JAIF B-21 Global Radio Frequency Identification (RFID)
Item Level Standard

AIAG PUBLICATIONS

An AIAG publication reflects a consensus of those substantially concerned with its scope and provisions. An AIAG publication is intended as a guide to aid the manufacturer, the consumer, and the general public. The existence of an AIAG publication does not in any respect preclude anyone from manufacturing, marketing, purchasing, or using products, processes, or procedures not conforming to the publication.

CAUTIONARY NOTICE

AIAG publications are subject to periodic review and users are cautioned to obtain the latest editions.

MAINTENANCE PROCEDURE

Recognizing that this AIAG publication may not cover all circumstances, AIAG has established a maintenance procedure. Please refer to the Maintenance Request Form at the back of this document to submit a request.

Published by:

Automotive Industry Action Group

26200 Lasher Road, Suite 200

Southfield, Michigan 48033

Phone: (248) 358-3570 • Fax: (248) 358-3253

APPROVAL STATUS

The AIAG Materials Management Steering Committee and designated stakeholders approved this document for publication on November 3, 2011.

INTERNATIONAL COPYRIGHT INFORMATION

This manual is subject to International Copyright as shown herein and unauthorized reproduction is not permitted. Contact one of the associations below to discuss reproduction permission. If translated, this manual has been translated and published for guidance only and is intended to help the understanding of the official English version.

AIAG Copyright and Trademark Notice:

The contents of all published materials are copyrighted by the Automotive Industry Action Group unless otherwise indicated. Copyright is not claimed as to any part of an original work prepared by a U.S. or state government officer or employee as part of the person's official duties. All rights are preserved by AIAG, and content may not be altered or disseminated, published, or transferred in part of such content. The information is not to be sold in part or whole to anyone within your organization or to another company. Copyright infringement is a violation of federal law subject to criminal and civil penalties. AIAG and the Automotive Industry Action Group are registered service marks of the Automotive Industry Action Group.

© 2011 Automotive Industry Action Group

Odette Copyright Notice:

© 2011 Odette International Limited - All rights reserved.

JAMA Copyright Notice

© 2011 Japan Automotive Manufacturers Association, Inc. (JAMA)

JAPIA Copyright Notice

© 2011 Japan Auto Parts Industries Association (JAPIA)

ISBN #: 978 1 60534 238 2

FOREWORD

This Radio Frequency Identification (RFID) Item Level Standard describes best practices, processes, and methods for item[1] identification, verification, traceability, product characteristics, and Vehicle Identification Number (VIN[2]) throughout the global automotive supply chain. An extensive effort has been undertaken by the automotive industry to make data interchangeable between 2D (e.g., Data Matrix / QR Code) optical symbols and electronic media such as RFID to permit the user to select the appropriate technology with a minimum impact on IT infrastructures. These technologies complement each other and may be used jointly or separately as the application may require. This document is focused on the application of RFID to achieve these ends.

Core to achieving data interchangeability is the use of ANSI MH10-based Data Identifiers (DIs) and ISO or EPC/GS1 standards-based Data Syntax; this document will reflect the adoption of these methodologies.

This standard will also provide additional data-use details: what data to put in which Memory Bank (MB), which data syntax standards to use where, and how to use them effectively. This standard provides details on the MB01 (0x01)-centric Monomorphic Unique Item Identifier (UII), also called Birth Record (what it is and how to use it), and what data to put into MB11 (0x11) (the User Memory Bank), and how that data should be placed there. The intent is to reduce ambiguity through concise explanations and details.

Two regional documents were used by the committee to create the core of this standard: the AIAG B-11 *Item-Level Radio Frequency Identification (RFID) Standard* and the Odette recommendation *RFID for Tracking of Parts and Assemblies / VDA 5510*.

> **NOTE:** Two processes not addressed within this standard are shipping labels and returnable containers. These processes are detailed in the following automotive global documents:
> JAIF B-16 *Global Transport Label Standard for the Automotive Industry*
> JAIF RC-6 *Global Guideline for Returnable Transport Items*

•

This standard is built on these assumptions:

1. Only passive or battery-assisted passive RFID tags are used.
2. The air interface protocol is ISO/IEC 18000-63 previously known as ISO/IEC 18000-6, Type C / GS1 UHF Gen 2.
    a. With trading partner agreement, only ISO/IEC 18000–3, Mode 3 (ASK) / GS1 HF Gen 2 (ASK) may be used.
    b. For the purposes of this document, ISO/IEC 18000-6, Type C / GS1 UHF Gen 2 shall be referred to as "UHF" and ISO/IEC 18000–3, Mode 3 (ASK) / GS1 HF Gen 2 (ASK) shall be referred to as "HF".
    c. For the purposes of this document, references to ISO/IEC 18000-63 also shall be applicable to ISO/IEC 18000-3, Mode 3 (ASK).
3. The data syntax is ISO/IEC 15962, or ISO 17367; Data Identifier (DI)-based.
4. This document also addresses GS1 SGTIN-96.

[1] Within this document, the terms *item, product, part, component, module*, and *assembly* are synonymous.

[2] VIN and vehicle identification are synonymous within this document.

In this document, the word “shall” indicates a requirement and the word “should” indicates a recommendation.

It is the supplier’s responsibility to provide RFID tags that meet this standard. Strict adherence to these specifications for RFID tags for item-level identification will reduce implementation costs and increase benefits throughout the industry.

Various number-type designators are available and have been used in various documents;

- 17**hex**, 17**h**, $17_{\mathbf{H}}$, $17_{\mathbf{h}}$, or **x**17 is used to denote HEX-based data.
- 15**b** or $15_{\mathbf{2}}$ is used to denote BINARY-based data.
- $10_{\mathbf{10}}$ is used to denote DECIMAL-based data.

> **NOTE:** For this document, the following ISO-based designators shall be used:

- **Decimal** numbers shall be shown with only the written number, without a radix; e.g., ‘123456’.
- For **hexadecimal** numbers, the prefix ‘0x” shall be used to the left of the number; e.g., ‘0x31”.
- For **binary** numbers, the prefix ‘0b” shall be used to the left of the number; e.g., 0b1011.

> **NOTE:** Long strings of binary numbers shall not be prefixed; however, the text shall clearly state “the binary (equivalent) (expression) (value) is as follows:”
>
> **NOTE:** There are a number of terms that are used frequently in this document: Their proper designation is Memory Bank 0b01 (MB01), Memory Bank 0b10 (MB10), Memory Bank 0b11 (MB11), PC Bit 0x15 (PC Bit 15), and PC Bit 0x17 (PC Bit 17). In this document, these terms shall be used as indicated within the parentheses; i.e., MB01, MB10, MB11, PC Bit 15, and PC Bit 17.
>
> **NOTE:** Spaces have been added to binary and hex data for clarity ONLY**.**

In this document, single control characters (ISO 646 and 6-bit encoding) shall be represented as <control character>. For example, $^{G}{}_{S}$ shall be shown as <GS>, $^{E}O_{T}$ shall be shown as <EOT>, and $^{R}{}_{S}$ shall be shown as <RS>.

Bold used within this document is only for emphasis and does not indicate a requirement. For example, Data Identifiers (DIs) **25S** and **I** are bolded to have them stand apart from the data.

ACKNOWLEDGEMENTS

AIAG	James Akright	General Motors LLC
	Dennis Barlow	AIAG Volunteer
	Mary Kay Blantz	E-Business Consulting, LLC
	Morris Brown	Chrysler Group, AIAG Loaned Executive
	Jerry Czernel	AIM Computer Solutions, Inc.
	James Graham	General Motors LLC
	Larry Graham	LG AutoID, LLC (Co-Chair)
	Craig K. Harmon	QED Systems
	Bill Hoffman	Hoffman Systems LLC
	Dan Kimball	SRA International
	Pat King	Michelin North America
	Steve Lederer	The Goodyear Tire & Rubber Company
	Marilyn Smith	General Motors LLC
	Gary Tubb	Unique RFID LLC
	Henry T. Ubik	Ford Motor Company
	Paul Wilson	Bridgestone Firestone N.A. Tire, LLC
	Akram Yunas	AIAG
	Jim Zamjahn	AIAG
ODETTE	John Canvin	Odette
	Christian Daller	SKF GmbH
	Marc Hammer	Michelin
	Olle Hydbom	Auto ID Expert, Scandinavia
	Sten Lindgren	Odette Sweden
	Markus Sprafke	Volkswagen Group
	Bob Van Broeckhoven	AB Volvo (Co-Chair)
JAMA	Hidemasa Ohshika	Toyota Motor Corporation
	Yoshikazu Shiozawa	Toyota Motor Corporation
	Nobuyuki Mitsuhashi	Japan Automobile Manufacturers Association, Inc.
	Takanao Ochiai	Fujitsu Limited
	Shigeru Takahashi	Fujitsu Limited
	Junko Tatematsu	Fujitsu Limited
JAISA	Akira Shibata	Denso Wave

TABLE OF CONTENTS

FIGURES

TABLES

1 SCOPE

This standard is based on ISO and GS1 standards. They ensure compatibility between readers and tags using Issuing Agency Codes from UN, OD, LA, VTD, D, and GS1.

This global standard recommends the basic features of data carriers as applied to an item, product, part, component, module, or assembly. In particular, this standard:

- Provides recommendations for the identification of "items":
  - As used in this document, the terms *item*, *product*, *part*, *component*, *module*, and *assembly* are synonymous terms. For descriptions, see **Product** in Section 3.
- Specifies the air interface standards required between the RF interrogator and RF tag.
- Specifies the semantics and data syntax to be used.
- Provides a unique identifier for traceability.
- Specifies the minimum RFID system performance requirements.
- Specifies a minimum User Memory Bank (MB11) size.
- Specifies the process to be used to interface with business applications and the RFID system.
- Provides specific business process application recommendations for:
  - Item Identification
  - Verification (error proofing)
  - Item Traceability
  - Item Characteristics
  - Vehicle Identification (VIN)
  - Anti-counterfeiting

2 NORMATIVE REFERENCES

For undated references, use the most recent published version. For dated references, use the most recent dated release if the date is newer than that referenced below.

49 CFR 574.5 Tire Identification Requirements.
AIAG B-4 *Parts Identification and Tracking Application Standard.*
AIAG B-11 *Item Level Radio Frequency Identification (RFID) Standard*
ANS MH10.8.2 *Data Identifier and Application Identifier Standard.*
DoD suppliers passive RFID info guide v14.pdf http://www.acq.osd.mil/log/rfid/guide/
GS1 Tag Data Standard
GS1 Radio-Frequency Identity Protocols Class-1 Generation 2 UHF RFID Protocol for Communications at 860 MHz – 960 MHz Version 1.2.0 (in this document, referred to as GS1 UHF Gen 2).
ISO/IEC 15418: *Information technology — Automatic identification and data capture techniques – GS1 Application Identifiers and ASC MH 10 Data Identifiers.*
ISO/IEC 15434: *Information technology — Automatic identification and data capture techniques – Syntax for high capacity ADC media.*
ISO/IEC 15459-1: *Information technology — Automatic identification and data capture techniques – Unique identification – Part 1: Individual transport units.*
ISO/IEC 15459-2: *Information technology — Automatic identification and data capture techniques – Unique identification – Part 2: Registration procedures.*
ISO/IEC 15459-3: *Information technology — Automatic identification and data capture techniques – Unique identification - Part 3: Common rules for unique identifiers.*
ISO/IEC 15459-4: *Information technology — Automatic identification and data capture techniques – Unique identification – Part 4: Individual products and product packages.*
ISO/IEC 15459-5: *Information technology — Automatic identification and data capture techniques – Unique identification – Part 5: Individual returnable transport items (RTIs).*
ISO/IEC 15459-6: *Information technology — Automatic identification and data capture techniques – Unique identification – Part 6: Groupings.*
ISO/IEC 15961: *Information technology — Radio frequency identification (RFID) for item management — Data protocol: application interface.*
ISO/IEC 15961-2: *Information technology — Radio frequency identification (RFID) for item management: Data protocol — Part 2: Registration of RFID data constructs.*
ISO/IEC 15961-3: *Information technology — Radio frequency identification (RFID) for item management — Data protocol — Part 3: RFID data constructs.*
ISO/IEC 15962: *Information technology — Radio frequency identification (RFID) for item management — Data protocol: data encoding rules and logical memory functions.*
ISO/IEC 15963: *Information technology — Radio frequency identification for item management — Unique identification for RF tags.*

ISO 17367: *Supply chain applications of RFID — Product tagging.*
ISO/IEC 18000-63 (18000-6C): *Information technology — Radio frequency identification for item management — Part 63: Parameters for air interface communications at 860 MHz to 960 MHz Type C.*
ISO/IEC 18046: *Information technology — Automatic identification and data capture techniques -- Radio frequency identification device performance test method.*
ISO/IEC 18047-6: *Information technology — Radio frequency identification device conformance test methods -- Part 6: Test methods for air interface communications at 860 MHz to 960 MHz.*
ISO/IEC 19762: *Information technology — Automatic identification and data capture techniques — Harmonized vocabulary (all parts).*
JAIF B-16 *Global Transport Label Standard for the Automotive Industry.*
Odette LR03 - *RFID for Tracking Parts and Assemblies / VDA 5510.*
Register of NEN assigned Issuing Agency Codes for ISO/IEC 15459: http://www.nen.nl/web/Normenontwikkelen/ISOIEC-15459-Issuing-Agency-Codes.htm

3 TERMS AND DEFINITIONS

Many terms and definitions associated with the subject of this standard have special meaning. Definitions of other related terms used in this document can be found in the documents referenced in Section 2.

Table 1: Terms and Definitions

TERM	DEFINITION
AFI	**A**pplication **F**amily **I**dentifier stored in the second byte of the Protocol Control Word (0x18 to 0x1F)
Alphanumeric	A character set that contains alphabetic characters (letters) and numeric digits (numbers) and usually other characters such as punctuation marks.
ANS **ANSI**	**A**merican **N**ational **S**tandards **I**nstitute document prefix.
ANS MH10	Unit Loads & Transport Packages committee under ANSI.
ANSI MH10.8	Coding and Labeling of Unit Loads subcommittee under ANS MH10.
Antenna	The conductive element that radiates and / or receives radio frequency energy to and from the **Tag.**
Assembly	Within this document, the terms *item, product, part, component, module,* and *assembly* are synonymous. See **Product.**
Assigned Relative OID	See **OID**.
Battery-assisted Passive RFID Tag	A tag that uses a battery to improve its functionality and range and functions as a passive tag if the battery is depleted.
Binary	A numbering system with only two values: 0 (zero) and 1 (one). Mathematical base 2, or numbers composed of a series of zeros and ones. Represented by 0bX, Examples: MB0b01, 0b011

TERM	DEFINITION
Birth Record	The result of creating a unique item identifier (UII) for an item and programming that UII into MB01 of the RFID tag for the item to which the RFID tag is attached. Once programmed, the UII shall remain unchanged in the RFID tag and shall remain uniquely associated to the item for the life of the item. The RFID tag shall remain physically attached to the item for the life of the item or process (e.g., "Smart Labels" as described in Appendix A). When PC Bit 17= "0": The UII shall consist of the appropriate GS1-serialized individual item data construction (e.g., SGTIN, GRAI, GIAI, etc). When PC Bit 17 = "1": The UII shall consist of the DI (Data Identifier) and IAC (Issuing Agency Code) and CIN (Company Identification) and SN (Serial Number). Serialized identification enables traceability. Within the Serial Number field, traceability can be achieved by concatenating such data elements as part number, batch / lot numbers, asset type, etc., along with a serial number.
Bit	Short for **B**inary Dig**it**; the smallest unit of data in a computer; i.e., "1", "0". Shown as 0b1 and 0b0.
Byte	There are 8 bits in a byte; usually data are stored or instruction sets are made up of bit multiples called bytes.
Company Identification Number (CIN)	As used in this document, the manufacturer or the owner of the item. It consists of NEN-assigned company identity (see **IAC**, below) or GS1-assigned Company Prefix.
Character	The smallest group of elements that represents one number, letter, punctuation mark, or other information.
Component	Within this document, the terms *item, product, part, component, module,* and *assembly* are synonymous. See **Product.**
CRC-16	**C**yclic **R**edundancy **C**heck, stored in the first word of MB01 (from bit 0x00 to 0x0F).
Customer	In a transaction, the party that receives, buys, or consumes an item or service.
Data Field	A message consisting of a Data Identifier immediately followed by its associated data.
Data Format	Letters and numbers used to denote the type of data allowed within the referenced data field and the total quantity of that type of data allowed in the data field. Examples: "an...6" means up to 6 characters of alphanumeric data are allowed. "n…12" means up to 12 characters of numeric-only data are allowed. "an6" means that 6 characters of alphanumeric data are required. "n12" means that 12 characters of numeric-only data are required.

TERM	DEFINITION
Data Identifier (DI)	A specified character string that defines the specific data that immediately follow as defined by ANS MH10.8.2.
Decimal	Numbered or preceding by tens; based on ten; the decimal system. A base 10 numbering system, whose numbers are represented by X when a decimal number needs to be denoted from a hexadecimal number (hexadecimal = 0xX).
DSFID (Data Storage Format IDentifier)	One-byte indication, conforming to ISO/IEC 15962, of how data are structured in the tag memory. In ISO/IEC 18000-6 Type C-based parts, it shall reside in the first byte location within the User (MB0b11).
EPC	The **E**lectronic **P**roduct **C**ode™, a globally unique serial number that identifies an object in the supply chain. EPC numbers may be associated to individual objects or to collections of items like cases and pallets.
GS1, Inc. (GS1)	A consensus-based, not-for-profit standards organization created as a joint venture between GS1 (formerly known as EAN International) and GS1 US™ (formerly the Uniform Code Council, Inc. ®).
Global Trade Identification Number - GS1 (GTIN)	GTIN describes a family of GS1 global data structures that employ 14 digits and can be encoded into various types of data carriers.
HEX Hexadecimal	A base-16 number system. The hexadecimal numbers are 0-9 and then the letters A-F, whose numbers are represented by 0xX when a hexadecimal number needs to be denoted from a decimal number (underscored characters = hex indicator; decimal = X).
High Frequency (HF)	That band of radio frequencies encompassing 3 MHz to 30 MHz. For the purpose of this standard, the frequency 13.56 MHz.
IAC	**I**ssuing **A**gency **C**ode.
ID	Abbreviation for Identification.
IEC	**I**nternational **E**lectrotechnical **C**ommission. International standards and conformity assessment for government, business, and society for all electrical, electronic, and related technologies.
Interrogator	See **Reader** and **Reader / Writer**.
ISO	**I**nternational **O**rganization for **S**tandardization. ISO is a network of the national standards institutes of 156 countries on the basis of one member per country. A Central Secretariat in Geneva, Switzerland, coordinates the system.
ISO/IEC	Represents work done and / or supported by both the ISO and IEC organizations.
ISO/IEC 18000-63	Synonymous with ISO/IEC 18000-6C
Item	Within this document, the terms *item, product, part, component, module,* and *assembly* are synonymous. See **Product.**

TERM	DEFINITION
Length of data	The length is stored in the Protocol Control word byte, in MB01, in bits 0x10 to 0x14, and indicates the number of 16-bit "words" in the UII data. The CRC-16 and PC Word of MB01 are excluded from this length. See also **Word.**
Life	The defined period for which the **Tag** is expected to be functional.
Lock / Locking **Lock** **Unlock**	The ability of the **Tag** protocol to cause a memory bank to be "read-only." Once "locked," that memory location can only be changed back to "read / write" by using the lock/unlock password.
LSB	**L**east **S**ignificant **B**yte
lsb	**l**east **s**ignificant **b**it
Manufacturer	Actual producer or fabricator of an item; not necessarily the supplier in a transaction. Best practice should include site-specific manufacturing identification for use in traceability.
Memory Bank (MB)	ISO/IEC 18000-63 - and GS1 TDS 1.4 or later-compliant tags have four (4) individually usable memory banks, numbered Memory Bank 00 (binary) through Memory Bank 11 (binary). Reference Figure 3.
Module	Within this document, the terms *item, product, part, component, module,* and *assembly* are synonymous. See **Product.**
Monomorphic UII	When PC Bit 17 of MB01 = 0b0, under the rules of GS1 TDS 1.4 or later, all of the features of a Monomorphic UII are self-declaring through the GS1 identity types, i.e., SGTIN-96. When PC Bit 17 of MB01 = 0b1, under the rules of ISO/IEC 15961-2, all of the features of a Monomorphic UII are self-declaring through the registration of the particular AFI code value used, i.e., 0xA1 or 0xA5 (hazard materials). This document references ISO 17367. Therefore, the Item Level identification data consist of a Monomorphic UII whose recommended maximum size is 35 characters. The Monomorphic Item Level UII shall not contain a DSFID. For ISO data representations, see section 5.2.1.2, and for GS1 data representations, see section 5.2.1.5.
MSB	**M**ost **S**ignificant **B**yte
msb	**m**ost **s**ignificant **b**it
NCAGE	**N**ATO **C**ommercial **and G**overnmental **E**ntity code
Object IDentifier (OID)	A unique, specially formatted number that is composed of a most significant part assigned by an internationally recognized standards organization to a specific owner and a least significant part assigned by the owner of the most significant part. For example, the unique alphanumeric / numeric identifier registered under the ISO registration standard to reference a specific object or object class.
Part	Within this document, the terms *item, product, part, component, module,* and *assembly* are synonymous. See **Product**.

TERM	DEFINITION
Part Serial Number (PSN)	Supplier-managed part serial number. In this document, it is a subset of the Serial Number component of the DI and IAC and CIN and SN data structure of the DI “25S”.
Passive RFID Tags	**Tags** are externally powered, usually by the carrier signal radiated from the interrogator’s antenna; usually containing no internal power source.
Permalock	The ability of the **Tag** protocol to cause a memory bank to be “read-only.” Once “locked,” that memory location cannot be unlocked.
Product	**Within this document, the terms *item, product, part, component, module,* and *assembly* are synonymous.** The result of a process. **NOTE 1:** There are four generic product categories, as follows: — services (e.g., transport); — software (e.g., computer program, dictionary); — hardware (e.g., engine mechanical part); — processed materials (e.g., lubricant). Many products comprise elements belonging to different generic product categories. Whether the product is then called *service, software*, *hardware,* or *processed material* depends on the dominant element. For example, the offered product “automobile” consists of hardware (e.g., tires), processed materials (e.g., fuel, cooling liquid), software (e.g., engine control software, driver's manual), and service (e.g., operating explanations given by the salesman). **NOTE 2:** Service is the result of at least one activity necessarily performed at the interface between the supplier and customer and is generally intangible. Provision of a service can involve, for example, the following: — an activity performed on a customer-supplied tangible product (e.g., automobile to be repaired); — an activity performed on a customer-supplied intangible product (e.g., the income statement needed to prepare a tax return); — the delivery of an intangible product (e.g., the delivery of information in the context of knowledge transmission); — the creation of ambience for the customer (e.g., in hotels and restaurants). Software consists of information and is generally intangible and can be in the form of approaches, transactions, or procedures. Hardware is generally tangible and its amount is a countable characteristic. Processed materials are generally tangible and their amount is a continuous characteristic. Hardware and processed materials often are referred to as “goods.”

TERM	DEFINITION
Protocol Control (PC) Word or Protocol Control Bits	The PC-word (16 bits) starts at memory location 0x10 and contains the following: • Length of UII stored in bits 0x10 to 0x14 (see **Length of data);** • User Memory indicator in bit 0x15; • Extended Protocol Control indicator in bit 0x16; • Numbering System Identifier (NSI) indicator in bit 0x17; and • Attribute / AFI stored in bits 0x18 to 0x1F.
Radio Frequency Identification (RFID)	Systems that read and / or write data to RF **Tags** that are present in a radio frequency field projected from RF reading / writing equipment.
Read	The excitation, extraction, decoding, and presentation of data from a **Tag**.
Read Range	The range at which an RFID system may reliably read from desired **Tags** under defined conditions.
Reader	The electronic device that extracts and separates the information from the **Tag** using a defined protocol.
Reader / Writer	The electronic device that can change the contents of a read / write **Tag**, according to a protocol, while the **Tag** is attached to its object.
Serial Number	A number indicating place in a series and used as a means of identification.
SGTIN	**S**erialized **G**lobal **T**ransport **I**dentification **N**umber. The SGTIN is a serialized version identifying individual items within a class. Also see **GTIN**.
Supplier / Vendor	In a transaction, the party that produces, provides, or furnishes a product or service.
Supplier / Vendor ID	The numeric or alphanumeric code used to identify the supplier / vendor. See **CIN**.
Tag	For the purposes of this standard, "**Tag**" refers to the RFID transceiver, plus the information storage mechanism, attached to an object and read-from / written-to using a **Reader**, or **Reader / Writer**.
Tag ID (TID)	**Tag ID** memory bank (MB10). Globally unique identification of the individual silicon chip. Shall be serialized and locked by the silicon manufacturer.
Traceability	Sometimes referred to as genealogy, history, or birth record of an item. Traceability shall answer at minimum the following four questions: the What, the Who, the Which, and the When. Example: What = part number; Who = manufacturer or assembler; Which = lot, batch, or serial number; When = date of production or assembly.
Ultra High Frequency (UHF)	That band of radio frequencies encompassing 300 MHz to 3 GHz. For the purposes of this standard, the frequencies from 860 MHz to 960 MHz.

TERM	DEFINITION
UII	**U**nique **I**tem **I**dentifier: Descriptor for data that reside within the UII Memory Bank (MB01) of an ISO/IEC 18000-63 / GS1 UHF Class 1 Gen 2 RFID tag. As used in this document, the terms *Birth Record* and *UII* are synonymous.
VIN **Vehicle Identification Number**	The VIN (sometimes incorrectly referred to as "VIN number") is a unique serial number used by the automotive industry to identify individual motor vehicles (also applies to towed vehicles, motorcycles. and mopeds as defined in ISO 3833).
Vendor	See **Supplier / Vendor ID**.
Verification	Confirmation by examination and provisions of objective evidence that specified requirements have been fulfilled; the act of reviewing, inspecting, testing, checking, auditing, or otherwise establishing and documenting whether or not items, processes, services, or documents conform to specified requirements.
Word	Data block of two bytes (16 bits) length. The Length of Data is stored in the Protocol Control Word and expresses the number of words. See also **Length of Data.**
Write	The process of transferring data to the **Tag**; the **Tag**'s internal operation of storing data. It may include reading the data to verify accuracy.
Write Range	The range at which an RFID system may reliably write to desired **Tags** under defined conditions.

4 INTRODUCTION

4.1 Positioning of RFID in the Automotive Environment

A fundamental benefit of RFID technology is to optimize processes by using automatic item identification. Not only are classes distinguishable (e.g., parts with the same part number) as in the past, but individual entities of a class (e.g., airbag module with part number 123 and part serial number 234) are also individually distinguishable. Processes can be controlled more precisely and in tighter control cycles through better differentiation.

RFID technology has been applied for decades in the automotive industry, primarily in closed loop systems, which serve exclusively for the use of internal company processes.

These in-house 'closed loop RFID applications' are limited in number and scope and are elementary in nature. They may also be part of a larger process in an open loop system.

The real potential of using RFID occurs when the different elements of the value chain use the same standards and technologies and information is processed at every stage in an open loop system where information is shared among trading partners. (See Figure 1)

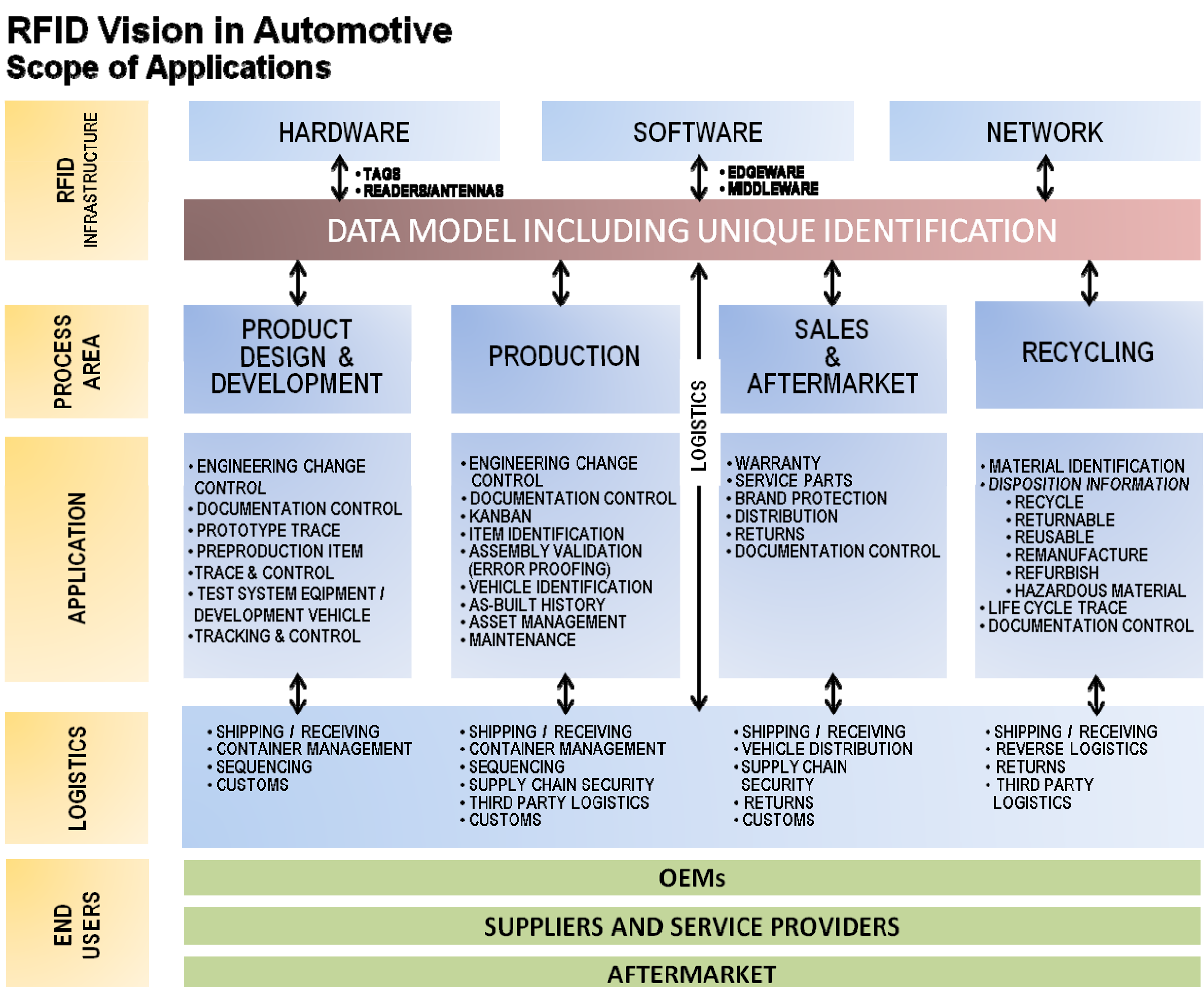


Figure 1: RFID Vision in Automotive

The use of a tag throughout an enterprise's processes has significant business benefits. For example, a tag in a component can support the following additional functions:

- Production documentation control.
- If an internal sequence is produced, the assignment of a sequence number to the part can take place.
- The correct installation of the assembly into the vehicle can be ensured through comparison of code number/order number with the customer order/build sequence/position.
- During the installation of the parts, part identification numbers together with the vehicle data can be automatically recorded and saved. If problems are later discovered in the production batch from which the part originated, targeted re-calls are possible with a minimum of disruption and cost.
- In the case of a warranty claim, it is possible to check whether the original part from the factory is still fitted.
- Optimization of process control within the replacement parts chain.
- For quality assurance purposes, safety-critical parts can be scanned to guarantee that the quality assurance requirements have been adhered to as specified in the supplier warranty documentation.
- In recycling operations, the material content can be determined from the data on the tag, which simplifies the sorting process.

The co-operative interaction within the whole value chain can only be realized when the required hardware components, data transmission protocols, system interfaces, and data structures are operated between the participating process partners in accordance with commonly defined and applied standards.

For additional general information on RFID systems implementation and operations, please see the ISO/IEC TR24729 - *Part 1: RFID-enabled labels and packaging supporting ISO/IEC 18000-63*.

RFID provides the automotive industry with a technology that operates exclusively in material flow and production control, i.e., it serves the synchronous representation of material flow and the flow of information. Precisely when a part / assembly is physically available, the accompanying information is also available and vice versa. This is why RFID cannot substitute permanently for the use of back-end systems and their linking via EDI processes. EDI data streams "hurry ahead of the product" and can also be used for planning. Back-end systems ensure a central view of the data. Therefore, particularly in the exchange between the partners in the supply chain, EDI information in parallel with the RFID-supported material flow cannot be abandoned for the foreseeable future.

For this standard, item processes and assembly processes between trading partners were reviewed. The individual processes are described within the Odette LR03 - *RFID for Tracking Parts and Assemblies* / *VDA 5510* documents. The data fields to be stored on the tag are described in Section 5 **Data Structures** of this B-21 document. Special attention has been directed to the encoding system as well as to the requirements for the tag and reader technology.

4.2 RFID; GENERAL

Only read / write, passive (or battery-assisted passive) RFID Tags shall be used, with additional specifications as outlined in the RFID Specifications (Section 7) of this standard.

All performance characteristics noted in the following RFID sections, and in Section 7, are for the tag and antenna / interrogator system **only**. They do not take into consideration host data-acquisition time, legacy-system latency time(s), or other human- or host-system-induced limiting factors.

With recent revisions to ISO/IEC 18000-63 and the GS1 Tag Data Standards Version 1.5 or later, the global RFID-using community now has globally accepted Standards to enable meeting the needs of both the OEMs and the supply chain (retail segments included) on the same tag at the same time.

With trading partner agreement only, ISO/IEC 18000–3, Mode 3 (ASK) / GS1 HF Gen 2 (ASK) may be used.

4.2.1 RFID Data Fields and Data Identifiers

Only one, monomorphic, UII shall be used in Memory Bank 0b01 (MB 0b01).

> **NOTE:** Once a term has been fully explained and a presumptive qualifier is used, the binary number shall not be prefixed; e.g., Memory Bank 0b01 (MB01). Once fully explained (as in the previous sentence), it shall be referred to thereafter as MB01.

Every data field encoded into MB0b11 shall consist of a Data Identifier (DI) followed by its associated data.

Only Data Identifiers complying with ANS MH 10.8.2 (reference ISO/IEC 15418) shall be used.

No data associated with the DIs shall exceed the maximum character length as defined in this standard or in ANS MH10.8.2.

> **NOTE:** All Data Identifiers, including the 5N series, are codified in the ANSI MH10.8.2 Data Identifier and Application Identifier Standard. This standard, and the Assigned Relative OID DI Table, can be found at: http://www.autoid.org/ANSI_MH10/ansi_mh10sc8_wg2.htm.

See Section 5.2 for MB01 data construction details and Section 5.3 for MB11 data construction details.

4.2.2 Using Data Fields in MB11

Whenever any data fields are used in MB11, Data Identifier usage shall be as described in Section 5.3.

4.3 AIDC Link to EDI

Automatic identification and data capture technologies (AIDC), e.g., barcodes, two-dimensional symbols, and radio frequency identification are a natural enhancement to electronic data interchange (EDI). EDI documents (or transactions) serve as the means by which trading partners communicate supply chain data with one another, from the purchase order, release schedule, then to the ship notice, delivery receipt, invoice, and material usage data. Of these, the Ship Notice/Manifest (a.k.a. ASN), captured in the ANSI ASC X12 856 or the UN/EDIFACT DESADV, provides detailed information on product that is shipped from a supplier to a customer.

However, when product arrives at the receiving dock, a need exists to link the received physical product with the previously received EDI data. This is where AIDC comes in, where the container transaction license plate serial number is read (refer to JAIF B-16 *Global Transport Label Standard for the Automotive Industry*), which in turn accesses the associated information in the received EDI file. Products can be received by serial number or product code, quantity, and purchase order reference.

5 DATA STRUCTURES

5.1 Reasons for and Use of the Data Structure

The data structure described here is based on the existing logistics process within the automotive industry and current standards. It contains data fields that are selected or modified in the various logistics processes.

Portions of MB11 can be locked. The remaining data fields can be optionally overwritten within the process flow as many times as necessary to ensure that they are always up to date.

5.1.1 Data Organization According to ISO/IEC 18000-63

The fundamental structure of the data on the tag is defined in the "air interface" according to ISO/IEC 18000-63. With trading partner agreement only, ISO/IEC 18000–3, Mode 3 (ASK) / GS1 HF Gen 2 (ASK) may be used.

ISO/IEC 18000-63 and ISO/IEC18000-3 Mode 3 (ASK) use identical data structures.

For the purposes of this document, ISO/IEC 18000-63 and GS1 UHF Gen 2 are the same. For the purposes of this document, 18000-3 Mode 3 (ASK) and GS1 HF Gen 2 (ASK) are the same.

If "user data" are stored on the tag in MB11, this shall be indicated on the tag according to ISO/IEC recommendations. ISO/IEC 18000-63 starts from a logical subdivision of the tag memory into four data segments, which are shown in Figure 2.

5.1.2 Data Structure on the Tag (Air Interface)

ISO/IEC 18000-63 assumes a logical division of the tag storage into four data segments, which are represented in the diagram below:

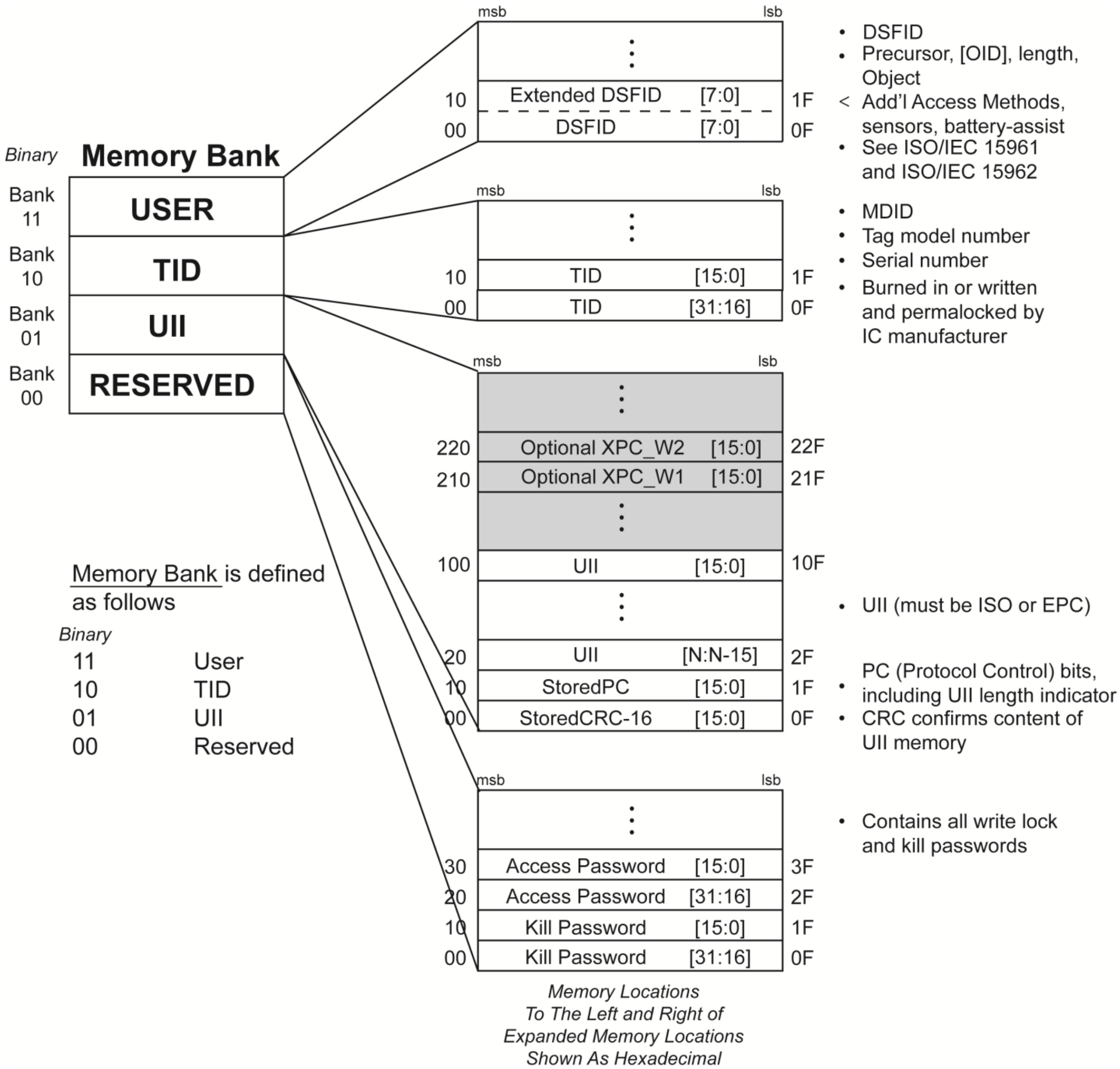


Figure 2: Memory Structure of ISO/IEC 18000-63 RFID Tag

Note: Shaded areas are not recommended by this standard.

The essential contents of these segments are:

5.1.2.1 **Reserved:**

MB00 = Password management

- Access password
- Kill password

5.1.2.2 **UII: (See Section 5.2 for additional details.)**

MB01 = UII (Unique Item Identifier);

- CRC: calculated checksum on the tag for data verification
- PC: contains several data fields, including:
  - Length of the UII field in words (one word equals two bytes)
  - Bit 15: Switch; whether user data are saved or not in MB11
  - Bit 17: Switch; if an EPC or an ISO number stands in the UII field. See Table 2.
  - Characteristic of the AFI field (Application Family Identifier)
- UII field: contains the unique part ID
  - This standard requires that the UII field shall not exceed 240 bits.

The UII shall be constructed according to GS1 SGTIN-96 or ISO/IEC 15459-4. This standard supports that both data syntaxes may exist within the same tag population environment.

The contents of MB01 have the capability of being locked. It is recommended that the UII, once programmed, be locked.

> **NOTE:** It must be noted that, when the UII memory bank (MB01) has been locked (and the password is not available) or Permalocked, software MAY NOT BE ABLE TO CHANGE the state of PC Bit 15 (User Memory indicator) of MB01. To protect the standards-based functionality of PC Bit 15, it is strongly recommended that BEFORE MB01 is locked or Permalocked, and if there are no data currently within MB11, a datum (0xFE is recommended) be programmed into the first byte of MB11 (as a placeholder), and that PC Bit 15 of MB01 be changed to "1". Then the UII can be locked or Permalocked and MB11 can still be used correctly.

The following four conditions can be represented via the assignment of PC Bits 15 and 17 in MB01:

Table 2: Usage of MB11 in an ISO/IEC 18000-63 Tag

		MB01: Protocol Control Bit 17	
		0 = EPC-based data in MB01	1 = Monomorphic ISO AFI-based data in MB01
MB01: Protocol Control Bit 15	0 = No data in MB11	- EPC-based UII Data in MB01 - No User Data in MB11	- Monomorphic ISO AFI-based UII Data in MB01 - No User Data in MB11
	1 = Data in MB11	- EPC-based UII Data in MB01 - User Data in MB11	- Monomorphic ISO AFI-based UII Data in MB01 - User Data in MB11

5.1.2.3 **TID:**

MB10 = TID (Tag ID);

- Unique part and serial number of the **Tag** (not the part to which the Tag is attached).

- Serialized and permanently locked by the tag manufacturer.

NOTE: The TID is created and programmed by the silicon manufacturer.

5.1.2.4 User Memory: (See Section 5.3 for additional details.)

MB11 = User Memory Bank:

- Data area which can be formatted and organised by the user
- Data organisation in accordance with the following data structure
  - o The structure of the data to be stored in MB11 shall contain unique Data Identifiers and related content. All fields are OPTIONAL, i.e., use of the data is determined by the specific processing rules agreed to between customer (OEM), supplier, Logistics Service Provider, and Container Service Provider. These fields are explained in detail in Section 5.3.
- MB11 has the capability of being locked.
  - o If data are ever desired to be written into MB11, and MB01 is also desired to be locked, data must be written into MB11 BEFORE MB01 is locked.

5.1.3 TID Memory Bank – MB10 (SERIALIZED AND LOCKED)

The Tag ID (TID) Memory Bank provides a number of chip identity options, called Allocation Class (AC). These options provide different tag identity profiles that can be used to uniquely, and unambiguously, identify individual RFID chips – and through the chips, the RFID tags that are made from them. See ISO/IEC 15693 for complete details on these Allocation Classes.

There are three (3) options listed under ISO/IEC 15963 that are relevant to this document: 0xE0, 0xE2, and 0xE3.

- 0xE0 is the AC for the basic ISO/IEC 7816-6 company issuer identifier codes consisting of the 8-bit identifier followed by a 48-bit serial number.
- 0xE2 will contain serialization if both of the following conditions are met:

  1. Bit 0x08 of the TID memory bank (the msb of the TAG MDID) has a value of ”1”.
  2. Bits 0x20-0x22 do not equal zero when treated as a 3-bit unsigned number. The msb of this number is bit 0x20.

  If condition “1”, above is met, there is a 16-bit XTID header present from address 0x20-0x2F of the TID memory bank. There may also be more data present in the TID that is not covered by this specification.

- 0xE3 builds on the existing AC 0xE0 for compatibility. The AC is followed by the 8-bit I.C. manufacturer’s registration number, the 2-byte User Memory present and size data, the 48-bit unique tag ID, the 1-byte XTID, and the 15-byte XTID Header data. Table B.2 of ISO/IEC 15963 shows the TID format for Allocation Class 0xE3.

This standard requires that the chip contain a uniquely serialized TID. The serialized TID shall also be “locked.” The Tag ID (MB10) shall be serialized (globally unique) and locked by the silicon manufacturer.

5.2 Data Structure for Unique Item Identifier (MB01)

Identification of the tagged object is accomplished via the UII as outlined under ISO/IEC 18000-63.

The UII shall be "locked" according to ISO/IEC 18000-63 or GS1 UHF Gen 2 air interface protocols.

The UII should contain no more than 35 characters. With a 240-bit UII, a maximum of forty (40) 6-bit characters can be encoded.

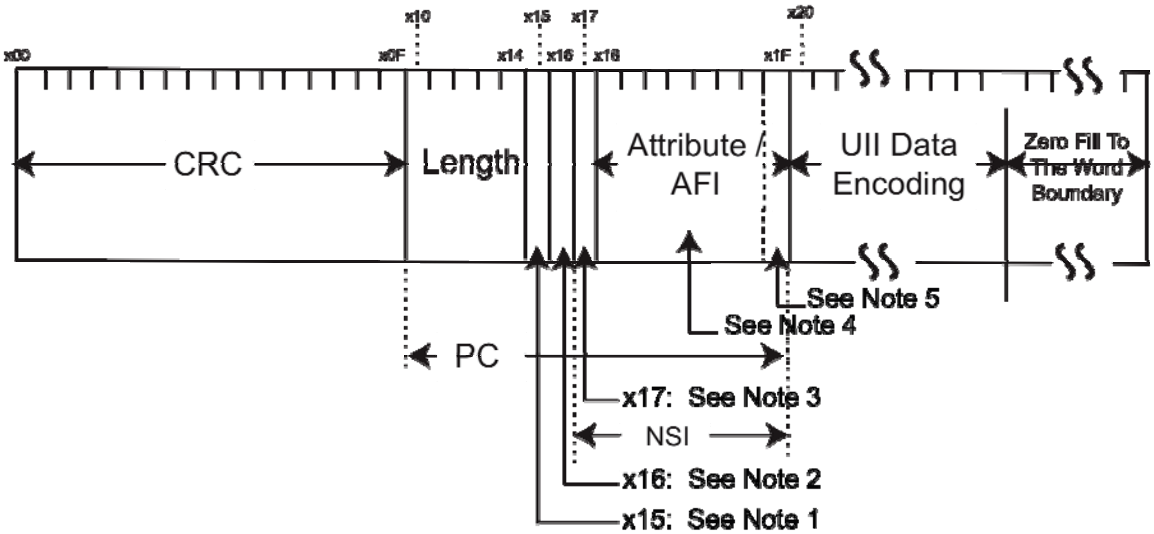


Figure 3: ISO/IEC 18000-63 MB01 layout

5.2.1 UII Coding Scheme with UN (DUNS), OD (Odette), LA (JIPDEC), VTD (TEIKOKU DATABANK), 0-9 (GS1), or D (NCAGE) format

The formatting of the data on the RFID tag shall use the "ASCII-Character-to-6-Bit-Encoding Table," according to ISO/IEC 15962 (see Table 11).

5.2.1.1 Odette (IAC = OD) Coding Scheme for Part Identification in MB01

The following table represents the data that are encoded into MB01 and contains two major sections: 1) the CRC and the Protocol Control Word, and 2) the UII.

> **NOTE:** The terms "Variable" and "Fixed" in the **Value** column of Table 3 refer to the actual data used, not the length of that data.

Table 3: Odette Coding Scheme for Parts Identification in MB01

Bit Location (HEX)	Data Type	Value	Size	Description
MB01: CRC + Protocol Control Word				
00 –0F	CRC	Hardware assigned	16 bits	Cyclic Redundancy Check
10 – 14	Length	Variable	5 bits	Represents the number of 16-bit words excluding the PC field and the Attribute/AFI field.
15	PC bit 0x15	0b0 or 0b1	1 bit	0 = No valid User Data, or no MB11 1 = Valid User Data in MB11
16	PC bit 0x16	0b0	1 bit	0 = "Extended PC word" not used
17	PC bit 0x17	0b1	1 bit	1 = Data interpretation rules based on ISO
18 – 1F	AFI	0xA1	8 bits	Application Family Identifier used according to ISO/IEC 15961 and ISO/IEC 17367.
	Subtotal		***32 bits***	
MB01: UII All UII data use 6-bit encoding values from Table 11 according to ISO/IEC 17367; not used positions are padded with leading zero(s) (ASCII "zero" [0x30]).				
Start at location 20 Go to end of data / end of available memory	DI	"**25S**"	3 an	Data Identifier for Parts Identification
	Issuing Agency Code (IAC)	"OD"	2 an	Issuing Agency Code, i.e. ,Odette
	Company Code (CIN)	As defined by the IAC	4 an	Company Identification Number
	Serial Number (SN) Consists of Part Number and Part Serial Number	Part Number	17 an	17 alphanumeric characters in capital letters
		Part Serial Number	1...8 an	Up to 8 an characters in capital letters

	Bit Padding	0b10, 0b1000 or 0b100000	2, 4 or 6 bits	Optional padding according to ISO/IEC 15962 Annex E.4 if appropriate. (4-bit padding needed)
	Word Padding	0b00000000	8 bits	Optional padding to end of 16-bit Word. (Not needed in this example)
	Subtotal		***Variable***	***Up to 208 bits***
	TOTAL MB01 BITS:		**VARI-ABLE**	**UP TO 240 BITS**

5.2.1.2 **DUNS (IAC = UN) Coding Scheme for Part Identification in MB01**

The following table represents the data that are encoded into MB01 and contains two major sections: 1) the CRC and the Protocol Control Word, and 2) the UII.

> **NOTE:** The terms "Variable" and "Fixed" in the **Value** column of this table refer to the actual data used, not the length of that data.

Table 4: DUNS Coding Scheme for Parts Identification in MB01

Bit Location (HEX)	Data Type	Value	Size	Description
MB01: CRC + Protocol Control Word				
00 – 0F	CRC	Hardware assigned	16 bits	Cyclic Redundancy Check
10 – 14	Length	Variable	5 bits	Represents the number of 16-bit words excluding the PC field and the Attribute/AFI field.
15	PC bit 0x15	0b0 or 0b1	1 bit	0 = No valid User Data, or no MB11 1 = Valid User Data in MB11
16	PC bit 0x16	0b0	1 bit	0 = "Extended PC word" not used
17	PC bit 0x17	0b1	1 bit	1 = Data interpretation rules based on ISO
18 – 1F	AFI	0xA1	8 bits	Application Family Identifier used in line with ISO/IEC 15961 and ISO/IEC 17367.
	Subtotal		***32 bits***	

MB01: UII **All UII data use 6-bit encoding values from Table 11 according to ISO/IEC 17367; not used positions are padded with leading zero(s) (ASCII "zero" [0x30]).**				
Start at 20 Go to end of data / end of available memory	DI	"**25S**"	3 an	Data Identifier for Parts Identification
	Issuing Agency Code (IAC)	"UN"	2 an	Issuing Agency Code, i.e., DUNS
	Company Code (CIN)	As defined by the IAC	9 n	Company Identification Number
	Serial Number (SN) Consists of Part Number and Part Serial Number	Part Number	17 an	17 alphanumeric characters in capital letters.
		Part Serial Number	1...8 an	Up to 8 an characters in capital letters
	Bit Padding	0b10, 0b1000 or 0b100000	2, 4 or 6 bits	Optional padding according to ISO/IEC 15962 Annex E.4 if appropriate. (6-bit padding needed)
	Word Padding	0b00000000	8 bits	Optional padding to end of 16-bit Word. (Not needed in this example)
	Subtotal		***Variable***	***Up to 240 bits***
	TOTAL MB01 BITS:		**VARI-ABLE**	**UP TO 272 BITS**

5.2.1.3 JAMA/JAPIA (IAC = LA) Coding Scheme for Part Identification in MB01

The following table represents the data that are encoded into MB01 and contains two major sections: 1) the CRC and the Protocol Control Word, and 2) the UII.

> **NOTE:** The terms "Variable" and "Fixed" in the **Value** column of this table refer to the actual data used, not the length of that data.

Table 5: JAMA/JAPIA (IAC = LA) Coding Scheme for Parts Identification in MB01

Bit Location (HEX)	Data Type	Value	Size	Description
MB01: CRC + Protocol Control Word				
00 – 0F	CRC	Hardware assigned	16 bits	Cyclic Redundancy Check
10 – 14	Length	Variable	5 bits	Represents the number of 16-bit words excluding the PC field and the Attribute/AFI field.
15	PC bit 0x15	0b0 or 0b1	1 bit	0 = No valid User Data, or no MB11 1 = Valid User Data in MB11
16	PC bit 0x16	0b0	1 bit	0 = “Extended PC word“ not used
17	PC bit 0x17	0b1	1 bit	1 = Data interpretation rules based on ISO
18 – 1F	AFI	0xA1	8 bits	Application Family Identifier used in line with ISO/IEC 15961 and ISO/IEC 17367.
	Subtotal		***32 bits***	
MB01: UII **All UII data use 6-bit encoding values from Table 11 according to ISO/IEC 17367; not used positions are padded with leading zero(s) (ASCII “zero” [0x30]).**				
Start at 20 Go to end of data / end of available memory	DI	**“25S”**	3 an	Data Identifier for Parts Identification
	Issuing Agency Code (IAC)	“LA”	2 an	Issuing Agency Code, i.e., JIPDEC
	Company Code (CIN)	As defined by the IAC	12 an	Company Identification Number
	Serial Number (SN) Consists of Part Number and Part Serial Number	Part Number	17 an	17 an characters in capital letters.
		Part Serial Number	1…6 an	Up to 6 an characters in capital letters

	Bit Padding	0b10, 0b1000 or 0b100000	2, 4 or 6 bits	Optional padding according to ISO/IEC 15962 Annex E.4 if appropriate
	Word Padding	0b00000000	8 bits	Optional padding to end of 16-bit Word
	Subtotal		***Variable***	***Up to 240 bits***
	TOTAL MB01 BITS:		**VARI-ABLE**	**UP TO 272 BITS**

5.2.1.4 **JAMA/JAPIA (IAC = VTD) Coding Scheme for Part Identification in MB01**

The following table represents the data that are encoded into MB01 and contains two major sections: 1) the CRC and the Protocol Control Word, and 2) the UII.

> **NOTE:** The terms "Variable" and "Fixed" in the **Value** column of this table refer to the actual data used, not the length of that data.

Table 6: JAMA/JAPIA Coding Scheme (IAC = VTD) for Parts Identification in MB01

Bit Location (HEX)	Data Type	Value	Size	Description
MB01: CRC + Protocol Control Word				
00 – 0F	CRC	Hardware assigned	16 bits	Cyclic Redundancy Check
10 – 14	Length	Variable	5 bits	Represents the number of 16-bit words excluding the PC field and the Attribute/AFI field.
15	PC bit 0x15	0b0 or 0b1	1 bit	0 = No valid User Data, or no MB11 1 = Valid User Data in MB11
16	PC bit 0x16	0b0	1 bit	0 = "Extended PC word" not used
17	PC bit 0x17	0b1	1 bit	1 = Data interpretation rules based on ISO

18 – 1F	AFI	0xA1	8 bits	Application Family Identifier used in line with ISO/IEC 15961 and ISO/IEC 17367.
	Subtotal		***32 bits***	
MB01: UII **All UII data use 6-bit encoding values from Table 11 according to ISO/IEC 17367; not used positions are padded with leading zero(s) (ASCII "zero" [0x30]).**				
Start at 20 Go to end of data / end of available memory	DI	"**25S**"	3 an	Data Identifier for Parts Identification
	Issuing Agency Code (IAC)	"VTD"	3 an	Issuing Agency Code, Teikoku Databank Ltd
	Company Code (CIN)	As defined by the IAC	9 n	Company Identification Number
	Serial Number (SN) Consists of Part Number and Part Serial Number	Part Number	17 an	17 an characters in capital letters.
		Part Serial Number	1..8 an	Up to 8 an characters in capital letters
	Bit Padding	0b10, 0b1000 or 0b100000	2, 4 or 6 bits	Optional padding according to ISO/IEC 15962 Annex E.4, if appropriate. (6-bit padding needed)
	Word Padding	0b00000000	8 bits	Optional padding to end of 16-bit Word. (Not needed).
	Subtotal		***Variable***	***Up to 240 bits***
	TOTAL MB01 BITS:		**VARI-ABLE**	**UP TO 272**

5.2.1.5 **NATO (NCAGE) Construct for Use with Passive Gen2 RFID Tags**

The DOD requires passive Gen2 RFID tags be attached to all shipment units at the case and pallet level. The requirement includes formatting the data in accordance with the GS1 Tag Data Specification. For those companies that do not own an EPC company manager number, or those that chose not to use their EPC company manager number, there is a "DOD Construct" a.k.a. "DOD 96" that allows use of the Commercial and Government Entity (CAGE) code in lieu of the EPC company manager number.

The North Atlantic Treaty Organization (NATO) Allied Committee 135 (AC 135) controls the issuance of NCAGE or NATO CAGE codes. The term "CAGE" nominally refers to an NCAGE code issued by AC 135 to the military of the member nation. AC 135 has published rules that ensure that NCAGE codes are individually unique. CAGE codes and NCAGE codes are the same length (5 alphanumeric characters).

Detailed instructions on how to assemble an EPC- and DOD-compliant RFID tag ID are found at the United States Department of Defense *Supplier's Passive RFID Information Guide, Version 14.0*.

How to use the UII memory area (MB01) is described in "GS1 Tag Data Standards," Version 1.5 or later.

The following table represents the data that are encoded into MB01 and contains two major sections: 1) the CRC and the Protocol Control Word, and 2) the UII.

> **NOTE:** The terms "Variable" and "Fixed" in the **Value** column of this table refer to the actual data used, not the length of that data.

Table 7: The Unique Identification (UII) of an RFID Tag (MB01) for DOD

Bit Location (HEX)	Data Type	Value	Size	Description
MB01: CRC + Protocol Control Word				
00 – 0F	CRC	Hardware assigned	16 bits	Cyclic Redundancy Check
10 – 14	Length	Variable	5 bits	Represents the number of 16-bit words excluding the PC field and the Attribute/AFI field.
15	PC bit 0x15	0b0 or 0b1	1 bit	0 = No valid User Data, or no MB11 1 = Valid User Data in MB11
16	PC bit 0x16	0b0	1 bit	0 = "Extended PC word" not used
17	PC bit 0x17	0b0	1 bit	0 = Data interpretation rules based on EPC 1 = Data interpretation rules based on ISO
18 – 1F	Attribute	0b00000000	8 bits	No AFI with EPC Format
	Subtotal		***32 bits***	

Bit Location (HEX)	Data Type	Value	Size	Description
MB01: UII **All UII data use 6-bit encoding values from Table 11 according to ISO/IEC 17367; not used positions are padded with leading zero(s) (ASCII "zero" [0x30]).**				
Start at 20 Go to end of data / end of available memory	Header	0x2F 00101111	8 bits	Header for DOD 96
	Filter Value	0b0000 or 0b0001 or 0b0010	4 bits	0000 = Pallet (palletized unit load) 0001 = Case (shipping and exterior container) 0010 = Unit Pack All other combinations = reserved for future use
	Government Managed Identifiers	"Space" followed by 5 alpha-numeric characters	48 bits	NCAGE or CAGE code preceded by a "space": Or DODAAC - 6 characters (reserved for use only to identify U.S. military activities) (Displayed in hex format – Annex I)
	Serial Number	Variable	36 bits	Binary format using left-padded zeros
	Subtotal		***96 bits***	
	TOTAL MB01 BITS:		**128 BITS**	

5.2.1.6 **UII Based on the GS1 Coding Scheme**

An EPC Identifier in SGTIN-96 format consists of the following data fields and has an overall length of 96 bits:

- **Header:** (8 bits); definition of the coding scheme, in which structure and length of the actual EPC identifier are defined (e.g., SGTIN, GRAI, etc.)
- **Filter Value:** (3 bits); provides a filtering of certain EPC identifiers by the reader, based on Header value.

- **Partition Field:** (3 bits); contains a code that indicates the number of bits in the GS1 Company Prefix field and the Indicator/Item Reference field.
- **Company Prefix:** (20 – 40 bits); assigned by GS1 to a managing entity. The Company Prefix is the same as the Company Prefix digits within a GS1 GTIN decimal code.
- **Item Reference:** (24 – 4 bits); assigned by the managing entity to a particular object class. The Item Reference for the purposes of EPC encoding is derived from the GTIN by concatenating the Indicator Digit of the GTIN and the Item Reference digits and treating the result as a single integer.
- **Serial Number:** (38 bits); assigned by the managing entity to an individual object. The serial number is not part of the GTIN code but is formally a part of the SGTIN.

The following table represents the data that are encoded into MB01 and contains two major sections: 1) the CRC and the Protocol Control Word, and 2) the UII.

Table 8: EPC Coding Scheme for Part ID (SGTIN-96)

Bit Location (HEX)	Data Type	Value	Size	Description
MB01: CRC + Protocol Control Word				
00 – 0F	CRC	Hardware assigned	16 bits	Cyclic Redundancy Check
10 – 14	Length	0b00110	5 bits	Represents the number of 16-bit words excluding the PC field and the Attribute/AFI field.
15	PC bit 0x15	0b0	1 bit	0 = No valid User Data, or no MB11 1 = Valid User Data in MB11
16	PC bit 0x16	0b0	1 bit	0 = “Extended PC word“ not used
17	PC bit 0x17	0b0	1 bit	0 = Data interpretation rules based on EPC
18 – 1F	Attribute	0b00000000	8 bits	No AFI with EPC Format
	Subtotal		***32 bits***	
MB01: UII				
Start at 20 Go to end of data /	Header	48	8 bits	Definition of coding scheme, i.e., EPC SGTIN-96
	Filter	1	3 bits	Filter value for selection

end of available memory	Partition	0 – 6	3 bits	Partitioning of the bits available for Company Prefix and Item Reference
	Company Prefix	6 -12 digits*	20 -40 bits	Company Identification Number variable length but sum of available Company Prefix and Part Number digits is fixed
	Indicator Digit and Item Reference	1 - 7 digits*	24 - 4 bits	Part Number
	Serial Number	11 digits	38 bits	A value between 0 and 274877906943
	Subtotal		***96 bits***	
	TOTAL MB01 BITS:		**128 BITS**	

***NOTE:** The maximum decimal value range of the Company Prefix and Item Reference fields varies according to the contents of the Partition field.

5.2.1.7 ISO-based Coding Scheme for Vehicle Identification (VIN) in MB01

NOTE: The coding scheme for vehicle identification is valid for all IACs using ISO coding schemes.

The following table represents the data that are encoded into MB01 and contains two major sections: 1) the CRC and the Protocol Control Word, and 2) the UII.

NOTE: The terms “Variable” and “Fixed” in the **Value** column of this table refer to the actual data used, not the length of that data.

Table 9: Coding Scheme for Vehicle Identification (VIN) in MB01

Bit Location (HEX)	Data Type	Value	Size	Description
MB01: CRC + Protocol Control Word				
00 –0F	CRC	Hardware assigned	16 bits	Cyclic Redundancy Check
10 – 14	Length	Variable	5 bits	Represents the number of 16-bit words not excluding the PC field and the Attribute/AFI field.
15	PC bit 0x15	0b0 or 0b1	1 bit	0 = No valid User Data, or no MB11 1 = Valid User Data in MB11
16	PC bit 0x16	0b0	1 bit	0 = “Extended PC word“ not used
17	PC bit 0x17	0b1	1 bit	1 = Data interpretation rules based on ISO
18 – 1F	AFI	0xA1	8 bits	Application Family Identifier used according to ISO/IEC 15961 and ISO/IEC 17367.
	Subtotal		***32 bits***	
MB01: UII **All UII data use 6-bit encoding values from Table 11 according to ISO/IEC 17367; not used positions are padded with leading zero(s) (ASCII “zero” [0x30]).**				
Start at location 20	DI	“I”	1 an	Data Identifier for Vehicle Identification
	Serial Number		17 an	Vehicle Identification number (VIN)
Go to end of data / end of available memory	Byte Padding	0b1000	4 bits	Byte padding according to ISO/IEC 15962 Annex E.4
	Subtotal		***Fixed***	***112 bits***
	TOTAL MB01 BITS:		**FIXED**	**144 BITS**

5.3 Data Structure in the User Memory Bank (MB11)

For MB11, data structures shall conform to ISO/IEC 15434 and ISO/IEC 15418 (which refers to ANSI MH 10.8.2), Data Format 06 "Data using ASC MH 10 Data Identifiers" to enable conversion between optical symbology and RFID media.

When using RFID, ISO/IEC 15962 Access Method 0 (No Directory) Format 3 or Access Method 0 (No Directory) Format 13 shall be used.

The minimum User Memory Bank (MB11) size shall be 512 bits.

5.3.1 Data Requirements

A unique identifier shall be saved in MB01 on the tag as a minimum requirement (see Section 5.2). In application Scenarios 2 and 3 in Section 6, additional user data are required. The requirements vary depending upon the business processes and agreements with the participating partners.

"User Memory" Memory Bank 11 (MB11) can take numerous forms from a data semantics and syntax perspective. When PC Bit 15 = 0b1 (indicating that there are data in MB11) and PC Bit 17 = 0b1 (indicating an ISO versus an EPC structure), the first 8 to 16 bits of user memory are encoded according to the DSFID rules in ISO/IEC 15961-1, ISO/IEC 15961-2, and ISO/IEC 15962.

This document describes, and recommends, Access Method 0 and Formats 3 or 13 for encoding data within MB11.

5.3.2 Data Storage Format Identifier (DSFID)

The single-byte DSFID (Data Storage Format IDentifier) defines the Access Method and the Data Format, and has the following structure:

- Bits 8 and 7 define the Access Method, which determines how data are encoded on the tag. See Table 10.
    - This standard recommends Access Method "No Directory," with binary bit value '00'.
- Bit 6 is the Extended Syntax indicator (recommended = "0").
    - When bit 6 = "1" the following apply:
        - Bit 8 is the Extensibility Bit
        - Bits 7 and 6 are the Extended Syntax
        - Bits 5 and 4 are the Memory Length
        - Bits 3 to 1 are the Battery-assisted, Full-function Sensor, Simple Sensor indicators
        - Subsequent bytes contain the data, in 6-bit encoding.
        - When using Format 3, the data stream is terminated by the "<EOT>" character.
- Bits 5 through 1, inclusive, describe the Data Format.
    - A DSFID value of "0x03" indicates the use of "No Directory, ISO/IEC 15434 - based Data Syntax." The SG1 standing document "SG1 Guidelines for 15962" explains the use of this method with Format 06, 6-bit encoding.
    - A DSFID value of "0x0D" indicates the use of "No Directory using the ISO/IEC 15962 Assigned Relative OID DI Table." **This is the recommended format.**

- A DSFID value of "0x8D" indicates the use of "Packed Objects, using the ISO/IEC 15962 Assigned Relative OID DI Table."

NOTE: The Assigned Relative OID DI Table can be found at: **http://www.autoid.org/ANSI_MH10/ansi_mh10sc8_wg2.htm**.

Table 10: Assigned and Reserved Access Methods

15961 integer code	15962 DSFID bit code	15962 SFF bit code	Name	Description
0	00	00	**No-Directory**	This structure supports the contiguous abutting of all the Data-Sets.
1	01	00	**Directory**	The data are encoded exactly as for **No-Directory** but the RFID tag supports an additional directory, which is first read to point to the address of the relevant object identifier.
2	10	00	**Packed-Objects**	This is an integrated compaction and encoding scheme that formats data in an indexed structure as defined by the Application administrator (see ISO/IEC 15961-2).
3	11	00	**Tag-Data-Profile**	This is an integrated compaction and encoding scheme for a fixed set of data elements, each of a defined length.
4 - 15	00	01	**RFU**	Reserved for future revisions of ISO/IEC 15962.

5.3.2.1 Access Method 0, Format 3; ISO/IEC 15961-2 Defined Encoding Assigned to ISO/IEC 15434.

Using the DSFID and Precursor, ISO/IEC 15962's Format 3 eliminates the need to encode the following portions of the ISO/IEC 15434 message envelope: the Compliance Indicator and Format Trailer "**[ ) >** <**RS**>", the Format Header (as used in this document for DIs; "**06**<**GS**>"), and the message's final Format Trailer/Message Trailer characters "**<RS> <EOT>**". Data encoding is accomplished by using the 6-bit data characters in Table 11. For further details, see ISO 17367.

Table 11: Six-bit Character Encodation Table

Character	Binary Value	Character	Binary Value	Character	Binary Value	Character	Binary Value
Space	100000	0	110000	@	000000	P	010000
<EOT>	100001	1	110001	A	000001	Q	010001
<Reserved>	100010	2	110010	B	000010	R	010010
<FS>	100011	3	110011	C	000011	S	010011

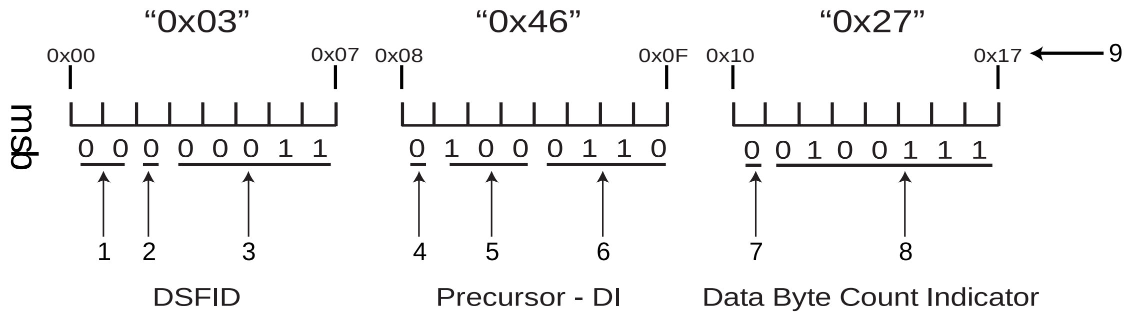
Character	Binary Value	Character	Binary Value	Character	Binary Value	Character	Binary Value
<US>	100100	4	110100	D	000100	T	010100
<Reserved>	100101	5	110101	E	000101	U	010101
<Reserved>	100110	6	110110	F	000110	V	010110
<Reserved>	100111	7	110111	G	000111	W	010111
(	101000	8	111000	H	001000	X	011000
)	101001	9	111001	I	001001	Y	011001
*	101010	:	111010	J	001010	Z	011010
+	101011	;	111011	K	001011	[	011011
,	101100	<	111100	L	001100	\	011100
-	101101	=	111101	M	001101	]	011101
.	101110	>	111110	N	001110	<GS>	011110
/	101111	?	111111	O	001111	<RS>	011111

Note: Table 11, above, is 6-bit encoding created through the simple removal of the two high-order bits from the ISO 646 8-bit ASCII character set, save the shaded values. The shaded values are re-assigned, as provided, to minimize the bit count when using the ISO/IEC 15434 envelope.

As used in this document, only the following characters from Table 11 are allowed:
0, 1, 2, 3, 4, 5, 6, 7, 8, 9, A, B, C, D, E, F, G, H, I, J, K, L, M, N, O, P, Q, R, S, T, U, V, W, X, Y, Z, * (asterisk), + (plus sign), - (dash), . (period or full stop), <EOT>, <RS> and <GS>.

5.3.2.2 **Synopsis of Access Method 0, Format 3 encoding:**

- The first byte of memory is a DSFID that always has the fixed value 0x03, indicating that memory uses Access Method 0 and encodes Format 3 (ISO/IEC 15434) messages. See ISO 17367 for encoding/decoding and translation guidance.
- The second byte of memory is a Precursor, which is a fixed value for a given ISO/IEC 15434 Format Indicator (true for all but the exception conditions covered in ISO/IEC 15962 and 15434).
  - To be compliant with this standard when using Format 3, a fixed Precursor value of 0x46 will always be used, indicating 6-bit encoding and ISO/IEC 15434 Format Indicator 6 (for DIs).
- The third byte of memory indicates the length (in bytes) of the data, encoded as an EBV-8 value. For all but the longest ISO/IEC 15434 messages (those longer than 127 bytes), this number is encoded in a single byte.
- The subsequent bytes (whose length was indicated by the preceding byte) contain the data, in 6-bit encoding (reference Table 11).
  - This standard recommends that only one 15434 message be encoded.



EXPLANATION OF NUMBERS IN FIGURE:

1 Access Method: "0" (as listed in Table 7 – ISO/IEC 15962)
2 Extend Syntax – turns on additional byte of DSFID Byte (turned off in this instance)
3 Data Format "03" (ISO/IEC 15434)
4 Extension Bit – (see Clause 9.2.8 and Annex T – ISO/IEC 15962)
5 Compaction bits (indicating 6-bit table)
6 Format Envelope (specifically DI "06")
7 Byte Count Indicator switch (set to "0" to signify final byte of byte count)
8 Bit values for Byte Count Indicator (variable based on length of data)
9 Bit position in memory

Figure 4: ISO/IEC 18000-63 and ISO/IEC 18000-3 Mode 3 Structure of Memory Bank "11"; First 16 Bits

NOTE: For the purpose of the above example, battery-assist and sensors are shown as not present.

5.3.2.3 **Access Method 0 Format 13; ISO/IEC 15961-2 Defined Encoding Assigned to ISO/IEC 15962 Assigned Relative OID DI Table**

- When using Format 13, the first byte of memory is a DSFID that always has a fixed value of 0x0D, indicating that user memory is encoded using Access Method 0 and Format 13 (ISO/IEC 15962 Assigned Relative OID DI Table).
- The second byte of memory is a Precursor, consisting of an Offset bit, Compaction code (3 bits), and Assigned Relative OID DI Table value for the DI used (4 bits).
- The third byte of memory indicates the length (in bytes) of the data, encoded as an EBV-8 value.
- The subsequent bytes (whose length was indicated by the preceding byte) contain the data, using the encoding schema as denoted within the Precursor (Integer, Numeric, 5-bit, 6-bit, or 7-bit).

The above pattern of Precursor, Length of Data, and Data is repeated for each datum written into User Memory.

6 RFID TAG DATA SCENARIOS

The tag data structures described below permit three fundamentally different scenarios of the RFID tag data handling processes; each places different demands on the total system. 'Total system' in this context means all resources necessary for the RFID-centric process, including the tag, reader, network, middleware, applications, organization, etc.

Addressing the tag data in the description of these scenarios below, there is a reference to the logical 'part ID' (UII) stored in MB01. The exact definition of this ID is explained in Section 5.2.

'User data' refers to all data stored in the User Memory Bank - MB11.

All three scenarios allow read-access to the part ID on the tag.

6.1 Scenario 1: Tag Contains UII in MB01 (locked); No Data in MB11

In this scenario there is a UII in MB01; there are no data in MB11, and the only data field that is read on the tag is the globally unique Part ID (UII). The UII shall be locked and cannot be over-written (see definition "**Lock/Locking**").

In a 'UII-only' scenario, all other information related to the part shall be accessed via the relevant back-end system or other databases.

This scenario represents a secure and expedient way of accessing data on the RFID tag.

This can be of importance when performing bulk reads on a pallet containing multiple RFID tags (e.g., via an RFID-enabled portal).

Additional "item data" shall be read in this case from the back-end system or database.

6.2 Scenario 2: Tag Contains UII in MB01 (locked) and Data in MB11 (locked)

In addition to the UII (globally unique Part ID) written to MB01, this scenario covers item-attendant data programmed into the User Memory Bank (MB11).

In this scenario, the UII (MB01) and the item–attendant user data (MB11) are written to the tag at the same time and locked.

Access to the tag data is 'read only' in all process steps.

All additional data required, beyond the data from the tag, shall be accessed via the relevant back-end system(s).

This scenario provides point-of-use access to relevant data contained within the RFID tag (e.g., color, revision level, part number, or manufacturing date).

6.3 Scenario 3: Tag Contains UII in MB01(locked) and Data in MB11 (not locked)

In addition to the UII (globally unique Part ID, written to MB01), this scenario covers item-attendant data contained in the User Memory Bank (MB11). The UII (MB01) is locked. The item-attendant user data (MB11) written to the tag are unlocked.

Whatever data need to be exchanged between partners in the supply chain relating to parts shall be defined bilaterally between the partners.

> **NOTE:** In accordance with ISO/IEC18000-63, the ability to write data to MB11 is only possible for the entire User Memory Bank and not at the individual field level. In addition, security permissions based on generic roles assigned to a specific group of users for specific data access permission (read-only, update, etc.) is not possible.

7 TECHNICAL SPECIFICATIONS FOR RFID TAGS

The specifications described below satisfy the process requirements of the automotive industry. These process requirements are included in Section 6. The technical specifications shall be strictly complied with in order to achieve an acceptable RFID system.

7.1 Tag Types

Only passive or battery-assisted passive UHF-based (specifically 860 MHz - 960 MHz) and, with trading partner agreement, HF-based (13.553 MHz – 13.567 MHz) RFID tags are used.

- Air interface shall be ISO/IEC 18000-63 for UHF and shall be ISO/IEC 18000-3M3 (ASK) for HF.
- To be compliant to ISO/IEC 18000-63, ISO/IEC 18047-6 *Compliance testing of the Air Interface* and ISO/IEC18046 *Performance test of an RFID System* shall be followed.
- To be compliant to ISO/IEC 18000-3M3 (ASK), ISO/IEC 18047-3 *Compliance testing of the Air Interface* and ISO/IEC18046 *Performance test of an RFID System* shall be followed.

7.2 Tag Memory Size

The Tag's overall minimum memory size should be no less than 1,024 bits (128 bytes).

A Tag memory size of 256 bytes (2,048 bits) or larger may be required to meet the requirements of certain applications.

A minimum UII Memory Bank (MB01) size of 240 bits is recommended for the ISO-based birth record data of up to 35 characters (6-bit encoding).

The minimum User Memory Bank (MB11) size shall be 512 bits.

7.3 Anti-collision Mechanism for Multi-reader Systems

Shall be in accordance with ISO/IEC 18000-63.

The following should also be noted upon installation of the system:

In order to exclude collision errors caused by multiple UHF readers with overlapping fields of detection, the recommendations of ETSI (European Telecommunications Standards Institute) in the Technical Report ETSI TR 102 436 *Electromagnetic Compatibility and Radio Spectrum Matters (ERM); Installation and Commissioning of RFID Systems Operating at UHF* should be followed.

7.4 Environmental Conditions

Protection mechanism for reader/writer and tag: IP67.

The IP 67 (International Protection rating) protection should be the minimum to ensure proper efficiency in the logistics/production environments subject to commercial agreements between RFID users and RFID Technology Providers. The protection that IP **67** provides is described in Figure 5.

Figure 5: Definition of IP67

6	Effective against	Dust tight. No ingress of dust; complete protection against contact	
7	Liquids	Immersion up to 1 m. Ingress of water in harmful quantity shall not be possible when the enclosure is immersed in water under defined conditions of pressure and time (up to 1 m of submersion).	Test duration: 30 minutes Immersion at depth of 1 m

Selecting protection for the tag shall also consider the following;

- Complete protection of the tag from physical damage that would cause the tag to malfunction during its intended use life, water and dust.
- Protection against adverse conditions caused by submersion in oil, coolants, water, snow, rain, container cleaning agents, etc. Limited functionality while under such adverse conditions is acceptable. Full functionality shall occur once the tag is removed from the adverse condition.
- Operational temperature range:
    - -40° to +80° Celsius
- Storage temperature range:
    - –50° to + 120° Celsius
- Appropriate measures that guarantee readability are necessary under adverse weather conditions.
- Protection for exposure to oil spray, cutting fluid, steam, etc. shall be considered on an individual basis.

7.5 Tag Location

The tag should be placed in such a position that reading with relevant reading equipment will be possible. Consideration of production processes coupled with requirements for installation of parts and any need to read the part after installation shall be considered and agreed to between trading partners.

7.6 Security / Locking Data (Permalock Command)

The RFID protocols that this standard is based on provide the ability to “lock” data through specific commands in the interrogator-based software command set.

Using the Permalock command for the UII data within MB01 shall be required.

> **NOTE:** It must be noted that, when the UII memory bank (MB01) has been locked (and the password is not available) or Permalocked, software MAY NOT BE ABLE TO CHANGE the state of PC Bit 15 (User Memory indicator) of MB01. To protect the standards-based functionality of PC Bit 15, it is strongly recommended that BEFORE MB01 is locked or Permalocked, and if there are no data currently within MB11, a datum (0xFE is recommended) be programmed

into the first byte of MB11 (as a placeholder) and that PC Bit 15 of MB01 be changed to "1". Then the UII can be locked or Permalocked and MB11 can still be used correctly.

7.7 EMI and EMC

Electromagnetic Interference (EMI) is anything electromagnetic that causes undesired responses or degradation of performance in electrical or electronic equipment. EMI can occur through conduction (electric current), radiation (electromagnetic field), inductive coupling (magnetic field), and capacitive coupling (electric field).

Surprisingly enough, EMI can cause not only “no-reads” but also “side-reads” as well (reading tags other than that at which the reader is aimed, due to reflection, spurious emissions, etc.).

Electromagnetic Compatibility (EMC) is the capability of equipment to operate in their intended environment without causing or receiving degradation due to unintentional EMI.

Useful sources of information for solving Electromagnetic Compatibility (EMC) interference problems are:

- ETSI TR 102 436 *Electromagnetic compatibility and Radio spectrum Matters (ERM); Installation and commissioning RFID systems operating at UHF*
- ISO/IEC 24729– 3, *RFID Implementation Guidelines.*

7.8 Tag Life Span

Passive tags shall, at a minimum, last for the life of the part to which they are attached.

7.9 Relative Speed Through Portals

The higher the speed of transit through the portal, the higher the probability of read errors.

Effective reading speed depends on several factors, such as:

- Location of the tag on the part, in relation to the antenna
- Distance between tag and antenna
- Location/position of antenna relative to tag
- Type of tag chip
- Size of tag antennas
- Environment (e.g., nearby metal, ambient RF activity/noise, type of material the tag is mounted to, etc.)

7.10 RFID Regulations

Tags compliant to this standard shall meet the following requirements:

- The tag shall be interrogated within the approved power levels, number of channels, channel widths, channel separations, duty cycle, and other regulatory parameters of the applicable country / region / standard in which the tag and antenna are used.

7.11 Tag Identification Mark

RF tags and RF "Smart Labels" compliant with this standard should include one or more of the internationally accepted RFID emblems. For further information on the RFID emblems, refer to ISO/IEC 29160.

Figure 6: RFID Emblem for ISO/IEC 18000-63 Using ISO/IEC 15434 and Data Identifiers

Figure 7: RFID Emblem for ISO/IEC 18000-3, Mode 3 (ASK) Using ISO/IEC 15434 and Data Identifiers

Figure 8: RFID Emblem for ISO/IEC 18000-63 Using EPC – SGTIN

Figure 9: Generic RFID Emblem Not Showing Air Interface or Data Format

8 BUSINESS PROCESS APPLICATIONS

Within the scope of this standard, the following common industry process applications have been identified. See Figure 1 for context within the RFID Vision in Automotive.

- Item Identification (See Section 8.2)
- Verification (error proofing) (See Section 8.3)
- Item Traceability (See Section 8.4)
- Item Characteristics (See Section 8.5)
- VIN (See Section 8.6)
- Anti-counterfeiting (See Section 8.7)

8.1 Application-Specific Data structures

To increase security and integrity of the data encoded in the RFID tag, the tags shall have the TID (globally unique Tag Identification of the individual silicon chip.) serialized and locked by the silicon manufacturer.

A unique identifier shall be saved in Memory Bank 01 (MB01) on the tag as a minimum requirement. In all the applications described below, Memory Bank 11 (MB11) or “User Memory” can contain additional user data and are described in Section 5.3. The requirements vary depending upon the business processes and agreements with the participating partners.

The schema detailed in the next sections shall be followed in order to support the applications described in Sections 8.2 to 8.7.

8.1.1 Summary of Tag Memory Layout

ISO/IEC 18000-63 RFID tags support four memory banks; MB00, MB01, MB10, and MB11. Germane to this document are Memory Banks MB01, MB10, and MB11.

This section and Table 12 show the data structure for MB01, the names for and contents of MB10 and MB11, and the to-be-compliant-to-this-document status of these memory banks. See Figure 2 for additional memory bank details.

Table 12: Comparison of ISO and EPC Data Structures in MB01, MB10 and MB11

	MB01 (UII)				MB10	MB11
ISO (See Table 3 for MB01 encoding)	DI	IAC	CIN	SN	TID	USER MEMORY
EPC (See Table 8 for MB01 encoding)	Header and Filter Value and Partition		Company Prefix	Item Reference and Serial Number		

- MB01 UII (Unique Item Identification) - Globally unique, locked, mandatory – user-created.
- MB10 TID (Tag Identification) - Globally unique, locked, mandatory – silicon manufacturer-created.
- MB11 (User Memory) – Optional data, optional locking.

8.1.2 Data Field Identification

The Data Identifier(s) (DIs) are defined in the application sections; see also Annex H.

If SGTIN is used, follow the requirements as detailed in Annex F.

8.1.3 Maximum Data Length

When placed in MB01, identification data should contain a maximum of 35 characters (exclusive of the Data Identifier), and shall contain a maximum of 240 bits (40 6-bit characters).

For example,

25SUN315016295000000007D094143501C283.

In the example above, the DI is 3 characters and the data are 35 characters in length.

If SGTIN is used, then follow the requirements as detailed in Annex F.

8.1.4 Character Set

As used in this document, only the following characters are allowed: 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, A, B, C, D, E, F, G, H, I, J, K, L, M, N, O, P, Q, R, S, T, U, V, W, X, Y, Z, ***** (asterisk), **+** (plus sign), **-** (dash), **.** (period or full stop), <EOT>, <RS>, and <GS> from Table 22.

8.1.5 UII (MB01) Data Structure

The acronyms used in Table 13 below are explained in detail in the sections that follow the table.

Table 13: UII (MB01) Data Structure

Identifier	Structure		
25S	IAC	CIN	SN (Consists of PN and part SN)
SGTIN-96	Header; Filter Value; Partition	Company Prefix	Item Reference and Serial Number
I	VIN		

8.1.5.1 **Issuing Agency Code (IAC)**

The Issuing Agency Code (IAC), which consists of a range from one (1) to a maximum of three (3) characters, is a code used to identify the entity / organization / company authorized by the appropriate registration authority as an issuing agency in accordance with ISO/IEC 15459-2. Any IACs registered by NEN may be used with mutual agreement between trading partners. Recommended IACs include UN (Dun & Bradstreet), OD (Odette Europe), VTD (Teikoku Databank Ltd.), LA (JIPDEC), and D (NATO AC135).

8.1.5.2 **Company Identification Number (CIN)**

The Company Identification Number (CIN) is a unique code that is assigned by the issuing agency to each individual company. Each issuing agency has its own format for the CIN.

To be in compliance with this standard, the user shall have a unique CIN allocated by the appropriate issuing agency. See Table 14.

Table 14: Data Format Examples - CIN

IAC		CIN
Odette	OD	4 alphanumeric
DUNS	UN	9 numeric
JIPDEC	LA	12 alphanumeric
TEIKOKU DATABANK LTD.	VTD	9 numeric
NATO AC135	D	Reference AC135
GS1 (SGTIN)	See Annex F	

8.1.5.3 **Serial Number (SN)**

The SN, in combination with IAC and CIN, shall constitute a globally unique identifier for the item. Once created and attached to an item, the IAC, CIN, and SN combination shall be fixed and unchangeable for that specific item throughout its lifetime.

The Serial Number (SN) shall be composed of the character set identified in Section 8.1.4.

The data structure for the Serial Number for an item should be composed, minimally, of a Part Number (PN) and a Part Serial Number (PSN), as illustrated in Table 13.

- PN – "by trading partner agreement"-assigned part number
- PSN – supplier-managed part serial number

The supplier-managed part serial number (PSN) may be constructed in many different manners such as identification of producing site, date produced, time produced, production line, final test station identification, lot, batch, heat, etc. See Annex K.

8.2 Item Identification – MB01-centric (Data Identifiers 25S or SGTIN)

The ability to identify an item at its lowest level is the part number. In addition, since the same part number may be provided by multiple suppliers, it is desirable to identify the specific source (site) that produced the item. The minimal data elements to meet item identification shall contain one of the following (also see Table 23 and Table 24):

- An ISO-defined globally unique item identity (25S) (See also Section 5.2)
- EPC Coding Scheme for Item Identification (SGTIN-96) (See Annex F)

Table 15: Item Identification Structures Used in MB01

ISO	"25S" and IAC			CIN	SN (PN and part SN)	
SGTIN-96	Header	Filter Value	Partition	Company Prefix	Item Reference	Serial Number

> **NOTE:** When DI "25S" contains the part number as a component of its construct, as defined by trading partner agreement, it is possible to parse the data to extract the part number.

> **NOTE:** Since the data are represented in binary, trading partners shall agree on how data would be represented in Human-Readable (HR) and / or within a 2D symbol if required.

Table 16: ISO-based Example of MB01 Unique Item Identification

MB01														
Value to be stored	**Electronic representation of data in tag memory (Binary)**													**Bits**
25SUN 014841806 PART NUMBER 0000001 A2B3C4	**2**	**5**	**S**	**U**	**N**	**0**	**1**	**4**	**8**	**4**	**1**	**8**	**0**	**224**
	110010	110101	010011	010101	001110	110000	110001	110100	111000	110100	110001	111000	110000	
	6	**P**	**A**	**R**	**T**	**N**	**U**	**M**	**B**	**E**	**R**	**0**	**0**	
	110110	010000	000001	010010	010100	001110	010101	001101	000010	000101	010010	110000	110000	
	0	**0**	**0**	**0**	**1**	**A**	**2**	**B**	**3**	**C**	**4**	Bit pad		
	110000	110000	110000	110000	110001	000001	110010	000010	110011	000011	110100	10		

8.2.1 ISO-96 Bit UII – Future Item Identification

Work is under way in ISO TC 122 and ISO/IEC JTC 1/SC 31 to create a binary data structure similar to and incorporating EPC. There are 256 possible headers with 8 bits. Reserving the value of 0x00 for "Un-programmed tag" and 0xFF for "Header values greater than 8 bits" leaves 254 possible headers. EPC has thus far assigned only 17 headers through TDS 1.6. All header values are less than 0x3F (decimal 63). The proposed structure is to retain all EPC values through 0x3F, with ISO responsible for values 0x40 and above.

Table 17: Example of Proposed ISO 96-Bit UII Structure

Header	Filter Value	Partition	Company Prefix	Item Reference	Serial Number
8	3	3	20-40	24-4	38
1000 0101 (Binary value)	(Refer to Table xx for values)	(Refer to Table xx for values)	999,999 – 999,999,999,999 (Max. decimal range)	9,999,999 – 9 (Max. decimal range)	274,877,906,943 (Max. decimal value)

8.2.2 MB11-Based Customer-Assigned Source and Item Identification

To facilitate a customer-assigned item identification, it is required that it be programmed into User Memory (MB11). The information in this section is based on that presumption.

The encodation details of Access Method 0, Format 13 are shown in Table 18 and in Annex E.

Table 18: Example of a Customer-Assigned Item Identification Placed in MB11

Value to be stored	Electronic representation in tag memory (HEX)	Bytes	Bits
18VUN987654321<GS>P 87654321	**0D 4F 11 12 54 EE 78 DF 6D 74 CF 2C 7C 1D 3F 90 E3 7D B5 D3 3C B1**	**22**	**176**

> **NOTE:** Since the data are represented in binary, trading partners shall agree on how data would be represented in Human-Readable (HR) and / or within a 2D symbol if required.

8.3 Verification

Verification or error proofing is the process of validating that the correct item is selected and / or installed in an assembly / production operation. An item may be engineered so that it can only be installed in a specific manner. In other cases, a “picking light” system might be used to identify the item to be used.

A Part Number provides the ability to identify an item at its lowest level – in a non-globally unique fashion.

See Table 16 for an ISO-based MB01-centric data construct.

8.4 Item Traceability Data Placed into MB01 (25S or SGTIN-96)

The ability to track (trace) a vehicle’s “as-built” history (in conjunction with the VIN) through the product life cycle (cradle to grave) processes requires a unique differentiation of items.

The supplier shall use data identifier “25S” or “SGTIN-96” to assign globally unique item identification, with trading partner agreement. See Table 23.

The supplier’s item data is collected during the individual manufacturing / production / assembly processes and stored long-term. When used with the customer’s database, and as outlined in this section, the data allows, in combination with any campaigns, an exact determination of those vehicles in which a specific item or items were installed. The customer and supplier shall ensure that the data record of an item is maintained over its life cycle and remains globally unique. All detailed information about this item from the supplier (e.g., batch of raw materials used, manufacturers of the items supplied, test results, default settings, manufacturing site, manufacturing equipment, etc.) are to be documented and archived so that, when required, data are available on the functional, manufacturing, and material quality of the item(s) quickly, accurately, and unambiguously.

8.5 Item Characteristic(s): 25S or SGTIN (MB01) and User Memory (MB11)

Having globally unique item identification (25S or SGTIN) coupled with key specific item characteristics enable business processes. This capability supports warranty validation, quality control, product development, calibration parameters, service, etc. RFID tags are read/write capable, i.e., the data can be both static (locked but readable) and dynamic (readable and writable). “New capabilities = new possibilities.”

Examples of product characteristics are test stand results for radiator pressure test; current draw for a lighting module; measured torque of a fastener; Radio Frequency (RF) of a device or any other significant measurement data that are determined to be significant for quality control or warranty. In some cases, product characteristics are needed for use by processes such as piston size match to cylinder bore.

A major opportunity for leveraging product characteristics is when new technologies are being introduced, for example, electric vehicle batteries. “We don’t know what we don’t know,” because there is no history yet. Consider an individual Lithium-ion (Li-ion) cell as an example. Having product characteristics such as open circuit voltage, internal cell resistance, ambient temperature of the test stand, etc., might be key parameters that engineering would like to know. For instance, a cell's open circuit voltage is measured at the customer's first quality control station and compared with the value recorded on the tag by the supplier's final test audit. This triggers a go/no-go event and supports documentation for the return process. Critical data are available at site of use.

8.5.1 Unique Serial Number with Product Characteristic

A supplier-assigned globally unique item datum shall use the "25S" data identifier in MB01 and product data using "7Q" data identifier with data separator "<GS>" placed into MB11.

When used, data Identifier "7Q" SHALL be used with the appropriate appended unit of measure qualifier ANSI X12.3 Data Element Number 355 Unit of Measure. The data, if appropriate, may contain a decimal point for the required precision. Example of a voltage measurement of 14.7 Volts Direct Current (VDC) would be "7Q14.72H", where "2H" is the qualifier for "VDC".

For encodation details, see Annexes D and E.

Table 19: Example of Product Characteristics Data Stored in MB11

MB11			
Value to be stored	Electronic representation in tag memory (HEX)	Bytes	Bits
7Q14.72H<GS>7Q21.4CE	**0D 4F 06 0E C7 4B B7 C8 8F 07 4F ED D1 CB 1B B4 0C 58**	**18**	**144**

8.6 MB01-centric Vehicle Identification Number (VIN) DI = I

A vehicle identification number (VIN) is the automotive equivalent of human DNA. It sets one vehicle apart from millions of other vehicles (passenger cars, multipurpose passenger vehicles, trucks, buses, trailers (including trailer kits), incomplete vehicles, and motorcycles). It details a vehicle's uniqueness and heritage and provides a form of "factory (birth) to scrap yard (death)" identification. It can be used to track recalls, registrations, warranty claims, thefts, and insurance coverage. VINs also have many other important uses. For example, service shops use VINs to identify the engine, transmission, and brake systems installed by manufacturers so that they can properly service the vehicles. Law enforcement agencies use VINs to identify and recover stolen cars and car parts. Auto manufacturers use VINs when they resolve safety recalls.

The VIN (sometimes incorrectly referred to as "VIN number") is a unique serial number used by the automotive industry to identify individual motor vehicles (also applies to towed vehicles, motorcycles, and mopeds as defined in ISO 3833). Prior to 1981, there was no accepted standard for these numbers, so different manufacturers used different formats. Since 1981, the VIN consists of 17 characters, which do not include the letters I (i), O (o), or Q (q) (to avoid confusion with numerals 1 and 0).

Modern-day Vehicle Identification Number systems are based on two related standards, originally issued in 1979 and 1980; ISO 3779 and ISO 3780, respectively. Compatible but somewhat different implementations of these ISO standards have been adopted by the European Union and the United States. The VIN is composed of only the DI "I" and the 17-character vehicle identification number. See Table 20 and Table 21.

Table 20: Vehicle Identification Number (VIN) Definition

Position	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17
Function	World Manufacturer Identifier (WMI)			Vehicle Descriptor Section (VDS)						Vehicle Identifier Section (VIS)							

World Manufacturer Identifier (WMI): The first three characters uniquely identify the manufacturer of the vehicle using the World Manufacturer Identifier or WMI code (see Annex L for list of country codes). A manufacturer who builds fewer than 500 vehicles per year uses a 9 as the third digit, and the 12th, 13th and 14th position of the VIN for a second part of the identification. Some manufacturers use the third character as a code for a vehicle category (e.g., bus or truck), a division within a manufacturer, or both. The Society of Automotive Engineers (SAE) assigns WMIs to countries and manufacturers. The first character of the WMI is the region in which the manufacturer is located. In practice, each is assigned to a country of manufacture, although in Europe the country where the continental HQ is located can assign the WMI to all vehicles produced in that region (example: GM Europe cars--whether produced in Germany, Spain, UK, Belgium, or Poland--carry the W0 WMI because GM Europe is based in Germany).

Vehicle Descriptor Section (VDS): The 4th to 9th positions in the VIN are the Vehicle Descriptor Section or VDS. This is used, according to local regulations, to identify the vehicle type and may include information on the automobile platform used, the model, and the body style.

> **NOTE:** The 9th position is used by the United States as a check digit as detailed under 49 CFR, §§ 565.

Vehicle Identifier Section (VIS): The 10th to 17th positions are used as the Vehicle Identifier Section or VIS. This is used by the manufacturer to identify the individual vehicle in question. This may include information on options installed or engine and transmission choices, but often is a simple sequential number.

Model Year: The 10th digit is required worldwide to encode the model year of the vehicle. Besides the three letters that are not allowed in the VIN itself (I, O, and Q), the letters U and Z and the digit 0 are not used for the Model Year Code.

Plant Code: The 11th character encodes the factory of manufacture of the vehicle. Although each manufacturer has its own set of plant codes, its location in the VIN is standardized.

8.6.1 MB01 Encodation Example: VIN

The vehicle manufacturer shall assign VIN data and shall use the "I" data identifier.

The data in Table 21 are encoded according to ISO/IEC 17367. For encodation details, see Annex C.

> **NOTE:** Along with encoding the VIN into the UII Tag Encoding segment of MB01, an AFI of "0xA1" would be encoded into the Attribute / AFI segment of MB01. See Figure 3.

Table 21: Example of 6-Bit Encoding of the VIN Data into the UII of MB01

Value to be stored	Electronic representation of data in tag memory													Bits
I1G3 NL52 T71 C000000	I	1	G	3	N	L	5	2	T	7	1	C	0	112
	001001	110001	000111	110011	001110	001100	110101	110010	010100	110111	110001	000011	110000	
	0	0	0	0	0	Bit Pad								
	110000	110000	110000	110000	110000	1000								

NOTE: Table 21 does not show the AFI encoded; see Annex C for a full encodation of MB01 when PC Bit 17 = 1.

NOTE: This record shall be locked via the Permalock command. This is the ability of the tag protocol to cause a memory location to be "read-only." Once "locked," that memory location cannot be unlocked.

8.7 Anti-counterfeiting (TID and 25S or SGTIN (MB01))

Increasingly more attention is being paid to the requirement for anti-counterfeiting measures to protect product and brand authenticity.

Attention shall be paid to ISO TC 247 for developments in this arena.

As a best practice, this standard shall leverage TID, a globally unique identification of the individual silicon chip. The TID shall be serialized and locked by the silicon manufacturer. Since the TID is "burned into" the silicon and cannot be altered, and the UII is Permalocked, there is a significant level of data security. And security is further enhanced as the TID and the UII are created and controlled by two separate entities.

To use the benefits of RFID technology, automatic identification should ideally be combined with protection against copies. Basically, two different approaches, which can be also combined, are possible:

- System-side security
- Object-oriented security

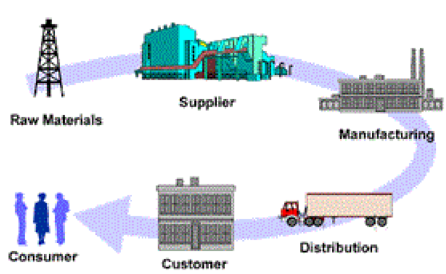


Figure 10: System-side Security - Supply Chain Transparency

The system-side approach to anti-counterfeiting protection is achieved by increased transparency in the supply chain. In this case, the reading events generated for the respective marked component are collected by production and logistical processes. In their totality, they combine into an electronic product life file. Using plausibility tests, discrepancies can be discovered and potential counterfeits can be identified more easily and the source can be contained more tightly.

A high hurdle is created for counterfeiters to circulate plausible copies. Because the individual reading events are dispersed at the enterprises involved and not made available as a whole, a complete, plausible product history can hardly ever be faked--or only with massive effort. Indeed, this approach assumes a high degree of system integration. Applications that should assess the likelihood of a forgery need standardized access to the reading events of the parts throughout the whole supply chain.

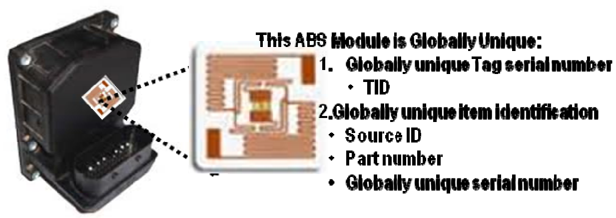


Figure 11: Object-oriented Security

Object-oriented protection against copies is aimed at protecting the identification of a product and possible additional user data directly on the RFID tag by encoding. Then only authorized parties can decipher these data.

Experts speak of symmetrical and asymmetrical encryption procedures. Highly simplified, it can be said that symmetrical procedures are easier to implement at the chip level; however, they require an IT infrastructure and are more susceptible to external attacks. Asymmetrical procedures can check the original identity on site; however, they require more chip surface (more processor power, more memory, and therefore are more expensive) and will not be realized in the near future on a UHF basis. If encoding procedures are used in parts marking, from the point of view of the users, the asymmetrical (On-Site Identification) is preferable to the symmetrical (Network Identification).

The question of which approach to copy protection is more logical has no simple answer. Taking into account costs and availability, interim steps are also possible.

8.8 Data Retention Requirements

Should the need arise (e.g., safety or quality event), the supplier shall have (with mutual trading partner agreement) the capability to obtain sufficient data from their system(s) to provide applicable data (plant site, materials, processes, test data, etc.) and these data should be retained in compliance with the respective country's requirements.

ANNEX A: SMART LABEL; INFORMATIVE

A barcode label that contains an RFID tag is termed a “smart label.” Since the label is able to store information (like a serial number) and communicate with an RFID reader, it is considered "smart."

Labels may be attached to individual parts, or groups of parts, e.g., affixed to a shipping container or pallet.

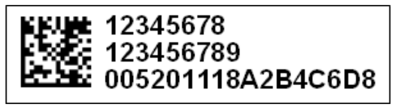


Figure 12: Example of a Smart Label that Could Be Affixed to an Item Where Both Human-Readable and/or 2D Barcode Data Might Be Required

Figure 13: Example of Smart Label that Could Be Affixed to a Shipping Container Complying with the JAIF B-16 *Global Transport Label Standard for the Automotive Industry*

Thermal direct and thermal transfer printers are available that not only print the human-readable and 1D/2D symbologies but also are capable of programming the embedded RFID tag.

Figure 14: Example of a Smart Label Printer

ANNEX B: ASCII-CHARACTER TO 6-BIT-ENCODING

Reference ISO 17367.

Table 22: ASCII-Character-to-6-Bit Encoding Table

Character	Binary Value	Character	Binary Value	Character	Binary Value	Character	Binary Value
Space	100000	0	110000	@	000000	P	010000
<EOT>	100001	1	110001	A	000001	Q	010001
<Reserved>	100010	2	110010	B	000010	R	010010
<FS>	100011	3	110011	C	000011	S	010011
<US>	100100	4	110100	D	000100	T	010100
<Reserved>	100101	5	110101	E	000101	U	010101
<Reserved>	100110	6	110110	F	000110	V	010110
<Reserved>	100111	7	110111	G	000111	W	010111
(	101000	8	111000	H	001000	X	011000
)	101001	9	111001	I	001001	Y	011001
*	101010	:	111010	J	001010	Z	011010
+	101011	;	111011	K	001011	[	011011
,	101100	<	111100	L	001100	\	011100
-	101101	=	111101	M	001101	]	011101
.	101110	>	111110	N	001110	<GS>	011110
/	101111	?	111111	O	001111	<RS>	011111

Note: As used in this document, only the following data characters are allowed: 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, A, B, C, D, E, F, G, H, I, J, K, L, M, N, O, P, Q, R, S, T, U, V, W, X, Y, Z, *(asterisk), + (plus sign), -(dash), . (period or full stop), <EOT>, <RS>, and <GS> from Table 22.

ANNEX C: MB01; ISO- AND EPC-BASED DATA FORMAT EXAMPLES; INFORMATIVE

NOTE: The **AFI** shall NOT be encoded as part of the UII when PC Bit 17 = 1. The **AFI** is programmed into the **Attribute / AFI** section of MB01, starting at memory location 0x18 through memory location 0x1F. See ISO/IEC 15961-2 for details on the AFI. See Figure 2.

ISO-Based Birth Record (UII) Data Format Example

The data were encoded according to ISO 17367.

Table 23: ISO-based Birth Record Example: When PC Bit 17 = 1

AFI	UII				
	DI	IAC	CIN	SN	
				Part Number	Part Serial Number
0xA1	25S	UN	987654321	87654321012345678	A2B4C6D8

The data in the above example would be encoded in MB01, starting at memory location 0x18, as (binary: 248 bits total; UII = 240 bits, AFI = 8 bits):

Table 24: MB01 Structure of AFI and UII (DUNS) Using 6-bit Encoding, When PC Bit 17 = 1

AFI = 0xA1			2	5	S	U	N	9	8	7	6	5	4	3	2
1010 0001			110010	110101	010011	010101	001110	111001	111000	110111	110110	110101	110100	110011	110010
1	8	7	6	5	4	3	2	1	0	1	2	3	4	5	6
110001	110111	111000	110110	110101	110100	110011	110010	110001	110000	110001	110010	110011	110100	110101	110110
7	8	A	2	B	4	C	6	D	8	Bit Pad					
110111	111000	000001	110010	000010	110100	000011	110110	000100	111000	100000					

Beyond those conditions already set forth, this document does not define how the Serial Number is to be constructed. The overall result shall be that the UII provides a globally unique identity.

SGTIN-96 Birth Record Example: When PC Bit 17 = 0;

Table 25: SGTIN-96 Birth Record Example: When PC Bit 17 = 0

Attribute	UII					
	IAC			CIN	SN	
	Header	Filter Value	Partition	Company Prefix	Item Reference	Serial Number
0x00	48	0	6	123456	012345	123456789012

The data in the above example would be encoded in MB01, starting at memory location 0x18, as (hex):
00 30 18 78 90 03 DC 9E 5C BE 99 1A 14

(13 HEX bytes, inclusive of the Attribute data. Spaces are added for clarity only.)

ANNEX D: MB11; ACCESS METHOD 0, FORMAT 3 DATA EXAMPLE; INFORMATIVE

ENCODING

The following data encoding example is provided to show how the "As-Built" example of an ISO/IEC 15434 message is to be encoded in User Memory, using Access Method 0 and Format 3, as outlined in ISO/IEC FDIS 15962:2011, Annex T.

The message to be encoded and stored in MB11 for the "As-Built" example shown in Annex J is:

[)><RS>06<GS>
P34567812<GS>12V345678912<GS>TCC09030333333333<RS>06<GS>
P23456781<GS>12V234567891<GS>TBB09018222222222<RS>06<GS>
P12345678<GS>12V123456789<GS>TAA08274111111111<RS>06<GS>
P45678123<GS>12V456789123<GS>TDD09019444444444<RS>06<GS>
P56781234<GS>12V567891234<GS>TEE09016555555555<RS><EOT>

NOTE: As described in Section 5 and Annex J, the first data element from the "As-Built" example, "**25SUN98765432100000000087654321A2B4C6D8E**", will be encoded in UII memory, not User memory.

Using Access Method 0, Format 3, the encodation of the "As-Built" example would proceed as follows:

- The system will verify that the input is a properly formatted ISO/IEC 15434 message.
- The system will remove the ISO/IEC 15434 Message Header "[)><RS>" and the Message Trailer "<RS><EOT>".
- For the example above, this will result in the following data:

  P34567812<GS>12V345678912<GS>TCC09030333333333<RS>06<GS>
  P23456781<GS>12V234567891<GS>TBB09018222222222<RS>06<GS>
  P12345678<GS>12V123456789<GS>TAA08274111111111<RS>06<GS>
  P45678123<GS>12V456789123<GS>TDD09019444444444<RS>06<GS>
  P56781234<GS>12V567891234<GS>TEE09016555555555

- The system will encode the initial Format Indicator (in this example, "06") within the Precursor.
- Using the "shorthand" option, the system will then strip off each Format Header (nn<**GS**>) that follows a Format Trailer (<RS>) in which the Format Indicator is the same as the Format Indicator stored in the Precursor.
  - In the example below, the Format Indicator "06<GS>" has been removed from the data to be programmed because it is the same as the Format Indicator stored in the Format Header.
  - Encode an <EOT> character after the last encoded data character
  - Using the "shorthand" example results in the following data:

    P34567812<GS>12V345678912<GS>TCC09030333333333<RS>
    P23456781<GS>12V234567891<GS>TBB09018222222222<RS>
    P12345678<GS>12V123456789<GS>TAA08274111111111<RS>
    P45678123<GS>12V456789123<GS>TDD09019444444444<RS>

P56781234<GS>12V567891234<GS>TEE09016555555555<EOT>

> **NOTES**: When the system sees a Format Indicator value in a data stream segment that is different from the Format Indicator encoded in the Precursor, it will retain the entire Format Trailer and Format Header (<RS>nn<GS>) with that data segment.
>
> The shorthand option saved 12 characters (72-bits, using 6-bit encoding).
>
> The encoder may replace the ASCII non-printable control characters <GS> and <RS> with the characters "^" and "_" when the data are encoded in an RFID tag. The non-printable control characters <GS>, <RS>, and <EOT> do not have a human-readable / printable representation, i.e., a symbol to represent them. Therefore, software programmers might code their programs to represent these control characters using ISO 646 viewable / printable characters to aid human readability. For example, <RS> as ▲; <GS> as ↔, and <EOT> as ♦ or simply as a space between data fields without <RS> or <EOT> being displayed. Unfortunately, there is no consistent recommendation or standard, so it is left up to the software providers to adopt their own representations.

The system now proceeds with standard ISO/IEC 15962 encoding, constructing a data set using Access Method 0, Format 3:

- The relative OID ("06", the initial Format Indicator in this example) is less than 15, and so the value "6" is directly encoded in binary as the lower 4 bits of the Precursor;
- Using Table 11, select the 6-bit binary value for each character in the message. The resulting stream of 6-bit characters is shown below.
- The stream of 6-bit characters is assembled into 8-bit bytes (the hex bytes encoded into User Memory are shown below).
- Note that:
  - The first byte of memory is the DSFID, whose value is 0x03, indicating Access Method 0 Format 3 (and is always that value, in this application).
  - The second byte of memory is the Precursor, whose value is 0x46 (and is always that value, in this application).
    - The uppermost bit is an "offset" bit (always 0b0 in this application);
      The next 3 bits indicate the compaction mode ('0b100' indicates 6-Bit Compaction);
    - And the lower 4 bits of this value are the Relative OID, indicating the leading Format Indicator of the original ISO/IEC 15434 message ('0b0110' in this case, representing Format Indicator "06" for DI messages).
  - The next 2 bytes are 0x81 and 0x1A, representing the object length (in bytes). Because this is a relatively long message, the length requires 2 bytes.
    - The first byte has its most significant bit set to 0b1 (indicating that there is another length byte), and the second byte has its most significant bit set to 0b0 (indicating that it is the last length byte): "1"bbbbbbb "0"bbbbbbb.
    - The data length is encoded in the 14 "b" bits.
      - Decimal 154 = binary 10011010 = "1"bbbbbb1 "0"0011010 = 10000001 00011010 = hex 81 1A.

For example, the first four 6-bit data values are: **010000 110011 110100 110101**

These 24 bits form the 3-byte sequence: **01000011 00111101 00110101**, (0x43 3D 35), as shown below

The system output is shown below:

DSFID: 00000011 / 0x**03**

Precursor: 01000110 / 0x**46**

Object Length: 10000001 00011001 / 0x**81** 0x**1A**

Compacted Data:

```
01000011 00111101 00110101 11011011 01111110 00110001 11001001 11101100
01110010 01011011 00111101 00110101 11011011 01111110 00111001 11000111
00100111 10010100 00001100 00111100 00111001 11000011 00111100 00110011
11001111 00111100 11110011 11001111 00111100 11110011 01111101 00001100
10110011 11010011 01011101 10110111 11100011 00010111 10110001 11001001
01101100 10110011 11010011 01011101 10110111 11100011 10011100 01011110
01010000 00100000 10110000 11100111 00001100 01111000 11001011 00101100
10110010 11001011 00101100 10110010 11001001 11110100 00110001 11001011
00111101 00110101 11011011 01111110 00011110 11000111 00100101 10110001
11001011 00111101 00110101 11011011 01111110 00111001 01111001 01000000
01000001 11000011 10001100 10110111 11010011 00011100 01110001 11000111
00011100 01110001 11000111 00010111 11010000 11010011 01011101 10110111
11100011 00011100 10110011 01111011 00011100 10010110 11010011 01011101
10110111 11100011 10011100 01110010 11001101 11100101 00000100 00010011
00001110 01110000 11000111 10011101 00110100 11010011 01001101 00110100
11010011 01001101 00011111 01000011 01011101 10110111 11100011 00011100
10110011 11010001 11101100 01110010 01011011 01011101 10110111 11100011
10011100 01110010 11001111 01000111 10010100 00010100 01011100 00111001
11000011 00011101 10110101 11010111 01011101 01110101 11010111 01011101
01110101 10000110
```

To make the last datum equal to 8 bits, the binary pad value "10" was added.

Bytes encoded into User Memory:

03 46 81 1A 43 3D 35 DB 7E 31 C9 EC 72 5B 3D 35 DB 7E 39 C7 27 94 0C 3C 39 C3 3C 33 CF 3C F3 CF 3C F3 7D 0C B3 D3 5D B7 E3 17 B1 C9 6C B3 D3 5D B7 E3 9C 5E 50 20 B0 E7 0C 78 CB 2C B2 CB 2C B2 C9 F4 31 CB 3D 35 DB 7E 1E C7 25 B1 CB 3D 35 DB 7E 39 79 40 41 C3 8C B7 D3 1C 71 C7 1C 71 C7 17 D0 D3 5D B7 E3 1C B3 7B 1C 96 D3 5D B7 E3 9C 72 CD E5 04 13 0E 70 C7 9D 34 D3 4D 34 D3 4D 1F 43 5D B7 E3 1C B3 D1 EC 72 5B 5D B7 E3 9C 72 CF 47 94 14 5C 39 C3 1D B5 D7 5D 75 D7 5D 75 86

DECODING

Using Access Method 0, Format 3, the decoding of the encoded "As Built" example would proceed as follows:

- Hexadecimal data stream read from MB11 (including Protocol overhead, DSFID; Precursor and Length of Data):

03 46 81 1A 43 3D 35 DB 7E 31 C9 EC 72 5B 3D 35 DB 7E 39 C7 27 94 0C 3C 39 C3 3C 33 CF 3C F3 CF 3C F3 7D 0C B3 D3 5D B7 E3 17 B1 C9 6C B3 D3 5D B7 E3 9C 5E 50 20 B0 E7 0C 78 CB 2C B2 CB 2C B2 C9 F4 31 CB 3D 35 DB 7E 1E C7 25 B1 CB 3D 35 DB 7E 39 79 40 41 C3 8C B7 D3 1C 71 C7 1C 71 C7 17 D0 D3 5D B7 E3 1C B3 7B 1C 96 D3 5D B7 E3 9C 72 CD E5 04 13 0E 70 C7 9D 34 D3 4D 34 D3 4D 1F 43 5D B7 E3 1C B3 D1 EC 72 5B 5D B7 E3 9C 72 CF 47 94 14 5C 39 C3 1D B5 D7 5D 75 D7 5D 75 86

> **NOTE**: The data have been placed into rows, with spaces, for clarity. The example does not represent the actual data output from the RFID tag. The actual output from the RFID tag will be a continuous hex-based data stream.

- The system will first process the Protocol overhead:

  - The first byte of the data stream (0x03) above is the DSFID.
  - The second byte of the data stream (0x46 or 0b01000110) is the Precursor.
    - The uppermost bit of the Precursor is an "offset" bit;
    - The next 3 bits indicate the compaction mode ('0b100' indicates 6-Bit Compaction);
    - The lower 4 bits of this value are the Relative OID, indicating the leading Format Indicator of the original ISO/IEC 15434 message ('0b0110' in this case, representing Format Indicator "06" for DI messages).
      - This Format Indicator value will be used to restore the Format Header (nn<GS>) of the ISO/IEC 15434 message when the "shorthand" option was used in data encodation.
- The next 2 bytes are 0x81 1A (10000001 00011010) representing the length of encoded data (in bytes). The total length of the encoded data is ~~1000000~~1 ~~0~~0011010 = 10011010 = 154 bytes.
- The system will then parse the remaining data stream, after the Protocol overhead, into 6-bit data, removing any added pad bits.

```
010000 110011 110100 110101 110110 110111 111000 110001 110010 011110 110001
110010 010110 110011 110100 110101 110110 110111 111000 111001 110001 110010
011110 010100 000011 000011 110000 111001 110000 110011 110000 110011 110011
110011 110011 110011 110011 110011 110011 110011 011111 010000 110010 110011
110100 110101 110110 110111 111000 110001 011110 110001 110010 010110 110010
110011 110100 110101 110110 110111 111000 111001 110001 011110 010100 000010
000010 110000 111001 110000 110001 111000 110010 110010 110010 110010 110010
110010 110010 110010 110010 011111 010000 110001 110010 110011 110100 110101
110110 110111 111000 011110 110001 110010 010110 110001 110010 110011 110100
110101 110110 110111 111000 111001 011110 010100 000001 000001 110000 111000
110010 110111 110100 110001 110001 110001 110001 110001 110001 110001 110001
110001 011111 010000 110100 110101 110110 110111 111000 110001 110010 110011
011110 110001 110010 010110 110100 110101 110110 110111 111000 111001 110001
110010 110011 011110 010100 000100 000100 110000 111001 110000 110001 111001
110100 110100 110100 110100 110100 110100 110100 110100 110100 011111 010000
110101 110110 110111 111000 110001 110010 110011 110100 011110 110001 110010
010110 110101 110110 110111 111000 111001 110001 110010 110011 110100 011110
010100 000101 000101 110000 111001 110000 110001 110110 110101 110101 110101
110101 110101 110101 110101 110101 110101 100001
```

- Now, using the "shorthand method," the system will convert each 6-bit datum into its ASCII character. The result of this conversion is as follows;

P34567812<GS>12V345678912<GS>TCC09030333333333<RS>
P23456781<GS>12V234567891<GS>TBB09018222222222<RS>
P12345678<GS>12V123456789<GS>TAA08274111111111<RS>
P45678123<GS>12V456789123<GS>TDD09019444444444<RS>
P56781234<GS>12V567891234<GS>TEE09016555555555<EOT>

- System deletes the encoded <EOT> from the end of the message.
- Using the "shorthand method," the system will scan through the entire ASCII message looking for the Format Trailer (<RS>).
- When the system finds the Format Trailer (<RS>), it looks at the next three characters;

  - If it is NOT “nn<GS>”, then the system will insert the Format Indicator value contained within the Precursor and the Data Element Separator (<GS>) to the data segment that follows, i.e., “06<GS>”, in this application.
  - If it DOES see an “nn<GS>”, the system will leave these three characters where they are and will look for the next Format Trailer (<RS>).
  - The system repeats this process until it hits the end of data.

- The re-constructed message, following the above process, is as follows;

P34567812<GS>12V345678912<GS>TCC09030333333333<RS>06<GS>
P23456781<GS>12V234567891<GS>TBB09018222222222<RS>06<GS>
P12345678<GS>12V123456789<GS>TAA08274111111111<RS>06<GS>
P45678123<GS>12V456789123<GS>TDD09019444444444<RS>06<GS>
P56781234<GS>12V567891234<GS>TEE09016555555555

- Now, the system will restore the ISO/IEC 15434 Message Header “[)><RS>”, and the Message Trailer “<RS><EOT>”.
  - The fully re-constructed message is as follows:

[)><RS>06<GS>
P34567812<GS>12V345678912<GS>TCC09030333333333<RS>06<GS>
P23456781<GS>12V234567891<GS>TBB09018222222222<RS>06<GS>
P12345678<GS>12V123456789<GS>TAA08274111111111<RS>06<GS>
P45678123<GS>12V456789123<GS>TDD09019444444444<RS>06<GS>
P56781234<GS>12V567891234<GS>TEE09016555555555<RS><EOT>

Note: The data have been placed into rows for clarity only.

ANNEX E: MB11; ACCESS METHOD 0, FORMAT 13 DATA EXAMPLES; INFORMATIVE

The following data encoding examples are provided to show how the ISO/IEC 15962 data syntax rules are used in MB11 (User Memory).

DSFID **0x0D** (No Directory Access Method) is used in all examples.

NOTE: Spaces have been added to binary and hex data for clarity ONLY.

1) Part Number:

P1234567890ABCDEFGH

DI: P

- Using the ISO/IEC 15962 Assigned Relative OID DI Table, the value for DI "P" = 15 / 0b1111 / 0x0F. As the Relative OID is larger than 14, it must be encoded in a separate byte; see Precursor, below.
- Compact data (excluding DI) using **6-Bit Compaction**, Precursor value = 0b100.

Data: 1234567890ABCDEFGH

- Convert each data character (disregard DI) to HEX (using Table 29 in Annex I), then to byte-based Binary:
  - 0X31 32 33 34 35 36 37 38 39 30 41 42 43 44 45 46 47 48
  - 00110001 00110010 00110011 00110100 00110101 00110110 00110111 00111000 00111001 00110000 01000001 01000010 01000011 01000100 01000101 01000110 01000111 01001000
- Strip off leading binary "00" and "01" from each byte, and concatenate remaining 6-bit segments to bit-string;
  - 110001110010110011110100110101110110110111111000111001110000000001000010000011000100000101000110000111001000
- Divide bit-string, starting with MSB, into 8-bit bytes. Pad as appropriate (binary "1000" in this example):
  - 11000111 00101100 11110100 11010111 01101101 11111000 11100111 00000000 01000010 00001100 01000001 01000110 00011100 1000**<u>1000</u>**
- Convert binary to HEX: C7 2C F4 D7 6D F8 E7 00 42 0C 41 46 1C 88

Complete Message Construction:

- Precursor: 0b01001111 / 0x**4F**
  - o Offset = 0b0
  - o Compaction Code = 0b100
  - o Relative OID value >14 = 0b1111 (Requires Relative OID be placed in additional byte)
- Relative OID: 15 – 15 = 0 / 0b0000 / 0x**00**
- Length code: 14 / 0b1110 / 0x**0E**

Message encoded (in hex): **4F 00 0E C7 2C F4 D7 6D F8 E7 00 42 0C 41 46 1C 88**

- In this example, the DSFID "0x**0D**" is encoded as the first byte within User Memory (MB11).

Complete MB11 Message: ASCII Data to DSFID, Precursor, Relative OID, Length of Data, Compacted Data - in HEX:

P1234567890ABCDEFGH ⟺ 0D 4F 00 0E C7 2C F4 D7 6D F8 E7 00 42 0C 41 46 1C 88

The complete encoding requires 18 bytes, or 144 bits.

2) Unique Item Identification

25SUN98765432187654321012345678A2B4C6D8

DI: 25S

Using the ISO/IEC 15962 Assigned Relative OID DI Table, the value for DI "25S" = 1 / 0b01 / 0x01, and encodes directly into the Precursor; see Precursor, below.

Compact contents using **6-Bit Compaction**, Precursor value = 0b100.

Data: **UN98765432187654321012345678A2B4C6D8**

Convert each data character (disregard DI) to HEX (using Table 29), then to byte-based binary:

41 1B 54 EE 78 DF 6D 74 CF 2C 78 DF 6D 74 CF 2C 70 C7 2C F4 D7 6D F8 07 20 B4 0F 61 38

01000001 00011011 01010100 11101110 01111000 11011111 01101101 01110100 11001111 00101100 01111000 11011111 01101101 01110100 11001111 00101100 01110000 11000111 00101100 11110100 11010111 01101101 11111000 00000111 00100000 10110100 00001111 01100001 00111000

Strip off leading "0b00" and "0b01" from each byte, and concatenate remaining 6-bit segments to bit-string:

000001011011010100101110111000011111101101110100001111101100111000011111101101110100001111101100110000000111101100110100010111101101111000000111100000110100001111100001111000

Divide the bit-string, starting with the MSB, into 8-bit bytes. Pad as appropriate (a pad of binary "10" needed in this example):

00000101 10110101 00101110 11100001 11111011 01110100 00111110 11001110 00011111 10110111 01000011 11101100 11000000 01111011 00110100 01011110 11011110 00000111 10000011 01000011 11100001 111000<u>**10**</u>

Convert binary to HEX: **05 B5 2E E1 FB 74 3E CE 1F B7 43 EC C0 7B 34 5E DE 07 83 43 E1 E2**

Complete Message Construction:

Precursor = 0b01000001 / 0x**41**

Offset = 0b0

6-Bit Compaction Code = 0x100

Relative OID; 01 / 0b0001

Length of data = 22 / 0b10110 / 0x**16**

Compacted data in HEX = **05 B5 2E E1 FB 74 3E CE 1F B7 43 EC C0 7B 34 5E DE 07 83 43 E1 E2**

In this example, the DSFID "0x**0D**" is encoded as the first byte within User Memory (MB11).

Complete Message: ASCII Data to DSFID, Precursor, Relative OID, Length of Data, Compacted Data - in HEX:

25SUN123456789A2B4C6D8E ⇔ 0D 41 16 05 B5 2E E1 FB 74 3E CE 1F B7 43 EC C0 7B 34 5E DE 07 83 43 E1 E2

The complete encoding requires 30 bytes, or 240 bits.

3) Vehicle Identification Number:

I1G3NL52T71C000000

DI = I

Using the ISO/IEC 15962 DI Table, the value for DI "I" = 96 / 0b1100000 / 0x60. As the Relative OID is larger than 14, it must be encoded in a separate byte; see Precursor, below.

Data: 1G3NL52T71C000000

- Compact contents using **6-Bit Compaction**, Precursor value = 0b100.
- Convert each data character (disregard DI) to HEX (using Table 29), then to byte-based binary:
  - 31 47 33 4E 4C 35 32 54 37 31 43 30 30 30 30 30 30
  - 00110001 01000111 00110011 01001110 01001100 00110101 00110010 01010100 00110111 00110001 01000011 00110000 00110000 00110000 00110000 00110000 00110000
- Strip off leading binary "00" and "01" from each binary byte and concatenate remaining data into bit-string:
  - 110001000111110011001110001100110101110010010100110111100010000 111100000110000011000001100000110000110000
- Divide bit-string, starting with MSB, into 8-bit bytes. Pad as appropriate ("0b10" in this example):
  - 11000100 01111100 11001110 00110011 01011100 10010100 11011111 00010000 11110000 11000011 00001100 00110000 11000010
- Convert binary to HEX: C4 7C CE 33 5C 94 DF 10 F0 C3 0C 30 C2

Complete Message Construction:

- Precursor: (binary/hex) 01001111/**4F**
  - Offset (in binary) = 0
  - 6-Bit Compaction Code (in binary) = 100
- Relative OID value >14 = 0b1111 (Requires Relative OID be placed in additional byte)
  - Relative OID: 96 – 15 = 81/0b1010001/0x**51**
- Length code: 13 /0b1101/0x**0D**
- Message encoded in HEX: **4F 51 0D C4 7C CE 33 5C 94 DF 10 F0 C3 0C 30 C2**

In this example, the DSFID "0x0D" is encoded as the first byte within User Memory (MB11).

Complete Message: ASCII Data to DSFID, Precursor, Relative OID, Length of Data, Compacted Data - in HEX:

I1G3NL52T71C000000 ⇔ 0D 4F 51 0D C4 7C CE 33 5C 94 DF 10 F0 C3 0C 30 C2

The complete encoding requires 17 bytes, or 136 bits.

> **NOTE:** This record shall be locked via Permalock command. This is the ability of the tag protocol to cause a memory location to be "read-only." Once "locked," that memory location cannot be unlocked**.**

4) Mutually Defined;

ZBETWEENTRADINGPARTNERS12345

DI: Z

> **NOTE: This DI shall be used only with trading partner agreement and definition.**

Using the ISO/IEC 15962 DI Table, the value for DI "Z" = 103 / 0b1100111 / 0x67. As the Relative OID is larger than 14, it shall be encoded in a separate byte; see Precursor, below.

Compact contents using **6-Bit Compaction**, Precursor value = 0b100.

Data: BETWEENTRADINGPARTNERS12345

Convert each data character (disregard DI) to HEX (using Table 32 in Annex I), then to byte-based Binary:

42 45 54 57 45 45 4E 54 52 41 44 49 4E 47 50 41 52 54 4E 45 52 53 31 32 33 34 35

01000010 01000101 01010100 01010111 01000101 01000101 01001110 01010100 01010010 01000001 01000100 01001001 01001110 01000111 01010000 01000001 01010010 01010100 01001110 01000101 01010010 01010011 00110001 00110010 00110011 00110100 00110101

Strip off leading binary "00" or "01" from each byte and concatenate remaining 6-bit segments to bit-string:

000010000101010100010111000101000101001110010100010010000001000100001001001110000111010000000001010010010100001110000101010010010011110001110010110011110100110101

Divide bit-string, starting with MSB, into 8-bit bytes. Pad as appropriate (binary "100000" in this example):

00001000 01010101 00010111 00010100 01010011 10010100 01001000 00010001 00001001 00111000 01110100 00000001 01001001 01000011 10000101 01001001 00111100 01110010 11001111 01001101 01**100000**

Convert binary to HEX: 08 55 17 14 53 94 48 11 09 38 74 01 49 43 85 49 3C 72 CF 4D 60

Complete Message Construction:

- Precursor (binary/hex) = 0b01001111 / 0x**4F**
  - Offset (binary) = 0b0
  - 6-Bit Compaction Code (binary) = 0b100
  - Relative OID value >14 = 0b1111 (Requires Relative OID be placed in additional byte)
- Relative OID; 103 – 15 = 88 / 0b100001 / 0x**41**
- Length of data = 21 / 0b10101 / 0x**15**

- Compacted data in HEX = **08 55 17 15 53 94 48 11 09 38 74 01 49 43 85 49 3C 72 CF 4D 60**
- In this example, the DSFID "0x**0D**" is encoded as the first byte within User Memory (MB11).

Complete Message: ASCII Data to DSFID, Precursor, Relative OID, Length of Data, Compacted Data - in HEX:

ZBETWEENTRADINGPARTNERS12345 ⇔ 0D 4F 41 15 08 55 17 14 53 94 48 11 09 38 74 01 49 43 85 49 3C 72 CF 4D 60

The complete encoding requires 25 bytes, or 200 bits.

5) Tire ID;

21SMKB5A8WR2405

DI: 21S

Using the ISO/IEC 15962 DI Table, 21S = 08 / 0b1000 / 0x08 and encodes directly into the Precursor; see **Encoding the Precursor field**, below.

Encoding the Tire ID user data length code

Length code (hex): 09
Tire ID message length = 9 bytes.
(Length between 0 and 127 bytes is encoded with 1 byte and a lead binary bit of 0.)

Binary: 00001001

HEX: 09 48 **09** 34 B0 B5 07 85 D2 CB 4C 35

Encoding the Tire ID user data

Data: MKB5A8WR2405
Compacts as (hex): 34 B0 B5 07 85 D2 CB 4C 35
Compaction: 6-bit

M	**K**	**B**	- (balance of data =) 5A8WR2405
0x4D	0x4B	0x42	- HEX conversion
01001101	01001011	01000010	- Binary conversion of HEX data
001101	001011	000010	- Drop first two bits in each byte
00110100	10110000		- Concatenate string; group into bytes; convert to HEX
0x**34**	0x**B0**	(etc.…)	48 09 **34 B0 B5 07 85 D2 CB 4C 35**

(Block shows balance of user data converted.)

Encoding the Precursor field

Precursor (hex): 48
Encodes the compaction scheme and the Relative-OID
Bit 8 = zero for no offset
0b01001000
0x48 **48** 09 34 B0 B5 07 85 D2 CB 4C 35

Full encoding of Tire ID code and what the data would look like after using a "Read All" type of tag command. Example assumes that Tire ID is the only data encoded in MB11:

21SMKB5A8WR2405 (ASCII)⇔**0D 48 09 34 B0 B5 07 85 D2 CB 4C 35** (HEX)

The complete encoding requires 12 bytes, or 96 bits.

ANNEX F: SGTIN-96 DATA FORMAT; INFORMATIVE

Table 26: SGTIN-96 Data Format

Header	Filter Value	Partition	Company Prefix	Item Reference	Serial
8 bits	3 bits	3 bits	20-40 bits	24-4 bits	38 bits
0011 0000 (actual value)	0-7** (decimal capacity)	0-7*** (decimal capacity)	999,999 - 999,999,999,999 (Max decimal range*)	9,999,999 - 9 (Max decimal range*)	274,877,906,943* (decimal capacity)

* Maximum decimal value range of Company Prefix and Item Reference fields varies according to the contents of the Partition field.

** Refer to EPC Tag Data Standard (latest published version).

*** Refer to EPC Tag Data Standard (latest published version).

The SGTIN consists of the following information elements:

- The **Company Prefix:** assigned by GS1 to a managing entity.
- The **Item Reference:** assigned by the managing entity to a particular object class. The Item Reference for the purposes of UII encoding is derived from the GTIN by concatenating the Indicator Digit of the GTIN and the Item Reference digits and treating the result as a single integer.

> **NOTE:** The Indicator Digit (binary 0b0 or 0b1), when set to 0, indicates that there is NOT a leading zero digit in the Item Reference number. When the Indicator Digit is set to 1, this indicates that there IS a leading zero digit in the Item Reference number.

- The **Serial Number:** assigned by the managing entity to an individual object.

> **NOTE:** The data in this Annex are provided for informational purposes only. To obtain the complete construction details of the SGTIN-96, refer to GS1's Tag Data Standard, latest revision.

ANNEX G: MINIMUM RFID SYSTEM PERFORMANCE; INFORMATIVE

RFID system performance shall be as specified within the ISO/IEC 18000-63 / GS1 UHF Gen 2 air-interface standards. The following information was provided by ODIN Technologies, which has sole responsibility for its accuracy.

ISO/IEC 18000-63 / GS1 UHF Gen 2 Write Process:

The Gen 2 command sequence for a write operation takes place in two steps:

- **Step 1**: Inventory – Time estimation – 5 milliseconds (10^{-3} seconds or ms)

  The total time for inventory depends on the reader platform, link frequencies, query command, and Q value. The Inventory process takes place in the following sequence:

  1. Reader sends a Query
  2. Tag responds back with an RN16
  3. Reader acknowledges with an ACK
  4. Tag responds back with MB01; CRC, PC Word, UII.

- **Step 2**: Write (16-bit) – Time estimation - approximately 15 ms

  The write process takes place only after a successful Inventory cycle. The entire inventory and write cycle is repeated until all the bits are fully written. The time taken to complete the write cycle largely depends on the reader platform. The write process takes place in the following sequence:

  1. Reader request for the RN # (ReqRN)
  2. Tag responds back with a RN16
  3. Reader sends a write command, which contains the address where to write the UII, user memory, or reserved memory data
  4. Tag writes 16 bits
  5. Tag transmits RN16
  6. Reader acknowledges the RN16
  7. The above steps are repeated for each 16-bit data string

Theoretical calculations:

The calculated write time for encoding 96 bits UII:

5 ms (inventory) + ([96/16] * 15 ms) (write) = 95 ms

Actual measurements: (Information provided by ODIN Technologies)
Measured time to encode Alien Higgs 3, Impinj Monza 3, and NXP G2XX silicon (96 bits)
Reader used for the test = Sirit IN510
Power level = 1 watt at the connector
Antenna gain used for the test = 6 dBi

Table 27: Actual Read/Write Measurements

Silicon	Approximate time (ms)	Approximate Encode Speed (feet/min)
Alien Higgs 3	95	335
Impinj Monza 3	51	620
NXP G2XX	51	620

Note: Factors such as business process, barcode printing, graphics printing, etc. can impact encode speeds.

ANNEX H: ANS MH10.8.2 DATA IDENTIFIERS USED IN THIS DOCUMENT; INFORMATIVE

Table 28: ANS MH10.8.2 Data Identifiers Used in this Document.

DI	Title	Data Format	Total Field Length	Description
I	VEHICLE IDENTIFICATION NUMBER (VIN)	an...17	18	Exclusive Assignment - Vehicle Identification Number (VIN) as defined in the U.S. under 49 CFR, §§ 565 and internationally by ISO 3779. (These are completely compatible data structures.)
P	CUSTOMER PART NUMBER	an	-	Item Identification Code assigned by the Customer.
21S	TIRE ID	an	16	Tire Identification Number as defined by the U.S. Department of Transportation (D.O.T) under U.S. Code 49 CFR 574.5.
25S	UNIQUE ITEM IDENTIFIER	an	-	Identification of a party to a transaction assigned by a holder of a Company Identification Number (CIN) and including the related Issuing Agency Code (IAC) in accordance with ISO/IEC 15459 and its registry, structured as a sequence of 3 concatenated data elements: IAC, followed by CIN, followed by the supplier assigned serial number that is unique within the CIN holder's domain (See Annex C.11) Annex C.11 Unique Identification of Items. The intended use of Data Identifier (DI) 25S is to indicate that the data following the DI represent a concatenated data string that uniquely identifies an item. The 25S data string is formed from two segments, which are an 18V segment and a supplier-assigned serial number segment. The serial number assigned by the supplier (designated by the 18V segment) must be unique for that supplier. The 18V segment is as defined in Section 1. The serial number segment consists of a unique serial number for the Company Identification Number (CIN) in 18V. For

DI	Title	Data Format	Total Field Length	Description
				companies that serialize within part number, and/or lot/batch, methods for creating unique item identification within the serial number segments are: • part number and serial number (unique for that part number for the CIN) • lot/batch number part serial number (unique within the lot/batch for the CIN) Data strings following 18V should not be parsed to obtain the component data elements.
7Q	PRODUCT CHARACTERISTIC VALUE	an	-	Quantity, Amount, or Number of Pieces in the format: Quantity followed by the two-character ANSI X12.3 Data Element Number 355 Unit of Measurement Code
T	CUSTOMER ASSIGNED TRACEABILITY NUMBER	an	-	Traceability Number assigned by the Customer to identify/trace a unique group of entities (e.g., lot, batch, heat)
12V	DUNS Number	an	9	DUNS number identifying manufacturer
18V	IDENTIFICATION OF A PARTY TO A TRANSACTION	an	-	Identification of a party to a transaction in which the data format consists of two concatenated segments. The first segment is the Issuing Agency Code (IAC) in accordance with ISO/IEC 15459. The second segment is a unique entity identification Company Identification Number (CIN) assigned in accordance with rules established by the issuing agency (see http://www.nen.nl/web/Normen-ontwikkelen/ISOIEC-15459-Issuing-Agency-Codes.htm)
Z	MUTUALLY DEFINED	an	-	Mutually Defined between Customer and Supplier.

NOTE: All field lengths are in Bytes. ANS MH10.8.2 Data Identifiers are maintained as continuous maintenance document at http://autoid.org/ANSI_MH10/ansi_mh10sc8_wg2.htm

ANNEX I: ISO 646 CHARACTER REPRESENTATION; NORMATIVE

Table 29: ISO 646 Character Set

HEX	DEC	ASCII /ISO 646 Character		HEX	DEC	ASCII /ISO 646 Character		HEX	DEC	ASCII /ISO 646 Character
00	00	NUL		2B	43	+		56	86	V
01	01	SOH		2C	44	,		57	87	W
02	02	STX		2D	45	-		58	88	X
03	03	ETX		2E	46	.		59	89	Y
04	04	EOT		2F	47	/		5A	90	Z
05	05	ENQ		30	48	0		5B	91	[
06	06	ACK		31	49	1		5C	92	\
07	07	BEL		32	50	2		5D	93	]
08	08	BS		33	51	3		5E	94	^
09	09	HT		34	52	4		5F	95	_
0A	10	LF		35	53	5		60	96	'
0B	11	VT		36	54	6		61	97	a
0C	12	FF		37	55	7		62	98	b
0D	13	CR		38	56	8		63	99	c
0E	14	SO		39	57	9		64	100	d
0F	15	SI		3A	58	:		65	101	e
10	16	DLE		3B	59	;		66	102	f
11	17	DC1		3C	60	<		67	103	g
12	18	DC2		3D	61	=		68	104	h
13	19	DC3		3E	62	>		69	105	i
14	20	DC4		3F	63	?		6A	106	j
15	21	NAK		40	64	@		6B	107	k
16	22	SYN		41	65	A		6C	108	l
17	23	ETB		42	66	B		6D	109	m
18	24	CAN		43	67	C		6E	110	n
19	25	EM		44	68	D		6F	111	o
1A	26	SUB		45	69	E		70	112	p
1B	27	ESC		46	70	F		71	113	q
1C	28	FS		47	71	G		72	114	r
1D	29	GS		48	72	H		73	115	s
1E	30	RS		49	73	I		74	116	t
1F	31	US		4A	74	J		75	117	u
20	32	SP		4B	75	K		76	118	v
21	33	!		4C	76	L		77	119	w
22	34	"		4D	77	M		78	120	x
23	35	#		4E	78	N		79	121	y
24	36	$		4F	79	O		7A	122	z
25	37	%		50	80	P		7B	123	{
26	38	&		51	81	Q		7C	124	|
27	39	'		52	82	R		7D	125	}
28	40	(		53	83	S		7E	126	~
29	41	)		54	84	T		7F	127	DEL
2A	42	*		55	85	U				

ANNEX J: RFID AND 2D SYMBOLOGIES; INFORMATIVE

While this version of this standard is principally centered on RFID, it is the intent of this standard to encourage interoperability between RFID tags and labels, especially via 2D-based data. Much benefit is to be gained by using both technologies seamlessly within an enterprise.

The AIAG standards B-4 *Parts Identification and Tracking Application* and B-11 *Item Level RFID* are recommended for your review. The JAIF B-16 *Global Transport Label Standard for the Automotive Industry* is recommended for your review as a reference for logistics.

> **NOTE:** The non-printable control characters <GS>, <RS> and <EOT> do not have a human-readable / printable representation, i.e., a symbol to represent them. Therefore, software programmers might code their programs to represent these control characters using ISO 646 viewable / printable characters to aid human readability. For example, <RS> as ▲; <GS> as ↔; and <EOT> as ♦ or simply as a space between data fields without <RS> or <EOT> being displayed. Unfortunately, there is no consistent recommendation or standard so it is left up to the software providers to adopt their own representations.

Optical-Media-to-RFID-6-Bit-Encoding Example

The creation of the unique item identity, UII or birth record, starts at the creation of the part. What is not shown in the following “As Built” example is that each sub-assembly’s UII has its own UII. It has already been created by the manufacturer and placed onto the sub-assembly using a 2D symbology-based label.

The following data example shows how data can be read from a 2D Data Matrix symbol and then seamlessly encoded within an RFID tag. It also shows where specific data are placed within the RFID tag, and how they are encoded.

Figure 15 illustrates a gas tank and the parts that are assembled into it, along with the label that is used to identify each part.

Figure 16 shows a 2D Data Matrix symbol with all five (5) pieces of data shown, plus a traceability code for the entire assembled gas tank encoded into it.

> **NOTE**: In the following examples, the DIs iare not included in the Human-Readable Information - by design. The DIs are encoded within the Data Matrix symbol. See Table 30.

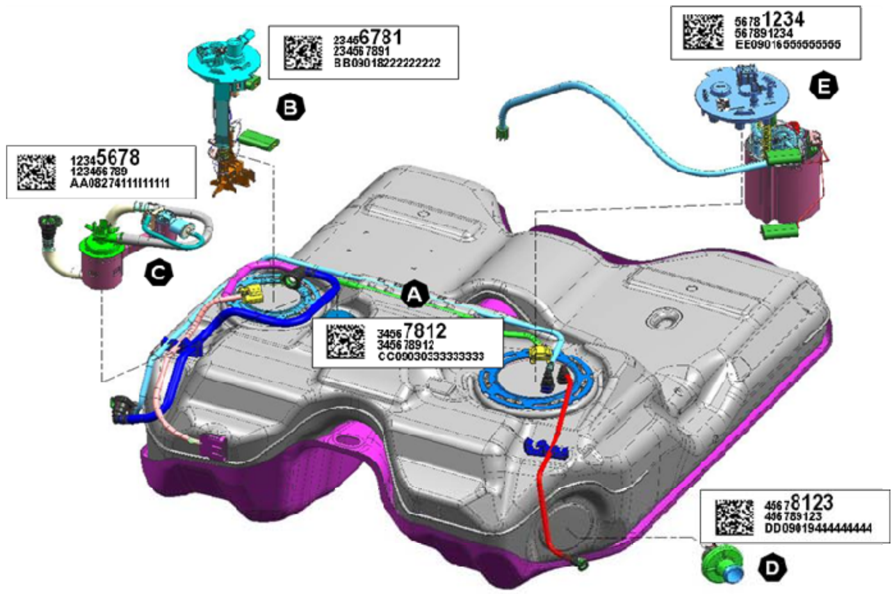


Figure 15: Gas Tank and Associated Items (parts)

0000000008765**4321**
987654321
A2B4C6D8E

34567812
345678912
CC09030333333333

23456781
234567891
BB09018222222222

12345678
123456789
AA08274111111111

45678123
456789123
DD09019444444444

56781234
567891234
EE09016555555555

Figure 16: 2D Symbol with "As Built" Data Encoded Within It

NOTE: The Human-Readable Information (HRI) shown in Figure 16 does not show the DIs used for each of the data elements. The complete construction of the data, including the DIs, is shown in Figure 17 below. (The data are shown in rows for clarity only).

~6

25SUN987654321000000000087654321A2B4C6D8E<RS>

P34567812<GS>12V345678912<GS>TCC09030333333333<RS>

P23456781<GS>12V234567891<GS>TBB09018222222222<RS>

P12345678<GS>12V123456789<GS>TAA08274111111111<RS>

P45678123<GS>12V456789123<GS>TDD0901444444444<RS>

P56781234<GS>12V567891234<GS>TEE09016555555555

Figure 17: The Data; As Encoded Within Data Matrix Symbol using Macro 06 (~6)

NOTE: Header and Trailer 06 Macro ("~6" represents Macro 06): The 06 Macro saves 8 alphanumeric characters in the encodation of ISO/IEC 15434 Data Syntax Standard. Data Matrix ISO/IEC 18004 provides a means of abbreviating the header and trailer into one character. This feature was created to reduce the number of symbol characters needed to encode data in a symbol using the ISO/IEC 15434 Data Syntax Standard. The 06 Macro character applies only when in the first symbol character position. The header will be transmitted as a prefix to the data stream and the trailer will be transmitted as a suffix to the data stream.

When PC Bit 17 of MB01 = "1", the "25S"-based data, which is the Traceability Code in the 2D symbol, will be used to create the Unique Item Identifier (UII) in MB01.

When PC Bit 17 of MB01 = "0", GS1 data constructs would be used for the UII, as outlined for Serialized Unique Item Identifiers under GS1's TDS v1.5 or higher.

The remaining five lines of data will be programmed into the User Memory Bank (MB11). See later in this section for details.

Figure 18 below shows the relationship between each data segment and the physical entity it describes.

Figure 18: The Data; As Encoded Within Data Matrix Symbol

A - Tank

3456 7812
3456 78912
CC09030333333333

B - Level Sensor

2345 6781
234567891
BB09018222222222

C - Pressure Sensor

1234 5678
123456789
AA08274111111111

D - Vent

4567 8123
456789123
DD09019444444444

E - Pump

5678 1234
567891234
EE09016555555555

Table 30: Subassembly Information that Makes Up the "As-Built" Data

Assembly	DI	Part Number	DI	IAC	DUNS	DI	Trace Code
Gas Tank	25S	00000000087654321	na	UN	987654321	na	A2B4C6D8E
A	P	34567812	12V	na	345678912	T	CC09030333333333
B	P	23456781	12V	na	234567891	T	BB09018222222222
C	P	12345678	12V	na	123456789	T	AA08274111111111
D	P	45678123	12V	na	456789123	T	DD09019444444444
E	P	56781234	12V	na	567891234	T	EE09016555555555

Table 31: Description of "As-Built" Data; As Programmed into MB01 and MB11

	Data Type	AFI / DI / EPC Type	Data	Character Count	Encoded # of Bits	Total # of Bits
MB01	MB01 Tag overhead	CRC16 (16 bits) + Protocol Control Word (16 bits)			32	32
MB01	**The following data would be placed in the UII partition (starting at 0x20) of Memory Block 01; when PC bit 17 = "0"** (**NOTE**: In the PC Bit 17 = 0 and PC Bit 17 = 1 examples below, the data used are NOT the same.)					
MB01	EPC See Annex C	SGTIN -96	**URI data:** urn:epc:tag:sgtin-96:0.1234567.012345.1234567890 12 Encoded data, in hex: 0x30 18 78 90 03 DC 9E 5C BE 99 1A 14	-	96	128
MB01	**The following data would be placed in the UII partition (starting at 0x20) of Memory Block 01; when PC bit 17 = "1"** (**NOTE:** When PC Bit 17 = 1, an AFI (in this document: "0xA1") shall be programmed into memory locations 0x18 to 0x1F of MB01. See Section 5.2.1)					
MB01	ISO See Annex C	25S	UN98765432100000000012345678 A2B4C6D8E	40	240	272
MB11	**The following data would be placed in Memory Block 11** (**NOTE:** MB01 PC bit 15 = 1; MB01 PC Bit 17 = 0 or 1; using Access Method 0, Format 3.)					
MB11	**Description**	**DI**	**Data**	**Data Separator / Terminator**	**Encoded # of Bits**	**Total # of Bits**
MB11	Protocol overhead (DSFID; Precursor; Length of Data)	-	"0x03" is the DSFID for "Access Method 0 (No Directory), Format 3 (ISO/IEC 15434)"; "0x46" is the Precursor for ISO/IEC 15434 Format Indicator 06 (DIs); "0x8119" is the length of the data in this example. (See **NOTE 1** following the table)		32 (**NOTE 1**)	32
MB11	Customer-Assigned Item Identification Code	**P**	34567812	<GS>	60	92

DUNS number identifying manufacturer *	**12V**	345678912	<GS>	78	170
Traceability Number assigned by the Customer	**T**	CC09030333333333	<RS>	108	278
Customer-Assigned Item Identification Code	**P**	23456781	<GS>	60	338
DUNS number identifying manufacturer	**12V**	234567891	<GS>	78	416
Traceability Number assigned by the Customer	**T**	BB09018222222222	<RS>	108	524
Customer-Assigned Item Identification Code	**P**	12345678	<GS>	60	584
DUNS number identifying manufacturer	**12V**	123456789	<GS>	78	662
Traceability Number assigned by the Customer	**T**	AA08274111111111	<RS>	108	770
Customer-Assigned Item Identification Code	**P**	45678123	<GS>	60	830
DUNS number identifying manufacturer	**12V**	456789123	<GS>	78	908
Traceability Number assigned by the Customer	**T**	DD09019444444444	<RS>	108	1016
Customer-Assigned Item Identification Code	**P**	56781234	<GS>	60	1076
DUNS number identifying manufacturer	**12V**	567891234	<GS>	78	1154
Traceability Number assigned by the Customer	**T**	EE09016555555555	<RS><EOT> (See **NOTE 2** following the table)	102	1256

* A complete list of Issuing Agency Codes can be found at
http://www.nen.nl/web/Normenontwikkelen/ISOIEC-15459-Issuing-Agency-Codes.htm

NOTE 1: The Protocol overhead consists of the following: DSFID (1 byte), Precursor (1 byte), and Length of Data (up to 2 bytes). The total Protocol overhead is 4 bytes (32 bits).

NOTE 2: The message trailer characters “<RS><EOT>” are not encoded. See Annex D for details.

For details on the Access Method 0, Format 03 example shown in Table 31 above, see Annex D.

For detailed Access Method 0, Format 13 examples, see Annex E.

ANNEX K: USE OF BATCH / LOT NUMBERS WITH SERIALIZATION; INFORMATIVE

Occasionally a company will produce a product that is serialized, not only within the part number but also within a lot or batch. Examples of batch and serialization are products like engines, where the block is manufactured and molded on a batch basis and the finished engine is valuable enough to be serialized for tracking and management. Examples of lot and serialization are automotive electronics, where the lot refers to items that are otherwise identical except for software loads affecting capability, so the end item is serialized for tracking and management.

The serial number is occasionally reinitialized to zero with each batch or lot. The batch or lot number must be combined with the serial number to achieve uniqueness. Thus the globally unique identity of the item would include the IAC, CIN, part number or lot or batch, and serial number.

ANNEX L: VIN REFERENCE TABLES; NORMATIVE

In the notation below, assume that letters precede numbers and that zero is the last number. For example, 8X-82 denotes 8X, 8Y, 8Z, 81, 82. In particular, this does not include 80.

Table 32: Table of Country Codes

AA-AH South Africa	ML-MR Thailand	VS-VW Spain	6X-60 not assigned
AJ-AN Ivory Coast	MS-M0 not assigned	VX-V2 Yugoslavia	7A-7E New Zealand
AP-A0 not assigned	NF-NK Pakistan	V3-V5 Croatia	7F-70 not assigned
BA-BE Angola	NL-NR Turkey	V6-V0 Estonia	8A-8E Argentina
BF-BK Kenya	NS-N0 not assigned	WA-W0 Germany	8F-8K Chile
BL-BR Tanzania	PA-PE Philippines	XA-XE Bulgaria	8L-8R Ecuador
BS-B0 not assigned	PF-PK Singapore	XF-XK Greece	8S-8W Peru
CA-CE Benin	PL-PR Malaysia	XL-XR Netherlands	8X-82 Venezuela
CF-CK Madagascar	PS-P0 not assigned	XS-XW USSR	83-80 not assigned
CL-CR Tunisia	RA-RE United Arab Emirates	Z6-Z0 not assigned	9A-9E Brazil
CS-C0 not assigned	RF-RK Taiwan	XX-X2 Luxembourg	9F-9K Colombia
DA-DE Egypt	RL-RR Vietnam	X3-X0 Russia	9L-9R Paraguay
DF-DK Morocco	RS-R0 not assigned	YA-YE Belgium	9S-9W Uruguay
DL-DR Zambia	SA-SM United Kingdom	YF-YK Finland	9X-92 Trinidad & Tobago
DS-D0 not assigned	SN-ST Germany	YL-YR Malta	93–99 Brazil
EA-EE Ethiopia	SU-SZ Poland	YS-YW Sweden	90 not assigned
EF-EK Mozambique	S1-S4 Latvia	YX-Y2 Norway	
EL-E0 not assigned	TA-TH Switzerland	Y3-Y5 Belarus	
FA-FE Ghana	TJ-TP Czech Republic	Y6-Y0 Ukraine	
FF-FK Nigeria	TR-TV Hungary	ZA-ZR Italy	
FL-F0 not assigned	TW-T1 Portugal	ZS-ZW not assigned	
GA-G0 not assigned	T2-T0 not assigned	ZX-Z2 Slovenia	
HA-H0 not assigned	UA-UG not assigned	Z3-Z5 Lithuania	
JA-JT Japan	UH-UM Denmark	1A-10 United States	
KA-KE Sri Lanka	UN-UT Ireland	2A-20 Canada	
KF-KK Israel	UU-UZ Romania	3A-3W Mexico	
KL-KR Korea (South)	U1-U4 not assigned	3X-37 Costa Rica	
KS-K0 not assigned	U5-U7 Slovakia	38-30 Cayman Islands	
LA-L0 China	U8-U0 not assigned	4A-40 United States	
MA-ME India	VA-VE Austria	5A-50 United States	
MF-MK Indonesia	VF-VR France	6A-6W Australia	

Table 33: Table of Model Year Codes

Code	Year	Code	Year	Code	Year	Code	Year
A	1980	L	1990	Y	2000	A	2010
B	1981	M	1991	1	2001	B	2011
C	1982	N	1992	2	2002	C	2012
D	1983	P	1993	3	2003	D	2013
E	1984	R	1994	4	2004	E	2014
F	1985	S	1995	5	2005	F	2015
G	1986	T	1996	6	2006	G	2016
H	1987	V	1997	7	2007	H	2017
J	1988	W	1998	8	2008	J	2018
K	1989	X	1999	9	2009	K	2019

ANNEX M: ANSI X12.3 DATA ELEMENT NUMBER 355 UNIT OF MEASURE CODE; INFORMATIVE

Table 34: Examples of ANSI X12.3 355 Data Element Number 355 Unit of Measure (Qualifier)

Qualifier	Definition	Qualifier	Definition
2G	Volts (AC)	68	Ampere
2H	Volts (DC)	CE	Centigrade
2N	Decibels	DN	Deci Newton-Meter
2P	Kilobyte	FA	Fahrenheit
2Z	Millivolts	G9	Gigabyte
4K	Milliamperes	HJ	Horsepower
4L	Megabytes	HP	Millimeter H_20
4S	Pascal	HZ	Hertz
70	Volt	NU	Newton-Meter

NOTE: A full list of codes representing unit of measurement is available from: http://autoid.org/ANSI_MH10/ansi_mh10sc8_wg2.htm

ANNEX N: DATA CONSTRUCTS; INFORMATIVE

This table is shown separately because the Data Constructs Steering Committee expects that some Data Formats can be used for different applications in a non-conflicting manner. The Data Formats are shown as decimal values.

Table 35: Data Formats

Data Format (Decimal)	Assigned Organization or Function	Root-OID	Comments
0	Not-Formatted	Not applicable	This value is the default for an RFID tag yet to be formatted, so the system may assume that the tag has no encoded data. The DSFID combining the Data Format with the Access Method) shall be $00_{HEX.}$
1	Full-Featured	Not applicable	This is used where each OID has to be encoded in full, i.e., without truncating the Root-OID. This is relevant where the RFID tag is likely to contain a mixture of OIDs from different domains, or one from a minor domain.
2	Root-OID-Encoded	Encoded on the tag	This is useful for small domains requiring to encode a set of data. This data format requires the Root-OID to be encoded but then truncates all OIDs so that only the Relative-OID needs to be encoded.
3	ISO-15434	1 0 15434	This standard supports a number of message structures originally intended for two-dimensional barcodes.
4	ISO-6523	1 3	This standard deals with the registration of domains to the level of the International Code Designator (ICD). Examples include DUNS, EAN location codes, all internet IP addresses, NATO, plus many government and multinational company schemes. This data format allows existing organization structures to be supported with the minimum of change. (See Note “ICD Issues.”)
5	ISO-15459	1 0 15459	ISO/IEC 15459 provides a mechanism to support unique identification codes across multiple industry and commercial sectors. The various parts of this standard provide the basis for track and trace codes for all levels in supply chains.
6	ISO-28560-2	1 0 15961 6	This standard addresses the encoding of data on RFID tags in the library community and defines all the Relative-OID values.
7	RESERVED	—	—
8	ISO-15961-Combined	1 0 15961	This allows any combinations of OID that have the common Root-OID {1 0 15961} to be encoded. The Relative-OID then becomes the next arc (which can differ for different objects), followed by another arc identifying the particular data element. The structure was developed so that two or

Data Format (Decimal)	Assigned Organization or Function	Root-OID	Comments
			more related organizations could share encoding - by agreement – without one having all the encoding efficiencies. Joint registration by the relevant organizations should be a requirement for data constructs to be assigned.
9	GS1-AI	1 0 15961 9	This data format enables GS1 Application Identifiers to be used as the Relative-OID in an Object Identifier structure.
10	ANS-DI-Algorithm	1 0 15961 10	This data format enables ANS MH10 Data Identifiers to be used as the Relative-OID in an Object Identifier structure. The alphanumeric DI is converted to a Relative-OID using an algorithm defined in ISO/IEC 15961:2004
11	RESERVED	—	—
12	IATA-Baggage	1 0 15961 12	The Relative-OID values are defined in the IATA RP1740C standard.
13	ANSI-DI-Mapping Table	1 0 15961 13	This data format enables ANS MH10 Data Identifiers to be used as the Relative-OID in an Object Identifier structure in a more efficient encoding manner than data format 10. The current mapping table is available at: http://www.autoid.org/ANSI_MH10/ansi_mh10sc8_wg2.htm
14 to 28	RESERVED	—	—
29	Closed system data fully encoded to ISO/IEC 15962 rules	The Root-OID is implied.	This Data Format enables closed system or prototype data to be encoded in a manner compliant with the 15962 encoding rules (i.e., by declaring the relevant Access Method). Because those implementing the application know the interpretation of the Relative-OID values, the Root-OID is implied and does not require to exist or to be registered.
30	Closed system data not encoded to ISO/IEC 15962 rules	Not applicable	The complete DSFID shall be 00011110, to indicate that the encoded bytes are passed through unchanged in ISO/IEC 15962 and ISO/IEC 24791 implementations.
31	Extended Data Format	Not applicable	Various extension mechanisms are included in ISO/IEC 15962 Rev1 for the DSFID; one enables an additional 256 Data Format to be assigned. If the First DSFID has the value xxx11111, then the Data Format that is encoded is one that is defined in the following rows on this table.
32 to 287	RESERVED		These values are reserved for open system applications as registered under Part 2 of ISO/IEC 15961, when an extension mechanism is invoked under ISO/IEC 15962 rules for a multiple-byte **DSFID**.

Table 36: Access Methods

Method	Structure	Comments
0	No-Directory	This **Access-Method** provides the simplest set of encoding rules. The basic rule is to link a data **Object** to an **Object-Identifier** and support this with appropriate syntactical components to create a Data-Set. The **Data-Sets** are concatenated to produce a contiguous sequence of encoded bytes in the Logical Memory Map.
1	Directory	The **Directory** structure supports all the encoding rules of the **No-Directory** structure but additionally supports the encoding of a directory at the higher address values within the Logical Memory Map on the RFID tag. When the **Directory** structure is used, any command seeking to selectively read data can first access the directory to find the memory address of the beginning of the **Data-Set**, and then go to that location to read the bytes that represent the encoding of the **Data-Set**. **NOTE:** The Directory may be written to the RFID at a later time than the initial encoded data.
2	Packed-Objects	The **Packed-Object** structure has been introduced in the first revision of ISO/IEC 15962 and this edition of this part of ISO/IEC 15961. The encoding scheme is fundamentally different because it takes a set of **Object-Identifiers** and their **Objects** and encodes them in an indexed structure that integrates compaction and encoding. The **Packed-Object** encoding scheme requires a rule-based table for each **Data-Format**, which calls up specific compaction schemes for the individual elements. The compaction schemes are standard, irrespective of the **Data-Format**. The **Packed-Object** scheme requires more complex encoding and decoding rules than the **No-Directory** and **Directory Access-Methods** but offers significant encoding efficiency even over the basic **No-Directory** structure when a number of **Object-Identifiers** need to be encoded. This can reduce the amount of memory required on the RFID tag and can reduce the size of message transferred across the air interface in response to a read command. Application domains need to define a table of indexed **Relative-OIDs** for this **Access-Method** to be implemented. It can only be applied as an alternative to the **No-Directory** or **Directory Access-Methods** on a particular RFID tag; however, tags with any of the **Access-Methods** may be intermixed in the same application.
3	Tag-Data-Profile	The **Tag-Data-Profile** scheme is intended to support fixed message structures, typically where a number of **Object-Identifiers** and their **Objects** need to be encoded. The fixed message structure is best suited for applications that are reasonably homogeneous and additionally have a consistent requirement for exactly the same **Object-Identifiers** to be encoded on all RFID tags in the application. Each **Tag-Data-Profile** requires registration.
4	Multiple Records	The **Multiple-Records** encoding process overlays a structure onto the **No-Directory**, **Packed-Objects**, and **Tag-Data-Profile Access-Methods** for multiple instances of these, and even a mixture of these

		Access-Methods, to be encoded in the same Logical Memory. This is achieved through the introduction of an encoded MR-header and a preamble for each record. This allows the encoding of individual records to fully comply with the inherent **Access-Method**. This also enables the same **Object-Identifier** to be encoded in separate records, with one rule that supports a list of the same data element in one record. There are three main classes of application supported by the **Access-Method** and these can also be intermixed. One class enables different data formats and owners of data to share the same RFID tag but under the control of the original owner of the tag. Another supports a time sequence of history records of the same type to be repeated. The third major class supports hierarchically related records, e.g., for supply chain delivery or bill of material types of structure.
5-15	RFU	Reserved for future encoding schemes to be defined in ISO/IEC 15962 and invoked under rules for a multiple byte **DSFID**. Some of the new **Access-Methods** might be possible to be inherent encoding schemes in **Multiple-Records**. This can only be determined as the encoding schemes are developed.

ANNEX O: ANSI DATA IDENTIFIER REQUEST FORM

Rev. ANS MH10.8.2-9354 (DI)

Reference: ___________________

Date: _______________________

ANS MH10.8.2 DATA IDENTIFIER REQUEST FORM

ALL REQUESTS SHALL BE TYPED or printed legibly in black ink. Complete all parts. Submit to:

Craig K. Harmon, Chairman, ASC MH 10 Data Identifier Maintenance Committee

c/o Q.E.D. Systems

3963 Highlands Lane, SE, Cedar Rapids, IA 52403-2140 USA

(V): +1 319/364-0212 * (M): +1 319/533-8092

(E): craig.harmon@qed.org * (U1): http://www.autoid.org

Incomplete forms or those with inadequate support for the change requested will be returned to the submitter. Submitters are notified of the status of the work request following review by the ANS MH10.8.2 DI Maintenance Committee.

Request for: ______ New Data Identifier

______ Data Identifier Interpretation

Organization: ____

Contact Person: ____

Address: ____

____

____

Telephone: ____

Telefax: ____

1. **PROPOSED DATA IDENTIFIER**

Provide a short description (20 words or fewer) that would be included as a description for the proposed Data Identifier. For an interpretation, provide a comprehensive description of the aspect of the identifier that needs interpretation.

Page 2 (Data Identifier Request)

2. **BUSINESS CASE**

Explain why you need the proposed assignment. Provide a complete scenario that tells what the business function, operation, or problem is that will be satisfied by a new assignment to the ANS MH10.8.2 Data Identifier Standard. If the proposed DI is already in use by your organization, please identify how long this identifier has been in use and other organizations you are aware of who employ the same identifier. The ANS MH10.8.2 DI Maintenance Committee requires enough information to be able to propose an alternate solution if necessary. Be specific because this will also appear in the ANS MH10.8.2 Voting Package and will be the only information that voters have on which to base their vote.

Page 3 (Data Identifier Request)

3. **DEFINITIONS**

Definitions for new assignments and for industry-specific terms shall be complete. For a new ANS MH10.8.2 DI, provide a proposed assignment and a DI definition.

RULES:

(1) Acronyms/abbreviations cannot be added to the standards - they shall be spelled out.

(2) Provide an expanded assignment definition for each DI that is not completely self-explanatory, that is, terms that are not in general business use or that are industry-specific.

(3) Provide code source references for all externally published (non-ANS MH10.8.2) code lists cited (use the Form for New or Revised Code Source Reference). If one exists, provide a precise description of the structure of the data as foreseen by your organization for this application. Indicate data elements involved and their format (numeric, alphanumeric, fixed or variable length, number of decimals). Indicate the business function of each data element in the application.

Page 4 (Data Identifier Request)

4. **MEDIA AND APPLICATION USE**

- With what media (e.g., bar code, 2D symbol, RF tag, etc.) do you intend to use the proposed Data Identifier?

- At what stage will the Data Identifier and data be created and applied?

- On to what and when will the media be applied (package, label, tag, document, etc.)?

- Why does the information need to be machine-readable?

- When and where are the media read?

- Describe the use of the Data Identifier by users other than the originator.

- What is the number of potential users?

5. **JUSTIFICATION**

Describe the benefits (hard and soft savings) expected from the application.

6. **ADDITIONAL INFORMATION**

Feel free to attach any additional information related to your organization and the application.

Date: ____________________ Signature: ______________________

Page 5 (Data Identifier Request)

Data Identifier Data Dictionary Record

<table>
<tr><th colspan="2">Data Dictionary Detailed Entry</th><th colspan="2"></th></tr>
<tr><td>NAME:</td><td colspan="2">Version</td><td>Key</td></tr>
<tr><td colspan="2">XML Tag:</td><td colspan="2">DI:</td></tr>
<tr><td colspan="4">Definition:</td></tr>
<tr><td colspan="4">Class: Numeric/Alpha/Alphanumeric/Binary Remarks:
Decimals: Yes (define if included in data) / No
Min_Length:
Max_Length:
Case Sensitive: Yes/No</td></tr>
<tr><td colspan="4">Business Rules:</td></tr>
<tr><td colspan="4">Data Element Source/Authority:</td></tr>
</table>

<table>
<tr><th colspan="4">APPLICATION AREAS</th></tr>
<tr><th>Area</th><th>Application</th><th>Category</th><th>Remarks</th></tr>
<tr><td></td><td></td><td></td><td></td></tr>
<tr><th colspan="4">USES</th></tr>
<tr><th>Application Area</th><th>Usage</th><th>Type</th><th>Specific Use</th></tr>
<tr><td></td><td></td><td></td><td></td></tr>
<tr><th colspan="4">ALIAS: e.g., Production Date</th></tr>
<tr><th colspan="4">Table footnote.</th></tr>
</table>

ANNEX P: MAINTENANCE REQUEST

If you find an error or other changes that should be made to this publication, please complete this form and return it to the proper address below.

Name of Submitter: ______________________ Date: ______________________

Company: ______________________

Company Address: ______________________

Phone: ______________ Fax: ______________ E-mail: ______________

CHANGE REQUEST (Use additional sheets if necessary)
Page Number of Change:
Document Currently Reads:

Recommended Changes/Should Read:

Recommended Additions:

Reason for Change:

Signature of Submitter: ______________________

Submit This Change Request to Your Organization at the Address Listed Below:

Automotive Industry Action Group
26200 Lahser Road, Suite 200
Southfield, MI 48033
USA
Phone: (248) 358-3570
Fax: (248) 358-3253
Web: www.aiag.org

Odette International Limited
71 Great Peter Street
London SW1P 2BN
UK
Phone: +44 207 344 9227
Fax: +44 207 235 7112
Web: www.odette.org

Japan Automobile Manufacturers' Association, Inc. (JAMA)
Jidosha Kaikan
1-30, Shiba Daimon 1 chome, Minato-ku
Tokyo 100-0004
Japan
Phone: +81 3-5405-6130
Fax: +81 3-5405-6136
Web: www.jama.or.jp

Japan Auto Parts Industries Association (JAPIA)
Jidosha Buhin Kaikan, 5th Floor,
1-16-15 Takanawa, Minato-ku
Tokyo 108-0074
Japan
Phone: +81 3-3445-4211
Fax: +81 3-3447-5372
Web: www.japia.or.jp