



701 Decatur Avenue North, Suite 107 • Minneapolis, MN 55427 • +1-763-746-8034 • www.rjs1.com

Explanation of the changes between RJS D4000 Firmware version A.03/A.05 and firmware version A.06

The D4000 firmware A.06 has a number of functional changes from the previous A.03 and A.05 firmware versions:

- Updated Terminology - The firmware will update the terminology used in both the symbology and the sub-symbology names as listed on the Setup menu options and the printed reports
- Addition of the Decodability Percentage and Grade to the Pass/Fail Analysis Screen (Displayed after a scan is captured)
- Full GS1-128 Application Identifier Support - Current D4000 units have a limit of 32 characters, and are missing some newer Application Identifiers (AIs). The new A.06 allows for the full GS1 limit of 48 data characters to be inspected and will not impose a limit on the maximum number of AIs in the bar code.
- Improvements for Interleave 2 of 5 and Code 39 - ratio testing will be upgraded, to report ratio warnings in addition to ratio failures

Setup Menu Options (Applies to D4000 Auto Optic and Laser)

Setting	Version	
	A.03 or A.05	A.06
Decode 3of9 as	USS 3of9	Code 3of9
Decode I2of5 as	Case Code	ITF14 Case Code
Decode I2of5 as	USS 2of5	Std I2of5
Decode C128 as	N/A	Std 128
Decode C128 as	N/A	GS1-128
Database Storage	0-20kb 0-20kb	N/A (all Database)

Pass/Fail Analysis Screen (Applies to D4000 Laser ONLY)

	Version	
	A.03 or A.05	A.06
Description of screen information:	<i>Displays Bar Tolerance Chart</i>	<i>Displays ISO/ANSI Decodability results</i>
Screen Examples:	<div style="border: 1px solid black; padding: 5px; width: fit-content; margin: auto;"> *1234ABCD* Code 3of9 -100% Tol. +100% -----RRARR+++ </div>	<div style="border: 1px solid black; padding: 5px; width: fit-content; margin: auto;"> *1234ABCD* Code 3of9 D/bility % .64 D/bility Grade A </div>

GS1-128 Testing (Applies to D4000 Laser with version A.06 ONLY)

Testing Parameters

The D4000 Laser will inspect all GS1 Application Identifier (AI) content and length, this includes:

- FNC1 (Variable length AIs must start with a FNC1 character)
- Multiple AI support (unlimited number of AIs in a bar code)
- Date encodation (AIs with dates will be tested for proper formatting)
- GTIN prefixes (Some AIs require a prefix digit in the GTIN),
- Linked AIs (Some AIs require another AI to be encoded in the bar code)
- Numeric requirements (Some AIs are numeric only)
- Testing to ensure 48 data characters (excludes sub-set changes) are not exceeded

FNC1 Testing

When a Code 128 symbol is decoded **AND** the first character after the Start character is **FNC1** then the symbol must follow the GS1-128 format and the verifier must have the following Code 128 sub-specifications setting:

Decode C128 as
GS1-128

When a Code 128 symbol is decoded with the Code 128 sub-specifications setting of **Std 128** but the first character after the Start character **is** a **FNC1** then the following error will be displayed:

Std 128
Format Warning

When a Code 128 symbol is decoded with the Code 128 sub-specifications setting of **GS1-128** and the first character after the Start character **is not** a **FNC1** then the following error will be displayed:

GS1-128
Format Warning

Data Content Testing

When Code 128 sub-specifications setting is GS1-128 and a GS1-128 bar code is inspected an additional screen will be inserted into the Data Analysis screens:

GS1-128
Acceptable
AI (01)

Example of a bad check digit in the GTIN:

GS1-128
Bad Mod. Check
AI (01)

Data Content Testing (continued)

Example of an alpha-character in a numeric only AI:

GS1-128
Expected Numeric
AI (3931)

Note:

If a bar code has multiple errors **only** the first error will be displayed

Example of a bar code with more than 48 data characters:

GS1-128
Exceeds 48 Chars
AI (250)

Note:

If a bar code data length is exceeded, the AI that exceeded the 48 character limit will be displayed

Example of an invalid date encoded in an AI:

GS1-128
Out-Of-Range
AI (17)

Note:

For Month and Year only encodes the Day may be encoded as "00"