



# Corporate FRU ID: SEEPROM Programming

## Overview and Procedures for FRU Vendors

**Document Number and Revision:** 950-3757-02 rev52

### Overview

FRU ID stands for field replaceable unit identification. The FRU ID program is the standard for identifying Oracle's FRU ID-enabled parts. This document details how to program serial electrically erasable and programmable, read-only memory (SEEPROM) components used for FRU ID and defines the data types and fields that must be programmed on all new and existing FRU ID-enabled hardware units (see a Bill of Materials, or BOM, for a list of FRU ID-enabled hardware units). All software and documentation supporting the FRU will be provided by the Operations Engineering representative.

### Audience

This document is for the FRU vendor's technical team.

### Contents

<b>Overview</b>	Error! Bookmark not defined.
<b>Audience</b>	<b>1</b>
<b>Introduction</b>	<b>3</b>
<b>1.0 Vendor Responsibilities</b>	<b>3</b>
<b>2.0 Programming the SEEPROM</b>	<b>3</b>
<b>2.1 SEEPROM Memory Organization</b>	<b>3</b>
<b>2.2 Write-Protected Hardware</b>	<b>4</b>
<b>2.3 SEEPROM Programming Software</b>	<b>4</b>
<b>3 Defining Vendor Programmed Write-Protected Fields</b>	<b>5</b>
<b>3.1 Understanding the Serial EEPROM Programming Instructions</b>	<b>5</b>
<b>3.2 Defining the Serial EEPROM Programming Data Fields</b>	<b>5</b>
3.2.1 FRU Header Information	6
3.2.2 Byte Template	7
3.2.3 Vendor Supplied Fields	8
3.2.4 Programming Instructions	9
<b>3.3 Programming Vendor Supplied Data</b>	<b>10</b>
Base_Part_IdentityR/UNIX_Timestamp32	10



3.3.2 Base_Part_IdentityR/Base_Part_Revision_Level	11
Base_Part_IdentityR/Base_Part_Serial_Number	11
3.3.4 Base_Part_IdentityR/Supplier_ID	11
3.3.5 Cyclic Redundancy Check/Checksum and CRC32	11
3.3.5.1 CRC32	11
3.3.5.2 Checksum32	12
<b>4 FRU Rework/Retest Before Shipping Policy</b>	<b>12</b>
<b>5 Getting Help</b>	<b>12</b>
<b>Appendix A Example Serial EEPROM Programming Instructions</b>	<b>13</b>
<b>Appendix B Example Bill of Materials - First Page</b>	<b>27</b>
<b>Related Information</b>	<b>28</b>
<b>Document History</b>	<b>28</b>

---

**List of images**

Figure 1 SEEPROM Memory Organization	5
Figure 2 FRU Header Information	6
Figure 3 Byte Template	8
Figure 4 Vendor Supplied Fields	9
Figure 5 Programming Instructions	10

## Introduction

The FRU is a part used to assemble a new system. The component must have a memory device, such as a SEEPROM, for programming FRU ID-specific data into the memory. The Oracle FRU ID program requires SEEPROM programming to identify and track all Oracle system hardware products. This SEEPROM captures error information for improved system and FRU failure analysis.

To program the FRU SEEPROM, constant data (for example, part number), vendor-specific data (for example, manufacture location), as well as FRU-specific data (for example, serial number or Ethernet addresses) are programmed into each FRU to record information about the FRU ongoing. This data is captured and sent to a FRU ID database for system identification and diagnostic purposes.

## 1.0 Vendor Responsibilities

FRU vendors are responsible for programming the SEEPROM to meet the FRU ID *Serial EEPROM Programming Instructions* specifications (see Section 3, "Defining Vendor Programmed Write-Protected Fields"). This establishes a manufacturing record into the write-protected memory portion of the SEEPROM component and initializes the writable memory portion.

Once the SEEPROM is programmed, an audit process must be established to ensure that images are properly programmed on an ongoing basis. This audit process requires a spot-check of fixed and variable data periodically.

A process to retest and rework the FRU data before shipment must be established without modifying existing write-protected data and with the option to add writable data as needed.

## 2.0 Programming the SEEPROM

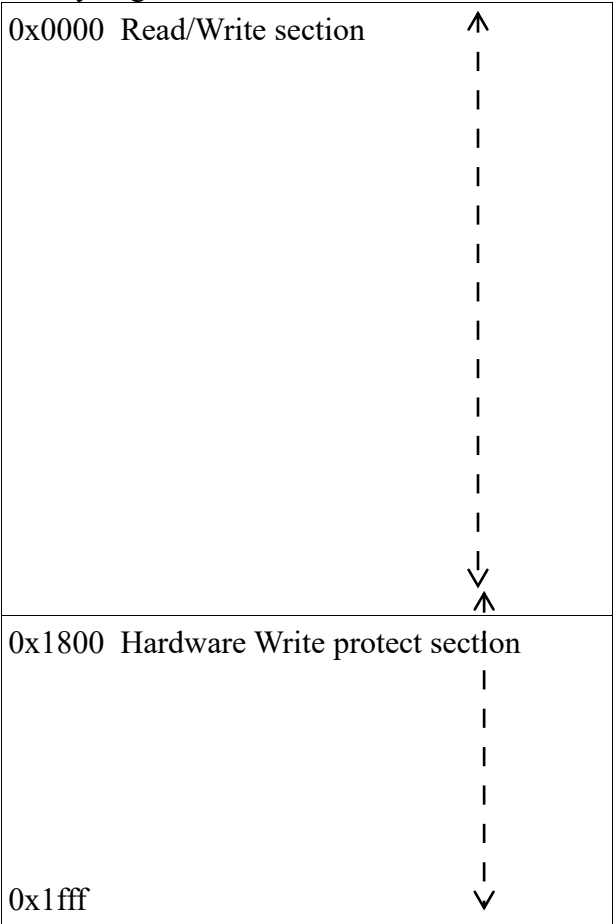
The following sections describe how to program the Microchip AT24C64 SEEPROM for FRU identification.

**Note** — If not using the Microchip SEEPROM for FRU ID, contact the Operations Engineering representative.

### 2.1 SEEPROM Memory Organization

The AT24C64 is an 64K (kilobyte) SEEPROM with erasable and programmable, read-only memory organized in 8K bytes. It includes a hardware write-protect pin for  $\frac{1}{4}$  of the array [0x1800 to 0x1fff]. Figure 2-1 shows the SEEPROM component's memory configuration.

**Figure 1** SEEPROM Memory Organization



2.2 Write-Protected Hardware

The write-protect pin 7 of the SOIC package on the SEEPROM must be forced low to program the write-protected section of the memory. In the system, the write-protect pin must be held high to prevent any programming over the write-protected data, that is, the manufacturing record and FRU-specific, write-protected elements.

Depending on the system implementation chosen to drive the write-protect pin high, there are several options for ensuring that the FRU's SEEPROM can be accessed for programming the write-protected portion. The Operations Engineering representative or Operations program manager can direct you to the resource to help determine a method for driving write-protect low on the FRU and programming write-protected data into the SEEPROM.

2.3 SEEPROM Programming Software

Software such as LabView, or a UniSite Data I/O machine with associated software, can be used to program SEEPROMs. Many test fixtures can be modified or programmed to incorporate a SEEPROM programming step into their functionality. Oracle currently has no requirements for supplier programming devices. However, Oracle has developed software

utilities that can aid in the programming process of FRU ID. Types of FRU ID utilities include: image verification tools, image conversion to binary tools, algorithm check utility, and tools to automate the programming of FRUs for FRU ID. Some tools run by command-line while others are web-based applications. Contact your Oracle supplier representative for suggestions on how to best set up the FRU ID programming process and to receive Oracle's utilities.

## 3 Defining Vendor Programmed Write-Protected Fields

There are data elements that must be programmed for each type of FRU. This document discusses generic data for all platforms. Additional platform-specific data requirements are called out in the *Serial EEPROM Programming Instructions* specifications document. This additional data can be either FRU-specific or vendor-specific.

### 3.1 Understanding the Serial EEPROM Programming Instructions

The *Serial EEPROM Programming Instructions* specifications document is a FRU ID image specification structured to the *BOM* of the FRU ID-enabled parts. It specifies the required write-protected and writable data initialization of the FRU ID SEEPROM. It serves as both a human readable specification and as an input file to validate the data using a verification tool supplied by Oracle.

**Note** — It is the FRU vendor's responsibility to ensure that all FRUs are programmed in compliance with the *Serial EEPROM Programming Instructions* specifications document for these FRUs.

Oracle's design engineering group generates and maintains the *Serial EEPROM Programming Instructions* specifications document for each Oracle platform. Any platform-specific data is included in the *Serial EEPROM Programming Instructions* specifications document and must be programmed in addition to the required generic data discussed in the following sections. Oracle engineers provide this additional data as a FRU-specific note in the Programming Instructions section of the *Serial EEPROM Programming Instructions* specifications document.

It is not Oracle's responsibility to dictate how to program this data into the SEEPROMs and validate the output; however, Oracle has several utilities available to support this effort.

### 3.2 Defining the Serial EEPROM Programming Data Fields

The *Serial EEPROM Programming Instructions* specifications document displays generic data to be entered when programming the FRU as well as additional vendor supplied or FRU-specific fields that need input. The documents are distributed in .html format. The data needs to be in a binary file before sending to Oracle.

The *Serial EEPROM Programming Instructions* specifications document covers the following topics:

- FRU Header Information
  - Byte template
  - Vendor supplied fields
  - Programming instructions

### 3.2.1 FRU Header Information

The FRU header information includes the *Serial EEPROM Programming Instructions* specifications document part number as well as the FRU's assigned part number and description.

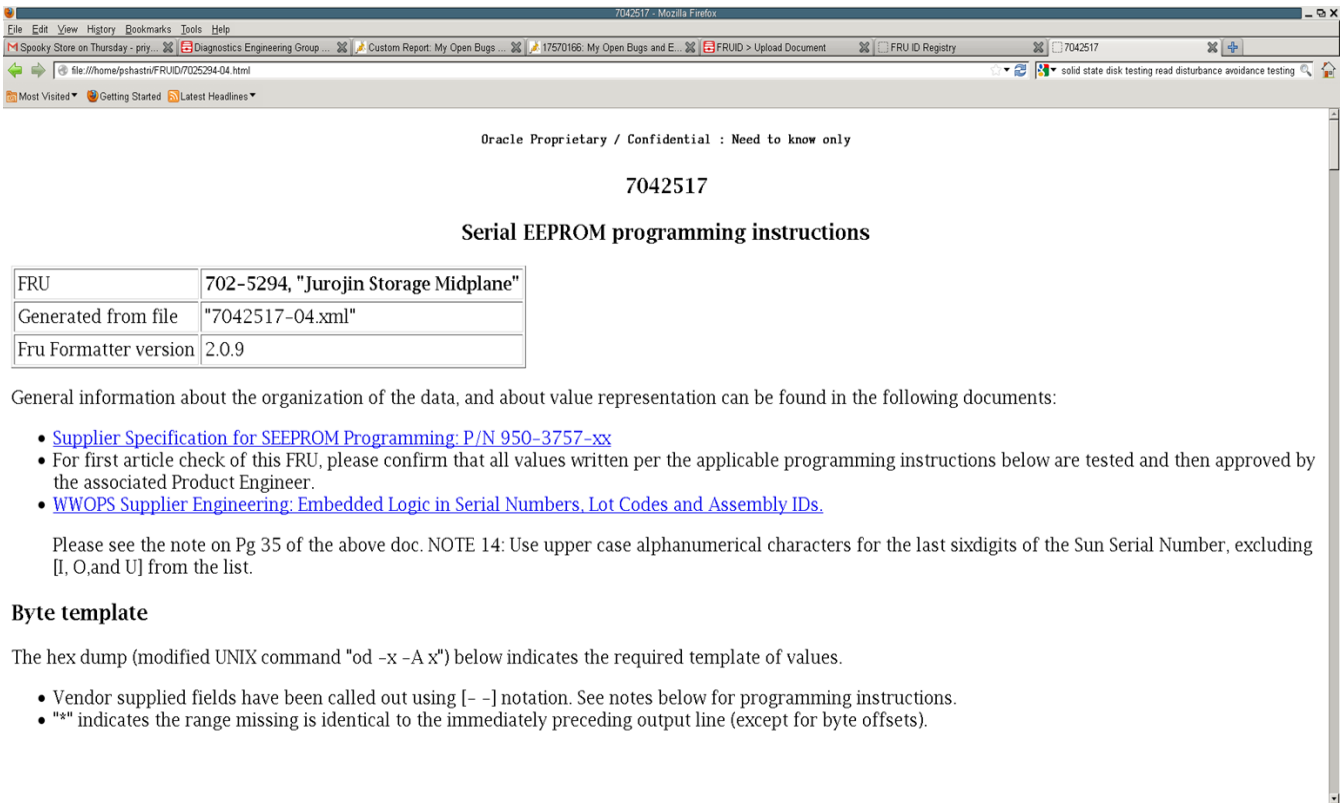
The following graphic is an example of the FRU Header Information section of the *Serial EEPROM Programming Instructions* specifications document. The document part number is 7 digits in length and is as follows:

SpecPartNo: 7123456.html

FRUPartNo: 702-3294

The FRU part number can vary in length from 7 digits to 10 digits.

**Figure 2 FRU Header Information**



### 3.2.2 Byte Template

The byte template shows the constant values that need to be programmed into the SEEPROM. The hex dump provides specific values for the addresses in the SEEPROM that have constant values (for example, values that are the same for each instance of a given FRU of a particular Oracle part number and dash level).

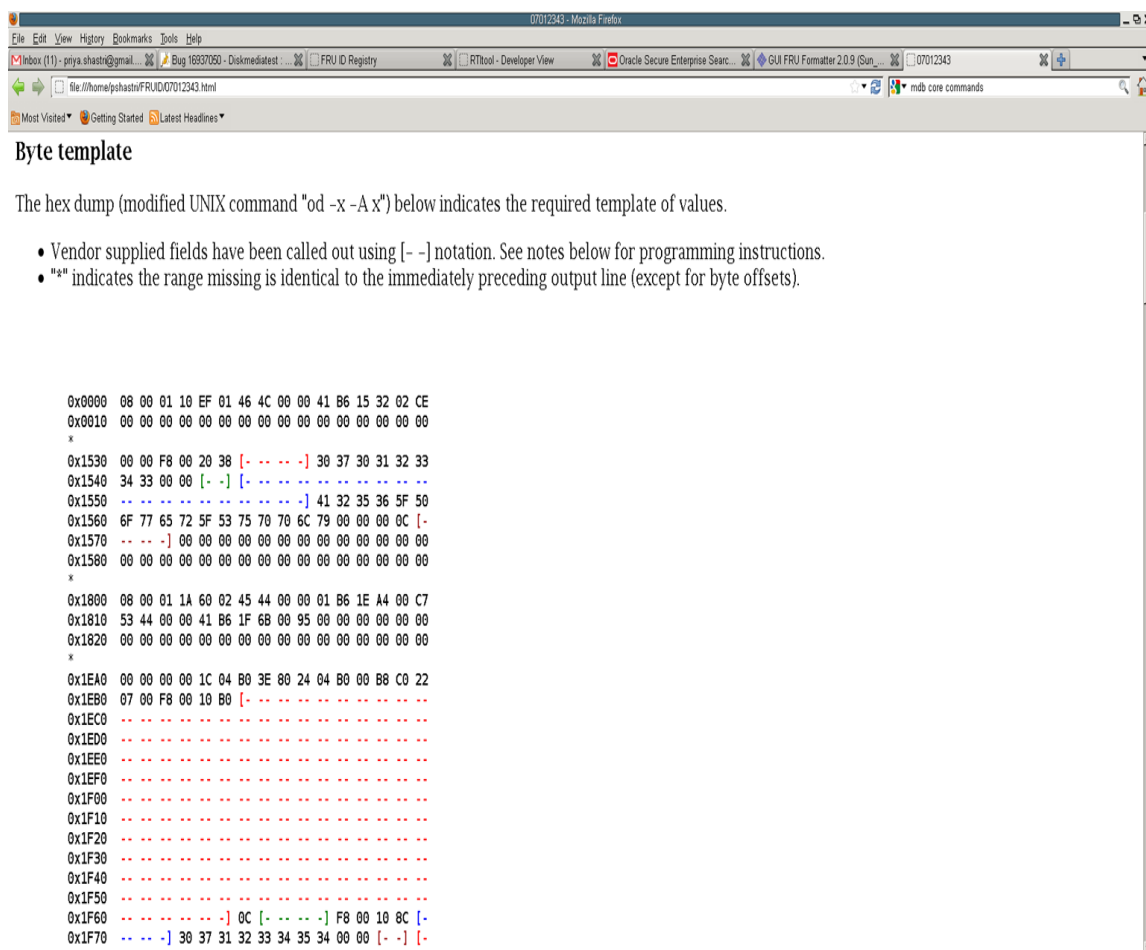
The fields with dashes, displayed in color matching the FRU descriptions in the Vendor Supplied Fields table, indicate variable fields that may not be the same for each instance of a given FRU. Programming specifications for these fields are in the vendor supplied fields (see Section 3.2.3, "Vendor Supplied Fields" for further explanation). Vendors are responsible for programming SEEPROMs to comply with the byte template.

Figure 3-2 SEEPROM programming byte template

The following graphic is an example of the byte template section of the *Serial EEPROM Programming Instructions* specifications document:

**Note** — Rows that contain an asterisk symbol (\*) imply repeated rows of the same values. For example, repeated rows programmed to all zero values are not shown, making the template easier to view.

Figure 3 Byte Template

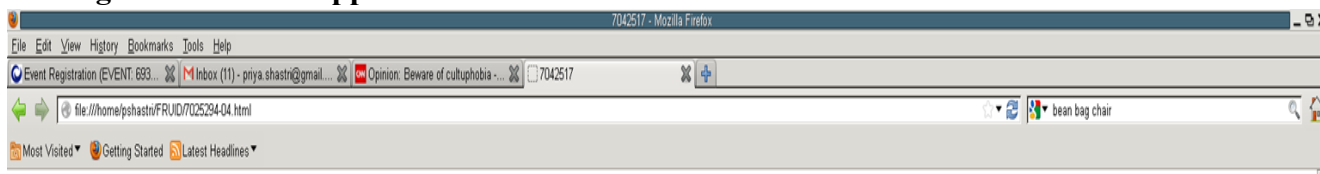




### 3.2.3 Vendor Supplied Fields

The vendor supplied fields table is matched to the associated fields in the byte template by location (address) and color. A brief description is provided for each field as well as a reference note to a more detailed description that appears in the programming instructions section (see Section 3.2.4, "Programming Instructions").

**Figure 4 Vendor Supplied Fields**



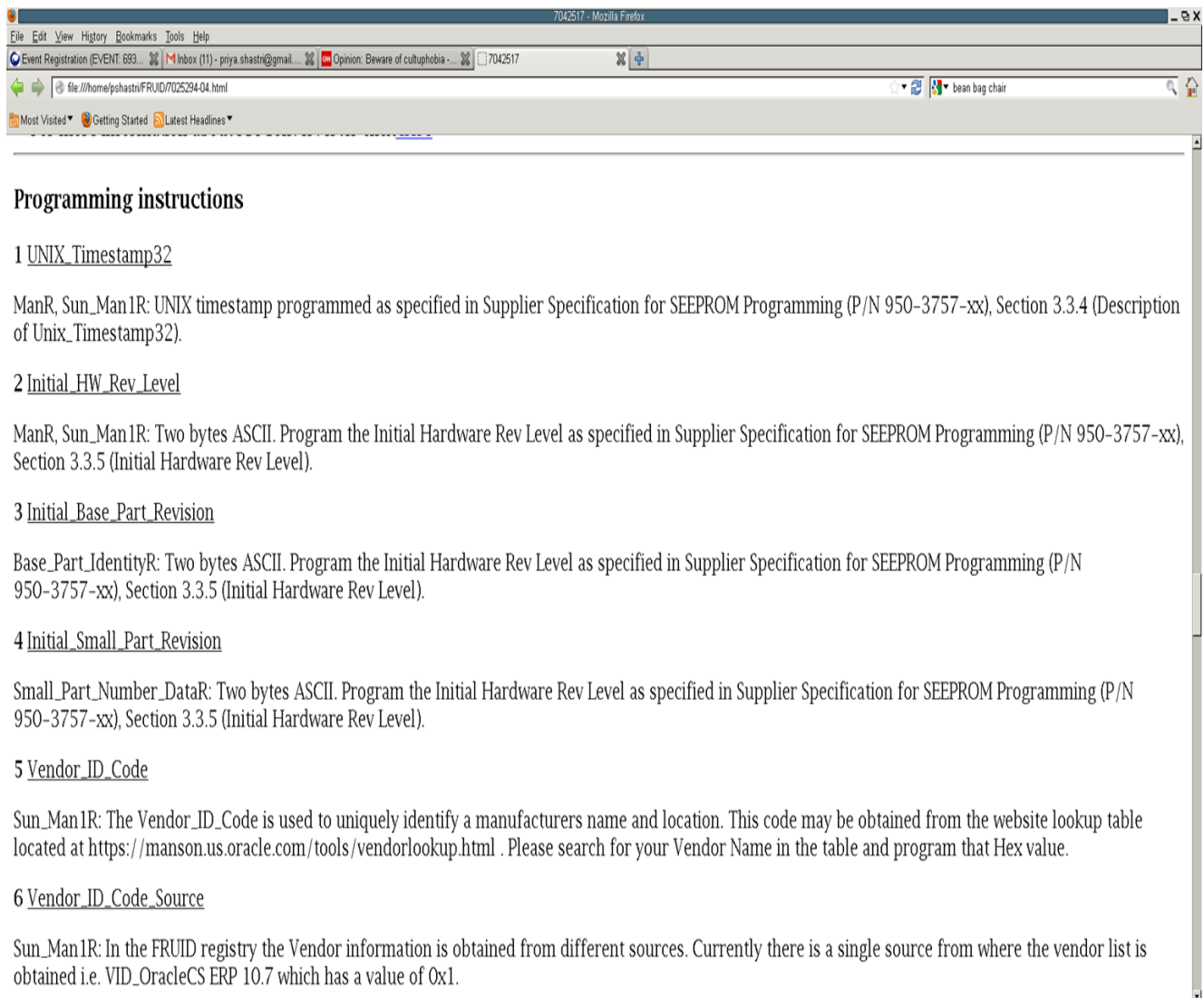
Field	Location	Length	Note/Doc
/FRU_IdentityR/UNIX_Timestamp32	0x1536	4	NOTE: 1
/FRU_IdentityR/FRU_Revision_Level	0x1544	2	NOTE: 27
/FRU_IdentityR/FRU_Serial_Number	0x1546	20	NOTE: 39
/Chassis_IdentityR/UNIX_Timestamp32	0x1571	4	NOTE: 1
/Chassis_IdentityR/Chassis_SerialNumber	0x159D	20	NOTE: 39
/System_IdentityR/UNIX_Timestamp32	0x15B5	4	NOTE: 40
/System_IdentityR/System_Part_Number	0x15B9	20	NOTE: 40
/System_IdentityR/System_Revision_Level	0x15CD	2	NOTE: 40
/System_IdentityR/System_Identity_Serial_Number	0x15CF	18	NOTE: 40
/System_IdentityR/System_Model_Name	0x15E1	20	NOTE: 40
/System_IdentityR/MACADDR	0x15F5	6	NOTE: 41
/System_IdentityR/HOSTID	0x15FB	4	NOTE: 41
/Product_IdentityR/UNIX_Timestamp32	0x1603	4	NOTE: 41
/Product_IdentityR/Product_Part_Number	0x1607	20	NOTE: 41
/Product_IdentityR/Product_Revision_Level	0x161B	2	NOTE: 41
/Product_IdentityR/Product_Identity_Serial_Number	0x161D	18	NOTE: 41
/Product_IdentityR/Product_Model_Name	0x162F	20	NOTE: 41
/Product_IdentityR/MACADDR	0x1643	6	NOTE: 41
/Product_IdentityR/HOSTID	0x1640	4	NOTE: 41



## 3.2.4 Programming Instructions

The programming instructions list each vendor supplied field, including a description or a definition, as well as a reference to find more details on a specific field, if needed.

**Figure 5 Programming Instructions**



*Figure 3-4 SEEPROM programming instructions*

**Note** — See Appendix A, "Example Serial EEPROM Programming Instructions (8xx-xxxx-xx) Specifications Document" with FRU\_specific, vendor-supplied programming data to input for a more complex example.

### 3.3 Programming Vendor Supplied Data

The following list indicates the set of write-protected data common to all FRU ID compliant FRUs. Additional FRU-specific data may be required and appears with explanation in the *Serial EEPROM Programming Instructions* specifications document prepared by the Oracle representative (see Appendix A, "Example Serial EEPROM Programming Instructions (8xx-xxxx-xx) Specifications Document").

- Base\_Part\_IdentityR/UNIX\_Timestamp32
- Base\_Part\_IdentityR/Initial\_Base\_Part\_Revision
- Base\_Part\_IdentityR/Base\_Part\_Serial\_Number
- Base\_Part\_IdentityR/Supplier\_ID

For 256 byte seeproms, Small\_Part\_Number\_DataR is programmed which has the following fields

- Small\_Part\_Number\_DataR/UNIX\_Timestamp32
- Small\_Part\_Number\_DataR/Initial\_Small\_Part\_Number
- Small\_Part\_Number\_DataR/Initial\_Small\_Part\_Revision
- Small\_Part\_Number\_DataR/Small\_Part\_Serial\_Number
- Small\_Part\_Number\_DataR/Supplier\_ID
- Small\_Part\_Number\_DataR/SpecPartNo

#### Base\_Part\_IdentityR/UNIX\_Timestamp32

The UNIX timestamp is a 32-bit big-endian integer that represents the number of seconds since January 1, 1970 at 00:00:00 UTC (GMT). This means the UNIX timestamp is calculated in terms of the UTC (GMT) timezone.

On UNIX and Windows systems, the ANSI C time() function provides this value regardless of the local timezone provided that the date-time of the machine is correctly set for local conditions and the format matches the one indicated in the following example.

```
#include <stdio.h> #include <time.h>
```

```
int main(void)
```

```
{
```

```
(void) printf("UNIX_Timestamp32 = 0x%lx\n",time(NULL));
```

```
}
```

For a system time of **Mar 29 11:26:25 2001 PST**, the value of time() is **0x3ac38c61**.

**Note** — Some machines will need to have their date-time formatting modified to produce the correct result.

**Note** — The UNIX date/time stamp should be recorded as close as possible to the date/time that the write-

protected data is programmed into the part. The date/time stamp needs to be accurate to within 24 hours of the actual programming of the part.

### 3.3.2 Base\_Part\_IdentityR/Base\_Part\_Revision\_Level

The initial hardware revision level is a section of the FRU part number indicating the current Oracle revision level for the FRU being manufactured. This is the only section of the part number that is not a constant field in the byte template.

Changes to the revision level may occur in instances where firmware is updated or the labeling changes. This revision level can be obtained from the first page of the BOM for the FRU (see Appendix B, "Example Bill of Materials - First Page"). The new revision level needs to be implemented into the SEEPROM programming process for the particular FRU and the new image must be verified by Oracle before production can proceed.

### Base\_Part\_IdentityR/Base\_Part\_Serial\_Number

Base\_Part\_IdentityR: Sun\_Serial\_Number is a unique identifier with max length of 30 characters. The serial number should be left justified.

### 3.3.4 Base\_Part\_IdentityR/Supplier\_ID

The supplier of this part from an enumerated list. The source for this encoding is Oracle R12 ERP.

### 3.3.5 Cyclic Redundancy Check/Checksum and CRC32

There are two types of calculation checks to check for data corruption, which are CRC32 and Checksum32. A utility named *crctool* is available to perform these calculations. The *crctool* can be run on Windows or UNIX.

FRU ID image verifier utilities can also be used to validate these calculations.

Command line versions of the verifier utility are available and can be rolled into the programming process to verify the binary images automatically.

All utilities are available from your Oracle Product Engineer.

#### 3.3.5.1 CRC32

CRC32 detects data corruption in the write-protected data segment in the upper 2KB of the SEEPROM memory.

The CRC32 is calculated over the interval indicated in the vendor supplied fields table of the *Serial EEPROM Programming Instructions* specifications document. This is a closed interval (that is, the end points of the interval are included in the calculations).

The *crctool* calculates the CRC32.

### 3.3.5.2 Checksum32

The Checksum32 programming requirements may appear in the Vendor Supplied Fields table when additional FRU-specific programming is required for certain data types in the data area of the SEEPROM.

The *crctool* also calculates the Checksum32.

**Note** — The source code for the *crctool* is available from Oracle.

## 4 FRU Rework/Retest Before Shipping Policy

Correctly programmed, write-protected data should never be changed. However, writable data may need changes from time-to-time. If the FRU was not programmed correctly to the *Serial EEPROM Programming Instructions* specifications, then correction to the write-protected data is required before shipping.

## 5 Getting Help

Direct all issues or questions concerning FRU ID SEEPROM programming or this document to the Oracle Supply Manager.

## Appendix A Example Serial EEPROM Programming Instructions

**Note:** This document is provided to the vendor by the Product Engineer and the values for the Oracle specific information are provided by the PE to the vendor. The vendor will not have access to internal Oracle websites.

### Example for 64K seeprom part

Oracle Proprietary / Confidential : Need to know only

70542342

Serial EEPROM programming instructions

FRU	7023456, "test part"
FRU shortname	"testpart"
Generated from file	"70542342.xml"
Fru Formatter version	2.0.9

General information about the organization of the data, and about value representation can be found in the following documents:

Supplier Specification for SEEPROM Programming: P/N 950-3757-xx

For first article check of this FRU, please confirm that all values written per the applicable programming instructions below are tested and then approved by the associated Product Engineer.

WWOPS Supplier Engineering: Embedded Logic in Serial Numbers, Lot Codes and Assembly IDs.(PN :923-3383-xx)

Please see the note on Pg 35 of the above doc. NOTE 14: Use upper case alphanumeric characters for the last sixdigits of the Sun Serial Number, excluding [I, O,and U] from the list.

### Byte template

The hex dump (modified UNIX command "od -x -A x") below indicates the required template of values.

Vendor supplied fields have been called out using [- -] notation. See notes below for programming instructions.

"\*" indicates the range missing is identical to the immediately preceding output line (except for byte offsets).

```
0x0000  08 00 01 10 EF 01 46 4C 00 00 41 B6 15 32 02 CE
0x0010  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
*
0x1530  00 00 F8 00 20 38 [- - - -] 36 30 32 34 35 36
```

```

0x1540 33 00 00 00 30 31 [- -- -- -- -- -- -- --
0x1550 -- -- -- -- -- -- -- -- -] 74 65 73 74 00 00
0x1560 00 00 00 00 00 00 00 00 00 00 00 00 00 00 F0 00
0x1570 A9 [- -- -- -- -- -- -- -- -- -- -- -- --
0x1580 -- -- -- -- -- -- -- -- -- -- -- -- -- -] 36
0x1590 30 32 32 33 34 35 00 00 00 00 0C [- -- -- -] 00
0x15A0 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
*
0x1800 08 00 01 1A 3B 02 45 44 00 00 01 B6 1E 9F 00 09
0x1810 53 44 00 00 41 B6 1E A8 01 58 00 00 00 00 00 00
0x1820 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
*
0x1E90 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 C0
0x1EA0 22 0D 08 0C D7 B5 55 EB F8 00 10 BF [- -- -- -]
0x1EB0 74 65 73 74 00 00 00 00 00 00 00 00 00 00 00 00
0x1EC0 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 [- -]
0x1ED0 [] [- -- -- -- -- -- -- -- -- -- -- -- --
0x1EE0 -- -- -- -- -- -- -- -- -- -- -- -- -- --
0x1EF0 -- -- -- -- -- -- -- -- -- -- -- -- -- --
0x1F00 -- -- -- -- -- -- -- -- -- -- -- -- -- --
0x1F10 -] 35 31 31 31 32 33 34 [- -- -- -- -- -- --
0x1F20 -- -- -- -- -- -- -- -- -- -- -- -- -- --
0x1F30 -- -- -- -- -- -] 34 56 33 46 31 2D 32 59 32 57
0x1F40 32 58 34 53 33 57 00 00 00 00 30 31 [- -] 74 65
0x1F50 73 74 70 61 72 74 00 00 00 00 00 00 00 00 4E 00
0x1F60 37 30 35 34 32 33 34 32 00 00 00 F8 00 10 8C [-
0x1F70 -- -- -] 37 30 32 33 34 35 36 00 00 00 [- -] [-
0x1F80 -- -- -- -- -- -- -- -- -- -- -- -- -- --
0x1F90 -- -- -- -- -- -- -- -- -- -- -- -] 74 65 73
0x1FA0 74 20 70 61 72 74 00 00 00 00 00 00 00 00 00 00
0x1FB0 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0x1FC0 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0x1FD0 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0x1FE0 00 00 00 00 00 00 00 00 00 00 00 00 00 00 [- -- -]

```

0x1FF0 37 30 35 34 32 33 34 32 00 00 00 0C [- -- -- -]

Byte Template Description

Location/Offset	Field Name	Data/Value
0x153A	FRU_Part_Number	6024563
0x1544	FRU_Revision_Level	01
0x155A	FRU_Description	test
0x158F	FRU_Part_Dash_Number	6022345
0x1EA1	Device_Type	d08
0x1EB0	Sun_Fru_Description	test
0x1F11	Sun_Part_Number	5111234
0x1F36	Serial_Number_Format	4V3F1-2Y2W2X4S3W
0x1F4A	Initial_HW_Dash_Level	01
0x1F4E	Sun_Fru_Shortname	testpart
0x1F5E	Sun_Hazard_Class_Code	N
0x1F60	Sun_SpecPartNo	70542342
0x1F73	Initial_Base_Part_Number	7023456
0x1F9D	Base_Part_Description	test part
0x1FF0	SpecPartNo	70542342



Vendor supplied Fields			
Field			Note/Doc
/FRU_IdentityR/UNIX_Timestamp32			NOTE: 26
/FRU_IdentityR/FRU_Serial_Number			NOTE: 27
/Sun_FRU_LabelR/Sun_Serial_Number			NOTE: 27
Checksum32			NOTE: 20 (0x1532 - 0x1599, Inclusive)
/Sun_Man1R/UNIX_Timestamp32			NOTE: 1
/Sun_Man1R/Vendor_ID_Code			NOTE: 5

/Sun_Man1R/Vendor_ID_Code_Source			NOTE: 6
/Sun_Man1R/Vendor_Name_And_Site_Location			NOTE: 7
/Sun_Man1R/Sun_Serial_Number			NOTE: 8
/Sun_Man1R/Initial_HW_Rev_Level			NOTE: 2
/Base_Part_IdentityR/UNIX_Timestamp32			NOTE: 1
/Base_Part_IdentityR/Initial_Base_Part_Revision			NOTE: 3
/Base_Part_IdentityR/Base_Part_Serial_Number			NOTE: 9

/Base_Part_IdentityR/Supplier_ID			NOTE: 12
CRC32			NOTE: 21 (0x1EA8 - 0x1FFA, Inclusive)

## XML Template

Batch.xml file that has to be provide as input to ICT Coverter

Note : XXXX should be replaced with actual Values

```
<?xml version="1.0"?>
<!--
Copyright (c) 2012, Oracle and/or its affiliates. All rights reserved.
-->

<!DOCTYPE ICTBatchData>
<ICTBatchData>

<Field name="/FRU_IdentityR/UNIX_Timestamp32" offset="0x1536" value="XXXX"/>
<Field name="/FRU_IdentityR/FRU_Serial_Number" offset="0x1546" value="XXXX"/>
<Field name="/Sun_FRU_LabelR/Sun_Serial_Number" offset="0x1571" value="XXXX"/>
<Field name="Checksum32" offset="0x159B" value="XXXX"/>
<Field name="/Sun_Man1R/UNIX_Timestamp32" offset="0x1EAC" value="XXXX"/>
<Field name="/Sun_Man1R/Vendor_ID_Code" offset="0x1ECE" value="XXXX"/>
<Field name="/Sun_Man1R/Vendor_ID_Code_Source" offset="0x1ED0" value="XXXX"/>
<Field name="/Sun_Man1R/Vendor_Name_And_Site_Location" offset="0x1ED1" value="XXXX"/>
<Field name="/Sun_Man1R/Sun_Serial_Number" offset="0x1F18" value="XXXX"/>
<Field name="/Sun_Man1R/Initial_HW_Rev_Level" offset="0x1F4C" value="XXXX"/>
<Field name="/Base_Part_IdentityR/UNIX_Timestamp32" offset="0x1F6F" value="XXXX"/>
<Field name="/Base_Part_IdentityR/Initial_Base_Part_Revision" offset="0x1F7D" value="XXXX"/>
<Field name="/Base_Part_IdentityR/Base_Part_Serial_Number" offset="0x1F7F" value="XXXX"/>
<Field name="/Base_Part_IdentityR/Supplier_ID" offset="0x1FED" value="XXXX"/>
<Field name="CRC32" offset="0x1FFC" value="XXXX"/>

</ICTBatchData>
```

For more information about ICTConververter click [here](#)

## Programming instructions

### 1 UNIX Timestamp32

ManR, Sun\_Man1R: UNIX timestamp programmed as specified in Supplier Specification for SEEPROM Programming (P/N 950-3757-xx), Section 3.3.4 (Description of Unix\_Timestamp32).

## **2 Initial HW Rev Level**

ManR, Sun\_Man1R: Two bytes ASCII. Program the Initial Hardware Rev Level as specified in Supplier Specification for SEEPROM Programming (P/N 950-3757-xx), Section 3.3.5 (Initial Hardware Rev Level).

## **3 Initial Base Part Revision**

Base\_Part\_IdentityR: Two bytes ASCII. Program the Initial Hardware Rev Level as specified in Supplier Specification for SEEPROM Programming (P/N 950-3757-xx), Section 3.3.5 (Initial Hardware Rev Level).

## **4 Initial Small Part Revision**

Small\_Part\_Number\_DataR: Two bytes ASCII. Program the Initial Hardware Rev Level as specified in Supplier Specification for SEEPROM Programming (P/N 950-3757-xx), Section 3.3.5 (Initial Hardware Rev Level).

## **5 Vendor ID Code**

Sun\_Man1R: The Vendor\_ID\_Code is used to uniquely identify a manufacturers name and location. This code may be obtained from the website lookup table located at <http://fruid-tools.us.oracle.com/tools/vendorlookup.html> . Please search for your Vendor Name in the table and program that Hex value.

## **6 Vendor ID Code Source**

Sun\_Man1R: In the FRUID registry the Vendor information is obtained from different sources. Currently there is a single source from where the vendor list is obtained i.e. VID\_OracleCS ERP 10.7 which has a value of 0x1.

## **7 Vendor Name And Site Location**

Sun\_Man1R: This information can be obtained by looking at the table at <http://fruid-tools.us.oracle.com/tools/vendorlookup.html> Please search for your Vendor Name in the table and program the text in the green box as 64 byte text. The values for this field need to be left justified. The length of this field should not exceed 63 characters. The last byte should be 0x00.

## **8 Sun Serial Number**

Sun\_Man1R: Sun\_Serial\_Number is a unique identifier with max length of 30 characters. The format is defined by Serial\_Number\_Format. The values for this field need to be left justified.

## **9 Base Part Serial Number**

Base\_Part\_IdentityR: Sun\_Serial\_Number is a unique identifier with max length of 30 characters. The format is defined by Serial\_Number\_Format. The values for this field need to be left justified.

## **10 Small Part Serial Number**

Small\_Part\_Number\_DataR: Sun\_Serial\_Number is a unique identifier with max length of 30 characters. The format is defined by Serial\_Number\_Format. The values for this field need to be left justified.

## **11 Vendor ID**

Sun\_Man1R: This information can be obtained by looking at the table at <http://fruid-tools.us.oracle.com/tools/vendorlookup.html> Please search for your Vendor Name in the table and program that Hex value(Convert the decimal number to Hex).

**12 Supplier ID**

Base\_Part\_IdentityR, Small\_Part\_Number\_DataR: String provided by PE which internally gets converted to 3 byte hex code. This information can be obtained from your Oracle Product Engineer.

**13 Vendor ID Source**

Sun\_ManR: In the FRUID registry the Vendor information is obtained from different sources. Currently there is a single source from where the vendor list is obtained i.e. VID\_OracleCS ERP 10.7 which has a value of 0x1.

**14 Vendor Site ID**

Sun\_ManR: Vendor\_Site\_ID is a four byte code used to uniquely identify the vendor's manufacturing location. The authority for the code is identified by Vendor\_Site\_ID\_Source.

**15 Vendor Site ID Source**

Sun\_ManR: Vendor\_Site\_ID\_Source is a one byte enumeration that identifies the source authority for the Vendor\_Site\_ID code. Initially this will be OracleCS ERP 10.7. Other source authorities may be identified in the future.

**16 SUN\_SN**

Sun\_ManR: SUN\_SN is a unique identifier and aligns with ODW max length of 30 characters. The format is defined by SN\_Format.

**17 Vendor Name**

ManR: Vendor code programmed as specified in Supplier Specification for SEEPROM Programming (P/N 950-3757-xx), Section 3.3.1 (Vendor Name Description).

**18 Sun Serial No**

ManR: is to be programmed with the ascii encoding of the value for the serial number which is printed on the associated FRU bar code of the assembly. The document "Bar Code Marking Standard for FRU's" (P/N 950-1037-xx) details which part of the bar code specifies the serial number. All digits must be supplied as ascii values. For instance, serial number 1 translates into the 6 bytes: 0x30 0x30 0x30 0x30 0x30 0x31. [hint: the bar code label for the assembly is typically an 266-xxxx-01 part]

**19 Manufacture Loc**

ManR: Manufacturing Location programmed as specified in Supplier Specification for SEEPROM Programming (P/N 950-3757-xx), Section 3.3.2 (Manufacturing Location). The length of this field should not exceed 63 characters. The last character must be 0x00.

**20 Checksum32**

ManR, Sun\_Man1R: Checksum32 to be calculated as per Supplier Specification for SEEPROM Programming (P/N 950-3757-xx)

**21 CRC32**

ManR, Sun\_Man1R: CRC32 to be calculated as per Supplier Specification for SEEPROM Programming (P/N 950-3757-xx)

**22 Sun Key Code**

Sun\_SPD\_DIMMR: Please contact Oracle to get the Key Code assigned. It's a required field to program the Certification Signature.

**23** Sun\_Certification

Sun\_SPD\_DIMMR: The first 112 bits of an RFC 2104 SHA-256 HMAC hash based on bytes 0-188 of the SPD. The key(s) will be chosen by Oracle. The actual key used is indicated by the Certification key code in bytes 187-188.

**24** HOSTID

Hostid :Convert the scanned MAC address from the previous step into a 32-bit HOSTID(refer to document 923-3521-xx for details). Use that value to program this field.

**25** Ethernet\_Addr

Ethernet\_Addr:Scan the MACADDR barcode label attached to the socketed prom and program this field.

**26** Instruction0

Enter UNIX timestamp

**27** Instruction1

Enter serial number.

**Oracle Proprietary / Confidential : Need to know only**

## **1. Example for 256 byte seeprom part**

**Note:** This document is provided to the vendor by the Product Engineer and the values for the Oracle specific information are provided by the PE to the vendor. The vendor will not have access to internal Oracle websites.

**Oracle Proprietary / Confidential : Need to know only**  
**70457654**

**Serial EEPROM programming instructions**

FRU	7023456, "testpart"
Generated from file	"70457654.xml"
Fru Formatter version	2.0.9

General information about the organization of the data, and about value representation can be found in the following documents:

- Supplier Specification for SEEPROM Programming: P/N 950-3757-xx
- For first article check of this FRU, please confirm that all values written per the applicable programming instructions below are tested and then approved by the associated Product Engineer.
- WWOPS Supplier Engineering: Embedded Logic in Serial Numbers, Lot Codes and Assembly IDs.(PN :923-3383-xx)

Please see the note on Pg 35 of the above doc. NOTE 14: Use upper case alphanumerical characters for the last sixdigits of the Sun Serial Number, excluding [I, O,and U] from the list.

**Byte template**

The hex dump (modified UNIX command "od -x -A x") below indicates the required template of values.

- Vendor supplied fields have been called out using [- -] notation. See notes below for programming instructions.
- "\*" indicates the range missing is identical to the immediately preceding output line (except for byte offsets).

```
0x0000 08 00 01 06 56 00 00 00 00 00 00 00 00 00 00
0x0010 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
*
0x0080 08 00 01 10 85 01 43 44 00 00 41 B6 00 B1 00 4F
0x0090 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
*
0x00B0 00 C0 35 [- - - -] 33 30 33 34 35 36 37 [- -
0x00C0 - - - -] [- -] 30 31 F0 00 B0 [- - - -] 37
0x00D0 30 32 33 34 35 36 00 00 00 [- -] [- - - - -
0x00E0 - - - - - - - - - - - -] [- - - -]
0x00F0 37 30 34 35 37 36 35 34 00 00 00 0C [- - - -]
```



## Byte Template Description

Location/Offset	Field Name	Data/Value
0x00B7	Sun_Part_No	3034567
0x00C6	Initial_HW_Dash_Level	01
0x00CF	Initial_Small_Part_Number	7023456
0x00F0	SpecPartNo	7045765

## Vendor supplied Fields

Field	Location	Length	Note/Doc
/Sun_Part_Number_DataR/UNIX_Timestamp32	0x00B3	4	NOTE: 1
/Sun_Part_Number_DataR/Sun_Serial_No	0x00BE	6	NOTE: 18
/Sun_Part_Number_DataR/Vendor_Name	0x00C4	2	NOTE: 17
/Small_Part_Number_DataR/UNIX_Timestamp32	0x00CB	4	NOTE: 1
/Small_Part_Number_DataR/Initial_Small_Part_Revision	0x00D9	2	NOTE: 4
/Small_Part_Number_DataR/Small_Part_Serial_Number	0x00DB	18	NOTE: 10
/Small_Part_Number_DataR/Supplier_ID	0x00ED	3	NOTE: 12
CRC32	0x00FC	4	NOTE: 21 (0x00B1 - 0x00FA, Inclusive)

## XML Template

Batch.xml file that has to be provide as input to ICT Converter

- Note : XXXX should be replaced with actual Values

```
<?xml version="1.0"?>
```

```
<!--
```

Copyright (c) 2012, Oracle and/or its affiliates. All rights reserved.

```
-->
```

```
<!DOCTYPE ICTBatchData>
```

```
<ICTBatchData>
```

```
<Field name="/Sun_Part_Number_DataR/UNIX_Timestamp32" offset="0x00B3" value="XXXX"/>
```

```
<Field name="/Sun_Part_Number_DataR/Sun_Serial_No" offset="0x00BE" value="XXXX"/>
```

```
<Field name="/Sun_Part_Number_DataR/Vendor_Name" offset="0x00C4" value="XXXX"/>
```

```
<Field name="/Small_Part_Number_DataR/UNIX_Timestamp32" offset="0x00CB" value="XXXX"/>
```

```

<Field name="/Small_Part_Number_DataR/Initial_Small_Part_Revision" offset="0x00D9"
value="XXXX"/>
<Field name="/Small_Part_Number_DataR/Small_Part_Serial_Number" offset="0x00DB"
value="XXXX"/>
<Field name="/Small_Part_Number_DataR/Supplier_ID" offset="0x00ED" value="XXXX"/>
<Field name="CRC32" offset="0x00FC" value="XXXX"/>
</ICTBatchData>

```

- For more information about ICTConververter click [here](#)

## Programming instructions

### 1 UNIX\_Timestamp32

ManR, Sun\_Man1R: UNIX timestamp programmed as specified in Supplier Specification for SEEPROM Programming (P/N 950-3757-xx), Section 3.3.4 (Description of Unix\_Timestamp32).

### 2 Initial\_HW\_Rev\_Level

ManR, Sun\_Man1R: Two bytes ASCII. Program the Initial Hardware Rev Level as specified in Supplier Specification for SEEPROM Programming (P/N 950-3757-xx), Section 3.3.5 (Initial Hardware Rev Level).

### 3 Initial\_Base\_Part\_Revision

Base\_Part\_IdentityR: Two bytes ASCII. Program the Initial Hardware Rev Level as specified in Supplier Specification for SEEPROM Programming (P/N 950-3757-xx), Section 3.3.5 (Initial Hardware Rev Level).

### 4 Initial\_Small\_Part\_Revision

Small\_Part\_Number\_DataR: Two bytes ASCII. Program the Initial Hardware Rev Level as specified in Supplier Specification for SEEPROM Programming (P/N 950-3757-xx), Section 3.3.5 (Initial Hardware Rev Level).

### 5 Vendor\_ID\_Code

Sun\_Man1R: The Vendor\_ID\_Code is used to uniquely identify a manufacturers name and location. This code may be obtained from the website lookup table located at <http://fruidd-tools.us.oracle.com/tools/vendorlookup.html> . Please search for your Vendor Name in the table and program that Hex value.

### 6 Vendor\_ID\_Code\_Source

Sun\_Man1R: In the FRUID registry the Vendor information is obtained from different sources. Currently there is a single source from where the vendor list is obtained i.e. VID\_OracleCS ERP 10.7 which has a value of 0x1.

### 7 Vendor\_Name\_And\_Site\_Location

Sun\_Man1R: This information can be obtained by looking at the table at <http://fruidd-tools.us.oracle.com/tools/vendorlookup.html> Please search for your Vendor Name in the table and program the text in the green box as 64 byte text. The values for this field need to be left justified. The

length of this field should not exceed 63 characters. The last byte should be 0x00.

#### **8 Sun\_Serial\_Number**

Sun\_Man1R: Sun\_Serial\_Number is a unique identifier with max length of 30 characters. The format is defined by Serial\_Number\_Format. The values for this field need to be left justified.

#### **9 Base\_Part\_Serial\_Number**

Base\_Part\_IdentityR: Sun\_Serial\_Number is a unique identifier with max length of 30 characters. The format is defined by Serial\_Number\_Format. The values for this field need to be left justified.

#### **10 Small\_Part\_Serial\_Number**

Small\_Part\_Number\_DataR: Sun\_Serial\_Number is a unique identifier with max length of 30 characters. The format is defined by Serial\_Number\_Format. The values for this field need to be left justified.

#### **11 Vendor\_ID**

Sun\_Man1R: This information can be obtained by looking at the table at <http://fruid-tools.us.oracle.com/tools/vendorlookup.html> Please search for your Vendor Name in the table and program that Hex value(Convert the decimal number to Hex).

#### **12 Supplier\_ID**

Base\_Part\_IdentityR, Small\_Part\_Number\_DataR: String provided by PE which internally gets converted to 3 byte hex code.This information can be obtained from your Oracle Product Engineer.

#### **13 Vendor\_ID\_Source**

Sun\_ManR: In the FRUID registry the Vendor information is obtained from different sources. Currently there is a single source from where the vendor list is obtained i.e. VID\_OracleCS ERP 10.7 which has a value of 0x1.

#### **14 Vendor\_Site\_ID**

Sun\_ManR: Vendor\_Site\_ID is a four byte code used to uniquely identify the vendor's manufacturing location. The authority for the code is identified by Vendor\_Site\_ID\_Source.

#### **15 Vendor\_Site\_ID\_Source**

Sun\_ManR: Vendor\_Site\_ID\_Source is a one byte enumeration that identifies the source authority for the Vendor\_Site\_ID code. Initially this will be OracleCS ERP 10.7. Other source authorities may be identified in the future.

#### **16 SUN\_SN**

Sun\_ManR: SUN\_SN is a unique identifier and aligns with ODW max length of 30 characters. The format is defined by SN\_Format.

#### **17 Vendor\_Name**

ManR: Vendor code programmed as specified in Supplier Specification for SEEPROM Programming (P/N 950-3757-xx), Section 3.3.1 (Vendor Name Description).

#### **18 Sun\_Serial\_No**

ManR: is to be programmed with the ascii encoding of the value for the serial number which is printed

on the associated FRU bar code of the assembly. The document "Bar Code Marking Standard for FRU's" (P/N 950-1037-xx) details which part of the bar code specifies the serial number. All digits must be supplied as ascii values. For instance, serial number 1 translates into the 6 bytes: 0x30 0x30 0x30 0x30 0x30 0x31. [hint: the bar code label for the assembly is typically an 266-xxxx-01 part]

#### **19 Manufacture\_Loc**

ManR: Manufacturing Location programmed as specified in Supplier Specification for SEEPROM Programming (P/N 950-3757-xx), Section 3.3.2 (Manufacturing Location). The length of this field should not exceed 63 characters. The last character must be 0x00.

#### **20 Checksum32**

ManR, Sun\_Man1R: Checksum32 to be calculated as per Supplier Specification for SEEPROM Programming (P/N 950-3757-xx)

#### **21 CRC32**

ManR, Sun\_Man1R: CRC32 to be calculated as per Supplier Specification for SEEPROM Programming (P/N 950-3757-xx)

#### **22 Sun\_Key\_Code**

Sun\_SPD\_DIMMR: Please contact Oracle to get the Key Code assigned. It's a required field to program the Certification Signature.

#### **23 Sun\_Certification**

Sun\_SPD\_DIMMR: The first 112 bits of an RFC 2104 SHA-256 HMAC hash based on bytes 0-188 of the SPD. The key(s) will be chosen by Oracle. The actual key used is indicated by the Certification key code in bytes 187-188.

#### **24 HOSTID**

Hostid :Convert the scanned MAC address from the previous step into a 32-bit HOSTID(refer to document 923-3521-xx for details). Use that value to program this field.

#### **25 Ethernet\_Addr**

Ethernet\_Addr:Scan the MACADDR barcode label attached to the socketed prom and program this field.

### **2. Oracle Proprietary / Confidential : Need to know only**

## Appendix B Example Bill of Materials - First Page

The *Bill of Materials (BOM)* is sent to all FRU vendors when production on a new system with FRU-enabled parts is about to begin. The *BOM* indicates the rev, or revision, level for a particular FRU on the first page. This document accompanies the *Serial EEPROM Programming Instructions* specifications document. The following is an example of the data printed on the first page of a *BOM* indicating the rev level.

### EXAMPLE BILL OF MATERIALS DATA - FIRST PAGE

004 BEAVERTON, US Sun Multi-Level Bill of Material Structure Report with AVL D  
Report Date: 21-FEB-2001

Item Selection:      Specific

Item: XXX-XXXX-01

Alternate Selection : All

Alternate:

Revision:      03

Date :      21-FEB-01 12:53

Item From:

To: COSTSOURCE

Category Set:

Category From:

Level to Explode: 1

Implemented Only: Yes

Display Option: Current

Explosion Quality : 1

Assembly Detail:No

Order Entry Detail : No

Lead Time Detail: No

Reference Designators: No

Substitute Components: No

Order By : Op Seq, Item Seq

Use Planning Percent :No

## Related Information

### Document History

<b>Rev</b>	<b>Date</b>	<b>Description of Change</b>	<b>Originator</b>
0	8/17/1999	Initial release.	N/A
01	1/12/2000	Made mostly generic to all FRUs.	N/A
02	1/17/2000	Added relevant examples	N/A
03	1/21/2000	Added relevant descriptions	N/A
04	9/1/2000	Added serial number details	N/A
01 50	05/01/2001	Initial release of newly formatted document	N/A
01 51	4/26/2001	Expanded data element descriptions	N/A
01 52	9/14/2001	Made corrections in section 3.2.2 due to inconsistency in the note and the Figure 3.2	N/A
1 02	12/3/2001	Changed URL <a href="http://64.41.217.110:8080/fruid/fruvendor.html">http://64.41.217.110:8080/fruid/fruvendor.html</a> to	N/A
02 51	11/14/2013	Updated format of 885 document to reflect Oracle records.	N/A
02 52	2/24/2024	Updated links, removed internal reference documents, updated to Redwood format	N/A