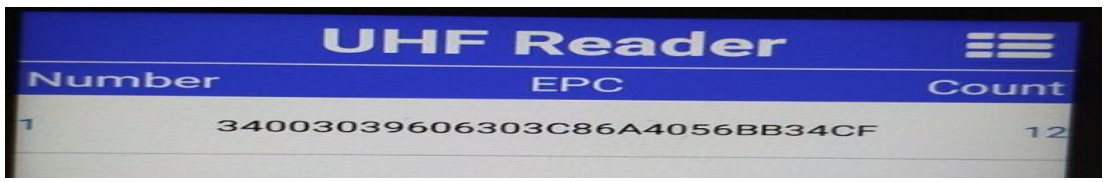
Title	**RFID Data - GS1 - DATA - ENCODING - DECODING Procedure**
Prepared By	**K.Mohamed illyas**
Version	**V1.0**
Date	**19-10-2022**

Decathlon Garment RFID data GS1 Standard data - Decoding Procedure

Garment Tag (GTIN14) : (01) 03608449916578(21) 001455109327

Corresponding Hex when scanned this tag using rfid reader :34003039606303C86A4056BB34CF

Decoding Procedure :



Filter first 4 digit (PC + CR)

Hex value to be converted : 3039606303C86A4056BB34CF

Binary: : 00110000001110010110000001100011000000111100100001101010010000000101011010111011000110100110011111

Partition : 0011 0000 --- 001 --- 110 --- 0101 1000 0001 1000 1100 ---- 0000 1111 0010 0001 1010 1001 ---- 00000001010110101110110011010011001111

Header ==== 0011 0000 - 8bit

filter ==== 001 - 3 bit

partition === 110 - 3 bit

GS1 comapny ==== 0101 1000 0001 1000 1100 - 20 bit === Binary to decimal----> 360844 --- 6digit

Item ===== 0000 1111 0010 0001 1010 1001 - 24 bit ==== Binary to decimal ------> 991657 – A zero to be added infront for 7 digit ---0991657

Serial no ===== 00000001010110101110110011010011001111 ==== 38 bit Binary to decimal------> 1455109327

Now concatenate values as below :

360844.0991657.1455109327

Then **360844.0991657.1455109327**

0360844991657.1455109327

Now need to find the check sum for GTIN 14

	0	3	6	0	8	4	4	9	9	1	6	5	7
Step1: Multiply	x	x	x	x	x	x	x	x	x	x	x	x	x
	3	1	3	1	3	1	3	1	3	1	3	1	3
Step 2	0	3	18	0	24	4	12	9	27	1	18	5	21

Total sum = 142

Step 3 : Subtract the sum from nearest equal or higher multiple of ten = 150 – 142 = **8**

Check sum = 8

Add (01) – GS1 application identifier for GTIN 14 ;;;; (21) - GS1 application identifier for serial number

Finally Tag will (01) 03608449916578(21) 001455109327.

Decathlon Component RFID data GS1 Standard data – Encoding Procedure

Component Tag Formation by GS1 – GTIN14 Standard : Ex: (01)02112345670003(21)000000000001

GS1 application identifier for GTIN 14 – (01) – Default

EAN13 Prefix - 021 – Default

Component Item code – 1234567

Complettion status – 000

check sum – 3

GS1 application identifier for serial number - (21) – Default

Serial Number – 000000000001

Check digit calculation :

	0	2	1	1	2	3	4	5	6	7	0	0	0
Step1 : Multiply	x	x	x	x	x	x	x	x	x	x	x	x	x
	3	1	3	1	3	1	3	1	3	1	3	1	3
Step 2	0	2	3	1	6	3	12	5	18	7	0	0	0

Total sum = 57

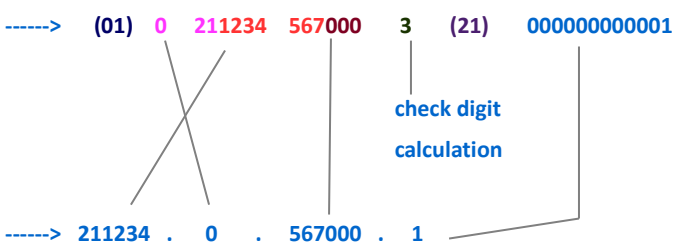
Step 3 : Subtract the sum from nearest equal or higher multiple of ten = 60 – 57 = 3

Check sum = 3

Step 1 : Parse data for EPC Pure Identity URI Generation

Parse the Data as Below for GS1 Element string to EPC Pure Identity URI Generation :

(01)02112345670003(21)000000000001



Step 2 : EPC Pure Identity URI Generation format

EPC storage bit = 96 - Default

Filter value = 3 (reserved)

EPC Pure Identity URI Generation format : E**pcbit .filter.GS1 comapny prefix.Item ref indicator.Serial**

Now align the above data as in EPC URI : **96:3: 211234.0567000.1**

Step 3: EPC Tag URI TO EPC Binary Encoding format

Important Point to Note :

Header ==== 8bit

filter ==== 3 bit

partition === 3 bit

GS1 comapny ==== 20 bit - 6 digit

Item ===== 24 bit - 7 digit

Serial no ===== 38 bit - 12 digit

Header	Filter	partition	GS1 Company prefix (20 bit)	Item / indicator (24 bit)	Serial (38bit)
	3	.	**211234**	**567000**	1
'00110000'	'011'	'101'	'00110011100100100010'	'000010001010011011011000'	'00000000000000000000000000000000000001'

Step 3: EPC Binary data concatenation :

Generate EPC Binary by concatenate all the values :

00110000011101001100111001001000100000100010100110110110000000000000000000000000000000000000000001

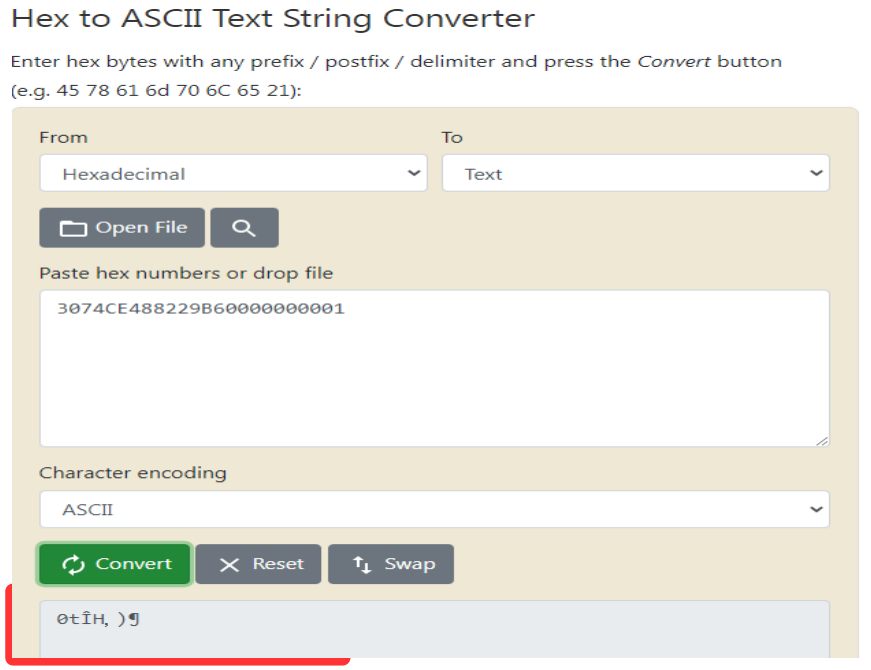
This binary data has to be converted into hex and to be written in the RFID tag .

Step 4: Read the RFID tag in device :

While we scan the tag in RFID reader value shown will be " 3074CE488229B60000000001"

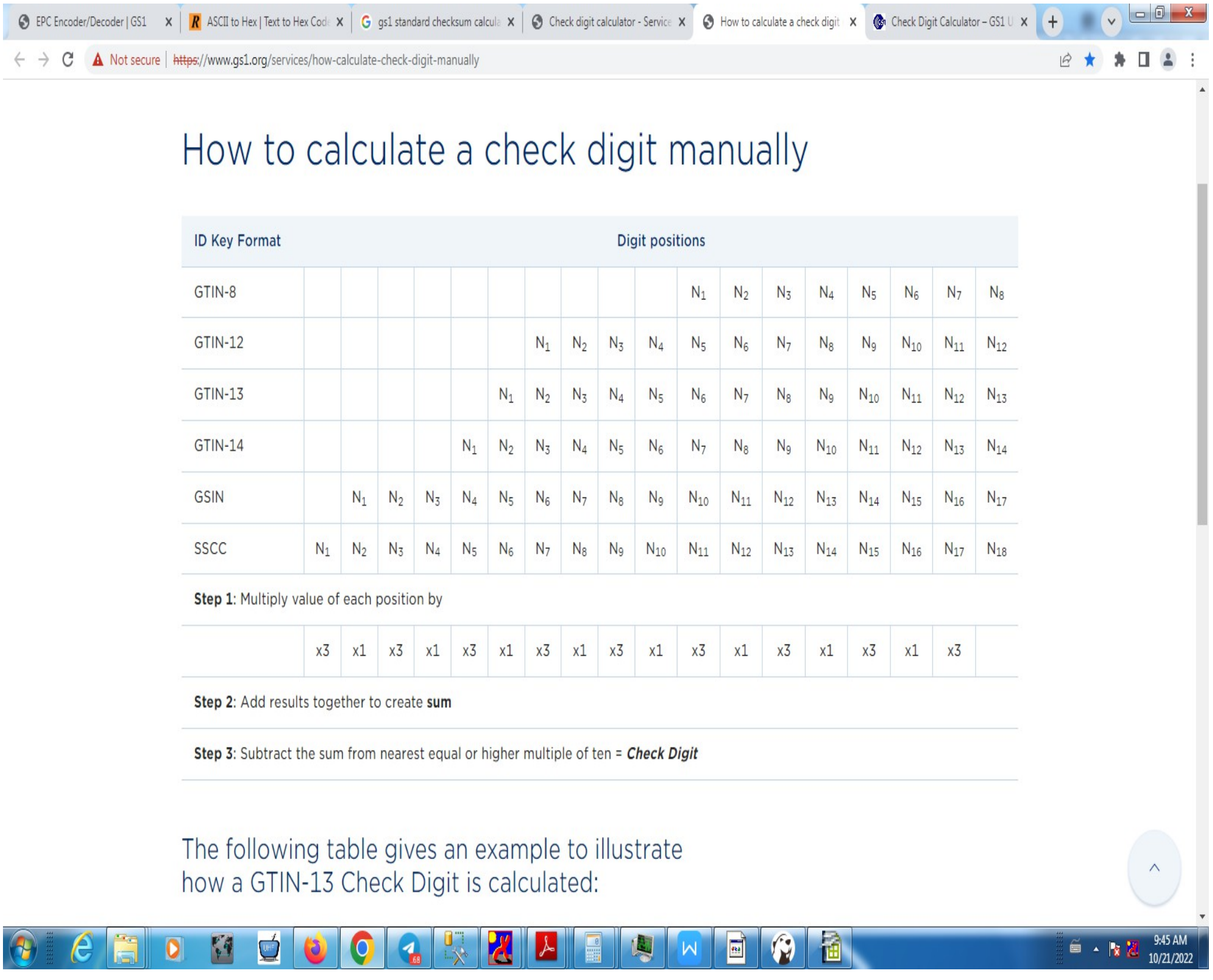
This value is based On Converting the above binary value to Hex :

3074 CE488229B60000000001



Value will be encoded and Ascii value can't be found .

Check sum calculation procedure :



EPC Encoder/Decoder | GS1 | ASCII to Hex | Text to Hex Cod | gs1 standard checksum calcula | Check digit calculator - Service | How to calculate a check digit | Check Digit Calculator – GS1 U

Not secure | https://www.gs1.org/services/how-calculate-check-digit-manually

How to calculate a check digit manually

ID Key Format	Digit positions																
GTIN-8										N_1	N_2	N_3	N_4	N_5	N_6	N_7	N_8
GTIN-12							N_1	N_2	N_3	N_4	N_5	N_6	N_7	N_8	N_9	N_{10}	N_{11}	N_{12}
GTIN-13						N_1	N_2	N_3	N_4	N_5	N_6	N_7	N_8	N_9	N_{10}	N_{11}	N_{12}	N_{13}
GTIN-14					N_1	N_2	N_3	N_4	N_5	N_6	N_7	N_8	N_9	N_{10}	N_{11}	N_{12}	N_{13}	N_{14}
GSIN		N_1	N_2	N_3	N_4	N_5	N_6	N_7	N_8	N_9	N_{10}	N_{11}	N_{12}	N_{13}	N_{14}	N_{15}	N_{16}	N_{17}
SSCC	N_1	N_2	N_3	N_4	N_5	N_6	N_7	N_8	N_9	N_{10}	N_{11}	N_{12}	N_{13}	N_{14}	N_{15}	N_{16}	N_{17}	N_{18}
Step 1: Multiply value of each position by																		
	x3	x1	x3	x1	x3	x1	x3	x1	x3	x1	x3	x1	x3	x1	x3	x1	x3	
Step 2: Add results together to create **sum**																		
Step 3: Subtract the sum from nearest equal or higher multiple of ten = ***Check Digit***																		

The following table gives an example to illustrate how a GTIN-13 Check Digit is calculated:

9:45 AM
10/21/2022