

Data Parsing on Zebra Scanners

Overview

This document explains how to program and demonstrate data parsing on Zebra handheld scanners. Zebra's Data Parsing technology supports the following industry solutions.

1. UDI Parsing (UDI Scan+: GS1, HIBBC and ICCBBA)
2. Label Parsing (Label Parse+ for GS1 labels)
3. Blood Bag Parsing ISBT 128 (Blood Bag Parse+)

This document can also be left with a customer to kick start their compliance testing.

Note. This document does not cover Zebra's USA Driver's License Parsing solution.

What is Data Parsing

Data Parsing allows a Zebra scanner to scan a UDI label or GS1 label or Blood Bags with one or more barcodes encoded with multiple data fields (date of manufacture, expiration date, batch number, GTIN, SSCC ...) and transmit select data fields and not others, in a specific order to a host application. Simply wave the scanner over all the barcodes while holding the trigger and the scanner will take care of the rest.

The scanner will find and transmit only the required data fields, even if they are spread across multiple barcodes and on different sides of the container. In addition, the scanner can insert field separators (tab, enter, slash ...) to automate data entry into a host application.

Programming your scanner is easy using 123Scan's intuitive drag and drop interface.

To watch a video on Creating a Data Parsing Rule using 123Scan, go to:

<http://www.zebra.com/ScannerHowToVideos>

Scan a UDI Label using UDI Scan+

Government regulatory agencies* have established Unique Device Identification (UDI) standards to identify and monitor the distribution and use of medical devices within healthcare environments. These UDI standards identify medical devices from manufacturing through distribution to patient use - enabling complete traceability of the millions of individual medical devices utilized for patient care. To enable UDI compliance, all medical devices must carry a UDI label to enable "track and trace" from the point of production, during shipment, through the product's use and disposal.

* United States Food and Drug Administration (FDA), European Commission, International Medical Device Regulatory Forum

Scan a GS1 Label using Label Parse+

The GS1 Organization, an international standards body, has released specifications used worldwide for generating shipping labels. These labels are used when shipping packages (logistics), raw materials and produce.

For additional information see https://www.gs1.org/docs/tl/GS1_Logistic_Label_Guideline.pdf

Scan a Blood Bag Label using Blood Bag Parse+

The ICCBBA Organization, an international standards body, has released a specification used worldwide for generating blood bag labels. These labels are used when shipping, storing and using blood bags.

For additional information see <https://www.iccbba.org/tech-library/iccbba-documents/standards-documents/standard-labeling-blood2>

Required Equipment

The following is required to demonstrate data parsing on a Zebra Scanner.

1. 123Scan, scanner configuration utility. See Appendix B for details.
2. A Zebra scanner like the DS8108 or DS8178.
3. This document, “Data Parsing User Guide.doc”.

How to Use This Document

1. Watch the Data Parsing how to video
 - a. www.zebra.com/ScannerHowToVideos
 - b. Video Name: Zebra 123Scan: Creating a Data Parsing Rule (UDI, GS1 Label, Blood Bag)
Its grouped with the other 123Scan videos.
2. Update your DS8108/DS8178 scanner to the latest firmware. See Appendix B for details.
3. Program a data parsing rule. See Example 1 through 16 for step by step instructions in Appendix A.
4. Close 123Scan
5. Launch Microsoft’s Notepad. Place the mouse’s cursor within the Notepad window.
6. Scan the UDI/GS1/Blood Bag barcode from the examples below.
 - a. UDI Parsing : Example 1 – 10
 - b. GS1 Label Parsing : Example 11 – 13
 - c. ADF Modify Data Parsed Output : Example 14
 - d. Blood Bag Parsing : Example 15 – 17

Overview

1. What is Zebra's Data Parsing Technology
2. What is a GS1 Label
3. What is Unique Device Identification (UDI) Label
4. What is a Blood Bag (ISBT 128) Label
5. What Zebra Scanners Support Data Parsing technology

General

6. Where can I get a complete list of GS1 data fields
7. Can an untrained user scan data parsing capable barcodes
8. What is the difference between MDF and data parsing
9. Can I scan all the data fields within multiple barcodes in one trigger pull
10. What if the required data fields are spread across both 1D and 2D barcodes
11. What does a GS1 barcode that supports data parsing look like
12. How do I scan into different screens in a host application
13. What is output if I scan a data parsing enabled barcode that lacks ALL the required data fields
14. Does data parsing support both handheld (triggered) and handsfree (presentation) scanning
15. How many data parsing rules can be programmed on a scanner
16. How many barcodes can be scanned from a UDI Label / GS1 Label / Blood Bag Label
17. How many data fields can be programmed within a single data parsing rule
18. How many separator fields can be inserted between two data fields
19. Can data parsing be enable / disabled by my application
20. Can an AIM Code ID be used when Data Parsing
21. Can a Function Key(s) Be Inserted Using ADF into Data Parsed Output
22. What data fields can have their date format modified by the scanner
23. What If I scan a new, previously undocumented or incorrectly formatted data field

123Scan

24. How is a scanner programmed to perform data parsing
25. Can the order of output data fields be programmed in 123Scan
26. How long do I have to find all the barcodes (data fields) on the label
27. Can my data parsing programmed scanner scan regular barcodes
28. What if one required data field on a label is not present
29. Retrieving a data parsing rule from a scanner
30. Can a data parsing rule be programmed through Remote Management

UDI

31. What UDI standards are supported by Zebra Scanners
32. What are the most common UDI data fields tracked by hospitals
33. What GS1 / HIBCC / ICCBBA UDI output options are available within 123Scan
34. What HIBCC Data Field Identifiers can be scanned
35. What scanner parameters must be enabled to scan UDI properly

Blood Bag

36. What ISBT 128 Blood Bag fields are tracked by hospitals
37. What is the "Blood Bag Parsing Session Timeout"
38. Settings for scanning optional barcodes, that are not always present on the label
39. What scanner parameters must be enabled to scan blood bags

Trouble Shooting

40. The Data Matrix (GS1) output is not correct
41. The QR Code (GS1) output is not correct
42. Blood Bag scanning and ICCBBA UDI Scanning are not working correctly

43. In HID Keyboard, the Tab and / or Enter key is not recognized by your App

44. What options are available under “Other Settings”

1. Overview - What is Zebra’s Data Parsing Technology

Data Parsing allows a Zebra scanner to scan a UDI label or GS1 label or Blood Bags with one or more barcodes encoded with multiple data fields (date of manufacture, expiration date, batch number, GTIN, SSCC ...) and transmit select data fields and not others, in a specific order to a host application. Simply wave the scanner over all the barcodes while holding the trigger and the scanner will take care of the rest.

The scanner will find and transmit only the required data fields, even if they are spread across multiple barcodes and on different sides of the container. In addition, the scanner can insert field separators (tab, enter, slash ...) to automate data entry into a host application.

Programming your scanner is easy using 123Scan’s intuitive drag and drop interface.

To watch a video on Creating a Data Parsing Rule using 123Scan, go to:

<http://www.zebra.com/ScannerHowToVideos>

2. Overview - What is a GS1 Label

The GS1 Organization, an international standards body, has released specifications used worldwide for generating shipping labels. These labels are used when shipping packages (logistics), raw materials and produce.

GS1 Label Specification: https://www.gs1.org/docs/tl/GS1_Logistic_Label_Guideline.pdf

For a complete list of GS1 data fields (application identifiers – AI) click

<https://www.gs1.org/standards/barcodes/application-identifiers?lang=en>

3. Overview - What is Unique Device Identification (UDI) Label

Government regulatory agencies* have established Unique Device Identification (UDI) standards to identify and monitor the distribution and use of medical devices within healthcare environments. These UDI standards identify medical devices from manufacturing through distribution to patient use - enabling complete traceability of the millions of individual medical devices utilized for patient care. To enable UDI compliance, all medical devices must carry a UDI label to enable “track and trace” from the point of production, during shipment, through the product’s use and disposal.

* United States Food and Drug Administration (FDA), European Commission, International Medical Device Regulatory Forum

4. Overview - What is a Blood Bag (ISBT 128) Label

The ICCBBA Organization, an international standards body, has released a specification used worldwide for generating blood bag labels. These labels are used when shipping, storing and using blood bags.

For additional information see <https://www.iccbba.org/tech-library/iccbba-documents/standards-documents/standard-labeling-blood2>

5. Overview - What Zebra Scanners Support Data Parsing technology

Select 2D imaging scanners support Zebra’s Data Parsing technology.

- | | | |
|------------------------------|----------|---------------------------------------|
| • DS8108 All Configurations* | 10.21.20 | firmware update required – Healthcare |
| • DS8178 All Configurations* | 10.21.20 | firmware update required – Healthcare |
| • CS6080 All Configurations* | 10.01.20 | Ships functional from factory |

• DS9908 All Configurations*	Q4 '20	firmware update required – Healthcare
• DS9308 All Configurations*	Q4 '20	firmware update required – Healthcare
• DS4608 All Configurations*	Q4 '20	firmware update required – Healthcare
• DS3608 All Configurations*	Q4 '20/Q1 '21	firmware update required – Mfg. / T&L
• DS3678 All Configurations*	Q4 '20/Q1 '21	firmware update required – Mfg. / T&L

* To confirm presence in the firmware, see 123Scan release notes. Data Parsing capability will be called out as “Added 1) UDI Parsing / Scan+; 2) Label Parsing / Parse+ for GS1; 3) Blood Bag Parsing / Parse+”

6. General - Where can I get a complete list of GS1 data fields

For a complete list of GS1 data fields (application identifiers – AI), click this link from the GS1 Organization’s website: <https://www.gs1.org/standards/barcodes/application-identifiers?lang=en>

7. General - Can an untrained user scan data parsing capable barcodes

Yes, just point the scanner at the desired UDI, GS1 Labeling barcode(s) and pull the trigger. The scanner will automatically decode all the barcodes within the field of view in one trigger pull. Alternatively, if the barcodes are on opposite sides of a box, the user can keep the trigger depressed, while aiming at one barcode then the next to capture all the barcodes. The scanner will accumulate the data and output it.

8. General - What is the difference between MDF and data parsing

MDF allows a scanner to scan multiple barcodes in one trigger pull and each barcode can only contain one data field. For example, 3 barcodes could contain 3 data fields (3 pieces of information). The barcodes must also be unique in some way, different symbology, different length ...

Data parsing allows a scanner to scan one to many barcodes in one trigger pull, but each barcode can contain more than one data field. For example, 3 barcodes could contain 10 data fields (10 pieces of information). Unlike MDF, the barcodes are not required to be unique, they can be the same symbology and the same length.

9. General - Can I scan all the data fields within multiple barcodes in one trigger pull

Yes. In an industry first, Zebra’s Data Parsing solution solves this problem. Simply wave the scanner over all the barcodes while holding the trigger and the scanner will take care of the rest.

The scanner will find and transmit only the required data fields, even if they are spread across multiple barcodes and on different sides of the container. In addition, the scanner can insert field separators (tab, enter, slash ...) to automate data entry into a host application.

Programming your scanner is easy using 123Scan’s intuitive drag and drop interface.

10. General - What if the required data fields are spread across both 1D and 2D barcodes

Zebra’s Data Parsing solution is focused on data fields. If part of the required data fields are in a 1D barcode but the remaining data fields are in a 2D barcode, the scanner will capture and output all the required data fields even though they are spread across multiple barcode and symbologies.

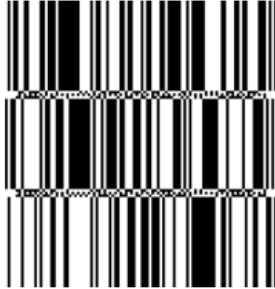
11. General - What does a GS1 barcode that supports data parsing look like

The following GS1 Parsing supported barcode formats can be scanned using select Zebra 2D imaging scanners.

- i. 1D - GS1-128 (Single barcode) 
(01)10705031003576(17)230206(10)J1802717

- ii. 1D - GS1-128 (Multiple barcodes) 
(01)10705031003576
(17)230206(10)J1802717

- iii. 1D - GS1 DataBar 
(01)10705031003576

- iv. 1D - GS1 DataBar Stacked Barcode 
(01) 10705031003576
(17) 230206
(10) J1802717

- v. 2D - GS1 Datamatrix 
(01) 10705031003576
(17) 230206
(10) J1802717
(21) 2541584697

- vi. 2D - GS1 QR Code 
(01) 10705031003576
(17) 230206
(10) J1802717
(21) 2541584697

12. General - How do I scan into different screens in a host application

Depending on the host app's requirement per screen (data fields and separator fields), one of the following will apply.

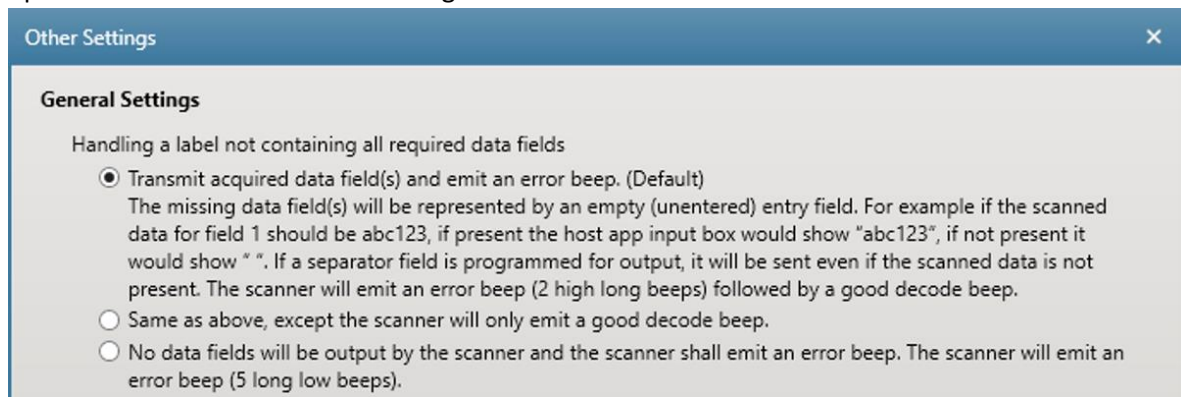
Option A – If the same data fields and separator are required for both screens, the same UDI parsing rule will work on both screens.

Option B – If different data fields and / or separator are required per screen, a different UDI parsing rules will be required per screen. The recommended scanner programming methodology is scanning a 2D programming barcode generated from 123Scan. Each screen will require a unique 2D programming barcode. A possible approach would be to tape rule 1 and rule 2 near the monitor for scanning upon switching screens.

13. General - What is output if I scan a data parsing enabled barcode that lacks ALL the required data fields

If no required data fields are present, the scanner will not output any data fields.

If one or more data fields is missing but the rest present, the output will vary with the selected output option chosen from the Other Settings menu.

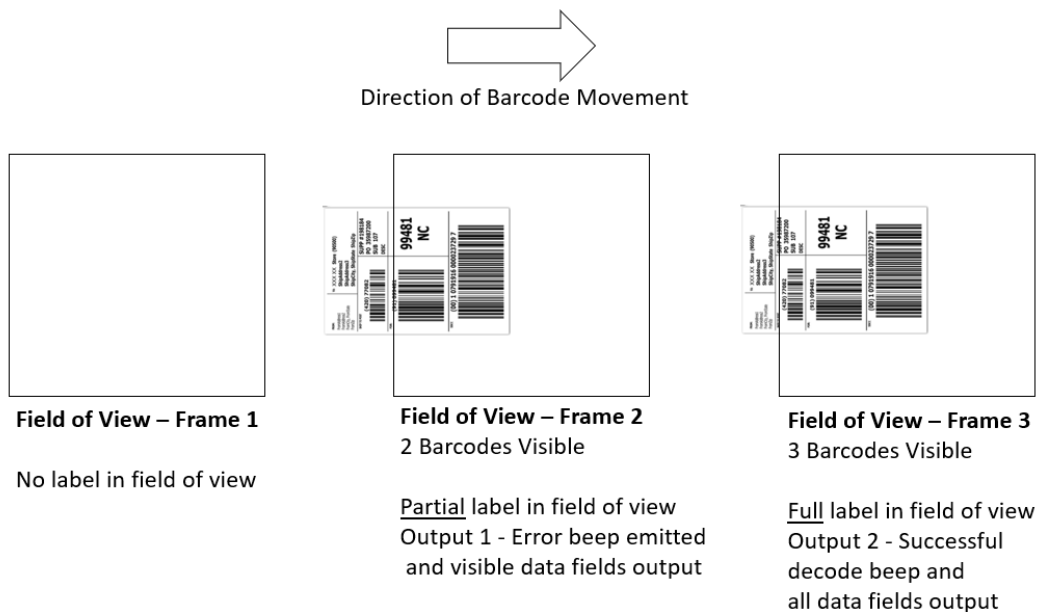


14. General – Does data parsing support both handheld (triggered) and handsfree (presentation) scanning

Zebra Data Parsing capable scanners support both handheld (triggered) scanning and handsfree (presentation) scanning.

Though data parsing capable scanners support handsfree (presentation) scanning, its implementation can be tricky. Depending on the scanning condition, a handsfree scanning session may yield unexpected, undesired multiple outputs as a label passes through the field of view. The results may be anywhere between a complete set of data fields (successfully decode beep) to a partial set of data fields (Error beep) or both.

This problem is more likely to occur in presentation mode on a slow moving, complex label. As the label is moving through the field of view, the label is first partially read (some of the data fields are in the field of view – See frame 2 below) then a second decode can occur as it is fully read (all the data fields are in the field of view – See frame 3 below).

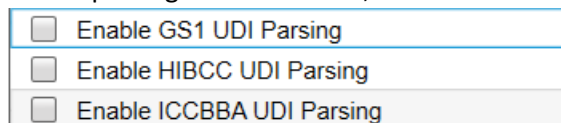


15. General - How many data parsing rules can be programmed on a scanner

Only one data parsing rule can be stored in a scanner at a time; Either a UDI parsing rule* or GS1 label parsing rule or Blood bag parsing rule. In the example below from 123Scan, a Blood bag rule is selected.



* Note if a UDI parsing rule is selected, it can contain all three UDI formats: GS1, HIBCC and ICCBBA.



16. General - How many barcodes can be scanned from a UDI Label / GS1 Label / Blood Bag Label

This question does not make sense in the data parsing world because a single barcode can contain numerous data fields. Traditionally, in a non-data parsing scenario, a barcode equals a single data field, but not here. In the example below, a single data parsing barcode contains 3 data fields, which traditionally in a non-data parsing scenario would require 3 barcodes.



With this explained, a Zebra scanner can scan 10 or more barcodes in one trigger pull when performing data parsing. The barcodes can be a combination of 1D and 2D symbologies.

Generally, **UDI labels** include up to three 1D stacked barcodes and possibly one additional 2D barcode. Generally, **GS1 labels** include up to three 1D stacked barcodes and possibly one additional 2D barcode.

Generally, **Blood Bag** labels include up to seven 1D barcodes and possibly one additional 2D barcode.

17. General - How many data fields can be programmed within a single data parsing rule

A data parsing rule can be written with up to 15 or more data fields, until the data parsing memory is completely used on the scanner. Memory usage is shown by clicking on the "Memory used" link on

 [Other Settings](#) |  [Memory used](#)  [Back](#)

Parsing Session Timeout (sec) [What is this?](#)

123Scan's data parsing screen. Parsing Prioritization Time (ms) [What is this?](#)

18. General - How many separator fields can be inserted between two data fields

Up to ten separator fields can be inserted between two data fields.

Note only one pause can be inserted between data fields. To allow for shorter or longer pauses, the pause duration is programmable. At the end of the rule, after the last data field, a pause can only be inserted as the last field separator.

19. General - Can data parsing be enable / disabled by my application

Yes. An application can programmatically enable and disable data parsing using a Zebra driver and a Remote Scanner Management (RSM) command.

- | | |
|-------------------------------|---|
| i. UDI - GS1 Parsing | - RSM attribute number 881. 1 = Enable; -1 = Disable. |
| ii. UDI - HIBCC Parsing | - RSM attribute number 882. 1 = Enable, -1 = Disable. |
| iii. UDI - ICCBBA Parsing | - RSM attribute number 883. 1 = Enable, -1 = Disable. |
| iv. GS1 Label Parsing | - RSM attribute number 881. 1 = Enable; -1 = Disable. |
| v. Blood Bag ISBT 128 Parsing | - RSM attribute number 884. 1 = Enable, -1 = Disable. |

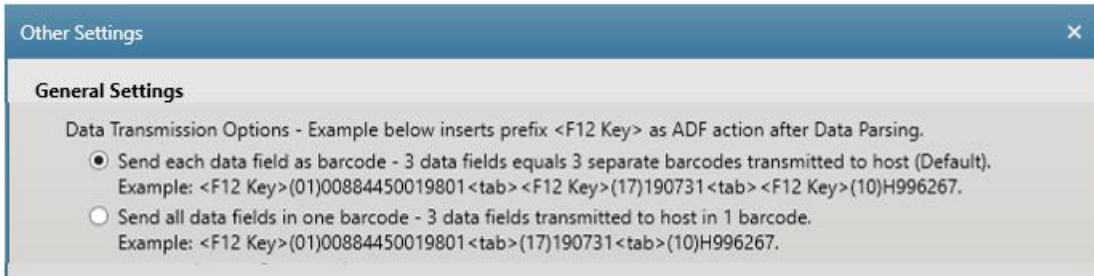
20. General - Can an AIM Code ID be used when Data Parsing

Yes. If the AIM Code Identifier has been enabled (Transmit Code ID Character = AIM Code ID) within 123Scan. The following two AIM Code ID output options exist.

- If you select "Send each data field as a barcode (default)" the scanner will output the AIM ID before each data field like below.
 -]x0(01)00884450019801 tab]x0(17)190731 tab]x0(10)H996267
- If you select "Send all data fields in one barcode" the scanner will output AIM ID once before all data fields like below.
 -]x0(01)00884450019801 tab (17)190731 tab (10)H996267

To enable, find the "Data Transmission Options" under General Settings by clicking 123Scan's

 [Other Settings](#) :



Note using “Send each data field as a barcode”, pause must be inserted through ADF.
 Note a generic UID AIM ID will be output, which will show up as an 'unknown' AIM code type

21. General - Can a Function Key(s) Be Inserted Using ADF into Data Parsed Output

Yes. See Appendix A / Barcode Example 13.

22. General - What data fields can have their date format modified by the scanner

UDI GS1 – Date fields can have their output format modified; the default is YYMMDD.

UDI HIBCC - Common Fields – Date fields can have their output format modified; the default is YYMMDD. An option also exists to output the date fields in the same format as present in the barcode.

UDI HIBCC - All Fields in Numeric Order – Output in native HIBCC format.

UDI ICCBBA - Common Fields – Date fields can have their output format modified; the default is YYMMDD. An option also exists to output the date fields in the same format as present in the barcode.

UDI ICCBBA - All Fields in Numeric Order – Date fields can have their output format modified; the default is YYMMDD. An option also exists to output the date fields in the same format as present in the barcode.

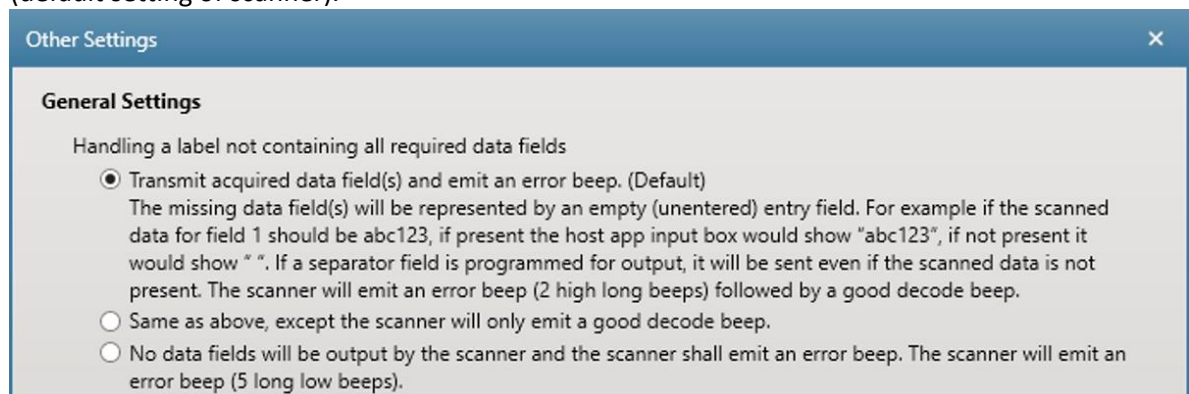
GS1 Label Parsing – Date fields can have their output format modified; the default is YYMMDD.

Blood Bag – Date fields are output in the same format as present in the barcode, YYYYDD.

23. General - What if I scan a new, previously undocumented or incorrectly formatted data field

If a new data field or incorrectly formatted data field appears in a label that is not recognized by the scanner’s firmware, it will behave in one of the following ways:

- If the scanner 1st captures a good data field(s) before hitting an undocumented and or incorrectly formatted field – The scanner emit beeps as selected in Other Settings / “Handling a label not containing all required data fields” and transmits the good data fields to the host (default setting of scanner).



- If the undocumented field and or incorrectly formatted field is the first data field in the barcode – The scanner emits a convert error beep (5 long error beeps) and no data will be transmitted to the host.
- If the scanner captures all the required data fields, then hits an undocumented and or incorrectly formatted data field – The scanner emits a good decode beep and transmits the good, required data fields to the host.

24. 123Scan - How is a scanner programmed to perform data parsing

123Scan, Zebra’s scanner programming utility, is used to program data parsing enabled scanner. The parsing functionality is found in 123Scan’s configuration wizard by clicking the Modify Data tab, then clicking Program Complex Data Modifications / Advanced / UDI Parsing or GS1 Label Parsing.

25. 123Scan - Can the order of output data fields be programmed in 123Scan

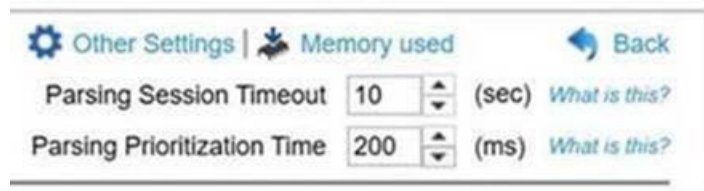
Yes. 123Scan can be used to write a parsing rule which modifies the order of output data fields and transmit some or all of the data fields to a host application. In addition, to enable the data fields to be entered into the host app’s appropriate entry fields, the UDI parsing rule can insert a separator like a tab between the data fields. 123Scan’s separator options include comma, carriage return, line feed, enter, space, tab, pause, in addition to others.

The UDI parsing functionality is found in 123Scan’s configuration wizard by clicking the Modify Data tab, then clicking Program Complex Data Modifications / Advanced / UDI Parsing or GS1 Label Parsing.

26. 123Scan - How long do I have to find all the barcodes (data fields) on the label

The “Parsing Session Timeout” is the maximum time (while depressing the trigger) to acquire all required data fields. The scanning session will be exited after this time expires, similar to releasing the trigger. The longer this setting, the more time you have to scan multiple barcodes (while keeping the trigger depressed).

The Parsing Session Timeout is customer programmable within 123Scan’s data parsing screen. Default value = 10 seconds.



27. 123Scan - Can my data parsing programmed scanner scan regular barcodes

Yes, a scanner programmed with a data parsing rule can be used to scan regular barcodes.

If a scanner has a data parsing rule present, 123Scan can be used to fine tune how long a trigger must be depressed (called the “Parsing Prioritization Time”) before outputting a non-data parsing barcode.

Barcodes lacking the required data fields, will be ignored for a programmed amount of time called the “Parsing Prioritization Time”. During this time, only barcode(s) with the specified data parsing fields can be output.

After the “Parsing Prioritization Time” has passed and then a non-data parsing (regular) barcode is scanned, it will be output.

If only a regular barcode was scanned before the “Parsing Prioritization Time” has passed, that regular barcode will be output after the “Parsing Prioritization Time” has expired.

The screenshot shows the 'Other Settings' tab in the 123Scan configuration wizard. It features two settings: 'Parsing Session Timeout' set to 10 seconds and 'Parsing Prioritization Time' set to 200 milliseconds. Each setting has a 'What is this?' link. Navigation links for 'Other Settings', 'Memory used', and 'Back' are at the top.

The “Parsing Prioritization Time” is found in 123Scan’s configuration wizard by clicking the Modify Data tab, then clicking Program Complex Data Modifications / Advanced / UDI Parsing or GS1 Label Parsing.

28. 123Scan - What if one required data field on a label is not present

If a required data field is missing, not found when scanning, 123Scan gives the following programming options under the Other Settings menu.

The screenshot shows the 'Other Settings' dialog box with the 'General Settings' tab selected. Under the heading 'Handling a label not containing all required data fields', there are three radio button options:

- ☒ Transmit acquired data field(s) and emit an error beep. (Default)

The missing data field(s) will be represented by an empty (unentered) entry field. For example if the scanned data for field 1 should be abc123, if present the host app input box would show "abc123", if not present it would show " ". If a separator field is programmed for output, it will be sent even if the scanned data is not present. The scanner will emit an error beep (2 high long beeps) followed by a good decode beep.
- ☐ Same as above, except the scanner will only emit a good decode beep.
- ☐ No data fields will be output by the scanner and the scanner shall emit an error beep. The scanner will emit an error beep (5 long low beeps).

29. 123Scan - Retrieving a data parsing rule from a scanner

123Scan is not able to retrieve a data parsing rule from a scanner, for editing within 123Scan.

But the data parsing rule is likely saved in the 123Scan configuration file used originally to program the scanner.

Note, though 123Scan cannot edit a data parsing rule retrieved from a scanner, it can save the data parsing rule into a configuration file for cloning and remote management.

30. 123Scan - Can a data parsing rule be programmed through Remote Management

123Scan can be used to generation a SMS Package containing a data parsing rules for remote management. Zebra’s SMS can be used to push this package to a data parsing enabled Zebra scanner.

31. UDI - What UDI standards are supported by Zebra Scanners

All three UDI parsing standards as set forth by the United States FDA (Food and Drug Administration) are supported on select Zebra scanners.

- i. GS1 Parsing
- ii. HIBCC Parsing
- iii. ICCBBA Parsing


32. UDI - What are the most common UDI data fields tracked by hospitals

Hospitals most often track GS1 UDI barcodes, in particular these three GS1 UDI fields.

- GS1 data field 01 – Device Identifier (GTIN - Global Trade Item Number)
- GS1 data field 10 – Batch / Lot Number
- GS1 data field 17 – Use by expiration date (YYMMDD)

For a complete list of GS1 data fields (application identifiers – AI) click
<https://www.gs1.org/standards/barcodes/application-identifiers?lang=en>

33. UDI - What GS1 / HIBCC / ICCBBA UDI output options are available within 123Scan

123Scan gives the following options under the  **Other Settings** menu.

Other Settings

General Settings

Handling a label not containing all required data fields

- ☒ Transmit acquired data field(s) and emit an error beep. (Default)
The missing data field(s) will be represented by an empty (unentered) entry field. For example if the scanned data for field 1 should be abc123, if present the host app input box would show "abc123", if not present it would show " ". If a separator field is programmed for output, it will be sent even if the scanned data is not present. The scanner will emit an error beep (2 high long beeps) followed by a good decode beep.
- ☐ Same as above, except the scanner will only emit a good decode beep.
- ☐ No data fields will be output by the scanner and the scanner shall emit an error beep. The scanner will emit an error beep (5 long low beeps).

Data Transmission Options - Example below inserts prefix <F12 Key> as ADF action after Data Parsing.

- ☒ Send each data field as barcode - 3 data fields equals 3 separate barcodes transmitted to host (Default).
Example: <F12 Key>(01)00884450019801<tab><F12 Key>(17)190731<tab><F12 Key>(10)H996267.
- ☐ Send all data fields in one barcode - 3 data fields transmitted to host in 1 barcode.
Example: <F12 Key>(01)00884450019801<tab>(17)190731<tab>(10)H996267.

GS1 Settings

☐ Include AI field number with output barcode data. [What is this?](#)

☐ Include parenthesis around AI field.

Handling Decimal point in barcode data

- ☒ Decimal is present. Sample output is 1234.56 (Default).
- ☐ Comma is present. Sample output is 1234,56.
- ☐ Nothing present. Sample output is 123456.

Handling date fields with the DD (day) set as "00" within the barcode

- ☒ Show date field for DD (day) as "00".
- ☐ Show date field for DD (day) as last day of the month, for example, November would be "30" and December would be "31".

ICCBBA (UDI & ISBT 128 Blood Bag) Settings

☐ UDI - Include data identifier(s) with output barcode data. For data field 07 (Collection Date and Time), the data identifier &* would be append in front of the barcode data.

☐ Blood Bag - Include data identifier(s) with output barcode data. For data field 07 (Collection Date and Time), the data identifier &* would be append in front of the barcode data.

HIBCC Settings

☐ Include data identifier(s) with output barcode data. For data field 05 (Lot Alphanumeric), the data identifier \$ would be append in front of the barcode data.

OK
CANCEL

34. UDI - What HIBCC Data Field Identifier can be scanned

HIBCC has a complicated data field identifier structure, the symbol(s) that identify a data field for example a single lone + indicates the Device Identifier. In some cases, multiple data field identifiers exist for the same piece of information. For example an expiration date can be identified with any of these thirteen sequences: \$\$, \$\$2, \$\$3, \$\$4,\$\$5, \$\$6, \$\$+, \$\$+2, \$\$+3, \$\$+4,\$\$ +5, \$\$+6 and 14D. All thirteen of these identifier sequences contain the expiration date but each is encoded in a slightly different format for example YYMMDD, YYMMDDHH, MMDDYY, MMY...

The scanner can decipher all HIBCC data field identifier structures and output it in the desired format, as programmed through 123Scan.

35. UDI - What scanner parameters must be enabled to scan UDI properly

Both GS1 and ICCBBA UDI scanning requires configuration of your scanner symbologies, in addition to writing a UDI data parsing rule. For details, see the trouble shooting section of this document.

Note, HIBCC UDI scanning does not require any special configuration of your scanner's symbologies.

36. Blood Bag - What ISBT 128 Blood Bag fields are tracked by hospitals

The first four blood bag fields, listed below, are commonly used by hospitals. All the fields below are accessible as part of the Zebra Blood Bag Parsing solution.

1. DIN (Donation Identification Number)	Top left barcode on label
2. Blood type	Top right barcode on label
3. Product Code	Bottom left barcode on label
4. Expiration Date and Time (Expiration Info.)	Bottom right barcode on label
5. Special Testing	Option barcodes, not always on the label
6. Collection Date and Time	Option barcodes, not always on the label

37. Blood Bag – What is the “Blood Bag Parsing Session Timeout”

The Blood Bag Parsing Session Timeout is the maximum time (while depressing the trigger) to scan all required blood bag barcodes. The scanning session will stop after this time period and the scanned barcodes transmitted to the host, even if the trigger is still depressed. The longer this setting, the more time you have to scan barcodes.

[Other Settings](#) | [Memory used](#) [Back](#)

Blood Bag Parsing Session Timeout (sec) [What is this?](#)

Parsing Prioritization Time (ms) [What is this?](#)

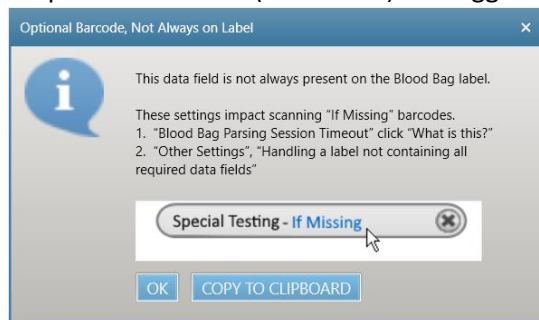
Setting accessible in 123Scan.

Recommended Session Timeout

Most users = 10 seconds

When scanning optional barcodes* = 0.5 seconds

* Optional barcodes (data fields) are tagged with a “If Missing” label within 123Scan

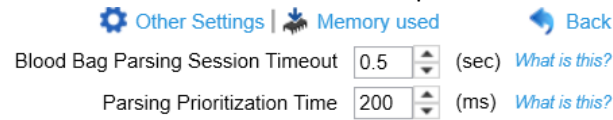


38. Blood Bag – Setting for scanning optional barcodes, that are not always present on the label

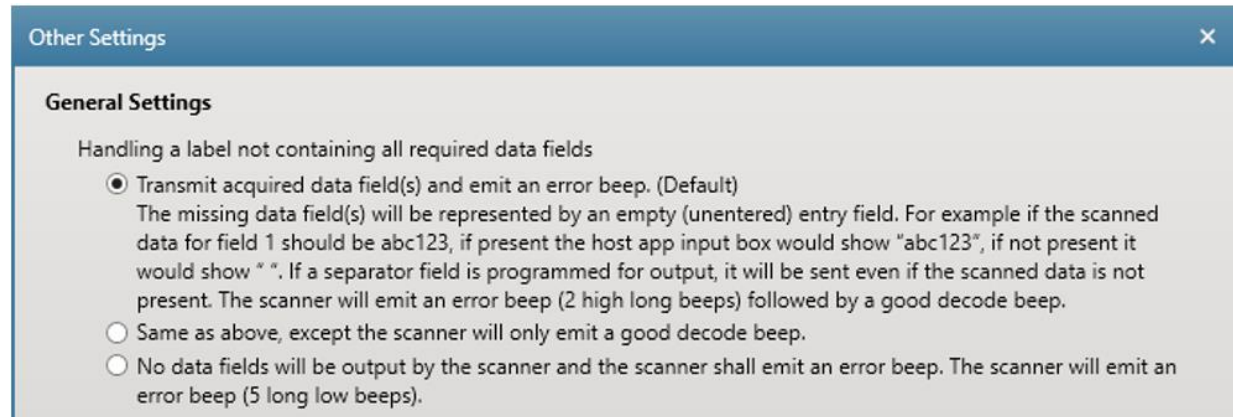
The “Special Testing” and “Collection Date and Time” barcodes are not required on a blood bag label but can be include as an optional barcodes* (present on the label if the printer of the label chose to include

it). If you are including one/both of these optional barcodes in your scanner's output, we recommend making these settings changes within 123Scan's Blood Bag Parsing screen.

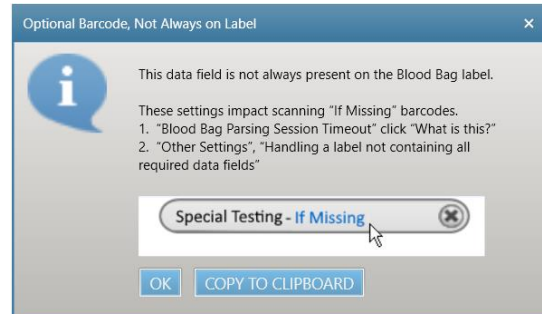
1. "Blood Bag Parsing Session Timeout" - Change the recommended session timeout from 10 seconds to 0.5 seconds like in this picture.**



2. "Handling a label not containing all required data fields"*** – Decide what data and decode sound(s) get output if an optional barcode is not present on the label. Select from the options below.



* Optional barcodes (data fields) are tagged with a "If Missing" label within 123Scan



** Changing the "Blood Bag Parsing Session Timeout" to 0.5 seconds allows the scanner to output data even if an optional barcode is not present (on the label) in under a second, even with the trigger depressed. If the timeout was left at the default 10 seconds, when scanning a label missing the optional barcode and the scanner was programmed to look for this optional barcode, no data would be output until the 10 seconds expired or the trigger was released – Not very intuitive scanning behavior.

*** For more details see the 123Scan FAQ labeled "What if one required data field on a label is not present"

39. Blood Bag - What scanner parameters must be enabled to scan blood bags

ISBT 128 blood bag scanning requires configuration of your scanner symbologies, in addition to writing a blood bag data parsing rule. For details, see the trouble shooting section of this document.

40. Trouble Shooting - The Data Matrix (GS1) output is not correct



2D - GS1 Datamatrix

(01) 10705031003576
(17) 230206
(10) J1802717
(21) 2541584697

To decode and parsing these GS1 Datamatrix data fields, you must enable “GS1-Datamatrix” on the scanner (123Scan / Config file / Symbologies / Other 2D Symbologies / GS1-Datamatrix). This Symbology is generally disabled by default.

If you do not enable the GS1-Datamatrix symbology on the scanner, only standard Data Matrix will be enabled. The barcode above, even though it is a GS1 Datamatrix, will be treated and output as a standard Datamatrix (without any data parsing applied). In this case, all the data present in the barcode will be output including AI numbers as 01107050310035761723020610J1802717212541584697.

41. Trouble Shooting - The QR Code (GS1) output is not correct



2D - GS1 QR Code

(01) 10705031003576
(17) 230206
(10) J1802717
(21) 2541584697

To decode and parsing these GS1 QR Code data fields, you must enable “GS1-QR Code” on the scanner (123Scan / Config file / Symbologies / Other 2D Symbologies / GS1-QR Code). This Symbology is generally disabled by default.

If you do not enable the GS1-QR Code symbology on the scanner, only standard QR Code will be enabled. The barcode above, even though it is a GS1-QR Code, will be treated and output as a standard QR Code (without any data parsing applied). In this case, all the data present in the barcode will be output including AI numbers as 01107050310035761723020610J1802717212541584697.

42. Trouble Shooting – Blood Bag scanning and ICCBBA UDI Scanning are not working correctly

To decode and parse ICCBBA based Blood Bags (ISBT 128) and UDI barcodes, make sure your scanner has the following settings

- ISBT 128 is enabled
- Code 128 ISBT Concatenation is set to “Disabled ISBT Concatenation”
- In some cases depending on your label, Datamatrix may need to be enabled.

Change your bar code settings

Your scanner is already set up to recognize the most common types of bar codes. To modify your **decoding options**, select a symbology (bar code type) from the drop-down below.

Code 128 ^

Select Code Types

☒ Code 128

☒ ISBT 128

☒ GS1-128 (UCC/EAN-128)

☐ Options for Code 128

☐ Options for ISBT ^

ISBT
Concatenation
Redundancy

10

ISBT
Concatenation ^

Disable ISBT Concatenation

Check ISBT Table

☒

43. Trouble Shooting – In HID Keyboard, the Tab and / or Enter key is not recognized by your App

To make Tab work properly in your application, you must “Enable Function Key Mapping”. After this is enabled, the scanner will send a different TAB key, for more information see the article below or your scanner’s Product Reference Guide.

To enable “Function Key Mapping” (while in USB HID keyboard communication mode), either scan the barcode below or modify your 123Scan config file. To modify your 123Scan config file go to 123Scan’s Config Wizard / Cable Connection / HID Keyboard Options / Place an x in the Function Key Mapping


check box ☒ Function Key Mapping ^.



Enable Function Key Mapping

For more information, see this article: <https://supportcommunity.zebra.com/s/article/Enter-or-Tab-keys-in-my-scanned-data-are-not-recognized-properly-within-my-application>

44. Trouble Shooting – What options are available under “Other Settings”

123Scan gives the following options under the  Other Settings menu.

Other Settings

General Settings

Handling a label not containing all required data fields

- ☒ Transmit acquired data field(s) and emit an error beep. (Default)
The missing data field(s) will be represented by an empty (unentered) entry field. For example if the scanned data for field 1 should be abc123, if present the host app input box would show "abc123", if not present it would show " ". If a separator field is programmed for output, it will be sent even if the scanned data is not present. The scanner will emit an error beep (2 high long beeps) followed by a good decode beep.
- ☐ Same as above, except the scanner will only emit a good decode beep.
- ☐ No data fields will be output by the scanner and the scanner shall emit an error beep. The scanner will emit an error beep (5 long low beeps).

Data Transmission Options - Example below inserts prefix <F12 Key> as ADF action after Data Parsing.

- ☒ Send each data field as barcode - 3 data fields equals 3 separate barcodes transmitted to host (Default).
Example: <F12 Key>(01)00884450019801<tab><F12 Key>(17)190731<tab><F12 Key>(10)H996267.
- ☐ Send all data fields in one barcode - 3 data fields transmitted to host in 1 barcode.
Example: <F12 Key>(01)00884450019801<tab>(17)190731<tab>(10)H996267.

GS1 Settings

☐ Include AI field number with output barcode data.
[What is this?](#)

☐ Include parenthesis around AI field.

Handling Decimal point in barcode data

- ☒ Decimal is present. Sample output is 1234.56 (Default).
- ☐ Comma is present. Sample output is 1234,56.
- ☐ Nothing present. Sample output is 123456.

Handling date fields with the DD (day) set as "00" within the barcode

- ☒ Show date field for DD (day) as "00".
- ☐ Show date field for DD (day) as last day of the month, for example, November would be "30" and December would be "31".

ICCBBA (UDI & ISBT 128 Blood Bag) Settings

☐ UDI - Include data identifier(s) with output barcode data. For data field 07 (Collection Date and Time), the data identifier &* would be append in front of the barcode data.

☐ Blood Bag - Include data identifier(s) with output barcode data. For data field 07 (Collection Date and Time), the data identifier &* would be append in front of the barcode data.

HIBCC Settings

☐ Include data identifier(s) with output barcode data. For data field 05 (Lot Alphanumeric), the data identifier \$ would be append in front of the barcode data.

OK
CANCEL

Appendix A: Examples

- UDI Parsing : Example 1 – 10
- GS1 Label Parsing : Example 11 – 13
- ADF Modify Data Parsed Output : Example 14
- Blood Bag Parsing : Example 15 – 17

Barcode Example 1 – UDI Label Parsing (GS1)

Symbology: GS1-128 Barcode

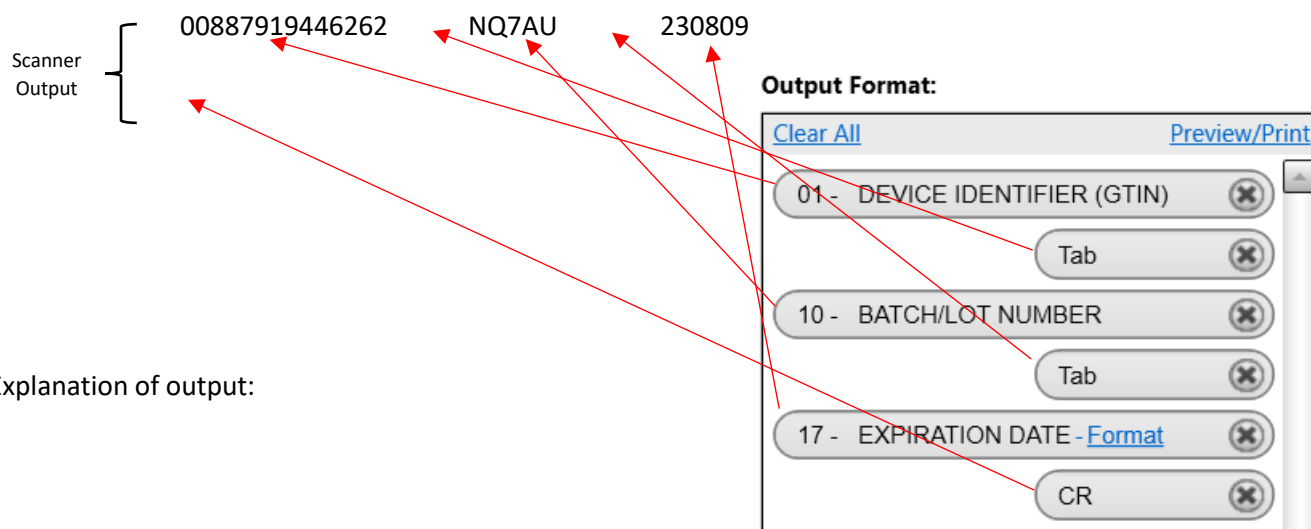
What is Demonstrated:

Scanning a single 1D GS1 UDI barcode.

Changing the order of the data fields 01, 17, 10 within the barcode to 01,10, 17 output from the scanner.



Output



Explanation of output:

The UDI Parsing rule from 123Scan (above) outputs the data in the manner illustrated (above).

The scanner collected all the UDI data fields present in the barcode.

After finding the requested fields (01, 10 and 17), the scanner then outputs only those data fields, per the 123Scan UDI Parsing rule, in the desired order (01, 10 then 17).

This output order is opposite of the way the data is encoded in the barcode (Order of data fields within barcode: 01, 17 then 10).

After outputting the three data fields, the scanner outputs a carriage return (CR), moving the cursor to the next line.

Barcode Example 2 – UDI Label Parsing (GS1)

What is Demonstrated:

One trigger pull is all it takes.

The scanner will collect all the UDI data fields present across both barcodes and output them to the host application.

Symbology: GS1-128 Stacked 1D Barcodes



Output

Scanner Output { 10705036012399 T92X03 240331

Explanation of output:

The scanner collected all the UDI data fields present across both barcodes.

After finding the requested fields (01, 17 and 10), the scanner then outputs only those data fields, per the 123Scan UDI Parsing rule, in the desired order (01, 10 then 17).

Hint

Make sure your scanner's field of view (illuminated scanning area) covers both barcodes.

Barcode Example 3 – UDI Label Parsing (GS1)

What is Demonstrated:



Barcode 1

We understand nurses are busy and can't focus on scanning. Just pull the trigger and Zebra's scanner will take care of the rest.

The scanner will give you time (customer programmable) to find one barcode or multiple barcodes.

The scanner will find and transmit only the required data fields, even if they are spread across multiple barcodes.

The scanner can insert field separators (tab, enter ...) to automate data entry into a host application.

How to demonstrate

- A) Pull and keep holding the trigger while scanning **barcode 1**.
- B) While still keeping the trigger depressed, count to 5 (this simulates the noise and bustle of a busy environment).
- C) While still keeping the trigger depressed, change your aim to the **barcode 2**, at the bottom of the page.

Symbology: GS1-128 Multiple 1D Barcodes (*Same barcode as used in Example 2*)

Output

Scanner Output	{	10705036012399	T92X03	240331
-------------------	---	----------------	--------	--------

Explanation of output:

Even though the barcodes are spread out, Zebra scanning solution keeps track of the scanned data fields. Barcode 1 has one of the required data fields (01). Barcode 2 has the other two required data fields (10, 17). Once both barcodes have been scanned, within the same scanning session (one trigger pull), the scanner outputs the required data fields.

The Parsing Session Timeout gives extra time to scan barcodes by just keeping the trigger depressed while aiming at all the barcodes.

The Parsing Session Timeout is the maximum time (while depressing the trigger) to acquire all required data fields. The scanning session will be exited after this time expires. The longer this setting, the more time you have to scan multiple barcodes. The Parsing Session Timeout is customer programmable within 123Scan's UDI Parsing screen. Default value = 10 seconds.



Barcode 2

Barcode Example 4 – UDI Label Parsing (GS1)

What is Demonstrated:

The nurse can point the scanner at either set of barcodes, the scanner will output the correct results.

First, scan the single 2D Datamatrix.

Second, scan the two 1D barcodes

Symbology: Both GS1-128 Multiple 1D Barcodes and a GS1-Datamatrix



Output

Scanner Output	{	20886705001795	L862995	230430
-------------------	---	----------------	---------	--------

Explanation of output:

Zebra's UDI solution allows nurses to scan UDI barcodes worry free, just pull the trigger and wave the scanner over the barcodes. The scanner will take care of the rest.

Barcode Example 5 – UDI Label Parsing (GS1)

What is Demonstrated:

Ability to decode GS1 QR Code

Symbology: GS1-QR Code



(01) 10705031003576

(17) 230206

(10) J1802717

(21) 2541584697

Output

Scanner
Output { 10705031003576 J1802717 230206

Explanation of output:

The scanner collected all the UDI data fields present across both barcodes.

After finding the requested fields (01, 17 and 10), the scanner then outputs only those data fields, per the 123Scan UDI Parsing rule, in the desired order (01, 10 then 17).

Notice data field 21, which is not required by the hospital's host application, is parsed out of (removed from) the data transmitted to the host.

Barcode Example 6 – UDI Label Parsing (GS1)

What is Demonstrated:

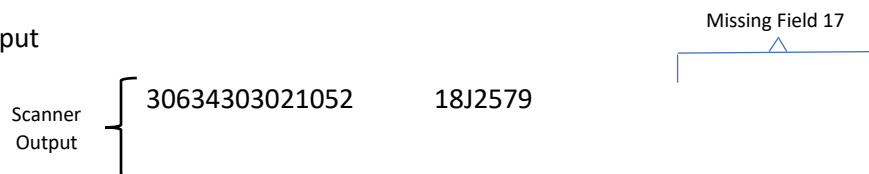
If a UDI barcode is missing a required field(s), per the 123Scan UDI Parsing rule, the scanner will do the following.

1. The scanner collected all the UDI data fields present in the barcode
2. If a required UDI data field(s) is missing (not encoded within the barcode) the following occurs.
 - a. Upon release of the trigger,
 - i. The scanner will emit an error beep
 - ii. The scanner will transmit the data fields present in the barcode to the host
 - iii. The missing data field will be left empty, allowing the nurse to enter the data if it exist

Symbology: GS1-128 Multiple 1D Barcodes



Output



Explanation of output:

The scanner collected all the UDI data fields present across both barcodes, in this case data fields 01 and 10.

Note the 123Scan UDI Parsing rule specifies data fields 01, 10 then 17 are required for output.

In this case, data field 17 is not encoded within the barcode and hence cannot be output by the scanner.

The error beep (upon trigger release or 10 seconds*) notifies the user a required data field is missing.

The missing data field(s) will be represented on the host by an empty (unentered) input field. Hypothetically if field 17 was 220731, if present the host app input box would show “220731”, if not present it would show “ ”.

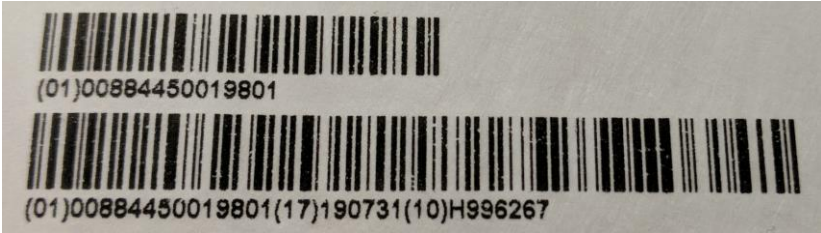
* Parsing Session Timeout (sec), programmable within 123Scan. See FAQ for more information (How long do I have to find all the UDI barcodes (data fields) on the label).

Barcode Example 7 – UDI Label Parsing (GS1)

What is Demonstrated:

This UDI barcode is incorrectly printed. Data field 01 is duplicated (present) in both the top and bottom barcodes.

Symbology: GS1-128 Multiple 1D Barcodes



Output

Scanner Output	{	00884450019801	H996267	190731
-------------------	---	----------------	---------	--------

Explanation of output:

The scanner collected all the UDI data fields present across both barcodes.

After finding the requested fields (01, 17 and 10), the scanner then outputs only those data fields, per the 123Scan UDI Parsing rule, in the desired order (01, 10 then 17).

Barcode Example 8 – UDI Label Parsing (GS1)

What is Demonstrated:

Extra barcode data like data field 20 are ignored and omitted from the scanner’s output since it is not required per the 123Scan UDI Parsing rule.

Start by pulling the trigger and pointing the scanner at field 20, then move the aim to fields 01, 10 and 17.

Symbology: GS1-128 Multiple 1D Barcodes



Output

Scanner Output { 30705031044122 LJZ648 220731

Explanation of output:

Notice data field 20, which is not required by the hospital’s host application, is parsed out of (removed from) the data transmitted to the host.

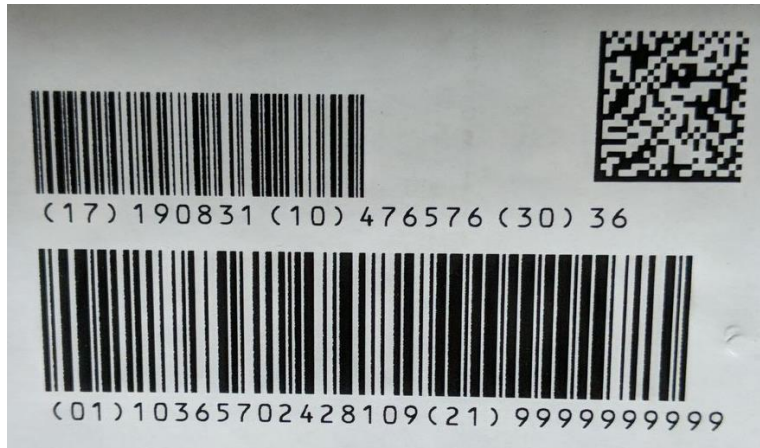
Barcode Example 9 – UDI Label Parsing (GS1)

What is Demonstrated:

Data fields 21 and 30 are not required for the hospital's host application.

The hospital's host application requires data fields 01, 10, 17, which are output upon scanning either the 1D or 2D barcodes.

Symbology: Both GS1-128 Multiple 1D Barcodes and GS1-Datamatrix



Output

Scanner Output	{	10365702428109	476576	190831
-------------------	---	----------------	--------	--------

Explanation of output:

The scanner collected all the UDI data fields present across both barcodes.

After finding the requested fields (01, 10 and 17), the scanner then outputs only those data fields, per the 123Scan UDI Parsing rule, in the desired order (01, 10 then 17).

Notice the data fields (21 and 30), which are not required by the hospital's host application, are parsed out of (removed from) the data transmitted to the host.

Barcode Example 10 – UDI Label Parsing (HIBCC)

What is Demonstrated:

One trigger pull is all it takes.

The scanner will collect all the UDI data fields present across both barcodes and output them to the host application using the standard HIBCC Data Parsing rule from 123Scan.

Symbology: HIBCC Multiple 1D Barcodes



Output

Scanner Output { H8195120108015 16215017 210801

Explanation of output:

The scanner collected all the UDI data fields present in the barcode.

After finding the requested fields (Device Identifier, Batch/Lot Number and Expiration Date), the scanner then outputs only those data fields, per the 123Scan UDI Parsing rule, in the desired order (Device Identifier, Batch/Lot Number and Expiration Date in YYMMDD).

After outputting the three data fields, the scanner outputs an enter key (Enter), moving the cursor to the next line.

Note HIBCC has a complicated data field identifier structure, the symbol(s) that identify a data field for example a single lone + indicates the Device Identifier. In some cases, multiple data field identifiers exist for the same piece of information. For example an expiration date can be identified with any of these thirteen sequences: \$\$, \$\$2, \$\$3, \$\$4,\$\$5, \$\$6, \$\$+, \$\$+2, \$\$+3, \$\$+4,\$\$ +5, \$\$+6 and 14D. All thirteen of these identifier sequences contain the expiration date but each is encoded in a slightly different format for example YYMMDD, YYMMDDHH, MMDDYY, MMY...

Output Format:

[Clear All](#) [Preview/Print](#)

DEVICE IDENTIFIER	×
Tab	×
BATCH/LOT NUMBER	×
Tab	×
EXPIRATION DATE - Format	×
Enter	×

The scanner can decipher all HIBCC data field identifier structures and output it in the desired format, as programmed through 123Scan.

To give you a feel for the complexity of the HIBCC identifier structure, let's dissect the barcode above. But note, all HIBCC UDI barcodes do not follow this exact identifier structure.

"*" at the beginning and end of the text is human readable only and isn't encoded within the barcode data

"+" only, with no \$\$ afterwards indicates the "Device Identifier"

"D" at the end of the first barcode is a check character

\$\$3 indicates the "Expiration Date followed by Lot Number (YYMMDD)"

"+" at the end of the second barcode is a check character

The second last "D" at the end of the second barcode is a Link Character and is intended to link the Primary and Secondary Code Data Structures when encoded in separate linear symbols. The Link Character for the Secondary Data Structure is the last character from the Primary Data String in the Primary Symbol (Check Character).

Barcode Example 11 – GS1 Label Parsing

What is Demonstrated:

Capture data fields: 00, 02, 10, 37, 400

But output them in this order with tabs and an enter: 00, tab, 400, tab, 02, tab, 10, tab, 37, enter

Ignore data field: 11

Symbology: Both GS1-128 Multiple 1D Barcodes



Output

Scanner Output	{	356010110000189102	100120	05010677911528	14133PT100	120
-------------------	---	--------------------	--------	----------------	------------	-----

Output Format:

[Clear All](#)[Preview/Print](#)

00 - SSCC

Tab

400 - ORDER NUMBER

Tab

02 - CONTENT

Tab

10 - BATCH/LOT NUMBER

Tab

37 - COUNT

Enter

The UDI Parsing rule from 123Scan (above) specifies the output fields, their order and separator fields (tabs & enter). Data field 11 is ignored (not output to the host).

Barcode Example 12 – GS1 Label Parsing

What is Demonstrated:

Capture data fields: 00, 02, 10, 37, 400

But output them in this order with tabs and an enter: 00, tab, 400, tab, 02, tab, 10, tab, 37, enter

Ignore data field: 17

Symbology: Both GS1-128 Multiple 1D Barcodes and GS1-Datamatrix



Output

Scanner Output	{	384105002114064072	PT20010A	25601050026515	140618	144
-------------------	---	--------------------	----------	----------------	--------	-----

Output Format:

[Clear All](#)[Preview/Print](#)

00 - SSCC	X
Tab	X
400 - ORDER NUMBER	X
Tab	X
02 - CONTENT	X
Tab	X
10 - BATCH/LOT NUMBER	X
Enter	X
37 - COUNT	X
Enter	X

The UDI Parsing rule from 123Scan (above) specifies the output fields, their order and separator fields (tabs & enter). Data field 17 is ignored (not output to the host).

Barcode Example 13 – GS1 Label Parsing

What is Demonstrated:

Capture data fields: 00, 11, 37, 400

But output them in this order with tabs and an enter: 00, tab, 400, tab, 11, tab, 37, enter

Ignore data fields: 02, 10, 17

Symbology: Both GS1-128 Multiple 1D Barcodes and GS1-Datamatrix

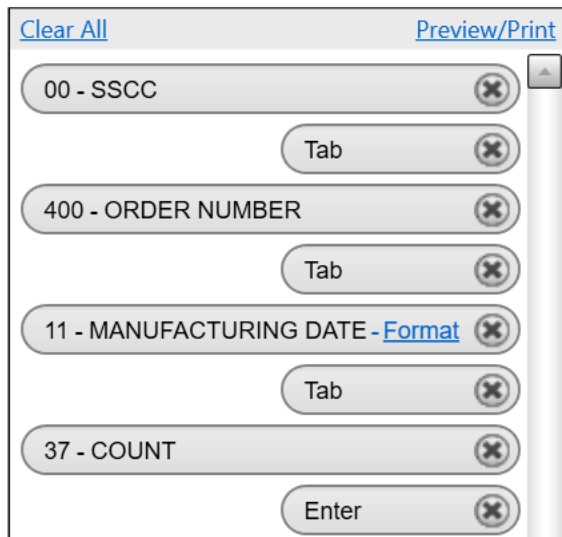


Output

Scanner Output { 384105002114064072 PT20010A 144

Missing Field 11

Output Format:



The UDI Parsing rule from 123Scan (above) specifies the output fields, their order and separator fields (tabs & enter). Data field 02, 10 and 17 are ignored (not output to the host).

Note the 123Scan Parsing rule specifies data fields 00, 400, 11 then 37 are required for output.

In this case, data field 11 is not encoded within the barcode and hence cannot be output by the scanner.

The error beep (upon trigger release or 10 seconds*) notifies the user a required data field is missing.

The missing data field(s) will be represented on the host by an empty (unentered) input field. Hypothetically if field 11 was 220731, if present the host app input box would show “220731”, if not present it would show “ ”.

* Parsing Session Timeout (sec), programmable within 123Scan. See FAQ for more information (How long do I have to find all the UDI barcodes (data fields) on the label).

Barcode Example 14 – Adding Function Key(s) using ADF Post Processing to a Data Parsed Output

What is Demonstrated:

This example demonstrates how to add a function key(s) to Data Parsed output using ADF.

Desired Output: <F12 Key> 01 10 <F2 Key> 17 <F11 Key>

The function keys do not exist within the scanned barcode data.

Add a F12 key as a prefix, before data field 01 (Device Identifier)

Add a F2 key after data field 10 (Batch/Lot Number)

Add a F11 key as a suffix, after data field 17 (Expiration Date)

Symbology: Both GS1-128 Multiple 1D Barcodes and a GS1-Datamatrix



Output

Scanner Output	{	<F12 Key>	20886705001795	L862995	<F2 Key>	230430	<F11 Key>
----------------	---	-----------	----------------	---------	----------	--------	-----------

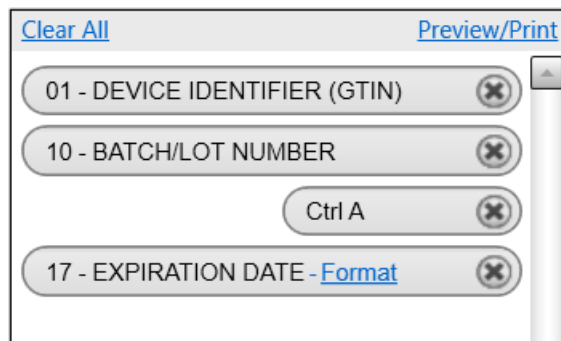
Explanation of output:


Process Overview:

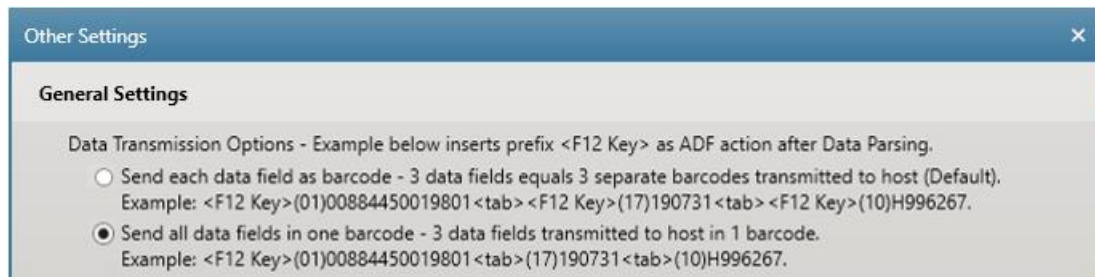
- Using ADF, function key(s) is added to the Data Parsing output.
- Step 1 inserts a character into the scanned data, that is replaced with a function key(s) in step 3.
- Step 2 Change the Data Transmission Option so each field will be sent as a sperate barcode
- Step 3 appends a prefix <F12 Key>, replaces SOH (Ctrl A) with a <F2 Key>, and appends a suffix <F11 Key> to the barcode data.

Step 1 - Write a Data Parsing rule (shown below) that inserts a “place holder character” where the function 2 key is required. In the example below, a “Ctrl A” is used as the place holder character. You can use any character as the place holder, if it is never present in any of the data parsing barcodes (UDI Scan+, Label Parse+, Blood Bag Parse+) that are to be scanned.

Output Format:



Step 2 - Change to the Data Transmission Option shown below under the  **Other Settings** menu.



Step 3 - Write an ADF rule that replaces the “place holder character” with the desired function 2 key and inserts the prefix <F12 Key> and suffix <F11 Key>.

First select “Data Parsed Code” as the code type, see image below.

Second, insert the six Actions, as shown below.

Advanced Data Formatting

Rule list: 1 ▾

Rule 1

Criteria ▾ 1% Total: 236b

Rule set: Default ▾

Rule name UDI ADF Parsing

Code type Data Parsed Code
[Select codes](#)

Code Length Any ▾ 1

String Trigger bar code

String at position ANY ▾ 1

Actions Total:5

1. Send Function Key <F 12> (X)
2. Send Up to <SOH> Start of... (X)
3. Send Function Key <F 2> (X)
4. Skip Ahead <1> (X)
5. Send All that remains (X)
6. Send Function Key <F 11> (X)

[Add action](#)

- ← Insert Prefix <F12 Key> before scanned data
- Send up to place holder character (Ctrl A, also known as SOH).
Replaces the place holder character (Ctrl A) with <F2 Key>.
Skip over place holder character
- ← Insert Suffix <F11 Key> after scanned data

- ☐ UDI Parsing / Scan+ (GS1, HIBCC and ICCBBA)
- ☐ Label Parsing / Parse+ for GS1 (Logistics, Raw Material, Product ...)
- ☐ Blood Bag Parsing / Parse+ (ISBT 128)

Note “Data Parsed Code” applies to these data formatting options

Barcode Example 15 – Blood Bag Scanning

What is Demonstrated:

In one trigger pull, scan the blood bag label and output the 4 most commonly used barcodes. All 4 barcodes are required for most host systems and the barcodes are always printed on a blood bag label.

01 – Donation Identification Number (DIN)	Always on label
02 – Blood Group	Always on label
03 – Product Code	Always on label
05 – Expiration Date and Time	Always on label

Symbology: ISBT 128



Output

Scanner Output	{	A99991512345600	8400	E0361V00	0153262359
----------------	---	-----------------	------	----------	------------

Explanation of output:

The default Blood bag parsing rule from 123Scan outputs the data in the manner illustrated (above).

The scanner will capture and output the four most commonly used barcodes on the label. These 4 barcodes are required by the ICCBBA Blood Bag specification to be present on all blood bag labels.

Output Format:

[Clear All](#)[Preview/Print](#)

01 - DONATION IDENTIFICATION...	X
Tab	X
02 - BLOOD GROUPS	X
Tab	X
03 - PRODUCT CODE	X
Tab	X
05 - EXPIRATION DATE AND TIME	X
Enter	X

Barcode Example 16 – Blood Bag Scanning

What is Demonstrated:

In one trigger pull, scan the blood bag label and output these barcodes. Of the 5 desired barcodes, 4 barcodes are required / always printed on a blood bag label. The fifth barcode, field 07, is an optional field meaning the label may or may not include it on the blood bag (the ISBT 128 Blood Bag specification does not require field 07 to be present on the label). In this example field 07 is present on the label.

01 – Donation Identification Number (DIN)	Always on label
02 – Blood Group	Always on label
03 – Product Code	Always on label
05 – Expiration Date and Time	Always on label
07 – Collection Date and Time	Optional (Not always present) on label

The Blood Bag Parsing Session Timeout is set to 0.5 seconds

Under “Other Settings”, “Handling a label not containing all required data fields” is set to transmit acquired data field(s) and emit an error beep followed by a good decoded beep.

Symbology: ISBT 128



Output

Scanner Output { A99991512345600 8400 0152841415 E0361V00 0153262359

Data Parsing on Zebra Scanners

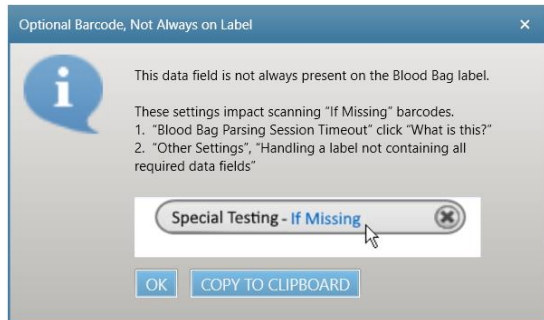
Zebra Technologies – v2

Explanation of output:

The blood bag parsing rule outputs the data in the manner illustrated (above).

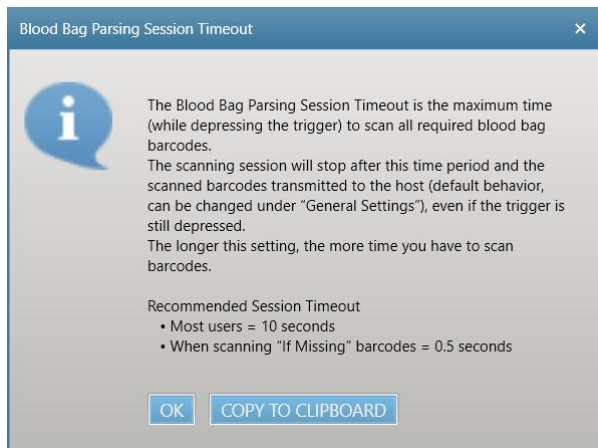
The 4 data fields 01, 02, 03, 05 are always present on the label and were discussed in Example 14.

But data field 07, Collection Date and Time, is not always present on a blood bag label. 123Scan handles blood bag barcodes that are sometimes on a label with a “If Missing” label. Click the “If Missing” link will provide the information outlined



below.

1. “Blood Bag Parsing Session Timeout” click “What is this?” The help menu is shown below.



Output Format:

[Clear All](#) [Preview/Print](#)

01 - DONATION IDENTIFICATION...	Tab
02 - BLOOD GROUPS	Tab
07 - COLLECTION DATE... - If Missing	Tab
03 - PRODUCT CODE	Tab
05 - EXPIRATION DATE AND TIME	Enter

[Other Settings](#) | [Memory used](#) [Back](#)

Blood Bag Parsing Session Timeout (sec) [What is this?](#)

Parsing Prioritization Time (ms) [What is this?](#)

2. Under “Other Settings / General Settings”, “Handling a label not containing all required data fields” is set to transmit acquired data field(s) and emit an error beep followed by a good decoded beep.

General Settings

Handling a label not containing all required data fields

- ☒ Transmit acquired data field(s) and emit an error beep. (Default)
The missing data field(s) will be represented by an empty (unentered) entry field. For example if the scanned data for field 1 should be abc123, if present the host app input box would show "abc123", if not present it would show " ". If a separator field is programmed for output, it will be sent even if the scanned data is not present. The scanner will emit an error beep (2 high long beeps) followed by a good decode beep.
- ☐ Same as above, except the scanner will only emit a good decode beep.
- ☐ No data fields will be output by the scanner and the scanner shall emit an error beep. The scanner will emit an error beep (5 long low beeps).

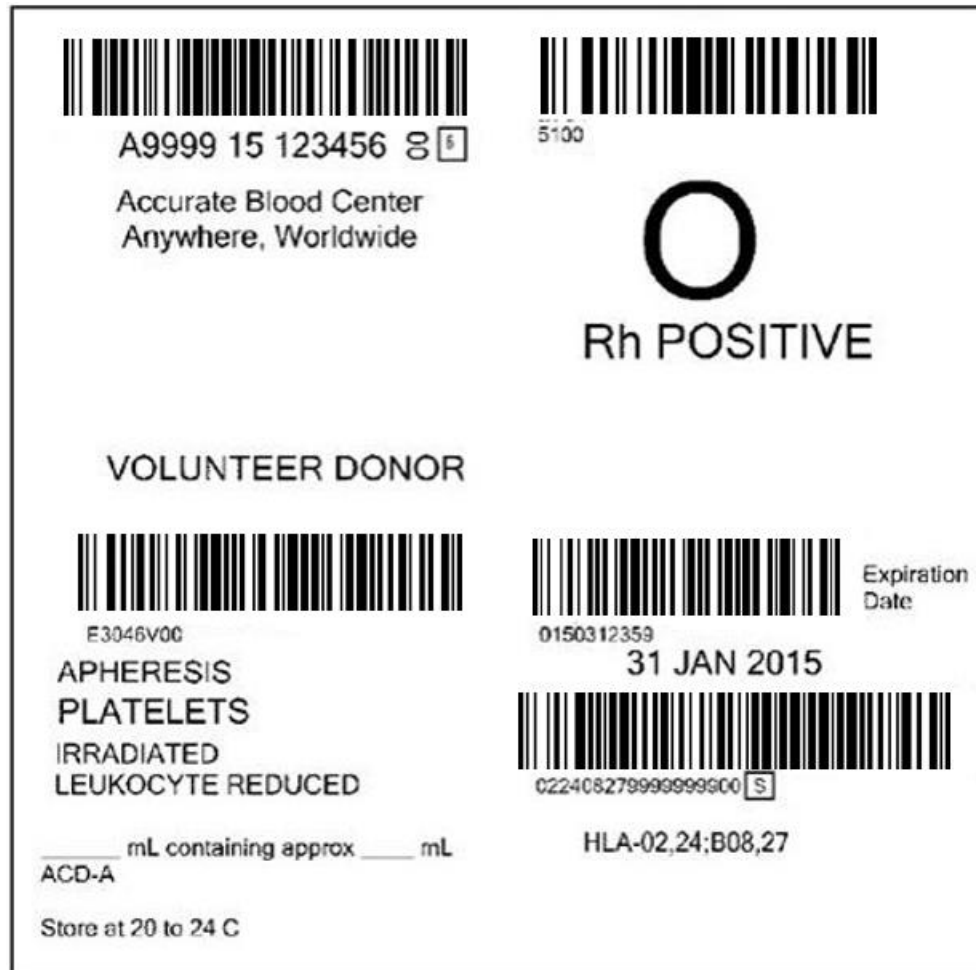
Barcode Example 17 – Blood Bag Scanning

What is Demonstrated:

Building on the previous example. In this example field 07, Collection Date and Time, is not present on the label.

01 – Donation Identification Number (DIN)	Always on label
02 – Blood Group	Always on label
03 – Product Code	Always on label
05 – Expiration Date and Time	Always on label
07 – Collection Date and Time	Optional (Not always present) on label

Symbology: ISBT 128

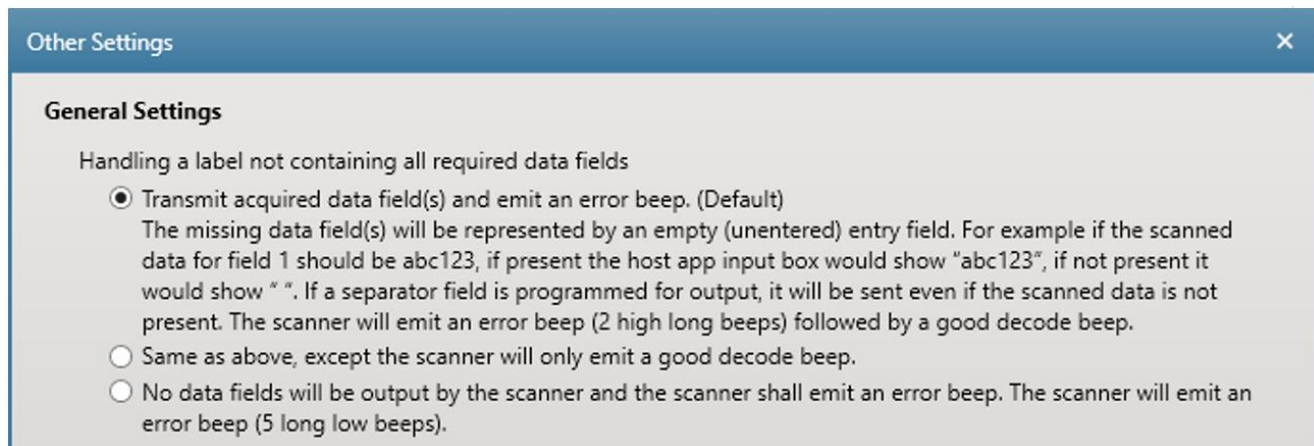


Output

Scanner Output	{	A99991512345600	5100	E3046V00	0150312359
-------------------	---	-----------------	------	----------	------------

Explanation of output:

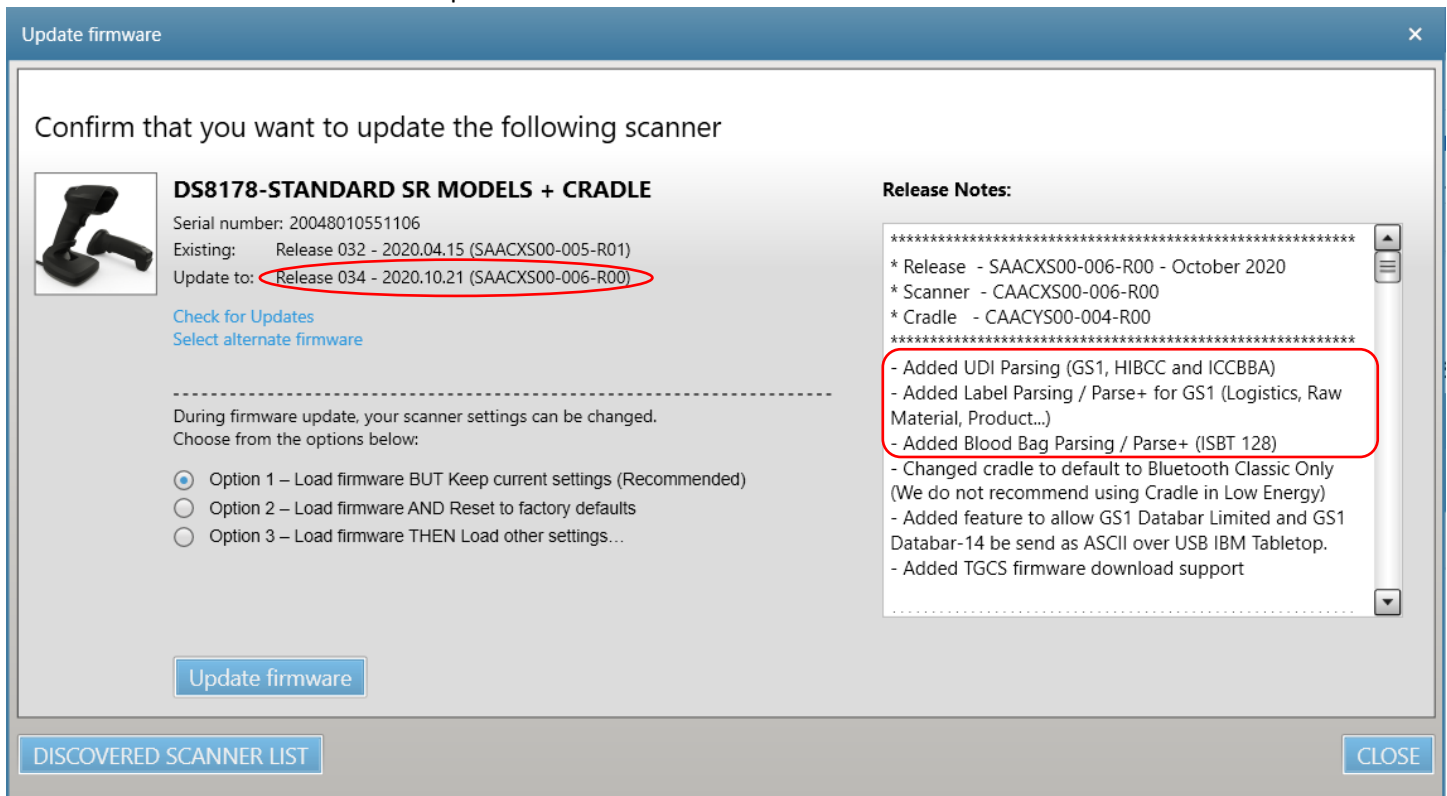
Because one of the programmed fields is not present on the barcode (field 07) and “Handling a label not containing all required data fields” is set to transmit acquired data fields (see below for more details), the output is missing field 07’s data.



The missing data field is represented on the host by an empty (unentered) input field. Hypothetically if field 07 was 220731, if present on the barcode label the host app input box would show “220731”, if not present it would show “ ”.

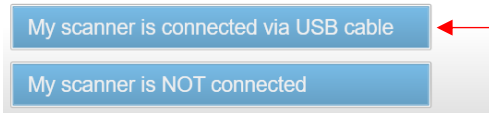
Appendix B: Programming Your Scanner for Data Parsing

1. First watch this Data Parsing How-to-Video:
 - i. www.zebra.com/ScannerHowToVideos
 - ii. Video Name: Zebra 123Scan: Creating a Data Parsing Rule (UDI, GS1 Label, Blood Bag)
2. **Use 123Scan 5.3.6** (version 5.3 Build 6.0) or later.
3. Using 123Scan, **upgrade your DS8108 or DS8178 scanner to the required firmware** (already present in 123Scan)
 - a. Minimum Firmware Version
 - i. DS8108 SR Model = Release 026 - 2020.10.21 (CAACXS00-006-R00)
 - ii. DS8178 SR Model = Release 034 - 2020.10.21 (SAACXS00-006-R00)
 - b. Upgrade Process
 - i. From the start screen click “Update scanner firmware”.
 - ii. Then click “Update firmware”



4. **Program your scanner for data parsing using 123Scan**
 - a. **Create a configuration file for data parsing**
 - i. **Plug your scanner's USB cable into 123Scan**
 - ii. From 123Scan's start screen **select “Create new configuration file”**.
 - iii. Let 123Scan **select the correct plugin** for your scanner.

Select an option to start creating a configuration file



iv. **Name the configuration file “Data Parsing”.**

Path = 123Scan / Config file / Name and notes.

Name and notes

Please be sure to give your settings a configuration name. You can also add notes and give the configuration name a version number.



v. **Enable the required GS1 barcode symbologies (UDI and GS1 Label Parsing).**

1. See trouble shooting section for details

vi. **Enable the required ICCBBA barcode symbologies (Blood Bags and UDI Parsing).**

1. See trouble shooting section for details

vii. **Create a Parsing Rule.**

Path = 123Scan / Config file / Modify Data.

1. Access the UDI Parsing programming screen by clicking the 3 buttons in the sequence shown below.

Modify Data

If required, create a **data formatting** rule to modify scanned data before its transmission to the host.

- ☐ Send data as scanned, don't append any keystrokes. (*Factory Default*).
- ☐ Append an enter key: Adds an Enter key to the end of scanned data.
- ☐ Append a tab key: Adds a Tab key to the end of scanned data.

- ☒ Program complex data modifications or select a rule from your Data Formatting Library. ^ **Step 1**

☐ Basic

Prefix / Suffix Simple Formatting

☒ Advanced

[View / Edit](#)

ADF - Advanced Data Formatting (Scan one bar code)

Step 3 →

[View / Edit](#)

Scripting Capabilities

- ☐ MDF - Multicode Data Formatting (Scan many bar codes at once)
- ☒ UDI Parsing / Scan+ (GS1, HIBCC and ICCBBA) **Step 2**
- ☐ Label Parsing / Parse+ for GS1 (Logistics, Raw Material, Product ...)
- ☐ Blood Bag Parsing / Parse+ (ISBT 128)

2. Enable the data parsing editor by putting a check in the checkbox

Parsing Standards ⓘ



3. Use the default Parsing rule, pre-programmed in 123Scan.
You are not required to do anything on this step.

Output Format:

[Clear All](#)
[Preview/Print](#)

01 - GTIN	<input type="button" value="X"/>
Tab	<input type="button" value="X"/>
10 - BATCH/LOT NUMBER	<input type="button" value="X"/>
Tab	<input type="button" value="X"/>
17 - EXPIRATION DATE - Format	<input type="button" value="X"/>
CR	<input type="button" value="X"/>

For details on this UDI Parsing rule's output, see Barcode Example 1.

4. Save configuration file to your PC's hard drive.

Path = 123Scan / Config file / Load and print.

Save to PC

- viii. **Load the configuration file to your scanner.** Path = 123Scan / Config file / Load and print.

Either print and scan a programming barcode or download the configuration file electronically.

Load and print

Choose from the options below to program your scanner

Save to PC

Print bar code

Load to scanner

- ix. **Close 123Scan**

You are now ready to scan UDI barcodes into Microsoft Notepad. See Examples 1 through 16, in this document.