APPENDIX C - BEACON ENCODED LOCATION CODING

C.1 ENCODED LOCATION PROTOCOL

This section defines the encoded location protocol which can be used with the 406 MHz beacon message formats for encoding beacon position data in the digital message transmitted by a 406 MHz distress beacon.

C.2 Summary

Encode location is represented differently in second generation beacons. In first generation beacons, degrees, minutes and seconds were used. For second generation beacons, degrees and decimal part of the degrees are used. Moreover, there is no coarse and fine offset fields: all information appears in one field. Moreover, there is no separate return link location protocol. Instead, RLS data is contained in one of the rotating fields.

C.3 Default Values in Position Data

The following default values shall be used in all encoded position data fields of the location protocols, when no valid data is available:

a) all bits in the degrees are set to "1", with N/S, E/W flags set to "0"; and

b) the bits in the decimal parts of the degrees are set to the following patterns:
i. Latitude: 000001111100000 (Note the first five decimal parts of degrees bits are set to “0”s, the middle five bits are set to “1” s and the last 5 bits are set to “0”s)

ii. Longitude: 111110000011111 (Note the first five decimal parts of degrees bits are set to “1”s, the middle five bits are set to “0” s and the last 5 bits are set to “1”s)

The default pattern shall also be transmitted in the self-test mode. Additionally, if a location protocol beacon includes an optional GNSS self-test and this fails to provide a valid location to encode into the transmitted self-test message, then the beacon may radiate a single self-test message with the above default data. However if a location protocol beacon with optional GNSS self-test obtains a location, then the beacon shall radiate a single self-test message with encoded position.

This default bit pattern is different from first generation beacons. The degree bits are all set to “1”s as in first generation beacons. But it is not easy to set the minutes bit to “0”s and the seconds bits to “1”s as the decimal parts do not fall on integer minute or second boundaries.

C.4 **Definition of Location Data Fields**

The general structure of encoded location data is illustrated below.

C.4.1 Encoded Location field

The 47 bits available in the main digital message are defined as follows:

a) bits 44-66: latitude data (23 bits), including:

- bit 44: N/S flag (N=0, S=1)
- bits 45-51: degrees (0 to 90) in 1 degree increments
- bits 52-66: decimal parts of degrees

b) bits 67-90: longitude data (24 bits), including:

- bit 67: E/W flag (E=0, W=1)
- bits 68-75: degrees (0 to 180) in 1 degree increments
- bits 76-90 decimal parts of degrees

C.4.2 Encoded Location Data (1)

All position information is encoded as degrees, and decimal parts of latitude or longitude. Latitude and longitude data are rounded off (i.e., not truncated) to the available resolution. All rounding shall follow normal rounding conventions, for example with a resolution of 4, 0.000 to 1.999 shall be rounded down to 0 and 2.000 to 3.999 shall be rounded up to 4. In each location field the Most Significant Bit (MSB) is the lowest numbered bit in the message which is not a N/S, or E/W flag bit.

The following table illustrates the bit assignments for the degrees portion of the encoded location field.

Latitude Bit assignment	Longitude Bit assignment	Bit value in degrees
N/A	68 (MSB)	128
45 (MSB)	69	64
46	70	32
47	71	16
48	72	8
49	73	4
50	74	2
51	75	1

The following table illustrates the bit assignments for the decimal parts of the degrees portion of the encoded location field. The equivalent in minutes and seconds is also provided. The resolution in meters of each bit of longitude at the equator is also given, with the resolution of each latitude bit 0.169% less than for the longitude bit (at the equator). This reflects the fact that the circumference around the earth at the equator is 40,075.16 Km and 40,008 Km for each meridian.

Latitude bit assignment	Longitude Bit assignment	Bit value of decimal parts in degrees	Bit value of decimal parts in minutes	Bit value of decimal parts in seconds	Resolution in meters (equator)
52	76	0.5	30	1800	55566.67
53	77	0.25	15	900	27783.33
54	78	0.125	7.5	450	13891.67
55	79	0.0625	3.75	225	6945.833
56	80	0.03125	1.875	112.5	3472.917
57	81	0.015625	0.9375	56.25	1736.458
58	82	0.0078125	0.46875	28.125	868.2292
59	83	0.00390625	0.234375	14.0625	434.1146
60	84	0.001953125	0.1171875	7.03125	217.0573
61	85	0.000976563	0.05859375	3.515625	108.5286
62	86	0.000488281	0.029296875	1.7578125	54.26432
63	87	0.000244141	0.014648438	0.87890625	27.13216
64	88	0.00012207	0.007324219	0.439453125	13.56608
65	89	6.10352E-05	0.003662109	0.219726563	6.78304
66	90	3.05176E-05	0.001831055	0.109863281	3.39152

C.5 **Instructions for converting Latitudes and Longitudes to a Binary Number**

Global Navigation Satellite System receivers (e.g., GPS, Glonass, Galileo, BDS, etc.) normally output position data using an IEC 61162-1 (NMEA 0183) formatted sentence. This will provide a position in decimal degrees, minutes and parts of a minute as a decimal fraction in a defined format for example "3546.295, N, 14821.291, W". That is 35 degrees and 46.295 minutes North by 148 degrees and 21.291

minutes West. The size of the decimal fraction is not defined In IEC 61162-1, but in order to ensure adequate accuracy initially and during subsequent rounding processes the number of digits after the decimal point should not be less than 3.

In order to transmit this as a part of a Second Generation Beacon message it is necessary to convert the position into a binary number expressed as degrees and a decimal fraction of a degree. The following text provides an example of how this can be achieved.

Referring to C/S T.018 Table 3.1 Encoded GNSS Location we can see that the required message format is as follows:

No of Bits	Content
1	N/S flag (N=0, S=1)
7	Degrees (0 to 90) in 1 degree increments
15	Decimal parts of a degree (0.5 to 0.00003)
1	E/W flag (E=0. W=1)
8	Degrees (0 to 180) in 1 degree increments
15	Decimal parts of a degree (0.5 to 0.00003)

The use of the bits for the N/S or E/W flags and the encoding of Degrees is straightforward and no further explanation is deemed necessary. But converting minutes and a decimal fraction of a minute to decimal parts of a degree and then to binary is less obvious, so an example of this is provided below.

Using the example above “3546.295, N” in binary would appear as follows:

N/S	Degrees							Decimal Parts of a Degree														
0/1	64	32	16	8	4	2	1	1/2	1/4	1/8	1/16	1/32	1/64	..	..	..	..	..	..	..	..	..
0	0	1	0	0	0	1	1	1	1	0	0	0	1	0	1	1	0	0	0	0	1	1

Initially the minutes and decimal fraction of minutes must be converted into a decimal fraction of a degree, this is achieved by simply dividing the whole number by 60 e.g., 46.295 Minutes divided by 60 equals 0.77158333 Degrees. Again in order to maintain accuracy this number should be rounded to no less than [5] decimal places e.g., [0.77158]. So 35 Degrees and 46.295 Minutes becomes [35.77158] Degrees.

Two procedures for converting decimal parts of a degree to binary are provided below, the first uses the Successive Multiplication Method while the second uses the Integral Number Conversion Method.

<u>Successive Multiplication Method</u>

1) Start with the decimal fraction part and multiply it by 2 (add it to itself)
2) If the result is greater than 1 then the first decimal fraction is a 1, if the result is less than 1 then the first decimal fraction is a 0
3) If the result was greater than 1 then subtract 1 from the result
4) Multiply the remaining number by 2
5) Repeat step 2) to obtain the second decimal fraction and then repeat steps 3) and 4)
6) Repeat step 5) for all the remaining decimal fractions

7) On completing the final step to obtain the fifteenth digit again repeat step 3), if the remainder is 0.5 or greater then increase the computed binary number by one to round to the closest possible number, or if the remaining number is less than 0.5 then use the binary number as computed

The above example is provided below:

1) 0.77158 x 2 = 1.54316 thus the first digit is a 1
2) 1.54316 – 1 = 0.54316
3) 0.54316 x 2 = 1.08632 thus the second digit is a 1
4) 1.08632 – 1 = 0.08632
5) 0.08632 x 2 = 0.1728 thus the third digit is a 0
6) 0.17264 x 2 = 0.34528 thus the fourth digit is a 0
7) 0.34528 x 2 = 0.69056 thus the fifth digit is a 0
8) 0.69056 x 2 = 1.38112 thus the sixth digit is a 1
9) 1.38112 – 1 = 0.38112
10) 0.38112 x 2 = 0.76224 thus the seventh digit is a 0
11) Keep going as above
12) 0.78336 x 2 = 1.56672 thus the fourteenth digit is a 1
13) 1.56672 – 1 = 0.56672
14) 0.56672 x 2 = 1.13344 thus the last digit is a 1
15) 1.13344 – 1 = 0.13344
16) If the remaining number is 0.5 or greater then increase the computed binary number above by one to round to the closest possible number, or if the remaining number is less than 0.5 then use the binary number computed above
17) In this example, one computed 110001011000011, applying the rounding rule above means that in this instance the actual binary number to use is the same

<u>Integral Number Conversion Method</u>

1) Start with the decimal fraction part and multiply it by 2^{15} (2 to the power of 15)
2) Round the result to the nearest whole number
3) Convert this number to binary
4) The result is the binary fraction of the decimal fraction

The above example is provided below:

1) 0.77158 x 2^{15} = 25283.13344
2) Rounding gives us 25283
3) Converting 25283 to binary gives 110001011000011

- END OF APPENDIX C -