VIDEOJET SIMPLICiTY SERIES PRINTERS

WSI COMMUNICATION PROTOCOL

P/N 463392-01

Revision: AC, March 2023

Table of Contents

1 Introduction

This document outlines the WSI Simple Protocol for the SIMPLICiTY Printers. It is a simple to implement protocol and it provides the following:

- An upgrade path for existing Videojet customers with applications utilizing the simpler protocols which existed in earlier Videojet products.
- Communication with PLC’s which are not flexible enough to operate more complex protocols.
- A quick and simple protocol which can be used to provide an individual integrated solution to a coding or marking application.
- A fast, light-weight protocol for use in higher speed printing applications.
- Remote operation of the ink jet (starting and stopping).

This protocol is not the full WSI Protocol and additional features such as the following are not available:

- Operation using RS-485 multi-drop environment.
- Creation or deletion of jobs, user fields and logos.

The protocol operates on ports COM1, COM2, or on a TCP/IP stream using port 3100 and supports two encoding formats:

- ASCII for legacy systems supports the extended ASCII range 0x80 to 0xFF in addition to the standard ASCII range,
- Partial UTF-8, which uses UTF-8 encoding for some fields, which is highlighted in this document.

2 Connectors and Cables

Note: *All references to 1580 refer to printer models 1580, 1580 + and 1580 C, unless otherwise stated.*

2.1 Connector Panel

2.1.1 Videojet 1240/1280 Connector Panel

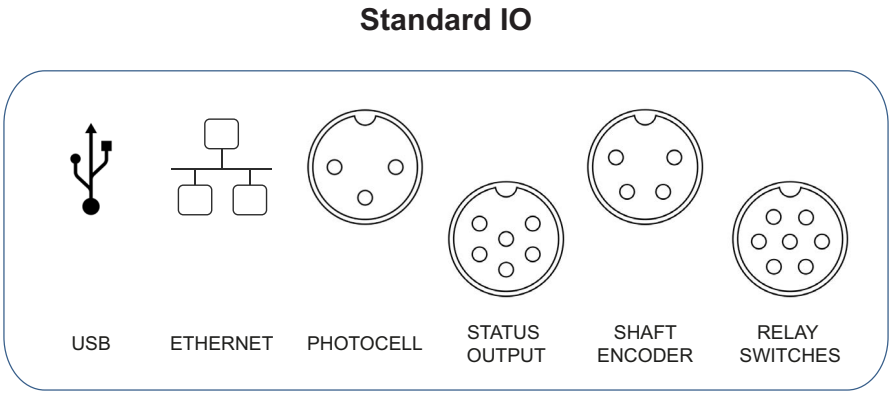


Figure 2-1: Connector Panel (Videojet 1240/1280)

2.1.2 Videojet 1580 Connector Panel

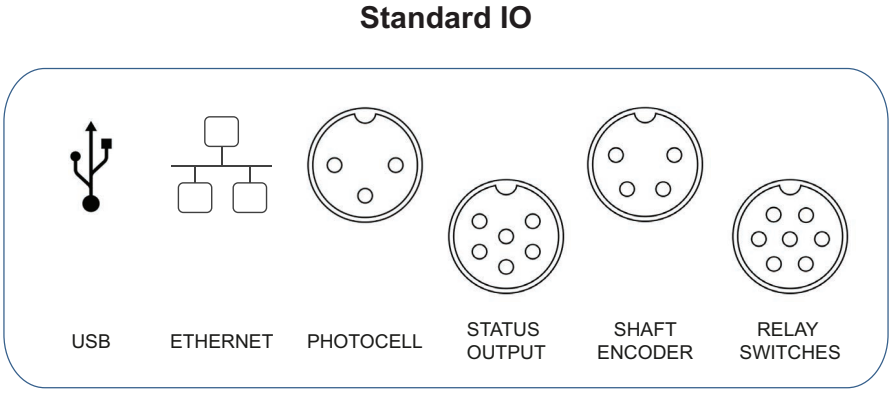


Figure 2-2: Connector Panel (Videojet 1580)

2.1.3 Videojet 1880 Connector Panels

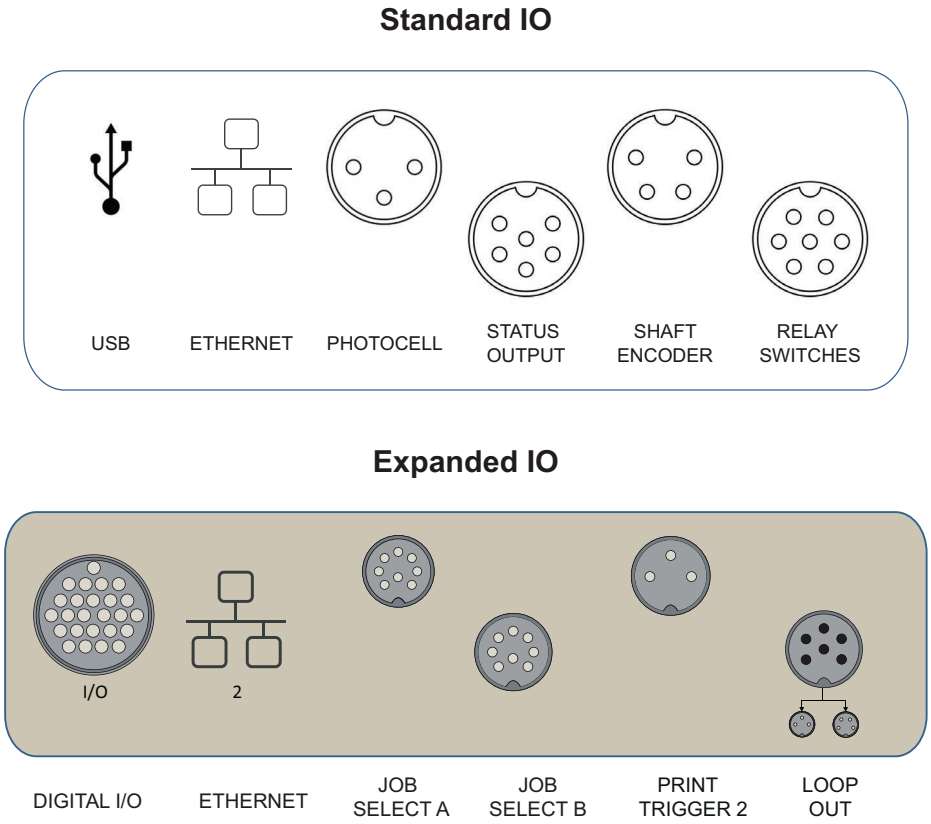
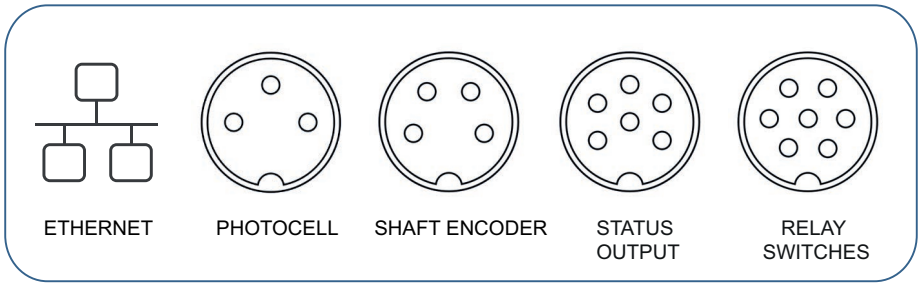


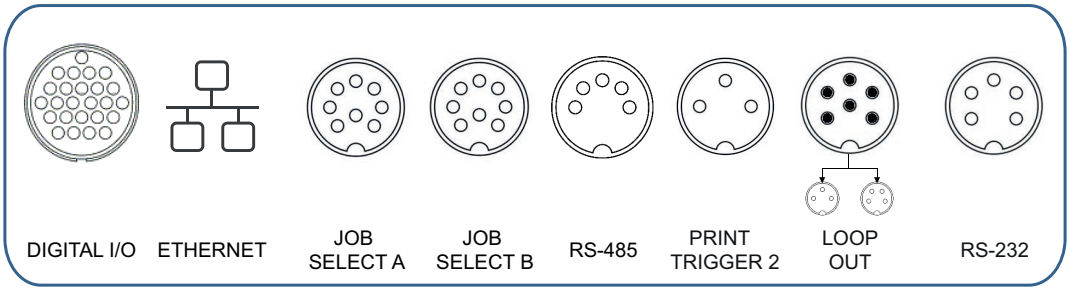
Figure 2-3: Connector Panels (Videojet 1880)

2.1.4 Videojet 1860 Connector Panels

Standard IO



Expanded IO



Expanded IO RS Connector only

Figure 2-4: Connector Panels (Videojet 1860)

2.2 Ethernet Communication

The Ethernet connector is used to connect the printer to a remote computer or network to import data or control the printer remotely.

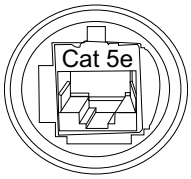


Figure 2-5: Ethernet Connector

The standard IO for each printer has one Ethernet Port.
The expanded IO for 1860 and 1880 printer has a second Ethernet port.

Available Parts:

Part Number	Description
SP223213	External Ethernet Connector Cable (5 meter)

Table 2-1: Parts List

2.3 RS-485 Communication

RS-485 connection for WSI Protocol is not supported on the SIMPLICiTY printers.

Videojet 1240/1280/1580/1880

There is no RS-485 connection available on the 1240/1280/1580/1880 printer.

Videojet 1860

The printer has one RS-485 port, this port is not available for WSI communication due to the asynchronies of the protocol.

2.4 RS-232 Communication

2.4.1 Videojet 1240/1280/1580/1880

1240/1280/1580/1880 RS-232 communication is achieved using a USB to RS-232 cable. There is no dedicated RS-232 connector.

Typical RS-232 Connection

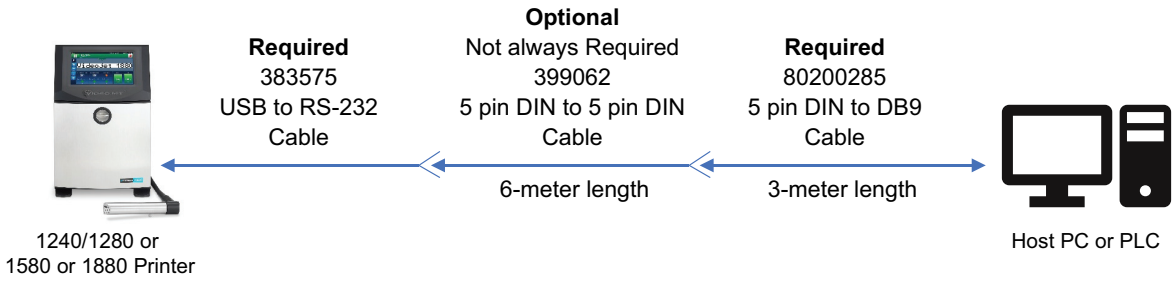


Figure 2-6: Typical RS-232 Connection (Videojet 1240/1280/1580/1880)

Available Parts:

Part Number	Description
383575	USB to RS-232 Cable
399062	RS-232 Extension Cable (6M)*
80200258	RS-232 Serial Communication Cable (3M)

Table 2-2: Parts List

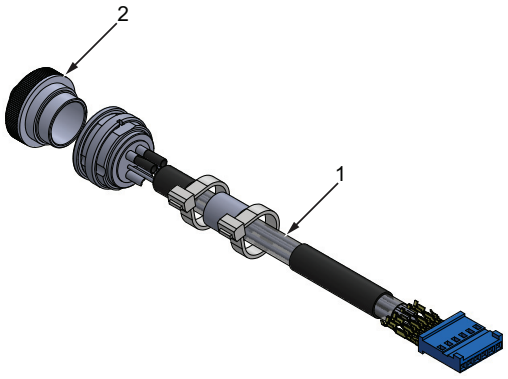
*The Videojet 6M extension cable (P/N: 399062) is 5 pin DIN to 5 pin DIN optional cable which connects the printer cabinet to the RS-232 cable (P/N: 80200258).

***Note**: It is recommended that maximum length of the RS-232 cable does not exceed 15 m (50 ft).*

Please refer 2.4.3 RS-232 Connections on page 2-7 for connections.

2.4.2 Videojet 1860

Videojet 1860 offers a dedicated RS-232 connector with either expanded IO or with the optional RS-232 connector. The RS-232 connector can be ordered separately if required.



1. MCB to RS-232 Cable **2. Binder Protection Cap**

Figure 2-7: RS-232 Connector

Available Parts:

Part Number	Description
611197	RS-232 Connector

Table 2-3: Parts List

Typical RS-232 Connection

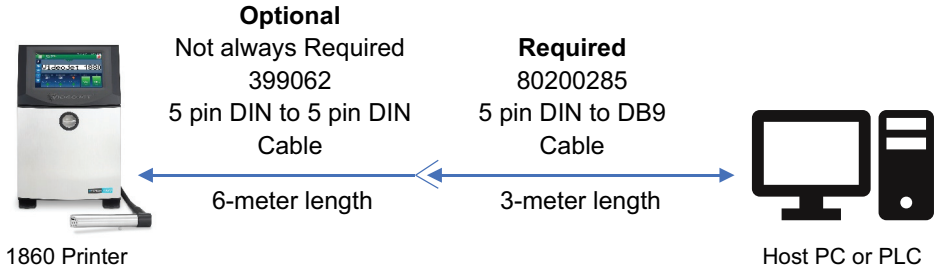


Figure 2-8: Typical RS-232 Connection (Videojet 1860)

Available Parts:

Part Number	Description
399062	RS-232 Extension Cable (6M)*
80200258	RS-232 Serial Communication Cable (3M)

Table 2-4: Parts List

*The Videojet 6M extension cable (P/N: 399062) is 5 pin DIN to 5 pin DIN optional cable which connects the printer cabinet to the RS-232 cable (P/N: 80200258).

***Note**: It is recommended that maximum length of the RS-232 cable does not exceed 15 m (50 ft).*

Please refer 2.4.3 RS-232 Connections on page 2-7 for connections.

2.4.3 RS-232 Connections

5-pin DIN Connector

- 1860 RS-232 5-pin DIN connector (COM1)
- USB to RS-232 Cable 5-pin DIN connector

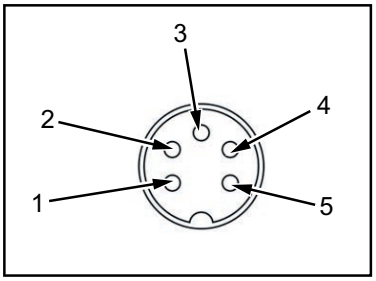


Figure 2-9: COM1 Connector (Female) - Cabinet Side View

DIN Pins	Function
1	0 V isolated
2	Transmit data out from the printer
3	Receive data to the printer
4	DTR output from the printer
5	DCD input to the printer

Table 2-5: Pin Function

5-pin DIN to DB9 Cable Connectors

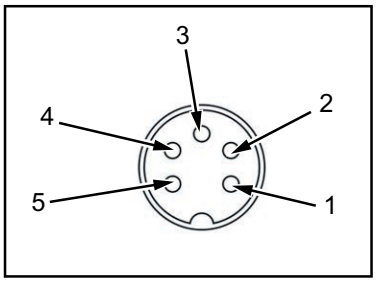


Figure 2-10: COM1 Connector (Male) - Cabinet Side View

DIN Connector (Male Pins)		RS-232 Connector (Female Pins)	
DIN Pins	**Function**	**DB9 Pins**	**Function**
1	Common	5 (Black Wire)	Common
2	TX	2 (Green Wire)	RX
3	RX	3 (Red Wire)	TX
4	DTR	6 (White Wire)	DSR
5	DCD	1 (Orange Wire)	DCD
		7+8 (connected)	

Table 2-6: COM1 Pin Connection Information – DIN to DB9

Note: *The connector cables are soldered to the rear of the connector as per the connector pins from the cabinet side view.*

This view is from male pins DIN to DB9 female pin point to point. It is recommended to perform continuity test for cables.

Available Parts:

Part Number	Description
500-0036-582	RS-232 Male DIN 5 Pin Connector
399062	RS-232 Extension Cable (6M)
80200258	RS-232 Serial Communication Cable (3M)

Table 2-7: Parts List

Figure 2-11: Serial Communication Cable (P/N: 80200258)

2.5 General Information

1. Input stack buffer first in first out (FIFO) buffer size: 100 jobs.
   a. The printer accepts up to 100 jobs, the maximum length of each job is up to 500 characters.
2. RS-232 Protocol (These settings are customer configurable)
   a. Only one RS-232 port available for 1240/1280/1580/1860/1880
   b. Baud Rate: 9600, 19200, 38400, 57600, 115200
   c. Word length: 7, 8
   d. Parity: None, Odd, Even, Mark, Space
   e. Stop Bit: 0, 1
   f. XON/XOFF Flow control

2.6 Ethernet Information

Ethernet connection is Ethernet TCP/IP 100 base T. Table 2-8 shows the port number for different ports. These ports are just a suggestion to match the older 1000 series printers that were fixed. For 1240/1280/1580/1860/1880 printers, the port numbers are configurable via the user interface.

Port	Port Number
WSI Communication Port	3100
WSI Communication Port (Videojet 1610 DH Head 2)	3102

Table 2-8: Port Number

Setup the following settings for Ethernet connection:

1. Ethernet TCP/IP
2. IP Address
3. Subnet setting
4. Default Gateway setting
5. DHCP setting (DHCP is not currently supported)
6. Protocol

To communicate with the Ethernet TCP/IP, use the same command set and ASCII characters used for RS-232. The printer communicates with full ASCII characters. All the characters sent from the printer are ASCII characters. Your program must be able to send printable ASCII characters and non-printable ASCII characters. The printer can be setup to utilize Unicode (UTF-8). This will be needed if you need to access characters beyond 255.

This document shows many of the commands in hexadecimal form. This is for ease of documentation purposes. All characters are sent and received in ASCII. In this manual the hex value is denoted when characters are placed between “[ ]”. The digits between the brackets are the hex value of the ASCII characters you want to send or that you receive.

Note: *UTF-16 Unicode is not supported.*

2.7 WSI Examples

Example 1:

This is a command to request the software version currently on the printer. After sending this command the printer will return to the host, the ASCII character string for the version of software currently on the printer.

Send String: [02]H[03]

[02]: is the hex value of the STX character in ASCII

H: is the ASCII character “H” this is the command character request part number

[03]: is the hex value of the ETX character in ASCII

This will return to the host the printer’s current software number: [02]021140293[03]

Example 2:

This is to select a job from the printer’s memory. Create a job on the printer’s keyboard called “MESSAGE1”.

Send String: [02]MMESSAGE1[03]

[02]: is the hex value of the STX character in ASCII

M: is the ASCII character “M” this is the command character for job select from memory

MESSAGE1: is the ASCII characters for the name of the stored job you want to recall

[03]: is the hex value of the ETX character in ASCII

This will recall the MESSAGE1 to the current job.

The printers will send the host to the checksum for the command just sent: $[38][33]

***Note**: Your program must be able to send non-printable ASCII characters.*

3 Getting Started

3.1 Setup WSI Communication Protocol

Do the following tasks to setup the printer for WSI communication protocol:

1. Navigate to the Home screen of the printer. Touch the *Login* button as shown in Figure 3-1.

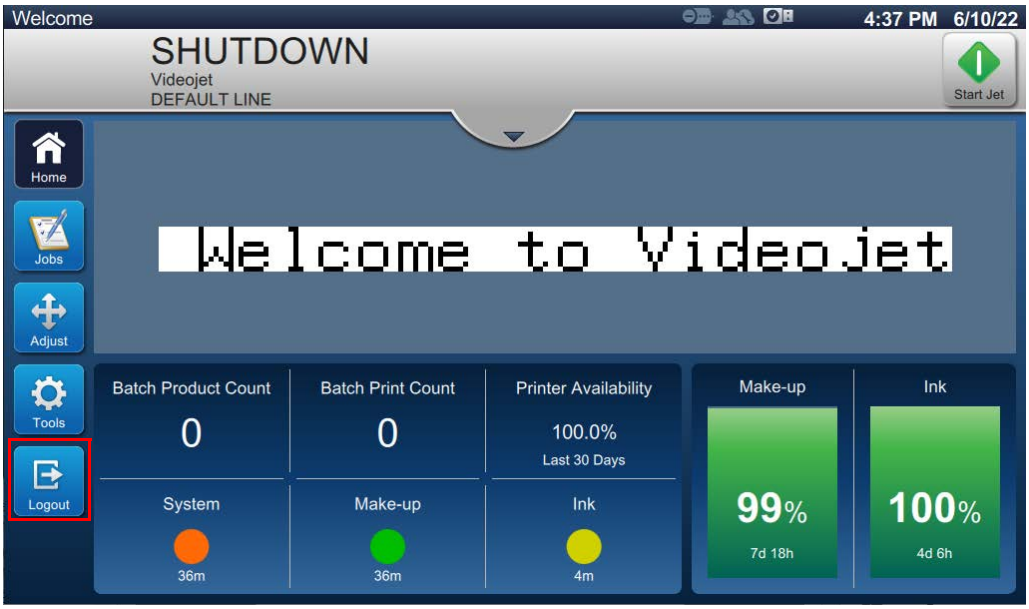


Figure 3-1: Home Screen

2. The Login to System screen appears as shown in Figure 3-2.

Figure 3-2: Login to System Screen

3. Touch the *Role* drop down button and select Admin role. Enter the password in the Password field (default password for Admin role: 3333).
4. Touch the *Tools* button on the Home screen to access the Tools screen and select the *Communications* button as shown in Figure 3-3.

Figure 3-3: Tools Screen

5. The Communications screen appears as shown in Figure 3-4.

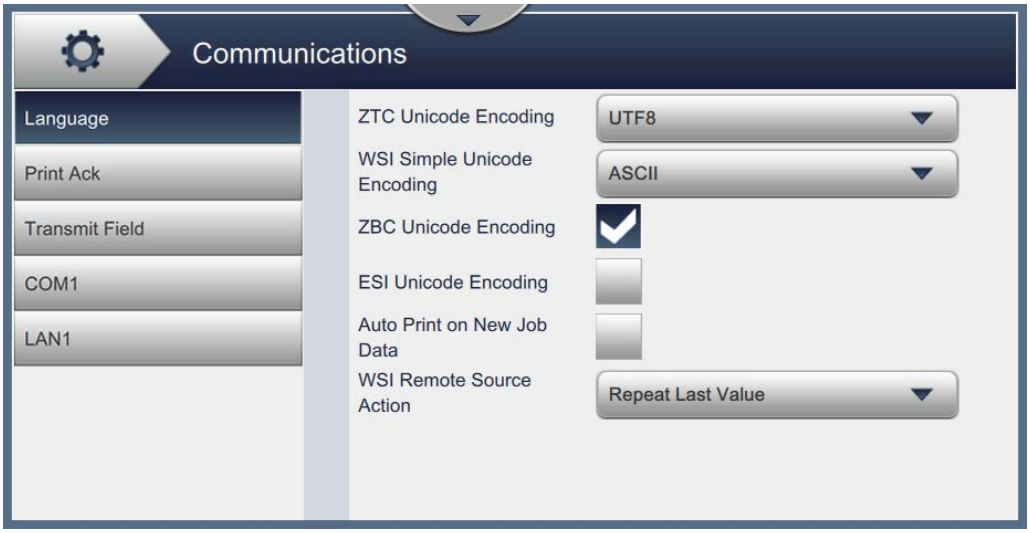


Figure 3-4: Communications Screen

6. Touch the *WSI Simple Unicode Encoding* drop down list (see Figure 3-4) and select the required encoding mode as shown in Figure 3-5. Touch the *OK* button to confirm the selection.

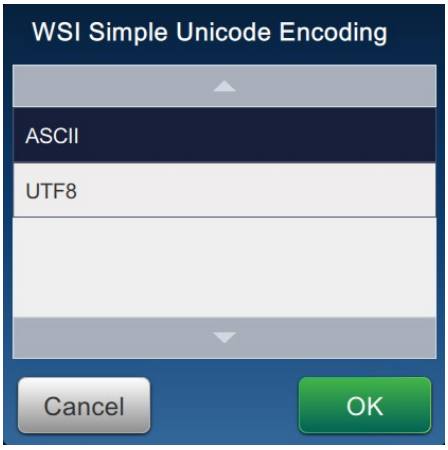


Figure 3-5: WSI Simple Unicode Encoding Screen

7. Touch the *WSI Remote Source Action* drop down list (see Figure 3-4) and select the required action for WSI Remote Data Command as shown in Figure 3-6. Touch the *OK* button to confirm the selection.

Figure 3-6: WSI Remote Source Action Screen

Action	Description
Repeat Last Value	The printer will continue to print the last data received.
Stop	The printer will drop from print, when the last data has been printed and the input buffer is empty.

Table 3-1: WSI Remote Source Action

3.1.1 Setup Communication with RS-232

Do the following tasks to setup the printer for WSI communication with RS-232 port:

1. Select any open COM port and click on *Configuration* tab from the Communications screen as shown in Figure 3-7.
2. Configure the COM connection parameters.

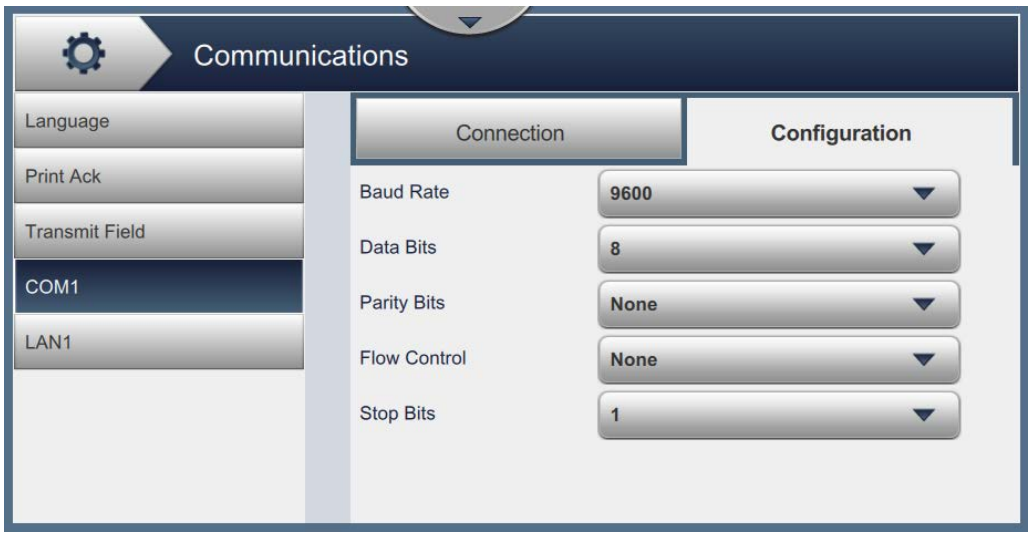


Figure 3-7: COM Settings Screen - Configuration Tab

3. Click on *Connection* tab from the Communications screen as shown in Figure 3-8.

Figure 3-8: COM Settings Screen - Connection Tab

4. Touch the *Protocol* drop down list. The Protocol screen opens as shown in Figure 3-9.

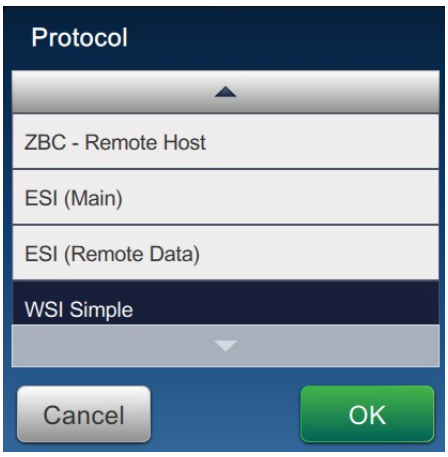


Figure 3-9: Protocol Screen

5. Select *WSI Simple* option from the Protocol screen as shown in Figure 3-9. Touch the *OK* button to confirm the selection. The *Under Remote Control* message in the banner will appear when protocol is selected.
6. The COM screen with updated parameters is shown in Figure 3-10.

Figure 3-10: COM Screen

3.1.2 Setup Communication with Network (Ethernet- TCP/IP)

Do the following tasks to setup the printer for WSI communication with Ethernet-TCP/IP port:

1. Touch *LAN1* option and click on *Configuration* tab from the Communications screen as shown in Figure 3-11.
2. Configure the LAN1 connection parameters.

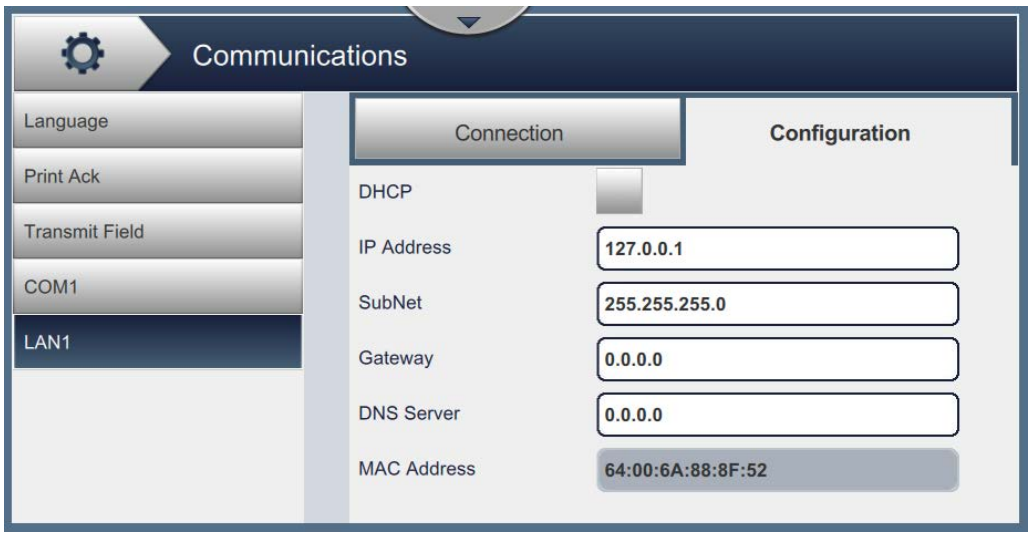


Figure 3-11: LAN1 Screen - Configuration Tab

3. Click on *Connection* tab from the Communications screen as shown in Figure 3-12.

Figure 3-12: LAN1 Screen - Connection Tab

4. Touch the *Add Protocol and Port* button to update the protocol and port number. The Add Protocol and Port screen appears as shown in Figure 3-13.

Figure 3-13: Add Protocol and Port Screen

5. Set the Port Number to 3100 if replacing 1000 Series printer.

   Note: *It is recommended to select the Port Number prior to Protocol.*

6. Touch the *Protocol* drop down list. The Protocol screen opens as shown in Figure 3-14.

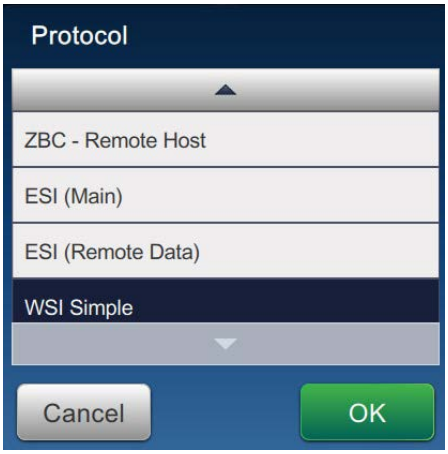


Figure 3-14: Protocol Screen

7. Select *WSI Simple* option from Protocol screen as shown in Figure 3-14. Touch the *OK* button to confirm the selection.

8. The Add Protocol and Port screen with updated parameters is shown in Figure 3-15. Touch the *Accept* button to save the changes. The *Under Remote Control* message in the banner will appear when the Protocol is selected and LAN connection is open.

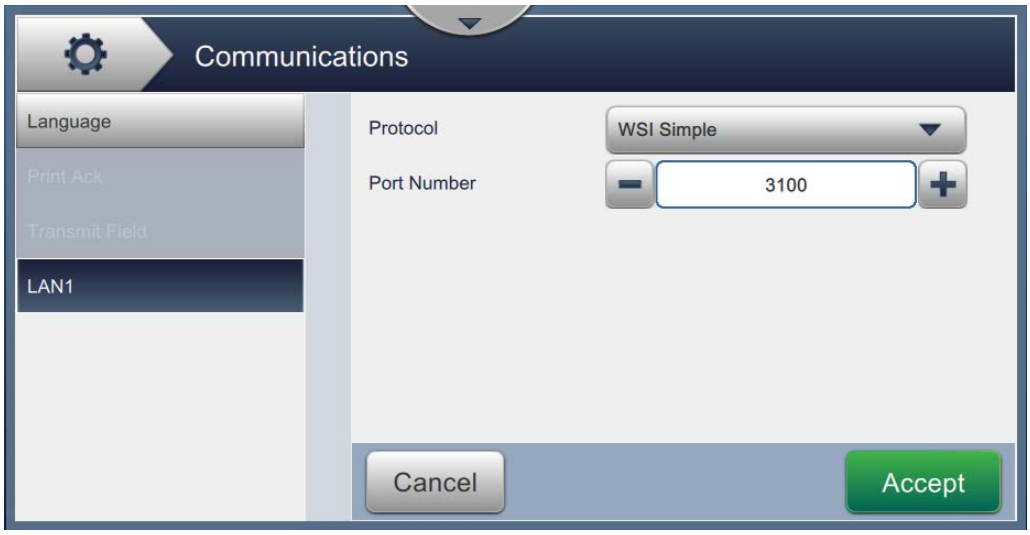


Figure 3-15: Add Protocol and Port Screen

9. The LAN1 screen with updated parameters is shown in Figure 3-16.

Figure 3-16: LAN1 Screen

4 Supported Packet Types

4.1 Supported Packet Types

The following packet types are supported:

Command Name	Description	Command	Sub Command
Remote Data	Supports remote data records same as in WSI-full	**A/a**	
Set Module Widths	Sets the module widths (bars and gaps) for printing barcodes	**B/b**	
Delete Job Text	Clears all existing text from the currently printing Job	**C/c**	
Clear User Field Data	Clears data from an existing named user field	**D/c**	
Error Status Request	Returns the errors currently active within the software	**E/e**	
Generic Get Command	Returns the current value of Print Counter	**G/g**	**A/a**
	Returns the current value of Product Counter		**B/b**
	Get Last Image Printed		**C/c**
	Get Next Image Printed		**D/d**
	Get Alarms and Warnings		**E/e**
	Get Date and Time		**F/f**
Request Part Number	Return the Software Part Number from the machine	**H/h**	
Start Jet	Operates the same as the printer Start Jet button	**J/j**	
Stop Jet	Operates the same as the printer Stop Jet button	**K/k**	
Update Logo Data	Update the bit pattern of an existing named logo	**L/l**	
Job Select	Selects a Job to print from the list of existing Jobs	**M/m**	
Print On/Off	Enables and Disables print (EHT)	**O/o**	
Request Current Selected Job	Returns the name of the current job	**Q/q**	

Command Name	Description	Command	Sub Command
Modify Job Parameters	Set the print parameters associated with the currently printing Job	**P/q**	
Generic Reset Command	Reset Print Counter	**R/r**	**A/a**
	Reset Product Counter		**B/b**
	Reserved		**C/c**
	Reserved		**D/d**
	Clear Alarms and Warnings		**E/e**
	Reserved		**F/f**
Update Job Text	Updates the Job content of the currently printing Job	**T/t**	
Update User Field Data	Update the data within an existing named user field	**U/u**	
Set Date and Time	Sets system date and time	**Z/z**	

Note: *Bold type characters denote the commands, which exist in the first release of 1000 series printers.*

5 General Protocol Packet Specification

5.1 Command Format

Each protocol packet sent to the printer must adhere to the following general format:

[STX] [TYPE] [DATA] [ETX]

[STX]	Is a fixed single ASCII character 02h which is used to indicate the start of a protocol packet.
[TYPE]	Is a single ASCII character in the range ‘A’ to ‘Z’ and 'a' to 'z' used to describe the type of protocol packet which is being sent.
[DATA]	Is a multiple character block. Two encoding formats will be supported, controlled by a UI toggle - ASCII or UTF-8. Within the DATA block, user may find SEP (0Ah) characters which serve as field separators for blocks comprising multiple fields. (Please refer “Example” on page 5-26.)
[ETX]	Is a fixed single ASCII or UTF-8 character 03h which is used to indicate the end of a protocol packet.

5.2 Response Format

In response, the printer will reply with a three character check sequence as follows:

$XX	$ indicates the command was executed successfully. XX is a pair of ASCII characters within the range ASCII „0' to „9' or „A‟ to „F‟. The character pair XX describes the modulo 256 sum of all characters between the [STX] and [ETX] fields, (not including those fields). The first character represents the high four bits of the checksum, and the second character the low four bits of the checksum.
!XX	! indicates the command failed or unknown command. XX is a pair of ASCII characters within the range ASCII ‘0' to ‘9' or ‘A’ to ‘F’. The character pair XX describes the modulo 256 sum of all characters (bytes) between the [STX] and [ETX] fields, (not including those fields). The first character represents the high four bits of the checksum, and the second character the low four bits of the checksum.

5.2.1 Checksum Calculation for Response

Checksum = (Sum of all bytes between the [STX] and [ETX]) % 256

Example

Job Selection - M/m Command (ASCII)

Job Name:	JOB1
Command:	02 4D 4A 4F 42 31 03
Checksum =	(SUM(4D 4A 4F 42 31)) % 100h = 59

Job Selection - M/m Command (UTF-8)

Job Name:	JOB2
Command:	02 4D CE 8F CE B0 C4 84 C5 85 C7 AC CE A6 CE B2 CE B4 03
Checksum =	(SUM(4D CE 8F CE B0 C4 84 C5 85 C7 AC CE A6 CE B2 CE B4)) % 100h = A3

5.2.2 Special response (Remote Logo Command)

#XX[CR] or !XX[CR]	Is used after the Remote Logo Data packet in case the Logo Data is not accepted. #XX[CR] is used if the logo data is not accepted because of the incoming buffer being full. !XX[CR] is used if the logo data is not accepted because the print is enabled and the logo dimensions don't match the dimensions of the buffers prepared to store the logo. In case of disabled print this logo should be accepted, and the buffers should be re-created to have the new logo dimensions. All the old size logos, which can be stored in the buffers to that moment should be discarded.
%XX[CR]	Is used if an attempt is made to reset the internal remote logo queue when the printer is in print mode. This response indicates that reset failed.

5.3 List of Commands

5.3.1 “Job Select” Packet Format

Description: Selects a Job to print from the list of existing Jobs stored within the machine. If the Job does not exist the currently printing Job will remain unchanged.

Format: **[STX] [TYPE] [JOB NAME] [ETX]**

- The TYPE field contains the single ASCII character ‘M’/‘m’.
- The JOB NAME field is a case insensitive string of between 1 and 30 ASCII or UTF-8 characters, without Job file extension describing an existing Job stored within the machine.

Response:

- Send Standard Success Response when successfully able to load job for printing
- Send Standard Failure Response when
  - Fails to load job
  - Job is not there on printer
  - Empty Job Name
  - Dependent file missing for job
  - Not able to clear the command queue
    - In this case, existing loaded job will not be modified.

Additional Information:

- This command does not support Unicode Job Names
- While loading the Job on the UI, it will clear all the Jobs from the Command Queue
- Job data will not be updated or changed while selecting the Job for printing using this command. The Job Data behavior is unknown and not specified

Example: Select Job Command Type “M”

- Create a job via the printer’s keyboard called “MSG1”
- The text in the job will be MSG1 - Save Job
- Create a job via the printer’s keyboard called “MSG2”
- The text in the job will be MSG2 - Save Job

WSI Command:

- Send: [02]MMSG1[03]
- Response: $[36][35]
- Printer home screen job changes to MSG1 with the text MSG1
- Send: [02]MMSG2[03]
- Response: $[36][36]
- Printer home screen job changes to MSG2 with the text MSG2

5.3.2 “Request Current Selected Job” Packet Format

Description: Returns the Current selected/loaded Job name for printing.

Format: **[STX] [TYPE] [ETX]**

- The TYPE field contains the single ASCII character ‘Q’/‘q’.

Response: The machine responds with the following packet:

[STX] [JOB NAME] [ETX]

- The JOB NAME field contains the name of the job selected in ASCII or UTF-8 format, without Job filename extension.
- Send Standard Failure Response when no Job Loaded on Printer.

Additional Information:

- This command does not support Unicode Job Names

5.3.3 “Delete Job Text” Packet Format

Description: Clears all existing text from the currently printing Job, leaving the Job content blank, but the Job parameters unchanged. This command does not delete the Job.

Format: **[STX] [TYPE] [ETX]**

- The TYPE field contains the single ASCII character ‘C’/‘c’.

Response:

- Send Standard Success Response when successfully able to delete all fields from Job
- Send Standard Failure Response when
  - Not able to delete any fields from Job
  - No Job loaded on Printer
  - Not able to clear the Command queue
    - In this case, Existing loaded job will not be modified

Additional Information:

- This command clears all the Jobs from the Command Queue before deleting all fields from Job

5.3.4 “Update Job Text” Packet Format

Description: Updates the Job content of the currently printing Job in line with the new data supplied within the data field of the packet. This command deletes the current Job content before the new text is inserted. The command is processed in synchronous mode – this means that the packet will not be Acknowledged until processed completely.

Format: **[STX] [TYPE] [JOB DATA 1] [SEP]... [JOB DATA n] [ETX]**

- The TYPE field contains the single ASCII character ‘T’/'t'.
- The JOB DATA fields contain several sub fields as described below. There may be several JOB DATA fields within a single Job, each separated by a SEP (0Ah) field.

5.3.4.1 JOB DATA Field Format

Each JOB DATA field is of the following format:

[FONT NUM] [HORC] [VERC] [ATTRIB] [JOB TEXT]

- The FONT NUM field consists of exactly 2 characters (ASCII ‘0' to ‘9') which represent a decimal value between 00 and 99, it describes the font number to be used for the following text.
  - The font number follows the order in which the fonts appear when using the editor, i.e. font 0 is always 7hi, but the rest will be dynamically allocated according to the fonts fitted. (00 = 7hi, 01 = 9hi, 02 = 12hi, 03 = 16hi, 04 = 24hi, 05 = 34hi, 06 = 5hi).
  - Please refer “Font Mapping” on page A-1 for more details.
- The HORC field consists of exactly 4 characters (ASCII ‘0' to ‘9') which represent a decimal value between 0 and 9999, it describes the relative horizontal position at which the text fragment should be placed within the Job.
  Note: *This position is relative, and subject to left justification.*
  It should be considered as a fragment ordering control rather than a means of placing the text fragment in an exact location. i.e. the text fragment with a HORC of 1 will always be to the left of another text fragment with a HORC of greater than 1, etc.
- The VERC field consists of exactly 3 characters (ASCII ‘0' to ‘9') which represent a decimal value between 0 and 999, it describes the vertical position at which the text field should be placed within the Job.
  Note: *The whole Job is subject to upward justification.*
  If blank lines are required at the top of a Job a dummy fragment containing a single space character in an appropriate font must be sent. The algorithm of field positioning on the basis of HORC and VERC is as follows:
  1. Select all fields with minimal value of HORC.
  2. Place them to the Job image in a left aligned column from top to bottom in the order of VERC values with 1 pixel gap.
  3. Select all fields with second minimal value of HORC and place them in same column to the right of the fields with minimal HORC.

4 Shift each of the fields of the second column horizontally to the left until there is a minimal gap of 1 pixel with any of the fields of the first column or will come to the left boundary of the Job image.

5 Continue selecting fields with next value of HORC, piling them in a column and shifting left where possible until all the fields will take their place.

- The ATTRIB field consists of exactly 6 characters (ASCII ‘0' to ‘9' or ‘A’ to ‘F’). The first four characters represent a hexadecimal value between 0 and FFFF, it describes a bit mask which is used to apply various attributes to the Job text field. The sixth character represents a hexadecimal value between 0 and F, it describes the code page number to be used. The code page number, whose value in base 16 ranges from 0 to F, is encoded as a single ASCII character (of range‘0’-‘9’,’A’-‘F’).

  The bits in each attribute are set according to whether the corresponding properties are set, as follows:

  ATTRIBUTE 1

Bits	Description
Bit 0	Invert
Bit 1	Reverse
Bit 2	Clockcode. Indicates that this job data field is to present Date/Time field.
Bit 3	Double dots

  ATTRIBUTE 2

Bits	Description
Bit 0	Triple dots
Bit 1	Tower printing
Bit 2	Prompted user field. Indicates that this job data is to present a user prompted unnamed text field. This bit is ignored if Attribute 1 bit 2 Clockcode is set.
Bit 3	Barcode HR on

  ATTRIBUTE 3

Bits	Description
Bit 0	Use custom font
Bit 1	Invert video
Bit 2	Barcode checksum on
Bit 3	Logo

ATTRIBUTE 4 & 5 (Barcode Type)

Bits	Description
0	None
1	ITF
2	EAN-8
3	EAN-13
4	UPC-A
5	UPC-E
6	Code 128
7	EAN-128
8	Code 39
9	Datamatrix 10x10
10	Datamatrix 12x12
11	Datamatrix 14x14
12	Datamatrix 16x16
13	Datamatrix 18x18
14	Datamatrix 20x20
15	Datamatrix 22x22
16	Datamatrix 24x24
17	Datamatrix 26x26
18	Datamatrix 32x32
19	Datamatrix 8x18
20	Datamatrix 8x32
21	Datamatrix 12x26
22	Datamatrix 12x36
23	Datamatrix 16x36
24	Datamatrix 16x48
25	Reserved
26	QRCode with Low ECLevel
27	QRCode with Medium ECLevel
28	QRCode with Quartil ECLevel
29	QRCode with High ECLevel

ATTRIBUTE 6 (Code Page Number)

If 6 byte attribute is not used, it will send all 0.

- The Job TEXT field consists of up to 200 ASCII or UTF-8 characters which describe the text to be entered within the Job, or (if it starts with the character “@”) the name (case insensitive) of a single user field. This arrangement allows plain text, barcodes, counters, or other real-time data to be inserted within the Job. The print orientation and presentation is controlled via the attribute field.

Response:

- Send Standard Success Response when successfully able to add fields to currently loaded job
- Send Standard Failure Response when
  - No Job loaded for Printing
  - Total command size is less than 16 bytes or invalid command packet
    - In this case, Existing loaded job will not be modified
  - Not able to clear the Command queue
    - In this case, Existing loaded job will not be modified
  - During the fields update, any field deletion fails
    - In this case, all fields will be cleared from loaded job
  - Barcode generation fails
    - In this case, all fields will be cleared from loaded job
- Please refer “Fields Addition and Validation Rules” on page A-23 for more details.

Additional Information:

- This command clears all the Jobs from Command Queue before updating Job with new fields
- During the fields update, few field will be ignored based upon the improper data and continue with other field adding to the job. Please refer “Fields Addition and Validation Rules” on page A-23 for more details.
- Field addition will be ignored if Position is invalid with respect to the selected font for that field
- Please refer “T Command Structure and Parameters Mapping” on page A-3 for more details.
- Please refer “T Command Examples” on page A-5 for more details.
- Please refer “Date Format” on page A-19, “Time Format” and “Date and Time Format - Separator” on page A-22 for details on supported format for Date and Time Fields.
- Please refer “Field Positioning” on page A-27 for details on rules for field positioning and example for various field positioning.

5.3.5 “Modify Job Parameters” Packet Format

Description: Updates the print parameters associated with the currently printing Job in line with the new parameter list supplied within the data field of the packet.

Format: **[STX] [TYPE] [REV] [INV] [WID] [EHT] [GAP] [EXP] [HEJRA][DLY] [BLD] [DRP] [RASSUB][RLEN][RAS][RDLYLEN][RDLY][ETX]**

- The TYPE field contains the single ASCII character ‘P’/‘p’.
- The REV field consists of a single ASCII character where ‘0' represents OFF and ‘1' represents ON, it describes the Job parameter “Job reverse”.
- The INV field consists of a single ASCII character where ‘0' represents OFF and ‘1' represents ON, it describes the Job parameter “Job invert”.
- The WID field consists of exactly 4 characters (ASCII ‘0' to ‘9') which represent a decimal value ranged as indicated in table, it describes the Job parameter “width”.
- The EHT field consists of exactly 2 characters (ASCII ‘0' to ‘9') which represent a decimal value between 1 and 10, it describes the Job parameter “Job height”.
- The GAP field consists of a single character (ASCII ‘0' to ‘9') which represents a decimal value between 0 and 9, it describes the Job parameter “inter-character gap”.
- The EXP field consists of exactly 5 characters (ASCII ‘0' to ‘9') which represent a decimal value between 0 and 32767, it describes the Job parameter “expiry date - days ahead counter”.
- The HEJRA field consists of exactly 5 characters (ASCII ‘0' to ‘9') which represent a decimal value between 0 and 32767, it describes the Job parameter “Hejra expiry days ahead”.
- The DLY field consists of exactly 5 characters (ASCII ‘0' to ‘9') which represent a decimal value ranged as indicated in table, it describes the Job parameter “product delay”.
- The BLD field consists of exactly 2 characters (ASCII ‘0' to ‘9') which represent a decimal value between 1 and 10, it describes the Job parameter “raster repeat”.
- The DRP field consists of exactly 2 characters (ASCII ‘0' to ‘9') which represent a decimal value between 1 and 34, it describes the Job parameter “printed dots”.
- The RASSUB field consists of a single ASCII character where ‘0' represents OFF and ‘1' represents ON, it describes the Job parameter “Raster Substitution”.
- The RLEN field consists of exactly 3 characters (ASCII ‘0' to ‘9') which represent a decimal value between 1 and 255, it describes the length of the raster filename that is to follow.
- The RAS field consists of ASCII characters as per the previous RLEN field, it is the case insensitive file name of the raster - Job parameter “Raster”.

- The RDLYLEN field consists exactly 1 (ASCII ‘0' to ‘9') character which represents the Length of the Reverse Print Delay Value.
- The RDLY field consists of ASCII characters as per the previous RDLYLEN field, RDLY represents a decimal value ranged as indicated in table, it describes the job parameter “reverse product delay”.

Please refer Section 6.2.2 “Modify Job Parameters” on page 6-2 for not supported parameters.

Response:

- Send Standard Success Response when successfully able to modified Job Parameters
- Send Standard Failure Response when
  - No Job Loaded on Printer
  - Command received with empty data
  - Command received with incomplete data i.e. command data size is less than or equal to 32
  - Command received with invalid data (Ex. Alpha instead of Numeric)
  - Load Job fails after parameter update

Additional Information:

Please refer “Raster Substitution and Raster Name” on page A-38 for Raster update behavior.

Width Parameter

- Unit of this field is depending on the “Display Unit of measure” & “Job Configuration” parameters in printer.
- In case measuring unit is selected as Inches, unit of this field will be 1/100th of inch e.g 1.25” should be sent as 125.
- In case of Metric it will be in mm.

Metric (mm)		1/100 Inches	
Divider / Strokes	Linear Units	Divider / Strokes	Linear Units
1-255	1-999	1-255	1-3936 (1=0.01”)

Delay Parameter

- Unit of this field is depending on the “Display Unit of measure” & “Job Configuration” parameters in printer.
- In case measuring unit is selected as Inches, unit of this field will be 1/100th of inch e.g 1.25” should be sent as 125.
- In case of Metric it will be in mm.

Metric (mm)		1/100 Inches	
Divider / Strokes	Linear Units	Divider / Strokes	Linear Units
0-9999	0-999	0-9999	0-3936 (1 = 0.01")

Reverse Delay Parameter

- Unit of this field is depending on the “Display Unit of measure” & “Job Configuration” parameters in printer.
- In case measuring unit is selected as Inches, unit of this field will be 1/100th of inch e.g 1.25” should be sent as 125.
- In case of Metric it will be in mm.

Metric (mm)		1/100 Inches	
Divider / Strokes	Linear Units	Divider / Strokes	Linear Units
0-9999	0-999	0-9999	0-3936 (1 = 0.01")

5.3.6 “Clear User Field Data” Packet Format

Description: Clears data from an existing named user field, leaving the user field blank. This command does not delete the user field.

Format: **[STX] [TYPE] [USER FIELD NAME] [ETX]**

- The TYPE field contains the single ASCII character ‘D’.
- The USER FIELD NAME field is a case sensitive string of between 1 and 30 ASCII or UTF-8 characters describing an existing user field stored within the machine.

Response:

- Send Standard Success Response when successfully able to clear first text or user prompted text field in custom reference
- Send Standard Failure Response when
  - Custom reference is not available on the Printer
  - First field is other than text and User Prompted Text field

Additional Information:

- This command only clears Text and User Prompted Text fields in Custom reference

5.3.7 “Update User Field Data” Packet Format

Description: Update the data within an existing named user field in line with the new data supplied within the data field of the packet. This command deletes the current user field content before the new text is inserted.

User fields of type: text, serializer and date/time can be configured using this job.

Format: **[STX] [TYPE] [USER FIELD NAME] [SEP] [USER FIELD DATA] [ETX]**

or

[STX] [TYPE] [USER FIELD NAME] [ETX]

- The TYPE field contains the single ASCII character ‘U’.
- The USER FIELD NAME field is a case sensitive string of between 1 and 30 ASCII or UTF-8 characters describing an existing user field stored within the machine.
- The SEP field, if present, may only contain the single ASCII characters LF (0Ah), and is used to separate data fields within a user field data packet.
- The USER FIELD DATA field, if present, is a string of between 1 and 50 ASCII or UTF-8 characters describing the new value to be placed within the named user field.

Serializer/Counter USER FIELD DATA:

[START value]	ASCII [‘0’-‘9’,’A’-‘Z’,’a’-‘z’]
[SEP]	0Ah
[CURRENT value]	ASCII [‘0’-‘9’,’A’-‘Z’,’a’-‘z’]
[SEP]	0Ah
[END value]	ASCII [‘0’-‘9’,’A’-‘Z’,’a’-‘z’]
[SEP]	0Ah
[STEP value]	ASCII [‘0’-’9’]
[SEP]	0Ah
[DIRECTION value]	ASCII [‘0’- decrement, ’1’ – increment]
[SEP]	0Ah
[REPEAT value]	ASCII [‘0’-’9’]
[SEP]	0Ah
[PAD CHARACTER]	ASCII [‘0’-‘9’,’A’-‘Z’,‘a’-‘z’] or SPACE or EMPTY
[SEP]	0Ah

Note: *For incrementing counters, Start < End*
For decrementing counters, Start > End

Text USER FIELD DATA:

- Text field information is sent verbatim, without markup or padding in either ASCII or UTF-8 encoding.

Date/Time USER FIELD DATA:

Type of offset	1 ASCII ['N' - DT_OFFSET_NONE 'D' - DT_OFFSET_DAY 'W' - DT_OFFSET_WEEK 'M' - DT_OFFSET_MONTH 'Y' - DT_OFFSET_YEAR]
Offset value	3 ASCII digits [000 - 999]
Clock format string	46 ASCII characters

If SEP and USER FIELD DATA are not present:

- The printer will respond with the current value of the named User Field.
- The format of the current value will mirror the formats described above.

Response:

- Send Standard Success Response when successfully able to update first field in custom reference
- Send Standard Failure Response when
  - User field (Custom Reference) is not available on the Printer
  - User field (Custom Reference) Name is greater than 30
  - User field (Custom Reference) Data is greater than 50
  - User field (Custom Reference) name is empty
  - User field (Custom Reference) first field type is other than Text, User Prompted Text, User Prompted Counter and Offset Date
  - For Counter Insert,
    - End value is greater than Start value and direction is increment
    - Start value is greater than End value and direction is decrement
    - Start value, Current value and End value length is greater than 6 for Alpha and 9 for Numeric.
    - Repeat value is less than zero
    - Direction value is other than 0 or 1
    - Pad character length is more than one.
    - Invalid values for counter type (Ex. try to update numeric value in Alpha Counter)
    - Step size is not a factor of counter range

Additional Information:

- To match the behavior with 1000L Printer
  - When counter/serializer data is received and the custom reference contains the first field as Text, then the start value will be updated in the text field in string format
    - No other parameters (ex. padding) will be updated in Text or User Prompted Text
  - When Date/Time data is received and the custom reference contains the first field as Text, then the entire date and time will be updated in the text field in string format.
  - User Prompted Text will not be updated by this command. If the custom reference contains first field as User Prompted Text, then it will not be updated and will send standard success response.
- This command currently supports the update Text and User Prompted Counter inserts in Custom Reference only. Normal Counter will not be updated and will send standard failure response.

5.3.8 “Update Logo Data” Packet Format

Description: Updates the bit pattern of an existing named logo in line with the new data supplied within the data field of the packet. This command deletes the current logo content before the new bit pattern is inserted.

Format: **[STX] [TYPE] [LOGO NAME] [SEP] [NUM DROPS] [NUM RASTERS] [LOGO DATA] [ETX]**

- The TYPE field contains the single ASCII character ‘L’.
- The LOGO NAME field is a case insensitive string maximum of 30 ASCII or UTF-8 characters describing an existing user field stored within the machine.
- The SEP field may only contain the single ASCII characters LF (0Ah), and is used to separate data fields within a logo data packet.
- The NUM DROPS field consists of exactly 2 characters (ASCII ‘0' to ‘9') which represent a decimal value between 5 and 26, it describes the number of drops per raster in the logo data which follows. The data is right-justified and zero-padded.
- The NUM RASTERS field consists of exactly 3 characters (ASCII ‘0' to ‘9') which represent a decimal value between 1 and 255, it describes the number of rasters in the logo data which follows. The data is right-justified and zero-padded.
- The LOGO DATA field is a list of pairs of ASCII characters (‘0' to ‘9' and ‘A’ to ‘F’) describing the new logo bit pattern to be placed within the named logo. Each pair of ASCII characters are encoded as shown below:
  - LOGO DATA is encoded as a list of pairs of ASCII characters, whereupon each pair represent 8-bits of image data.

Response:

- Send Standard Success Response when successfully able to update requested logo's bitmap file.
- Send Standard Failure Response when
  - Logo bitmap file does not exist on the Printer
  - Logo bitmap file name is greater than 30 characters
  - NUM DROPS is not between 5 and 34.
  - NUM RASTERS is not between 1 and 255
  - LOGO DATA size is not sufficient to draw the LOGO of the size specified
  - Unable to update the LOGO bitmap file due to R/W issue

Additional Information:

The process in which the graphic is converted into ASCII characters is not trivial and is as follows:

- The image is broken down into rasters NUM DROPS high. There are NUM RASTERS quantity of rasters.
- A raster is depicted as a row of ink-drop data, in which each drop position may or may not be “inked”, and each drop is assigned a zero-based drop number. The drop number ascends from the top of the image – drop number 0 being at the top.
- To encode each raster a bit array is created whose size is either equal to NUM DROPS if NUM DROPS is multiple of 8, or the next highest multiple of 8 if not.
- The state of each drop, whether it is inked or not, in the raster results in the corresponding bit in the array being set or cleared, respectively.
- Each bit is set/cleared from the lowest bit, corresponding to the topmost drop. Where the array size is not equal to NUM DROPS, the bit position in the array populated is equal to the drop number offset by the addition of the constant array size – NUM DROPS. Any unused bits should be left clear.
- As each raster is coded into a bit array, starting at the least significant bit, each group of 4-bits is interpreted as a base-16 number (0-F), and this number is converted into the corresponding ASCII character (in range ‘0'-‘9',‘A’-‘F’).
- Each ASCII character as it is produced is appended onto the LOGO DATA buffer.
- The process is repeated for each raster from left to right, until there are no more rasters.

Example logo and encoding:

	Byte Offset												
		00	**02**	**04**	**06**	**08**	**10**	**12**	**14**	**16**	**18**	**20**	**22**	**24**
Byte n	**Bit 7**													
	Bit 6			0	0	0	0	0	0	0	0			
	Bit 5		0	0	0	0	0	0	0	0	0	0		
	Bit 4	0	0	0							0	0	0	
	Bit 3	0	0								0	0	0	
	Bit 2	0	0							0	0	0	0	
	Bit 1	0	0						0	0	0	0	0	
	Bit 0	0	0					0	0	0		0	0	
Byte n+1	**Bit 7**	0	0				0	0	0			0	0	
	Bit 6	0	0			0	0	0				0	0	
	Bit 5	0	0		0	0	0					0	0	
	Bit 4	0	0	0	0	0						0	0	
	Bit 3	0	0	0	0							0	0	
	Bit 2	0	0	0							0	0	0	
	Bit 1		0	0	0	0	0	0	0	0	0	0		
	Bit 0			0	0	0	0	0	0	0	0			

The above logo would be represented by the following data:

0x1F, 0xFC, 0x3F, 0xFE, 0x70, 01F, 0x60, 0x3B, 0x60, 0x73, 0x60, 0xE3, 0x61, 0xC3, 0x63, 0x83, 0x67, 0x03, 0x7E, 0x07, 0x3F, 0xFE, 0x1F, 0xFC, 0x00, 0x00

It would be transmitted to the machine using the following packet:

<STX>L16 High Zero Logo<LF>160131FFC3FFE701F603B607360E
361C3638367037E073FFE1FFC0000<ETX>

5.3.9 “Set Module Widths” Packet Format

Description: Sets the module widths (bars and gaps) for printing barcodes, to ensure that barcodes are readable regardless of the ink spread on the substrate.

Format: **[STX] [TYPE] [BW1] [BW2] [BW3] [BW4] [GW1] [GW2] [GW3] [GW4][ETX]**

- The TYPE field contains the single ASCII character ‘B’.
- The BW1 field consists of exactly 2 characters (ASCII ‘0' to ‘9') which represent a decimal value between 1 and 12, it describes the number of rasters in the narrowest bar of the printed raster.
- The BW2 field consists of exactly 2 characters (ASCII ‘0' to ‘9') which represent a decimal value between 1 and 12, it describes the number of rasters in the next narrowest bar of the printed raster.
- The BW3 field consists of exactly 2 characters (ASCII ‘0' to ‘9') which represent a decimal value between 1 and 12, it describes the number of rasters in the next to widest bar of the printed raster.
- The BW4 field consists of exactly 2 characters (ASCII ‘0' to ‘9') which represent a decimal value between 1 and 12, it describes the number of rasters in the widest bar of the printed raster.
- The BG1 field consists of exactly 2 characters (ASCII ‘0' to ‘9') which represent a decimal value between 1 and 12, it describes the number of rasters in the narrowest gap of the printed raster.
- The BG2 field consists of exactly 2 characters (ASCII ‘0' to ‘9') which represent a decimal value between 1 and 12, it describes the number of rasters in the next narrowest gap of the printed raster.
- The BG3 field consists of exactly 2 characters (ASCII ‘0' to ‘9') which represent a decimal value between 1 and 12, it describes the number of rasters in the next to widest gap of the printed raster.
- The BG4 field consists of exactly 2 characters (ASCII ‘0' to ‘9') which represent a decimal value between 1 and 12, it describes the number of rasters in the widest gap of the printed raster.

Response:
- Send Standard Success Response when successfully able to Set Module Widths
- Send Standard Failure Response when
  - Invalid command size
  - Parameters is not digits

Additional Information:
- If Barcode Width and Barcode Gap values are outside of range, it will set to minimum or maximum based on the value received.

5.3.10 “Start Jet” Packet Format

Description: Starts the jet as if “Start Jet” button has been pressed on the touchscreen.

Format: **[STX] [TYPE] [ETX]**

- The TYPE field contains the single ASCII character ‘J’/‘j’.

Response:
- Send Standard Success Response when
  - Jet is in SHUTDOWN state
- Send Standard Failure Response when
  - Jet is in STARTING UP, OFFLINE, RUNNING and SHUTTING DOWN state
  - Unable to start jet for some reason (Ex. Printer is in fault state)

5.3.11 “Stop Jet” Packet Format

Description: Stops the jet as if “Stop Jet” button has been pressed on the touchscreen.

Format: **[STX] [TYPE] [ETX]**

- The TYPE field contains the single ASCII character ‘K’/‘k’.

Response:
- Send Standard Success Response when
  - Jet is in OFFLINE and RUNNING state
- Send Standard Failure Response when
  - Jet is in STARTING UP, SHUTDOWN and SHUTTING DOWN state

5.3.12 “Request Part Number” Packet Format

Description: Returns the “Software Part Number” string stored within the machine.

Format: **[STX] [TYPE] [ETX]**

- The TYPE field contains the single ASCII character ‘H’/‘h’.

Response: The machine responds with the following packet:

[STX] [PART NUMBER] [ETX]

- The PART NUMBER field contains the software version of Printer which contains exactly 16 ASCII characters. If part number length is less than 16 then it will add space at the end.

Additional Information: **Example:**

- Printer software version is 0.211.41437
  - Response in ASCII - 02 30 2E 32 31 31 2E 34 31 34 33 37 20 20 20 20 20 03

5.3.13 “Error Status Request” Packet Format

Description: Returns the Error Status Word and the Alarm Status Nibble.

Format: **[STX] [TYPE] [ETX]**

- The TYPE field contains the single ASCII character ‘E’.

Response: The machine responds with the following packet:

[STX] [EEEEEE][A] [ETX]

- The EEEEEE field contains the 6 byte hexadecimal error status as follows:

Byte	Bit 0	Bit 1	Bit 2	Bit 3
0	Charge Error	EHT Trip	Gutter Fault	Mixer Empty
1	Pump Fault	Elect Too Hot	Ink Service Over Due	No Viscosity Control
2	Bad Nozzle	Moddriver Over temp	No Phase Data	Mod Readback Error
3	Raster Memory Overflow	Valve Error	Sys Fill Failed	Sys Fill Again
4	RTC Invalid	Ink Core Change	No Inksys	EHT Not Calibrated
5	Reserved	Reserved	Reserved	Reserved

- The A field contains the 4 bit hexadecimal alarm status word as follows:

Byte	Bit 0	Bit 1	Bit 2	Bit 3
6	Green Trafficator	Amber Trafficator	Red Trafficator	Alarm Activated - NOT USED

5.3.14 “Clear Alarms and Warnings” Packet Format

Description: Clears any active Alarms and Warnings which are manually clearable.

Format: **[STX] [TYPE] [DATA] [ETX]**

- The TYPE is command ‘R’- Generic Reset Command.
- The DATA is command ‘E’/‘e’ - Clear Alarms and Warning

Response:

- Send Standard Success Response
  - When successfully able to Clear all Alarms and Warning
  - When there is no alarm and warning on Printer
- Send Standard Failure Response
  - When not able to clear any alarms or warning which is not clearable

5.3.15 “Get Alarms and Warnings” Packet Format

Description: Get alarms and warning currently raised on printer. This command returns comma separated EID of alarms and warning.

Format: **[STX] [TYPE] [DATA] [ETX]**

- The TYPE is command ‘G’- Generic Get Command.
- The DATA is command ‘E’/‘e’ - Get Alarms and Warning.

Response: The machine responds with the following Packet:

[STX] XXXX,XXXX [0A] XXXX,XXXX [ETX]

- Where ‘X’ represents ASCII characters within the range ASCII ‘0' to ‘9'. XXXX represents the EID of alarms and warning.

Example:

Printer has the No Line Selected (E1012) fault and External Job Selection Mode (E2023), Invalid Job File (E1214) warnings then below response will be sent:

[02] [31] [30] [31] [32] [0A] [32] [30] [32] [33] [2C] [31] [32] [31] [34] [03]

Printer does not have any fault or warning the below response will be sent:

[02] [0A] [03]

5.3.16 “Print / Product Counter Request” Packet Format

Description: Returns the Print or Product Counter value depending on the data sent with the command. To request the Print Counter value data ‘A’/’a’ is used, and to get the Product Counter data ‘B’/’b’ is used.

Format: **[STX] [TYPE] [DATA] [ETX]**

- The TYPE is command ‘G’- for requesting of counters.
- The DATA command can be ‘A’/’a’ or ‘B’/’b’ depending on the Print and Product Counter.

Response: The machine responds with the following Packet:

[STX] [X X X X X X X X X X] [ETX]

- Where ‘X’ represents ASCII characters within the range ASCII ‘0' to ‘9'.The X is the Value of the Requested Counter represented in ten individual digits.

5.3.17 “Print / Product Counter Reset” Packet Format

Description: This command resets the Print or Product Counter depending on the data sent with the command. To reset the Print Counter value data ‘A’/’a’ is used, and to reset the Product Counter data ‘B’/’b’ is used.

Note: *This is the Print or Product Counter only and is not Batch Print or Batch Product Count, Production Print or Production Product Count.*

Format: **[STX] [TYPE] [DATA] [ETX]**

- The TYPE is command ‘R’- for resetting of the counters.
- The DATA command can be ‘A’/’a’ or ‘B’/’b’, depending on the counter requested.
  - A/a - Reset Print Counter
  - B/b - Reset Product Counter

Response:

- Send Standard Success Response when successfully able to reset the Print and Product Counters.
- Send Standard Failure Response on fail to do so.

5.3.18 “Set Date and Time” Packet Format

Description: Sets the specified Date and Time for the printer.

Format: **[STX] [TYPE] [YY] [MM] [DD] [hh] [mm] [ss] [ETX]**

- The TYPE field contains the single ASCII character ‘Z’ / ‘z’.
- The YY field consists of exactly 2 characters (ASCII ‘0' to ‘9') which represent a decimal value between 06 and 99 (2006 and 2099), it describes the last two digits of the year.
- The MM field consists of exactly 2 characters each (ASCII ‘0' to ‘9') which represent a decimal value between 1 and 12, it describes month in digits.
- The DD field consists of exactly 2 characters each (ASCII ‘0' to ‘9') which represent a decimal value between 1 and 31, it describes days in digits.
- The hh field consists of exactly 2 characters each (ASCII ‘0' to ‘9') which represent a decimal value between 0 and 23, it describes hours in digits.
- The mm field consists of exactly 2 characters each (ASCII ‘0' to ‘9') which represent a decimal value between 0 and 59, it describes minutes in digits.
- The ss field consists of exactly 2 characters each (ASCII ‘0' to ‘9') which represent a decimal value between 0 and 59, it describes seconds in digits.

Response:

- Send Standard Success Response when successfully able to Set Date and Time of Printer
- Send Standard Failure Response when
  - Date and Time Data is invalid

5.3.19 “Get Date and Time” Packet Format

Description: Get the Date and Time of the printer.

Format: **[STX] [TYPE] [DATA] [ETX]**

- The TYPE field contains the single ASCII character ‘G’ / ‘g’.
- The DATA command is F/f to get date and time

Response: The machine responds with the following Packet:

[STX] [DATA] [ETX]

- DATA = YYYY-MM-DD hh:mm:ss
  - YYYY = 4 digit year
  - MM = 2 digit month
  - DD = 2 digit day of month
  - hh = 2 digit hour (24 hr format)
  - mm = 2 digit minute
  - ss = 2 digit second

Example:

Send: [STX]GF[ETX]

Receive: [STX]2012-07-25 17:09:20[ETX]

5.3.20 “Print On/Off” Packet Format

Description: Turns ON or OFF printing.

Format: **[STX] [TYPE] [Y] [ETX]**

- The TYPE field contains the single ASCII character ‘O’ / ‘o’.
- Y refers to either of the following:
  - ‘0’ disables print mode for head on which command is received.
  - ‘1’ enables print mode for head on which command is received.

Response:

- Send Standard Success Response when
  - Successfully able to ON/OFF Printing
  - Send Print ON in RUNNING state
  - Send Print OFF in OFFILINE, SHUTDOWN, STARTING UP and SHUTTING DOWN State
- Send Standard Failure Response when
  - Not able to ON/OFF Printing
  - Send Print ON in SHUTDOWN, STARTING UP and SHUTTING DOWN states

5.3.21 “Request Current Selected Job” Packet Format

Description: Returns the Current selected Job Name for printing.

Format: **[STX] [TYPE] [ETX]**

- The TYPE field contains the single ASCII character ‘Q’ / ‘q’.

Response: The machine responds with the following Packet:

[STX] [JOB NAME] [ETX]

- The JOB NAME field contains the name of the job selected in ASCII or UTF-8 format, without Job filename extension.
- Send Standard Failure Response when no Job Loaded on Printer.

5.3.22 “Remote Data” Packet Format

Description: Each printed job can contain inserts, the content of which comes from remote data. One of these “remote data” packets specifies the data for up to 10 inserts for one print.

Up to 200 of these sets of remote data are buffered by the printer.

Format: **[STX] [TYPE] [USER FIELD 1 DATA] [SEP] [USER FIELD 2 DATA] [SEP]... [USER FIELD 10 DATA] [ETX]**

- The TYPE field contains the single ASCII character 'A'/'a'.
- The USER FIELD X DATA field is a string of between 1 and 50 ASCII or UTF-8 characters describing the new text to be placed within remote data user field 'x'.
- The SEP field may only contain the single ASCII character LF (0Ah), and is used to separate data fields within a user field data packet.
- The appearance of ASCII CAN symbol (18h) should cause the buffer (up to 200 entries) to be wiped at the moment the character is processed. The CAN symbol may be preceded or followed by regular user field data or separator. Preceding data will be wiped, following data will be processed as normal and form the first record in the buffer.
- Missing Remote Data in the queue on print trigger should cause either Repeat Last Value or Stop Print depending on the UI setting (Please refer Section 3.1 “Setup WSI Communication Protocol” on page 3-1).

Response:

- Send Standard Success Response when successfully able to add remote data to queue
- Send Standard Failure Response when
  - Not able to add remote data in queue
  - No Job Loaded
  - Not able to clear the queue when requested using CAN symbol

Additional Information:

- For Custom Reference Field, Only first insert will be updated by this command if it is user prompted.

Example

For the printer to use the remote data features of WSI the job will need to be created with user prompted text fields in it for all the remote data fields.

In the following example, REMOTE #1, REMOTE #2, REMOTE #3 are all created within the job as text user prompted fields.

[02]A REMOTE #1[0A] REMOTE #2[0A] REMOTE #3[03]

[02]	
A	Type
REMOTE #1	Remote Data Field 1
[0A]	Seperator
REMOTE #2	Remote Data Field 2
[0A]	Seperator
REMOTE #3	Remote Data Field 3
[03]	

Figure 5-1: Job Preview - Remote Data Example

5.3.23 “Get Last Job Printed” Packet Format

Description: Get Last Job Printed

Format: **[STX] [TYPE] [DATA] [ETX]**

- The TYPE field contains the single ASCII character ‘G’ / ‘g’.
- The DATA command is ‘C’/‘c’ to get last job printed

Response: The machine responds with the following Packet:

[STX] [DATA] [ETX]

- DATA is the content of each field concatenated in alphabetical order of the field name. And when line designator inserted as first character in the field name then each line will be delimited with a line feed character.
  - Line designator can be alphanumeric i.e. 1,2,3,a,b,c,A,B,C, etc.
  - Alphabetical order of field name follows digit, Upper Case, Lower case and UTF-8 characters.
- Supports both ASCII and UTF-8

Additional Information:

- All field values will send in string format
- For Logo field, Logo name will send in string format
- For Custom Reference, value of each insert will send in text format
- If Job is empty and no job loaded, empty data will send
- Please refer “Additional Examples” on page A-38.

Example

Job Data	Command Response
Field001=Text|Field002=13/05/2022| Field003=17:30:16PM|Field004=0899| Field005=123456789| Field006=2D-Barcode|Field007=VJ	[STX] Text13/05/202217:30:16PM 0899123456789 2D-BarcodeVJ [ETX] 02 54 65 78 74 31 33 2F 30 35 2F 32 30 32 32 31 37 3A 33 30 3A 31 36 50 4D 30 38 39 39 31 32 33 34 35 36 37 38 39 32 44 2D 42 61 72 63 6F 64 65 56 4A 03
1Field001=Text|**1**Field002=13/05/2022| **1**Field003=17:30:16PM|**2**Field004=0899| **2**Field005=123456789| **2**Field006=2D-Barcode|**3**Field007=VJ	[STX] Text13/05/202217:30:16PM [0A] 0899123456789 2D-Barcode [0A] VJ [ETX] 02 54 65 78 74 31 33 2F 30 35 2F 32 30 32 32 31 37 3A 33 30 3A 31 36 50 4D 0A 30 38 39 39 31 32 33 34 35 36 37 38 39 32 44 2D 42 61 72 63 6F 64 65 0A 56 4A 03
1Field001=Text|**1**Field002=13/05/2022| **1**Field003=17:30:16PM|**B**Field004=0899| **B**Field005=123456789| **B**Field006=2D-Barcode|**C**Field007=VJ	[STX] Text13/05/202217:30:16PM [0A] 0899123456789 2D-Barcode [0A] VJ [ETX] 02 54 65 78 74 31 33 2F 30 35 2F 32 30 32 32 31 37 3A 33 30 3A 31 36 50 4D 0A 30 38 39 39 31 32 33 34 35 36 37 38 39 32 44 2D 42 61 72 63 6F 64 65 0A 56 4A 03

5.3.24 “Get Next Job Printed” Packet Format

Description: Returns data defining the next job to be printed

Format: **[STX] [TYPE] [DATA] [ETX]**

- The TYPE field contains the single ASCII character ‘G’ / ‘g’.
- The DATA command is ‘D’/‘d’ to get next job to be printed

Response: The machine responds with the following Packet:

[STX] [DATA] [ETX]

- DATA is the content of each field concatenated in alphabetical order of the field name. And when line designator inserted as first character in the field name then each line will be delimited with a line feed character.
  - Line designator can be alphanumeric i.e. 1,2,3,a,b,c,A,B,C, etc.
  - Alphabetical order of field name follows digit, Upper Case, Lower case and UTF-8 characters.
- Supports both ASCII and UTF-8

Additional Information:

- All field values will send in string format
- For Logo field, Logo name will send in string format
- For Custom Reference, value of each insert will send in text format
- If Job is empty and no job loaded, empty data will send
- Please refer “Additional Examples” on page A-38.

Example

Job Data	Command Response
Field001=Text|Field002=13/05/2022| Field003=17:30:16PM|Field004=0899| Field005=123456789| Field006=2DBarcode|Field007=VJ	[STX] Text13/05/202217:30:16PM 08991234567892D-BarcodeVJ [ETX] 02 54 65 78 74 31 33 2F 30 35 2F 32 30 32 32 31 37 3A 33 30 3A 31 36 50 4D 30 38 39 39 31 32 33 34 35 36 37 38 39 32 44 2D 42 61 72 63 6F 64 65 56 4A 03
1Field001=Text|**1**Field002=13/05/2022| **1**Field003=17:30:16PM|**2**Field004=0899| **2**Field005=123456789| **2**Field006=2D-Barcode|**3**Field007=VJ	[STX] Text13/05/202217:30:16PM [0A] 08991234567892D-Barcode [0A] VJ [ETX] 02 54 65 78 74 31 33 2F 30 35 2F 32 30 32 32 31 37 3A 33 30 3A 31 36 50 4D 0A 30 38 39 39 31 32 33 34 35 36 37 38 39 32 44 2D 42 61 72 63 6F 64 65 0A 56 4A 03
1Field001=Text|**1**Field002=13/05/2022| **1**Field003=17:30:16PM|**B**Field004=0899| **B**Field005=123456789| **B**Field006=2D-Barcode|**C**Field007=VJ	[STX] Text13/05/202217:30:16PM [0A] 08991234567892D-Barcode [0A] VJ [ETX] 02 54 65 78 74 31 33 2F 30 35 2F 32 30 32 32 31 37 3A 33 30 3A 31 36 50 4D 0A 30 38 39 39 31 32 33 34 35 36 37 38 39 32 44 2D 42 61 72 63 6F 64 65 0A 56 4A 03

6 Differences between 1000 Series and SIMPLICiTY Series WSI Protocol

6.1 Different Terminology between 1000 Series Printer and SIMPLICiTY based printer

Description	1000 Series Printer	SIMPLICiTY Based Printer
Print Job	Referred as Print Message	Referred as Print Job
User Fields	Referred as User Inserts/User Edits	Referred as Custom Reference

6.2 Following are the Differences between the Printers

6.2.1 Update Job Text

Description	1000 Series Printer	SIMPLICiTY Based Printer
User Prompted Fields	In case the new Job text includes prompted user fields the packet will not be Acknowledged until the operator inputs the prompted text.	Operator will not be prompted for input data.
Font Mapping	Please refer “Font Mapping” on page A-1	
Fonts		When both options “Tower Printing” and “Custom Font” selected, It will set the Tower Fonts
Barcode Type Supported		GS1 Databar barcode type will not be supported by SIMPLICiTY based Printer
Barcode Generation Fails	Send NACK	Ignored
Clock Code	Please refer “Clock Code” on page A-14	
Logo Field Type	• Logo field type is supported via User Edits	• New Attribute is added for Logo Field Type (ATTRIB 3 BIT 4) • Job Text will be considered as Logo Name

Description	1000 Series Printer	SIMPLICiTY Based Printer
Size check	There is not a size check for this command.	Minimum size for command data is 15 bytes. If Printer receives less than minimum size, it will send NACK.
Bold	Double Dots Multistoke - 2 Triple Dots Multistoke - 3 When both option “Double Dots” and “Triple Dots” selected then it will set to Multistoke - 4	Double Dots Bold1 Triple Dots Bold2 When both option “Double Dots” and “Triple Dots” selected then it will set to “Bold2”

6.2.2 Modify Job Parameters

The table below lists the parameters that will not be supported by SIMPLICiTY based Printer. User can either send 0 or any other value. These parameters will be ignored by the printer:

- Height (EHT)
- Expiry Date (EXP)
- Hejra expiry days ahead (HEJRA)
- Printed Dots (DRP)

Job Parameter Range Differences

Job Parameter	1000 Series Printer	SIMPLICiTY Based Printer	
		UI	Protocol
Width (WID)	0.004 to 39.366 Inches 0.100 to 999.90 mm	Print Length: 0.02 to 100.00 Inches 0.4 to 2540.0 mm	Print Length: 0.02 to 99.9 Inches 1 to 2540.0 mm
Product Delay (DLY)	0 to 39.366 Inches 0 to 999.90 mm	0.04 to 3937.01 Inches 1 to 100000 mm	0.04 to 3937.01 Inches 1 to 100000 mm
Reverse Delay (RDLY)	0 to 39.366 Inches 0 to 999.90 mm	0.00 to 393.70 Inches 0.0 to 10000 mm	
Character Gap (GAP)	1 to 10	0 to 9	0 to 9
Raster Repeat (BLD)	1 to 10	1 - Off 2 - Bold 1 3 - Bold 2	1 - Off 2 - Bold 1 3 - Bold 2

6.2.3 Update User Field Data

Description	1000 Series Printer	SIMPLICiTY Based Printer
Counter - Direction		Direction option is not available in simplicity based printer. Start and Stop value will be verify and updated base on direction value, • For incrementing counters, Start < End • For decrementing counters, Start > End
Counter - Current Value		Current value is mapped to user prompted (current) value. Current value option will not be available on User Interface
Counter - Repeat Value	Repeat value is number of time counter needs to be rollover when it reaches to end value.	Repeat value is mapped to “Restart On Rollover” 0 = Restart On Rollover Unchecked 1 or any positive value = Restart On Rollover checked
Counter - Padding		Pad Character = Empty • Disable Padding Pad Character = ASCII [‘0’-‘9’,’A’-‘Z’,‘a’-‘z’] or SPACE • Enable Padding • Update ASCII [‘0’-‘9’,’A’-‘Z’,‘a’-‘z’] or SPACE to Pad Character
Counter Type		It does not support alphanumeric counter

6.2.4 Set Module Widths

Description	1000 Series Printer	SIMPLICiTY Based Printer
Functionality	Set absolute value for bars and gaps width	Set offset value for bars and gaps width
Value Range	• From UI: 1 to 8 • From WSI Protocol: 1 to 12	• From UI: -1 to 5 • From WSI Protocol: -1 to 5

6.2.5 Print On/Off

Description	1000 Series Printer	SIMPLICiTY Based Printer
Y Field Values	For 1610DH, Y can be • 0 = print OFF (In Independent Mode, turn OFF the print head on which command is received while in System Mode, turn the print OFF for both heads.) • 1 = print ON, Turn Print ON on the head that the command is received. • 2 = print ON for both heads	Y can be 0 or 1 forhead 1. • ‘0’ disables print mode • ‘1’ enables print mode

6.2.6 Error Status Request

Description	1000 Series Printer	SIMPLICiTY Based Printer
Not Supported Error Status		• Charge Error • RTC Invalid • Ink Service Over Due • No Inksys • EHT Not Calibrated Zero will be sent for unsupported error status

6.2.7 Remote Data

Description	1000 Series Printer	SIMPLICiTY Based Printer
Buffering Concept	• It has 10 queues with 200 capacity store 10 Remote Data • Remote data will be fetched from individual queues	• It has one queue with 201 capacity • Remote data will be stored as a group in queue
Remote Data Fields Job	• Remote data field will be created using User edit and added into the Message • Remote data field has its own remote data id • Remote data id is 1 to 10	• User Prompted Text in Job will act as remote data field
Remote Data Update	• Remote data will be updated based on Remote data id	• Remote data will be updated in Job in order user prompted text is created • Example: Job Contains - Field 1 - User Prompted Text, Field 2 - User Prompted Text, Field 3 - User Prompted Text Remote Data 1 will go to Field 1, Remote Data 2 will go to Field 2, Remote Data 3 will go to Field 3.
Field Types	• Remote Data supports Text Field only	• Remote Data supports User Prompted Text, User Prompted Date and User Prompted Counter, User Prompted 2D barcode and Fields marked as Customer reference (Only first insert of Custom reference if it is user prompted)
Remote Data Source Action	• When Remote Data Source Action value change, change will be take into affect immediately • Example: When action change to “Stop” from “Repeat on Last Value” selected, Printer stops printing after last message.	• In SIMPLICiTY, Remote Data Source Action value considered when remote data pushed to Queue. • Remote Data Source Action value change will not considered after pushing data to queue. • Example: When remote data pushed to queue with “Repeat on Last Value”, It will print the last job continuously. Print will not stop if user change value to “Stop” in configuration option.
Speed / Performance	• Based Architecture and RAM Queue, Remote data execution is faster in 1000L	• Because of CQM, Remote data execution will be slow.

A Additional Specification Information

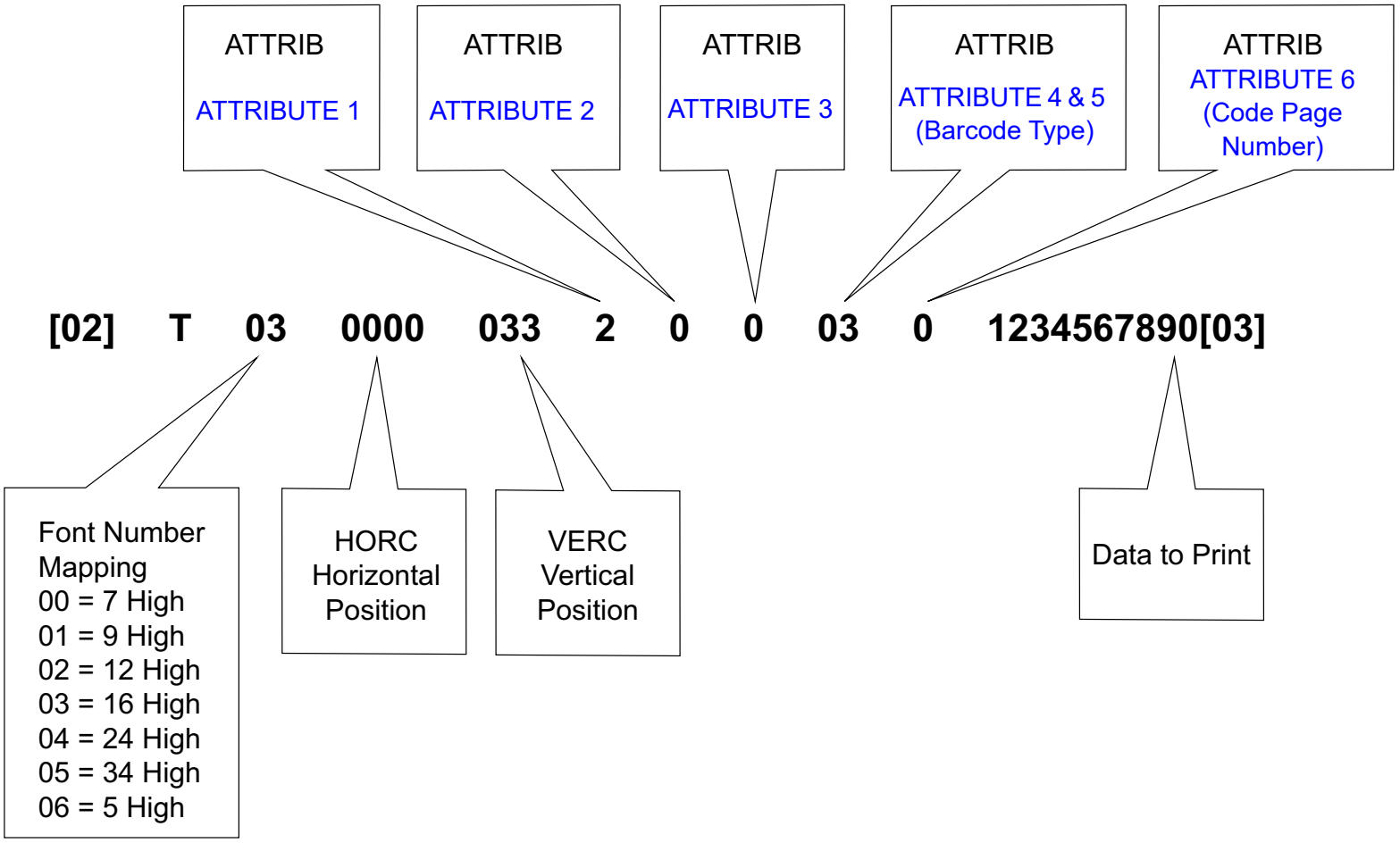
A.1 Font Mapping

1000L Printer	SIMPLICiTY Based Printer
00 = "7 high International" 01 = "9 high International" 02 = "12 high International" 03 = "16 high International" 04 = "24 high International" 05 = "34 high International" 06 = "5 high International"	**Font number mapping:** 00 = "7 high International" 01 = "9 high International" 02 = "12 high International" 03 = "16 high International" 04 = "24 high International" 05 = "34 high International" 06 = "5 high International" 07 = "7 high International Narrow" 08 = "5 high International Narrow" 09 = "7 high Chinese" 10 = "9 high Chinese" 11 = "12 high Chinese" 12 = "16 high Chinese" 13 = "24 high Chinese" 14 = "34 high Chinese" 15 = "5 high Chinese" 16 = "7 high Traditional Chinese" 17 = "9 high Traditional Chinese" 18 = "12 high Traditional Chinese" 19 = "16 high Traditional Chinese" 20 = "24 high Traditional Chinese" 21 = "34 high Traditional Chinese" 22 = "5 high Traditional Chinese" 23 = "9 high Chinese Narrow" ***Note:*** *Numbers other than above list will select the by default "7 high International".* **Font number mapping when Tower Printing attribute selected:** 00 = "7 high Tower" 01 = "5 high Tower" ***Note:*** *Numbers other than above list will select the by default "7 high Tower".*

1000L Printer	SIMPLICiTY Based Printer
	Font number mapping when Use Custom Font attribute selected: 00 = "7 High_Custom" 01 = "9 High_Custom" 02 = "12 High_Custom" 03 = "16 High_Custom" 04 = "24 High_Custom" 05 = "34 High_Custom" 06 = "5 High_Custom" ***Note:*** *Only above mention Pre-defined custom font will work with WSI.* ***Note:*** *If the custom font is not available or Numbers other than above list will select the by default "7 high International".* **Font number mapping with barcode height when field type is barcode:** 00 = 07 01 = 09 02 = 12 03 = 16 04 = 24 05 = 34 06 = 5 ***Note:*** *Barcode Human Readable text will be always in 5 high international. We cannot set Font for human readable text using WSI.*

A.2 T Command Structure and Parameters Mapping

A.2.1 T Command Structure



A.2.2 Parameters Mapping

1000L Printer	SIMPLICiTY Based Printer						
Attributes	**Text**	**1D Barcode**	**2D Barcode**	**Date**	**Time**	**Custom Reference**	**Logo**
Invert	Invert	Invert	Invert	Invert	Invert	Invert	Invert
Reverse	Mirror	Mirror	Mirror	Mirror	Mirror	Mirror	Mirror
Clock Code	NA	NA	NA	Date Type: Custom	Time Type: Custom	NA	NA
Double Dots	Bold 1	Barcode Magnification 84.6%(2px)	NA	Bold 1	Bold 1	Bold 1	NA
Triple Dots	Bold 2	Barcode Magnification 126.9%(3px)	NA	Bold 2	Bold 2	Bold 2	NA
Prompted Field	User Prompted	NA	User Prompted	NA	NA	NA	NA
Barcode HR	NA	Add Human Readable Text	NA	NA	NA	NA	NA
Custom Font	Please refer “Font Mapping” on page A-1	NA	NA	Please refer “Font Mapping” on page A-1	Please refer “Font Mapping” on page A-1	Please refer “Font Mapping” on page A-1	NA
Invert Video	Inverse	Inverse	Inverse	Inverse	Inverse	Inverse	Inverse
Barcode Checksum	NA	Calculated Checksum	NA	NA	NA	NA	NA
Logo	NA	NA	NA	NA	NA	NA	Logo

A.2.3 T Command Examples

Example 1

[02]T010001009000000HELLO STAR CENTER[03]	
[02]	
T	
01	9 High Font
0001	Horizontal Position
009	Vertical Position
0	Attribute 1
0	Attribute 2
0	Attribute 3
00	Attribute 4 & 5
0	Code Page
HELLO STAR CENTER	Job Data
[03]	

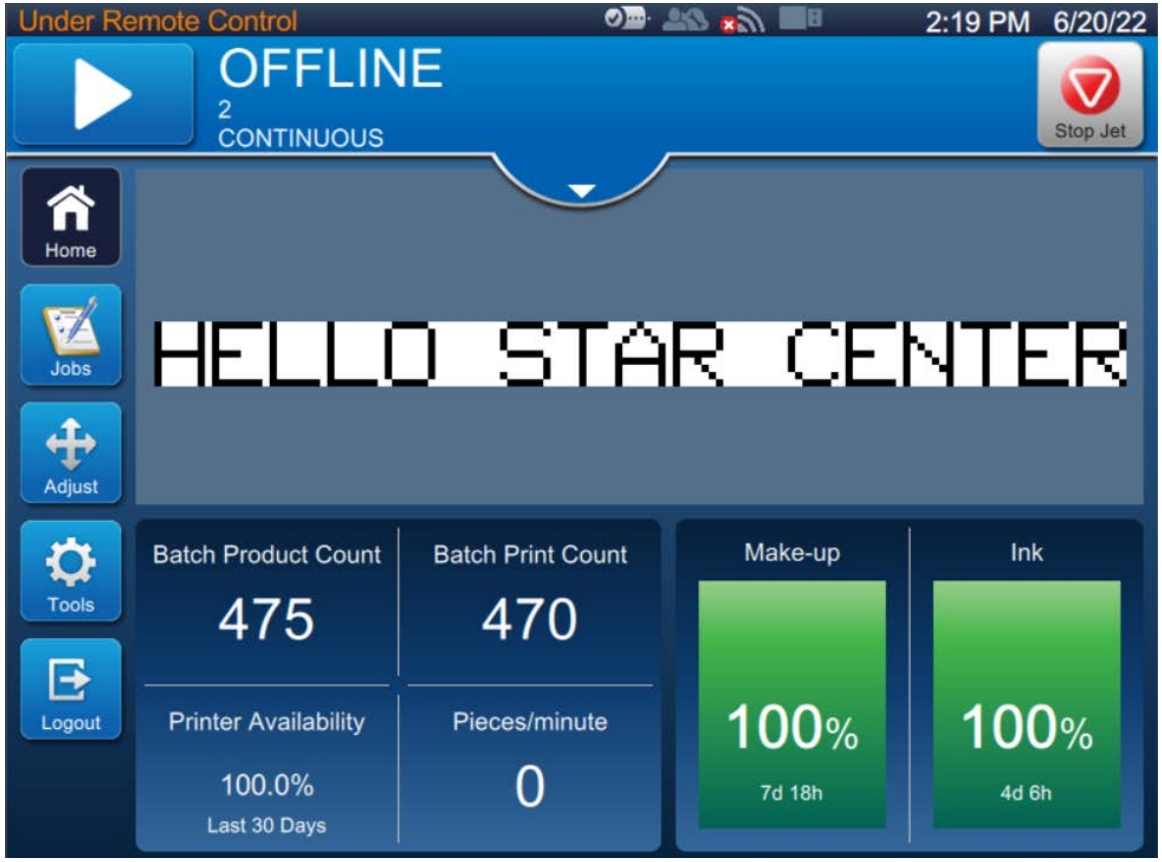


Figure A-1: WSI T Command Example 1

Example 2

[02]T000001007000000WSI PROTOCOL 10[0A]000001015000000WSI PROTOCOL 11[03]	
[02]	
T	
00	7 High Font
0001	Horizontal Position
007	Vertical Position
0	Attribute 1
0	Attribute 2
0	Attribute 3
00	Attribute 4 & 5
0	Code Page
WSI PROTOCOL 10	Job Data
[0A]	Line Feed Character
00	7 High Font
0001	Horizontal Position
015	Vertical Position
0	Attribute 1
0	Attribute 2
0	Attribute 3
00	Attribute 4 & 5
0	Code Page
WSI PROTOCOL 11	Job Data
[03]	

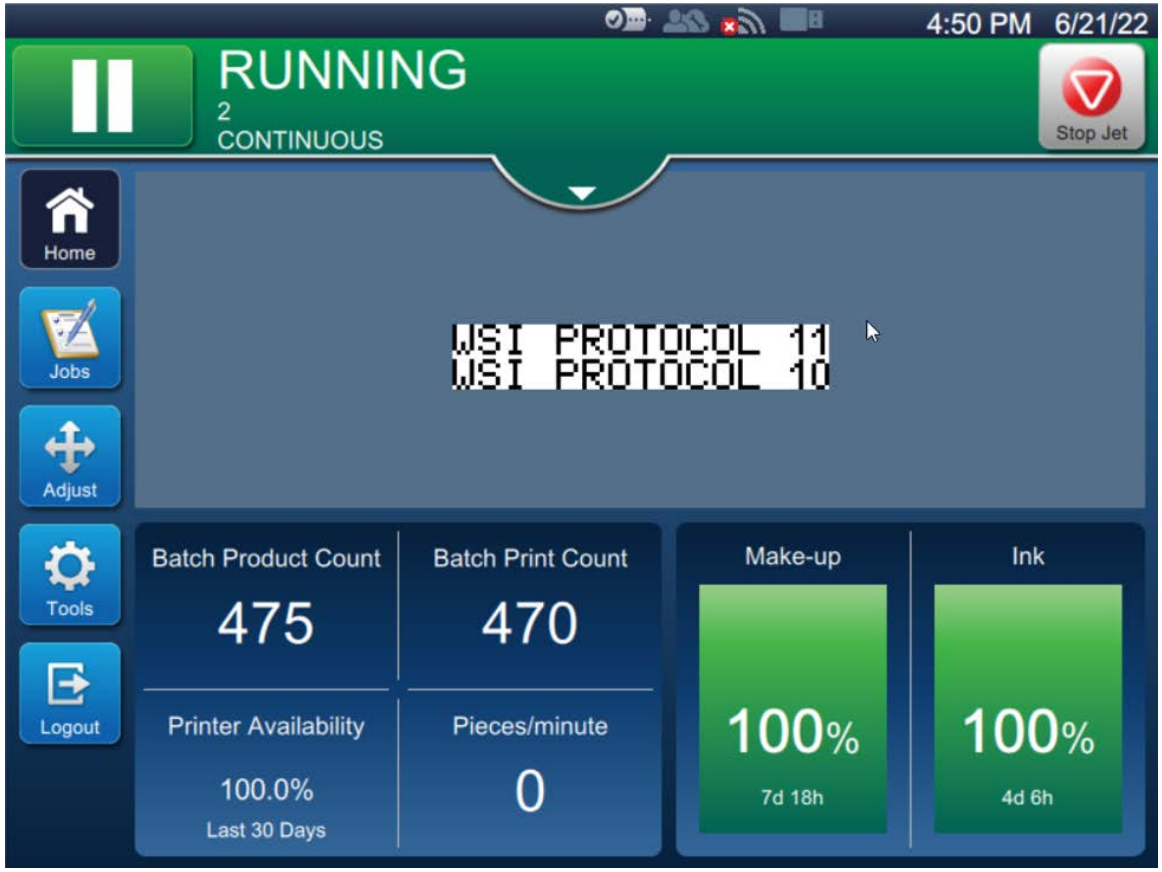


Figure A-2: WSI T Command Example 2

Example 3

[02]T000001007300000ABCD123[03]	
[02]	
T	
00	7 High Font
0001	Horizontal Position
007	Vertical Position
3	Attribute 1 (Invert Message ON, Reverse Message ON)
0	Attribute 2
0	Attribute 3
00	Attribute 4 & 5
0	Code Page
ABCD123	Job Data
[03]	

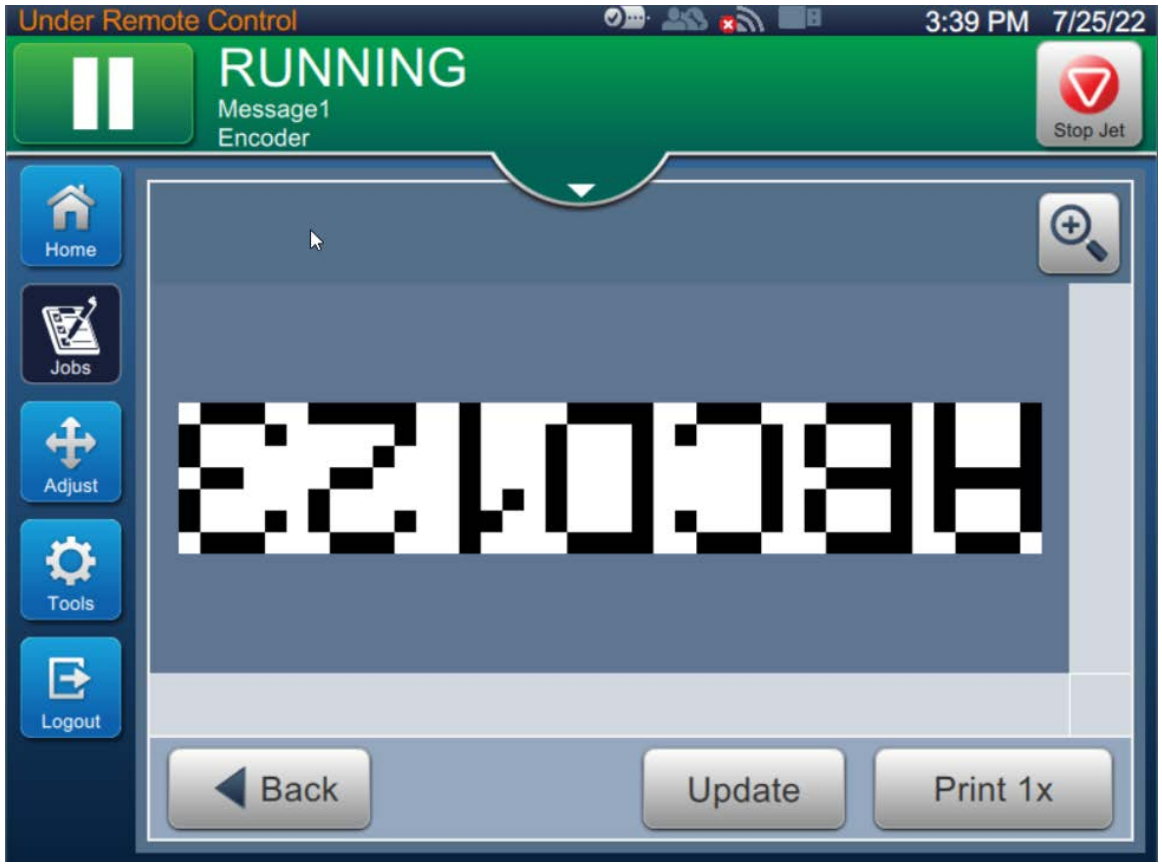


Figure A-3: WSI T Command Example 3

Example 4

[02]T000001007000000WSI LINE 1[0A]000001015000000WSI LINE 2[0A]000001023000000WSI LINE 3 [03]	
[02]	
T	
00	7 High Font
0001	Horizontal Position
007	Vertical Position
0	Attribute 1
0	Attribute 2
0	Attribute 3
00	Attribute 4 & 5
0	Code Page
WSI LINE 1	Job Data
[0A]	Line Feed Character
00	7 High Font
0001	Horizontal Position
015	Vertical Position
0	Attribute 1
0	Attribute 2
0	Attribute 3
00	Attribute 4 & 5
0	Code Page
WSI LINE 2	Job Data

[0A]	Line Feed Character
00	7 High Font
0001	Horizontal Position
023	Vertical Position
0	Attribute 1
0	Attribute 2
0	Attribute 3
00	Attribute 4 & 5
0	Code Page
WSI LINE 3	Job Data
[03]	

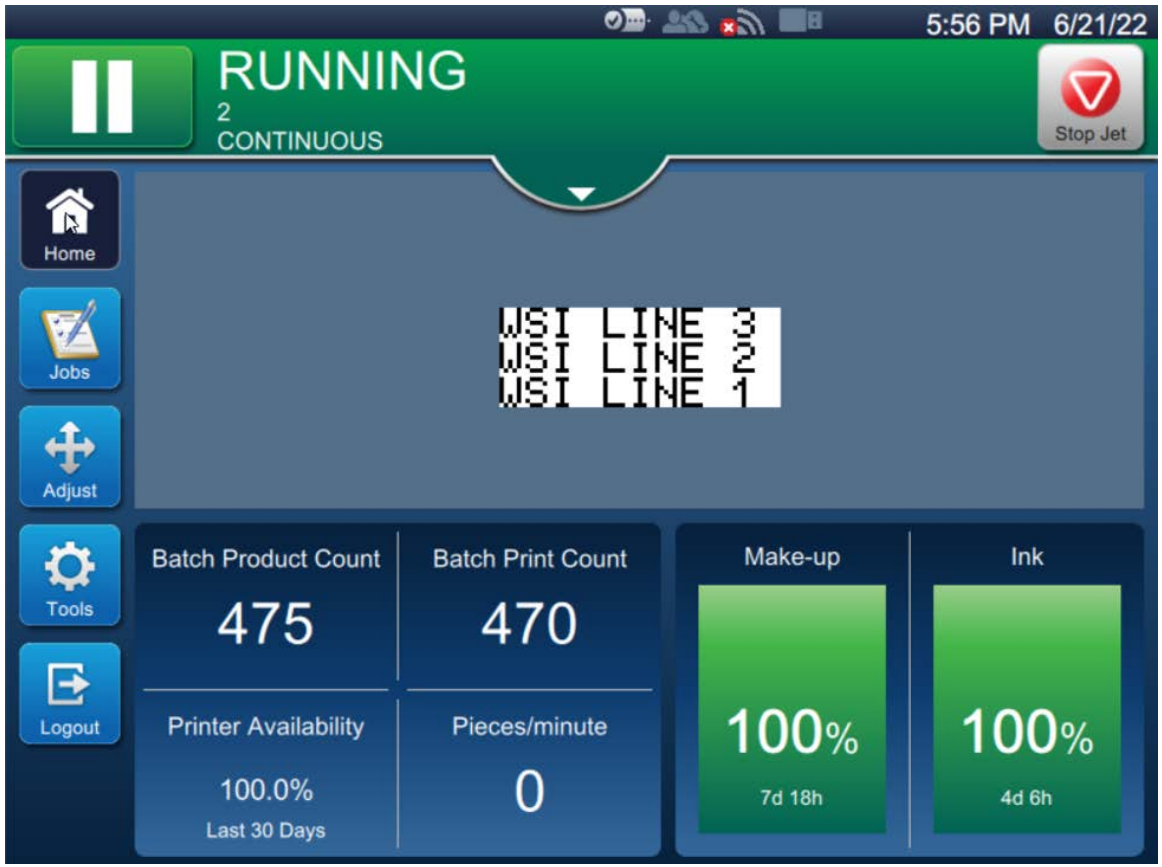


Figure A-4: WSI T Command Example 4

A.3 Clock Code

A.3.1 1000L Printer and SIMPLICiTY based printer Mapping

Clock Code (1000L Printer)	Description (1000L Printer)	Clock Code (SIMPLICiTY Based Printer)	Clock Code Combinations (1000L Printer)	Clock Code Combinations (SIMPLICiTY Based Printer)	Description (SIMPLICiTY Based Printer)
A	DAY	ddd			ddd = Day of the week, e.g. 1, 2, 3
B	YEAR_TENS_DIGIT		BC	yy	yy = Year represented by only the last two digits, with leading zero, e.g. 2001=01
C	YEAR_ONES_DIGIT	y			y = Year represented by only the last digit, e.g. 2010=0
D	MONTH_TENS_DIGIT		DE	MM	M = Numeric representation of the month, without leading zeros
E	MONTH_ONES_DIGIT				MM = Numeric representation of the month, with leading zeros
F	DATE_TENS_DIGIT		FG	dd	d = Numeric representation of the day of the month, without leading zeros
G	DATE_ONES_DIGIT				dd = Numeric representation of the day of the month, with leading zeros
H	HOUR_TENS_DIGIT		HI	HH	HH = Hours with leading zeros (24-hour clock)
I	HOUR_ONES_DIGIT				

Clock Code (1000L Printer)	Description (1000L Printer)	Clock Code (SIMPLICiTY Based Printer)	Clock Code Combinations (1000L Printer)	Clock Code Combinations (SIMPLICiTY Based Printer)	Description (SIMPLICiTY Based Printer)
J	MIN_TENS_DIGIT		JK	mm	mm = Minutes with leading zeros
K	MIN_ONES_DIGIT				
L	SEC_TENS_DIGIT		LM	ss	ss = Seconds with leading zeros
M	SEC_ONES_DIGIT				
N	FIRST_ALPHA_MONTH		NOP	MMM	MMM = Three-letter abbreviation for the name of the month
O	SECOND_ALPHA_MONTH				
P	THIRD_ALPHA_MONTH		NOPy	MMMM	MMMM = Full name of the month
y	FOURTH_ALPHA_MONTH				
Q	JULIAN_HUNDREDS_DIGIT		QRS	jj/JJ	jj - Julian day of the Year (where 29th Feb is day 060), with leading zeros
R	JULIAN_TENS_DIGIT				
S	JULIAN_ONES_DIGIT				JJ - Julian day of the Year (where 29th Feb is day 366), with leading zeros
T	NORTH_AMERICAN_WEEK_TENS_DIGIT		TU	uu	u = Week number (North America)
U	NORTH_AMERICAN_WEEK_ONES_DIGIT				uu = Week number (North America), with leading zeros

Clock Code (1000L Printer)	Description (1000L Printer)	Clock Code (SIMPLICiTY Based Printer)	Clock Code Combinations (1000L Printer)	Clock Code Combinations (SIMPLICiTY Based Printer)	Description (SIMPLICiTY Based Printer)
V	EUROPEAN_DIGIT_WEEK_TENS_DIGIT		VW	ww	w = Week of the year, without leading zeros ww = Week of the year, with leading zeros
W	EUROPEAN_DIGIT_WEEK_ONES_DIGIT				
X	ENCODED_DAY_OF_WEEK	dddc			dddc = Day of the week code, e.g. A, B, C
Z	HEJRA_YEAR_THOUSANDS_DIGIT		h		
a	HEJRA_YEAR_HUNDREDS_DIGIT		i		
b	ENCODED_HOUR	hhc			hhc = Code for hour of the day
h	YEAR_THOUSANDS_DIGIT		hiBC	yyyy	yyyy = Year represented by all four digits
i	YEAR_HUNDREDS_DIGIT				
j	HEJRA_YEAR_TENS_DIGIT		j	yy	yy = Year represented by only the last two digits, with leading zero, e.g. 2001=01 y = Year represented by only the last digit, e.g. 2010=0
k	HEJRA_YEAR_ONES_DIGIT	y	k		
l	HEJRA_MONTH_TENS_DIGIT		l	MM	M = Numeric representation of the month, without leading zeros MM = Numeric representation of the month, with leading zeros
m	HEJRA_MONTH_ONES_DIGIT		m		

Clock Code (1000L Printer)	Description (1000L Printer)	Clock Code (SIMPLICiTY Based Printer)	Clock Code Combinations (1000L Printer)	Clock Code Combinations (SIMPLICiTY Based Printer)	Description (SIMPLICiTY Based Printer)
n	FIRST_ALPHA_DAY_OF_WEEK		nop	DDD	DDD = Capitalised day of the week, e.g. MON, WED
o	SECOND_ALPHA_DAY_OF_WEEK				
p	THIRD_ALPHA_DAY_OF_WEEK				
q	HEJRA_DATE_TENS_DIGIT		q	dd	d = Numeric representation of the day of the month, without leading zeros
r	HEJRA_DATE_ONES_DIGIT		r		dd = Numeric representation of the day of the month, with leading zeros
z	JULIAN_DAY	j			
^B	CURRENT_YEAR_TENS_DIGIT		^B ^C	yy	yy = Year represented by only the last two digits, with leading zero, e.g. 2001=01
^C	CURRENT_YEAR_ONES_DIGIT	y			
^D	CURRENT_MONTH_TENS_DIGIT		^D ^E	MM	MM = Numeric representation of the month, with leading zeros
^E	CURRENT_MONTH_ONES_DIGIT				
^F	CURRENT_DATE_TENS_DIGIT		^F ^G	dd	dd = Numeric representation of the day of the month, with leading zeros
^G	CURRENT_DATE_ONES_DIGIT				
^N	CURRENT_FIRST_ALPHA_MONTH		^N ^O ^P	MMM	MMM = Three-letter abbreviation for the name of the month
^O	CURRENT_SECOND_ALPHA_MONTH				
^P	CURRENT_THIRD_ALPHA_MONTH				

Clock Code (1000L Printer)	Description (1000L Printer)	Clock Code (SIMPLICiTY Based Printer)	Clock Code Combinations (1000L Printer)	Clock Code Combinations (SIMPLICiTY Based Printer)	Description (SIMPLICiTY Based Printer)
^h	CURRENT_YEAR_THOUSANDS_DIGIT		^h ^i ^B ^C	yyyy	yyyy = Year represented by all four digits
^i	CURRENT_YEAR_HUNDREDS_DIGIT				
@	DAY_OF_WEEK_SHORT	DDD			DDD = Capitalised day of the week, e.g. MON, WED
@#$	HOUR_OF_WEEK	iii			iii = Hour of the week, without leading zeros (from a user defined start of week)
!E	MONTH_NO_LEADING	M			M = Numeric representation of the month, without leading zeros
?G	DAY_OF_MONTH_NO_LEADING	ddc			ddc = Day of the month, e.g. 1, 2, 3

Note: *HEJRA date formats will be achieved by setting the language as "Arabic - Saudi Arabia".*

A.3.2 Date Format

Clock Code (1000L Printer)	Description (1000L Printer)	Clock Code (SIMPLICiTY Based Printer)	Description (SIMPLICiTY Based Printer)
A	DAY	ddd	ddd = Day of the week, e.g. 1, 2, 3
BC	(YEAR_TENS_DIGIT)(YEAR_ONES_DIGIT)	yy	yy = Year represented by only the last two digits, with leading zero, e.g. 2001=01
C	YEAR_ONES_DIGIT	y	y = Year represented by only the last digit, e.g. 2010=0
DE	(MONTH_TENS_DIGIT)(MONTH_ONES_DIGIT)	MM	MM = Numeric representation of the month, with leading zeros
FG	(DATE_TENS_DIGIT)(DATE_ONES_DIGIT)	dd	dd = Numeric representation of the day of the month, with leading zeros
NOP	(FIRST_ALPHA_MONTH)(SECOND_ALPHA_MONTH) (THIRD_ALPHA_MONTH)	MMM (cl:MMM)	MMM = Three-letter abbreviation for the name of the month
NOPy	(FIRST_ALPHA_MONTH)(SECOND_ALPHA_MONTH) (THIRD_ALPHA_MONTH)(FOURTH_ALPHA_MONTH)	MMMM (cl:MMMM)	MMMM = Full name of the month
QRS	(JULIAN_HUNDREDS_DIGIT)(JULIAN_TENS_DIGIT) (JULIAN_TENS_DIGIT)	jj/JJ (cl:jj)	jj - Julian day of the Year (where 29th Feb is day 060), with leading zeros JJ - Julian day of the Year (where 29th Feb is day 366), with leading zeros
TU	(NORTH_AMERICAN_WEEK_TENS_DIGIT) (NORTH_AMERICAN_WEEK_ONES_DIGIT)	uu (cl:uu)	uu = Week number (North America), with leading zeros
VW	(EUROPEAN_DIGIT_WEEK_TENS_DIGIT) (EUROPEAN_DIGIT_WEEK_ONES_DIGIT)	ww (cl:ww)	ww = Week of the year, with leading zeros

Clock Code (1000L Printer)	Description (1000L Printer)	Clock Code (SIMPLICiTY Based Printer)	Description (SIMPLICiTY Based Printer)
X	ENCODED_DAY_OF_WEEK	dddc (cl:dddc)	dddc = Day of the week code, e.g. A, B, C
Zajk	(HEJRA_YEAR_THOUSANDS_DIGIT) (HEJRA_YEAR_HUNDREDS_DIGIT)(HEJRA_YEAR_TENS_DIGIT) (HEJRA_YEAR_ONES_DIGIT)	yyyy	yyyy = Year represented by all four digits
hiBC	(YEAR_THOUSANDS_DIGIT)(YEAR_HUNDREDS_DIGIT) (YEAR_TENS_DIGIT)(YEAR_ONES_DIGIT)	yyyy	yyyy = Year represented by all four digits
jk	(HEJRA_YEAR_TENS_DIGIT)(HEJRA_YEAR_ONES_DIGIT)	yy	yy = Year represented by only the last two digits, with leading zero, e.g. 2001=01
k	HEJRA_YEAR_ONES_DIGIT	y	y = Year represented by only the last digit, e.g. 2010=0
lm	(HEJRA_MONTH_TENS_DIGIT)(HEJRA_MONTH_ONES_DIGIT)	MM	MM = Numeric representation of the month, with leading zeros
nop	(FIRST_ALPHA_DAY_OF_WEEK)(SECOND_ALPHA_DAY_OF_WEEK) (THIRD_ALPHA_DAY_OF_WEEK)	DDD (cl:DDD)	DDD = Capitalised day of the week, e.g. MON, WED
qr	(HEJRA_DATE_TENS_DIGIT)(HEJRA_DATE_ONES_DIGIT)	dd	dd = Numeric representation of the day of the month, with leading zeros
z	JULIAN_DAY	j/J (cl:j)	j = Julian day of the Year (where 29th Feb is day 60), without leading zeros J = Julian day of the Year (where 29th Feb is day 366), without leading zeros
^B^C	(CURRENT_YEAR_TENS_DIGIT)(CURRENT_YEAR_ONES_DIGIT)	yy	yy = Year represented by only the last two digits, with leading zero, e.g. 2001=01

Clock Code (1000L Printer)	Description (1000L Printer)	Clock Code (SIMPLICiTY Based Printer)	Description (SIMPLICiTY Based Printer)
^C	CURRENT_YEAR_ONES_DIGIT	y	y = Year represented by only the last digit, e.g. 2010=0
^D^E	(CURRENT_MONTH_TENS_DIGIT)(CURRENT_MONTH_ONES_DIGIT)	MM	MM = Numeric representation of the month, with leading zeros
^F^G	(CURRENT_DATE_TENS_DIGIT)(CURRENT_DATE_ONES_DIGIT)	dd	dd = Numeric representation of the day of the month, with leading zeros
^N^O^P	(CURRENT_FIRST_ALPHA_MONTH) (CURRENT_SECOND_ALPHA_MONTH) (CURRENT_THIRD_ALPHA_MONTH)	MMM (cl:MMM)	MMM = Three-letter abbreviation for the name of the month
^h^i^B^C	(CURRENT_YEAR_THOUSANDS_DIGIT) (CURRENT_YEAR_HUNDREDS_DIGIT) (CURRENT_YEAR_TENS_DIGIT) (CURRENT_YEAR_ONES_DIGIT)	yyyy	yyyy = Year represented by all four digits
@	DAY_OF_WEEK_SHORT	DDD (cl:DDD)	DDD = Capitalised day of the week, e.g. MON, WED
!E	MONTH_NO_LEADING	M	M = Numeric representation of the month, without leading zeros
?G	DAY_OF_MONTH_NO_LEADING	ddc (cl:ddc)	ddc = Day of the month, e.g. 1, 2, 3

Note: *HEJRA date formats will be achieves by setting the language as "Arabic - Saudi Arabia"*

A.3.3 Time Format

Clock Code (1000L Printer)	Description (1000L Printer)	Clock Code (SIMPLICiTY Based Printer)	Clock Code Combinations (1000L Printer)	Clock Code Combinations (SIMPLICiTY Based Printer)	Description (SIMPLICiTY Based Printer)
HI	(HOUR_TENS_DIGIT) (HOUR_ONES_DIGIT)	HH	HI	HH	HH = Hours with leading zeros (24-hour clock)
JK	(MIN_TENS_DIGIT) (MIN_ONES_DIGIT)	mm	JK	mm	mm = Minutes with leading zeros
LM	(SEC_TENS_DIGIT) (SEC_ONES_DIGIT)	ss	LM	ss	ss = Seconds with leading zeros
b	ENCODED_HOUR	hhc (cl:hhc)			hhc = Code for hour of the day
@#$	HOUR_OF_WEEK	iii (cl:iii)			iii = Hour of the week, without leading zeros (from a user defined start of week)

A.3.4 Date and Time Format - Separator

Description	Symbol
Forward Slash	/
Back Slash	\
Space	
Dash	-
Full Stop	.
Comma	,
Colon	:

A.4 Fields Addition and Validation Rules

Field Type	Add Field	Add Field (Ignore the failure)	Discard and Do not add field (Do not send NACK)	Discard and Do not add field (Send NACK)
Text	• Job text should be of 1-200 length	• Fail to set any of below parameter, ignore and go with default value ◦ Font ◦ Invert ◦ Inverse ◦ Bold ◦ Mirror	• Field position (with respect to Font) is invalid • Set Field position to IM nodes ◦ Delete added Field	• If Delete field fails ◦ Leave Job as empty and save the Job
User Prompted Text	• Job text should be of 1-200 length	• Fail to set any of below parameter, ignore and go with default value ◦ Font ◦ Invert ◦ Inverse ◦ Bold ◦ Mirror • If not able to set as User prompted then go with fixed text	• Field position (with respect to Font) is invalid • Set Field position to IM nodes ◦ Delete added Field	• If Delete field fails ◦ Leave Job as empty and save the Job

Field Type	Add Field	Add Field (Ignore the failure)	Discard and Do not add field (Do not send NACK)	Discard and Do not add field (Send NACK)
Logo	• Logo available in Printer	• Printer does not have specified logo ◦ Convert into Text and Add Text Field • Fail to set any of below parameter, ignore and go with default value ◦ Font ◦ Invert ◦ Inverse ◦ Bold ◦ Mirror	• Field position (with respect to Font) is invalid • Set Field position to IM nodes ◦ Delete added Field	• If Delete field fails ◦ Leave Job as empty and save the Job
1D Barcode	• Contains valid Barcode text in JobText according to barcode type selected.	• Fail to set any of below parameter, ignore and go with default value ◦ Height ◦ Invert ◦ Inverse ◦ Mirror ◦ Human Readable	• Field position (with respect to Font) is invalid • Set Field position to IM nodes ◦ Delete added Field • Checksum is invalid ◦ Delete added Field • Barcode Generation fails ◦ Delete added Field	• If Delete field fails ◦ Leave Job as empty and save the Job

Field Type	Add Field	Add Field (Ignore the failure)	Discard and Do not add field (Do not send NACK)	Discard and Do not add field (Send NACK)
2D Barcode	• Contains valid Barcode text in JobText according to barcode type selected. • If received Barcode Type is Datamatrix or QR.	• Fail to set any of below parameter, ignore and go with default value ◦ Invert ◦ Inverse ◦ Mirror	• Field position (with respect to Font) is invalid • Symbol size/Error correction level ◦ Delete added Field • Set Field position to IM nodes ◦ Delete added Field • Barcode Generation fails ◦ Delete added Field	• If Delete field fails ◦ Leave Job as empty and save the Job
Custom Reference	• Contains valid Custom Reference formats	• Printer does not have specified CR ◦ Update Default Text to CR name ◦ Leave Text Type as Fixed Text • Fail to set any of below parameter, ignore and go with default value ◦ Font ◦ Invert ◦ Inverse ◦ Bold ◦ Mirror	• Field position (with respect to Font) is invalid • Set Field position to IM nodes ◦ Delete added Field	• If Delete field fails ◦ Leave Job as empty and save the Job

Field Type	Add Field	Add Field (Ignore the failure)	Discard and Do not add field (Do not send NACK)	Discard and Do not add field (Send NACK)
Date	• Contains valid date formats - Either Normal or Hejra	• Fail to set any of below parameter, ignore and go with default value ◦ Font ◦ Invert ◦ Inverse ◦ Bold ◦ Mirror	• Field position (with respect to Font) is invalid • Set Field position to IM nodes ◦ Delete added Field • Contains invalid Date Formats • Contains mix normal dates and Hejra dates format • Empty Format String	• If Delete field fails ◦ Leave Job as empty and save the Job
Time	• Contains valid time formats	• Fail to set any of below parameter, ignore and go with default value ◦ Font ◦ Invert ◦ Inverse ◦ Bold ◦ Mirror	• Field position (with respect to Font) is invalid • Set Field position to IM nodes ◦ Delete added Field • Contains invalid Time Formats • Empty Format String	• If Delete field fails ◦ Leave Job as empty and save the Job

A.5 Field Positioning

A.5.1 Horizontal Position [HORC]

- *The HORC field consists of a value between 1 and 9999, it describes the relative horizontal position at which the text fragment should be placed within the job.*
- ***Note:*** *This position is relative and subject to left justification. It should be thought of as a fragment ordering control rather than a means of placing the text fragment in an exact location. i.e. the text fragment with a HORC of 1 will always be to the left of another text fragment with a HORC of greater than 1 etc.*

A.5.2 Vertical Position [VERC]

- *The VERC field consists of exactly 3 characters (ASCII ‘0' to ‘9') which represent a decimal value between 0 and 999, it describes the vertical position at which the text field should be placed within the job.*
- ***Note:*** *For printing the whole job is subject to upward justification. If blank lines are required at the top of a job a dummy fragment containing appropriate number of space characters in an appropriate font must be sent.*

A.5.3 Field Positioning Rules

Field positioning rules are based on the HORC and VERC values, as follows:

1. Select all fields with lowest value of HORC.
2. Place them into the Job image in a left aligned column from bottom to top in the order of VERC values.
   a. Lowest VERC values start from bottom
   b. If two fields overlap with each other in same column, then the field with higher VERC will push right to the next column
3. Select all fields with second lowest value of HORC and place them in next column to the right of the fields with minimal HORC.
4. Shift each of the fields of the next column horizontally to the left until there is minimum 1 pixel gap to any of the fields of the previous column or will come to the left boundary of the Job image.

5. Continue selecting fields with next value of HORC, piling them in a column and shifting left where possible until all the fields are placed.

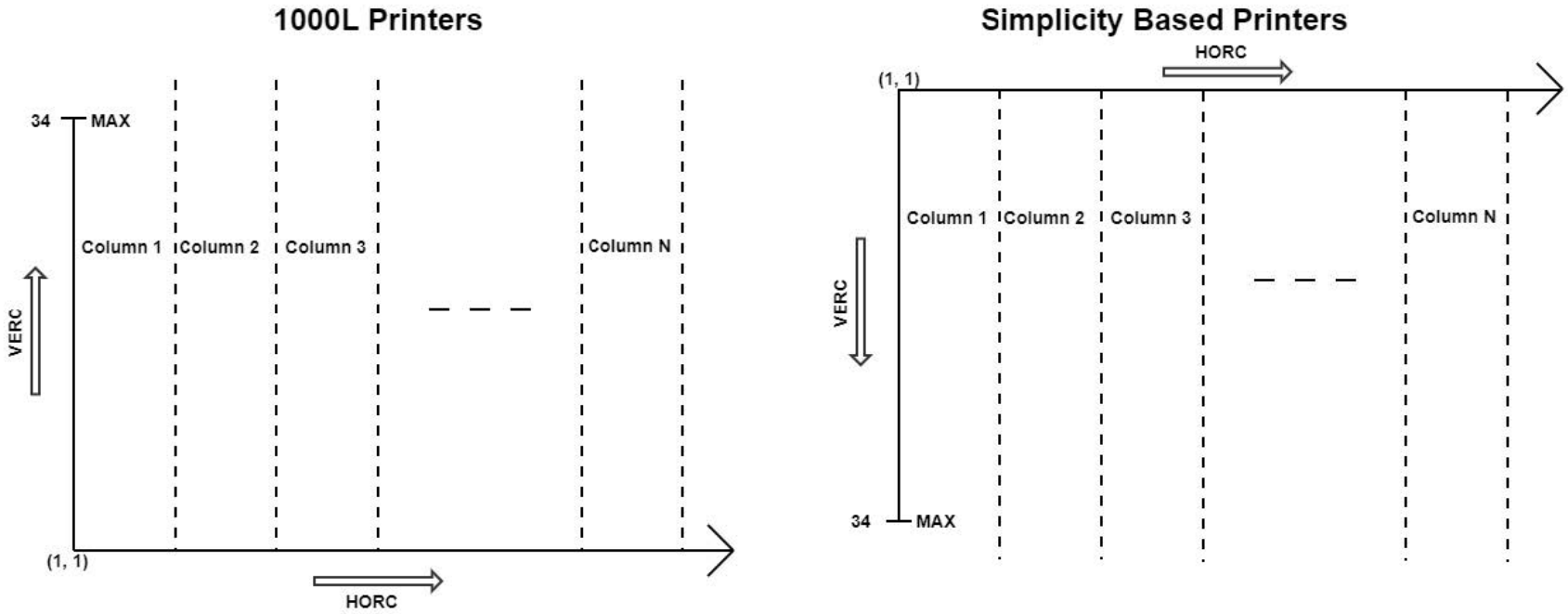
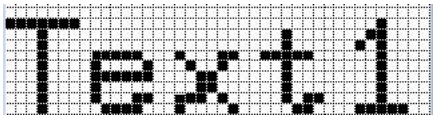
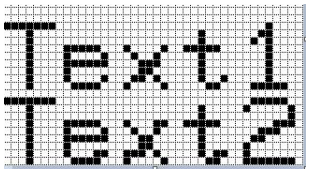
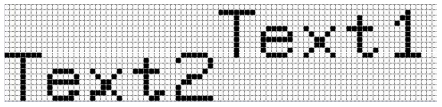
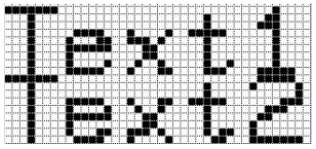
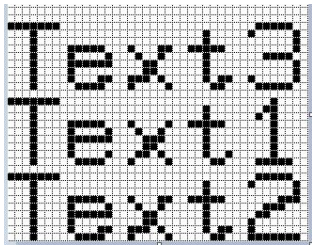
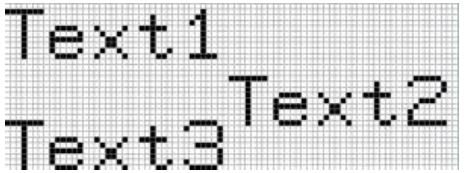
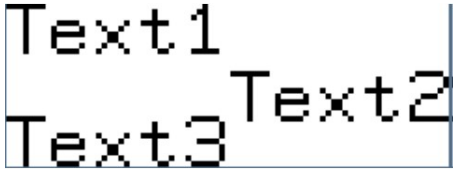
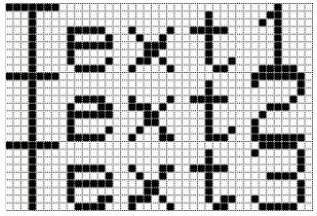
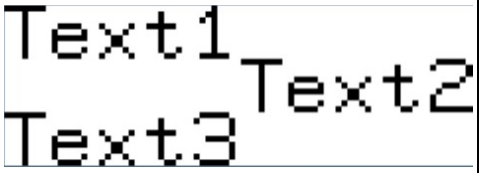
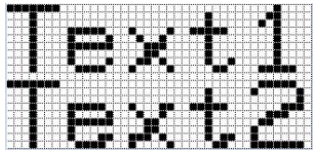
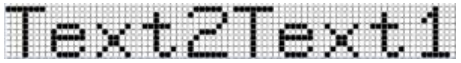
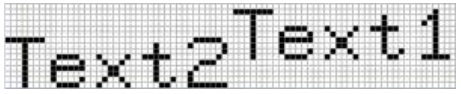
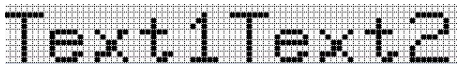
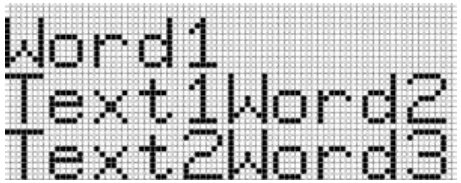
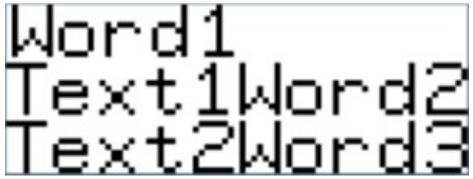


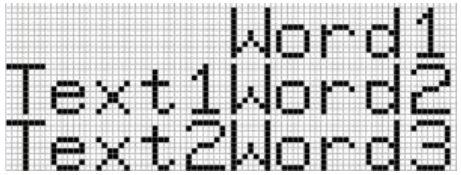
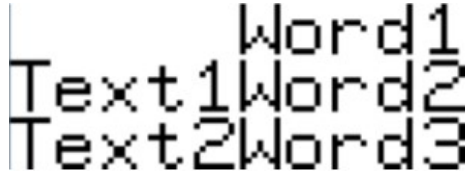
Figure A-5: Field Positioning Rules

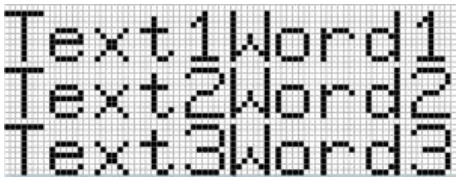
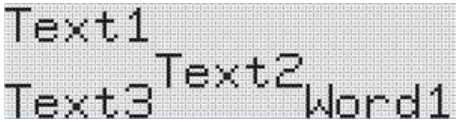
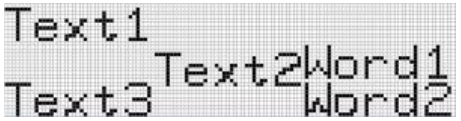
A.5.4 Field Positioning Examples

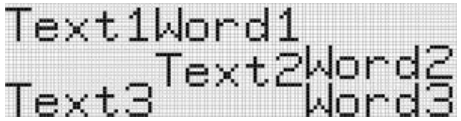
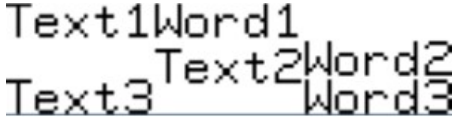
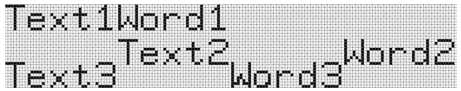
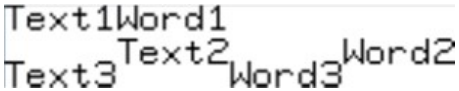
<table>
<tr><th>Description</th><th>Data</th><th>1000L Output</th><th>SIMPLICiTY Output</th></tr>
<tr><td>Single Line</td><td>
<table>
<tr><th>Font</th><th>HORC</th><th>VERC</th><th>Text</th></tr>
<tr><td>9-high</td><td>01</td><td>09</td><td>Text1</td></tr>
</table>
</td><td>  </td><td>  </td></tr>
<tr><td>Twin Line</td><td>
<table>
<tr><th>Font</th><th>HORC</th><th>VERC</th><th>Text</th></tr>
<tr><td>9-high</td><td>01</td><td>19</td><td>Text1</td></tr>
<tr><td>9-high</td><td>01</td><td>09</td><td>Text2</td></tr>
</table>
</td><td>  </td><td>  </td></tr>
<tr><td>Twin Line with overlapping fields</td><td>
<table>
<tr><th>Font</th><th>HORC</th><th>VERC</th><th>Text</th></tr>
<tr><td>9-high</td><td>01</td><td>17</td><td>Text1</td></tr>
<tr><td>9-high</td><td>01</td><td>09</td><td>Text2</td></tr>
</table>
</td><td>  </td><td>  </td></tr>
<tr><td>Twin Line with no one pixel gap</td><td>
<table>
<tr><th>Font</th><th>HORC</th><th>VERC</th><th>Text</th></tr>
<tr><td>9-high</td><td>01</td><td>18</td><td>Text1</td></tr>
<tr><td>9-high</td><td>01</td><td>09</td><td>Text2</td></tr>
</table>
</td><td>  </td><td>  </td></tr>
<tr><td>Tri-Line</td><td>
<table>
<tr><th>Font</th><th>HORC</th><th>VERC</th><th>Text</th></tr>
<tr><td>9-high</td><td>01</td><td>19</td><td>Text1</td></tr>
<tr><td>9-high</td><td>01</td><td>09</td><td>Text2</td></tr>
<tr><td>9-high</td><td>01</td><td>29</td><td>Text3</td></tr>
</table>
</td><td>  </td><td>  </td></tr>
</table>

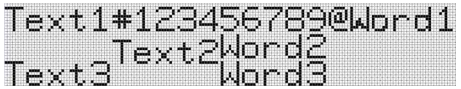
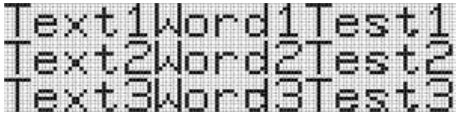
<table>
<tr><th>Description</th><th>Data</th><th>1000L Output</th><th>SIMPLICiTY Output</th></tr>
<tr><td>Tri-Line with overlapping fields</td><td>
<table>
<tr><th>Font</th><th>HORC</th><th>VERC</th><th>Text</th></tr>
<tr><td>9-high</td><td>01</td><td>29</td><td>Text1</td></tr>
<tr><td>9-high</td><td>01</td><td>17</td><td>Text2</td></tr>
<tr><td>9-high</td><td>01</td><td>09</td><td>Text3</td></tr>
</table>
</td><td> 
</td><td> 
</td></tr>
<tr><td>Tri-Line with no one pixel gap</td><td>
<table>
<tr><th>Font</th><th>HORC</th><th>VERC</th><th>Text</th></tr>
<tr><td>9-high</td><td>01</td><td>27</td><td>Text1</td></tr>
<tr><td>9-high</td><td>01</td><td>18</td><td>Text2</td></tr>
<tr><td>9-high</td><td>01</td><td>09</td><td>Text3</td></tr>
</table>
</td><td> 
</td><td> 
</td></tr>
<tr><td>Tri-Line with bottom line out of range vertical position</td><td>
<table>
<tr><th>Font</th><th>HORC</th><th>VERC</th><th>Text</th></tr>
<tr><td>9-high</td><td>01</td><td>29</td><td>Text1</td></tr>
<tr><td>9-high</td><td>01</td><td>19</td><td>Text2</td></tr>
<tr><td>9-high</td><td>01</td><td>08</td><td>Text3</td></tr>
</table>
</td><td> 
</td><td> 
</td></tr>
<tr><td>Single Line followed by Single Line</td><td>
<table>
<tr><th>Font</th><th>HORC</th><th>VERC</th><th>Text</th></tr>
<tr><td>9-high</td><td>02</td><td>25</td><td>Text1</td></tr>
<tr><td>9-high</td><td>01</td><td>25</td><td>Text2</td></tr>
</table>
</td><td> 
</td><td> 
</td></tr>
<tr><td>Single Line followed by Single Line (Different VERC)</td><td>
<table>
<tr><th>Font</th><th>HORC</th><th>VERC</th><th>Text</th></tr>
<tr><td>9-high</td><td>02</td><td>25</td><td>Text1</td></tr>
<tr><td>9-high</td><td>01</td><td>20</td><td>Text2</td></tr>
</table>
</td><td> 
</td><td> 
</td></tr>
</table>

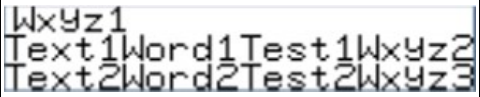
<table>
<tr><th>Description</th><th>Data</th><th>1000L Output</th><th>SIMPLICiTY Output</th></tr>
<tr><td>Single Line followed by Single Line (Big HORC Values)</td><td>
<table>
<tr><th>Font</th><th>HORC</th><th>VERC</th><th>Text</th></tr>
<tr><td>9-high</td><td>01</td><td>09</td><td>Text1</td></tr>
<tr><td>9-high</td><td>500</td><td>09</td><td>Text2</td></tr>
</table>
</td><td> 
</td><td> 
</td></tr>
<tr><td>Twin-Line followed by Tri-Line</td><td>
<table>
<tr><th>Font</th><th>HORC</th><th>VERC</th><th>Text</th></tr>
<tr><td>9-high</td><td>01</td><td>19</td><td>Text1</td></tr>
<tr><td>9-high</td><td>01</td><td>09</td><td>Text2</td></tr>
<tr><td>9-high</td><td>02</td><td>29</td><td>Word1</td></tr>
<tr><td>9-high</td><td>02</td><td>19</td><td>Word2</td></tr>
<tr><td>9-high</td><td>02</td><td>09</td><td>Word3</td></tr>
</table>
</td><td> 
</td><td> 
</td></tr>
<tr><td>Twin-Line followed by Tri-Line

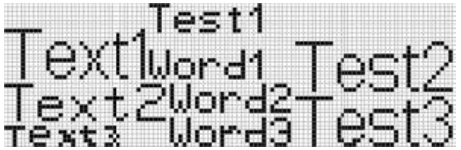
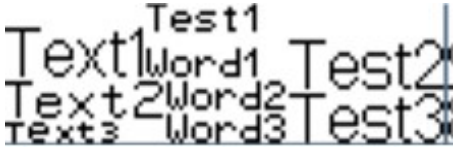
(With Space at Top of Twin Line)</td><td>
<table>
<tr><th>Font</th><th>HORC</th><th>VERC</th><th>Text</th></tr>
<tr><td>9-high</td><td>01</td><td>29</td><td><5 Space></td></tr>
<tr><td>9-high</td><td>01</td><td>19</td><td>Text1</td></tr>
<tr><td>9-high</td><td>01</td><td>09</td><td>Text2</td></tr>
<tr><td>9-high</td><td>02</td><td>29</td><td>Word1</td></tr>
<tr><td>9-high</td><td>02</td><td>19</td><td>Word2</td></tr>
<tr><td>9-high</td><td>02</td><td>09</td><td>Word3</td></tr>
</table>
</td><td> 
</td><td> 
</td></tr>
</table>

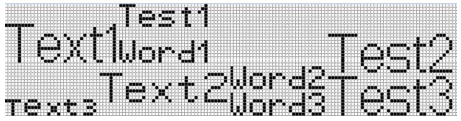
<table>
<tr><th>Description</th><th>Data</th><th>1000L Output</th><th>SIMPLICiTY Output</th></tr>
<tr><td>Tri-Line followed by Tri-Line</td><td>
<table>
<tr><th>Font</th><th>HORC</th><th>VERC</th><th>Text</th></tr>
<tr><td>9-high</td><td>01</td><td>29</td><td>Text1</td></tr>
<tr><td>9-high</td><td>01</td><td>19</td><td>Text2</td></tr>
<tr><td>9-high</td><td>01</td><td>09</td><td>Text3</td></tr>
<tr><td>9-high</td><td>02</td><td>29</td><td>Word1</td></tr>
<tr><td>9-high</td><td>02</td><td>19</td><td>Word2</td></tr>
<tr><td>9-high</td><td>02</td><td>09</td><td>Word3</td></tr>
</table>
</td><td> 
</td><td> 
</td></tr>
<tr><td>Tri-Line (Overlapping fields) followed by Signal Line</td><td>
<table>
<tr><th>Font</th><th>HORC</th><th>VERC</th><th>Text</th></tr>
<tr><td>9-high</td><td>01</td><td>29</td><td>Text1</td></tr>
<tr><td>9-high</td><td>01</td><td>17</td><td>Text2</td></tr>
<tr><td>9-high</td><td>01</td><td>09</td><td>Text3</td></tr>
<tr><td>9-high</td><td>02</td><td>09</td><td>Word1</td></tr>
</table>
</td><td> 
</td><td> 
</td></tr>
<tr><td>Tri-Line (Overlapping fields) followed by Twin Line</td><td>
<table>
<tr><th>Font</th><th>HORC</th><th>VERC</th><th>Text</th></tr>
<tr><td>9-high</td><td>01</td><td>29</td><td>Text1</td></tr>
<tr><td>9-high</td><td>01</td><td>17</td><td>Text2</td></tr>
<tr><td>9-high</td><td>01</td><td>09</td><td>Text3</td></tr>
<tr><td>9-high</td><td>02</td><td>19</td><td>Word1</td></tr>
<tr><td>9-high</td><td>02</td><td>09</td><td>Word2</td></tr>
</table>
</td><td> 
</td><td> 
</td></tr>
</table>

<table>
<tr><th>Description</th><th>Data</th><th>1000L Output</th><th>SIMPLICiTY Output</th></tr>
<tr><td>Tri-Line (Overlapping fields) followed by Tri-Line</td><td>
<table>
<tr><th>Font</th><th>HORC</th><th>VERC</th><th>Text</th></tr>
<tr><td>9-high</td><td>01</td><td>29</td><td>Text1</td></tr>
<tr><td>9-high</td><td>01</td><td>17</td><td>Text2</td></tr>
<tr><td>9-high</td><td>01</td><td>09</td><td>Text3</td></tr>
<tr><td>9-high</td><td>02</td><td>29</td><td>Word1</td></tr>
<tr><td>9-high</td><td>02</td><td>19</td><td>Word2</td></tr>
<tr><td>9-high</td><td>02</td><td>09</td><td>Word3</td></tr>
</table>
</td><td> 
</td><td> 
</td></tr>
<tr><td>Tri-Line (Overlapping fields) followed by Tri-Line (Overlapping fields)</td><td>
<table>
<tr><th>Font</th><th>HORC</th><th>VERC</th><th>Text</th></tr>
<tr><td>9-high</td><td>01</td><td>29</td><td>Text1</td></tr>
<tr><td>9-high</td><td>01</td><td>17</td><td>Text2</td></tr>
<tr><td>9-high</td><td>01</td><td>09</td><td>Text3</td></tr>
<tr><td>9-high</td><td>02</td><td>29</td><td>Word1</td></tr>
<tr><td>9-high</td><td>02</td><td>17</td><td>Word2</td></tr>
<tr><td>9-high</td><td>02</td><td>09</td><td>Word3</td></tr>
</table>
</td><td> 
</td><td> 
</td></tr>
</table>

<table>
<tr><th>Description</th><th>Data</th><th>1000L Output</th><th>SIMPLICiTY Output</th></tr>
<tr><td>Tri-Line (Overlapping fields) followed by Single Line followed by Tri-Line</td><td>
<table>
<tr><th>Font</th><th>HORC</th><th>VERC</th><th>Text</th></tr>
<tr><td>9-high</td><td>01</td><td>29</td><td>Text1</td></tr>
<tr><td>9-high</td><td>01</td><td>17</td><td>Text2</td></tr>
<tr><td>9-high</td><td>01</td><td>09</td><td>Text3</td></tr>
<tr><td>9-high</td><td>02</td><td>29</td><td>#123456789@</td></tr>
<tr><td>9-high</td><td>03</td><td>29</td><td>Word1</td></tr>
<tr><td>9-high</td><td>03</td><td>19</td><td>Word2</td></tr>
<tr><td>9-high</td><td>03</td><td>09</td><td>Word3</td></tr>
</table>
</td><td>  </td><td>  </td></tr>
<tr><td>Tri-Line followed by Tri-Line followed by Tri-Line</td><td>
<table>
<tr><th>Font</th><th>HORC</th><th>VERC</th><th>Text</th></tr>
<tr><td>9-high</td><td>01</td><td>29</td><td>Text1</td></tr>
<tr><td>9-high</td><td>01</td><td>19</td><td>Text2</td></tr>
<tr><td>9-high</td><td>01</td><td>09</td><td>Text3</td></tr>
<tr><td>9-high</td><td>02</td><td>29</td><td>Word1</td></tr>
<tr><td>9-high</td><td>02</td><td>19</td><td>Word2</td></tr>
<tr><td>9-high</td><td>02</td><td>09</td><td>Word3</td></tr>
<tr><td>9-high</td><td>03</td><td>29</td><td>Test1</td></tr>
<tr><td>9-high</td><td>03</td><td>19</td><td>Test2</td></tr>
<tr><td>9-high</td><td>03</td><td>09</td><td>Test3</td></tr>
</table>
</td><td>  </td><td>  </td></tr>
</table>

<table>
<tr><th>Description</th><th>Data</th><th>1000L Output</th><th>SIMPLICiTY Output</th></tr>
<tr><td>Tri-Line followed by Tri-Line followed by Tri-Line (With Spaces)</td><td>
<table>
<tr><th>Font</th><th>HORC</th><th>VERC</th><th>Text</th></tr>
<tr><td>9-high</td><td>01</td><td>29</td><td><5 Space></td></tr>
<tr><td>9-high</td><td>01</td><td>19</td><td><5 Space></td></tr>
<tr><td>9-high</td><td>01</td><td>09</td><td><5 Space></td></tr>
<tr><td>9-high</td><td>02</td><td>29</td><td><5 Space></td></tr>
<tr><td>9-high</td><td>02</td><td>19</td><td>Word2</td></tr>
<tr><td>9-high</td><td>02</td><td>09</td><td><5 Space></td></tr>
<tr><td>9-high</td><td>03</td><td>29</td><td><5 Space></td></tr>
<tr><td>9-high</td><td>03</td><td>19</td><td><5 Space></td></tr>
<tr><td>9-high</td><td>03</td><td>09</td><td><5 Space></td></tr>
</table>
</td><td> 
</td><td> 
</td></tr>
<tr><td>Twin-Line followed by Twin-Line followed by Twin-Line followed by Tri-Line</td><td>
<table>
<tr><th>Font</th><th>HORC</th><th>VERC</th><th>Text</th></tr>
<tr><td>9-high</td><td>01</td><td>19</td><td>Text1</td></tr>
<tr><td>9-high</td><td>01</td><td>09</td><td>Text2</td></tr>
<tr><td>9-high</td><td>02</td><td>19</td><td>Word1</td></tr>
<tr><td>9-high</td><td>02</td><td>09</td><td>Word2</td></tr>
<tr><td>9-high</td><td>03</td><td>19</td><td>Test1</td></tr>
<tr><td>9-high</td><td>03</td><td>09</td><td>Test2</td></tr>
<tr><td>9-high</td><td>04</td><td>29</td><td>Wxyz1</td></tr>
<tr><td>9-high</td><td>04</td><td>09</td><td>Wxyz2</td></tr>
<tr><td>9-high</td><td>04</td><td>09</td><td>Wxyz3</td></tr>
</table>
</td><td> 
</td><td> 
</td></tr>
</table>

<table>
<tr><th>Description</th><th>Data</th><th>1000L Output</th><th>SIMPLICiTY Output</th></tr>
<tr>
<td>Tri-Line followed by Twin-Line followed by Single Line (With multiple fonts)</td>
<td>
<table>
<tr><th>Font</th><th>HORC</th><th>VERC</th><th>Text</th></tr>
<tr><td>9-high</td><td>01</td><td>29</td><td>Text1</td></tr>
<tr><td>9-high</td><td>01</td><td>19</td><td>Text2</td></tr>
<tr><td>9-high</td><td>01</td><td>09</td><td>Text3</td></tr>
<tr><td>9-high</td><td>02</td><td>34</td><td>Word1</td></tr>
<tr><td>24-high</td><td>02</td><td>24</td><td>Word2</td></tr>
<tr><td>34-high</td><td>03</td><td>34</td><td>Test1</td></tr>
</table>
</td>
<td> 
</td>
<td> 
</td>
</tr>
<tr>
<td>Tri-Line followed by Tri-Line followed by Tri- Line (With multiple fonts)</td>
<td>
<table>
<tr><th>Font</th><th>HORC</th><th>VERC</th><th>Text</th></tr>
<tr><td>12-high</td><td>01</td><td>28</td><td>Text1</td></tr>
<tr><td>9-high</td><td>01</td><td>15</td><td>Text2</td></tr>
<tr><td>5-high</td><td>01</td><td>05</td><td>Text3</td></tr>
<tr><td>7-high</td><td>02</td><td>23</td><td>Word1</td></tr>
<tr><td>7-high</td><td>02</td><td>15</td><td>Word2</td></tr>
<tr><td>7-high</td><td>02</td><td>07</td><td>Word3</td></tr>
<tr><td>7-high</td><td>03</td><td>34</td><td>Test1</td></tr>
<tr><td>12-high</td><td>03</td><td>25</td><td>Test2</td></tr>
<tr><td>12-high</td><td>03</td><td>12</td><td>Test3</td></tr>
</table>
</td>
<td> 
</td>
<td> 
</td>
</tr>
</table>

<table>
<tr><th>Description</th><th>Data</th><th>1000L Output</th><th>SIMPLICiTY Output</th></tr>
<tr>
<td>Tri-Line (Overlapping fields) followed by Tri-Line followed by Tri- Line (With multiple fonts)</td>
<td>
<table>
<tr><th>Font</th><th>HORC</th><th>VERC</th><th>Text</th></tr>
<tr><td>12-high</td><td>01</td><td>28</td><td>Text1</td></tr>
<tr><td>9-high</td><td>01</td><td>13</td><td>Text2</td></tr>
<tr><td>5-high</td><td>01</td><td>05</td><td>Text3</td></tr>
<tr><td>7-high</td><td>02</td><td>23</td><td>Word1</td></tr>
<tr><td>7-high</td><td>02</td><td>15</td><td>Word2</td></tr>
<tr><td>7-high</td><td>02</td><td>07</td><td>Word3</td></tr>
<tr><td>7-high</td><td>03</td><td>34</td><td>Test1</td></tr>
<tr><td>12-high</td><td>03</td><td>25</td><td>Test2</td></tr>
<tr><td>12-high</td><td>03</td><td>12</td><td>Test3</td></tr>
</table>
</td>
<td> 
</td>
<td> 
</td>
</tr>
<tr>
<td>Tri-Line (Overlapping fields) followed by Tri-Line followed by Tri-Line (With multiple fonts)</td>
<td>
<table>
<tr><th>Font</th><th>HORC</th><th>VERC</th><th>Text</th></tr>
<tr><td>12-high</td><td>01</td><td>25</td><td>Text1</td></tr>
<tr><td>9-high</td><td>01</td><td>13</td><td>Text2</td></tr>
<tr><td>5-high</td><td>01</td><td>05</td><td>Text3</td></tr>
<tr><td>7-high</td><td>02</td><td>23</td><td>Word1</td></tr>
<tr><td>7-high</td><td>02</td><td>15</td><td>Word2</td></tr>
<tr><td>7-high</td><td>02</td><td>07</td><td>Word3</td></tr>
<tr><td>7-high</td><td>03</td><td>34</td><td>Test1</td></tr>
<tr><td>12-high</td><td>03</td><td>25</td><td>Test2</td></tr>
<tr><td>12-high</td><td>03</td><td>12</td><td>Test3</td></tr>
</table>
</td>
<td> 
</td>
<td> 
</td>
</tr>
</table>

A.6 Additional Examples

Get Next Message and Last Message Command Examples:

Job Data	Command Response
ABCD=Text|CDEF=13/05/2022|XYZ=17:30:16PM|LMNO=0899| QRST=123456789|GHIJ=2D-Barcode|KLMN=VJ	[STX] Text [0A]13/05/2022[0A]2D-Barcode[0A]VJ[0A]0899[0A] 123456789[0A]17:30:16PM [ETX]
1Field001=Text|1Field002=13/05/2022|1Field003=17:30:16PM| Field004=0899|Field005=123456789|Field006=2D-Barcode| Field007=VJ	[STX] Text13/05/2022217:30:16PM [0A] 08991234567892DBarcodeVJ[ETX]
Job contains the Text field as Custom reference. Custom Reference contains three inserts - Text, Text and Counter 1Field001=DefaultTextDefaultText0000	[STX] DefaultTextDefaultText0000 [ETX]

A.7 Raster Substitution and Raster Name

WSI Command Input		Printer Output (On UI)		
Raster Substitution	**Raster Received as Part of command**	**Auto Raster**	**Select By Group**	**Set Raster in Dropdown list**
ON	It is valid Raster Family Name	OFF	ON	Set Valid Family Name
ON	It is valid Raster Name	ON	X	X
ON	It is not valid Raster Family Name or Raster Name	ON	X	X
OFF	It is valid Raster Family Name	ON	X	X
OFF	It is valid Raster Name	OFF	OFF	Set Valid Raster Name
OFF	It is not valid Raster Family Name or Raster Name	ON	X	X

*X - Do not care

A.8 'U' - Update Custom Reference Command

A.8.1 User Prompted Counter Field Rules:

1. New Start Value and Old End Value should not be the same.
2. Step Size is initially updated to default value and post update of Start, End and Current Values, the requested Step Size will be updated.
3. During the update of User Prompted Counter Values, if the process fails due to any validations, the already modified Counter Values will remain updated.

   Order of processing is:

   a. Step Size with default value
   b. Start Value from the command
   c. End Value
   d. Step Size
   e. Current Value
   f. Roll Over Enable
   g. Padding Enable
      i. If Padding Enabled, then Pad Character is required
4. In negative scenarios, that is when the counter properties (like Start Value/ End Value/ Step Size) are updated after the Current Value, if Current Value is expected without actually re-loading the job, then below is applicable:
   a. Current Value received in the request will internally update the adjustment value (Counter Data Source Value) to get the Field Value as received Current Value. (ex: for Increment Counter)
      i. Start Value + Adjustment Value = Counter Field Value / Current Value.
      ii. When the Start Value is updated, the Counter Field Value/ Current Value is adjusted automatically with respect to the updated Start Value and Adjustment Value.
   b. In negative scenarios, expectation of the Counter Value is not defined, but it is made sure that there are no fatal errors.
5. Start, End and Current Values, all should be of same character (Alpha upper case or Alpha lower case or Numeric).
6. Start and End Value validation depends of Increment/Decrement Counter.

A.8.2 WSI User Field Counter Setup Example

To use the WSI command to update a counter, create the counter as a custom reference field as a user prompted counter.

1. Navigate to *Tools* > *Custom Reference Builder* and touch the “+” button to create a new custom reference.

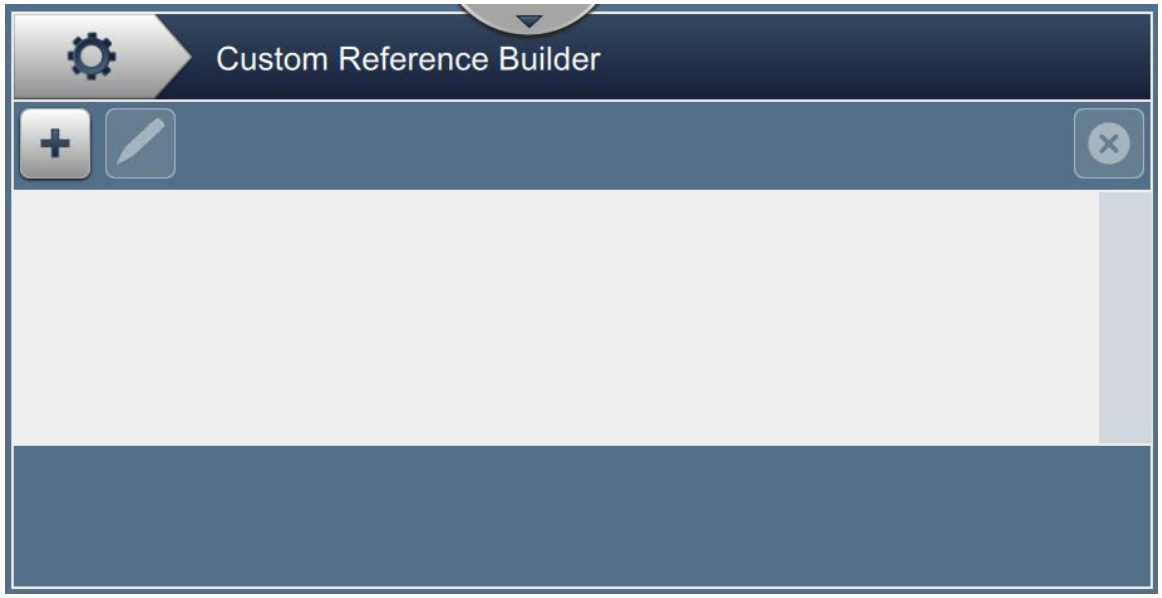


Figure A-6: Custom Reference Builder Screen

2. Enter the reference name using utility keypad and touch the *Accept* button. In this example, “COUNTER2” is created as custom reference.

3. Select option *User Prompted Counter* and touch “+”. The User Prompted Counter gets added to the custom reference builder list.

Figure A-7: Custom Reference Builder List

4. To edit the counter setup, select option *User Prompted Counter* from the custom reference builder list and touch *Edit (Pencil)* button.

5. Touch the *Option* and ensure that the length of the counter is same as required length.

   Note: *If the Number Of Digits is less than required, sending command using WSI may fail.*

Figure A-8: User Prompted Counter Options

6. Touch the *Save* button to save the custom job.

To create a job with Custom Reference Builder field (COUNTER2):

7. Navigate to *Jobs* and touch the “+” button to create a new job.
8. Touch the Job Name text box to enter the job name for the new job and enter the required Job Name using utility keypad and touch the *Accept* button. In this example, job “C” is created.
9. Touch the *Done* button on the Job Settings screen.

10. Touch the “+” button and select the *Text* option to add Text field. Ensure the Text field type is selected and touch the *OK* button.
11. Select *Custom Reference* from the Text Type drop down list and touch the *OK* button.
12. Select *COUNTER2* from the Custom Field Reference drop down list and touch the *OK* button.

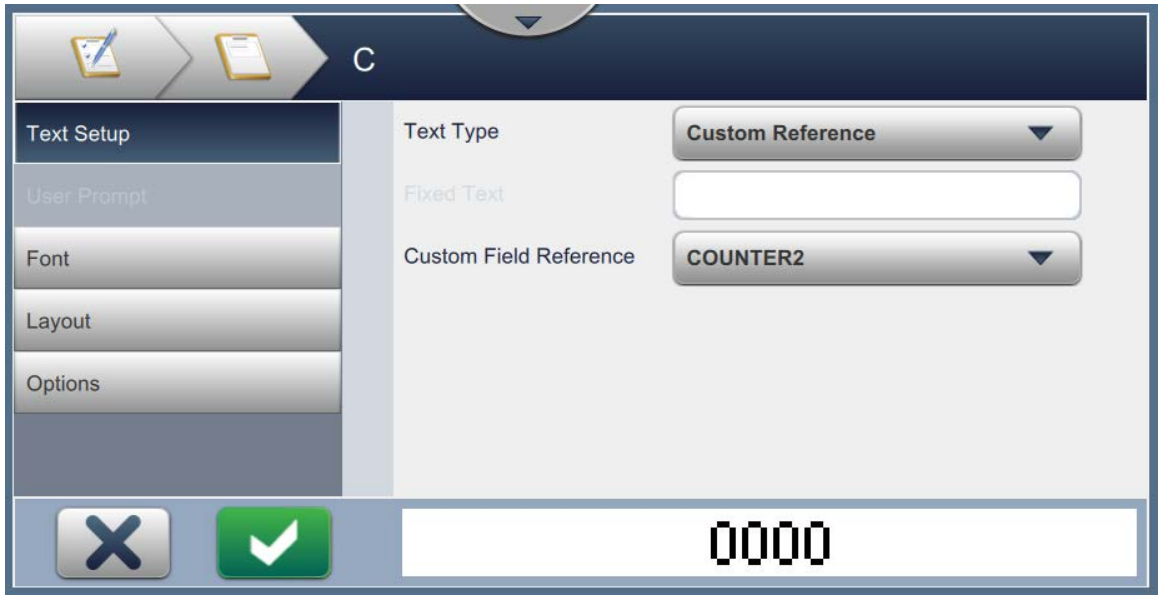


Figure A-9: Job Custom Reference

13. Touch the green Check button to save the required format of the field and touch the *Save* button.

14. Touch the *Run Job* button. The Run Job screen opens.

Figure A-10: Run Job Screen

15. Touch the *Approve All* button and then touch *Continue* button and *Load Job* button to the load job.

16. Send the U Command for the setup of the counter.

[02]UCOUNTER2[0A]00001[0A]00001[0A]00020[0A]1[0A]1[0A]0[0A][0A][03]