

Range Servant software integration API

Introduction

This document describes how to communicate with Range Servant Select.Lite ball dispensers from a software different from our own.

Communications

The data is sent over TCP/IP to a pre-agreed port.

The data is transferred using IP Sockets.

Each message is expected to be sent in a new connection.

The data packets should be formatted in the following way:

Packet length	2 Byte Integer	The length of the entire packet including itself.
Message type	2 byte Integer	Type of message held in the message data
Message	String of message data	Can be any data dependent on message type
Termination	2 Bytes	Has to be CR LF (0x0D 0x0A) These do NOT add to length. (Some languages adds these automatically to a string)

Message format

All data transferred across this interface has message type 4.

These messages are encoded in a key = value pair string. Each key, value pair is terminated with an '&' character.

All sent characters should be encoded in the following way:

A-Z	A-Z
a-z	a-z
0-9	0-9
Space	+
All other characters	%## Where ## is the hexadecimal number for that character.

Example: If you were to send a phone number (+46 35 152954) it would look like

Phonenumber=%2B46+35+152954&

Messages

The following messages are used in the communication. The fields of each message may be sent in any order.

ValidateRangeId

Sends a request to authenticate a card number or barcode, receives validation and number of balls available for dispensing.

Variable name	Mandatory	Description
Command	Y	ValidateRangeId
RangeType	Y	Either P for pincode (or barcode) or C for card
Reference	Y	Either pincode (barcode) number or card number
Date	Y	Date (format dd/mm/yy)
Time	Y	Time (format hh:mm:ss)

The response would contain

Variable	Description
RangeOk	Equals 1 if reference is valid or 0 if not
Error	An error description if RangeOk=0
CardBalance	Only supplied if RangeType='C'
CustomerName	Customer name if available
NumberOfRowsRS	Number of enabled rows in a RS dispenser
Price1	Price for first issue group
Number1	Number of balls for first issue group
Price2	Price for second issue group
Number2	Number of balls for second issue group
Price3	Price for third issue group
Number3	Number of balls for third issue group
Price4	Price for fourth issue group
Number4	Number of balls for fourth issue group

Please note that issue group 4 is not currently used.

The dispenser will not do any comparisons between CardBalance and Prices. It will enable any Price / Number combo sent that has a ball amount > 0. (i.e negative balance is allowed, also buckets with 0 price)

We recommend using Price/Number 1 for your smallest basket, 2 for medium and 3 for large, as this will make the sizes correspond correctly to the basket sizes in the dispenser interface.

For barcodes Select.Lite supports direct dispense. This means that the customer can scan his ticket and the dispenser will start dispensing immediately without him even touching the screen. When this is used (configured in settings on the dispenser) the dispenser will use ONLY the Number1 variable and dispense that. This is enabled by default and most sites like to keep it that way. It is possible to disable and in that case the customer will be offered the Price/Numbers that have been received.

SetRangeUsage

Is sent after a successful dispense

Variable	Mandatory	Description
Command	Y	SetRangeUsage
RangeType	Y	Either P for pincode (or barcode) or C for card
Reference	Y	Either pincode (barcode) number or card number
IssueGroup	Y	The issue group used. Valid values are 1, 2, and 3.
Date	Y	Same date as sent in ValidateRangeld
Time	Y	Same time as sent in ValidateRangeld

Response

Variable	Description
RangeOk	1 if OK, 0 if not
Error	Error description if RangeOk=0

Example

An example of a full ValidateRangeld packet: (Barcode no 777777, date 10/10/10, time 00:00:00)

```
0]04Command=ValidateRangeID&RangeType=P&Reference=777777&Date=10%2F10%2F10&Time=00%3A00%3A00&(CR)(LF)
```

The ']' is the char representation of the length of the entire packet (93).

The response could look like: (valid Barcode, Price1=1.5, Number of balls=10, Price2=3, Number of balls2 = 25)

```
0:04RangeOk=1&Price1=1%2E5&Number1=10&Price2=3&Number2=25&(CR)(LF)
```

Again, the ':' is char representation of the full length of the packet (58).

The user dispenses 25 balls (issue group 2)

```
0h04Command=SetRangeUsage&RangeType=P&Reference=777777&IssueGroup=2&Date=10%2F10%2F10&Time=00%3A00%3A00&
```

Response:

```
0(S0)04RangeOk=1&(CR)(LF)
```

Other

Please take care to make sure your receiving software can handle incoming 'junk' card numbers and codes, as some times customers scans cards and codes from other sources.

Change Log

2016-05-09	Added info on barcode direct dispense
2015-03-26	Added clarification on that each transmission has to be CR LF terminated.
2014-07-08	Dispenser no longer checks CardBalance vs Prices, enabling negative CardBalances.
2014-07-02	Added the NumberOfRowsRS variable to be used with RS dispensers.