

RAIN Alliance Tag Encoding Guideline for use with the RAIN Application Family Identifier (AFI)

DISCLAIMER: This document, and all other information, materials, or services, if any, provided by RAIN Alliance in connection with this document, are provided "as is," and RAIN Alliance makes no representations or warranties, express, implied, statutory, or otherwise, and expressly disclaims any representation or warranty that implementation of any technical or business specifications or methods portrayed in this document will not infringe any third-party intellectual property rights, as well as any implied warranties of merchantability, fitness for a particular purpose, correctness, accuracy, reliability, or any equivalents under the laws of any jurisdiction that might arise from products, activities, or information disclosures relating to this document, or any act, omission, or requirement by any third party. If you do not understand or agree with the foregoing, you should not access this document or implement any element of it.

RAIN RFID Alliance Guideline

V.1.0 September 2021

RAIN Alliance Tag Encoding Guideline

1. Introduction

The RAIN Alliance has a registered ISO/IEC 15961 Application Family Identifier (AFI) and it is the hex value "0xAE". The use of the RAIN AFI shall indicate that the tag encoding is for a closed system application using a proprietary or vendor-defined data format. The tag encoding shall include a Company Identifying Number (CIN) assigned by the RAIN Alliance to identify the owner of the tag data.

2. Tag encoding

This tag encoding guideline applies only to Memory Bank 01 (UII Memory). MB01 contains three mandatory addressed areas and two optional addressed areas:

1. **StoredCRC** – This is a mandatory 16-bit value stored in UII Memory at Word 0x00 (bit addresses 0x00 to 0x0F). The value is calculated by the tag itself.

2. **StoredPC** (a.k.a. PC Word) – This is a mandatory 16-bit value stored in UII Memory at Word 0x01 (bit addresses 0x10 to 0x1F). The PC Word contains the following fields:

a. L (Length) – This 5-bit value is stored at bit addresses 0x10 to 0x14. L shall be set accordingly for the number of 16-bit words necessary for the UII (see below) in the tag reply to an *ACK* command.

b. **UMI** (User Memory Indicator) – This 1-bit value is stored at bit address 0x15. **UMI** is set by the tag itself.

c. XI (XPC_W1 Indicator) – This 1-bit value is stored at bit address 0x16. This bit indicates if XPC_W1 is included in the tag reply to an *ACK* command. XI is set by the tag itself.

d. **T** (Toggle Indicator) – This 1-bit value is stored at bit address 0x17. **T** shall be set **T**=1 when using the RAIN AFI.

e. **AFI** (Application Family Identifier) – This 8-bit value is stored at bit addresses 0x18 to 0x1F. **AFI** shall be set **AFI**=0xAE when using the RAIN AFI.

3. **UII** (Unique Item Identifier) – This is a mandatory area of variable length and starts in UII Memory at Word 0x02 (bit addresses 0x20 and higher). The **UII** shall begin with the CIN assigned by the RAIN Alliance starting at bit address 0x20. The CIN shall be encoded using the EBV-8 format described below. The remainder of the **UII** encoding following the CIN shall be considered as proprietary or vendor-defined encoding that is defined by the entity identified by the CIN.

4. **XPC_W1** – This is an optional 16-bit value stored in UII Memory at Word 0x21 (bit addresses 210 to 21F). If a tag implements XPC_W1, then most of the value is calculated by the tag itself. **XPC_W1** contains the following fields, which are recommended to be set appropriately for the object associated with the tag:

a. **TN** (Tag Notification Indicator) – This 1-bit value is stored at bit address 0x21B. This is an optional bit that is IC vendor defined and may or may not be supported by the tag.

b. **NR** (Nonremovable Indicator) – This 1-bit value is stored at bit address 0x21E. It is recommended that **NR** be set as appropriate for the tagged object.

c. **H** (HazMat Indicator) – This 1-bit value is stored at bit address 0x21F. It is recommended that **H** be set as appropriate for the tagged object.

5. **XPC_W2** – This is an optional 16-bit value stored in UII Memory at Word 0x22 (bit addresses 220 to 22F). If a tag implements XPC_W2, then the value is calculated by the tag itself.

3. EBV-8 encoding format

EBV (Extensible Bit Vector) is a data structure with an extensible range. An EBV is an array of blocks with each block containing a single extension bit followed by a specific number of data bits. For EBV-8, there are 8 bits in one block and each block contains an extension bit followed by 7 data bits.

The data value represented by an EBV-8 is simply the bit string formed by the data bits as read from left-to-right, ignoring the extension bits. Because each block has 7 available data bits, an EBV-8 can represent numeric values between 0 and 127 with a single block. To represent the value 128, set the extension bit to 1 in the first block, and append a second block to the EBV-8. In this manner, an EBV-8 can represent arbitrarily large data values. RAIN CIN values are in the range 0 to 99,999,999 so the corresponding EBV-8 encoding will require from 1 to 4 bytes of memory.

Numbe	er	EBV-8 Encoding										
(i.e. RAIN	CIN)	I	Byte 1		Byte 2	Byte 3		l	Byte 4			
		ext	data	ext	data	ext	data	ext	data			
0	2 ⁰ – 1	0	0000000		n/a	n/a		n/a				
1	2 ⁰	0	0000001		n/a		n/a		n/a			
127	2 ⁷ – 1	0	1111111		n/a		n/a	n/a				
128	2 ⁷	1	0000001	0	0000000		n/a		n/a			
16,383	2 ¹⁴ – 1	1	1111111	0	1111111		n/a		n/a			
16,384	2 ¹⁴	1	0000001	1	0000000	0	0000000		n/a			
2,097,151	2 ²¹ – 1	1	1111111	1	1111111	0	1111111		n/a			
2,097,152	2 ²¹	1	0000001	1	0000000	1	0000000	0	0000000			
268,435,453	2 ²⁸ – 1	1	1111111	1	1111111	1	1111111	0	1111111			

4. UII encoding examples

The following examples illustrate the proper tag encoding for the **UII** based on the EBV-8 encoding length required for the CIN issued by RAIN. All the examples assume the most common of RAIN tags which have MB01 (UII Memory) comprised of the 16-bit **StoredCRC**, the 16-bit **StoredPC**, and a 96-bit **UII** encoding.

4.1. UII encoding for a 1-byte CIN

Assume an entity is issued the 2-digit CIN = 12 by RAIN, which is the 1-byte value 0x0C using EBV-8 encoding. The UII memory is then encoded as follows with the 96-bit UII shaded in green:

MB01 Bit Address	MB01 Word Address	msb 0	1	2	3	4	5	6	7	8	9	А	В	С	D	E	lsb F		
$00_h - 0F_h$	00 _h		StoredCRC																
10 _h – 1F _h	01 _h		StoredPC																
IOn – IFn	UTh		L=	<u>=0011</u>	<mark>0</mark> 2		UMI	XI	T=1	AFI = 0xAE									
$20_h-2F_h \\$	02h			CIN	byte	1 = 0	x0C			data byte 1									
$30_h - 3F_h$	03 _h				data I	oyte 2	2			data byte 3									
$40_h - 4F_h \\$	04 _h				data I	oyte 4	ł			data byte 5									
$50_h - 5F_h$	05h				data I	oyte 6	6			data byte 7									
$60_h - 6F_h$	06h		data byte 8								data byte 9								
70_h-7F_h	07 _h			C	data b	yte 1	0			data byte 11									

Memory Bank 01 (UII Memory)

4.2. UII encoding for a 2-byte CIN

Assume an entity is issued the 4-digit CIN = 1,234 by RAIN, which is the 2-byte value 0x8952 using EBV-8 encoding. The UII memory is then encoded as follows with the 96-bit UII shaded in green:

				Men	nory	Bank	(01 (UII N	/lemo	ory)							
MB01 Word Address	msb 0 1 2 3 4 5 6 7 8 9 A B C D											E	lsb F				
00h	StoredCRC																
01 _b		StoredPC															
011		<mark>=0011</mark>	<mark>0</mark> 2		UMI	XI	T=1				<mark>AFI =</mark>	0xAE					
02h				<mark>l byte</mark>	1 = 0	<mark>)x89</mark>			CIN byte 2 = 0x52								
03 _h				data k	oyte 1				data byte 2								
04 _h				data I	oyte 3	3			data byte 4								
05h				data I	oyte 5	5			data byte 6								
06 _h				data I	oyte 7	,			data byte 8								
07 _h				data I	oyte S)			data byte 10								
	Word Address 00h 01h 02h 03h 04h 05h 06h	Word Address Insp 0 00h 0 01h	Word Address IIISD 0 1 00h 0 1 00h	MB01 Word Address msb 0 1 2 00h 1 2 00h	MB01 Word Address msb 0 1 2 3 00h 1 2 3 00h	MB01 Word Address msb 0 1 2 3 4 00h 1 2 3 4 00h	MB01 Word Address msb 0 1 2 3 4 5 00h 1 2 3 4 5 01h 1 1 2 3 4 5 01h 1 1 2 3 4 5 02h 1 1 2 3 4 5 03h 1 1 2 3 4 5 04h 1 1 2 3 4 5 04h 1 1 2 3 4 5 04h 1 1 1 1 1 1 1 1 04h 1 1 1	$\begin{array}{c c c c c c c c c c c c c c c c c c c $	MB01 Word Address msb 0 1 2 3 4 5 6 7 00h 1 2 3 4 5 6 7 00h Image: Stress of the st	$\begin{array}{c c c c c c c c c c c c c c c c c c c $		MB01 Word Address msb 0 1 2 3 4 5 6 7 8 9 A 00h Image:	MB01 Word Address msb 0 1 2 3 4 5 6 7 8 9 A B 00h IIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIII	MB01 Word Address msb 0 1 2 3 4 5 6 7 8 9 A B C 00h	MB01 Word Address msb 0 1 2 3 4 5 6 7 8 9 A B C D 00h 00h 5 5 6 7 8 9 A B C D 00h StoredCRC 01h L=001102 UMI XI T=1 AFI = 0xAE 02h CIN byte 1 = 0x89 02h CIN byte 1 = 0x89 CIN byte 2 = 0x52 03h data byte 3 data byte 2 data byte 2 04h data byte 4 data byte 5 data byte 6 data byte 6 06h data byte 7 4ata byte 8 data byte 8	MB01 Word Address msb 0 1 2 3 4 5 6 7 8 9 A B C D E 00h Image: Image	

4.3. UII encoding for a 3-byte CIN

Assume an entity is issued the 6-digit CIN = 123,456 by RAIN, which is the 3-byte value 0x87C440 using EBV-8 encoding. The UII memory is then encoded as follows with the 96-bit UII shaded in green:

					IVIEI	noi y	Bauk	(OT (VIEIII	JIYJ						
MB01 Bit Address	MB01 Word Address	msb 0	1	2	3	4	5	6	7	8	9	A	В	С	D	E	lsb F
$00_h-0F_h \\$	00h		StoredCRC														
10 _h – 1F _h	01 _h								redPC								
	UTh		L=	<mark>-0011</mark>	<mark>0</mark> 2		UMI	XI	T=1	AFI = 0xAE							
20_h-2F_h	02 _h			CIN	<mark>l byte</mark>	1 = ()x87			CIN byte 2 = 0xC4							
$30_h - 3F_h$	03 _h			CIN	<mark>l byte</mark>	3 = (<mark>)x40</mark>			data byte 1							
$40_h - 4F_h \\$	04h				data I	oyte 2	2			data byte 3							
$50_h - 5F_h$	05h				data I	oyte 4	ł			data byte 5							
$60_h - 6F_h$	06h				data I	oyte 6	6						data I	oyte 7			
$70_h - 7F_h$	07 _h				data I	oyte 8	}						data I	oyte 9			

Memory Bank 01 (UII Memory)

4.4. UII encoding for a 4-byte CIN

Assume an entity is issued the 8-digit CIN = 12,345,678 by RAIN, which is the 4-byte value 0x85F1C24E using EBV-8 encoding. The UII memory is then encoded as follows with the 96-bit UII shaded in green:

	-				Men	nory	Bank	: 01 (UII N	/lemo	ory)						
MB01 Bit Address	MB01 Word Address	msb 0	1	2	3	4	5	6	7	8	9	А	в	С	D	E	lsb F
$00_h - 0F_h$	00h		StoredCRC														
10 _h – 1F _h 01 _h	01	StoredPC															
	UIn		L=	- <mark>0011</mark>	<mark>0</mark> 2		UMI	XI	T=1				<mark>AFI =</mark>	0xAE			
20_h-2F_h	02h			CIN	<mark>l byte</mark>	1 = 0	<mark>x85</mark>		•	CIN byte 2 = 0xF1							
$30_h - 3F_h$	03h			CIN	byte	3 = 0	xC2			CIN byte 4 = 0x4E							
40_h-4F_h	04 _h				data I	oyte 1				data byte 2							
$50_h - 5F_h$	05h				data I	oyte 3	5			data byte 4							
60_h-6F_h	06 _h				data I	oyte 5	;			data byte 6							
$70_h - 7F_h$	07h				data I	oyte 7	•			data byte 8							

5. Background and Contributors

This document was developed within the RAIN Alliance Application ID Workgroup. The following contributors played a major role in shaping the final document:

Jim Springer Bertus Pretorius Megan Brewster Steve Halliday Michael Fein (EM Microelectronic) (Tönnjes) (Impinj) (RAIN RFID) (Zebra)

ABOUT RAIN RFID ALLIANCE

The RAIN RFID Alliance is an organization supporting the universal adoption of RAIN UHF RFID technology. A wireless technology that connects billions of everyday items to the internet, enabling businesses and consumers to identify, locate, authenticate, and engage each item. The technology is based on the EPC Gen2 UHF RFID specification, incorporated into the ISO/IEC 18000-63 standard.

Join the RAIN RFID Alliance to enable connectivity for your business and consumers: identify, locate, authenticate, and engage items in our everyday world. For more information, visit <u>www.RAINRFID.org</u>.



RAIN RFID Alliance

401 Edgewater Place, Suite 600 Wakefield, MA, 01880, USA

Visit the RAIN RFID website - RAINRFID.org.

If you are interested in learning more about the RAIN RFID Alliance, contact us at info@rainrfid.org.