TECHNOLOGY SOLUTIONS UK LTD
part of HID

TSL ASCII PROTOCOL 2.2

www.tsl.com
Design • Development • Manufacture

CONTENT

History

Version	Date	Modifications
2.0 Rev A	29/04/2013	First Release
2.1 Rev A	25/07/2013	Added Monza extensions, Fast ID, Blockwrite and Tag focus.
		Minor document formatting changes.
		Correction to the .da example, date should input as yy not yyyy,
		Correction to the .ws and .wr commands, 32 words can be input not 16.
		Added AC: response field.
		Added *Bluetooth*® error code.
		Changed the barcode escape character and enabled it by default.
		Added BR response for barcodes when escape is disabled.
2.1 Rev B	04/10/2013	Correction to the Version Information Command – added missing 'RS:' to the valid response field headers section
		.wr command default -qv value corrected was 6 should be 4.
		Hardware Specific Connection Notes section added
2.2 Rev A	16/05/2014	Amended the .rd –dl description to include the 253 word limit.
		Added the .lk command and LK: response.
		Fixed bug in .rd command which limited reads to a maximum 2032 bits.
		Fixed dropped <CRLF> bug in log when commanded from autorun file.
		Added –rd and –rs parameters to the .sa command to control the switch repeat delay.
		Added BA: response to the .vr command

INTRODUCTION

This document describes the Technology Solutions ASCII 2.1 protocol. The commands and their parameters are detailed along with the responses. Each command has a few examples to aid understanding. The commands all have default parameters and most can be used without change.

It is important to note that this document is not a transponder tutorial and users should have a basic understanding of UHF Class 1 Gen 2 transponders.

The latest UHF Class 1 Gen 2 Standard is available for download from
http://www.gs1.org/gsmp/kc/epcglobal/uhfc1g2

"The *Bluetooth*® word mark and logos are registered trademarks owned by Bluetooth SIG, Inc. and any use of such marks by TSL is under license. Other trademarks and trade names are those of their respective owners."

COMMANDS OVERVIEW

All commands start with a period (.)(0x2E) followed by 2 lower case characters.

Commands should be terminated with a Carriage Return<CR>, a Linefeed<LF> or a combination of the two. All combinations are acceptable <CR>,<LF>,<CRLF> or <LFCR>.

Command echoing can be enabled with the .ec command. This is useful for testing commands with a terminal program but generally should be avoided particularly over the *Bluetooth*® interface where it will impact throughput.

Commands are parameterised with the parameters preceded by a minus sign (-)(0x2D)
e.g.

```
        .rd -a00000000 -bu <CRLF>
```

Some parameter values need to be enclosed in double quotes (“)(0x22).
e.g.

```
        .bt -f “1128-EU-000001” <CRLF>
```

All parameter values are stored in the reader, if a parameter is not included the stored value will be used.

When a command uses a parameter its stored value is updated.

When the reader is powered up the stored values are loaded with defaults with the exception of the .bt command whose settings are persistent.

The parameter -x restores the stored parameters to their default values.

The parameter -n prevents the command from taking any actions, other included parameters will be stored. This allows parameters to be saved without the command being performed.

The parameter -p displays the current values of the supported parameters. Any other parameters included will be stored first.

Once the command has started (. detected) a gap of greater than 30 seconds will cause an error response to be sent.

Spaces (0x20) in commands are ignored unless in quotes.

The command .<CRLF> will repeat the last command.

COMMAND PARSER

The Command Parser is fairly simple, only the expected parameters are validated, anything else will be ignored.

For example the following two commands will perform the same.

```
. al –v off <CRLF>
.al this text is ignored –v off so is this <CRLF>
```

This is quite a useful feature as it can be used to identify the source of a command.

As an example, there may be a need to distinguish between a command issued over the USB interface and one issued from a trigger press. To achieve this, the commands sent via the USB interface could have “USB” inserted into the commands and then the .CS response can be used to separate the responses.

e.g.

Command from USB:

```
.iv USB <CRLF>
```

Response to command from USB:

```
CS: .iv USB <CRLF>
EP: 310833B2DDD906C000001234<CRLF>
EP: 341486E37C00000000004255<CRLF>
EP: 341486E37C00000000004254<CRLF>
OK:<CRLF><CRLF>
```

Response to .iv command issued from a switch press:

```
CS: .iv<CRLF>
EP: 310833B2DDD906C000001234<CRLF>
EP: 341486E37C00000000004255<CRLF>
EP: 341486E37C00000000004254<CRLF>
OK:<CRLF><CRLF>
```

If individual commands need to be identified a sequential numeric identifier could be used.

RESPONSES

Responses can return multiple values so field headers are used to separate the data.

Field headers are two upper case characters followed by a colon (:) (0x3A).

Each field is terminated by a Carriage Return Linefeed pair <CRLF>.

All responses end with either "OK: <CRLF><CRLF>" or "ER:nnn<CRLF><CRLF>"

For example, a typical response to the .vr command is:

```
CS: .vr<CRLF>
MF: Technology Solutions UK ltd. <CRLF>
US: 1128-EU-000001<CRLF>
PV: V2.0.0<CRLF>
UF: V3.0.0<CRLF>
UB: V1.3.0<CRLF>
RS: 1116-ET-000001<CRLF>
RF: V2.6.0<CRLF>
RB: V1.2.0<CRLF>
AS: 1128-S1-EU-000001<CRLF>
OK:<CRLF><CRLF>
```

AB: AUTORUN FILE BEGIN

This field header indicates the beginning of the Autorun file and is terminated by <CRLF>.

Format:

AB: optional text <CRLF>

Examples:

AB: **** Beginning of Autorun file **** <CRLF>

AC: AUTHENTICATION CHIP TEST

This field header indicates the result of the test on the authentication chip and is terminated by <CRLF>.

Format:

AC: Passed/Failed <CRLF>

Examples:

AC: Passed <CRLF>

AE: AUTORUN FILE END

This field header indicates the end of the Autorun file and is terminated by <CRLF>.

Format:

AE: optional text <CRLF>

Examples:

AE: **** End of Autorun file **** <CRLF>

AS: ANTENNA SERIAL NUMBER

This field header precedes the antenna serial number which is terminated by <CRLF>.

Format:

AS: “Antenna serial number” <CRLF>

Examples:

AS: 1128-S1-EU-000001 <CRLF>

BA: *BLUETOOTH®* ADDRESS

This field header precedes the *Bluetooth®* address which is terminated by <CRLF>.

Format:

BA: "*Bluetooth®* Address" <CRLF>

Examples:

BA: 00:07:80:41:a4:c6 <CRLF>

BC: BARCODE DATA

This field header precedes the barcode value when escape characters are enabled (default) and is terminated by <CRLF>.

Since some barcodes can contain the termination characters<CRLF> the use of an escape character, <ESC> (0x1B), is used to distinguish between <CRLF> contained in the Barcode and the real termination characters. Should the escape character be read as part of the barcode it will be escaped as well.

Format:

BC: "Barcode Value" <CRLF>

Examples:

Example using escape character to read the following barcode data
"abc<ESC>123456789<CR><LF>abcdef"

BC: abc<ESC><ESC>123456789<ESC><CR><ESC><LF>abcdef <CRLF>

BP: BATTERY PERCENTAGE LEVEL

This field header precedes the battery level value which is terminated by <CRLF>.

Format:

BP: "Battery level Value" <CRLF>

Examples:

BP:100% <CRLF>

BP:78% <CRLF>

BP:4% <CRLF>

BR: BARCODE RAW

This field header precedes the barcode value when escape characters are disabled and is terminated by <CRLF>. Care must be taken since some barcodes can contain the termination characters <CRLF>.

Format:

BR: “Barcode Value” <CRLF>

Examples:

BR: ISBN:5884688-6686 <CRLF>

CH: CHARGE STATUS

This field header precedes charge status and is terminated by <CRLF>.

Format:

CH: Off/Charging/ Complete /Error<CRLF>

Examples:

CH: Complete<CRLF>

CR: TRANSPONDER EPC CRC VALUE

This field header precedes Transponder EPC CRC value (in hex) which is terminated by <CRLF>.

Format:

CR: “CRC Value” <CRLF>

Examples:

CR: 3E4F<CRLF>

CS: COMMAND STARTED

This field header is sent to indicate that a command has started. It precedes the command that initiated it and is terminated by <CRLF>.

Format:

CS: “Command” <CRLF>

Examples:

CS: bc -t 6<CRLF>

DA: DATE

This field header precedes the Date, in ISO 8601 extended format, terminated by <CRLF>.

Format:

DA: “Date” <CRLF>

Examples:

DA: 2013-01-24 <CRLF>

DP: DOUBLE PRESS USER DEFINED SWITCH ACTION

This field header precedes the action that will be executed when “.sa -d usr” is configured, it is terminated by <CRLF>.

Format:

DP: Switch command <CRLF>

Examples:

DP: .bc -t4 <CRLF>

DP: .iv <CRLF>

DT: DATE/TIME STAMP

This field header precedes the Time Stamp, in ISO 8601 extended format, terminated by <CRLF>.

Format:

DT: “Date/Time Stamp” <CRLF>

Examples:

DT: 2013-01-24T13:12:16 <CRLF>

EA: TRANSPONDER ACCESS ERROR CODE

This field header precedes the Transponder access error code and is terminated by <CRLF>. Access errors are generated by the reader while trying to access a transponder and apply to the transponder with the preceding EPC value.

Format:

EA:nnn<CRLF>   nnn = 000 to 255

Error codes:

001 Handle mismatch.
002 CRC error on transponder response.
003 No transponder reply.
004 Invalid password.
005 Zero kill password.
006 Transponder lost.
007 Command format error.
008 Read count invalid.
009 Out of retries.
255 Operation failed.

Examples:

EA:004<CRLF>

EA:006<CRLF>

EB: TRANSPONDER BACKSCATTER ERROR CODE

This field header precedes the Transponder backscatter error code and is terminated by <CRLF>.
Backscatter errors are generated by the transponder and apply to the transponder with the preceding EPC value.

Format:

EB:nnn<CRLF>   nnn = 000 to 255

Error codes:

000 General error.
003 Memory does not exist or the PC value is not supported.
004 Memory is lock or permalocked.
011 Transponder has insufficient power.
015 Transponder does not support error specific codes.

Examples:

EB:003<CRLF>

EB:011<CRLF>

EP: TRANSPONDER EPC VALUE

This field header precedes Transponder EPC value (in hex) which is terminated by <CRLF>.

Format:

EP: “EPC Value” <CRLF>

Examples:

EP: 000000000000000000010E6<CRLF>

ER: ERROR

This field header is used to terminate the response to an unsuccessful command.
It is always followed by a three digit error code then two Carriage return Linefeed pairs <CRLF><CRLF>.

Format:

ER:nnn<CRLF><CRLF>   nnn = 000 to 255

Error codes:

001 Syntax Error.
002 Parameter not supported.
003 Action not enabled.
004 Command not supported by hardware
005 No transponder found
006 No Barcode found
007 Parameter configuration invalid
008 Antenna Error (Wrong region or Not Fitted)
009 Battery level too low
010 Scanner not ready
011 Command not supported on interface
012 Command not supported from Autorun file
013 Write Failure
014 Switch already in use
015 Command Aborted
016 Lock Failure
017 *Bluetooth*® Error
018 Licence Key is not blank
255 System Error.

Examples:

ER:001<CRLF><CRLF>

ER:004<CRLF><CRLF>

FN: *BLUETOOTH*® FRIENDLY NAME

This field header precedes the *Bluetooth*® friendly name and is terminated by <CRLF>.

Format:

FN: "*Bluetooth*® Friendly Name" <CRLF>

Examples:

FN: 1128-EU-000001 <CRLF>

IX: INDEX NUMBER

This field header precedes an index number used for referencing multiple responses to a command, it is terminated by <CRLF>.

Format:

IX: 4 digit ASCII hex Index number <CRLF>

Examples:

IX: 004A <CRLF>

KS: KILL SUCCESS

This field header indicates the kill command has succeeded and is terminated by <CRLF>.

Format:

KS: Kill Success <CRLF>

Examples:

KS: Kill Success <CRLF>

LB: LOG BEGIN

This field header indicates the beginning of the log and is terminated by <CRLF>.

Format:

LB: optional text <CRLF>

Examples:

LB: **** Beginning of Log ****

LE: LOG END

This field header indicates the end of the log and is terminated by <CRLF>.

Format:

LE: optional text <CRLF>

Examples:

LE: **** End of Log **** <CRLF>

LK: LICENCE KEY

This field header precedes the licence key string and is terminated by <CRLF>.
Any printable ASCII characters are allowed.

Format:

LK: licence Key <CRLF>

Examples:

LK: IamAlicenceKey <CRLF>

LS: LOCK SUCCESS

This field header indicates the lock command has succeeded and is terminated by <CRLF>.

Format:

LS: Lock Success <CRLF>

Examples:

LS: Lock Success <CRLF>

ME: MESSAGE

This field header precedes message text and is terminated by <CRLF>.
It usually precedes the ER: field header to provide additional error information.

Format:

ME: “Message text”<CRLF>

Examples:

ME: “Vibration not supported” <CRLF>

MF: MANUFACTURER NAME

This field header precedes the manufacturers name and is terminated by <CRLF>.

Format:

MF: “Manufacturer name”<CRLF>

Examples:

MF: Technology Solutions UK ltd. <CRLF>

OK: OK RESPONSE

This field header is used to terminate the response to a successful command.
It is always followed by two Carriage return Linefeed pairs <CRLF><CRLF>.

Format:

OK:<CRLF><CRLF>

Examples:

OK:<CRLF><CRLF>

PC: TRANSPONDER EPC PC VALUE

This field header precedes Transponder EPC PC value (in hex) which is terminated by <CRLF>.

Format:

PC: “PC Value” <CRLF>

Examples:

PC: 3000<CRLF>

PR: PARAMETERS

This field header precedes parameter values which are terminated by <CRLF>.

Format:

PR: “Parameter Values” <CRLF>

Examples:

PR: -v on -b off <CRLF>

PV: PROTOCOL VERSION NUMBER

This field header precedes the protocol version number which is terminated by <CRLF>.

Format:

PV: “Protocol version number” <CRLF>

Examples:

PV:V2.0.0<CRLF>

RB: RADIO BOOTLOADER VERSION NUMBER

This field header precedes the radio bootloader version number which is terminated by <CRLF>.

Format:

RB: “Radio bootloader version number” <CRLF>

Examples:

RB:V1.2.0<CRLF>

RD: TRANSPONDER READ DATA

This field header precedes data read from a Transponder (in hex) which is terminated by <CRLF>.

Format:

RD: “Read Data” <CRLF>

Examples:

RD: 0F078010<CRLF>

RF: RADIO FIRMWARE VERSION NUMBER

This field header precedes the radio firmware version number which is terminated by <CRLF>.

Format:

RF: “Radio firmware version number” <CRLF>

Examples:

RF:V2.6.0<CRLF>

RI: TRANSPONDER RSSI VALUE

This field header precedes Transponder RSSI value in dBm which is terminated by <CRLF>.

Format:

RI: “RSSI Value” <CRLF>

Examples:

RI: -63<CRLF>

RS: RADIO SERIAL NUMBER

This field header precedes the radio serial number which is terminated by <CRLF>.

Format:

RS: “Radio serial number” <CRLF>

Examples:

RS: 1116-ET-000001 <CRLF>

SP: SINGLE PRESS USER DEFINED SWITCH ACTION

This field header precedes the action that will be executed when “.sa -s usr” is configured, it is terminated by <CRLF>.

Format:

SP: Switch command <CRLF>

Examples:

SP: .bc -t4 <CRLF>

SP: .iv <CRLF>

SW: SWITCH STATUS

This field header precedes the switch state value and is terminated by <CRLF>.

Format:

SW: “off/single/double” <CRLF>

Examples:

SW: single <CRLF>

SW: off <CRLF>

TD: TRANSPONDER TID VALUE

This field header precedes the Transponder TID value (in hex) which is terminated by <CRLF>.

Format:

TD: “TID Value” <CRLF>

Examples:

TD: 1C2B3A495867768594A3B2C1<CRLF>

TM: TIME

This field header precedes the Time, in ISO 1601 extended format, terminated by <CRLF>.

Format:

TM: “Time” <CRLF>

Examples:

TM: 13:12:16 <CRLF>

UB: UNIT BOOTLOADER VERSION NUMBER

This field header precedes the unit bootloader version number which is terminated by <CRLF>.

Format:

UB: “Unit bootloader version number” <CRLF>

Examples:

UB:V1.2.0<CRLF>

UF: UNIT FIRMWARE VERSION NUMBER

This field header precedes the unit firmware version number which is terminated by <CRLF>.

Format:

UF: “Unit firmware version number” <CRLF>

Examples:

UF:V2.6.0<CRLF>

US: UNIT SERIAL NUMBER

This field header precedes the unit serial number which is terminated by <CRLF>.

Format:

US: “Unit serial number” <CRLF>

Examples:

US: 1128-EU-000001 <CRLF>

WW: NUMBER OF WORDS WRITTEN

This field header precedes the number of words successfully written by a transponder write command and is terminated by <CRLF>.

Format:

WW: nn <CRLF>

Examples:

WW: 3 <CRLF>

FILES ON SD CARD

On wake up the default configuration is loaded. If the “auto.txt” file is present on the SD card it is executed as if sent from a host. This may overwrite some or the entire loaded configuration.

Files on SD

AUTO.TXT	Script which runs on start up
LOG.TXT	Event log file

.AL ALERT COMMAND

Configures and alerts the user.
Returns an error if vibrate and buzzer are both off.

Parameters:

-b on/off	Buzzer on or off
-d sho/med/lon	Buzzer/Vibrate duration short, medium or long
-n	No action, just set the parameters
-p	List the current parameter values
-t low/med/hig	Buzzer tone low, medium or high
-v on/off	Vibrate on or off
-x	Reset the parameters to defaults

Parameter interpretation order

The parameters are interpreted in the following order.

-x, -b, -d, -t, -v, -p, -n

Parameter defaults

The parameter defaults are:

-b on,-d sho, -t hig, -v on,

Valid response field headers:

CS: ER: ME: OK: PR:

Examples:

Command	Response	Action
.al -bon -voff <CRLF>	CS: .al -bon -voff <CRLF> OK: <CRLF> <CRLF>	Buzzer sounds
.al -p -n <CRLF>	CS: .al -p -n <CRLF> PR: -b on -d sho -n -p -t hig -v off -x<CRLF> OK:<CRLF><CRLF>	None
.al -boff -voff <CRLF>	CS: .al -boff -voff <CRLF> ME: Buzz and Vibrate are both off<CRLF> ER:003<CRLF><CRLF>	None

.AB ABORT COMMAND

Abort command, terminates the current command and any pending commands. It also stops any software switch presses that are in progress.

Parameters:

No parameters are used for this command.

Valid response field headers:

CS: ER: ME: OK:

Examples:

Command	Response	Action
.ab <CRLF>	CS: .ab<CRLF> OK: <CRLF> <CRLF>	Abort complete

.BC BARCODE COMMAND

Initiates a barcode read.
Returns an error if no barcode is read.

Parameters:

-al on/off		perform an alert after a successful barcode read using the current alert parameters from the .al command
-dt on/off		Include date time response
-e on/off		Use escape character 0x1B to precede <CR>, <LF> and itself
-n		No action, just set the parameters
-p		List the current parameter values
-tn	n=1..9	Read duration in seconds
-x		Reset the parameters to defaults

Parameter interpretation order

The parameters are interpreted and applied in the following order.

-x, -al, -e, -dt, -t,-p, -n

Parameter defaults

The parameter defaults are:

-al on, -dt off, -e on, -t 9

Valid response field headers:

BC: BR: CS: DT: ER: ME: OK: PR:

Examples:

Command	Response	Action
.bc <CRLF>	CS: .bc<CRLF> BC: ISBN:5884688-6686 <CRLF> OK: <CRLF> <CRLF>	Barcode reads
.bc -e off<CRLF>	CS: .bc -e off<CRLF> BR: ISBN:5884688-6686 <CRLF> OK: <CRLF> <CRLF>	Barcode reads with escape
.bc <CRLF>	CS: .bc<CRLF> ME: No barcode found<CRLF> ER:006<CRLF><CRLF>	Barcode fails to read
.bc -p -n <CRLF>	CS: .bc -p -n <CRLF> PR: -al on -dt off -e on -n -p -t 9 -x <CRLF> OK:<CRLF><CRLF>	None

.BL BATTERY LEVEL COMMAND

Returns Battery level as percentage and Charge status.

Parameters:

No parameters are used for this command.

Valid response field headers:

CH: CS: BP: ER: ME: OK:

Examples

Command	Response	Action
.bl <CRLF>	CS: .bl<CRLF> BP: 100% <CRLF> CH: Complete <CRLF> OK:<CRLF><CRLF>	None
.bl <CRLF>	CS: .bl<CRLF> BP: 30% <CRLF> CH: Charging <CRLF> OK:<CRLF><CRLF>	None
.bl <CRLF>	CS: .bl<CRLF> BP: 7% <CRLF> CH: Off <CRLF> OK:<CRLF><CRLF>	None

.BT *BLUETOOTH*® FUNCTIONS

Reads the *Bluetooth*® address, additionally the *Bluetooth*® can be reset and configured. This command is only available over the USB interface and cannot be used in the Autorun file.
Settings made with this command are persistent and not reset to defaults at power up.

Parameters:

-bi "bundleID"	set iOS Bundle ID, up to 80 characters, enclosed in double quotes (")
-bs "BundleSeedID"	set iOS Bundle Seed ID, 10 characters enclosed in double quotes (")
-f "name"	set the *Bluetooth*® friendly name, up to 20 characters, enclosed in double quotes (")
-p	List the available parameters (not their values)
-w nnnn	set the 4 character pairing code
-x	reset *Bluetooth*® settings to defaults

Parameter interpretation order

The parameters are interpreted and applied in the following order.

-x, -f,-w, -bi, -bs, -p

Parameter defaults

-bi "TSL demo app Bundle ID", -bs "TSL demo app Bundle Seed ID",
-f "unit serial number", -w 0000

Valid response field headers:

AC: BA: CS: ER: FN: ME: OK: PR:

Examples:

Command	Response	Action
.bt <CRLF>	CS: .bt<CRLF> BA: 00:07:80:41:a4:c6 <CRLF> FN: 1128-EU-000001 <CRLF> AC: Passed <CRLF> OK: <CRLF> <CRLF>	The *Bluetooth*® address and friendly name is read and the authentication chip is working.
.bt -w 0123 <CRLF>	CS: .bt -w 0123 <CRLF> BA: 00:07:80:41:a4:c6 <CRLF> FN: 1128-EU-000001 <CRLF> AC: Passed <CRLF> OK: <CRLF> <CRLF>	The *Bluetooth*® pairing code is set to 0123

.DA DATE COMMAND

Reads or sets the date.

Parameters:

-s yymmdd	Set the date

Parameter interpretation order

The parameters are interpreted and applied in the following order.

-s

Parameter defaults

No parameter defaults are required:

Valid response field headers:

CS: DA: ER: ME: OK:

Examples:

Command	Response	Action
.da <CRLF>	CS: .da<CRLF> DA: 2012-11-05<CRLF> OK: <CRLF> <CRLF>	The date is read
.da -s 130120<CRLF>	CS: .da -s 130120<CRLF> DA: 2013-01-20<CRLF> OK: <CRLF> <CRLF>	The date is set

.DP USER DEFINED SWITCH DOUBLE PRESS ACTION

Reads or sets the double press user action used when “.sa -d usr” is configured.

Parameters:

-s user action command	Set the user action

Parameter interpretation order

The parameters are interpreted and applied in the following order.

-s

Parameter defaults

No parameter defaults are required:

Valid response field headers:

CS: ER: ME: OK: DP:

Examples:

Command	Response	Action
.dp <CRLF>	CS: .dp<CRLF> DP: .bc<CRLF> OK: <CRLF> <CRLF>	User action is read
.dp -s .iv<CRLF>	CS: .dp -s .iv<CRLF> DP: .iv<CRLF> OK: <CRLF> <CRLF>	User action is set

.EA EXECUTE AUTORUN COMMAND

This command will force the autorun file to run.

Parameters:

No parameters are used for this command.

Valid response field headers:

CS: ER: ME: OK:

Examples:

Command	Response	Action
.ea <CRLF>	CS: .ea<CRLF> OK: <CRLF> <CRLF>	Auto.txt will be run

.EC ECHO

Determines whether commands are echoed back to the host.

Parameters:

-e on/off	Turn echo on or off
-p	List the current parameter values
-x	Reset the parameters to defaults

Parameter interpretation order

The parameters are interpreted and applied in the following order.

-x, -e, -p

Parameter defaults

The parameter defaults are:

-e off

Valid response field headers:

CS: ER: ME: OK: PR:

Examples:

Command	Response	Action
.ec -x <CRLF>	CS: .ec -x<CRLF> OK: <CRLF> <CRLF>	.ec defaults loaded
.ec -p <CRLF>	CS: .ec -p<CRLF> PR: -e off -p -x <CRLF> OK:<CRLF><CRLF>	None

.FD FACTORY DEFAULTS COMMAND

Restores all command parameters to their default settings.

Parameters:

No parameters are used for this command.

Valid response field headers:

CS: ER: ME: OK:

Examples:

Command	Response	Action
.fd <CRLF>	CS: .fd<CRLF> OK: <CRLF> <CRLF>	Reset complete

.IV INVENTORY COMMAND

Performs an inventory of transponders within the read range of the reader.
Returns an error if no transponders are found.

Parameters:

-al on/off — perform an alert after a successful inventory using the current alert parameters from the .al command
-c on/off — Include EPC checksum response
-dt on/off — Include date time response
-e on/off — Include EPC PC information response
-fi on/off — Impinj fast ID extension. If set, Transponders which support this feature will include their TID in the response
-io on/off — Inventory only, if set to on then no select will be performed before the inventory round is started
-ix on/off — Index number each transponder response
-n — No action, just set the parameters
-o nn — Output power in dBm, where nn is in the range 10 to 29
-p — List the parameter and their current values
-qa fix/dyn — Q algorithm
-ql all/nsl/sl — query Select
-qs s0,s1,s2,s3 — query Session
-qt a/b — query Target
-qv nn — Q value for fixed Q operations (0-15)
-r on/off — Include transponder RSSI response
-sa n — Select action

Parameter	Matching Action		Non Matching Action	
n=	SL Flag	Session Flag	SL Flag	Session Flag
0	assert	Set A	deassert	Set B
1	assert	Set A	nothing	nothing
2	nothing	nothing	deassert	Set B
3	toggle	toggle	nothing	nothing
4	deassert	Set B	assert	Set A
5	deassert	Set B	nothing	nothing
6	nothing	nothing	assert	Set A
7	nothing	nothing	toggle	toggle

-sb epc/tid/usr — Bank to use for the select mask
-sd xx..zz — Select mask data in 2 character ASCII Hex pairs padded to ensure full bytes
-sl xx — Length in bits of the select mask, where xx is a 2 character ASCII Hex value
-so xxxx — Number of bits from the start of the block to the start of the select mask, where xxxx is a 4 character ASCII Hex value
-st s0,s1,s2,s3,sl — Select target
-tf on/off — Impinj tag focus extension. Only effective if qs=s1, qt=a and if supported by the transponder.
-x — Reset the parameters to defaults

Parameter interpretation order

The parameters are interpreted and applied in the following order.

-x, -al, -c, -e, -r, -dt, -ix, -sb,-so, -sl, -sd, -o, -io, -sa, -st, -qa, -ql, -qs, -qt, -qv, -fi, -tf, -p, -n

Parameter defaults

The parameter defaults are:

-al on, -c off, -dt off, -e off, -fi off, -io on, -ix off, -o 29, -qa dyn, -ql all, -qs s1, -qt a, -qv 6,

-r off, -sa 0, -sb epc, -sd, -sl 00, -so 0000, -st s1, -tf off

Valid response field headers:

CR: CS: DT: EP: ER: IX: ME: OK: PC: PR: RI: TD:

Examples:

Command	Response	Action
.iv -p -n <CRLF>	CS: .iv -p -n<CRLF> PR: -al on -c off -dt off -e off -fi off -io on -ix off -n -o 29 -p -qa dyn -ql all -qs s1 -qt a -qv 6 -r off -sa 0 -sb epc -sd -sl 00 -so 0000 -st s1 -tf off -x<CRLF> OK:<CRLF><CRLF>	none
.iv <CRLF>	CS: .iv<CRLF> EP: 310833B2DDD906C000001234<CRLF> EP: 341486E37C00000000004255<CRLF> EP: 341486E37C00000000004254<CRLF> OK:<CRLF><CRLF>	3 transponders read
.iv -dt on -e on <CRLF>	CS: .iv -dt on -e on <CRLF> DT: 2000-01-26T19:00:52<CRLF> EP: 341486E37C00000000004254<CRLF> PC: 3000<CRLF> EP: 341486E37C00000000004255<CRLF> PC: 3000<CRLF> EP: 310833B2DDD906C000001234<CRLF> PC: 3000<CRLF> OK:<CRLF><CRLF>	3 transponders read

.KI KILL COMMAND

This command is used to kill transponders. The selected transponders with matching access and kill passwords will be permanently killed.

Parameters:

-ap xxxxxxxx Access password, where xxxxxxxx is an 8 character ASCII Hex value
-c on/off Include EPC checksum response
-dt on/off Include date time response
-e on/off Include EPC PC information response
-io on/off Inventory only, if set to on then no select will be performed before the inventory round phase is started
-ix on/off Index number each transponder response
-kp xxxxxxxx Kill password, where xxxxxxxx is an 8 character ASCII Hex value
-n No action, just set the parameters
-o nn Output power in dBm, where nn is in the range 10 to 29
-p List the parameter and their current values
-ql all/nsl/sl query Select
-qs s0,s1,s2,s3 query Session
-qt a/b query Target
-qv nn Q value (0-15)
-r on/off Include transponder RSSI response
-sa n Select action

Parameter n=	Matching Action		Non Matching Action	
	SL Flag	Session Flag	SL Flag	Session Flag
0	assert	Set A	deassert	Set B
1	assert	Set A	nothing	nothing
2	nothing	nothing	deassert	Set B
3	toggle	toggle	nothing	nothing
4	deassert	Set B	assert	Set A
5	deassert	Set B	nothing	nothing
6	nothing	nothing	assert	Set A
7	nothing	nothing	toggle	toggle

-sb epc/tid/usr Bank to use for the select mask
-sd xx..zz Select mask data in 2 character ASCII Hex bytes padded to ensure full bytes, up to a maximum of 32 bytes.
-sl xx Length in bits of the select mask, where xx is a 2 character ASCII Hex value
-so xxxx Number of bits from the start of the block to the start of the select mask, where xxxx is a 4 character ASCII Hex value
-st s0,s1,s2,s3,sl Select target
-x Reset the parameters to defaults

Parameter interpretation order

The parameters are interpreted and applied in the following order.

-x, -c, -e, -r, -dt, -ix, -sb,-so, -sl, -sd, -o, -io,-sa, -st, -ql, -qs, -qt, -qv, -ap, -kp, -p, -n

Parameter defaults

The parameter defaults are:

-ap 00000000, -c off, -dt off, -e off, -io off, -ix off, -kp 00000000, -o 29, -ql all, -qs s1, -qt b, -qv 2, -r off, -sa 4, -sb epc, -sd, -sl 00, -so 0000, -st s1

Valid response field headers:

CR: CS: DT: EA: EB: EP: ER: IX: KS: ME: OK: PC: PR: RI:

Examples:

Command	Response	Action
.ki -p -n <CRLF>	CS: .ki -p -n <CRLF> PR: -ap 00000000 -c off -dt off -e off -io off -ix off -kp 00000000 -n -o 29 -p -ql all -qs s1 -qt b -qv 2 -r off -sa 4 -sb epc -sd 3005FB63AC1F3681EC880468 -sl 60 -so 0020 -st s1 -x <CRLF> OK: <CRLF><CRLF>	No Action
.ki -ap 00000001 -kp 00000001<CRLF>	CS: .ki -ap 00000001 -kp 00000001 <CRLF> EP: 3005FB63AC1F3841EC880467 <CRLF> KS: Kill Success <CRLF> OK: <CRLF> <CRLF>	Transponder is killed

.LK LICENCE KEY

This command reads, writes and deletes the Licence Key stored in the reader's non-volatile memory. The Licence Key can be up to 127 characters long and can contain any printable characters with the exception of double quotes (").
How the licence key function is used is up to the programmer, it could be used to store a simple password or some form of hashing could be used with the two unique strings, the serial number and Bluetooth address, returned from the .vr command.

Parameters:

-d yes	Delete the Licence Key
-p	List the parameters
-s "Licence Key"	Set the licence key, up to 255 characters, enclosed in double quotes(").The licence key can contain any printable character except for double quotes ("). The Licence key can only be written if it is currently blank.

Parameter interpretation order

The parameters are interpreted and applied in the following order.

-d, -s, -p

Valid response field headers:

CS: ER: LK: ME: OK: PR:

Examples:

Command	Response	Action
.lk -p <CRLF>	CS: .lk -p<CRLF> PR: -d -p -s<CRLF> LK: ThisIsMyLicenceKey<CRLF> OK: <CRLF> <CRLF>	Licence key and parameters read
.lk -d yes -s "NewLicence" <CRLF>	CS:.lk -d yes -s "NewLicence" <CRLF> LK: NewLicence<CRLF> OK: <CRLF> <CRLF>	New licence key written

.LO LOCK COMMAND

This command is used to control access to the memory banks and passwords of the selected transponders. The lock is controlled by a 20 bit lock payload, please refer to the latest UHF Class 1 Gen 2 Standard for details. Care should be taken since some lock actions are permanent.

Parameters:

-ap xxxxxxxx	Access password, where xxxxxxxx is an 8 character ASCII Hex value
-c on/off	Include EPC checksum response
-dt on/off	Include date time response
-e on/off	Include EPC PC information response
-io on/off	Inventory only, if set to on then no select will be performed before the inventory round phase is started
-ix on/off	Index number each transponder response
-lp xxxxx	20 bit Lock command payload where xxxxx is a 5 character ASCII Hex value
-n	No action, just set the parameters
-o nn	Output power in dBm, where nn is in the range 10 to 29
-p	List the parameter and their current values
-ql all/nsl/sl	query Select
-qs s0,s1,s2,s3	query Session
-qt a/b	query Target
-qv nn	Q value (0-15)
-r on/off	Include transponder RSSI response
-sa n	Select action

Parameter n=	Matching Action		Non Matching Action	
	SL Flag	Session Flag	SL Flag	Session Flag
0	assert	Set A	deassert	Set B
1	assert	Set A	nothing	nothing
2	nothing	nothing	deassert	Set B
3	toggle	toggle	nothing	nothing
4	deassert	Set B	assert	Set A
5	deassert	Set B	nothing	nothing
6	nothing	nothing	assert	Set A
7	nothing	nothing	toggle	toggle

-sb epc/tid/usr	Bank to use for the select mask
-sd xx..zz	Select mask data in 2 character ASCII Hex bytes padded to ensure full bytes, up to a maximum of 32 bytes.
-sl xx	Length in bits of the select mask, where xx is a 2 character ASCII Hex value
-so xxxx	Number of bits from the start of the block to the start of the select mask, where xxxx is a 4 character ASCII Hex value
-st s0,s1,s2,s3,sl	Select target
-x	Reset the parameters to defaults

Parameter interpretation order

The parameters are interpreted and applied in the following order.

-x, -c, -e, -r, -dt, -ix, -sb,-so, -sl, -sd, -o, -io,-sa, -st, -ql, -qs, -qt, -qv, -lp, -ap, -p, -n

Parameter defaults

The parameter defaults are:

-ap 00000000, -c off, -dt off, -e off, -io off, -ix off, -lp 00000, -o 29, -ql all, -qs s1,

-qt b, -qv 2, -r off, -sa 4, -sb epc, -sd, -sl 00, -so 0000, -st s1

Valid response field headers:

CR: CS: DT: EA: EB: EP: ER: IX: LS: ME: OK: PC: PR: RI:

Examples:

Command	Response	Action
.lo -n -p<CRLF>	CS: .lo -n -p <CRLF> PR: -ap 00000000 -c off -dt off -e off -io off -ix off -lp 00000 -n -o 29 -p -ql all -qs s1 -qt b -qv 2 -r off -sa 4 -sb epc -sd -sl 00 -so 0000 -st s1 -x <CRLF> OK: <CRLF><CRLF>	No Action
.lo -lp 08020 -ap 00000001 <CRLF>	CS: .lo -lp 08020 -ap 00000001<CRLF> EP: 3005FB63AC1F3681EC880468 <CRLF> LS: Lock Success <CRLF> OK: <CRLF><CRLF>	EPC memory bank lock is set so it is writable only in the secured state

.PD PUSH SWITCH DOUBLE PRESS

Starts a timed software switch double press, this is equivalent to a double press and hold of the hardware switch. If the switch is already in use an error will be returned. If the duration is set to zero the switch press will continue until stopped with an abort command. If the switch action is set to barcode then the switch press will end on completion of the barcode read.

Parameters:

-n		No action, just set the parameters
-p		List the current parameter values
-tn	n=0..99	Press duration in seconds
-x		Reset the parameters to defaults

Parameter interpretation order

The parameters are interpreted and applied in the following order.

-x, -t,-p, -n

Parameter defaults

The parameter defaults are:

-t 5

Valid response field headers:

CS: ER: ME: OK: PR:

Examples:

Command	Response	Action
.pd -p -n <CRLF>	CS: .pd -p -n<CRLF> PR: -n -p -t 5 -x<CRLF> OK: <CRLF> <CRLF>	No Action
.pd <CRLF>	CS: .pd<CRLF> ME: Switch already in use<CRLF> ER:014<CRLF><CRLF>	The switch press was refused because the switch was already in use

.PS PUSH SWITCH SINGLE PRESS

Starts a timed software switch single press, this is equivalent to a single press and hold of the hardware switch. If the switch is already in use an error will be returned. If the duration is set to zero the switch press will continue until stopped with an abort command. If the switch action is set to barcode then the switch press will end on completion of the barcode read.

Parameters:

-n		No action, just set the parameters
-p		List the current parameter values
-tn	n=0..99	Press duration in seconds
-x		Reset the parameters to defaults

Parameter interpretation order

The parameters are interpreted and applied in the following order.

-x, -t,-p, -n

Parameter defaults

The parameter defaults are:

-t 5

Valid response field headers:

CS: ER: ME: OK: PR:

Examples:

Command	Response	Action
.ps -p -n <CRLF>	CS: .ps -p -n<CRLF> PR: -n -p -t 5 -x<CRLF> OK: <CRLF> <CRLF>	No Action
.ps <CRLF>	CS: .ps<CRLF> ME: Switch already in use<CRLF> ER:014<CRLF><CRLF>	The switch press was refused because the switch was already in use

.RA READ THE AUTORUN FILE

Reads the Autorun file, additionally the Autorun file can be deleted. This command cannot be used in the

Autorun file.

Parameters:

-d yes	delete the Autorun file

Parameter interpretation order

The parameters are interpreted and applied in the following order.

-d

Parameter defaults

No parameter defaults are required:

Valid response field headers:

CS: ER: AB: AE: ME: OK:

Potentially any other response field could be contained in the log file.

Examples:

Command	Response	Action
.ra <CRLF>	CS: .ra<CRLF> AB: *Autorun* file data AE: OK: <CRLF> <CRLF>	The autorun file is read
.ra -d yes <CRLF>	CS: .ra -d yes <CRLF> OK: <CRLF> <CRLF>	The autorun file is deleted

.RD READ TRANSPONDER COMMAND

Performs an inventory of transponders within the read range of the reader and then reads the data from the requested bank. Returns an error if no transponders are found.

Parameters:

-al on/off	perform an alert after a successful read using the current alert parameters from the .al command
-ap xxxxxxxx	Access password, where xxxxxxxx is an 8 character ASCII Hex value
-c on/off	Include EPC checksum response
-db epc/tid/usr/res	Data bank to read from
-dl xx	Data length, size of the data to read (number of 16 bit words in the range 1 to 253 decimal i.e. 01 to FD hex), where xx is a 2 character ASCII Hex value
-do xxxx	Data offset, the number of 16 bit words, from the start of the block, to start read from, where xxxx is a 4 character ASCII Hex value
-dt on/off	Include date time response
-e on/off	Include EPC PC information response
-io on/off	Inventory only, if set to on then no select will be performed before the inventory round phase is started
-ix on/off	Index number each transponder response
-n	No action, just set the parameters
-o nn	Output power in dBm, where nn is in the range 10 to 29
-p	List the parameter and their current values
-ql all/nsl/sl	query Select
-qs s0,s1,s2,s3	query Session
-qt a/b	query Target
-qv nn	Q value (0-15)
-r on/off	Include transponder RSSI response
-sa n	Select action

Parameter n=	Matching Action		Non Matching Action	
	SL Flag	Session Flag	SL Flag	Session Flag
0	assert	Set A	deassert	Set B
1	assert	Set A	nothing	nothing
2	nothing	nothing	deassert	Set B
3	toggle	toggle	nothing	nothing
4	deassert	Set B	assert	Set A
5	deassert	Set B	nothing	nothing
6	nothing	nothing	assert	Set A
7	nothing	nothing	toggle	toggle

-sb epc/tid/usr	Bank to use for the select mask
-sd xx..zz	Select mask data in 2 character ASCII Hex bytes padded to ensure full bytes, up to a maximum of 32 bytes.
-sl xx	Length in bits of the select mask, where xx is a 2 character ASCII Hex value
-so xxxx	Number of bits from the start of the block to the start of the select mask, where xxxx is a 4 character ASCII Hex value
-st s0,s1,s2,s3,sl	Select target
-x	Reset the parameters to defaults

Parameter interpretation order

The parameters are interpreted and applied in the following order.

-x, -al, -c, -e, -r, -dt, -ix, -sb,-so, -sl, -sd, -o, -io,-sa, -st, -ql, -qs, -qt, -qv, -db, -do,
-ap, -dl, -p, -n

Parameter defaults

The parameter defaults are:

-al on, -ap 00000000, -c off, -db epc, -dl 02, -do 0000, -dt off, -e off, -io off, -ix off
-o 29, -ql all, -qs s2, -qt b, -qv 6, -r off, -sa 4, -sb epc, -sd, -sl 00, -so 0000, -st s2

Valid response field headers:

CR: CS: DT: EA: EB: EP: ER: IX: ME: OK: PC: PR: RD: RI:

Examples:

Command	Response	Action
.rd -p -n <CRLF>	CS: .rd -p -n<CRLF> PR: -al on -ap 00000000 -c off -db epc -dl 02 -do 0000 -dt off -e off -io off -ix off -n -o 29 -p -ql all -qs s2 -qt b -qv 6 -r off -sa 4 -sb epc -sd -sl 00 -so 0000 -st s2 -x <CRLF> OK:<CRLF><CRLF>	none
.rd <CRLF>	CS: .rd<CRLF> EP: 310833B2DDD906C000001234<CRLF> RD: C3993000<CRLF> EP: 341486E37C00000000004254<CRLF> RD: 70CF3000<CRLF> OK:<CRLF><CRLF>	2 transponders read
.rd -dt on -e on <CRLF>	CS: .rd -dt on -e on <CRLF> DT: 2000-01-26T19:00:52<CRLF> EP: 341486E37C00000000004254<CRLF> PC: 3000<CRLF> RD: 70CF3000<CRLF> EP: 310833B2DDD906C000001234<CRLF> PC: 3000<CRLF> RD: C3993000<CRLF> OK: <CRLF><CRLF>	2 transponders read

.RL READ THE LOG FILE

Reads log file, additionally the log file can be deleted or logging can be turned on or off. If the delete option is used the log file is erased and then a new log file is created. This command cannot be used in the Autorun file.

Parameters:

-c on/off	turn command logging on or off
-d yes	delete the log file
-n	No action, just set the parameters
-p	List the parameter and their current values
-x	Reset the parameters to defaults

Parameter interpretation order

The parameters are interpreted and applied in the following order.

-x, -d,-c, -p, -n

Parameter defaults

-c on

Valid response field headers:

CS: ER: LB: LE: ME: OK: PR:

Potentially any other response field could be contained in the log file.

Examples:

Command	Response	Action
.rl <CRLF>	CS: .rl<CRLF> LB: *Log* file data LE: OK: <CRLF> <CRLF>	The log file is read
.rl -d yes <CRLF>	CS: .rl -d yes<CRLF> LB: **** Beginning of Log ****<CRLF> DT: 2000-01-14T20:17:24<CRLF> EV: Log File created<CRLF> LE: **** End of Log ****<CRLF> OK: <CRLF> <CRLF>	The log file is deleted

.SA SWITCH ACTION COMMAND

Defines what happens when the switch is pressed.

Parameters:

-a on/off	Turn asynchronous switch status reporting on or off
-d off/rd/wr/inv/bar/usr	Set the double press switch action
-p	List the current parameter values
-rd xxx	Sets the delay before the switch double press action is repeated when the switch is held pressed. Where xxx is a value between 1 and 999 ms
-rs xxx	Sets the delay before the switch single press action is repeated when the switch is held pressed. Where xxx is a value between 1 and 999 ms
-s off/rd/wr/inv/bar/usr	Set the single press switch action
-x	Reset the parameters to defaults

Parameter interpretation order

The parameters are interpreted and applied in the following order.

-x, -a, -d, -s, -p

Parameter defaults

The parameter defaults are:

-a off, -d bar, -s inv

Valid response field headers:

CS: ER: ME: OK: PR:

Examples:

Command	Response	Action
.sa -x <CRLF>	CS: .sa -x<CRLF> OK: <CRLF> <CRLF>	.sa defaults loaded
.sa -p <CRLF>	CS: .sa -p<CRLF> PR: -a off -d bar -p –rd 100 –rs 100 -s inv -x <CRLF> OK:<CRLF><CRLF>	None

.SL SLEEP

This command puts the unit into sleep mode. This will disconnect any active *Bluetooth*® connections.

Parameters:

No parameters are used for this command.

Valid response field headers:

CS: ER: ME: OK:

Examples:

Command	Response	Action
.sl <CRLF>	CS: .sl<CRLF> OK:<CRLF><CRLF>	Unit goes to sleep

.SP USER DEFINED SWITCH SINGLE PRESS ACTION

Reads or sets the single press user action used when “.sa -s usr” is configured.

Parameters:

-s user action command	Set the user action

Parameter interpretation order

The parameters are interpreted and applied in the following order.

-s

Parameter defaults

No parameter defaults are required:

Valid response field headers:

CS: ER: ME: OK: SP:

Examples:

Command	Response	Action
.sp <CRLF>	CS: .sp<CRLF> SP: .iv<CRLF> OK: <CRLF> <CRLF	User action is read
.sp -s .bc -t 4<CRLF>	CS: .sp -s .bc -t 4<CRLF> SP: .bc -t 4<CRLF> OK: <CRLF> <CRLF>	User action is set

.SS SWITCH STATE COMMAND

Reads the state of the switch

Parameters:

No parameters are used for this command.

Valid response field headers:

CS: SW: ER: ME: OK:

Examples:

Command	Response	Action
.ss <CRLF>	CS: .ss<CRLF> SW: single <CRLF> OK: <CRLF> <CRLF>	The state of the switch is read and it was on by a single click
.ss <CRLF>	CS: .ss<CRLF> SW: off <CRLF> OK:<CRLF><CRLF>	The state of the switch is read and it was off

.ST SLEEP TIMEOUT

Sets the timeout before the reader sleeps if there are no connections to the reader.

Parameters:

-p		List the current parameter values
-tn	n=15..999	Sleep duration in seconds
-x		Reset the parameters to defaults

Parameter interpretation order

The parameters are interpreted and applied in the following order.

-x, -t,-p

Parameter defaults

The parameter defaults are:

-t 60

Valid response field headers:

CS: ER: ME: OK: PR:

Examples:

Command	Response	Action
.st -p <CRLF>	CS: .st -p<CRLF> PR: -p -t60 -x<CRLF> OK: <CRLF> <CRLF>	No Action
.st -t 15<CRLF>	CS: .st -t 15<CRLF> OK: <CRLF> <CRLF>	The sleep timeout is set to 15 Seconds

.TM TIME COMMAND

Reads or sets the time.

Parameters:

-s hhmmss	Set the time

Parameter interpretation order

The parameters are interpreted and applied in the following order.

-s

Parameter defaults

No parameter defaults are required:

Valid response field headers:

CS: ER: ME: OK: TM:

Examples:

Command	Response	Action
.tm <CRLF>	CS: .tm<CRLF> TM: 18:24:02<CRLF> OK: <CRLF> <CRLF>	The time is read
.tm -s 203100<CRLF>	CS: .tm -s 203100<CRLF> TM: 20:31:00<CRLF> OK: <CRLF> <CRLF>	The time is set

.TS TRANSPONDER SELECT

This command is used to set the state of transponders. It pushes matching and non- matching transponders in to the state determined by the -sa parameter. This command can only be used with persistent target sessions as the carrier will be turned off after the command.

Parameters:

-n	No action, just set the parameters
-o nn	Output power in dBm, where nn is in the range 10 to 29
-p	List the parameter and their current values
-sa n	Select action

Parameter n=	Matching Action		Non Matching Action	
	SL Flag	Session Flag	SL Flag	Session Flag
0	assert	Set A	deassert	Set B
1	assert	Set A	nothing	nothing
2	nothing	nothing	deassert	Set B
3	toggle	toggle	nothing	nothing
4	deassert	Set B	assert	Set A
5	deassert	Set B	nothing	nothing
6	nothing	nothing	assert	Set A
7	nothing	nothing	toggle	toggle

-sb epc/tid/usr	Bank to use for the select mask
-sd xx..zz	Select mask data in 2 character ASCII Hex bytes padded to ensure full bytes, up to a maximum of 32 bytes.
-sl xx	Length in bits of the select mask, where xx is a 2 character ASCII Hex value
-so xxxx	Number of bits from the start of the block to the start of the select mask, where xxxx is a 4 character ASCII Hex value
-st s1,s2,s3,sl	Select target
-x	Reset the parameters to defaults

Parameter interpretation order

The parameters are interpreted and applied in the following order:

-x, -sb,-so, -sl, -sd, -o, -sa, -st, -p, -n

Parameter defaults

The parameter defaults are:

-o 29, -sa 4, -sb epc, -sd, -sl 00, -so 0000, -st s1

Valid response field headers:

CS: ER: ME: OK: PR:

Examples:

Command	Response	Action
.ts -p -n <CRLF>	CS: .ts -p -n<CRLF> PR: -n -o 29 -p -sa 4 -sb epc -sd -sl 00 -so 0000 -st s1 -x <CRLF> OK:<CRLF><CRLF>	none

.VR VERSION INFORMATION COMMAND

Reads the version information from the reader

Parameters:

No parameters are used for this command.

Valid response field headers:

CS: AS: BA: MF: PV: RB: RF: RS: UB: UF: US: ER: ME: OK:

Examples:

Command	Response	Action
.vr <CRLF>	CS: .vr<CRLF> MF: Technology Solutions UK ltd. <CRLF> US: 1128-EU-000001<CRLF> PV: V2.0.0<CRLF> UF: V3.0.0<CRLF> UB: V1.3.0<CRLF> RS: 1116-ET-000001<CRLF> RF: V2.6.0<CRLF> RB: V1.2.0<CRLF> AS: 1128-S1-EU-000001<CRLF> BA: 00:07:80:41:a4:c6 <CRLF> OK:<CRLF><CRLF>	The version information was read

.WA WRITE COMMAND TO AUTORUN FILE

Writes the following command to the end of the Autorun file. This command cannot be used in the Autorun file.

Parameters:

No parameters are used for this command.

Valid response field headers:

CS: ER: ME: OK:

Examples:

Command	Response	Action
.wa .al -voff -n<CRLF>	CS: .wa .al -voff - n<CRLF> OK: <CRLF> <CRLF>	The ".al -voff -n" command is appended to the the end of the Autorun file, this command is turning off the vibrate at wake up.

.WR WRITE TRANSPONDER COMMAND

This command tries to write data to any transponder that meets the select criteria. It will return an error if no transponder is found.
For each transponder the number of words written follows the WW: response header.

Parameters:

-al on/off Perform an alert after a successful write using the current alert Parameters from the .al command

-ap xxxxxxxx Access password, where xxxxxxxx is an 8 character ASCII Hex value

-c on/off Include EPC checksum response

-da xxxx..zzzz The data to write in 4 character ASCII Hex words. Up to a maximum of 32 words

-db epc/tid/usr/res Data bank to write to

-dl xx Length in words of the data to write, where xx is a 2 character ASCII Hex value up to a maximum of 20hex

-do xxxx Data bank offset, this is the offset, in 16 bit words, from the start of the block to where the data will be written, where xxxx is a 4 character ASCII Hex value

-dt on/off Include date time response

-e on/off Include EPC PC information response

-io on/off Inventory only, if set to on then no select will be performed before the inventory round phase is started

-ix on/off Index number each transponder response

-n No action, just set the parameters

-o nn Output power in dBm, where nn is in the range 10 to 29

-p List the parameter and their current values

-ql all/nsl/sl query Select

-qs s0,s1,s2,s3 query Session

-qt a/b query Target

-qv nn Q value (0-15)

-r on/off Include transponder RSSI response

-sa n Select action

Parameter n=	Matching Action		Non Matching Action	
	SL Flag	Session Flag	SL Flag	Session Flag
0	assert	Set A	deassert	Set B
1	assert	Set A	nothing	nothing
2	nothing	nothing	deassert	Set B
3	toggle	toggle	nothing	nothing
4	deassert	Set B	assert	Set A
5	deassert	Set B	nothing	nothing
6	nothing	nothing	assert	Set A
7	nothing	nothing	toggle	toggle

-sb epc/tid/usr Bank to use for the select mask

-sd xx..zz Select mask data in 2 character ASCII Hex bytes padded to ensure full bytes, up to a maximum of 32 bytes.

-sl xx Length in bits of the select mask, where xx is a 2 character ASCII Hex value

-so xxxx Number of bits from the start of the block to the start of the select mask, where xxxx is a 4 character ASCII Hex value

-st s0,s1,s2,s3,sl	Select target
-wm s/b	Set the Write Mode to either single or block. Block mode is not supported by all transponders
-wx a/1/2	Impinj Block Write Mode extension. Auto, Force 1 or Force 2 block write. Unless the transponder population is known, use Auto. This option is only valid if -wm=b
-x	Reset the parameters to defaults

Parameter interpretation order

The parameters are interpreted and applied in the following order.

-x, -al, -c, -e, -r, -dt, -ix, -sb,-so, -sl, -sd, -o, -io, -sa, -st, -ql, -qs, -qt, -qv, -dl, -da, -db,

-do, -ap, -wm, -wx, -p, -n

Parameter defaults

The parameter defaults are:

-al on, -ap 00000000, -c off, -da, -db usr, -dl 00, -do 0000, -dt off, -e off, -io off, -ix off, -o 29,

-ql all, -qs s2, -qt b, -qv 6, -r off, -sa 4, -sb epc, -sd, -sl 00, -so 0000, -st s2, -wm s, -wx a

Valid response field headers:

CR: CS: DT: EA: EB: EP: ER: IX: ME: OK: PC: PR: RI: WW:

Examples:

Command	Response	Action
.wr -p -n <CRLF>	CS: .wr-p-n<CRLF> PR: -al on -ap 00000000 -c off -da -db usr -dl 00 -do 0000 -dt off -e off -io off -ix off -n -o 29 -p -ql all -qs s2 -qt b -qv 4 -r off -sa 4 -sb epc -sd -sl 00 -so 0000 -st s2 -wm s -wx a -x<CRLF> OK: <CRLF><CRLF>	none
.wr -da1234 -dl01<CRLF>	CS: .wr -da1234 -dl01<CRLF> EP: 310833B2DDD906C000001234<CRLF> WW: 1<CRLF> EP: 310833B2DDD906C000001235<CRLF> WW: 0<CRLF> EP: 310833B2DDD906C000001236<CRLF> WW: 1<CRLF> OK:<CRLF><CRLF>	Write 0x1234 to a transponders

.WS WRITE SINGLE TRANSPONDER COMMAND

This command writes data to a single transponder. It uses a single time slot so there must be enough mask information to select a single transponder.
Returns an error if no transponder is found or if the write is incomplete.

Parameters:

-al on/off	Perform an alert after a successful write using the current alert Parameters from the .al command
-ap xxxxxxxx	Access password, where xxxxxxxx is an 8 character ASCII Hex value
-c on/off	Include EPC checksum response
-da xxxx..zzzz	The data to write in 4 character ASCII Hex words. Up to a maximum of 32 words
-db epc/tid/usr /res	Data bank to write to
-dl xx	Length in words of the data to write, where xx is a 2 character ASCII Hex value up to a maximum of 20hex
-do xxxx	Data bank offset, this is the offset, in 16 bit words, from the start of the block to where the data will be written, where xxxx is a 4 character ASCII Hex value
-dt on/off	Include date time response
-e on/off	Include EPC PC information response
-n	No action, just set the parameters
-o nn	Output power in dBm, where nn is in the range 10 to 29
-p	List the parameter and their current values
-r on/off	Include transponder RSSI response
-sb epc/tid/usr	Bank to use for the select mask
-sd xx..zz	Select mask data in 2 character ASCII Hex bytes padded to ensure full bytes, up to a maximum of 32 bytes.
-sl xx	Length in bits of the select mask, where xx is a 2 character ASCII Hex value
-so xxxx	Number of bits from the start of the block to the start of the select mask, where xxxx is a 4 character ASCII Hex value
-x	Reset the parameters to defaults

Parameter interpretation order

The parameters are interpreted and applied in the following order.

-x, -al, -c, -e, -r, -dt, -sb,-so, -sl, -sd, -o, -dl, -da, -db, -do, -ap, -p, -n

Parameter defaults

The parameter defaults are:

-al on, -ap 00000000, -c off, -da, -db usr, -dl 00, -do 0000, -dt off, -e off, -o 29,

-r off, -sb epc, -sd, -sl 00, -so 0000

Valid response field headers:

CR: CS: DT: EP: ER: ME: OK: PC: PR: RI: WW:

Examples:

Command	Response	Action
.ws -p -n <CRLF>	CS: .ws-p-n<CRLF> PR: -al on -ap 00000000 -c off -da -db usr -dl 00 -do 0000 -dt off -e off -n -o 29 -p -r off -sb epc -sd -sl 00 -so 0000 -x<CRLF> OK: <CRLF><CRLF>	none
.ws -da1234 -dl01<CRLF>	CS: .ws -da1234 -dl01<CRLF> EP: 310833B2DDD906C000001234<CRLF> WW: 1<CRLF> OK:<CRLF><CRLF>	Write 0x1234 to a transponder

HARDWARE SPECIFIC CONNECTION NOTES

For specific information about how to communicate with a particular Technology Solutions UHF reader using the ASCII Protocol please refer to the User Guide for that product. User guides are available for download from the Product Downloads section of each product page (free registration required)

http://www.tsl.com/products/1119-uhf-rfid-reader-for-the-motorola-mc55-65/

http://www.tsl.com/products/1126-desktop-uhf-rfid-reader-with-usb/

http://www.tsl.com/products/1128-bluetooth-handheld-uhf-rfid-reader/

The core of the ASCII protocol is based on reading and writing serial data over a serial connection. Typically this connection is 115200 baud, 8 data bits, 1 stop bit and no parity with CTS/RTS (hardware) handshaking. Although some products use a standard serial port those that communicate over USB and *Bluetooth*® use a virtual serial port.

When communicating via *Bluetooth*® you can either use a *Bluetooth*® virtual serial port and have your application open a com port using serial APIs or alternatively you can usually open a *Bluetooth*® stream using the Serial Port Profile service identifier without having to use a serial port layer. Refer to the samples provided with the SDKs or contact support@tsl.com for more information.

ABOUT

ABOUT TSL®

TECHNOLOGY SOLUTIONS UK LTD
part of HID

Technology Solutions UK Ltd (TSL®), part of HID Global, is a leading manufacturer of high performance mobile RFID readers used to identify and track products, assets, data or personnel.
For over two decades, TSL® has delivered innovative data capture solutions to Fortune 500 companies around the world using a global network of distributors and system integrators. Specialist in-house teams design all aspects of the finished products and software ecosystems, including electronics, firmware, application development tools, RF design and injection mould tooling.
TSL® is an ISO 9001:2015 certified company.

ISO 9001 Registered Firm | International Accreditation Board
Certificate No. GB2001267 | Registration No. 0044/1

ISO 9001: 2015

CONTACT

Address:	Technology Solutions (UK) Ltd, Suite A, Loughborough Technology Centre, Epinal Way, Loughborough, Leicestershire, LE11 3GE, United Kingdom.
Telephone:	+44 1509 238248
Fax:	+44 1509 214144
Email:	enquiries@tsl.com
Website:	www.tsl.com

ABOUT HID GLOBAL

HID Global powers the trusted identities of the world's people, places and things. We make it possible for people to transact safely, work productively and travel freely. Our trusted identity solutions give **people** convenient access to physical and digital **places** and connect **things** that can be identified, verified and tracked digitally. Millions of people around the world use HID products and services to navigate their everyday lives, and billions of things are connected through HID technology. We work with governments, educational institutions, hospitals, financial institutions, industrial businesses and some of the most innovative companies on the planet. Headquartered in Austin, Texas, HID Global has over 4,000 employees worldwide and operates international offices that support more than 100 countries. HID Global is an ASSA ABLOY Group brand.
For more information, visit www.hidglobal.com.

Technology Solutions (UK) Ltd reserves the right to change its products, specifications and services at any time without notice.