

Author Scanflow	No. SCO-SW-106	
Subject Scanflow Cash Register Protocol (SCRP)	Date 2013-03-27	Rev. 1.9

TABLE OF CONTENTS

TABLE OF CONTENTS.....	1
1 GENERAL.....	4
1.1 Introduction.....	4
1.2 Document History.....	4
2 OVERVIEW.....	6
2.1 Background.....	6
2.2 Terminology.....	7
2.3 References.....	7
3 SPECIFICATION.....	8
3.1 General.....	8
3.2 Major Concepts.....	8
3.2.1 Cash Register Signing.....	8
3.2.2 Registration Accounts.....	8
3.2.3 Transaction Methods.....	10
3.2.3.1 Bank Transaction with Signature Upon Receipt.....	10
3.2.4 Error state & Resuming.....	11
3.2.5 Display Receipt contents.....	11
3.2.6 Receipt Printing.....	11
3.2.6.1 CR-Printing.....	11
3.2.6.2 SFU-Printing.....	12
3.2.6.3 Account number & Store/Recall barcode.....	12
3.2.6.4 HTML-style data format.....	12
3.3 Application Interface.....	13
3.3.1 General.....	13
3.3.2 Command/Response Format.....	13
3.3.3 Result Coding.....	14
3.3.3.1 Rules.....	14
3.3.3.2 Numeric Order List.....	14
3.3.4 Cash Register Variables.....	15
3.3.4.1 Reading out Event List.....	16
3.3.4.2 Reading out Account states.....	17
4 COMMAND/RESPONSE OVERVIEW.....	18
4.1 General Responses.....	18
4.2 Initialize SCRP Service.....	18
4.3 Terminate SCRP Service.....	18
4.4 Cash Register Sign On.....	19
4.5 Cash Register Sign Off.....	19
4.6 Get CR Variable.....	19
4.7 Open Account.....	19
4.8 Close Current Account.....	20
4.9 Request Article Identification.....	20
4.9.1 Response 211.....	20
4.9.2 Response 213.....	21
4.9.3 Error Responses.....	22
4.10 Request Article Registration.....	23
4.10.1 Examples.....	23
4.11 Request Account Transaction.....	23

Author Scanflow	No. SCO-SW-106	
Subject Scanflow Cash Register Protocol (SCRP)	Date 2013-03-27	Rev. 1.9

1	4.12	Print to Receipt (CR-Printing only)	25
2	4.13	Query Receipt (SFU-Printing only)	25
3	4.14	Round-up Account	25
4	4.15	Resume Operation	25
5	4.16	Request Account EndTotal	26
6	4.17	Request print of Receipt Hardcopy	26
7	4.18	Cash Register Reset	27
8	4.19	Certified Weighing	27
9	4.19.1	Response 214	27
10	4.19.2	Examples	28
11	4.20	Data Dictionary	29
12	4.21	Transport Interface	30
13	5	CASH PAYMENT	31
14	5.1	Cash Handling Machine support.....	31
15	5.2	TM_CASH Transaction Method.....	31
16	5.3	Definition of terms.....	31
17	5.4	Examples	31
18	6	EAN ATTRIBUTES	34
19	6.1	Available POS EAN attributs.....	34
20	6.2	Overview of Articles Categories and required POS EAN attributes	35
21	6.3	Loyalty Cards.....	36
22	6.3.1	example.....	37
23	6.4	Loyalty Points	37
24	6.5	Refund Coupons	37
25	6.6	E-vouchers	38
26	6.7	Account Cancel with Identification	38
27	6.8	Article Return (with Identification).....	38
28	6.9	Discount Articles.....	39
29	6.9.1	General.....	39
30	6.9.2	Changes.....	39
31	6.9.3	Examples	40
32	6.9.4	Examples	40
33	6.9.4.1	Age restricted articles.....	40
34	6.9.4.2	Weight articles	40
35	6.9.4.3	Blocked articles	41
36	6.9.4.4	Crate articles	41
37	6.10	Disclaimer	41
38	7	FLAWS.....	42
39	7.1	SIGNON SIGNOFF.....	42
40	7.2	Idle	43
41	7.3	Customer	44
42	8	RATIONALE.....	45
43	8.1	General.....	45
44	8.2	Issues.....	45
45	8.2.1	Cash Register Signing.....	45
46	8.2.2	Simultaneous Accounts.....	45
47	8.2.3	Transaction Methods	45
48	8.2.4	Receipt Printing.....	45
49	8.2.5	Protocol analogy with SMTP/FTP	46

Author Scanflow	No. SCO-SW-106	
Subject Scanflow Cash Register Protocol (SCRP)	Date 2013-03-27	Rev. 1.9

Author Scanflow	No. SCO-SW-106	
Subject Scanflow Cash Register Protocol (SCRP)	Date 2013-03-27	Rev. 1.9

1 GENERAL

1.1 Introduction

This document is the official specification of the Scanflow Cash Register Protocol (SCRP). The SCR protocol is the generic interface between a Scanflow Unit (SFU) and a Cash Register (CR). The term 'generic' reflects to the objective of interface uniformity for all Scanflow selfscanning derivatives, including PRO and TWINFLOW systems.

1.2 Document History

Revision	Date	By	Comment	SCRP version
0a	2002-04-23	Scanflow	Initial draft revision.	0.1
0b	2002-06-05	Scanflow	Major update after various reviews: - added sign on/off facility (new commands SIGNON and SIGNOFF); - added state-synchronization mechanism; - renamed state PAYING to AS_TRANSING; - added state AS_ENDING (new command IDLE); - separated response for display and printer; - added CR-printing / SFU-printing facility (new commands PRINT and RECEIPT); - added facility to report events; - altered account states reading-out; - renumbered result codes; - change in format of ARTID response; - added rationale.	0.2
1	2002-06-14	Scanflow	Minor changes: - responses extended with free text; - added section on 'Error state & Resuming' - added CR Variable to control store/recall barcode printing; - special meaning event type EV_ERROR; added RESUME command; - additional 'negative' response that indicates CR Error state; - modified response to GET command; - added password capability to SIGNON command.	1.0
1.1	2009-04-29	Johan Vereijken	Merging SCR documents. This document replaces: lp_is_0100 (older versions) lp_is_0104 lp_is_0108 lp_is_0109	2.0
1.2	2009-05-04	Johan Vereijken	Updated	2.0

Author Scanflow	No. SCO-SW-106
Subject Scanflow Cash Register Protocol (SCRP)	Date 2013-03-27
	Rev. 1.9

1.3	2009-06-23	T.J. Dekker	Added <amount> syntax to ARTREG, ARTRET and 212 response.	2.1
1.4	2010-04-20	Johan Möller	Added command RESETCR	2.1
1.5	2010-09-02	Johan Möller	Changed the text and the Figure 1 under chapter 2.1 Background.	2.1
1.5	2010-09-02	Johan Möller	Changed the header layout.	2.1
1.6	2011-03-16	Johan Möller	Added attribute "fardel" and "upr" to ARTID response 213.	2.1
1.7	2011-04-11	M. Stjärnås	Added support for certified weighing via CERTDATA command.	2.1
1.8	2011-04-21	Patrik Lundh	Updated the CERTDATA command	2.1
1.9	2011-10-17	Johan Möller	Updated the CERTDATA command with 531 Invalid account state and some more examples.	2.1

Author Scanflow	No. SCO-SW-106	
Subject Scanflow Cash Register Protocol (SCRP)	Date 2013-03-27	Rev. 1.9

2 OVERVIEW

2.1 Background

In a traditional check-out concept, the cashier has control over the Cash Register. With use of a keypad and barcode scanner, control actions and (article) data are entered by the cashier. Responses from the Cash Register are directed to a display and/or receipt printer for human interpretation.

In the Scanflow selfscanning concepts, the check-out is controlled entirely by an SFU, which means that a Scanflow Unit incorporates the functionality of the cashier and some of its traditional input and output devices. Refer to Figure 1 for an overview.

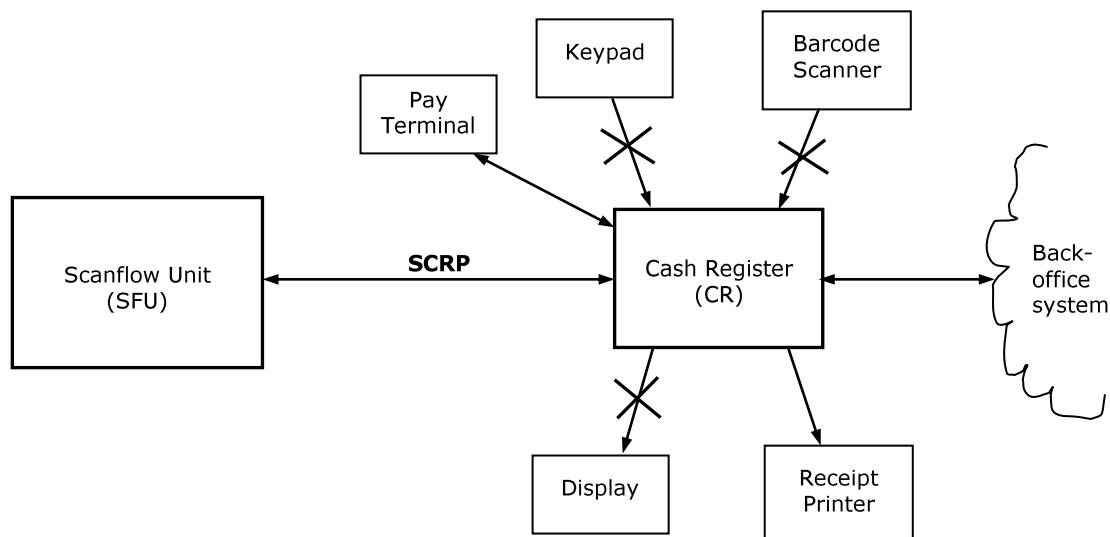


Figure 1 Scanflow Unit - Cash Register interfacing

As a consequence, a Scanflow-style Cash Register has no need for a keypad or barcode scanner input, nor it needs visual outlets towards a display. The resulting "Embedded" Cash Register is to perform the following traditional major tasks, of which the specification is beyond the scope of this document:

- managing the customer's article registration account;
- servicing article identification requests, considering linked articles and mix-match conditions;
- servicing article registration requests;
- handling the (electronic/cash) payment transaction;
- maintaining article registration journals, e.g. for daily turnover reports in the traditional way;
- manage printing of receipts;

In order to enable an SFU to perform its selfscanning task and to take over control of the check-out, the SCRCP protocol is designed.

Author Scanflow	No. SCO-SW-106	
Subject Scanflow Cash Register Protocol (SCRP)	Date 2013-03-27	Rev. 1.9

2.2 Terminology

Client system

A system that requests service from a server system in order to perform a certain task. A client system must therefore 'connect' to an appropriate server. In the context of this document, an SFU is a client system.

CR, Cash Register

A system or service (might be a process running on a host system) performing article accounting and payment transactions. A CR is typically part of some kind of administrative network ('back-office') from which it obtains its data.

FTP, File Transfer Protocol

Protocol for accessing files and file systems over the internet.

Pay Terminal

Device that enables (electronic) payment; is connected with a host system via a certified secure interface.

Scanflow

The Scanflow product family of selfscanning solutions.

SCRP, Scanflow Cash Register Protocol

Set of definitions that allows any SFU to interface with a CR in a well-defined manner.

Selfscanning

A concept of self-service in which a check-out is automated by a machine, partly controlled by the user of the selfscanning system (i.e. the customer).

Server system

A system that provides service to a client system in order to allow the client to perform a certain task. A server system must therefore 'accept' appropriate clients to connect. In the context of this document, a CR is a server system.

SMTP, Simple Mail Transfer Protocol

Protocol for transferring e-mail over the internet.

SFU, Scanflow Unit

A ITAB Scanflow selfscanning check-out system. In the context of this document, an SFU can be a TwinFlow unit as well as a MoveFlow unit.

2.3 References

- [1] Klensin, J., *Simple Mail Transfer Protocol*, RFC 2821, ISI, April 2001.
- [2] Postel, J. and Reynolds, J., *File Transfer Protocol (FTP)*, RFC 959, ISI, October 1985.
- [3] Postel, J. and Reynolds, J., *Telnet Protocol Specification*, RFC 854, ISI, May 1983.

Author Scanflow	No. SCO-SW-106	
Subject Scanflow Cash Register Protocol (SCRP)	Date 2013-03-27	Rev. 1.9

3 SPECIFICATION

3.1 General

The SCRCP protocol specification is divided in three parts, each describing the interface at the respective level. The first part describes the conceptual interface of the protocol and answers the question "What has to be done". The second and third part of the specification describe the application and transport interface of the protocol and give answers to the question "How is it done".

3.2 Major Concepts

3.2.1 Cash Register Signing

Traditionally, a CR needs to be 'signed on' before operation and 'signed off' afterwards. In some CR implementations, signing off a CR is required to enable a backoffice system to perform additional administrative and/or maintenance tasks. The SCRCP protocol facilitates signing in the following way.

The CR can be in one of the signing states listed in Table 1. The actual signing state is maintained by the CR, which reports it in the CR Variable CS_SIGN (see section 3.3.4). The SFU must synchronize *periodically* to the actual signing state by polling this variable at an appropriate rate. Both CR and SFU are allowed to control the signing state. The SFU controls signing with the commands SIGNON and SIGNOFF (see section 4).

Signing state	Description
SS_ON	CR is signed on
SS_OFF	CR is signed off, but able to be signed on
SS_HALT	CR is signed off and unable to be signed on

Table 1 List of possible CR signing states

The choice between SS_OFF and SS_HALT is determined by the CR. If the actual signing state is SS_OFF, a subsequent SIGNON command must be honoured. However, if SS_HALT is reported, a SIGNON command is expected to be rejected, hence does not make sense. In order to unaffected customer/system behavior, the CR should not sign off autonomously before all Registration Accounts are in state AS_IDLE.

3.2.2 Registration Accounts

The SCRCP protocol incorporates the concept of an account, maintained in the CR on behalf of the SFU, onto which articles can be registered. Registered articles form the basis for the amount for which the customer is charged. A registration account can be in state AS_IDLE, AS_OPEN, AS_CLOSED, AS_TRANSING or AS_ENDING, see Figure 2.

When the CR service is started, the account is typically initialized to AS_IDLE. An OPEN command (see section 4) issued by the SFU opens the account for registration. Registration is only possible in this state. After registration is finished, the CLOSE command must be invoked. In the AS_CLOSED state, a TRANS command initiates a (payment) transaction. This command must be re-issued until the transaction

Author Scanflow	No. SCO-SW-106	
Subject Scanflow Cash Register Protocol (SCRP)	Date 2013-03-27	Rev. 1.9

succeeds. A successful transaction causes the account to enter the AS_ENDING. In this state, the IDLE command makes the account AS_IDLE, i.e. available to be opened for the next session.

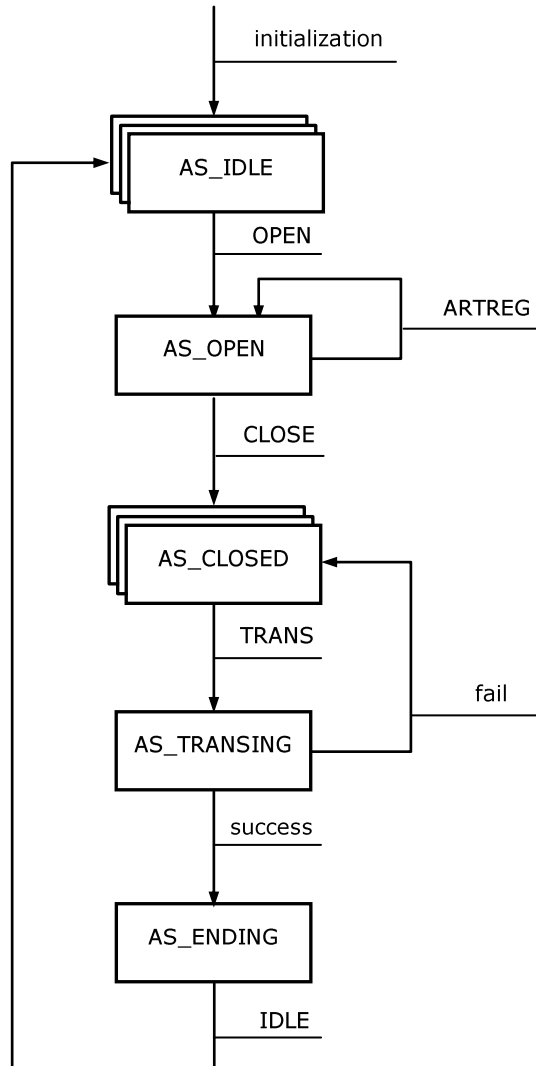


Figure 2 Registration Account State Machine

Depending on the (higher-level) system specification, it may be required that the CR supports Simultaneous Accounts. Accounts are defined to behave 'simultaneous' if multiple accounts are available for an SFU, and one account can be opened while others are (still) in state AS_CLOSED, AS_TRANSING or AS_ENDING. By definition, states AS_IDLE and AS_CLOSED are the only 'repository' states, meaning that at a given time, more than 1 account can be in those states. At maximum 1 account can be in state AS_OPEN at any instance: articles can only be registered to *one* account. At the same time, at maximum 1 account can be in state AS_TRANSING or AS_ENDING: transactions and rounding-up are performed *sequentially*, on the 'oldest' account from AS_CLOSED state.

Author Scanflow	No. SCO-SW-106	
Subject Scanflow Cash Register Protocol (SCRP)	Date 2013-03-27	Rev. 1.9

To handle recovery after a sudden power-down or reset of the CR and/or SFU, the following is defined. In stead of initializing to AS_IDLE, the CR service is allowed to proceed with the account states of before the power-down or reset. After connecting to the CR (see section 3.3), the SFU must synchronize at least *once* to the actual account states. More synchronizations are allowed, although redundant, since the life-cycle of the state machine is predictable. Synchronization is performed by querying the CR Variable CS_ACCNT in a prescribed manner (see section 3.3.4.2).

3.2.3 Transaction Methods

A (payment) transaction has to take place before an account turns from state AS_CLOSED to AS_ENDING. The capabilities of a CR in terms of transaction methods are defined by the CR Variables TC_BANK, TC_CASH, TC_STORE and TC_FLUSH, see section 3.3.4. These variables specify whether bank, cash, store or flush transactions are possible, respectively. At least one of these must be enabled for the SFU to operate properly.

TC_BANK and TC_CASH are intended to specify *local* transaction capabilities. That is, the transaction possibilities offered right after the account has been closed, for which the SFU user/customer has to make its choice instantaneously, nearby the SFU. If, on the other hand, payment is to take place *remote* (e.g. at a 'Customer Service Desk'), a 'payment store' transaction must be invoked. Such transaction effectively stores the account for recall later on by another CR. This is useful to release the account of a CR connected to an SFU and to let the payment take place at another location. If TC_STORE is enabled, the store/recall mechanism between CRs can be carried out at the responsibility of the CR implementation.

If TC_STORE indicates that the CR is not able to perform a store transaction, while remote payments are part of the system specification, a work-around can be employed. In this scenario, accounts must be stored/recalled 'externally', which is realized at the responsibility of the SFU network. In order for this to work properly, the CR must be capable to perform a 'flush transaction', effectively flushing the account involved. This capability is indicated by an enabled TC_FLUSH.

Next to the existing bank transaction method, an *alternative bank* transaction method is supported (BANK2). The CR must indicate its capability to handle this by having CR Variable "TC_BANK2" set to TRUE.

Consequently, the list of transaction methods/failure indications of Table 6, is extended with TM_BANK2. The semantics covered by TM_BANK and TM_BANK2 is not specified by the SCRCP protocol.

3.2.3.1 Bank Transaction with Signature Upon Receipt

Some bank/credit cards used for executing a bank transaction (either BANK or BANK2) require the customer to put a signature upon the receipt for self-authentication. For supporting these so-called 'Signature Upon Receipt' (SUR) cards, the transaction command "TRANS" (see section 4.11) is extended with the following positive response:

242 Transaction succeeded, needing SUR

This response must be returned by the CR after a successful TM_BANK or TM_BANK2 transaction executed with a SUR card. The "Numeric Order List" given at section 3.3.3.2 is extended with this new response.

Author Scanflow	No. SCO-SW-106	
Subject Scanflow Cash Register Protocol (SCRP)	Date 2013-03-27	Rev. 1.9

3.2.4 Error state & Resuming

If needed, the CR can turn into an Error state. The CR must report the reason for entering this state by generating an event of type EV_ERROR (see 3.3.4.1). When a command cannot be processed by the CR due to its Error state, the CR responds with a dedicated response (see 4.1).

The SFU must instruct the CR to (attempt to) continue operation by issuing the RESUME command. Only in Error state, this command has effect. The CR must not resume autonomously (i.e. without the RESUME command).

3.2.5 Display Receipt contents

Data on the display receipt of the SFU is obtained from the CR as side-effect of commands issued by the SFU. Two commands can generate these side-effects: ARTREG and CLOSE. When registering an article, 'display responses' must be produced by the CR that contain the receipt data associated with the article. Closing an account will typically initiate some kind of calculation in the CR to obtain the total amount. At any instance, the contents of the display responses must be consistent with the number of articles and (sub)totals responded on the commands mentioned.

3.2.6 Receipt Printing

Receipts are printed by a printer which is connected to the CR or to the SFU. The SCRCP protocol supports printing from either two systems, denoted as CR-printing and SFU-printing respectively. Refer to section 8.2.4 for considerations regarding the selection between these.

The printing capabilities of the CR service are reported in the CR Variables PC_CR and PC_SFU, for CR-printing and SFU-printing respectively. At least one of these must be enabled.

3.2.6.1 CR-Printing

When printing is carried out by the CR, the CR is allowed to print whatever and whenever it deems appropriate, with the following conditions:

1) The SFU has the opportunity to print additional textual messages and/or specific barcode(s) to the receipt. The PRINT command, with its argument in HTML-style (see section 3.2.6.4), is used for this purpose. The account state at the time of a PRINT command determines the location of the data produced on the receipt, see Table 2.

Account state	Print location on receipt
AS_IDLE	None, print data ignored
AS_OPEN	At the beginning, preceding the header
AS_CLOSED	At the end, following the footer
AS_TRANSING	
AS_ENDING	

Table 2 Account state vs. Print location on receipt

Author Scanflow	No. SCO-SW-106	
Subject Scanflow Cash Register Protocol (SCRP)	Date 2013-03-27	Rev. 1.9

2) The CR is responsible for storage and reproduction of all printer data in case of recovery after a sudden power-down or reset, or in case of a printer failure.

3) When printing in state AS_IDLE, the CR must ignore the <account> argument which is required by the PRINT command.

4) The IDLE command issued in state AS_ENDING is a.o. the trigger for the CR to release/cut the receipt paper.

3.2.6.2 SFU-Printing

When the SFU takes care of printing, the basic idea remains that the CR is responsible for the format and contents of the receipts printed by the SFU. This choice is motivated by the following:

- CRs are traditionally responsible for this task; most systems have tooling to maintain the format;
- in hybrid installations where both SFU systems and traditional CRs (that print receipts themselves) are operational, a uniform receipt format is better manageable;
- it simplifies the SCRCP protocol.

In state AS_ENDING, the SFU issues the RECEIPT command. The multiline response (see 3.3.2) to this command specifies the entire receipt contents in HTML format. The SFU post-processes the receipt to add its 'own' data to it and prints the receipt. The IDLE command is a.o. the acknowledge of the SFU to the CR that the receipt has been correctly printed. Up to this command, the CR is responsible for storage and regeneration of its receipt data in case of recovery after a sudden power-down or reset. This regeneration is performed by another RECEIPT command.

3.2.6.3 Account number & Store/Recall barcode

Using CR Variable MS_SRBC, the CR indicates whether or not it requires the SFU to issue PRINT commands to produce a store/recall barcode when appropriate.

When a store/recall barcode is printed on the receipt, the barcode number is derived from the account number. The CR can specify a fixed prefix for these barcodes using CR Variable MS_BCPFX. The SFU formats prefix and account number into an EAN13 barcode with the following rules:

- an array of 12 characters is zeroed;
- MS_BCPFX is left-aligned into the array;
- the account number is right-aligned into the array;
- the EAN13 checkdigit is calculated and appended to the array.

If MS_SRBC is FALSE, the value of MS_BCPFX is not used.

3.2.6.4 HTML-style data format

Receipt data, exchanged between CR and SFU, is formatted using a subset of the HyperText Markup Language (HTML). The following subset must (at least) be supported:

- 1 non-proportional font;

Author Scanflow	No. SCO-SW-106	
Subject Scanflow Cash Register Protocol (SCRP)	Date 2013-03-27	Rev. 1.9

- 3 font sizes (small, middle, large);
- bold, italic and underlined typefaces;
- left, centered and right alignment;
- tables;
- bitmaps.

With the following extension:

- barcode specification: `<barcode type="EAN13">contents</barcode>`

3.3 Application Interface

3.3.1 General

The SCRCP protocol involves communication between a client system (the SFU) and a server system (the CR). Many elements of the protocol are kept analogous to the Simple Mail Transfer Protocol (SMTP) [1] and/or the control connection of the File Transfer Protocol (FTP) [2].

As with SMTP and FTP, SCRCP defines the exchange of commands and responses at the application (3+) layer, while a lower layer takes care of the actual transport thereof. This section defines the interface at the application layer.

3.3.2 Command/Response Format

Communication between an SFU and a CR is based on a command/response dialog. Commands are from SFU to CR and responses vice versa. The SFU (SCRCP client) takes initiative to connect to the CR (SCRCP server) before communication can take place.

Both commands and responses are line-oriented variable length character strings, terminated by an End-of-Line <EOL> character sequence. The maximum line length, excluding <EOL> sequence, is 512 characters. The receiver of a command or response must take no action on a partly received line until <EOL> is received. A (new) command must not be issued before the response on a previous command has been received. Commands and responses are composed of characters from the ISO8859-15 "ASCII" character set.

SCRCP commands themselves consist of alphabetic characters followed by arguments, if applicable. Command and arguments are separated by a blank space <SP>; arguments themselves are separated by a colon `:`. Commands must be interpreted in a case-insensitive manner.

Responses consist of a three-digit alphanumerical result code (indicating failure or success) followed by a list of arguments, if applicable, and –depending on the result code– by a descriptive text string. Result code and arguments are separated by <SP> unless otherwise specified; arguments themselves are separated by a colon `:`. The text string is preceded by <SP>. If a text string is specified, the last argument in the list does not contain <SP> characters itself, so parsing the response string is rational. Any text string may optionally be extended with some free text. Prescribed text string and free text must be separated with the character sequence <SP>#.

Result code and arguments are for automated use by the client to determine what state to enter next, while the text string intended for human interpretation only. A strict implementation of this specification must not treat this string as optional, though.

Author Scanflow	No. SCO-SW-106	
Subject Scanflow Cash Register Protocol (SCRP)	Date 2013-03-27	Rev. 1.9

In the interest of improved interoperability, SCRП receivers are encouraged to tolerate superfluous <SP> characters following the command or result code or preceeding <EOL>. In order to improve backwards compatibility with future extensions, SCRП receivers should tolerate more command/response arguments than specified, in which case the additional arguments will be appended to the current lists (before the text string) and separated by a colon `:`.

In certain cases, a response may cover multiple lines. The SCRП format for multiline responses is identical to the definition of SMTP. That is, multiline responses require that every line, except the last, begins with the same (appropriate) result code immediately followed by a hyphen '-', also known as a minus, followed by arguments. The last line will –as usual– have <SP> or <EOL> following the result code. For example:

```
123-First line
123- Second line with leading spaces
123-234 text beginning with numbers
123 Last line
```

3.3.3 Result Coding

3.3.3.1 Rules

The SCRП result coding scheme partly follows the definition for FTP. The three digits of the result code each have a special significance. The first digit denotes the essence of the related action:

```
2yz  Ok – Command has been accepted and the action is successfully completed. A new
      command may be issued.
4yz  Again – Command was not a accepted as a result of a temporary error condition. It is
      encouraged to request the same action again.
5yz  Fail – Command was not accepted as a result of a permanent error condition. It is
      discouraged to request the same action again.
```

The following function categories are encoded in the second digit:

```
x0z  Syntax – Command contained a syntax error (50z), or default indication if no other
      function category applies (10z, 20z, 30z, 40z).
x1z  Information – Response related to access of information.
x2z  Connection – Response related to connection/service.
x3z  Account – Response related to account management.
x4z  Transaction – Response related to (payment) transaction.
x5z  Signing – Response related to sign on/off operations.
x6z  Printing – Response related to receipt printing.
```

The third digit gives a finer gradation in each of the function categories, as will be illustrated in the next paragraph.

3.3.3.2 Numeric Order List

```
201 Resumed operation
202 Cash Register restored
210 <cr_variable>:<cr_value>
211 <description>:<price>[:<weight_flag>]:[:<age_flag>]:[:<block_flag>]:[:<cont_barcode>]
212 <description>[:<price>]:[:<amount>]]
```

Author Scanflow	No. SCO-SW-106	
Subject Scanflow Cash Register Protocol (SCRP)	Date 2013-03-27	Rev. 1.9

213 desc="<description>" price=<price> [{<*Attribute>}] [<Group_flag>]]
 214 <weight>:<weight_price>:<price>:<barcode>
 220 SCRP Service ready
 221 SCRP Service terminating
 230 <endtotal> Account closed
 231 <account> Account opened
 232 <nr_articles>:<subtotal> Article registered
 233 Account idled
 240 Transaction succeeded
 250 Signed Off
 251 Signed On
 260 Data printed
 261 <html_text>
 270
 450 Signing rejected
 500 Unknown command
 501 Syntax error
 502 Command failed
 503 Error state
 510 No such variable
 511 No such article
 512 No stable weight
 530 No such account
 531 Invalid account state
 540 No such transaction method
 541 Busy transacting
 542 <failure_ind> Transaction failed
 550 Not signed on
 551 Authentication failed
 560 CR-printing inactive
 561 SFU-printing inactive

3.3.4 Cash Register Variables

A CR providing SCRP service incorporates a set of 'variables' accessible via SCRP command GET (see 4). Table 3 gives an overview of these variables. Variable names must be interpreted in a case-insensitive manner. Variables are static (steady, non-altering values) unless otherwise specified.

Category	Variable	Type	Description
System Information	SI_MODEL	<pr_string>	Identification of the CR (hardware) model
	SI_SOFTW	<pr_string>	Version of the CR software
	SI_SCRP	<si_scrp>	Version of the SCRP protocol
	SI_SYSID	<pr_string>	Identification/number of the system/check-out
Locale	LC_CNTRY	<pr_string>	Country code according internet TLD assignments (UK, IT, NL, NO, etc...)
	LC_CURCY	<pr_string>	Currency conform the ISO 4217 standard
	LC_SEPCH	<pr_char>	Currency separator character

Author Scanflow	No. SCO-SW-106	
Subject Scanflow Cash Register Protocol (SCRP)	Date 2013-03-27	Rev. 1.9

	LC_FRACT	<dec_nr>	Number of fractional characters
	LC_WEIGHT	<pr_string(1,6)>	Unit of weight. (GR, OUNCE, LBS...)
Transaction Capability	TC_BANK	<bool>	CR capability of bank transactions
	TC_BANK2	<bool>	CR capability of <i>alternative bank</i> transactions
	TC_CASH	<bool>	CR capability of cash transactions
	TC_STORE	<bool>	CR capability of store transactions
	TC_FLUSH	<bool>	CR capability of flush transactions
Printing Capability	PC_CR	<bool>	CR capability of direct printing receipts itself
	PC_SFU	<bool>	CR capability of supporting printing at the SFU
Miscellaneous Settings	MS_SRBC	<bool>	Indication whether store/recall barcode (number) must be printed by the SFU or not.
	MS_BCPFX	<string>	Prefix of store/recall barcode number
CR State	CS_EVENT*	See 3.3.4.1	List of CR events
	CS_SIGN*	<pr_string>	Actual signing state of the CR (see section 3.2.1)
	CS_ACCNT*	See 3.3.4.2	List of account states
Customer information ¹	AI_STAMP*	See 6.4	Loyalty points.

* These variables are dynamic, i.e. can alter during operation.

Table 3 Overview of CR Variables

3.3.4.1 Reading out Event List

The CR can report events using CR Variable CS_EVENT. Events must not be generated as replacement of a response to a command. Instead, the facility is intended for reporting side-effects or asynchronous ('external') events, e.g. "paper-out". The SFU must query the variable at an appropriate rate. Events are formatted as follows:

```
<ev_type>[:<text>]
```

 └─ Event text

 └─ Event type defined by Table 4

Event type	Description	CR/SFU behavior
EV_NONE	No event / list exhausted	Unchanged

¹ A Cash Register account can be associated with a customer in case the customer can be identified. The customer can, for example be identified through a loyalty card.

Author Scanflow	No. SCO-SW-106	
Subject Scanflow Cash Register Protocol (SCRP)	Date 2013-03-27	Rev. 1.9

EV_INFO	Informational event	
EV_WARN	Warning event	System remains operational; system administrator informed by SFU.
EV_ERROR	Error event	Implies that CR is in Error state. System is halted; system administrator alarmed by SFU.

Table 4 Event type definition

3.3.4.2 Reading out Account states

Reading CR Variable CS_ACCNT is used to obtain a list of all (simultaneous) accounts and their context. Each read returns one entry of the *cyclic* list. The format of the entries is as follows:

<dec_nr>:<ac_state>:<state_specific_context>

↓ ↓ ↓

Defined by Table 5

One of the account states

Descending sequence number, starting with a value representing the number of accounts

A sequence number of '1' indicates the tail of the cyclic list (i.e. the last account). Accounts must be sorted in descending order of age. That is, the 'oldest' account first. See the examples next.

Account state	Context
AS_IDLE	None
AS_OPEN	<account>:<nr_articles>:<subtotal>
AS_CLOSED	<account>:<nr_articles>:<endtotal>
AS_TRANSING	<account>:<nr_articles>:<subtotal>:<tr_method>
AS_ENDING	

Table 5 Account state contexts

Example 1:

A CR has three simultaneous accounts: one account is in AS_OPEN state while another one is still in AS_TRANSING, hence this one is older. A third account is not in use at the moment (AS_IDLE). The first three reads of CS_ACCNT produce the following response:

```
3 : AS_TRANSING : 20392 : 15 : 23.98 : TM_BANK
2 : AS_OPEN : 20393 : 2 : 2.95
1 : AS_IDLE
```

If CS_ACCNT is read again, the same (or an updated) list would be produced, starting from the first line.

Author Scanflow	No. SCO-SW-106	
Subject Scanflow Cash Register Protocol (SCRP)	Date 2013-03-27	Rev. 1.9

Example 2:

A CR has only one account, which is currently closed. Each attempt to read CS_ACCNT produces the following response:

1 : AS_CLOSED : 20394 : 8 : 17.45

4 COMMAND/RESPONSE OVERVIEW

4.1 General Responses

The following responses can, when appropriate, be generated as a response on any command.

500 Unknown command	In case the command is not recognized.
501 Syntax error	In case the command contains a syntax error (e.g. mandatory argument is missing).
502 Command failed	As a last resort, when no other appropriate response can be given.
503 Error state	In case the CR is in Error state and the requested command cannot be processed due to that (see 3.2.4).
504 Weighing not available	This response should be used in the case a Cash register has a scale configured, but no connection between the Cash register and the scale is established, the Cash register must go to an error state and send a 504 error (weighing not available).
550 Not signed on	If the CR requires to be signed on for the command involved.

4.2 Initialize SCRP Service

<i>Command</i>	: None; a transport connection suffices.
<i>Description</i>	: Initializes the SCRP service on the CR. The SFU must synchronize to the CR after the connection has been established.
<i>Response</i>	: 220 SCRP Service ready
<i>Versions</i>	: 1.0, 1.2, 2.0

4.3 Terminate SCRP Service

<i>Command</i>	: QUIT
<i>Description</i>	: Requests the CR to smoothly terminate the SCRP service. Although the SFU should wait for all accounts to be AS_IDLE before issuing this request, the CR must honour it in any state. Any non-IDLE account will be processed by the CR in an implementation defined manner. The CR remains ready to accept a (new) SFU connection.
<i>Response</i>	: 221 SCRP Service terminating
<i>Versions</i>	: 1.0, 1.2, 2.0

Author Scanflow	No. SCO-SW-106	
Subject Scanflow Cash Register Protocol (SCRP)	Date 2013-03-27	Rev. 1.9

1 4.4 Cash Register Sign On

Command : SIGNON <SFU_id>[:<password>]
Description : Requests the CR to sign on, making the actual signing state SS_ON. The CR is allowed to reject the request if the current signing state is SS_HALT.
Response : 251 Signed On
450 Signing rejected
551 Authentication failed
Versions : 1.0, 1.2, 2.0

4 4.5 Cash Register Sign Off

Command : SIGNOFF
Description : Requests the CR to sign off, making the actual signing state SS_OFF or SS_HALT. The CR is allowed to reject the request if not all accounts are in AS_IDLE state. Otherwise, any non-IDLE account will be processed by the CR in an implementation defined manner.
Response : 250 Signed Off
450 Signing rejected
Versions : 1.0, 1.2, 2.0

6 4.6 Get CR Variable

Command : GET <cr_variable>
Description : Requests to get the value of a certain CR Variable. The CR must honour this request in any state of the system. Possible variables and their values are listed in Table 3.
Response : 210 <cr_variable>:<cr_value>
510 No such variable
Versions : 1.0, 1.2, 2.0

8 4.7 Open Account

Command : OPEN [<account>|<barcode>]
Description : Opens an account for subsequent article registrations. The CR is requested to perform this action before registration of articles commences. If the CR implementation is such that the account state is "default opened" after the previous payment sequence, the command may internally have no effect but a positive response is still required. A positive response must report the identity of the account opened.
If <account> or <barcode> is specified (both may be used alternatively), the command requests to *recall* the associated stored account. The format of <barcode> complies to the specification in section 4.20 and is the same as used by the CR to produce a store/recall ticket.
A preferred implementation of this command returns an account number that is identical to the requested <account> (or identical to the account number from which <barcode> was constructed). Returning another account number is discouraged but allowed.
The newly opened account must be put in state AS_OPEN and must inherit all account data from the account that is recalled, including the *state specific context* as described in section 5.3.
Response : 231 <account> Account opened

Author Scanflow	No. SCO-SW-106	
Subject Scanflow Cash Register Protocol (SCRP)	Date 2013-03-27	Rev. 1.9

530 No such account
531 Invalid account state
Versions : 1.0, 1.2, 2.0

4.8 Close Current Account

Command : CLOSE
Description : Closes the current account for further registration. The CR is requested to perform this action before a transaction sequence is initiated. A positive response must include the total amount involved. The response may be preceded by a series of 'display responses', explaining the calculation from the previous subtotal to the current endtotal (e.g. mix-match calculation results).
Response : { 212 <description>[:<price>[:<amount>]] }
230 <endtotal> Account closed
531 Invalid account state
Versions : 1.0, 1.2, 2.0

4.9 Request Article Identification

Command : ARTID <barcode>
Description : Requests identification of an article, with <barcode> representing its key. This command is *not* a request to register the article. The CR must honor this request in any account state.
A positive response must be given by means of a *211 response* or a *213 response*, their usage is alternative, which means that is up to the CR to use one or the other to respond to a given ARTID command. However systems using the certified weighing capability of the Cash Register must use the *213 response*.
Compared to the *211 response*, the *213 response* allows the CR to provide a full-scale specification of the attributes that characterize an article. A detailed definition of these responses including the error responses, is given hereunder.
The length of the responses may never be larger than 512 characters.
Response : 211
213
Versions : 1.0, 1.2, 2.0

The "Request Article Identification" command has 2 positive responses. The 211 and the 213 response. Note that the 211 response is deprecated and shouldn't be used anymore.

4.9.1 Response 211

Author Scanflow	No. SCO-SW-106	
Subject Scanflow Cash Register Protocol (SCRP)	Date 2013-03-27	Rev. 1.9

Response : 211 <description>:<price>:[<weight_flag>]:[<age_flag>]:[<block_flag>]:[<cont_barcode>]

where:

<description>	Article description in text format. ²
<price>	Price of the article. It can be negative (see 6.5). ¹
<weight_flag>	Indication whether the price is weight-related (1) or piece-related (omission or 0); optional. ¹
<age_flag>	Indication whether age restrictions apply to this article (1) or not (omission or 0); optional. ¹
<block_flag>	Indication whether the article is to be blocked (1) or not (omission or 0); optional. ¹
<cont_barcode>	List of barcodes of articles contained in this article (combination pack); optional. ¹

Note: The 211 response is deprecated and should not be used anymore, the 213 response is the preferred response.

4.9.2 Response 213

Response : 213 desc="<description>" price=<price> [{<*Attribute>}] [<Group_flag>]

where:

<description>	Article description in text format. ¹
<price>	Price of the article. It can be negative (see 6.5). ¹
<*Attribute>	Optional: a unique list of attributes as described in the attribute table separated by a space.
<Group_flag>	Optional: ONE of the group flags as described in the group flag table.

Attribute	Type	Description
age	bool_flag	Age limited article: age restrictions apply to this article.
blk	bool_flag	Blocked article: article must be rejected (i.e. must not be processed by the SFU).
crtid=<barcode>	barcode	Article sold by crate: article can be optionally sold by crate. The associated crate is identified by <barcode>, which is either the EAN or PLU of that crate.
dsc=<barcode>	barcode	Discount articles: article is sold for a price that differs from that of the original (base) article identified by <barcode>, which is the EAN of the original (base) article). See 6.9 for further information.
wgt or wgt=<value>	bool_flag	Weight article: article has a weight-related price (i.e. price is determined by weight). This Attribute has 2 possible ways of usage: a) wgt, as a weight flag, in case weighing cannot be done by the Cash register because no scale is attached to it.

² Format specified in section 4.20 Data Dictionary.

Author Scanflow	No. SCO-SW-106	
Subject Scanflow Cash Register Protocol (SCRP)	Date 2013-03-27	Rev. 1.9

	dec_nr	b) wgt=<value>, defining a weight article with its certified weight information in units (specified by LC_WEIGHT, see Cash Register Variables). This response should always be given if a the product is a Weight article and a scale is connected to the Cash register.
swc	bool_flag	Skip weight-check article: article must be processed by the SFU performing limited weight validation.
fardel	bool_flag	Articles which represent a burden(fardello): Articles sold in packs and large amounts. Pack size and amount can be manually added and the articles do not need to be put on the belt/scale.
upr	bool_flag	Unit priced articles: Articles that has unit price, but still is packed in one bag. Only EasyFlow needs this flag.

Table 6 Attribute table

Group flag	Type	Description
cou	bool_flag	Coupon: article is a Coupon.
crt	bool_flag	Crate: article is a crate.
evo	bool_flag	e-voucher: article is an e-voucher.
lty	bool_flag	Loyalty card: article is a Loyalty Card.
rfd	bool_flag	Refund coupon: article is a Refund Coupon.
vsf	bool_flag	Vensafe product card: article is a Vensafe Product Card of a Vensafe vending machine.
wlz	bool_flag	Wurlitzer receipt: article is a receipt of a Wurlitzer vending machine.

Table 7 Group flag table

Note 1 : <Attribute_flag> and <Group_flag> format can be either of type:

<bool_flag>	::=	TRUE when present, FALSE when omitted.
<barcode>	::=	see definition at 4.20 Data Dictionary.
< dec_nr >	::=	see definition at 4.20 Data Dictionary.

Note 2 : The SFU allows extensions to the list of supported 213 response flags, which can be useful to e.g. meet specific needs of customer retail organization. Therefore, the set of flags defined here above must be seen as a snap-shot of the common set taken at the time this document was edited.

4.9.3 Error Responses

Response : 511 No such article

Author Scanflow	No. SCO-SW-106	
Subject Scanflow Cash Register Protocol (SCRP)	Date 2013-03-27	Rev. 1.9

1 4.10 Request Article Registration

Command : ARTREG [<barcode>[:<amount>]]

Description : Requests registration of an article, with <barcode> representing its key and <amount> representing the amount of articles to register. This request is not issued before an account has been opened successfully. A positive response must consist of the following elements:

- <nr_articles> Actual number of articles registered.
- <subtotal> Actual subtotal of articles registered.

The response must be preceded by a series of 'display responses' with all data related to the registration of this article (e.g. linked articles, discounts, etc...).

If <barcode> is omitted, the command requests the complete list (as a series of 'display responses') of articles registered in the current account, which might be a recalled account.

In case of a recalled account for which a partial payment was carried out on the originating (stored) account, the list of 'display responses' must include descriptive lines explaining the results of the partial payment(s) done; all in consistence with the value and definition of *subtotal*, see section 5.3.

If <amount> is omitted, it will default to 1.

Response : { 212 <description>[:<price>[:<amount>]] }
232 <nr_articles>:<subtotal> Article registered
511 No such article
531 Invalid account state

Versions : 1.0, 1.2, 2.0

2 4.10.1 Examples

(buying a single item)

```
C ARTREG 54491014
R 212 COCA COLA:0.34
R 232 1:0.34 Article registered
```

(buying a 6-pack)

```
C ARTREG 54491014:6
R 212 COCA COLA:2.04:6
R 232 6:2.04 Article registered
```

(buying multiple crates with linked items)

```
C ARTREG 54491012:2
R 212 CRATE AMSTEL BEER:17.38:2
R 212 CONTAINER DEPOSIT:7.80:2
R 232 4:25.18 Article registered
```

22 4.11 Request Account Transaction

Command : TRANS <tr_method>>[:<amount>]

Description : Requests transaction of the 'oldest' account from repository state AS_CLOSED. The account must be closed before the request can be honoured. For details regarding transaction methods, refer to section 3.2.3. A negative response (failure) must include an indicator that specifies the cause of the failure. Table 8 lists an overview of possible

Author Scanflow	No. SCO-SW-106	
Subject Scanflow Cash Register Protocol (SCRP)	Date 2013-03-27	Rev. 1.9

failures for each transaction method. Both transaction method and failure indication must be represented as ASCII strings and interpreted in a case-insensitive manner. If *<amount>* is omitted, the actual *endtotal* is assumed implicitly. Otherwise, *<amount>* indicates the requested transaction amount. The format of the *<amount>* argument is identical to that of *<price>* (see Data Dictionary of section 4.20).

The semantics of *<amount>* in relation to transaction methods TM_STORE and TM_FLUSH is undefined; in case *<amount>* is specified for these transaction methods, the response must report Failure Indication FI_AMOUNT (see table below).

After a successful transaction, the remaining *endtotal* must be updated accordingly and in correspondence with the applicable exchange calculation rules that reside in the CR; this *endtotal* value is also returned in the (new) "240"-response. The resulting *endtotal* is by definition equal to 0 if the transaction was successful and the *<amount>* argument was omitted, even for TM_STORE and TM_FLUSH.

Different from what is specified in the Account State Machine of Figure 2 the account state remains AS_CLOSED after a successful transaction while the resulting *endtotal* is unequal to 0. This allows for repeated/split TRANS requests.

Response : 240 Transaction succeeded
531 Invalid account state
540 No such transaction method
541 Busy transacting
542 *<failure_ind>* Transaction failed

Versions : 1.0, 1.2, 2.0

Transaction Method	Failure Indication	Cause Description
TM_BANK	FI_NOSUP	Bank transactions not supported (TC_BANK must be FALSE)
	FI_TECH	Technical problem in bank transaction system
	FI_LIMIT	Bank account does not suffice to withdraw requested amount
	FI_CANCEL	Operation is cancelled
	FI_READ	Failed to read card information (try again)
	FI_CARDTYPE	Inappropriate card type
	FI_AUTH	Invalid payment authorization
TM_BANK2	FI_NOSUP	Bank transactions not supported (TC_BANK2 must be FALSE)
	FI_TECH	Technical problem in bank transaction system
	FI_LIMIT	Bank account does not suffice to withdraw requested amount
	FI_CANCEL	Operation is cancelled
TM_CASH	FI_NOSUP	Cash transactions not supported (TC_CASH must be FALSE)
	FI_TECH	Technical problem in cash transaction system
	FI_CANCEL	Operation is cancelled
TM_STORE	FI_NOSUP	Store transactions not supported (TC_STORE must be FALSE)
	FI_TECH	Technical problem in store mechanism

Author Scanflow	No. SCO-SW-106
Subject Scanflow Cash Register Protocol (SCRP)	Date 2013-03-27
	Rev. 1.9

TM_FLUSH	FI_NOSUP	Flush transactions not supported (TC_FLUSH must be FALSE)
<all>	FI_AMOUNT	Inappropriate amount specification

Table 8 List of transaction methods / failure indications

Transaction Method	Failure Indication	Cause Description
--------------------	--------------------	-------------------

4.12 Print to Receipt (CR-Printing only)

Command : PRINT <account>:<html_text>
Description : Requests the CR to print HTML-style data to the receipt of the specified account. Refer to section 3.2.6.1 for more details.
Response : 260 Data printed
530 No such account
531 Invalid account state
560 CR-printing inactive
Versions : 1.0, 1.2, 2.0

4.13 Query Receipt (SFU-Printing only)

Command : RECEIPT
Description : Queries the CR to generate a multiline response with the entire receipt of the account currently in AS_ENDING state. Refer to section 3.2.6.2 for more details.
Response : { 261 <html_text> }
531 Invalid account state
561 SFU-printing inactive
Versions : 1.0, 1.2, 2.0

4.14 Round-up Account

Command : IDLE
Description : Requests rounding-up of the account in AS_ENDING state. A successful response makes the account state AS_IDLE. In case of CR-printing, this command may cause the CR to activate the receipt cutter. With SFU-printing, this is the acknowledge to the CR that the receipt has been correctly printed.
Response : 233 Account idled
531 Invalid account state
Versions : 1.0, 1.2, 2.0

4.15 Resume Operation

Command : RESUME

Author Scanflow	No. SCO-SW-106	
Subject Scanflow Cash Register Protocol (SCRP)	Date 2013-03-27	Rev. 1.9

Description : Requests the CR to continue operation from Error state. Once the CR is in Error state, this command is needed to resume normal operation, see 3.2.4 for more details
Response : 201 Resumed operation
503 Error state
Versions : 1.0, 1.2, 2.0

4.16 Request Account EndTotal

Command : ENDTOT
Description : Requests to calculate the actual *endtotal* for the current account. The request is only valid in state AS_OPEN. The response must be preceded by a series of 'display responses', explaining the difference between the current *subtotal* and the calculated *endtotal*. As such, the responses to this command are similar to those of the "CLOSE" command. In contrast with the "CLOSE" command, "ENDTOT" must not affect the account state and may be issued repeatedly.
Response : { 212 <description>[:<price>[:<amount>]] }
230 <endtotal> Account endtotal
531 Invalid account state
Versions : 1.2, 2.0

4.17 Request print of Receipt Hardcopy

Command : RHCOPY [<account>|<barcode>]
Description : Requests the CR to print a hardcopy of a receipt. If <account> or <barcode> is omitted, requests to print a hardcopy of the *last* printed receipt. This function is mandatory. A preferred implementation, however, must also support the arguments <account> or <barcode> (both may be used alternatively), in which case the command requests to print a hardcopy of the receipt associated to the specified account. The format of <barcode> complies to the specification in section 4.20 and is syntactically the same as used by the CR to produce a store/recall ticket. It is preferred to accept this command in all/most account states recognized by the SCRCP protocol. Accepting this command in account state AS_IDLE is required, though. The receipt hardcopy must consist of all CR-generated print data for the designated account, as well as all data that was PRINT-ed for that account on behalf of the SFU.
Response : 260 Data printed
530 No such account
531 Invalid account state
560 CR-printing inactive
Versions : 1.2, 2.0

The CR must indicate its capability to accept this command in the *current* account state by setting the CR Variable PC_RHCOPY. Table 3 of is therefore extended as follows:

Author Scanflow	No. SCO-SW-106
Subject Scanflow Cash Register Protocol (SCRP)	Date 2013-03-27
	Rev. 1.9

Category	Variable	Type	Description
Printing Capability	PC_RHCOPY*	<bool>	CR capability of accepting the RHCOPY command in the current account state

* This variable is dynamic, i.e. can alter during operation.

4.18 Cash Register Reset

Command : RESETCR
Description : Requests the CR to execute a reset. The function will only be possible to execute manually from within the SFU.
Response : 202 Cash Register restored
531 Invalid account state
Versions : 2.1

4.19 Certified Weighing

To fully comply with the certified weighting rules the SCRП protocol is extended with a special command. It's not enough to use ARTID 213 response to be fully compatible. This solution is based on weight or price embedded into the barcode. The special certified commands are followed by a normal ARTID/ARTREG scenario to actually register the article based on the barcode produced.

The CR needs to be physically connected to a scale that are certified. The certified CR is responsible to gather and calculate all weight priced information (weight, weight price & price). The server system need to answer and provide requested information as fast as possible (less than 100ms) to make sure that the human verification interaction do not feel unresponsive.

Command : GET CERTDATA <barcode>
Description : This command requests certified weight data from the CR to be presented on the screen for human verification. The format of <barcode> complies with the specification in section 4.20 but is missing the embedded weight or price information. The Command can only be issued during the state AS_OPEN.
Response : 214
512 No stable weight
531 Invalid account state
Versions : 2.1

4.19.1 Response 214

Response : <weight>:<weight_price>:<price>:<barcode>

where:

<weight> Certified weight of article.³
<weight_price> Price per weight unit.³
<price> Total price of the article. It can be negative (see 6.5).³
<barcode> Article identification.³

³ Format specified in section 4.20 Data Dictionary.

Author Scanflow	No. SCO-SW-106	
Subject Scanflow Cash Register Protocol (SCRP)	Date 2013-03-27	Rev. 1.9

4.19.2 Examples

(Checkout of a weight priced article (weight embedded barcode))

```

C  ARTID 2395412800000
R  213 desc=" Apple Pink Lady" price=10.00 wgt
C  GET CERTDATA 2395412800000
R  214 0.200:10.00:2.00:2395412802006
C  ARTREG 2395412802006
R  213 desc="Apple Pink Lady"

```

(Checkout of a weight priced article (price embedded barcode))

```

C  ARTID 2095412900000
R  213 desc=" Apple Granny Smith" price=8.00 wgt
C  GET CERTDATA 2095412900000
R  214 0.200:8.00:1.60:2095412901609
C  ARTREG 2095412901609
R  213 desc="Apple Granny Smith"

```

(Checkout of a weight priced article – constant polling)

```

C  ARTID 2395412800000
R  213 desc=" Apple Pink Lady" price=10.00 wgt
C  GET CERTDATA 2395412800000
R  214 0.00:10.00:0.00:2395412801979
C  GET CERTDATA 2395412800000
R  214 0.200:10.00:2.00:2395412802006
C  ARTREG 2395412802006
R  213 desc="Apple Pink Lady"

```

(Cancellation of a weight priced article)

```

C  ARTID 2395412800000
R  213 desc=" Apple Pink Lady" price=10.00 wgt
C  GET CERTDATA 2395412800000
R  214 0.200:10.00:2.00:2395412802006

```

(Checkout of a non weight priced article)

```

C  ARTID 54491014
R  213 desc="Coca cola" price=5.00
C  GET CERTDATA 54491014
R  502 Command failed

```

(Checkout of a weight priced article – unable to communicate with scale)

```

C  ARTID 2395412800000
R  213 desc=" Apple Pink Lady" price=10.00 wgt
C  GET CERTDATA 2395412800000
R  502 Command failed

```

(Checkout of a weight priced article – incorrect state)

```

C  ARTID 2395412800000
R  213 desc=" Apple Pink Lady" price=10.00 wgt

```

Author Scanflow	No. SCO-SW-106	
Subject Scanflow Cash Register Protocol (SCRP)	Date 2013-03-27	Rev. 1.9

C GET CERTDATA 2395412800000

R 531 Invalid account state

(Checkout of a weight priced article – unable to gather stable weight at first)

C ARTID 2395412800000

R 213 desc=" Apple Pink Lady" price=10.00 wgt

C GET CERTDATA 2395412800000

R 512 No stable weight

C GET CERTDATA 2395412800000

R 512 No stable weight

C GET CERTDATA 2395412800000

R 512 No stable weight

C GET CERTDATA 2395412800000

R 214 0.200:10.00:2.00:2395412802006

C ARTREG 2395412802006

R R 213 desc="Apple Pink Lady"

4.20 Data Dictionary

Freestyle Backus-Naur Form (FBNF)

```

<account>          ::= <dec_nr(1,8)>
<ac_state>         ::= AS_IDLE|AS_OPEN|AS_CLOSED|AS_TRANSING|AS_ENDING
<age_flag>         ::= <bool>
<amount>          ::= <dec_nr>
<barcode>         ::= <dec_nr>
<block_flag>      ::= <bool>
<bool>            ::= 0|1|FALSE|TRUE
<char>            ::= *Any ISO8859-15 character*
<cont_barcode>    ::= [<barcode>]{/<barcode>}
<cr_value>        ::= <string>
<cr_variable>     ::= *Any variables listed in Table 3
<dec_char>        ::= 0|1|2|3|4|5|6|7|8|9
<dec_nr(a,b)>     ::= a{<dec_char>}b
<description>     ::= <pr_string(1,20)>
<endtotal>        ::= <price>
<ev_type>         ::= *Any event type listed in Table 4*
<failure_ind>     ::= *Any failure indication listed in Table 8*
<html_text>       ::= *HTML subset as described in Section 3.2.6.4*
<major_nr>        ::= <dec_nr>
<minor_nr>        ::= <dec_nr>
<nr_articles>     ::= <dec_nr>
<pr_char>         ::= *Any ISO8859-15 character in range 0x20, 0x21 0x23 .. 0x39, 0x40..0x7E,
                    0xA0..0xFF. Reserved are 0x22 ( " ) and 0x3A ( : ), they should not be used*
<pr_string(a,b)>  ::= a{<pr_char>}b
<price>           ::= [-]<price_unit><LC_SEPCH><price_fraction>
<price_fraction> ::= <dec_nr(<LC_FRACT>)>
<price_unit>      ::= <dec_nr>
<si_scrp>         ::= <major_nr>.<minor_nr>
<SFU_id>          ::= <dec_nr(1,8)>
<string(a,b)>     ::= a{<char>}b
<subtotal>        ::= <price>
<text>            ::= <string>

```

Author Scanflow	No. SCO-SW-106	
Subject Scanflow Cash Register Protocol (SCRP)	Date 2013-03-27	Rev. 1.9

1 <tr_method> ::= *Any transaction method listed in Table 8*
 2 <weight> ::= <dec_nr><LC_SEPCH><weight_fraction>
 3 <weight_fraction> ::= <dec_nr(<LC_FRACT>)>
 4 <weight_price> ::= <price>
 5 <weight_flag> ::= <bool>

4.21 Transport Interface

11 The SCRCP protocol relies on a transport layer to take care of the actual transfer of application commands
 12 and responses. For this, SCRCP uses the Telnet protocol [3] on top of TCP/IP (as with the FTP control
 13 connection). Not the entire specification of Telnet is employed; instead, SCRCP complies with the key
 14 elements of Telnet, needed to assure basic interoperability between both.

Connection Setup

16 An SCRCP server (the CR) must listen for TCP connections on port 25801. It is up to the CR designer
 17 whether or not multiple simultaneous connections are accepted. However, each connection must provide
 18 independent SCRCP service as specified in this document. The SCRCP client (the SFU) initiates a connection
 19 with the server on the port mentioned and, once accepted by the server, expects the server to send a
 20 greeting message '220'.

Communication

24 The <EOL> character sequence used to terminate a command or response line is the ASCII <CR>
 25 character immediately followed by ASCII <LF>, together denoted as <CRLF> in this document. In order to
 26 comply with the Telnet protocol and to provide a transparent communication path, the following special
 27 precautions have to be made:

- 1) An SCRCP sender must insert a <NULL> character after each <CR> that is not part of an intended <EOL>. Complementary, the SCRCP receiver must strip each <NULL> that follows a <CR>. In this way the Telnet protocol is satisfied in that it requires individual <CR> characters to be followed by <NULL>, while it avoids a <CRLF> character sequence as part of a command or response to be misinterpreted as <EOL>.
- 2) The Telnet Command structure defines the "Interpret As Command" character (<IAC>, 0xFF) as escape character to invoke specific control operations. An SCRCP sender must insert an <IAC> character after each <IAC> occurrence in the command/response strings. Complementary, the SCRCP receiver must strip each second <IAC> in a double <IAC> character sequence before the data is evaluated.

Connection Breakdown

42 Both SCRCP client and server can terminate a connection. Client and server must both tolerate the abrupt
 43 loss of a connection, e.g. due to a sudden power down of one of the systems. In the ideal case, the SCRCP
 44 client requests the server to smoothly close the connection, which the server honours after sending the
 45 confirmation response '221' to the client.

Author Scanflow	No. SCO-SW-106	
Subject Scanflow Cash Register Protocol (SCRP)	Date 2013-03-27	Rev. 1.9

5 CASH PAYMENT

5.1 Cash Handling Machine support

This section describes all changes needed in the SCRCP protocol to facilitate direct connection of a Cash Handling Machine (CHM) to an SFU. A CHM allows customers to pay their account with cash money; as such, the CHM acts as a functional replacement of the traditional cash drawer in a CR.

The changes described here facilitate the following particular payment operations:

- *split payments (intra account)* allowing customers to pay partially with cash money and partially with a bank card;
- *partial payments (inter account)* allowing customers to pay partially on one system and partially on another system;
- *cash back (intra/inter account)* allowing customers to pay extra money with a bank card and getting this amount in cash returned.

One of the key aspects in the specified changes is that they guarantee *backwards compatibility* in the following sense: SFU systems that work with SCRCP implementations without CHM support will work correctly with SCRCP implementations including CHM support. In other words, CR systems with an SCRCP implementation that includes CHM support can be installed as a replacement for the ones that do not include CHM support.

5.2 TM_CASH Transaction Method

In order to inform the CR about mutations made to the Cash Handling Machine's money deposit, TM_CASH is used. The CR must indicate its capability to accept this transaction method by having CR Variable "TC_CASH" set to TRUE. The *<amount>* specification of TM_CASH transactions (see 4.11) is positive in case of cash deposits and negative in case of cash dispenses. A 'cancel' of a certain deposit operation is realized by a dispense of the same amount, and vice versa.

5.3 Definition of terms

- subtotal* Reflects the *running summation* of individual article prices, decreased by any (partial) payment done on the originating (stored) account; *<subtotal>* might be negative, indicating excessive payment (e.g. cash back).
- endtotal* Summation of the current *<subtotal>* and any applicable discount calculation (e.g. *mix-match*), decreased by any (split) payment yet carried out on the current account. As such, it represents the resulting amount for which the customer is (still) to be charged; *<endtotal>* might be negative, indicating excessive payment (e.g. cash back).

5.4 Examples

1 – A customer recalls its account – onto which two items (wine and cheese) were registered – and adds a third item to the recalled account. Next, the customer chooses to pay by bank and to have a certain amount of *cash back*. The following fragment shows a typical SCRCP dialog.

C OPEN 2888800102562

Customer recalls its account.

Author Scanflow	No. SCO-SW-106	
Subject Scanflow Cash Register Protocol (SCRP)	Date 2013-03-27	Rev. 1.9

```

1      R 231 090331 Account opened
2      C GET CS_ACCNT
3      R 210 CS_ACCNT:1:AS_OPEN:090331:2:12.98
4      C ARTREG
5      R 212 WINE:9.98
6      R 212 CHEESE:3.00
7      R 232 2:12.98 Article registered
8      C GET CS_ACCNT
9      R 210 CS_ACCNT:1:AS_OPEN:090331:2:12.98
10     C ENDTOT
11     R 212 WINE&CHEESE DISCOUNT:-1.00
12     R 230 11.98 Account endtotal
13     C GET CS_ACCNT
14     R 210 CS_ACCNT:1:AS_OPEN:090331:2:12.98
15     C ARTREG 8710408013862
16     R 212 WINE&CHEESE DISCOUNT:1.00
17     R 212 CABALLERO:3.55
18     R 232 3:16.53 Article registered
19     C GET CS_ACCNT
20     R 210 CS_ACCNT:1:AS_OPEN:090331:3:16.53
21     C ENDTOT
22     R 212 WINE&CABALL.DISCOUNT:-2.00
23     R 230 14.53 Account endtotal
24     C GET CS_ACCNT
25     R 210 CS_ACCNT:1:AS_OPEN:090331:3:16.53
26     C CLOSE
27     R 230 14.53 Account closed
28
29     C GET CS_ACCNT
30     R 210 CS_ACCNT:1:AS_CLOSED:090331:3:14.53
31
32     C TRANS TM BANK:34.53
33     R 240 -20.00 Transaction succeeded
34     C GET CS_ACCNT
35     R 210 CS_ACCNT:1:AS_CLOSED:090331:3:-20.00
36     C TRANS TM_CASH:-20.00
37     R 240 0.00 Transaction succeeded
38
39     C PRINT 090331:<p align="center">Scanflow TWINFLOW - SCO Ticket</p>
40     R 260 Data printed
41     C IDLE
42     R 233 Account idled

```

These two items where registered into this account.

There was a discount for these (combination of) items.

Rewind of discount statement.

Some other (more attractive) discount.

Doesn't make sense to repeat discount; is allowed if rewound first.

2 - Same situation; customer pays by cash a rounded amount (14.53 -> 14.55) in one way.

```

45     C TRANS TM_CASH:15.00
46     R 240 -0.47 Transaction succeeded
47     C TRANS TM_CASH:-0.45
48     R 240 0.00 Transaction succeeded

```

3 - Same situation; customer pays by cash a rounded amount (14.53 -> 14.55) in another way.

Author Scanflow	No. SCO-SW-106	
Subject Scanflow Cash Register Protocol (SCRP)	Date 2013-03-27	Rev. 1.9

```

1      C  TRANS TM_CASH:14.50
2      R  240 0.03 Transaction succeeded
3      C  TRANS TM_CASH:0.05
4      R  240 0.00 Transaction succeeded

```

4 – Same situation; customer is not capable to pay entire amount and wants to (re-)store the account.

```

9      C  TRANS TM_BANK:10.00
10     R  240 4.53 Transaction succeeded
11     C  TRANS TM_STORE
12     R  240 0.00 Transaction succeeded

```

5 – Same situation; system is not capable of dispensing *cash back*, so account is (re-)stored.

```

17     C  TRANS TM_BANK:34.53
18     R  240 -20.00 Transaction succeeded
19     C  TRANS TM_STORE
20     R  240 0.00 Transaction succeeded

```

6 – To show backwards compatibility...

```

25     C  TRANS TM_BANK
26     R  240 0.00 Transaction succeeded

```

Author Scanflow	No. SCO-SW-106	
Subject Scanflow Cash Register Protocol (SCRP)	Date 2013-03-27	Rev. 1.9

6 EAN ATTRIBUTES

6.1 Available POS EAN attributs

EAN Attribute	Name	Significance ("what is being indicated")
age	Age restriction	Item with age restriction, requiring special attention. Based on internal configuration, several flavors are available: <i>ignore</i> , <i>signal-only</i> or <i>restrictive</i> (optional feature F068). Can additionally influence allowed payment methods and force legitimation at a Service Desk (optional features F066/F067).
bch=<barcode>	Bunch	Item that can alternatively be registered as a single article or as a wrapped bundle of articles (<i>bunch</i>). The <i>bunch</i> is identified by <barcode>. Relates to EAN attribute 'btn'.
blk[=<text>]	Blocked	Item is blocked, i.e. not to be registered. Message <text> is used to inform the customer and overrides a default message.
btn=<text>	Button	When dealing with a <i>bunch</i> item, <text> is the name of the button to be pressed to select the bunch. Relates to EAN attribute 'bch'. If omitted, a default text (e.g. "CRATE") is used.
byb	By-passed	Item that is registered by-passing security check, requiring special attention (e.g. using the Scanflow MONITOR).
cnf=<number>	Confirmation	Item that requires customer's confirmation prior to registration (e.g. by means of a question "Are you sure?") when the same item has already been registered ≥ <number> of times. Optional feature F050.
cou	Coupon	Item is some kind of (reduction) coupon that needs to be collected from the customer. This is achieved by forcing payment at the Service Desk. Optional feature F073.
dsc=<barcode>	Discount item	Item is discounted. See 6.9 for a description. Optional feature F063.
dir	Direct registration	Item is directly registered, without being passed through the security check.
epc	Exitpoint code	Exitpoint code
lty	Loyalty	Item is a loyalty card; is a.o. used for ergonomic purposes, i.e. to suppress the superfluous question "Do you have a loyalty card?".
msg=<text>	Post-registration message	Item registration is accompanied with message <text> as feedback to the customer.
pvy[=<text>]	Postponed verified registration	Item needing <i>postponed</i> verification by an attendant; system doesn't block until payment phase. Needs to be unlocked by an attendant. Message <text> overrides a default message and facilitates instruction to the customer. Optional feature F059.
rfd	Refund coupon	Item is a refund coupon; is a.o. used for ergonomic purposes, i.e. to suppress the superfluous question "Do you have a refund coupon?".
sin=<on-time>:<off-time>[:<duration>]	Service Indicator	Item requiring additional attention. Controls an (audio/visual) indicator for a period of <duration> msec (default 5000) switching on/off with a duty-cycle determined by <on-time> and <off-time> (both in msec). Only applicable for "warning state"-triggering EAN attributes, like 'blk' and 'vfy'.

Author Scanflow	No. SCO-SW-106	
Subject Scanflow Cash Register Protocol (SCRP)	Date 2013-03-27	Rev. 1.9

EAN Attribute	Name	Significance ("what is being indicated")
swc	Suppress weight check	Item is registered after a 'fictive' security check; item needs pass this security check, though. Optional feature F060.
vfy[=<text>]	Verified registration	Item needing <i>immediate</i> verification by an attendant; system doesn't block but informs both customer and attendant about this condition. Message <text> overrides a default message and facilitates instruction to the customer. Optional feature F058.
wgt	Weight item	Item with weight-related price. Item's reference weight can be calculated based on the price-per-kilo which accompanies the "213"-response (see 4.9.2).

6.2 Overview of Articles Categories and required POS EAN attributes

Category	SCO behavior	POS EAN attributes
Age limitation	Customer can't perform any kind of payment until verification by Assistant has taken place. Assistant gets warning (<i>status indicator lamps, Scanflow Monitor</i>).	age
Weight article	Price indicated by POS is assumed as "kilo price"	wgt
Crate PLU	After scanning the product, Customer is asked to choose whether to purchase: <ul style="list-style-type: none"> a single product (e.g. a bottle of beer) , by putting this article on the weight unit, or a crate of the same product, by pressing the GUI button with label "CRATE" 	bch=<barcode> btn="CRATE"
Crate (not on weight unit)	Crate undergoes direct registration. Then, the following instruction is given on the GUI: "Crate registered. Please do not put it on the belt".	dir byp msg="Crate registered. Please do not put it on the belt."
Crate (on weight unit, weight verification disabled)	Crate is register after passing the tunnel. No weight verification takes place.	swc
Crate (on weight unit, normal weight verification)	Crate is register after passing the tunnel. Normal weight verification takes place.	(nothing)
E-Voucher	After first scanning, the E-Voucher undergoes direct registration. Then, the following instruction is given on the GUI: "Voucher registered. Please do NOT put the voucher on the belt." After subsequent scanning, customer is asked for confirmation before proceeding as above.	dir cnf=1 msg="Voucher registered. Please do NOT put the voucher on the belt."
Coupon	Coupon undergoes direct registration. Then, the following instruction is given on the GUI: "Coupon registered. Please do NOT put the coupon on the belt."	dir msg=" Coupon registered. Please do NOT put the coupon on the belt."
Invalid Coupon	Coupon is not accepted. After scanning, the following instruction is given on the GUI: "INVALID Coupon. This coupon can't be used at self-service checkout.".	blk="INVALID COUPON This coupon can't be used at self- service checkout."

Author Scanflow	No. SCO-SW-106	
Subject Scanflow Cash Register Protocol (SCRP)	Date 2013-03-27	Rev. 1.9

Category	SCO behavior	POS EAN attributes
Loyalty Card	Loyalty Card undergoes <i>direct registration</i> . Then, the following instruction is given on the GUI: <i>"Card registered. Please do NOT put the card on the belt."</i> . Just before entering the payment phase of the checkout procedure and only if no Loyalty Card has been registered yet, the following prompt is given on the GUI: <i>"Do you have a Loyalty Card? Please scan it or press NO."</i>	dir lty msg="Card registered. Please do NOT put the card on the belt."
Invalid Loyalty Card	Loyalty Card is not accepted. After scanning, the following instruction is given on the GUI: <i>"INVALID LOYALTY CARD. An assistant will come to help you."</i> . Assistant gets warning (<i>status indicator lamps, Scanflow Monitor</i>).	blk="INVALID LOYALTY CARD An assistant will come to help you." sin=500:500
Refund Receipt	Refund Receipt (e.g. Reverse Vending Machine Receipt) undergoes direct registration. Then, the following instruction is given on the GUI: <i>"Receipt registered. Please do NOT put the receipt on the belt."</i> Just before entering the payment phase of the checkout procedure and only if no refund Receipt has been registered yet, the following prompt is given on the GUI: <i>"Do you have any refund receipt? Please scan it or press NO."</i>	dir rfd msg="Receipt registered. Please do NOT put the receipt on the belt."
Bonus Card with Verified Registration	To allow verification and collection, the Bonus Card undergoes direct registration only when it is scanned by an assistant. When the Bonus Card is scanned by a customer, no registration takes place and the following instruction pops up on the GUI: <i>"BONUS CARD. An assistant will come to help you."</i> . Assistant gets warning (<i>status indicator lamps, Scanflow Monitor</i>). This state is abandoned, if another article is scanned.	dir vfy="BONUS CARD An assistant will come to help you." sin=500:500
Bonus Card with Postponed Verified Registration	To allow verification and collection, the Bonus Card undergoes direct registration only when it is scanned by an assistant. When the Bonus Card is scanned by a customer, no registration takes place and the following instruction pops up on the GUI: <i>"BONUS CARD. Your card will be soon collected by an assistant. Please proceed with the rest of your articles."</i> . Assistant gets warning (<i>status indicator lamps, Scanflow Monitor</i>). Then, the checkout procedure is set to halt before entering the payment phase and until assistant intervention takes place.	dir pvy=" BONUS CARD Your card will be soon collected by an assistant. Please proceed with the rest of your articles."
Self Vending Machine Receipt	Self Vending Machine Receipt (e.g. e Vensafe Receipt) undergoes direct registration. Then, the following instruction is given on the GUI: <i>"Receipt registered. Please do NOT put the receipt on the belt."</i>	dir msg="Receipt registered. Please do NOT put the receipt on the belt."
Invalid Payment Receipt	Payment Receipt is not accepted. After scanning, the following instruction is given on the GUI: <i>"INVALID PAYMENT RECEIPT. This receipt can't be used at this system."</i> .	blk="INVALID PAYMENT RECEIPT This receipt can't be used at this system."
Exit Receipt	Exit Receipt is not accepted. After scanning, the following instruction is given on the GUI: <i>"EXIT RECEIPT. Please scan this receipt at the exit gate to leave the self service area."</i> .	epc blk="EXIT RECEIPT Please scan this receipt at the exit gate to leave the self service area."

6.3 Loyalty Cards

Identification and registration of loyalty cards follow mainly the same schemes as for normal articles, i.e. the same commands and responses can be used in the same account states *and* in state AS_CLOSED.

Author Scanflow	No. SCO-SW-106	
Subject Scanflow Cash Register Protocol (SCRP)	Date 2013-03-27	Rev. 1.9

Both commands "ARTID" and "ARTREG" can be used, producing the usual responses, with the following remarks:

- *<price>* field in "211"-response is optional
- *<nr_articles>* and *<subtotal>* fields in "232"-response must either remain unchanged or be updated, following the chosen general CR behavior;
- if the CR rejects identification or registration of a loyalty card, it must respond with "511 No such article";
- CR may indicate the cause of rejection using the Event Report facility (CR Variable "CS_EVENT").

6.3.1 example

Using 211 response

```
C ARTID 2620327350411
R 211 LOYALTY CARD
```

Using 213 response

```
C ARTID 2620327350411
R 213 desc="LOYALTY CARD" lty
```

6.4 Loyalty Points

Loyalty (or *'bonus'*) *points* is a generic name for concepts like *trading stamps* etc. In AS_CLOSED state, the SFU interacts with the CR to ask for information or to initiate operations concerning loyalty points.

The command "GET AI_STAMP" must return the response "210 AI_STAMP:<nr_of_stamps>:<price>", where *<nr_of_stamps>* equals the number of loyalty points related to the account being dealt with, and *<price>* is the corresponding amount for which the customer is charged. If *<nr_of_stamps>* is zero, no loyalty points are available; if *<price>* is zero, the specified number of loyalty points are gratis (i.e. for free).

The command "STAMPREG <nr_of_stamps>" must be used to register the specified amount of loyalty points. The response on this command is similar to the response on command "ARTREG", where *<nr_articles>* must either remain unchanged or be increased (following the chosen general CR behavior), and *<subtotal>* must be adjusted consequently.

The CR is responsible for determining the amount of loyalty points available for a certain account, with emphasis to the interrelation with loyalty cards registration.

6.5 Refund Coupons

Identification and registration of refund coupons follow mainly the same schemes as for normal articles, i.e. the same commands and responses can be used in the same account states *and* in state AS_CLOSED.

Both commands "ARTID" and "ARTREG" can be used, producing the usual responses, with the following remarks:

- *<price>* field in "211"-response logically has a negative value, i.e first character is a '-';

Author Scanflow	No. SCO-SW-106	
Subject Scanflow Cash Register Protocol (SCRP)	Date 2013-03-27	Rev. 1.9

- `<nr_articles>` field in "232"-response must either remain unchanged or be updated, following the chosen general CR behavior;
- `<subtotal>` field in "232"-response must be consequently decreased;
- if the CR rejects identification or registration of a refund coupon (e.g. because already refunded or due to some failure), it must respond with "511 No such article";
- CR may indicate the cause of rejection using the Event Report facility (CR Variable "CS_EVENT"); lack of such indication yields a system with non-explanatory rejections.

6.6 E-vouchers

Identification and registration of e-vouchers follow the same schemes as for normal articles, i.e. the same commands and responses can be used in the same account states.

Both commands "ARTID" and "ARTREG" can be used, producing the usual responses, with the following remarks:

- if the CR rejects identification or registration of an e-voucher (e.g. because already sold or due to some failure), it must respond with "511 No such article";
- CR may indicate the cause of rejection using the Event Report facility (CR Variable "CS_EVENT"); lack of such indication yields a system with non-explanatory rejections.

6.7 Account Cancel with Identification

In order for the CR / back office administration to keep track of the initiator of an account cancel operation, the SCRП command involved is extended to: "TRANS TM_FLUSH[:<id>]", where <id> is the identification of the initiator, which may be a sequence of all printable ASCII characters, except ':' and white spaces.

6.8 Article Return (with Identification)

In AS_OPEN state it is possible to return an article (i.e. inverse registration), using the specific command "ARTRET <barcode>[:<id>[:<amount>]]", where <barcode> is the article being returned and <id> is the identification of the initiator, which may be a sequence of all printable ASCII characters, except ':' and white spaces. <amount> specifies the amount of articles to return, if omitted, defaults to 1. Note that the ARTRET command can still have an <amount> attribute even if the <id> attribute is omitted, see examples below.

Responses to the "ARTRET" command are identical to those of "ARTREG".

Examples:

(returning a single item, no id specified)

```

C  ARTREG 54491014
R  212 COCA COLA:0.34
R  232 1:0.34 Article registered
C  ARTRET 54491014
R  212 -COCA COLA:-0.34
R  232 0:0.00 Article registered

```

(returning a single item, id specified)

Author Scanflow	No. SCO-SW-106	
Subject Scanflow Cash Register Protocol (SCRP)	Date 2013-03-27	Rev. 1.9

```

C  ARTREG 54491014
R  212 COCA COLA:0.34
R  232 1:0.34 Article registered
C  ARTRET 54491014:adminID
R  212 -COCA COLA:-0.34
R  232 0:0.00 Article registered

```

(returning multiple items, id specified)

```

C  ARTREG 54491014:2
R  212 COCA COLA:0.68:2
R  232 2:0.68 Article registered
C  ARTRET 54491014:adminID:2
R  212 -COCA COLA:-0.68:2
R  232 0:0.00 Article registered

```

(returning multiple items, no id specified)

```

C  ARTREG 54491014:2
R  212 COCA COLA:0.68:2
R  232 2:0.68 Article registered
C  ARTRET 54491014::2
R  212 -COCA COLA:-0.68:2
R  232 0:0.00 Article registered

```

6.9 Discount Articles

6.9.1 General

Discount articles are defined as articles that are to be sold to a lower (generally: another) price than the original (base) price. To accomplish this, discount articles are labeled with a dedicated (i.e. own/unique) discount EAN code. Article identification look-up ("ARTID") of such EAN yields data of the discount article, among which the discount price.

In order to enable Scanflow systems to do maximum security checks on discount articles, "ARTID" of discount EAN codes must additionally return the EAN of the original (base) article. This allows for cross-reference of article information which is described in detail in the coming sections.

6.9.2 Changes

Article identification look-up ("ARTID") of a discount EAN code must comply to section 4.20 with the following notes:

- *<description>*, *<price>*, *<weight_flag>*, *<age_flag>* and *<block_flag>* are elements that identify the discount article;
- *<cont_barcode>* must specify the EAN code of the original (base) article; if this EAN represents an article with weight-related price, the variable field of this EAN must be filled with the original data (i.e. price or weight).

Author Scanflow	No. SCO-SW-106	
Subject Scanflow Cash Register Protocol (SCRP)	Date 2013-03-27	Rev. 1.9

6.9.3 Examples

1 - A bottle of wine (original EAN 8030423000198 with price 9.98) is to be sold to a discount price of 4.98. Originally, this is an age restricted article, whose *<age_flag>* is inherited by the discount article. The following fragment shows a typical SCRП dialog.

```
C ARTID 20100000000045
R 211 DISCOUNT WINE:4.98:0:1:0:8030423000198
C ARTID 8030423000198
R 211 WINE:9.98:0:1:0
C ARTREG 20100000000045
...
```

2 - A piece of cheese (original EAN 2192921004965 with price 4.96) is to be sold to a discount price of 2.98. Originally, this is an article with weight-related price (e.g. kilo price is 5.23); the *<weight_flag>* is *not* inherited by the discount article, as it's price of 2.98 is not weight-related (in fact, EAN 20100000000052 doesn't allow for a variable field). See the following SCRП dialog fragment.

```
C ARTID 20100000000052
R 211 DISCOUNT CHEESE:2.98:0:0:0:2192921004965
C ARTID 2192921004965
R 211 CHEESE:5.23:1:0:0
C ARTREG 20100000000052
...
```

6.9.4 Examples

6.9.4.1 Age restricted articles

A bottle of wine (age limited article). The following responses are equivalent:

Using 211 response:

```
C ARTID 8030423000198
R 211 WINE:9.98:0:1:0
```

Using 213 response:

```
C ARTID 8030423000198
R 213 desc="WINE" price=9.98 age
```

6.9.4.2 Weight articles

A piece of cheese with unit price 5.23:

Using 211 response

```
C ARTID 2192921004965
R 211 CHEESE:5.23:1:0:0
```

Using 213 response

```
C ARTID 2192921004965
R 213 desc="CHEESE" price=5.23 wgt
```

Using 213 response with the certified weight

Author Scanflow	No. SCO-SW-106	
Subject Scanflow Cash Register Protocol (SCRP)	Date 2013-03-27	Rev. 1.9

C ARTID 2192921004965
 R 213 desc="CHEESE" price=4.00 wgt=765
 Note: price in the 213 response with a certified weight is the real price,
 not a unit price.

6.9.4.3 Blocked articles

For some reasons, the same piece of cheese as at 6.9.4.2 can no longer be sold:

Using 211 response

C ARTID 2192921004965
 R 211 CHEESE:5.23:1:0:1

Using 213 response

C ARTID 2192921004965
 R 213 desc="CHEESE" price=5.23 wgt blk

6.9.4.4 Crate articles

A bottle of wine (bottle price: 9.98, 12-bottles crate price:113.78):

Using 211 response

C ARTID 2010000004145
 R 211 ITALIAN WINE:9.98:0:1:0
 C ARTID 8030
 R 211 CRATE 12xITALIAN WINE:113.78:0:1:0

Using 213 response

C ARTID 2010000004145
 R 213 desc="ITALIAN WINE" price=9.98 age crtld=8030
 C ARTID 8030
 R 213 desc="CRATE 12xITALIAN WINE" price=113.78 age crt

6.10 Disclaimer

The POS EAN attributes given at section 6.1 can be used in a virtually unlimited number of combinations. Nevertheless, a correct behavior of the ITAB Scanflow SCO systems can be only guaranteed if these attributes are used within the combinations that are specified at section 6.2. This means that any deviation from these combinations can produce an undefined behavior of the Scanflow SCO systems. On request, and only after a proper investigation, new combinations can be defined and added to the list.

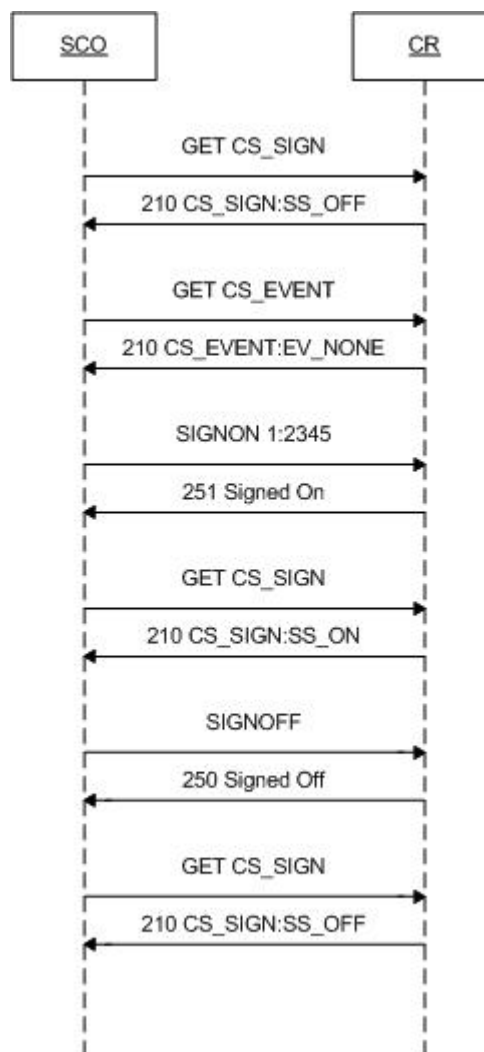
The texts specified between double quote marks (") at section 6.2 are the texts that will appear on the SCO GUI (e.g. instructions, information prompts, etc.) and should be tailored to meet the needs of the customer retail organization (language localization). When doing this, particular attention should be given to the usage of the *Carriage Return* character '\r', which controls the text formatting, that has been agreed with the customer retail organization.

Author Scanflow	No. SCO-SW-106	
Subject Scanflow Cash Register Protocol (SCRP)	Date 2013-03-27	Rev. 1.9

7 FLOWS

7.1 SIGNON SIGNOFF

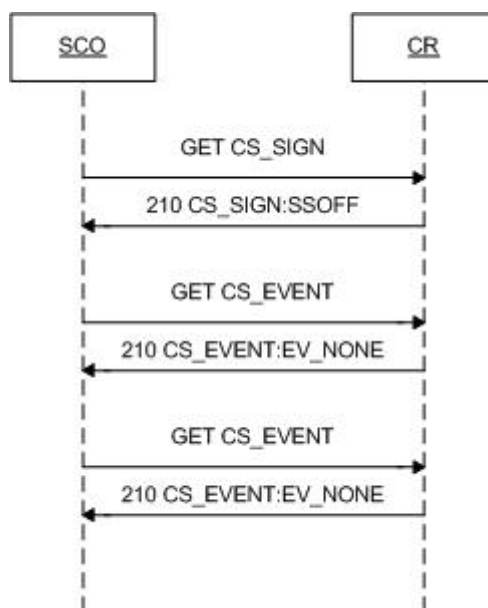
The basic SignON, SignOFF sequence is given by the figure below.



Author Scanflow	No. SCO-SW-106	
Subject Scanflow Cash Register Protocol (SCRP)	Date 2013-03-27	Rev. 1.9

7.2 Idle

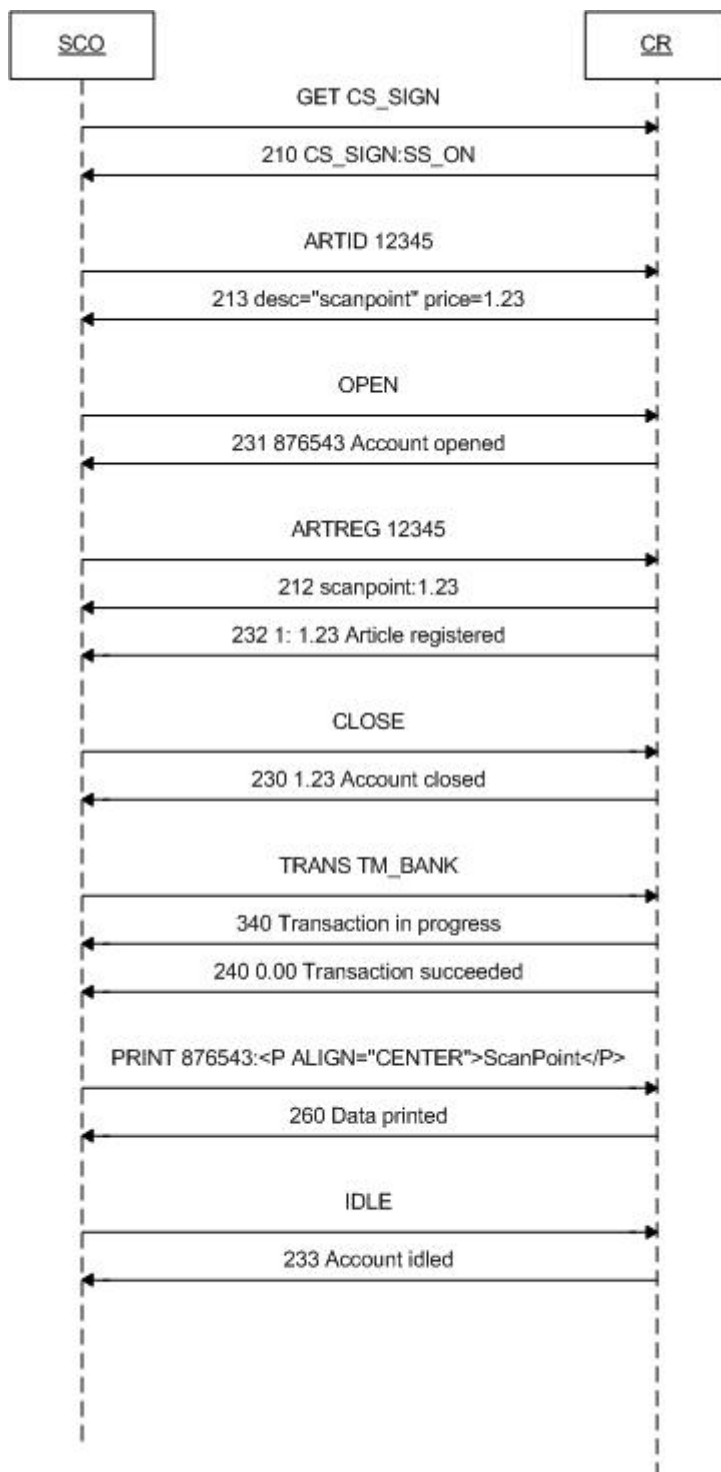
During the time the sco isn't used, the communication between the SCO and Cash Register is as follows:



The Get CS_SIGN request is send every 5 seconds and the GET CS_EVENT is send every 2 seconds.

Author Scanflow	No. SCO-SW-106	
Subject Scanflow Cash Register Protocol (SCRP)	Date 2013-03-27	Rev. 1.9

7.3 Customer



Author Scanflow	No. SCO-SW-106	
Subject Scanflow Cash Register Protocol (SCRP)	Date 2013-03-27	Rev. 1.9

8 RATIONALE

8.1 General

This section addresses some non-trivial issues of this specification and gives insight in considerations and motivations leading to the respective decisions. As such, the contents stated in this section is *not* part of the specification: in case of ambiguity, the remainder of the document is leading.

8.2 Issues

8.2.1 Cash Register Signing

The SS_HALT signing state can be used by the CR to *block* operation of an SFU: as part of an administrative action during which signing on is not allowed, or to explicitly disable the use of an SFU. The SFU can report this state with a temporary "Out of Use" message to the customer.

The SFU could be configured to perform an 'auto sign on' action when it detects the SS_OFF state. This can be useful in combination with a 'timed' sign off, carried out by the SFU or autonomously by the CR. Alternatively, the SFU can be supplied with a sign on / sign off button, accessible in e.g. maintenance or administration mode.

8.2.2 Simultaneous Accounts

Simultaneous Accounts are a solution to the problem that arises when one customer wants to use the SFU while a previous customer is still busy paying for his account, both on the *same* CR. This is the case when the electronic/cash pay terminal is located such that the customer using the terminal does not physically block access for a next customer to the system. As stated earlier, this is up to the (higher-level) system specification.

8.2.3 Transaction Methods

The set of supported transaction methods is interrelated with the physical layout of check-out area, the presence and location of electronic/cash pay terminal(s) and the issue of Simultaneous Accounts. All together they determine the *payment logistics* of the self scanning concept, which must be defined in the system specification.

8.2.4 Receipt Printing

Within the Scanflow self scanning concepts, the preferred scheme is to connect the printer to the SFU instead of the CR (see Figure 1). Flexibility is the most important motivation; the system specification may, for instance, require:

- that two printers must be connected to the system: one at the back near the pay terminal and one at the front, only used to print receipts of 'stored transactions';
- that payment is *only* possible at the Customer Service Desk, which allows the CR to be physically moved to that location, so local printing (near the SFU) cannot be carried out by the CR.

The major penalty of SFU-printing is the *possible necessity* of having the 'pay terminal - CR - SFU' combination certified by the authoritative organization. This tradeoff is likely to lead to the situation that

Author Scanflow	No. SCO-SW-106	
Subject Scanflow Cash Register Protocol (SCRP)	Date 2013-03-27	Rev. 1.9

CR-printing is implemented as a 'first step' towards integration of SCRП, which eventually results in SFU-printing to make it more future-proof.

8.2.5 Protocol analogy with SMTP/FTP

Both SMTP and FTP protocols are platform independent, proven client/server protocols with a simple human-readable command/response format. This resembles the wishes for the SCRП protocol. Testing and low-level debugging is possible with use of e.g. the Telnet client program.