode CC: 1: RegisterPos: License Status 2: Transaction: Report TrxRefPOS 3: Reversal
PARAM1	2	BV: If PARAM0 = 1 (stacker) 1: Stacker removed 2: Stacker inserted 3: Stacker full CG: If PARAM0 = 1 (Exact-Change) 0: Exact-Change is/became inactive 1: Exact-Change is/became active If PARAM0 = 2 (Fill mode) 1: Fill mode started 2: Fill mode completed. CC: If PARAM0 = 1 (RegisterPos: License Status) - RegisterRes(0 = reader new registered / 1 = already registered) If PARAM0 = 2 (Transaction: Report TrxRefPOS) - Amount If PARAM0 = 3 (Reversal) - 0 = OK, 100 = Too much retries, abort. Customer lost money!
PARAM2	2	CC: If PARAM0 = 1 (RegisterPos: License Status) - Number of available licenses for this merchant If PARAM0 = 2 (Transaction: Report TrxRefPOS) - TrxRefPOS (Transactioncounter at the FRAM) If PARAM0 = 3 (Reversal) - TrxRefPOS (Transactioncounter at the FRAM)
PARAM3	9	MDB Peripheral device (CG, BV, RD).

4.1.10. Transaction log Type `I` (Class 6S)

If one initialization has been released the transaction "I" will be written.

Field	Format	Description
PARAM0	2	Caused source 1 System-MEDIA (1/2 or 9/1 ...) 2 CMS 3 VMC (cause of VMC type changed) 4 Activated with test button 5 Configuration update (FTP / MDD / File DL Tag)
PARAM1	2	Options (what has been initialized?)
PARAM2	6	Interrogator ID First 6 decimals
PARAM3	5	Interrogator ID Last 4 decimals

4.1.11. Transaction log Type `L` (Class 6S)

Incomming coins during refill mode. This transaction is only written if parameter **MISC_ProtocolLevel 2** is activated.

Field	Format	Description
PARAM0	1	Decimal point 1-7: Cash-value of all Tubes Decimal point 8: Exact-Change (1 = Active / 0 = Inactive)
PARAM1	7 (HH.LL)	HH: Routing: 0= Cashbox / 1= Tube LL: Channel
PARAM2	2	Coin value
PARAM3	7 (HH.LL)	HH: Tube level referred to reader LL: Tube level referred to changer

4.1.12. Transaction log Type `M` (Class 6S)

State machine changes.

Field	Format	Description
PARAM0	2	1 MDB-VMC
PARAM1	2	1
	2	If PARAM0 = 2 (MDB-VMC) State machine: 1 VMC does not respond or does not poll (after timeout) – always written. The most important state changes will be logged: INACTIVE->DISABLED, DISABLED -> INACTIVE DISABLED->ENABLED, ENABLED -> DISABLED 2 MDB 'Command out of sequence' 4 MDB 'Just Reset'
PARAM2	7 (HH.LL)	State changes: 1 Inactive 2 Disabled 3 Enabled HH = old state LL = new state
PARAM3	RFU	RFU

4.1.13. Transaction log Type `n` (Class 6S)

Missing VMC ACK to MDB reader data.

Field	Format	Description
PARAM0	2	MDB Reader data (Byte 0)
PARAM1	2	VMC command which has been received instead of ACK
PARAM2	2	VMC Sub-command
PARAM3	2	Number of repeated MDB reader command (depending on VMC_MdbTransmitResp).

4.1.14. Transaction log Type `o` (Class 6S)

No response about the vend transaction has been sent from VMC after a timeout of 90sec.

U-KEY / MEDIA

Field	Format	Description
PARAM0	1	The user number from Header 2
PARAM1	2	Product price (base rate)
PARAM2	2	Selection number
PARAM3	8 PRICELIST (X.Y.ZZ)	Pricelist-information

Credit Card

Field	Format	Description
PARAM0	USER	The last 7 digits of the credit card number
PARAM1	2	Product price (base rate)
PARAM2	2	Selection number
PARAM3	8 PRICELIST (X.Y.ZZ)	Pricelist-information

Cash

Field	Format	Description
PARAM0	USER	0
PARAM1	2	Product price (base rate)
PARAM2	2	Selection number
PARAM3	8 PRICELIST (X.Y.ZZ)	Pricelist-information

4.1.15. Transaction log Type `P` (Class 6S)

If the voltage fell under 12V or increased over 15V a protocol is written.

Field	Format	Description
PARAM0	0	0: voltage fell under 12V 1: voltage increased over 15V 2: voltage control turned off, because of too many variation within a short time. The detection will be activated only after a restart. Specifically: If within 1 second four PowerWatch interrupts has been recognized, the Power Watch-IRQ will be disabled. 3: MDD6 powered by battery 4: MDD6 powered over USB
PARAM1	RFU	RFU
PARAM2	RFU	RFU
PARAM3	RFU	RFU

4.1.16. Transaction log Type `r` (Class 6S)

Activating / Start-Up / Deactivating external devices.

MDB slave reader: Reset command from VMC.

Field	Format	Description
PARAM0	2	Peripherals: 20: EXE-Change-Giver activated 21: EXE-Change-Giver inactivated 22: EXE-Change-Giver: Parameter change (limit cash => PARAM1) MDS-Connector: 60: SAM-Interface: Calypso-SAM initialized
PARAM1	2	PARAM0 = 20/22: (EXE-Change Giver) Cash-Limit PARAM0 = 60: (Calypso-SAM) Higher Word of SAM-ID
PARAM2	7 (HH.LL)	PARAM0 = 20/22: (EXE-Change Giver) Parameter value: HH=VND_AllowCash / LL=VND_AllowRevalue PARAM0 = 60: HH: unlocked: 0=success; 1=not necessary; 2=wrong key; 3=Other error LL: SAM slot: 1=Slot #1 / 2=Slot #2
PARAM3	2	PARAM0 = 60: (Calypso-SAM) Lower Word of SAM-ID

4.1.17. Transaction log Type `S` (Class 6S)

Statistic has been read out.

Field	Format	Description
PARAM0	7 (HH.LL)	HH: Activate source: 1: Stat-MEDIA (9/20) 2: FTS 3: DDCMP 4: DEX 5: dts-file generate by cashbox switch 6: System-MEDIA (1/2 or 9/1 ...) 7: CMS 8: Activated with the button 16: Statistic has been readout and stored in a backup file due to changes at the reader prices (Sys-MEDIA 9/20). 17: The cash box has been removed and the statistic has been stored („ymmddcc.dts“). 18: Statistic has been readout and stored in a backup file due to changing of the VMC type. LL: If HH=3 or HH=4 (DDCMP / DEX): (LL else =0) 4: On-Board Infrared 5: MDS-COM3, Cable 6: Infrared-Interface 7: Bluetooth-Interface
PARAM1	2	Number of readouts (EA301)
PARAM2	7 (HH.LL)	HH: 0 – 3 = Selected detail-level LL: 0 = Statistic not erase 1 = Mark statistic for erasing (daily statistic) 2 = Statistics have been initialized
PARAM3	RFU	RFU

4.1.18. Transaction log Type `T` (Class 6S)

Results associated with MICROtrans TCP-Layer

Field	Format	Description
PARAM0	0	1 = Socket allocated (MICT_PROT_SOCKET_ALLOCATE) 2 = Socket open active or passive (MICT_PROT_SOCKET_OPEN) 3 = Socket connected (MICT_PROT_SOCKET_CONNECTED) 4 = Socket timeout from LIB (MICT_PROT_SOCKET_TIMEOUT) 5 = Socket have been closed (MICT_PROT_SOCKET_CLOSED) 6 = MICROtrans time, received nothing more (MICT_PROT_TIMEOUT) 7 = MICROtrans is activated (MICT_PROT_ACTIVATED) 8 = MICROtrans received „activate“ with wrong version (MICT_PROT_VERSION_MISMATCH)
PARAM1	0	MICT_PROT_SOCKET_OPEN) -> 0 = listen (passive), 1 = connect (active) MICT_PROT_SOCKET_CLOSED) ->0 = Close Event, 1 = Abort Event
PARAM2	0	State of state machine 1 = STATE_NOT_CONNECTED 2 = STATE_WAIT_INITIAL, 3 = STATE_WAIT_ACTIVATE, 4 = STATE_ACTIVE, 5 = STATE_PROC_MESSAGE, 6 = STATE_SEND_MESSAGE,
PARAM3	RFU	RFU

4.1.19. Transaction log Type `t` (Class 6S)

Results of FTP-upload / download.

Field	Format	Description
PARAM0	9	File Type according name convention. „dts“ for EVA-DTS statistic file „pco“ for binary protocol file.File „evt“ for Eventfile (DTS Format) „evx“ for Eventfile (not conform to DTS) „mdu“ for Configuration file (Archive) „ack“ for Acknowledge File (FTP Update confirmation) „bin“ für Binary / Firmware File
PARAM1	7 (HH.LL)	HH: Module Number LL: Upload status Module Number: 25 = DCMF 28 = FTPC 42 = FTPM Upload status: 2 = Upload successful including renaming filename “Mode” from „D“to „Q“. 1 = Upload successful, renaming filename failed. (Error code at PARAM3) 0 = Upload failed (Error code at PARAM3) Downloadstatus: 16 = File search failed (error code in PARAM3) 17 = File Download failed (error code in PARAM3) 18 = File Update failed (error code in PARAM3) 19 = Update successful Common Status (from version 2.4 on) 32 = General Error
PARAM2	2	File extension: For „dts“: YYMMTTCC (CC → Daily counter) For „pco“: the File Number extracted from file name For „evt“: time extracted from file name (HHMMSS) For „evx“: extracted from file name (HHMMSS) For „mdu“: Version extracted from file name (0, if not known– e.g. if Download status 16)

PARAM3	7 (HH.LL)	<p>Error code, if PARAM1 < 2 High Byte: Error group Low Byte: Error Number</p> <p>FTPM: 01.01 = Manager's Task Queue is full 01.02 = File was not found on Reader 01.03 = File could not be renamed after upload 01.04 = Ethernet disconnected 01.05 = Configuration File missing / invalid 01.06 = Reader Update failed</p> <p>FTPC: 02.01 = Ethernet disconnected 02.02 = Invalid Parameters 02.03 = Existing (Duplicated) Task 02.04 = Task timed out 02.05 = DNS Lookup failed 02.06 = Hash / Encryption failed 02.07 = Could not create missing Directory on Server 02.08 = FTP : Timeout (KEIL) 02.09 = FTP : Login failed (KEIL) 02.10 = FTP : Access denied (KEIL) 02.11 = FTP : Remote File not found (KEIL) 02.12 = FTP : Remote Directory not found (KEIL) 02.13 = FTP : Could not open local File (KEIL) 02.14 = FTP : Generic Error (KEIL) 02.15 = Unknown Error</p>
--------	--------------	--

4.1.20. Transaction log Type `V` (Class 6S)

Application-Version, Applications-Number (After each start up, transaction log "6C" `R`)

Field	Format	Description
PARAM0	4 (XX.YY.ZZ)	Major.Minor.Bugfix
PARAM1	9	Application ("AM", "TM", "CM")
PARAM2	9	Application Number ("008")
PARAM3	2	Revision

4.1.21. Transaction log Type `X` (Class 6S)

Events associated with cash box.

Field	Format	Description
PARAM0	2	1: Cash-Box removed 2: Cash-Box inserted 3: Statistic-file successful created and stored 4: Failed to generate the statistic 5: File name<->conflict of date. (found .dts file with newer date, as the own reader date). -> *.dts file has been deleted.
PARAM1	2	If PARAM0 = 4: 1: Statistic is already reading out over another channel 2: Statistic is empty (Clear-flag is activated and STA_ClearFlagMode is set accordingly) 3: Not enough memory to store the statistic file 4: Error creating temporary file 5: Error renaming temp.dts -> yymmddcc.dts
PARAM2	2	If Param0 = 3: Number of the created file name without leading 0 and without („.dts“). Example: Created file „09120701.dts“ will be stored as „9120701“.
PARAM3	RFU	RFU

4.1.22. Transaction log Type `%` (Class 6S)

Price change at VMC prices.

Field	Format	Description
PARAM0	8 PRICELIST (X.Y.ZZ)	Price list
PARAM1	2	Selection Number
PARAM2	2	Old Price
PARAM3	2	New Price

5. Transaction log Class 'A' (Application)

Typ	Lib	A008 A108	A002	A003	A046	C000	A080	Transaction log-Class 6A
A		X						Application state changed from / to DISABLE / NO VMC / INHIBIT
a	X							Pay-out over MDB change giver. Automatic / Manually.
B *		X		X	X			A008: Successful Cash vend A003: Creating/Removing of a VEND Block. A046: SubCode-after Balance Transaction log
C	X							Access-check error during read in of a MEDIA.
D		X						A008: Dispense Cancel
d		X						A008: Transaction cancelled. Not enough credit
e	X							Token loaded from the Master-Token balance to the reader.
F *		X		X		X		A008: Free vend without media A003: Formatting/un-formatting of a Tag
f	X							Revalue denied due to wrong conditions
G		X [†]						Successful vend with credit card.
g	X							Pressing the payback lever at the MDB changer or coin acceptor.
H								Information concerning Happy-Meals.
h *	X							Successful bonus generation – paid out as credit to the vend block
i *		X						Successful bonus vend
i *	X							Successful bonus generation – paid out as bonus point
J *		X						Successful token vend
J *	X							Successful token revaluation
K *		X						Successful vend over cashless reader
k		X						Vend amount from Cashless Reader
L *		X	X	X	X			A008: Cash credit loaded successfully to e-purse A002: Cash credit loaded successfully to e-purse A003: Format license received A046: Credit loaded successfully to e-purse
l *		X	X		X			A008: Loading cash credit to e-purse: status is uncertain A002: Loading cash credit to e-purse: status is uncertain A046: Loading credit to e-purse: status is uncertain
M *	X							Successful distribution of loading reward in the form of turnover
m *	X							Successful distribution of loading reward in the form of money
n	X							Successful loading of credit over contribution
O		X						Vend failed
P	X							Credit received from MDB cashless reader (peripheral)
p	X							Incorrect signal detected at coin acceptor input of Mmaster6
Q *	X							Deleted cash credit value after timeout
q		X						Wrong conditions detected at „Revalue Limit Request
R *		X	X		X			A008: Successful loading of credit on Cashless Reader (MDB Peripheral) A002: Successful loading of credit on Cashless Reader (MDB Peripheral) A046: Reader locked or unlocked
r		X	X		X		X	A008: Failed to load credit on Cashless Reader (MDB Peripheral) A002: Failed to load credit on Cashless Reader (MDB Peripheral) A080: Statistic, transaction log or the support-package read out and stored on MDD A046: The reader or the card has been reset (Reset-Tag)
S	X							Showing e-purse credit after authorization
T *		X			X			A008: The transaction has been performed successfully A003: MIFARE PLUS SL1 -> SL3 migrated A046: User card locked or unlocked
t	X							History of Happy-Meals
V		X						A008: Vend success with cashless media
V **		X						A008: Product selection received from VMC
W		X						A008: Successful vend with 2D-Barcode Voucher
X	X							User media has been locked
Y	X							Cash value received from peripheral device
y	X							Successful cancelation after Vend Failure
~		X						A008: Authorized service media, balance from Test-TAG A003: Format credits transferred to Format units

[†]) This transaction is only written if parameter **MISC_ProtocolLevel 1** = 1 (extended transaction logs). . This feature must be activated over "direct reader access".

******) This transaction is only written if parameter **MISC_ProtocolLevel 2** = 2(extended transaction logs). This feature must be activated over "direct reader access".

5.1. Transaction Log Details

5.1.1. Transaction log Type `A` (Class 6A)

State changes of the application from or to STARTUP / INHIBIT / NO VMC will be recorded in order to visualize how long e.g. a VMC had set the reader to INHIBIT. Also it will be recorded when and how long there was no communication.

Field	Format	Description
PARAM0	2	Actual state 1 STARTUP 2 IDLE 3 FUND 4 NO FUND 5 DISPENSE 6 NO VMC 7 INHIBIT 8 EXCEPTION
PARAM1	2	Next state (same list as above)
PARAM2	9	VMC-type: „?“: „Autodetect“ active, Type not yet identified „MDB“ „E?B“: EXE/BDV (not yet identified which one) „EXE“ „BDV“
PARAM3	2	Additional info: 0: No info 1: Wrong VMC-type (only with PARAM0=6)

5.1.2. Transaction log Type `a` (Class 6A)

Recordings of payed out cash amount. This can be:

- Automatic after vend
- After vend-Failure
- Manual over payback lever
- Manual-Dispense-button

Field	Format	Description
PARAM0	2	1: Pay-out started 2: Pay-out finished 3: Manual Dispense
PARAM1	2	If PARAM0 = 1: Amount to be paid out If PARAM0 = 3: Value of manually paid out coin Else 0
PARAM2	2	If PARAM0 = 2/3: Paid out amount Else 0
PARAM3	2	If PARAM0 = 2: Rest amount If PARAM0 = 3: Rest level of the Tube Else 0

5.1.3. Transaction log Type `B` (Class 6A)

Used at A008

Successful Cash-Vend

Field	Format	Description
PARAM0	RFU	RFU
PARAM1	2	Product price (base rate)
PARAM2	2	The remaining amount after cash sale
PARAM3	8 PRICELIST (X.Y.ZZ)	Price list Information

Used at A003

A003 creates this transaction log:

- Creating of a VEND Block.
- Removing of a VEND Block.

Field	Format	Description
PARAM0	1	The user number from Header 2
PARAM1	7 (HH.LL)	HH: Number of the Vend block (1 or 2) LL: Block type before Transaction (1 or 2, 0 = created)
PARAM2	2	Initial- respectively last balance
PARAM3	7 (HH.LL)	HH: Number of the Vend block (1 or 2) LL: Block type after the Transaction (1 or 2, 0 = removed)

Used at A046

Transaction log of Sub-Code

Field	Format	Description
PARAM0	1	The user number from Header 2
PARAM1	RFU	RFU
PARAM2	2	Sub-code from the TAG (log. Block 19)
PARAM3	RFU	RFU

5.1.4. Transaction log Type `C` (Class 6A)

Access-check error during read in of a MEDIA.

PARAM2 und PARAM3 contain the same value (Extended Error Information), just formatted else:

The „Extended Error Information“ is depending on the Access-Check error an additional information: e.g. if „wrong country code“ the „Extended Error Information“ contain the country code of the MEDIA; if „block x.x locked“ the „Extended Error Information“ contain the concerned block number (the image of PARAM3 shows directly the concerned block).

Field	Format	Description
PARAM0	1	User Number
PARAM1	7 (HH.LL)	Number of the triggered error: MS-Byte: Error group LS-Byte: Error detail (Example: the hexadecimal value 0x0F02, will be interpreted as error group 15 with error detail 02)
PARAM2	2	Extended Error Information (2 Bytes) shown as simple number (e.g. for wrong country code, wrong company code...)
PARAM3	7 (HH.LL)	Extended Error Information (2 Bytes) shown as HH.LL (if the error is related to a block, then it appears directly the block number)

5.1.5. Transaction log Type `D` (Class 6A)

After the VMC has sent a „vend request” to the reader and right away a „vend cancel”, this will produce a „dispense cancel” at the application.

U-KEY / MEDIA

Field	Format	Description
PARAM0	1	The User number from Header 2
PARAM1	2	Price of product (basic unit)
PARAM2	2	Selection number
PARAM3	8 PRICELIST (X.Y.ZZ)	Price list information

Credit Card

Field	Format	Description
PARAM0	1	The last seven digits of the credit card number
PARAM1	2	Price of product (basic unit)
PARAM2	2	Selection number
PARAM3	8 PRICELIST (X.Y.ZZ)	Price list information

Cash

Field	Format	Description
PARAM0	1	0
PARAM1	2	Price of product (basic unit)
PARAM2	2	Selection number
PARAM3	8 PRICELIST (X.Y.ZZ)	Price list information

5.1.6. Transaction log Type `d` (Class 6A)

If the U-KEY, cash credit or the pre authorized value of the credit card is lower than the selection price, the „vend request” of the VMC cannot performed. The reader sends a „vend denied”.

U-KEY / MEDIA

Field	Format	Description
PARAM0	1	The User number from Header 2
PARAM1	2	Price of product (basic unit)
PARAM2	2	Selection number
PARAM3	8 PRICELIST (X.Y.ZZ)	Price list information

Credit Card

Field	Format	Description
PARAM0	1	The last seven digits of the credit card number
PARAM1	2	Price of product (basic unit)
PARAM2	2	Selection number
PARAM3	8 PRICELIST (X.Y.ZZ)	Price list information

Cash

Field	Format	Description
PARAM0	1	0
PARAM1	2	Price of product (basic unit)
PARAM2	2	Selection number
PARAM3	8 PRICELIST (X.Y.ZZ)	Price list information

5.1.7. Transaction log Type `e` (Class 6A)

Number of loaded tokens from the Master-Token balance to the reader.

The transfer to the reader has been made with the Master Transfer MEDIA (Diag. function 513).

Field	Format	Description
PARAM0	0	Reader serial no.
PARAM1	RFU	
PARAM2	0	Number of loaded tokens to the reader.
PARAM3	RFU	RFU

5.1.8. Transaction log Type `F` (Class 6A)

Used at A008

Free vend without MEDIA. Released either directly from VMC (EXE) if VMC-prices and with activated free-vend Flag (0x80) or, if reader prices and price = 0 on certain products.

Field	Format	Description
PARAM0	2	0: Virtual credit from VMC, which allows vend without MEDIA. 1: Free vend without MEDIA completed successful
PARAM1	2	If PARAM0 = 0 (before vend) VMC-Price of product If PARAM0 = 1 (after vend) Reader price of product (U-KEY Cat. 1), if set, otherwise VMC-price
PARAM2	2	If PARAM0 = 0 (before vend) Selection no. If PARAM0 = 1 (after vend) Balance of „virtual VMC credit“ (always = 0)
PARAM3	8 PRICELIST (X.Y.ZZ)	Price list information

Used at A003

A003 creates this transaction log at:

- Successful formatting of a Tag
- Successful un-formatting of a Tag

Field	Format	Description
PARAM0	1	The User number from Header 2
PARAM1	7 (HH.LL)	HH: License price in Units LL: Max logical blocks (de-formatted: 0), Bit7: with Ext Part
PARAM2	3	Bit 0-23 of Tag UID
PARAM3	7 (HH.LL)	HH: Tag type LL: Tag size in K (nominal)

5.1.9. Transaction log Type `f` (Class 6A)

„Revalue denied“ due to wrong conditions for the revaluation.

Field	Format	Description
PARAM0	1	The User number from Header 2
PARAM1	2	Amount in basic unit, which the system would revalue
PARAM2	3	Actual balance (without the amount to be loaded)
PARAM3	2	<p>General:</p> <p>01: Tried to revalue not onto media or MDB-Reader (Peripheral) but to different destination.</p> <p>07: VMC_MdbAllowRevalue = false</p> <p>Revalue to UKey – error source:</p> <p>02: Parameter UKEY_AllowRevalue is false.</p> <p>03: The total amount would exceed the limit in Par. LIM_LockBalance.</p> <p>04: The amount is higher as the limit in Par. LIM_LockBalance □ hence it can be verified if the cash amount at the master is higher as the par. LIM_LockBalance.</p> <p>05: Not possible to load to a NFC mobile phone.</p> <p>06: No changes written to the TAG (e.g TAG is no longer in the field, or VEND1 is read-only, ...)</p> <p>Balance-Transfer – Error source:</p> <p>10: Parameter PER_MdbRdBalanceTransfer is 0 (disabled)</p> <p>11: The limit LIM_LockBalance is already achieved.</p> <p>12: Not possible to load to a NFC mobile phone.</p> <p>13: No changes written to the TAG (e.g TAG read-only)</p> <p>14: Debiting not allowed on MDB-Reader (Revalue Denied 0E)</p> <p>Revalue to MDBRD – Error source:</p> <p>20: Parameter PER_MdbRdAllowRevalue is 0 (disabled)</p>

5.1.10. Transaction log Type `G` (Class 6A)

Used at bei A108

Successful vend with credit card. The VMC could successfully dispense the product (VMC Dispense OK).

Field	Format	Description
PARAM0	1	Last 7 digits from the credit card
PARAM1	2	Price of product (basic unit)
PARAM2	RFU	
PARAM3	8 PRICELIST (X.Y.ZZ)	Price list Information

5.1.11. Transaction log Type `g` (Class 6A)

Pressing the payback lever at the MDB changer or coin acceptor.

Field	Format	Description
PARAM0	2	1: Regular; payback lever pressed CG: 1000: Logging has been stopped, because within 60s more than 30 transaction logs should be stored. Only after a CG restart it will again record further transaction logs.
PARAM1	RFU	RFU
PARAM2	RFU	RFU
PARAM3	9	MDB slave (CG) or parallel cash acceptor (CA)

5.1.12. Transaction log Type `H` (Class 6A)

Information concerning Happy-Meals.

This transaction log is written before the 6A-`T` when a product has been purchased which is involved in a Happy-Meals (Cat. 1 -4).

If an entire menu is purchased the transaction log 6A-`t` which reports reference and price of all involved products.

UKey-Tag

Field	Format	Description
PARAM0	7 (HH.LL)	HH: 1 The 1st. product of a Happy-Meals has been purchased. A new period starts. All data at HMD1/HMD2 erased. 2 Purchase of a Happy-Meals complete. Refund to the VEND block. 3 Purchase of a Happy-Meals complete. Refund to the TOKEN block. 4 History no longer traced since limitation achieved. 5 History no longer traced since no time window for Happy-Meals actif. 6 Error: At this Badge block HMD1/HMD2 is missing. 7 Error: At this Badge block TOKEN is missing. LL: • If HH = 2 or 3: (Meal complete): Menu Index (1..16) • Else counter from HDM1 (only relevant, if purchase limited)
PARAM1	2	Menu price (If PARAM 0 = 2 or 3)
PARAM2	2	Amount, if refund on VEND block (if PARAM 0 = 2)
PARAM3	2	Total price of all products at the menu

5.1.13. Transaction log Type `h` (Class 6A)

Successful bonus generation – paid out as credit to the vend block #1 or #2.

U-KEY / MEDIA

Field	Format	Description
PARAM0	1	The User number from Header 2
PARAM1	2	Value paid out (basic unit money)
PARAM2	2	Current amount on MEDIA
PARAM3	RFU	RFU

5.1.14. Transaction log Type `I` (Class 6A)

Successful bonus vend – VMC could issue the product successfully (from VMC Dispense OK).

Field	Format	Description
PARAM0	1	The User number from Header 2
PARAM1	2	Bonus Price of product (basic unit)
PARAM2	2	Current bonus credit on the MEDIA after the vend
PARAM3	8 PRICELIST (X.Y.ZZ)	Price list information

5.1.15. Transaction log Type `i` (Class 6A)

Successful bonus generation – paid out as bonus point

Field	Format	Description
PARAM0	1	The User number from Header 2
PARAM1	2	Value paid out (basic unit bonus points)
PARAM2	2	Current bonus points on the MEDIA
PARAM3	RFU	RFU

5.1.16. Transaction log Type `J` (Class 6A)

Successful token vend – VMC could issue the product successfully (from VMC Dispense OK).

Token are either on the MEDIA, or directly on the reader (in an internal account). The difference is visible at the pricelist information.

Field	Format	Description
PARAM0	1	The User number from Header 2
PARAM1	2	Token price of product (By reader token always 1, otherwise depending on token price list)
PARAM2	2	The current token balance on MEDIA or in the reader after vend
PARAM3	8 PRICELIST (X.Y.ZZ)	Price list information

5.1.17. Transaction log Type `j` (Class 6A)

Successful token revaluation.

Field	Format	Description
PARAM0	1	The user number from Header 2
PARAM1	2	Number of loaded tokens
PARAM2	2	Current token credit on MEDIA after reloading token
PARAM3	2	Source: 4: Contribution 7: Token loaded System-Tag „3/515 Token value“ on user-Tag 8: Token loaded System-Tag „3/517 Debit Token“ on user-Tag

5.1.18. Transaction log Type `K` (Class 6A)

Successful vend over cashless reader (MDB peripheral) – VMC could issue the product successfully (from VMC Dispense OK).

Field	Format	Description
PARAM0	1	„Media-ID“ from cashless reader
PARAM1	2	Price of product (basic unit)
PARAM2	2	Current credit on MEDIA after vend
PARAM3	8 PRICELIST (X.Y.ZZ)	Price list information

5.1.19. Transaction log Type `k` (Class 6A)

The Cashless Reader (MDB-Peripheral) transferred at ‚Vend Approved‘ another price as at „Vend Request“. The MICROTRONIC System always uses the transferred price.

Field	Format	Description
PARAM0	RFU	RFU
PARAM1	2	Price of product (basic unit)
PARAM2	2	Received vend amount from Cashless Reader (basic unit)
PARAM3	RFU	RFU

5.1.20. Transaction log Type `L` (Class 6A)

Used at A008, A002 und A046

After successful loading, the loaded cash amount and the actual credit of the Tag will be logged.

Field	Format	Description
PARAM0	1	The User number from Header 2
PARAM1	2	Loaded cash credit in basic unit
PARAM2	3	Current credit on the MEDIA after loading
PARAM3	2	Source: 1: Over VMC-interface (MDB-Master send Revalue Request) 2: Amount over Peripheral (EXE, MDB, PCA) 5: Balance-Transfer (from MDB-Reader to UKey) 6: Open Transaction: The Tag was no longer present when the amount has been sent to the reader for revalue. Now the Tag is again in the antenna field. 7: Amount from System-Tag „3/514 Money value“ loaded to User-Tag 8: Amount from System-Tag „3/516 Debit Cash“ loaded to User-Tag

Used at A003

A003 creates this transaction log if:

- Realising of Format license Units over realisation Code (over E-Mail, Code 066).
- Transferring of licenses from reader -> Tag (Diagnose 7/66 or 7/68).
- Transferring of licenses from Tag -> reader (Diagnose 8/66 or 8/68).

Field	Format	Description
PARAM0	1	Bit 0-23 of Tag UID or Interrogator ID: 3 (M-Protocol)
PARAM1	3	766 or 866 if License Tag, 66 if Code until M-Protocol v125 768 or 868 if License Tag, 68 if Code since M-Protocol v126
PARAM2	3	The new license credit stock in units
PARAM3	3	Number of license units at this transaction

5.1.21. Transaction log Type `I` (Class 6A)

It is not possible to detect if the amount could be loaded to the MEDIA or not, if the media has been removed during the „Revalue“ procedure. In order to prevent violation in this situation, the command REVALUE_APPROVED will be however sent.

Field	Format	Description
PARAM0	1	The User number from Header 2
PARAM1	2	Loaded cash credit in basic unit
PARAM2	3	Current credit on the MEDIA after loading
PARAM3	2	Loading source: 1: Over VMC-Interface (MDB-Master send Revalue Request) 2: Change giver to Peripheral (EXE or MDB) 4: Contribution 5: Balance-Transfer (from MDB-Reader to UKey) 6: Open Transaction: The Tag was no longer present when the amount has been sent to the reader for revalue. Now the Tag is again in the antenna field. 7: Amount from System-Tag „3/514 Money value“ loaded to User-Tag 7: Amount from System-Tag „3/514 Money value“ loaded to User-Tag 8: Amount from System-Tag „3/516 Debit Cash“ loaded to User-Tag

5.1.22. Transaction log Type `M` (Class 6A)

Successful issue of a loading reward in the form of turnover. The value will be credited at the turnover account on the bonus block.

Field	Format	Description
PARAM0	1	The User number from Header 2
PARAM1	2	Amount of the loading reward: <i>Turnover</i> or <i>number of vends</i> , which will be added to the bonus-turnover counter respectively to the vend counter.
PARAM2	2	The actual amount of the bonus-turnover counter respectively from the vend counter after the reward has been added.
PARAM3	7 (XX.YZ)	<p>At revaluation it is possible to reach several levels for which loading rewards will be loaded. PARAM3 shows the first and the last level, as well as the type of a issued loading reward.</p> <p>XX: Loading reward type: 2 Turnover, 3 Number of vend Y: First level, at which is has been credited. Z: Last level, at which is has been credited.</p> <p><i>Example 1.</i> Reader settings: Loading reward of 1.00 Turnover @ 2.00 and loading reward of 1.20 @ 3.00:</p> <p>Revaluation of 5.00 -> PARAM1 = 1.00 + 1.20 = 2.20 -> PARAM2 = Bonus-turnover counter + 2.20 -> PARAM2 = 02.12 (Level 1 to 2 reached)</p> <p><i>Example 2:</i> Reader settings: Loading reward at 1 vend @ 2.00 and loading reward if 2 vends @ 3.00:</p> <p>Revaluation of 5.00 -> PARAM1 = 1 + 2 = 3 -> PARAM2 = Vend counter + 3 -> PARAM2 = 03.12 (Level 1 to 2 reached)</p>

5.1.23. Transaction log Type `m` (Class 6A)

Successful issue of a loading reward as credit to the vend block.

Field	Format	Description
PARAM0	1	The User number from Header 2
PARAM1	2	Loading rewards which has been credited: Amount added to the vend block.
PARAM2	2	The actual credit at the vend block after the loading reward has been added.
PARAM3	7 (XX.YZ)	<p>At revaluation it is possible to reach several levels for which loading rewards will be loaded. PARAM3 shows the first and the last level, as well as the type of a issued loading reward.</p> <p>XX: Loading reward type: 2 Turnover, 3 Number of vend Y: First level, at which is has been credited. Z: Last level, at which is has been credited.</p> <p><i>Example 1.</i> Reader settings: Loading reward of 1.00 @ 2.00 and loading reward of 1.20 @ 3.00:</p> <p>Revaluation of 5.00 -> PARAM1 = 1.00 + 1.20 = 2.20 -> PARAM2 = Actual credit + 2.20 -> PARAM2 = 01.12 (Level 1 to 2 reached)</p>

5.1.24. Transaction log Type `n` (Class 6A)

Report of the loaded amount over contribution.

Field	Format	Description
PARAM0	1	The User number from Header 2
PARAM1	2	Loaded amount
PARAM2	3	Current credit on media
PARAM3	RFU	RFU

5.1.25. Transaction log Type `O` (Class 6A)

The application receives a „dispense failed” from VMC.

U-KEY / MEDIA

Field	Format	Description
PARAM0	1	The User number from Header 2
PARAM1	2	Price of product (basic unit)
PARAM2	2	Selection number
PARAM3	8 PRICELIST (X.Y.ZZ)	Price list information

Credit Card

Field	Format	Description
PARAM0	1	The last seven digits of the credit card number
PARAM1	2	Price of product (basic unit)
PARAM2	2	Selection number
PARAM3	8 PRICELIST (X.Y.ZZ)	Price list information

Cash

Field	Format	Description
PARAM0	1	0
PARAM1	2	Price of product (basic unit)
PARAM2	2	Selection number
PARAM3	8 PRICELIST (X.Y.ZZ)	Price list information

Cashless Device

Field	Format	Description
PARAM0	1	„Media-ID“, if cashless reader sent it
PARAM1	2	Price of product (basic unit)
PARAM2	2	Selection number
PARAM3	8 PRICELIST (X.Y.ZZ)	Price list information

5.1.26. Transaction log Type `P` (Class 6A)

The reader receives an amount via MDB Peripheral-Device (MDB Cashless Reader, Change Giver, Bill validator or coin acceptor).

- The transaction log 'P' will be logged after each 'Begin Session' - (answer from MDB Peripheral-Slave) as soon as the amount is > 0.
- With cash acceptor (CA) every regular cash amount (coin or bill) will be logged.
- With bill validator (BV) the protocol will be written **after successful stacking**.
- Change giver: Coins will be logged with this transaction log only if the changer is not in fill mode. In fill mode 6S-L will be written.
- QR-Code Scanner (QRC): Successful read in of a valid voucher.

Cash-amounts (except from EXE-changer) will be currently recorded and persistently stored at the FRAM, in order to not lose the amount after a power breakdown. If such an amount is present after a reader start-up, it will be logged with 'FR'.

Field	Format	Description
PARAM0	1	RD: Media ID (User number) CG: Decimal place 1-5: Total cash amount Decimal place 7: Exact Change active Decimal place 8: 0 = Cash vend / 1 = Revalue CA/BV: Decimal place 1-7 → Total cash amount (Cash vend or revalue on TAG, see Dez. 8) Decimal place 8 → 0 = for cash vend 1 = for revalue on TAG FR: Specific amount, which will be read from the FRAM during the start-up. Depending PARAM1.HH CC: The last 7 digits of the PAN QRC: Token Number

PARAM1	7 HH.LL	<p>RD: 00.00 (No meaning)</p> <p>CG:</p> <p>HH = Routing Application</p> <p>1 = Coin in cashbox Cash vend</p> <p>2 = Coin in cashbox Revalue on TAG</p> <p>3 = Coin in cashbox Cash vend or revalue on TAG</p> <p>11 = Coin in tube Cash vend</p> <p>12 = Coin in tube Revalue on TAG</p> <p>13 = Coin in tube Cash vend or revalue on TAG</p> <p>LL = Channel number</p> <p>CA:</p> <p>HH = Application</p> <p>1 = Cash vend</p> <p>2 = Revalue on TAG</p> <p>3 = Cash vend or revalue on TAG</p> <p>LL = Cash index (see config ,?')</p> <p>BV:</p> <p>HH = Application</p> <p>1 = Cash vend</p> <p>2 = Revalue on TAG</p> <p>3 = Cash vend or revalue on TAG</p> <p>LL = Channel number</p> <p>FR:</p> <p>HH = 1: For Cash vend 3: For Cash vend or revalue</p> <p>LL = 0</p> <p>CC:</p> <p>HH = 0</p> <p>LL = AID (0 = PayPass, 1 = PayWave, ..., 5 = Unknown)</p> <p>QRC:</p> <p>HH = 0</p> <p>LL = Token Type</p>
PARAM2	2	<p>RD: Total amount in base unit</p> <p>CA, CG, BV: coin value / bill value</p> <p>FR: Total amount from FRAM</p> <p>CC: Preauthorization amount</p> <p>QRC: Token Value</p>
PARAM3	9	<p>MDB Slave (CG, BV, RD).</p> <p>Paralleler Cash Acceptor (CA)</p> <p>Value from FRAM during start-up (FR)</p> <p>Credit card (CC)</p> <p>QR-Code Scanner (QRC)</p>

5.1.27. Transaction log Type `p` (Class 6A)

- CA: Incorrect signal detected at parallel coin acceptor input of Mmaster6.
- BV: „Bill Stacked“ has been noticed for a bill channel which is either not communicated from the BV setup as available, or it should be permanently locked because not available in the reader configuration. In this case the bill will be moved to the stacker, but the reader did not receive the amount!
- CG: „Coins Deposited“ with „Coin routing“ = NOT USED or REJECT. Or for a channel which has not defined in setup and therefore should be permanently locked.
- CC: Error reading a credit card.
- QR: Error reading a C QR-code over 2D Bar-code scanner.

Field	Format	Description
PARAM0	1	CA: Active inputs at parallel cash interface by sampling point 0 (20ms) CG: 0 BV: 0 CC: Errorcode QR: 0: Correct data, but error at validation → check PARAM1 for Error code. 1: Timeout when receiving the header 2: Data received, but wrong length (should be 12 Bytes) → PARAM.LL = number of bytes 3: Header („QRC“) wrong 4: Structure-version wrong 5: Data length does not correspond to the structure length 6: Timeout when receiving structure data 7: Error at Ser->Read. Wrong data length → PARAM.LL = number of Bytes
PARAM1	7 HH.LL	CA: HH: 0 = Incorrect sampling (corrupted signal) 1 = good sampling (good signal, just entry at cash matrix not available. LL: last sampling-point. 0 → 20 ms → Point B 1 → 40 ms → Point C 2 → 65 ms → Point D 3 → 120 ms → Point E BV: HH: 1: Channel not communicated at BV-setup 2: No entry for this channel at .bil-config-file LL: Channel (1-16) CG: HH: 1: Channel not communicated at BV-setup 2: No entry for this channel at .chg-config-file 3: „Coin routing“ = NOT USED 4: „Coin routing“ = REJECT LL: Channel (1-16) CC: HH: 0 LL: AID (0 = PayPass, 1 = PayWave, ..., 5 = Unknown)
PARAM2	2	CC: Sub-Errorcode (e.g.. Status word of APDU) QRC: If PARAM0 = 0, PARAM1 = 00.06: TokenNr.
PARAM3	9	„CA“ / „BV“ / "CG" / "RD" / "CC" / "QR"

5.1.28. Transaction log Type `Q` (Class 6A)

Cash amount deleted after Timeout.

Field	Format	Description
PARAM0	RFU	RFU
PARAM1	RFU	RFU
PARAM2	2	Deleted amount Reported at CA8
PARAM3	RFU	RFU

5.1.29. Transaction log Type `q` (Class 6A)

Wrong conditions detected at „Revalue Limit Request“ from VMC, therefore the command Revalue Denied will be sent (Revalue Denied 0E).

Field	Format	Description
PARAM0	1	The User number from Header 2 / UserNo of Cashless
PARAM1	RFU	RFU
PARAM2	RFU	RFU
PARAM3	2	General: 07: VMC_MdbAllowRevalue = false Revalue to UKey – Error source: 02: Parameter UKEY_AllowRevalue is false. Revalue to MDBRD – Error source: 20: Parameter PER_MdbRdAllowRevalue is 0 (disabled)

5.1.30. Transaction log Type `R` (Class 6A)

Used at A008 and A002

After successful loading to the cashless reader (MDB peripheral), the loaded cash amount and the actual credit of the Tag will be logged.

Field	Format	Description
PARAM0	1	„Media-ID“ of cashless reader
PARAM1	2	Loaded amount in base unit
PARAM2	3	Actual balance on the Tag after the revaluation
PARAM3	2	Source: 1: Over VMC-interface (MDB-Master send Revalue Request) 2: Amount over Peripheral (EXE, MDB, PCA) 3: Open Transaction: The Tag was no longer present when the amount has been sent to the reader for revalue. Now the Tag is again in the antenna field.

Used at A046

Reader has been locked (Sub-code > 1) or unlocked (Sub-code = 1).

Field	Format	Description
PARAM0	RFU	
PARAM1	RFU	
PARAM2	2	Subcode of reader
PARAM3	2	Action: 0: Reader unlocked 1: Reader locked

5.1.31. Transaction log Type `r` (Class 6A)

Used at A008 and A002

If revalue to cashless reader has been refused from the device (Revalue Denied) the transaction will be logged.

Field	Format	Description
PARAM0	1	„Media-ID“ of cashless reader
PARAM1	2	Amount which should be loaded in base unit
PARAM2	3	Actual balance on the Tag
PARAM3	2	Source: 1: Over VMC-interface (MDB-Master send Revalue Request) 2: Amount over Peripheral (EXE, MDB, PCA) 3: Open Transaction: The Tag was no longer present when the amount has been sent to the reader for revalue. Now the Tag is again in the antenna field.

Used at A080

The Statistic, the transaction log or the support-package could be successfully read out from the reader and stored on the SD-card in MDD6.

Field	Format	Description
PARAM0	1	ID101 from the connected reader
PARAM1	2	What has been read out: 1: Statistics 2: Transaction log 3: Support-package
PARAM2	2	If PARAM1 = 1: Number of read out (EA301)
PARAM3	2	If PARAM1 = 1: 1 = Not erasing Statistics 2 = Statistics marked for erasing (daily counter)

Used at A046

The reader or the card has been reset.

Field	Format	Description
PARAM0	1	If PARAM1 = RD: ID101 from the connected reader If PARAM1 = TG: The User number from Header 2
PARAM1	9	RD = Reader reset TG = Tag (User Card) reset
PARAM2	2	If PARAM1 = RD: Amount of the erased counter If PARAM1 = TG: The erased balance
PARAM3	RFU	

5.1.32. Transaction log Type `S` (Class 6A)

After user media validation (country code, company code, department, blocked user and max. credit) the actual credit will be logged.

A003 creates this transaction log at:

- Read in of a balance block over Frame 63h or 69h (c.f. Section-6c.doc), the block must be at least read only (if two balance blocks on the Tag, two transaction logs will be created).
- Read back over Frame 64h or 6Ah, but only if balance or price list has been modified.

A003 does not create this transaction log if:

- A card has been presented but no client application is online, or this one not reads in the balance block (e.g. MIFdump).
- The same card again detected (e.g. playing around inside the antenna field).

UKey-Tag

Field	Format	Description
PARAM0	1	The user number from header 2
PARAM1	7 HH.LL	HH: Number of the vend block (1 or 2) LL: Category (1..255)
PARAM2	3	The actual credit on the TAG
PARAM3	7 HH.LL	HH: RFID-Technologie 10: Mifare Classic 11: Mifare Classic NFC-Handy 20: Mifare Plus SL3 (MIC Keys) 21: Mifare Plus SL1 (MIC Keys) 22: Mifare Plus SL3 (Non MIC Keys) 23: Mifare Plus SL1 (Non MIC Keys) 24: Mifare Plus SL2 25: Mifare Plus SL0 L1 26: Mifare Plus SL0 L3 30: Mifare Desfire 40: TIRF (not supported from version r4365) 50: Calypso 60: EMVCo (not supported from A003) LL: Size in kB (Only valid for Mifare) 0: unknown 1: 320 Bytes (Mifare MINI) 2: 1 KB 3: 2 KB 4: 4 KB 5: 8 KB

Example: [Mifare Classic 1K](#), VEND1, Cat. 1, Credit 620, UserNo 1044
6A.. 18.03.11 16:09:02 S 00001044 01.01 620 **10.01**

Example: [Calypso](#), VEND1, Cat. 1, Credit 120, UserNo 50233
6A.. 18.03.11 16:09:02 S 00050233 01.01 1120 **50.00**

Example: [Mifare Plus 2K](#), VEND2, Cat. 4, Credit 2130, UserNo 1001
6A.. 18.03.11 16:09:02 S 00001001 02.04 2130 **20.02**

5.1.33. Transaction log Type `T` (Class 6A)

Used at A008

U-KEY / TAG

The transaction has been performed successfully (amount deducted from the TAG-credit).

Field	Format	Description
PARAM0	USER	The User number from Header 2
PARAM1	0	Price of product (basic unit)
PARAM2	n	Actual credit on the TAG after the transaction
PARAM3	0	Selection number

Credit Card

The transaction has been confirmed to the credit card terminal.

Field	Format	Description
PARAM0	USER	The last seven digits of the credit card number
PARAM1	0	Price of product (basic unit)
PARAM2	n	0
PARAM3	0	Selection number

Cash

The transaction has been performed.

Field	Format	Description
PARAM0	1	0 (=Cash)
PARAM1	0	Price of product (basic unit)
PARAM2	3	The remaining cash credit
PARAM3	2	Selection number

Used at A003

A003 creates this transaction log at:

- Successful migration of a Mifare plus Tag from SL1 to SL3.
- Successful finalisation of a Mifare plus SL3 Tag from an external issuer.

Field	Format	Description
PARAM0	1	The User number from Header 2
PARAM1	7 (HH.LL)	HH: License price in Units LL: Max logical blocks (de-formatted: 0), Bit7: with Ext Part
PARAM2	3	Bit 0-23 of Tag UID
PARAM3	7 (HH.LL)	HH: Tag type LL: Tag size in K (nominal)

Used at A046

The Tag has been locked (Subcode > 1) or unlocked (Subcode = 1).

Field	Format	Description
PARAM0	1	If PARAM1 = 0: The User number from Header 2 else: Serial number (UID)
PARAM1	7 (HH.LL)	Tag Type: HH: Diagnose functions group LL: Diagnose function Examples: 00.00: User-Tag 08.113: Lock Tag (8/113) 07.113: Backup Lock Tag (7/113)
PARAM2	2	Subcode of the TAG (log. Block 19)
PARAM3	2	Action: 0: Tag unlocked 1: Tag locked

5.1.34. Transaction log Type `t` (Class 6A)

Reporting the history, if an entire Happy-Meals has been purchased.

For all products which are involved, price and reference are reported.

Field	Format	Description
PARAM0	2	Product price
PARAM1	2	Product reference
PARAM2	RFU	
PARAM3	RFU	

5.1.35. Transaction log Type `V` (Class 6A)

Used at A008

The VMC could issue the product successfully (from VMC Dispense OK) – Vend over VEND-Block.
U-KEY / TAG

Field	Format	Description
PARAM0	1	The User number from Header 2
PARAM1	2	Price of product (basic unit)
PARAM2	3	Actual credit on the TAG after the transaction
PARAM3	8 PRICELIST (X.Y.ZZ)	Price list information

5.1.36. Transaction log Type `v` (Class 6A)

Used at A008

This transaction is only written if parameter **MISC_ExtProtocols = 2** (extended transaction logs) . This feature must be activated over “direct reader access”.
The application received VMC Dispense Request from VMC.

Field	Format	Description
PARAM0	RFU	RFU
PARAM1	2	Price of product (basic unit)
PARAM2	2	Selection number
PARAM3	RFU	RFU

5.1.37. Transaction log Type `W` (Class 6A)

Used at A008

The VMC could issue the product successfully (from VMC Dispense OK) – Vend over 2D-Barcode Voucher.

UKey-Tag

Field	Format	Description
PARAM0	1	Gutscheinnummer (Token-Number)
PARAM1	2	Price of product (basic unit)
PARAM2	RFU	RFU
PARAM3	8 PRICELIST (X.Y.ZZ)	Price list information

5.1.38. Transaction log Type `X` (Class 6A)

If a user TAG which is entered at the blacklist (user number), or a Diagnose-TAG (Serial number) and the TAG is marked as locked has been presented.

Field	Format	Description
PARAM0	USER	User number from Header 2 on TAG or serial number on Diagnose-TAG
PARAM1	RFU	
PARAM2	RFU	
PARAM3	RFU	

5.1.39. Transaction log Type `Y` (Class 6A)

The reader receives an amount (for revaluation or cash sale) over a peripheral device.

The value will be accumulated.

EXE-Change giver: The transaction log. will be written after each "ACCEPT DATA" command, if the amount is > 0, or if the die information changed (e.g. value 0).

Field	Format	Description
PARAM0	0	Accumulated amount
PARAM1	0	Source: 1: EXE-Change giver
PARAM2	0	If source = 1: Exact-Change-Flag (set, if = 1)
PARAM3	HH.LL	If source = 1: HH = Scaling factor / LL = Dec. point. pos

5.1.40. Transaction log Type `y` (Class 6A)

Successful cancelation after Vend Failure. See also 6E-y.

Field	Format	Description
PARAM0	1	UKey: UserNo / Cashless: „Media-ID“ / CreditCard: No
PARAM1	2	Cancelled amount (=Product price of the failed vend)
PARAM2	2	Balance after cancelation 0 if credit card
PARAM3	7 (HH.LL)	HH: 01: UKey 02: Cashless-Reader LL: 00: If not UKey 01: Debit 02: Jeton 03: Bonus 04: Reader-Tokens

5.1.41. Transaction log Type `~` (Class 6A)

Used at A008

After validation of a Service-TAG with the function Test vend (country code, company code range, blocked user and max. credit) and credit initialization (Test credit 9999) the actual credit will be logged.

Field	Format	Description
PARAM0	1	Serial number of TAG
PARAM1	2	Number of vend block (always 1)
PARAM2	2	Current credit on the TAG
PARAM3	2	Category (always 1)

Used at A003

A003 creates this transaction log if:

- Transferring of mifare[®] Format credits → Format units (MDSedit version 6.10).

Field	Format	Description
PARAM0	2	MCX Number
PARAM1	2	The conversion factor (10)
PARAM2	2	New format units amount
PARAM3		0

6. Transaction Types of Class 'E' (Error)

Type	Transaction log-Class 6E
A	Assert
B	
C	Error with Credit card transaction / MPC-Protocol
D	Transaction cancelled
E	Handled or Unhandled Exception (ERROR)
e	Unhandled Event
F**	General error (INFO message)
G	Service media function not allowed or inactive
H	Heap error
I	
K	
L	
l	Successful correction of cashless transaction on application level
M	Errors related to MDB-peripheral
m**	MDB-lowlevel errors related to peripheral
N	
n	Error during correction of cashless transaction on application level
O	
P	Invalid price
Q	
R**	RFID-Error
r	Failed to read out Statistic, Protocol or Support-Package from reader.
S	
T	TCP socket: error during send procedure
U	Usart
u	
V	
v	
W	VMC watchdog of DISPENSE and NO_FUND state
X	
Y	
y	Error during correction of cashless transaction
Z	

** This transaction is only written if parameter **MISC_ExtProtocols** = 2. This feature must be activated over "direct reader access".

6.1. Transaction Log Details

6.1.1. Transaction log Type `A` (Class 6E)

Assert (unhandled software error). After this transaction log. The reader will be rebooted. The function name and the line at which the assert occurred will be logged. If possible an „error.log” will be written.

Field	Format	Description
PARAM0	T	Function name (the first three characters)
PARAM1	T	Function name (characters 4-5)
PARAM2	9	Function name (characters 6-8)
PARAM3	2	Line number

6.1.2. Transaction log Type `C` (Class 6E)

Error with Credit card transaction / MPC-Protocoll.

Field	Format	Description
PARAM0	7 (HH.LL)	<p>HH: Error category</p> <ul style="list-style-type: none"> 1: Socket-Error 2: SSL-Error 3: HTTP Error Code 4: ErrorNotification / Error in Response <p>LL: Request/Reponse</p> <ul style="list-style-type: none"> 0: -- 1: registerPos 2: checkStatus 3: transaction 4: reversal
PARAM1	2	<p>If PARAM0.HH=1 (Socket-Error)</p> <ul style="list-style-type: none"> 1: DNS Error (MPC-Server Host could not be mapped) 2: Error Creating Socket 3: Error returned from tcp_connect 4: Error Connecting Socket 5: Socket closed unexpectedly 6: Socket connection lost 7: Socket aborted <p>If PARAM0.HH=2 (SSL-Fehler)</p> <ul style="list-style-type: none"> 1: Error creating new client Session <p>If PARAM0.HH=4 und PARAM2 = 0 (Error code in Response)</p> <ul style="list-style-type: none"> 100: Declined – Generic 101: Declined – Invalid card number 102: Declined – Card expired 103: Declined – Card unknown 104: Declined – Duplicate transaction 105: Declined – Funds too low 106: Declined – Card blocked 107: Declined – by Fraud Management
PARAM2	2	<p>If PARAM0.HH=3 (HTTP Error Code)</p> <p>standardized HTTP Error Code http://www.w3.org/Protocols/rfc2616/rfc2616-sec10.html)</p> <p>If PARAM0.HH=4 and PRAM1 = 0 (ErrorNotification)</p> <ul style="list-style-type: none"> 1: Message parsing error 2: Command not supported 3 Invalid command parameter 4: Invalid credential type 100: POS transaction reference invalid. 101: PSP transaction reference invalid 102: Duplicate transaction. 103: POS Not registered 104: Currency not supported 105: Unknown AID 106: AID not accepted 199: Concentrator fault. 200: PSP service not available 201: PSP service timeout 300: PSP processing error 301: PSP busy
PARAM3	9	CC

6.1.3. Transaction log Type `D` (Class 6E)

The transaction-inquiry (Vend request) of the VMC could not processed, the reader sends ,Vend denied'

U-KEY / TAG

Field	Format	Description
PARAM0	1	The User number from Header 2
PARAM1	2	Price of product (basic unit)
PARAM2	2	Selection number
PARAM3	8 PRICELIST (X.Y.ZZ)	Price list information

Credit Card

Field	Format	Description
PARAM0	1	The last seven digits of the credit card number
PARAM1	2	Price of product (basic unit)
PARAM2	2	Selection number
PARAM3	8 PRICELIST (X.Y.ZZ)	Price list information

6.1.4. Transaction log Type `E` (Class 6E)

This transaction will be logged, if a handled or unhandled exception occurred. As well if an exception has been deleted.

Field	Format	Description
PARAM0	2	Exception-Type: 1: Handled Exception 2: Unhandled Exception 3: Cleared Unhandled Exception
PARAM1	7 (HH.LL)	Number of the issued error: MS-Byte: Error group LS-Byte: Error detail (e.g. the hexadecimal value 0x0F02, will be interpreted as error group 15 with error detail 02)
PARAM2	2	Extended Error Information (2 Bytes)
PARAM3	2	MS-Byte: Module Identification (from common.h)

6.1.5. Transaction log Type `e` (Class 6E)

This transaction will be logged, if an unhandled event occurred.

Field	Format	Description
PARAM0	2	Status of the received modul
PARAM1	2	Event code defined at transmitting modul
PARAM2	2	Modul ID of the transmitting modul
PARAM3	2	Modul ID of the receiving modul

6.1.6. Transaction log Type `F` (Class 6E)

This transaction is only written if parameter **MISC_ProtocolLevel** = 2 1 (extended transaction logs) . This error will be logged if a user error occurred. Additionally it will show a message on the display.

Field	Format	Description
PARAM0	1	The User number from Header 2
PARAM1	7 (HH.LL)	Number of the issued error: MS-Byte: Error group LS-Byte: Error detail (e.g. the hexadecimal value 0x0F02, will be interpreted as error group 15 with error detail 02)
PARAM2	2	Extended Error Information (2 Bytes)
PARAM3	2	MS-Byte: Module Identification (from common.h)

6.1.7. Transaction log Type `G` (Class 6E)

The service-TAG Function is disabled.

Field	Format	Description
PARAM0	USER	Serial number of the service TAG
PARAM1	0	Diagnose function group (Block type of log. Block 0)
PARAM2	0	Service TAG function (at log. Block 0)
PARAM3	HH.LL	Error code: 03.01 Service TAG Function not activated 03.02 Button not pressed

6.1.8. Transaction log Type `H` (Class 6E)

Heap Corrupt. This error only occurs, if the heap-watcher is active. If possible it will write a heap.log file and issue an assert error.

Logged will be the function name, on which the buffer (which is now corrupted) has been created and the buffer size. So the logged function name is not the place, on which buffer has been corrupted (this is not possible to detect).

Field	Format	Description
PARAM0	9	Function name (the first three characters)
PARAM1	9	Function name (characters 4-5)
PARAM2	9	Function name (characters 6-8)
PARAM3	2	Buffer size

6.1.9. Transaction log Type `L` (Class 6E)

Inconsistence detected from MDBslave driver at VMC command.

Field	Format	Description
PARAM0	2	Inconsistence <ul style="list-style-type: none"> 1: Inconsistent start-up sequence detected, proceeding the Enable command. 2: Enable Feature Options 3: Vend Request 4: Negative Vend Request 5: Revalue Request 6: Revalue Limit Request 7: Peripheral ID
PARAM1	7 (HH.LL)	HH: Expanded Currency Mode Bit: 0: Expanded Currency Mode Bit not set 1: Expanded Currency Mode Bit set LL: MDB Level: 0: Invalid Level 1: Level 1 2: Level 2 3: Level 3
PARAM2	2	Init Sequenz: Bit 0: ‚Setup Config Data‘ Bit 1: ‚Setup Max / Min Price‘ Bit 2: ‚Request ID‘ VMC Level 3 Expanded Currency Mode: Bit 3: ‚Enable Options‘ Bit 4: ‚Setup Max / Min Price‘ - 32 bit for Level 03+ Like PARAM2 in 6C `N` Protocol.
PARAM3	2	PARAM0 = 1: <ul style="list-style-type: none"> 1: VMC Level from previous start-up sequence loaded. 2: Missing Setup Config 3: Enable Option command received in another Level as Level3. PARAM0 = 2: <ul style="list-style-type: none"> 1: Enable Options command without Request ID and ECM Mode not set. 2: Enable Options command without Request ID and ECM Mode set. 3: Enable Options outside of start-up sequence received. PARAM0 = 3, 4, 5: <ul style="list-style-type: none"> 1: Level 3 command in ECM Mode received. 2: ECM command in Level 3 received. 3: Command received in Level 1, this is not allowed at this level. PARAM0 = 6: <ul style="list-style-type: none"> 1: Command received in Level 1, this is not allowed at this level. PARAM0 = 7: <ul style="list-style-type: none"> 0: RFU

6.1.10. Transaction log Type `I` (Class 6E)

Correction of a U-KEY-transaction at application base.

- If the application receives the command „dispense failed“ from VMC and the parameter VND_Reversal is = 1, the U-KEY driver loads the deduced amount back to the TAG, or the CCTRM driver causes a cancellation.
- This transaction log will be also written, if the TAG has been removed during the write operation and further replaced to close the session.

Does two cases can be differentiated. If the transaction log is in coherency with a corrected revalue operation, then it will also record an „application“ (class A) „I“ transaction log.

U-KEY / TAG

Field	Format	Description
PARAM0	1	The User number from Header 2
PARAM1	2	Refunded price in basic unit (depending on type of refund) Or Revalue amount If PARAM3 = 3: Refunded / deducted Number of bonus (difference between new and old bonus balance).
PARAM2	2	Balance before correction
PARAM3	2	Concerned block 0 Debit (Vend #1 or #2) 1 Token 2 Bonus 3 Write back Bonus data if refunding possible in case of debit vend (one price mode of operation).. 4 Token at account on the reader

CreditCard

Field	Format	Description
PARAM0	1	The last seven digits of the credit card number
PARAM1	2	Refunded price in basic unit
PARAM2	2	0
PARAM3	2	Type of refunding: 5 = Credit Card

6.1.11. Transaction log Type `M` (Class 6E)

Errors related to MDB-peripheral.

Field	Format	Description
PARAM0	0	BV: Slave-Error (Status byte according MDB spec.) <ul style="list-style-type: none"> 1: Defective Motor 2: Sensor Problem 4: Checksum Error 5: Bill jammed 7: Bill removed from ESCROW Position by an unknown means 10: Invalid ESCROW request 11: Bill Rejected. Could not be identified 12: Possible Credited Bill Removal RD: <ul style="list-style-type: none"> 1: Out of Sequence 2: Malfunction / Error
PARAM1	0	Configuration-errors / Setup errors BV: (1-5: BV locked. From 100: BV runs anyway) <ul style="list-style-type: none"> 1: Currency code from BV does not match with the one in the reader. At such cases the currency code will be logged at PARAM2. 2: No channels available! (See C-?) 3: On STARTUP command the BV returned invalid data. 4: On EXPANSION command the BV returned invalid data. 5: On STACKER command the BV returned invalid data. 100: On STARTUP command the BV returned data which does not correspond to MDB-standard. Anyway they will be accepted. At PARAM2 the number of received bytes will be logged. RD: (RD locked) <ul style="list-style-type: none"> 1: Currency code from RD does not match with the one in the reader. At such cases the currency code will be logged at PARAM2.
PARAM2	0	Detail-infos BV: <ul style="list-style-type: none"> If PARAM1 = 1: Currency code from BV If PARAM1 = 100: Number of received bytes (27 would be according to the standard.) RD: <ul style="list-style-type: none"> If PARAM1 = 1: CurrencyCode from RD If PARAM0 = 1: State from peripheral reader (Level > 1) <ul style="list-style-type: none"> 1: Inactive state 2: Disabled state 3: Enabled state 4: Session idle state 5: Vend state 6: Revalue state 7: Negative Vend state If PARAM0 = 2: Error Code
PARAM3	T	MDB Slave (RD,CG, BV)

6.1.12. Transaction log Type `m` (Class 6E)

MDB-lowlevel errors related to MDB peripheral. **Only available if PER_MdbExtProtocols enabled!**

Field	Format	Description
PARAM0	2	Information typ: 0: MDB-Transmission Error 1: Buffer overflow: MDB-Slave ignored the basic rules and transferred unrequested data.
PARAM1	2	If PARAM0 = 0 (MDB-Transmission Error) 1: MDB-Frame contains no data. 2: MDB-Frame with more than 36 bytes received. 3: Mod-Bit at last byte not set. 4: Mod-Bit set at wrong byte. 5: Checksum error 6: Slave answered to late (Default: slave must answer after 10ms). At PARAM2 the measured time will be logged. 7: Slave answered with NAK 8: Master cannot send ACK, because delay would be > 5ms! At PARAM2 the calculated delay [us] will be logged. Slave-Error
PARAM2	2	If PARAM1 = 6: Delay in [us]. Data ignored, because slave must no longer answer! If PARAM1 = 7: Delay in [us]. After this time the ACK would be send. -> But this has been inhibited.
PARAM3	9	If PARAM0 = 0 MDB Slave (RD,CG, BV)MDB Slave (RD,CG, BV)

6.1.13. Transaction log Type `n` (Class 6E)

Error during correction of a U-KEY-transaction at application base. If it's not possible to refund or to correct the amount to be loaded the transaction will be deleted and logged. If the correction was successful a ,I' transaction will be logged.

The transaction stays at the ErrorCorrectionCache" for 30 seconds. If the TAG is not presented the entries will be deleted without recording the transaction log.

Field	Format	Description
PARAM0	1	If U-KEY: TAG-User number If credit card:: The last seven digits of the credit card number
PARAM1	2	Deducted price in basic unit (depending on the concerned block) Or amount to be loaded
PARAM2	3	If U-KEY: The balance on the TAG If credit card: 0
PARAM3	2	Concerned block: 0 Debit (Vend #1 or #2) 1 Token 2 Bonus 3 – 4 – 5 Credit Card

6.1.14. Transaction log Type `P` (Class 6E)

Errors in context with prices.

Field	Format	Description
PARAM0	2	<p>0: The effective price received from the RFM is invalid.</p> <p>1: In case of vend with peripheral device (CHG, CLD, BV): The price does not match to the scaling factor (e.g. with cash sale, the product price at the changer must be an integer multiple of the scaling factor).</p> <p>2: At cash sale: The price is too high to vend via Change-Giver (at EXE only the scaling factor of 250 can be transmitted as price)</p> <p>3: Configuration of prices incorrect related to change giver: During the start up, the cash price lists will be checked through with the scaling factor of the change giver. If a price does not match to the pattern, the "P" transaction log will be written.</p> <p>4: Configuration of prices incorrect related to cashless device: During the start up, the cashless pricelist will be checked through with the scaling factor of the connected reader. . If a price does not match to the pattern, the "P" transaction log will be written.</p>
PARAM1	2	PARAM0 = 1, 2 Price
PARAM2	2	PARAM0 = 1, 2 Selection number
PARAM3	8 PRICELIST (X.Y.ZZ)	PARAM0 = 1, 2 Price list information

6.1.15. Transaction log Type `R` (Class 6E)

RFID-Error.

Field	Format	Description
PARAM0	2	0: General 1: Mifare
PARAM1	2	If PARAM0 = 0 (General) 1: PN512 has not set IRQ-Pin within the required timeout. If PARAM0 = 1 (MIFARE) Error code of BFL from NXP
PARAM2	2	If PARAM0 = 0 (General) 0 = Source: Function: PN5X_TransceiveEx 1 = Source: Function: mPN5X_MifareAuth If PARAM0 = 1 (MIFARE) Block-Address
PARAM3	2	If PARAM0 = 0 (General) 0 If PARAM0 = 1 (MIFARE) 1: AuthKey A 2: AuthKey B 3: Read 4: Write

PARAM0 = 0 (General): Per 30s only 1 Protocoll with the same parameters is written!

PARAM0 = 1 (Mifare): Only written, if **MISC_ProtocolLevel = 2!**

6.1.16. Transaction log Type `r` (Class 6E)

Error during read out of transaction log, statistic or support package, the file could **not** be read out successfully.

Field	Format	Description
PARAM0	1	ID101 from the connected reader
PARAM1	2	Which data has been read out: 1: Statistic 2: Transaction log 3: Support-Package
PARAM2	10	Error-Code
PARAM3	2	Actual progress, where the error occurred

6.1.17. Transaction log Type `T` (Class 6E)

If an error occurred sending a TCP-Telegram

Field	Format	Description
PARAM0	2	Module ID
PARAM1	2	Socket Handle

PARAM2	2	Socket Status: TCP_STATE_FREE 0 Entry is free and TCP_STATE_CLOSED 1 Entry allocated, socket still TCP_STATE_LISTEN 2 Socket waiting for TCP_STATE_SYN_REC 3 SYN frame TCP_STATE_SYN_SENT 4 SYN packet sent to establish a connect. TCP_STATE_FINW1 5 Tcp_close started FIN packet was sent TCP_STATE_FINW2 6 Our FIN ack-ed, waiting for remote FIN TCP_STATE_CLOSING 7 Received FIN independently of our FIN TCP_STATE_LAST_ACK 8 Waiting for last ACK for our FIN TCP_STATE_TWAIT 9 Timed waiting for 2MSL TCP_STATE_CONNECT 10 TCP Connection established
PARAM3	RFU	RFU

6.1.18. Transaction log Type `U` (Class 6E)

Error of the UART. This transaction is only written if parameter MISC_ExtProtocols = 1 (extended transaction logs).

Field	Format	Description
PARAM0	2	Usart Nr. 0 Usart 0 1 Usart 1 2 DB-Uart (MDS)
PARAM1	2	Error number of the triggered error (defined a Uart.h)
PARAM2	RFU	0
PARAM3	RFU	0

6.1.19. Transaction log Type `W` (Class 6E)

Monitoring of the DISPENSE and NO_FUND State. If VMC Watchdog responded, there will be a „W“ transaction log recorded before resetting the device.

Field	Format	Description
PARAM0	2	Actual state 1 STARTUP 2 IDLE 3 FUND 4 NO FUND 5 DISPENSE 6 NO VMC 7 INHIBIT 8 EXCEPTION
PARAM1	2	Next state (same list as above)
PARAM2	7 (HH.LL)	HH: VMC-Type: 0: „Auto detect“ active, Type not yet detected 1: MDB 2: EXE/BDV (not yet defined, which one of the both types) 3: EXE 4: BDV LL: Additional info: 0: No info 1: Wrong VMC-Type (Only incoherence with PARAM0=6)
PARAM3	RFU	RFU

6.1.20. Transaction log Type `y` (Class 6E)

Error during correction of cashless transaction after vend failure. Also see A-y.

Field	Format	Description
PARAM0	1	UKey: UserNo / Cashless: „Media-ID“ / CreditCard: No
PARAM1	2	Amount to be cancelled (= Product price of the failed vend)
PARAM2	2	1: No changes written on the media. If so cancellation will be repeated. 2: Not sure if changes could be written. If so cancellation will not be repeated. Maybe the transaction will be restored (E-I / E-n). 0 If credit card
PARAM3	7 (HH.LL)	HH: 01: UKey 02: Cashless-Reader 03: Credit card LL: 00: If not UKey 01: Debit 02: Jeton 03: Bonus 04: Reader-Tokens

7. Pricelist Information

Description of the third parameter of protocol types B, b, D, d, I, J, K, O, o, S, V, +, -

PARAM x	PRICELIST	Pricelist information
---------	-----------	-----------------------

The information consists of 3 values separated by a decimal point “.”

MS-Byte		LS-Byte
Bit 7..4	Bit 3..0	
Source Group	Pricelist of Source-Group	Category

Source Group		Pricelist		Category
0	UKey	0	Debit	0-255
		1	Debit Rate	--
		2	Token	--
		3	Bonus	--
		4	Reader Token	--
		5	Token Category	1-8
		6	Debit Rate Category	1-8
1	Credit Card	0	Credit Card Normal	--
		1	Credit Card Rate	
2	Cash	0	Cash Normal	--
		1	Cash Rate	--
3	Cashless	0	Cashless Normal	--
		1	Cashless Rate	--
15	Vmc	0	Automat	1-255
		1	Reduction Percent	1-255
		2	Reduction Amount	1-255
		3	Rate Factor	--
		4	Rate Reduction Percent	1-8
		5	Rate Reduction Amount	--
		6	Rate Reduction Amount Category	1-8
		7	Reduction Amount Category	1-8

Examples:

0.0.7 = Debit pricelist category 7 (from UKey VEND-block)
 0.4.0 = Reader Token
 0.5.0 = Token Vend with category > 8 (from UKey VEND-block)
 0.5.4 = Token Vend with category 4 (from UKey VEND-block)
 15.0.3 = Vmc default with category 3 (from UKey VEND-block)
 15.0.15 = Vmc default with category 15 (from UKey VEND-block)

8. Transaction Log Examples

8.1. Badge vend (VMC prices)

Action reader		Action VMC	Transaction Log					
			Cl	T	Par 0	Par 1	Par 2	Par 3
A TAG with user number 24 and balance 15.20 at vend block 1 with price-categoryK2 has been inserted.								
			6A..	S	24	01.02	1520	10.01
<i>Credit at system (Begin session) .</i>	→	Credit EUR 15.20 User No 24 PrixList V.0.002						
		A product with selection no. 3 and price of 1.30 has been chosen at the VMC.						
	←	Vend request						
The reader is checking the vend request and debits the amount of 1.30 from the TAG balance.			6A..	T	24	130	1390	3
<i>Vend approved</i>	→	Dispensing.. Product 03 Price EUR 1.30						
		VMC dispensing the product						
	←	The product has been dispensed successfully (vend success).						
After receiving of the vend success.		Credit EUR 13.90 User No 24 PrixList V.0.002		V	24	130	1390	0.0.01

8.1.1. Badge vend listing of transaction log:

Number	Class	Date	Time	Type	Par 0	Par 1	Par 2	Par 3
0000000001	6A..	21.06.10	09:03:05	S	00000024	01.02	1520	10.01
0000000002	6A..	21.06.10	09:03:07	T	00000024	130	1390	3
0000000003	6A..	21.06.10	09:04:15	V	00000024	130	1390	0.0.01

8.2. Token vend (VMC prices)

Action reader		Action VMC	Transaction Log					
			Cl	T	Par 0	Par 1	Par 2	Par 3
A TAG with user number 24 and 3 token balance (credit 13.90 at vend block 1) with price-category K2 has been inserted.			6A..	S	24	01.02	1390	10.01
<i>Credit at the system (Begin session).</i>	→	Token 3 Credit EUR 13.90 User No Nr 24						
		Chosen product nr. 7 at the VMC						
	←	Vend request						
The reader is checking the vend request and debits 2 token from the token balance.			6A..	T	24	2	2	7
<i>Vend approved</i>	→	Dispensing... Product 07 Token 2						
		VMC dispensing the product						
	←	The product has been dispensed successfully. (vend success).						
After receiving vend success.		Token 1 Credit EUR 13.90 User No Nr 24	6A..	J	24	2	1	0.5.01

8.2.1. Token vend listing of transaction log:

Number	Class	Date	Time	Type	Par 0	Par 1	Par 2	Par 3
0000000001	6A..	21.06.10	09:03:05	S	00000024	01.02	1390	10.01
0000000002	6A..	21.06.10	09:03:07	T	00000024	2	2	7
0000000003	6A..	21.06.10	09:04:15	J	00000024	2	1	0.5.01

8.3. Bonus vend (VMC prices)

Action reader		Action VMC	Transaction Log					
			Cl	T	Par 0	Par 1	Par 2	Par 3
A TAG with user number 24 and bonus balance 3 (credit 13.90 at vend block 1) with price-category K2 has been inserted.			6A..	S	24	01.02	1390	10.01
<i>Credit at the system (Begin session).</i>	→	Bonus 3 Credit EUR 13.90 User No Nr 24						
Chosen product nr. 7 at the VMC								
	←	Vend request						
The reader is checking the vend request and debits 1 bonus from the bonus balance.			6A..	T	24	1	2	7
<i>Vend approved</i>	→	Dispensing... Product 07 Bonus 1						
		VMC dispensing the product						
	←	The product has been dispensed successfully. (vend success).						
After receiving vend success.		Bonus 2 Credit EUR 13.90 User No Nr 24	6A..	I	24	1	2	0.3.00

8.3.1. Bonus vend listing of transaction log:

Number	Class	Date	Time	Type	Par 0	Par 1	Par 2	Par 3
0000000001	6A..	21.06.10	09:03:05	S	00000024	01.02	1390	10.01
0000000002	6A..	21.06.10	09:03:07	T	00000024	1	2	7
0000000003	6A..	21.06.10	09:04:15	I	00000024	1	2	0.3.00

8.4. Cash vend (VMC prices)

Action reader		Action VMC	Transaction Log					
			Cl	T	Par 0	Par 1	Par 2	Par 3
20-Cent-coin inserted			6A..	P	00000020	01.02	20	CG
Credit at the system (Begin session).	→	Cash EUR 0.20 Drop exact Money PrixList V.0.000						
20-Cent-coin inserted			6A..	P	00000040	01.02	20	CG
Credit at the system (Begin session).	→	Cash EUR 0.40 Drop exact Money PrixList V.0.000						
50-Cent-coin inserted			6A..	P	00000090	01.02	50	CG
Credit at the system (Begin session).	→	Cash EUR 0.90 Drop exact Money PrixList V.0.000						
Chosen product nr. 7 at the VMC								
	←	Vend request.						
The price is higher as the inserted amount. (vend denied)		Cash EUR 0.90 Drop exact Money PrixList V.0.000	6A..	d	00000000	100	7	2.0.00
20-Cent-coin inserted			6A..	P	00000110	01.02	20	CG
Credit at the system (Begin session).	→	Cash EUR 1.10 Drop exact Money PrixList V.0.000						
The reader is checking the vend request and debits the price from the cash credit.			6A..	T	00000000	100	10	7
Vend approved	→	Dispensing... Product 07 Price EUR 1.00						
	←	VMC dispensing the product						
After receiving vend success.			6A..	B		100	10	2.0.00
The reader cleared the cash credit after a timeout			6A..	Q	0	0	10	CG

8.4.1. Cash vend listing of transaction log:

Number	Class	Date	Time	Type	Par 0	Par 1	Par 2	Par 3
0000000001	6A..	21.06.10	09:03:05	P	00000020	01.02	20	CG
0000000002	6A..	21.06.10	09:03:07	P	00000040	01.02	20	CG
0000000003	6A..	21.06.10	09:03:09	P	00000090	01.02	50	CG
0000000005	6A..	21.06.10	09:03:20	d	00000000	100	7	2.0.00
0000000006	6A..	21.06.10	09:04:15	P	00000110	01.02	20	CG
0000000007	6A..	21.06.10	09:04:31	T	00000000	100	10	7
0000000008	6A..	21.06.10	09:04:50	B		100	10	2.0.00
0000000009	6A..	21.06.10	09:05:20	Q	00000000	0	10	CG

8.5. Loading to Media

Action reader		Action VMC	Transaction Log					
			Cl	T	Par 0	Par 1	Par 2	Par 3
A TAG with user number 24 and credit 13.90 at vend block with price-category K1 has been inserted.			6A..	S	24	01.01	1390	10.01
<i>Credit at the system (Begin session).</i>	→	Credit EUR 13.90 User No 24 PrixList V.0.001						
20-Cent-coin inserted			6A..	P	00000020	01.02	20	CG
<i>Credit at the system (Begin session).</i>	→	Credit EUR 14.10						
Loading			6A..	L	00000024	20	1410	2
10-Cent-coin inserted			6A..	P	00000010	01.01	10	CG
Loading			6A..	L	00000024	10	1420	2
<i>Credit at the system (Begin session).</i>	→	Credit EUR 14.20						
			6S..	H	0	0	0.00	CG
<i>TAG removed</i>	→		6S..	H	0	0	0.00	RD
		Ready (MDB) Drop exact Money 09:03:22	6C..	S	1	1		

8.5.1. Loading listing of transaction log:

Number	Class	Date	Time	Type	Par 0	Par 1	Par 2	Par 3
0000000001	6A..	06.09.10	09:03:05	S	00000024	01.01	1390	10.01
0000000002	6A..	06.09.10	09:03:07	P	00000020	01.02	20	CG
0000000003	6A..	06.09.10	09:03:09	L	00000024	20	1410	2
0000000004	6A..	06.09.10	09:03:11	P	00000020	01.01	10	CG
0000000005	6A..	06.09.10	09:03:17	L	00000024	10	1420	2
0000000006	6S..	06.09.10	09:03:20	H	0	0	0.00	CG
0000000007	6S..	06.09.10	09:03:22	H	0	0	0.00	RD
0000000008	6C..	06.09.10	09:03:24	S	1	1		