RFID Application Specification

Global Common

SD-1033

ISSUED December 2013
REVISED February 23, 2017

© 2017 Nexteer Automotive

All rights reserved.

Table of Contents

1. Scope and Purpose 4
1.1 Scope 4
1.2 Purpose and Objectives 5
1.3 General System Information and Requirements 5
2. Requirements (associated routine name) 6
2.1 BNI Master Routine (R14_BNI_Master) 6
2.2 RFID Control Routine (R15a_RFID_Px_Cntrl) 9
2.3 RFID Communication Routine (R15b_RFID_Px_Comm) 16
3. RFID Tag Structure 17
3.1 Pallet RFID 17
3.2 Part RFID 19
A. Pallet RFID Memory Map 22
B. Part RFID Memory Map 23

List of Figures

Figure 1: 4 IO-Link Master Configuration Logic 6
Figure 2: 8 IO-Link Master Configuration Logic 7
Figure 3: 4 IO-Link BNI Master Input Buffering CPS Instruction 7
Figure 4: Example - BNI IO-Link Port Configuration 8
Figure 5: Example – BNI Master Module Connection Fault 8
Figure 6: Example - BNI IO-Link Port Connection Faults 8
Figure 7: IO-Link RFID Control AOI 9
Figure 8: RFID Function Start Addresses and Lengths 10
Figure 9: RFID Control Faults (Example) 11
Figure 10: Example – Process Complete Tags 12
Figure 11: RFID Data EQU Instructions 12
Figure 12: RFID Device Not Busy OTE Instruction 12
Figure 13: RFID Fill Request Instructions 13
Figure 14: Example – RFID Read Header OTL Instruction 13
Figure 15: Example – RFID Read Data OTL Instruction 14
Figure 16: Example – RFID Write Header OTL Instruction 14
Figure 17: Example – RFID Write Results 1 OTL Instruction 14
Figure 18: Example – RFID Write Results 2 OTL Instruction 15
Figure 19: Pallet Handling Standard Process Flowchart 18
Figure 20: Standard Part RFID Process Flowchart 20
Figure 21: Pallet RFID Memory Map Requirement 22
Figure 22: Part RFID Memory Map Example 23

1. Scope and Purpose

1.1 Scope

1.1.1 RFID applications shall be treated as a subset of product traceability. Two RFID applications are identified in this specification:

- *Pallet RFID applications* are identified as RFID tags mounted to pallets and are used in asynchronous assembly operations for pallet identification.
- *Part RFID applications* are identified as RFID tags mounted on or otherwise connected to the part independent of the part transfer methodology.

Note: It is permissible to have both Pallet RFID and Part RFID applications in the same logic program.

1.1.2 This specification has associated logic files and HMI templates. Nexteer files and templates reflect the requirements of this specification. In addition, they provide logic routines, screens, and examples that may be applied to new equipment designs. These logic files and HMI templates are available at www.nexteerdataexchange.com.

1.1.3 The use of the word "shall" indicates requirements and the use of the word "should" indicates recommendations. The use of the word "may" indicates permission or allowance and the use of the word "can" indicates a possibility.

1.1.4 This specification is structured as follows:

1. Requirements and guidance applying the Nexteer logic files are detailed in Section 2.
2. The RFID tag data structure is detailed in Section 3.
3. Supporting references and documentation is detailed in the Annex sections.

1.2 Purpose and Objectives

1.2.1 The purpose of this specification is to provide Nexteer requirements and guidance to Original Equipment Manufacturers (OEM) for use in the design of PLC logic.

1.2.2 The objective of this specification is to provide common, maintainable, and cost effective RFID systems that enhance both the productivity and ease-of-use of the system, plus ensure the quality of Nexteer products produced. The application of this specification will results in common RFID systems that:

1. Ensures the machine processes the part correctly.

   To correctly process the part, the machine logic design needs to include significant consideration for the control functions in the event of failure such that the machine is not capable of processing the part incorrectly.

2. Provides ease of customer use.

   Ease of customer use relates to logic that provides plant personnel a quick understanding as to how the machine processes the part, logic that can be quickly used to troubleshoot failures, and logic that can be easily used to verify part quality. Nexteer's logic provides common structure and naming conventions for plant production, independent of which OEM supplied the equipment.

3. Facilitates the OEM design and Nexteer logic approval process.

   Nexteer's specifications require logic / software approval prior to MQ1. Nexteer's approval process, adherence to this specification, and use of Nexteer's logic, provides an opportunity for the OEM to demonstrate compliance to the requirements.

1.3 General System Information and Requirements

1.3.1 The RFID system interfaces with developed logic routines by including the required logic routines listed below. These routines are available in the Nexteer_Library logic file and meet the communication tag naming and structure described in this specification.

- R14_BNI_Master
- R15a_RFID_Px_Cntrl
- R15b_RFID_Px_Comm

1.3.2 The routines are required to be modified by the OEM to meet the required design needs of each application.

1.3.3 The OEM is expected to replace all PLC tags that begin with "REPLACE" with the appropriate PLC tag and data. For example: REPLACE_ReadHeader_StartAddress.

2. Requirements (associated routine name)

The following Routine requirements are based upon Nexteer's logic files for PLC based control systems.

2.1 BNI Master Routine (R14_BNI_Master)

The BNI_Master routine establishes the communication and setup of the Balluff BNI IO-Link Network Interface Module. Multiple BNI_Master routines may be required in a PLC logic file depending on the requirements of the system. This routine works in conjunction with the following routines:

- R15a_RFID_Px_Cntrl
- R15b_RFID_Px_Comm

Requirements:

2.1.1 The BNI_Master routine shall be present within the "MainProgram" under the Main Task of the Controller.

1. Single Station applications may have multiple BNI modules; however, modifications to the BNI_Master routine are required in order to configure additional modules into the same routine.
2. Multiple Station equipment (such as asynchronous assembly lines) may contain multiple BNI modules; however, the BNI_Master routines shall be located in the program dedicated for the Station and appropriately named to reflect the physical location of the device.

2.1.2 The Main routine shall call BNI_Master unconditionally through a JSR instruction.

2.1.3 The logic is preconfigured for an 8 port BNI Master module containing both 4 and 8 IO-Link Ports. Associated AOI instruction and configuration logic shall be removed depending on the number of IO-Link Ports contained in the Master Module.

Figure 1: 4 IO-Link Master Configuration Logic

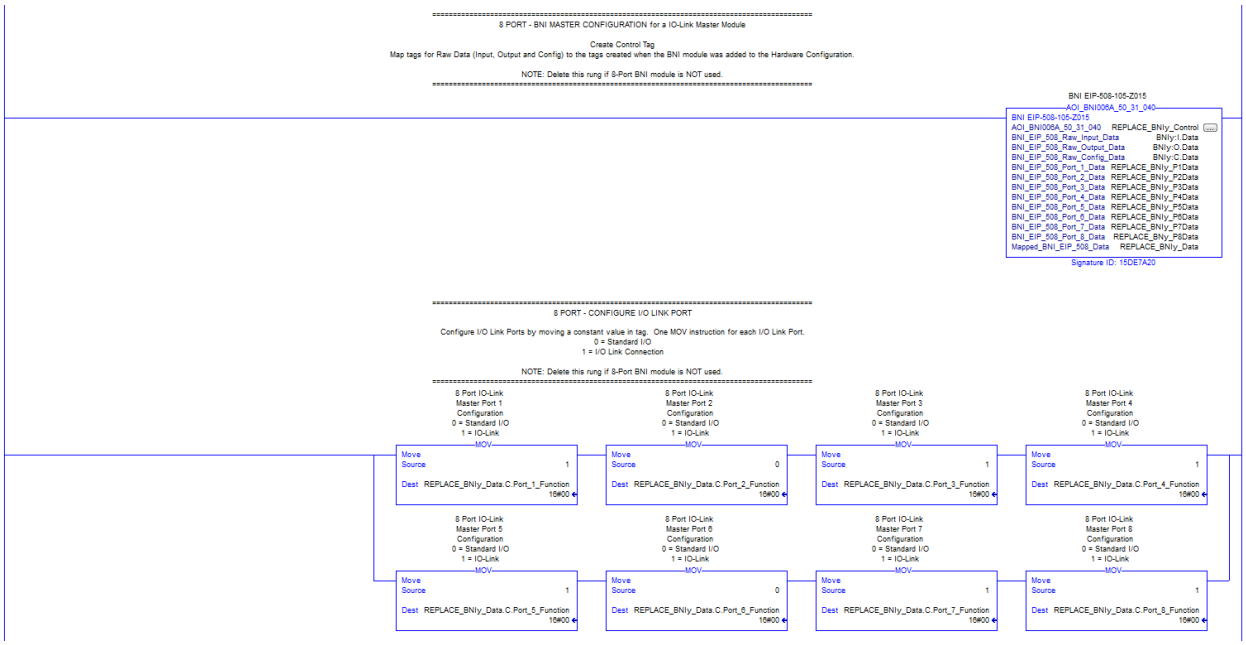

Figure 2: 8 IO-Link Master Configuration Logic

2.1.4 The AOI inputs "BNI_EIP_50x_Raw_Input_Data," "BNI_EIP_50x_Raw_Output_Date," and "BNI_EIP_50x_Raw_Config_Data" have associated tags that are set by the device property "Name" in the I/O Configuration in the Controller Organizer. The device name shall be updated to reflect an appropriate description of the device.

Note: Depending on which BNI Master Module is used (4 vs. 8 IO-Link Port configuration), the "x" is either "2" or "8" respectively.

Note: It is preferred that the device name in the I/O Configuration match the electrical schematics.

2.1.5 The remaining AOI inputs shall be renamed appropriately to match the device name to avoid confusion.

2.1.6 The CPS instruction provided in the logic is set up to buffers the available discrete inputs on a 4 IO-Link Master (physical ports 0, 1, 4, and 5). This CPS instruction shall be relocated to the Main routine. The tag IBNIx_REPLACE shall be updated to reflect an appropriate description of the device as described in Section 2.1.4.

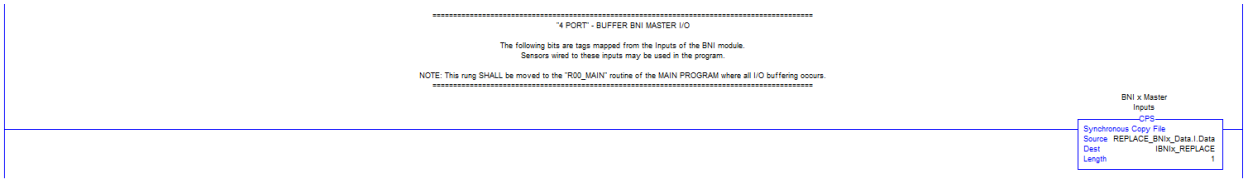

Figure 3: 4 IO-Link BNI Master Input Buffering CPS Instruction

2.1.7 The MOV instruction logic shall be modified to enable the IO-Link ports based on the application.

1. Sourcing “0” configures the IO-Link Port as discrete I/O (two discrete inputs are available).

   Note: Configuring any BNI IO-Link as an output is prohibited.

2. Sourcing “1” configures the IO-Link Port as an IO-Link capable network.

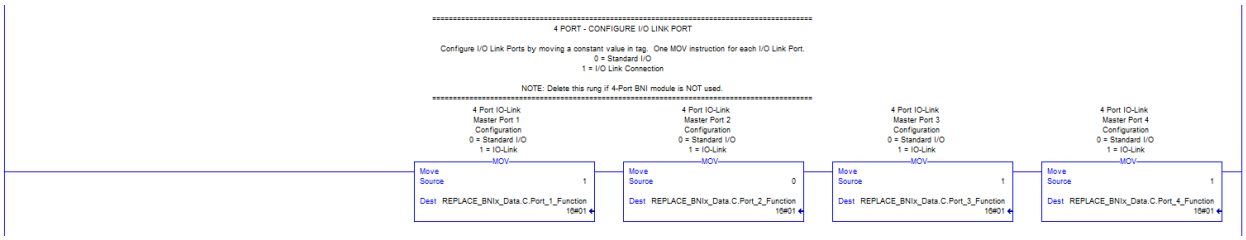

Figure 4: Example - BNI IO-Link Port Configuration

2.1.8 The BNI_Master routine has preconfigured faults that shall be moved to the appropriate Fault routine. These faults are typically immediate stop faults.

1. Fault_ImmedStop[1].7 indicates when communication to the BNI Master Module has been lost.

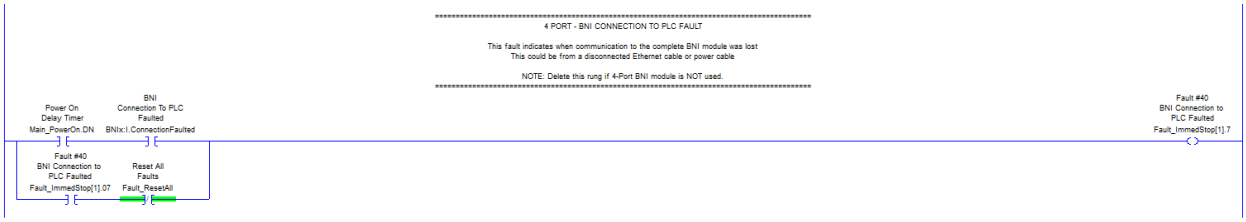

Figure 5: Example – BNI Master Module Connection Fault

2. Fault_ImmedStop[1].(8-11 or 8-15) are used to indicate when an IO-Link device has been disconnected from the BNI Master module.

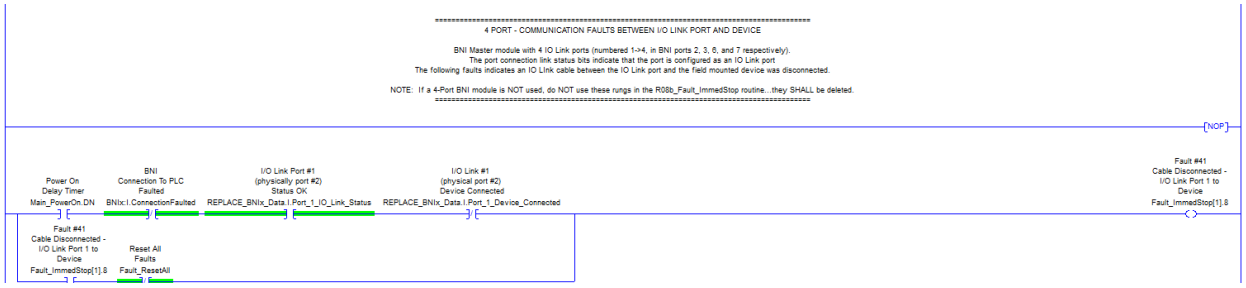

Figure 6: Example - BNI IO-Link Port Connection Faults

Note: Immediate Stop Faults have been preconfigured for both a 4 or 8 IO-Link Master Module. The IO-Link Master configuration not used shall be deleted from the logic.

2.2 RFID Control Routine (R15a_RFID_Px_Cntrl)

The RFID_Px_Cntrl routine contains the bulk of OEM programming involvement. This routine is used to configure all addressing, length of data, and communication to the RFID system triggers for a particular RFID device connected to a port on the BNI Master module. The RFID_Px_Cntrl routine works in conjunction with the R14_BNI_Master, R15b_RFID_Px_Comm, and R16_PartData or R18_Pallet routines.

Requirements:

2.2.1 The RFID_Px_Cntrl routine shall be present within the "MainProgram" under the Main Task of the Controller.

Note: Applications containing multiple BNI Master Modules (such as asynchronous assembly lines) shall have the corresponding RFID_Px_Cntrl routine(s) located in the program(s) dedicated for the Station.

2.2.2 The Main routine shall call the RFID_Px_Cntrl routine unconditionally through a JSR instruction.

2.2.3 An RFID_Px_Cntrl routine shall be provided for each IO-Link RFID device connected to a BNI Master module port. The "Px" shall be updated to correspond with the connecting port.

2.2.4 The RFID_Px_Cntrl routine's RFID Antenna Control Tag AOI instruction facilitates the communication to and from the BNI Master Module and the RFID device.

1. The AOI input "IO_Link_Port_Data" shall be updated to coordinate with the physical port to which the RFID device is connected. Refer to the BNI_Master AOI instruction for the proper port configurations (Section 2.1).
2. The AOI input "Mapped_BISMIOL_Data" contains the tag name to which data is sent / received from the RFID device. This tag is used in the corresponding RFID_Px_Comm routine to transfer data to and from the RFID device utilizing indirect addressing.

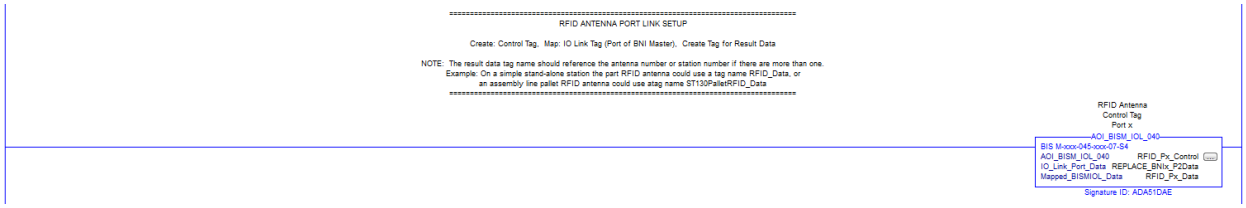

Figure 7: IO-Link RFID Control AOI

2.2.5 The tag "RFID_Px" shall be updated to reflect the BNI Master Module port to which the device is connected. This update facilitates indirect addressing functions in the RFID_Px_Comm routine.

2.2.6 There are six functions available for each RFID device. Depending on the application, more than one function may be required. The OEM shall assign the appropriate start address (*_StartAddress) and length (*_NoOfBytes) for each function implemented.

1. Read Header
2. Read Data
3. Manual Read (HMI input)
4. Write Header
5. Write Results 1
6. Write Results 2

Note: Write Results 1 and 2 are required when multiple locations must be updated in the RFID tag. Programming shall only allow Operation / Station specific data to be written to the RFID tag. Refer to the RFID Memory Map for specifics on Header and Data locations.

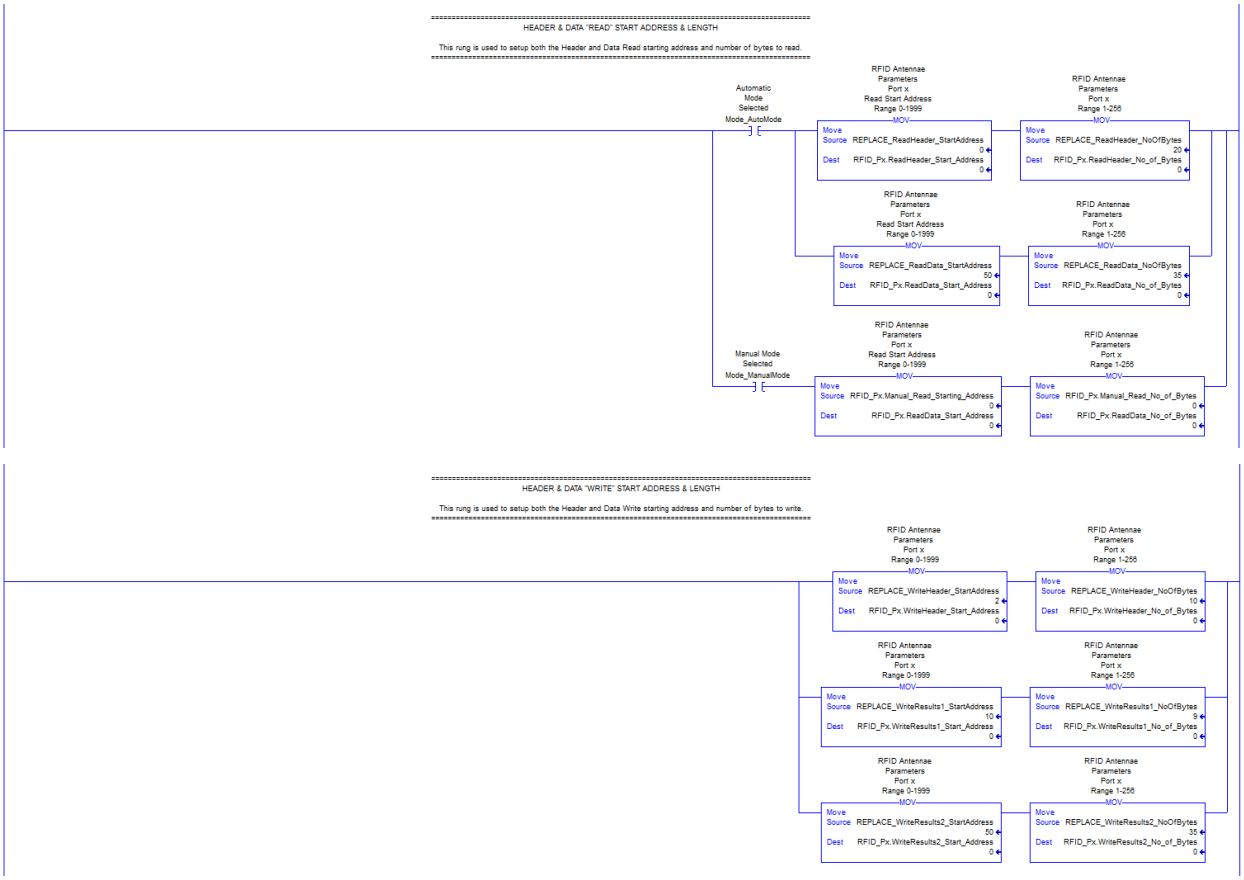

Figure 8: RFID Function Start Addresses and Lengths

2.2.7 The RFID_Px_Cntrl routine has preconfigured fault detection and messages that shall be moved to the appropriate Fault routine and correct fault array coils assigned. Fault messages include the following:

1. RFID Data Carrier (Tag) Not Present
2. RFID Data Read Error - Data Mismatch
3. RFID Data Read Error – Tag Removed
4. RFID Data Write Error – Data Mismatch
5. RFID Data Write Error – Tag Removed
6. RFID Data Write Error – Data Addressing Error
7. RFID Antenna Error – Device Not Recognized

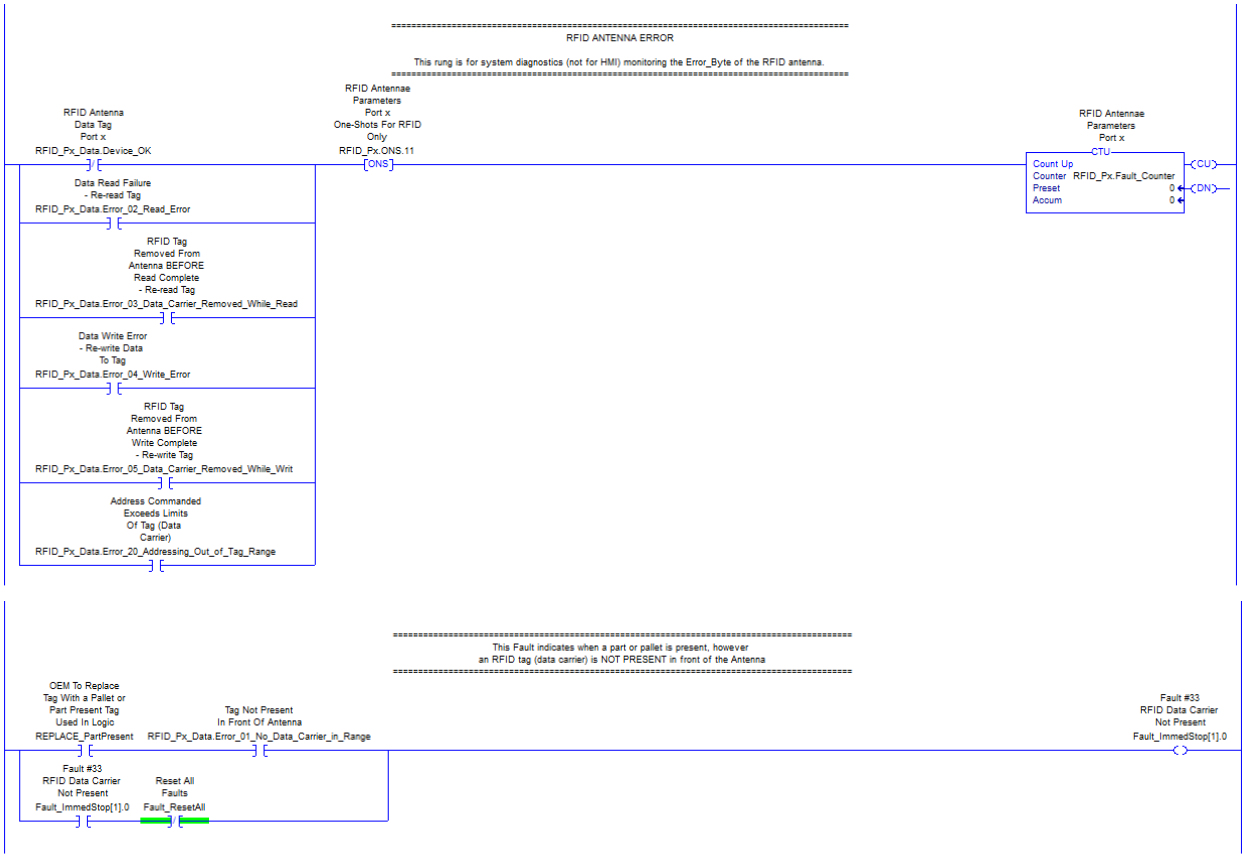

Figure 9: RFID Control Faults (Example)

Guidance:

2.2.8 During the read / write process, it is necessary to monitor the completion of each process. Cycle complete contacts have been preconfigured in the logic for each predetermined read and write process. These inputs should be included in the logic as needed.

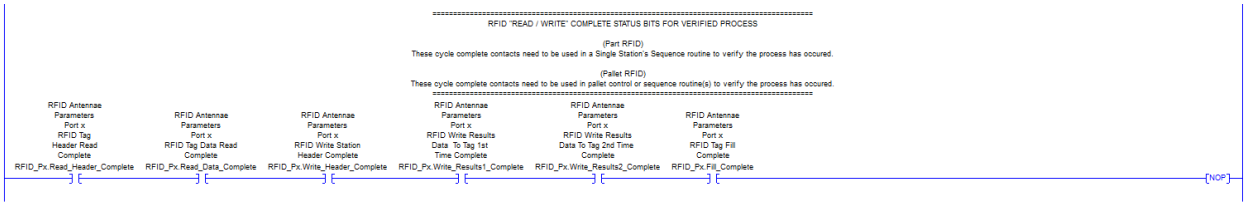

Figure 10: Example – Process Complete Tags

2.2.9 EQU instructions have been added to the logic that perform no function. These instructions are used to display the tag names that contain the read and write data arrays. Further programming by the OEM is required for extracting the data into a usable format to and from the arrays. The routines R16_PartData and R18_Pallet have been established to assist in this task.

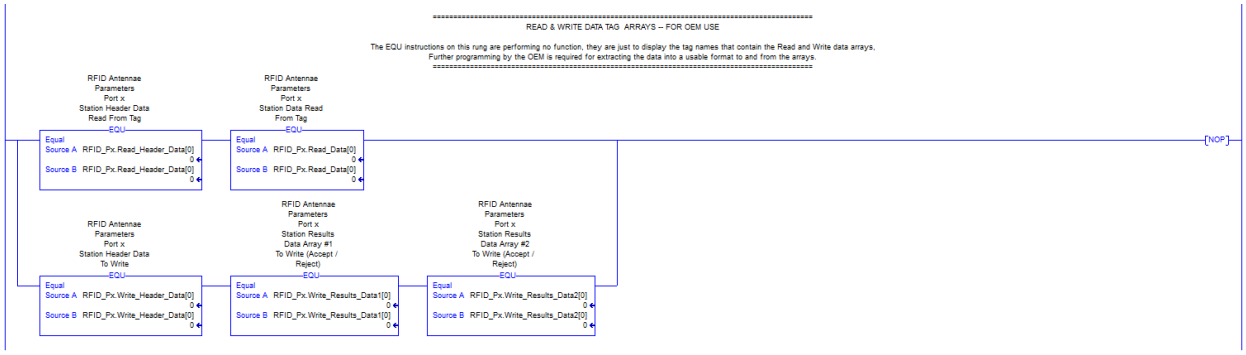

Figure 11: RFID Data EQU Instructions

2.2.10 An RFID device can only perform a single read or write function at a single time. Multiple read / write functions must be programmed so they do not overlap in the sequence. The *RFID_Head_Not_Busy tag is intended to help ensure a new request cannot be performed while the RFID processor is executing a function.

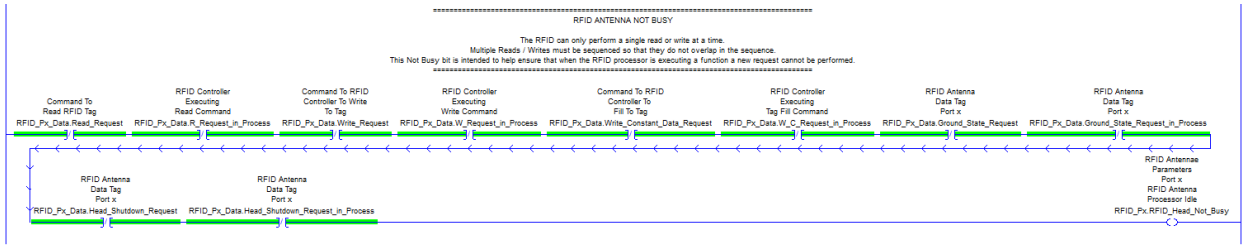

Figure 12: RFID Device Not Busy OTE Instruction

2.2.11 On a "Lead-Off Operation," it may be required to clear all data from the RFID tag. In order to perform this operation, it is necessary to assign a starting address and length, as well as enable the *Fill_Request OTL Instruction.

Note: The "Lead-Off Operation" is defined as the first Operation in a Cell or Value Stream. This is typically identified as the Operation when an RFID tag is initially placed on the product and must be cleared in order to ensure the tag is empty.

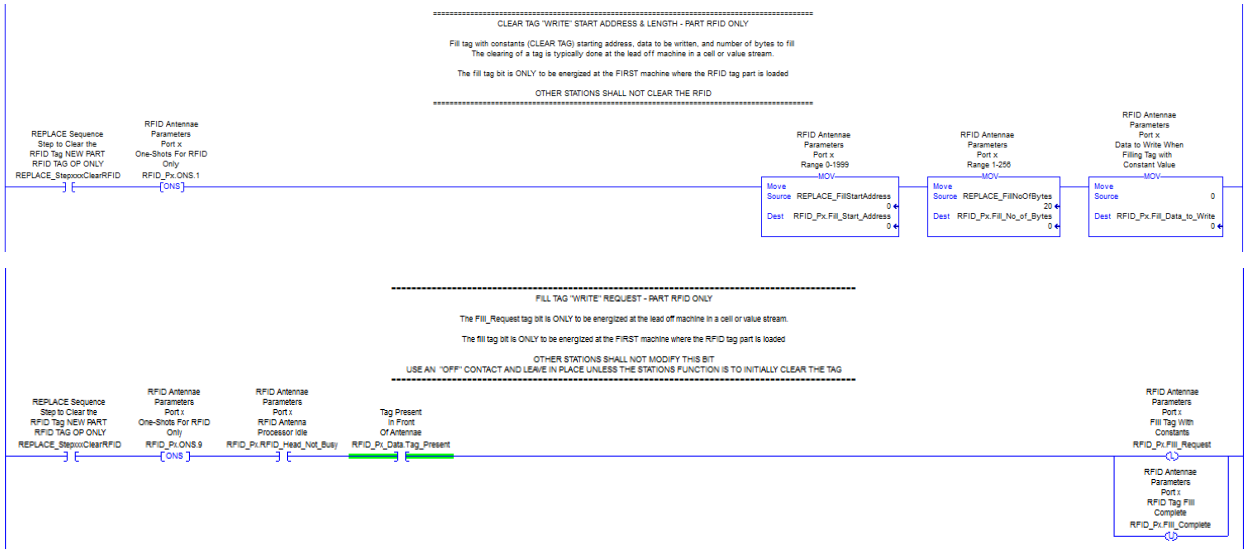

Figure 13: RFID Fill Request Instructions

2.2.12 The following tags result in a request to perform a particular RFID function. Each function has associated tags for *StartAddress and *NoOfBytes that must be appropriately assigned. Refer to the Memory Map for specific starting addresses and length (Section 2.2.6).

1. Read_Header_Request triggers a request to read the Header information as assigned by the starting address and number of bytes.

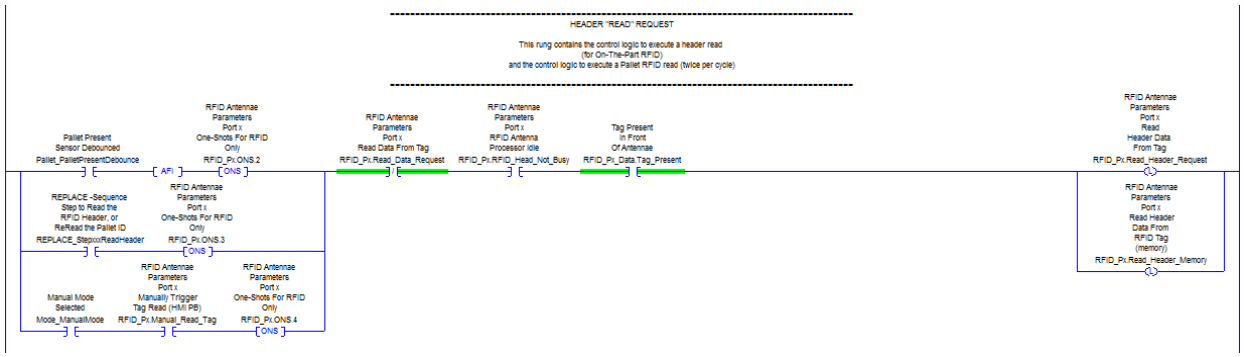

Figure 14: Example – RFID Read Header OTL Instruction

2. *RFID_ReadData triggers a request to read the Data information as assigned by the starting address and number of bytes.

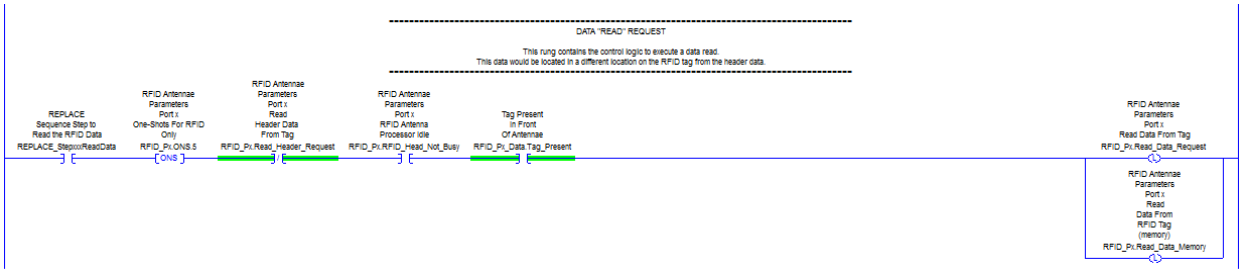

Figure 15: Example – RFID Read Data OTL Instruction

3. *RFID_Write_Header triggers a request to write the Header information as assigned by the starting address and number of bytes.

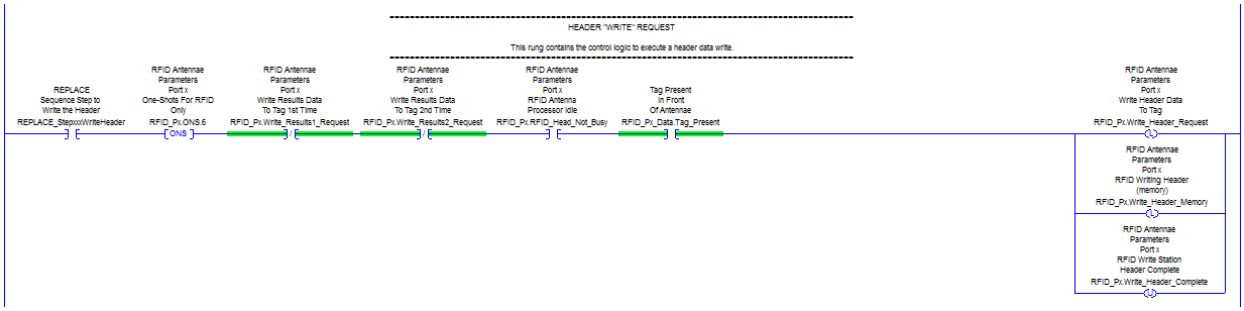

Figure 16: Example – RFID Write Header OTL Instruction

4. *RFID_WriteResults1 triggers a request to write the Results 1 information as assigned by the starting address and number of bytes.

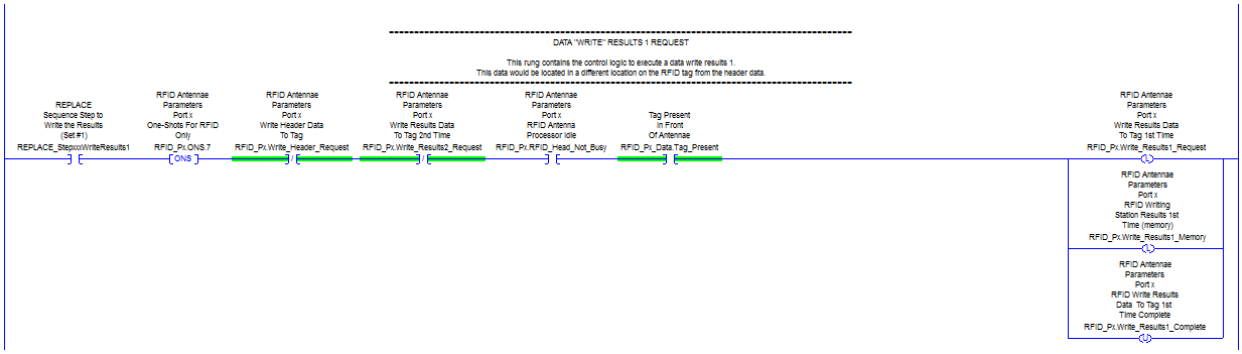

Figure 17: Example – RFID Write Results 1 OTL Instruction

5. *RFID_WriteResults2 triggers a request to write the Results 2 information as assigned by the starting address and number of bytes.

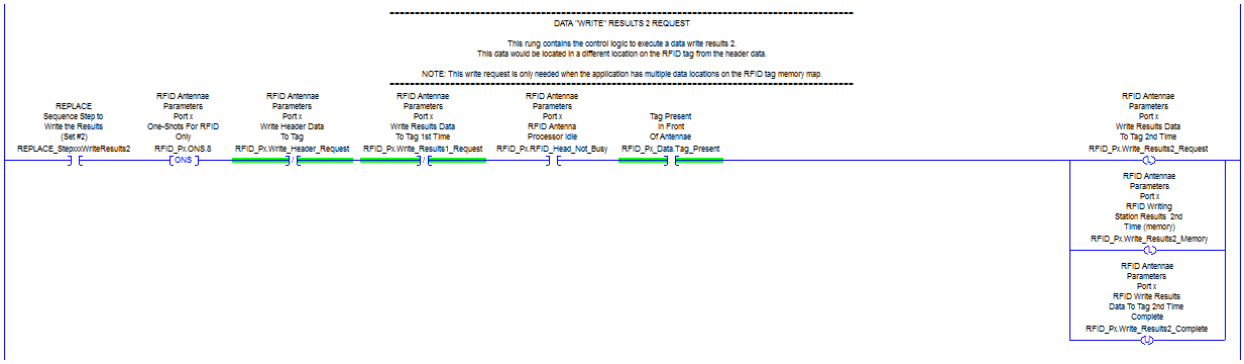

Figure 18: Example – RFID Write Results 2 OTL Instruction

Note: Write instructions are limited to 256 bytes of information per transaction; therefore, if more data is required, a subsequent RFID operation should be implemented.

2.3 RFID Communication Routine (R15b_RFID_Px_Comm)

This routine requires no modifications. The RFID_Comm routine contains the communication handshaking and data manipulation of the RFID device. The RFID_Comm routine works in conjunction with the R14_BNI_Master, R15a_RFID_Px_Cntrl, and R16_PartData or R18_Pallet routines.

Requirements:

2.3.1 The RFID_Px_Comm routine shall be present within the "MainProgram" under the Main Task of the Controller.

Note: Applications containing multiple BNI Master Modules (such as asynchronous assembly lines) shall have the corresponding RFID_Px_Comm routine(s) located in the program(s) dedicated for the Station.

2.3.2 The Main routine shall call the RFID_Px_Comm routine unconditionally through a JSR instruction.

2.3.3 An RFID_Px_Comm routine shall be provided for each IO-Link RFID device connected to a BNI Master Module port. The "Px" shall be updated to correspond with the connecting port and corresponding *RFID_Px_Cntrl routine.

2.3.4 The RFID_Px tag shall be updated to the corresponding tag to which the RFID_Px_Comm routine is associated.

3. RFID Tag Structure

3.1 Pallet RFID

Pallet RFID refers to applications that include RFID tags mounted to pallets for use in asynchronous assembly operations for pallet identification. The required Pallet RFID Memory Map is shown in Annex A of this document.

3.1.1 Pallet ID

1. The Pallet ID shall be assigned and contain a uniquely number code. The Pallet ID is identified by two bytes of information. An example of a Pallet ID is 0x0117.
   a. Byte 1: In this example, Line Number #1 (0x01) is represented.
   b. Byte 2: In this example, Pallet Number #23 (0x17) is represented.
2. Once the Pallet ID is assigned, it shall not be altered.
3. Because the Line Number (byte 1 of the Pallet ID) may not be required to identify the pallet, the Pallet Number may be represented in a single byte of information (byte 2 of the Pallet ID) in the Pallet UDT Array. See the associated Nexteer logic for an example Pallet UDT Array.

3.1.2 The following flowchart identifies the standard process flow for Pallet RFID applications.

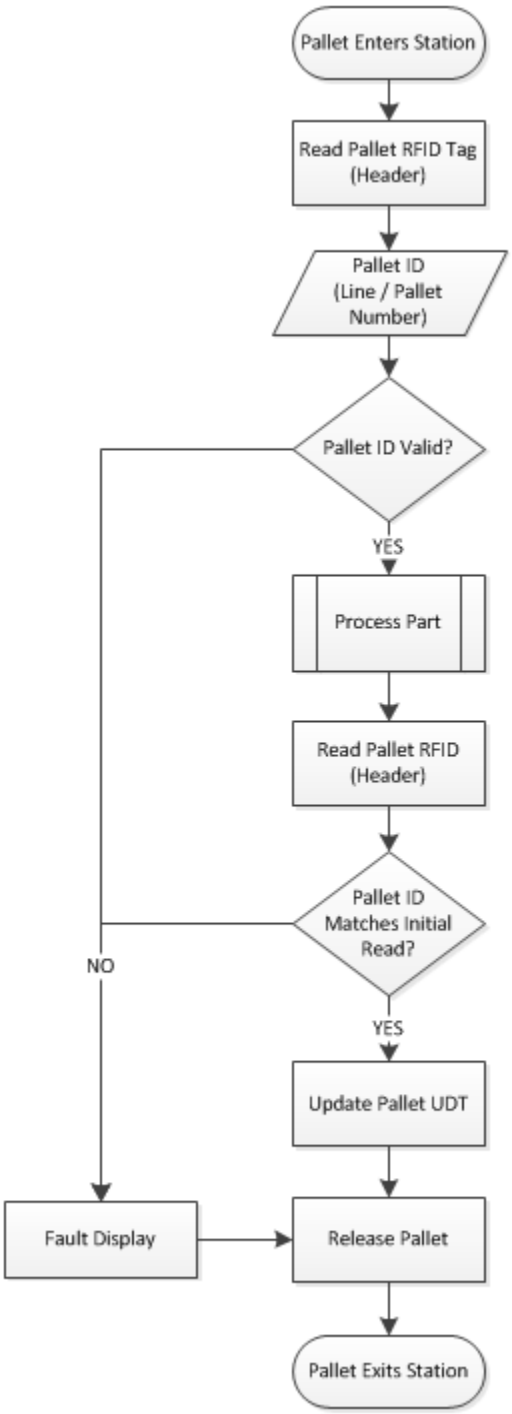

Figure 19: Pallet Handling Standard Process Flowchart

3.2 Part RFID

Part RFID refers to applications that include RFID tags mounted on or otherwise connected to the part independent of the part transfer methodology. The required minimal Header information and example Results Data are shown in Annex B of this document.

3.2.1 Product Serial Number

1. The Product Serial Number shall be applied to the RFID tag by the lead-off Operation. Refer to SD-1034 for Serial Number Generation guidance. The Product Serial Number shall not to be altered after the lead-off Operation at any time.

3.2.2 Last Assembly Operation Number

1. The Last Assembly Operation Number shall be updated at the each processing Operation in conjunction with the Part Status prior to the part being released (for example: prior to unclamp). When assembly of the part is performed on an asynchronous assembly line or dial table, the Last Assembly Operation Number is updated at the Un/Load Station(s).
2. Each Operation Number is two bytes in length. An example of an Operation Number is 0x00F0 (Operation #240).

3.2.3 Part Status

1. The Part Status shall be updated at each processing Station in conjunction with the Last Assembly Operation Number.
2. The Part Status is two bytes in length and identifies the current status of the part being processed.
   a. A good part status code is represented as 9999 (0x270F).
   b. An in-process part status code is represented as 9000 (0x2328). Implementation of in-process part status code shall be included for asynchronous assembly line or dial table assembly station systems.
   c. A reject part status is Operation specific starting with 1xxx. An example of a reject part status is 1025 (0x0401). Reject part statuses shall be unique for each Operation within an Assembly Operation.

3.2.4 Results Data stored on the Part RFID tag are Project specific. Therefore, an example Part RFID Memory Map is available in Annex B of this document that indicates how a Part RFID Memory Map could be identified. The specific Results Data shown in Annex B is not a requirement; its intent is for reference / example only.

3.2.5 The following flowchart identifies the standard process flow for Part RFID applications.

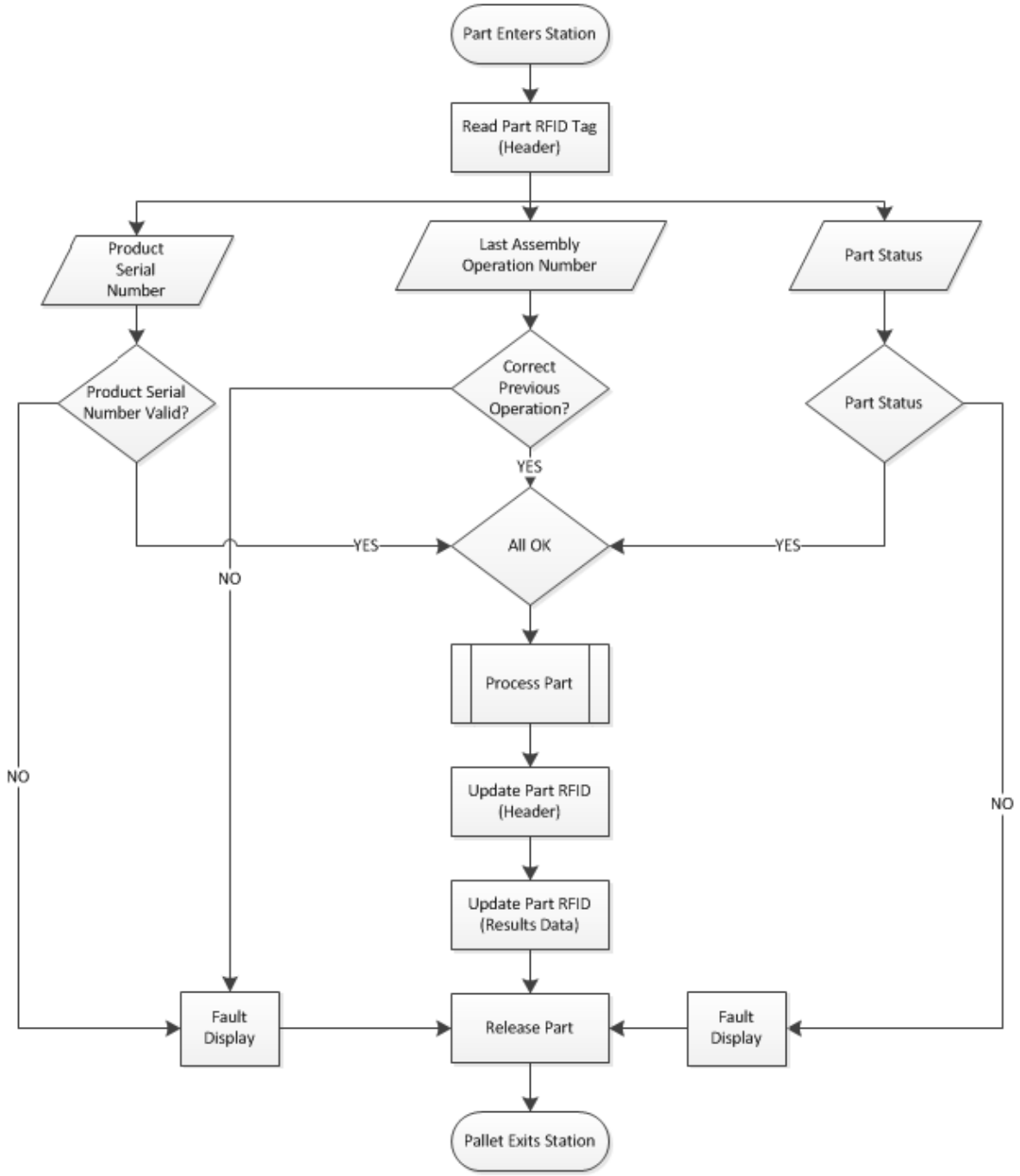

Figure 20: Standard Part RFID Process Flowchart

3.2.6 The standard process flow for parts entering and exiting the Operation shall be detailed in the Sequence routine:

1. Part Presence – as the part enters the Operation, the Sequence shall detect the part and prepare to read the RFID tag.
2. Read Part RFID – once part presence is establish, the Sequence routine shall trigger a read Header request in order to read the Part RFID tag.
3. The following minimal information shall be extracted from the RFID Tag:
   a. Product Serial Number
   b. Last Assembly Operation Number
   c. Part Status
4. Based on the information extracted, a determination of part acceptance is necessary.
5. Upon acceptance, the part is to be processed.
6. After the part has been processed and prior to the part being released (for example: prior to unclamp), the Part RFID tag is to be updated with the Last Assembly Operation Number and Part Status. Updates may be required for Results Data conditional upon the information to be collected on the RFID tag.

A. Pallet RFID Memory Map

PALLET RFID MEMORY MAP

	Description	Address	Length	Comment	
HEADER	*** RESERVED ***	0	10		HEADER
	Pallet ID	10	2	Assigned by the OEM. Consists of Line Number and Pallet Number.	
	Line #	10		Line Number represented by 1 byte. Example: Line #1 (0x01).	
	Pallet #	11		Pallet Number represeted by 1 byte. Example: Pallet #23 (0x17).	
RESULTS DATA					RESULTS DATA
	*** SPARE ***	12	1988	*** SPARE ***	
	TOTAL BYTES		2000		

*Reference MM Details Tab for a detailed explanation of RFID Memory Map.

Figure 21: Pallet RFID Memory Map Requirement

B. Part RFID Memory Map

PART RFID MEMORY MAP

	Description	Address	Length	Comment	
HEADER	*** RESERVED ***	0	10		HEADER
	Housing Serial Number (applied initially to housing)	10	20	Example: 38207248163102483500 (pppppppppYYJJssssssUU)	
	Last Assembly Operation Number (Reference Operation Number Definitions Tab)	30	2	Updated at each Operation.	
	Part Status 9999 (0x270F) = Good Part 9000 (0x2327) = In Process 1xxx (0x....) = Reject Part (Operation specific reject code)	32	2	Updated at each Operation.	
	*** RESERVED ***	34	2		
RESULTS DATA	*** SPARE ***	36	10	*** SPARE ***	RESULTS DATA
	Assembly Model Code (bushing def / input shaft def, etc...) (Reference Part Num Definitions Tab)	46	18	Example: 1A2B3C4D5E6F7G8H9I (reference Part Num Definitions Tab).	
	*** SPARE ***	64	4	*** SPARE ***	
	Housing Stake Counter - BSI Line	68	2	Updated by BSI Line for different stake positions.	
	Rework Counter - BSI Line	70	2	Updated by Teardown / Evaluation Area	
	Rework Counter - Sensor Assembly	72	2	Updated by Teardown / Evaluation Area	
	Rework Counter - Final Assembly	74	2	Updated by Teardown / Evaluation Area	
	*** SPARE ***	76	10	*** SPARE ***	
	Special Manufacturing Designator (Plant use to designate specific test and calibration assemblies)	86	8		
	*** SPARE ***	94	10	*** SPARE ***	
	*** SPARE ***	104	[illegible]	*** SPARE ***	
	Input Shaft Serial Number	114	[illegible]	example (xxxxxxxxYYDDDzssss00)	
	Pinion Serial Number	[illegible]	20	example (xxxxxxxxYYDDDzssss00)	
	*** SPARE ***	[illegible]	10	*** SPARE ***	
	Motor Serial Number	164	10	example X0123-M1234 per 24079068 specified format	
	Marriage Dial ID	174	2		
	*** SPARE ***	176	8	*** SPARE ***	
	Rack Process Serial Number	184	20	example (xxxxxxxxYYDDDzssss00)	
	Assist Cover Process Serial Number	204	20	example (xxxxxxxxYYDDDzssss00)	
	Belt Weight	224	2		
	Belt Tension	226	2		
	Rack Clearance	228	2		
	Avg Backdrive Force	230	2		
	BSI Line #	232	2		
	Noise Test Machine #	234	2		
	Backdrive Test Machine #	236	2		
	Sensor Assembly Line #	238	2		
	Final Function Test Machine #	240	2		
	Final Assembly Line #	242	2		
	*** RESERVED ***	244	10	10 bytes reserved but not defined for Data	
	*** SPARE ***	254	1746	*** SPARE ***	
	TOTAL BYTES		**2000**		

Example Results Data

Figure 22: Part RFID Memory Map Example

RECORD OF REVISIONS

Revision #	Date	Section	Description
001	23FE17	All	Changed document from a Guideline to a Specification.
002			
003			
004			
005			
006			
007			
008			
009			
010			
011			
012			
013			
014			
015			
016			
017			
018			
019			
020			