[Note: the section numbers are not sequencing properly in Word and are left as-is to indicate outline level.]

1.1 FEC encoder with shortening and puncturing

Shortening encoder consists of 3 steps:

Step 1: Pad zero bits to the payload bits to fit for the codeword size of the mother code, the entire bits are called mother code information bits and the coordinates corresponded to the padded zeros are called shortening coordinates

Step 2: Encode the information bits obtained in Step 1 using mother code encoder

Step 3: Delete the shortening coordinates: i.e. all the padded zero bits in Step 1.

Puncturing encoder consists of 2 steps:

Step 1: Encode the payload (with or without padding) using mother encode.

Step 2: Delete the puncturing coordinates of encoded codeword of Step 1.

1.2 Forward Error Correction encoder for the PHY link channel (PLC)

The downstream PHY-Link channel (PLC) applies a (384,288) puncturing LDPC code for its forward error correction, see Section 1.1 for the definition of puncturing

1.2.1 Mother code

The mother code is a rate 3/5 (480,288) binary LDPC code. A parity check matrix of the mother code is listed in Table XXX, where the sub-matrix size (lifting factor) is L = 48, see XXXX(101.xxx FEC encoding process) for the compact definition of parity check matrix.

Table XXX– (480, 288) LDPC code parity check matrix

16	1	28	9	40	38	16	-	-	-
28	42	36	11	39	9	8	38	-	-
5	2	18	16	25	47	-	2	19	-
18	18	40	18	0	34	-	-	7	32

1.2.2 Puncturing

Denote the information bits sent to the mother code encoder by $(a_0,\cdots,a_{287})$ and let the encoding output be $(a_0,\cdots,a_{287},b_{288},\cdots,b_{479})$, where $b_{288},\cdots,b_{479}$ are parity-check bits.

The puncturing coordinates are (also see Figure XXX)

- Period 1: 48 consecutive coordinates $a_{48},\cdots,a_{95}$
- Period 2: 48 consecutive coordinates $b_{384},\cdots,b_{431}$

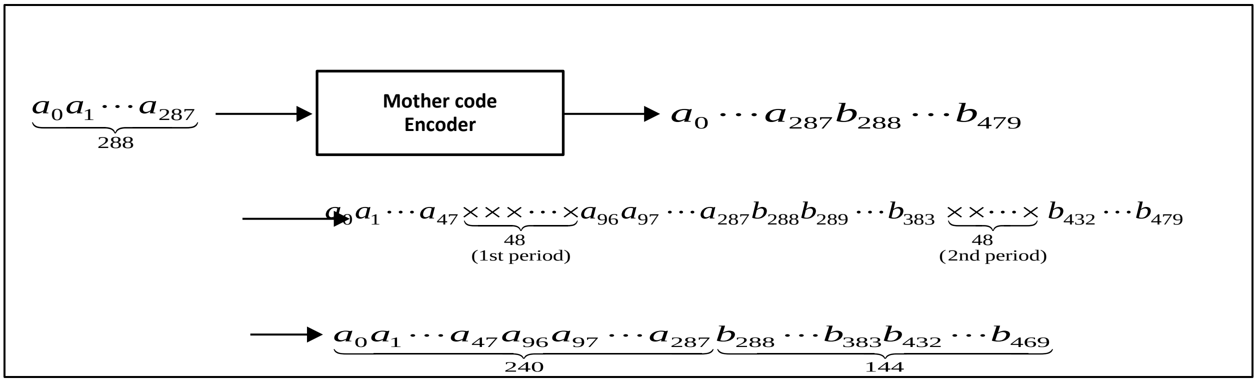

Figure XXX – Punctured LDPC Encoder for PLC FEC

Initial Ranging Signal

The initial ranging signal consists of a preamble sequence and a data part, as illustrated in Figure 7–18.

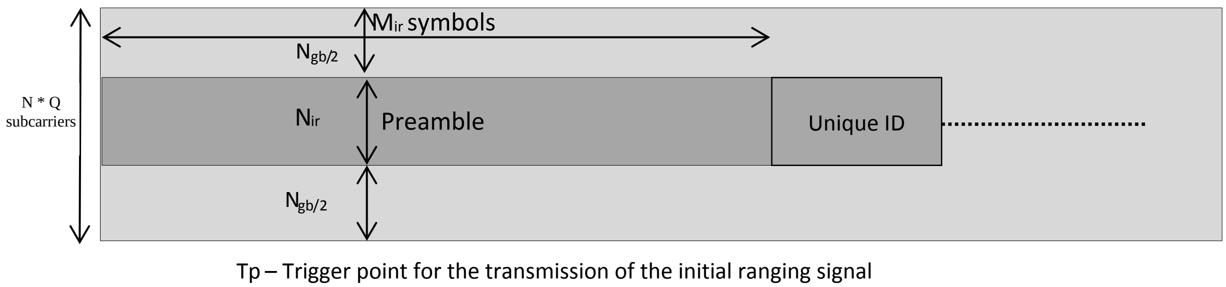

Figure 7–1 - Initial Ranging Signal

[Note to Editors: may want to use fixed known preamble rather than a configurable parameter. Leave as open issue at this time. Response content to be adjusted as per EPoC PLC requirements. Maintain the 24-bit CRC]

The preamble sequence is a BPSK binary sequence configured by the CLT and sent by the CNU. The length of the sequence is configured by the CLT, and the bits contained in the sequence are configured by the CLT.

The data portion of the initial ranging signal is the Initial Ranging Response message. It is composed of a 6-byte MAC address, plus a 1-byte downstream channel ID and 24 CRC bits. It is LDPC (128,80) encoded and randomized as described in the sections below. The preamble sequence and the Initial Ranging Response are duplicated and sent in a special structure of pair of symbols with identical BPSK content as described in Figure 7–19.

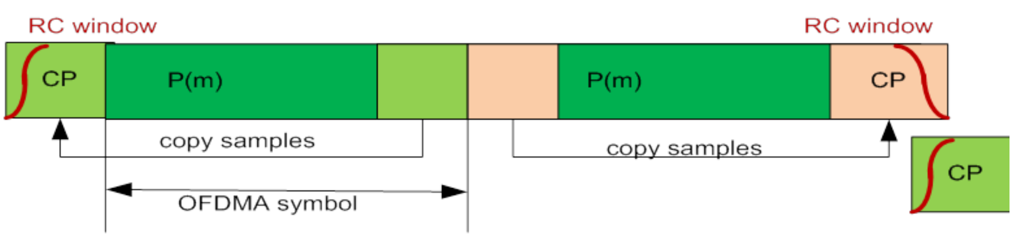

Figure 7–2 - Initial Ranging Admission Slot Structure

A block diagram of the initial ranging signal processing in the transmitter is described in the following figure:

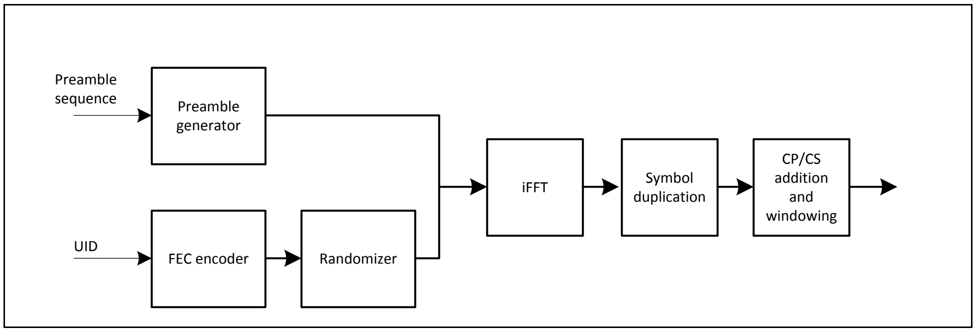

Figure 7–3 - Block Diagram of Initial Ranging Transmitter Processing

1.1.1.1.1 Preamble Construction

[Note: Whether to use a fixed known preamble for EPoC is T.B.D. This section is from DOCSIS 3.1 PHY and requires some changes to the EPoC way of building the preamble.]

The CLT MUST configure the BPSK Preamble sequence and its length Lp, with the limitations described in Section 7.4.15.2.6 and the number of subcarriers, N_{ir}, to be used for the transmission of the initial ranging signal.

The CNU MUST construct the preamble part of the initial ranging signal by converting the preamble sequence bits into BPSK symbols. The preamble is comprised of Mir symbols each with N_{ir} subcarriers.

The CNU MUST convert the first Nir*Mir bits in the preamble sequence into N_{ir}*Mir BPSK symbols in the following order: The first N_{ir} BPSK symbols are written to the N_{ir} subcarriers of the first preamble symbol starting from the lowest subcarrier, the next Nir BPSK symbols to the N_{ir} subcarriers of the second preamble symbol and the last N_{ir} BPSK symbols to the N_{ir} subcarriers of the last (the M_{ir}) preamble symbol.

1.1.1.1.2 Forward Error Correction encoder for Initial Ranging

The CNU MUST encode the 80 bit Initial Ranging Response message using the punctured LDPC (128, 80) encoder as described below, see Section 1.1 for the definition of puncturing.

1.1.1.1.2.1 Mother code

The mother code is a rate ½ (160, 80) binary LDPC code. A parity check matrix of the mother code is listed in Table xxx3, where the sub-matrix size (lifting factor) is L = 16, see Section XXXX(101.xxx FEC encoding process)for the compact definition of parity check matrix.

Table xxx1 - (160, 80) LDPC code Parity Check Matrix

1	11	10	12	7	9	-	-	-	-
2	1	14	15	14	14	12	-	-	-
0	9	3	2	-	-	11	7	-	-
6	8	-	10	3	-	-	10	4	-
12	13	11	-	0	-	-	-	5	2

1.1.1.1.2.2 Puncturing

Denote the information bits sent to the mother code encoder by $(a_0, \cdots, a_{79})$ and let the encoding output be $(a_0, \cdots, a_{79}, b_{80}, \cdots, b_{159})$, where $b_{80}, \cdots, b_{159}$ are parity-check bits. The puncturing coordinates are (also see Figure 7-21)

- Period 1: 16 consecutive bits $a_0, \cdots, a_{15}$
- Period 2: 16 consecutive bits $b_{144}, \cdots, b_{159}$

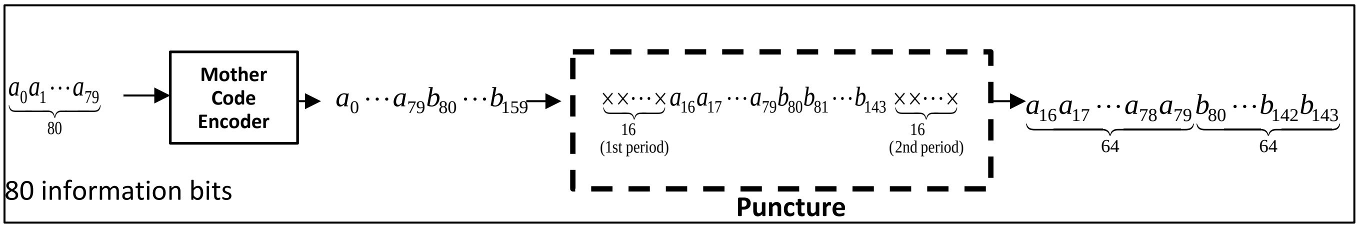

Figure 7-4 – LDPC two-period puncturing encoder for initial ranging FEC

1.1.1.1.3 Padding and Randomizing

The CNU MUST pad and randomize the 128 encoded bits as described below.

The CNU MUST calculate the number of symbols required to transmit the Initial Ranging Response message as follows:

> Nuid_sym = ceiling (128/Nir) where Nir is the number of subcarriers allocated for the Initial Ranging Response message.

CNU MUST pad the remaining bits with ones if the total number of bits (Nbits = Nuid_sym*Nir) is greater than 128.

The CNU MUST randomize the 128 encoded bits and the padding bits as described in Section ***Data Randomization***. The randomized bits are converted to BPSK symbols as defined in (BPSK constellation) and are appended to the preamble sequence for transmission.

The CNU MUST add the BPSK symbols to the data part of the initial ranging signal in the following order: The first Nir BPSK symbols written to the Nir subcarriers of the first symbol of the data part, the next Nir BPSK symbols to the next data symbol, until all BPSK symbols are written vertically symbol by symbol. First BPSK symbol is written to the lowest indexed subcarrier of a data symbol. Unfilled subcarriers in the last symbols are padded with 1s.

1.1.1.1.4 Symbol duplicating cyclic prefix and windowing

Each Initial Ranging OFDMA symbol is repeated twice. A cyclic prefix of N_{cp} samples is appended before the first repeated OFDMA symbol. A cyclic suffix of N_{rcp} ($N_{rcp} = N_{cp} + N_{rp}$) samples is appended after the second repeated OFDMA symbol.

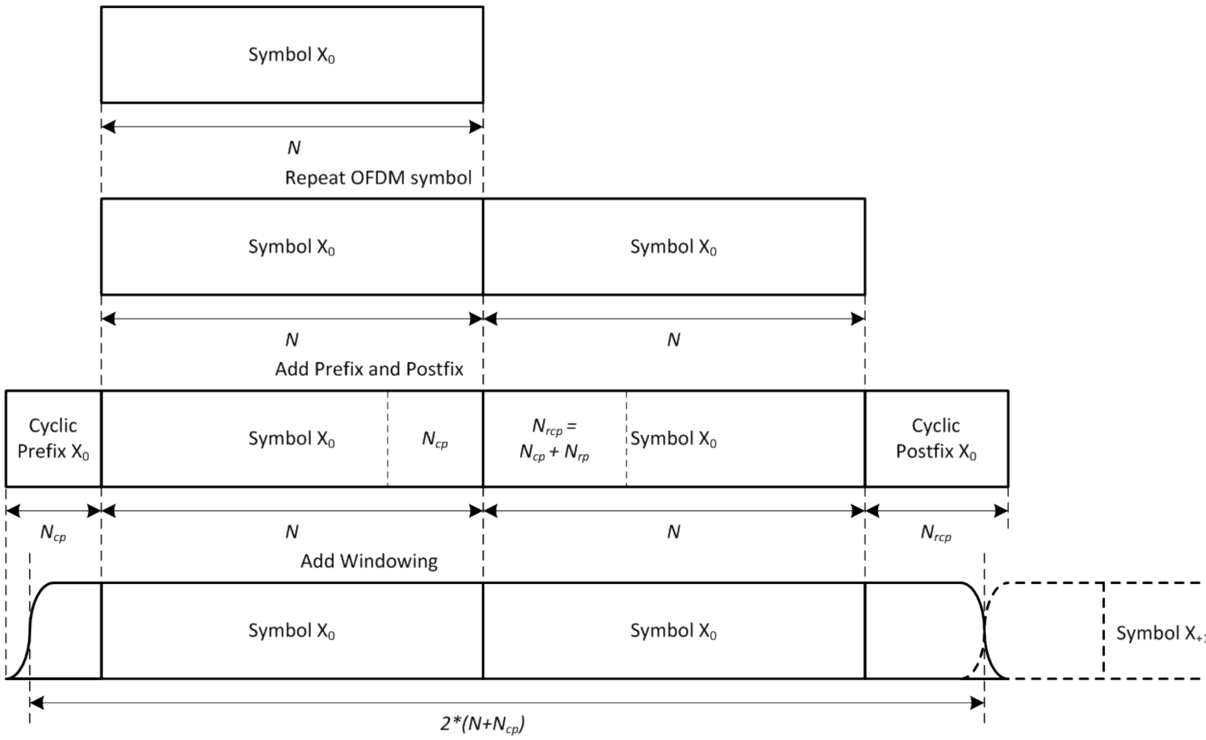

Figure 7–5 - Initial Ranging Symbol Pair Structure

Table 7–2 - Cyclic Prefix and Roll–Off Samples for Initial Ranging

Cyclic Prefix Samples (N_{cp})	Roll-Off Samples (N_{rp})
96	96
128	128
160	160
192	192
224	224
256	224
288	224
320	224
384	224
512	224
640	224

Initial Ranging Response

The Initial Ranging Response message is transmitted only on the upstream PLC channel in response to a PHY Discovery message sent by the CLT. .

MAC Address (6 bytes)	DS-CHAN-ID (1 byte)	CRC-24 (3 bytes)

Figure 6–6 - Initial Ranging Response Format

The parameters of the message transmitted by the CNU MUST be as follows:

MAC Address: MAC address of the CNU. This is a 6-byte field.

Downstream Channel ID: The identifier of the downstream PLC channel on which the CNU is receiving the information that describes this upstream. This is an 8-bit field.

CRC-24: CRC-24 over the MAC Address and DS-CHAN-ID. CRC-24 defined in Section ***24-bit Cyclic redundancy check (CRC) Code***. This is a 3-byte field.

1.1.1.2 Fine Ranging

This section describes Fine ranging operations for the CM transmitter.

Fine ranging is used by the CLT for the second step of the admission of a new CNU process, following successful initial ranging. During this step, a fine ranging signal is transmitted by a new CNU joining the network, according to transmission parameters provided by the CLT. When it receives the fine ranging signal, the CLT is able to fine-tune the joining CM's transmission power and transmission timing.

At the end of the fine ranging step, the CLT can assign transmission opportunities to the new CNU, using optimal transmission power, without interfering with coexisting transmitters on the same OFDMA frame.

1.1.1.2.1 Fine ranging signal

Figure 7–23 illustrates a fine ranging signal.

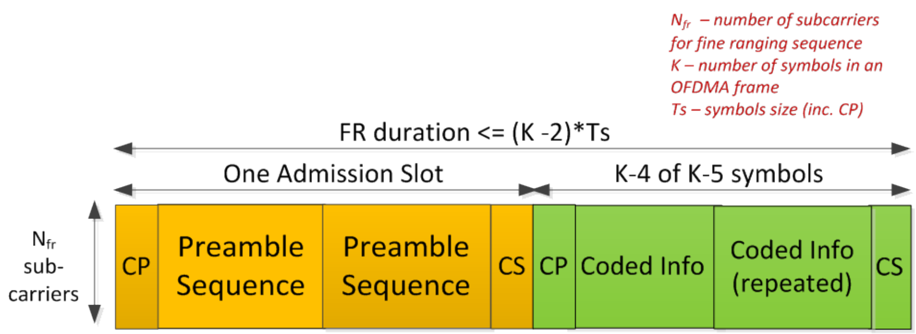

Figure 7–7 - Fine Ranging Signal

Fine ranging is comprised of two parts: a BPSK preamble sequence of one pair of preamble symbols (as defined in Section ***Symbol duplicating cyclic prefix and windowing*** for the initial ranging), and 34 bytes of FEC-encoded data spread over two or more OFDMA symbols. The data part of the fine ranging signal is QPSK-modulated and FEC encoded. The data part has a similar structure to the duplicated pair of symbols (refer to Section 7.4.15.1.2 for the initial ranging data structure).

The CNU MUST transmit the fine ranging signal when allocated to it, with the following configurable parameters:

- Time shift
- TX power
- PRBS sequence.

The CNU MUST use the first portion of the preamble sequence defined for the Initial Ranging signal for the BPSK PRBS sequence of the fine ranging.

1.1.1.2.2 Transmission of The Fine Ranging Signal

The CNU MUST duplicate the OFDMA symbols at the output of the IFFT as described in Section ***Symbol duplicating cyclic prefix and window***, adding a Cyclic Prefix to symbols 2n, and a Cyclic Suffix to symbols 2n+1, for n=0, 1, 2, …

The CNU MUST duplicate the OFDM symbols at the output of the IFFT, add cyclic prefix and suffix and apply windowing

The CNU MUST add the Cyclic Prefix as described in ***Section Cyclic Prefix and Windowing*** , using the same CP value used for all other symbols.

The CNU MUST add a Cyclic Suffix as described in Section ***Cyclic Prefix and Windowing***, and the value of the Cyclic Suffix MUST be equal to the Cyclic Prefix value.

The CNU MUST use the Roll-off value specified in Section ***Cyclic Prefix and Windowing*** ; the Roll-off value MUST be the same as that for all other symbols except Initial Ranging Symbols.

Note: The Roll-off value used for fine ranging may be different from the corresponding value used for Initial Ranging.

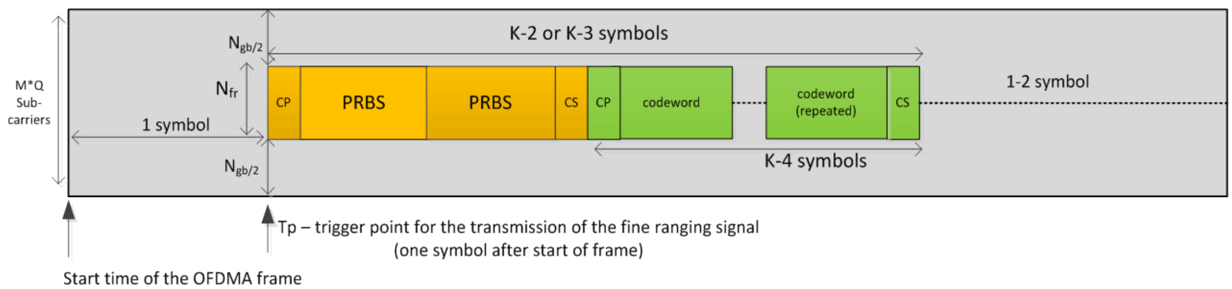

Figure 7–8 - Fine Ranging Signal Transmission

[Note: adjust wording as necessary for the PLC fine ranging and alignment procedure.]

The CNU MUST transmit the fine ranging signal using the upstream PLC channel.

A block diagram of the initial ranging signal processing in the transmitter is described in Figure 7–25.

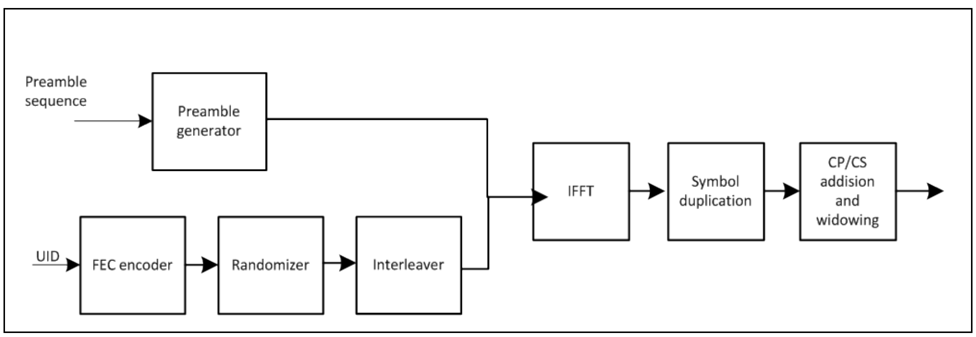

Figure 7–9 - Fine Ranging Transmitter Processing

1.1.1.2.3 Forward Error Correction encoder for Fine Ranging

The upstream finite ranging applies a (362, 272) shortened and punctured LDPC code for its forward error correction, see Section 1.1 for the definition of shortening and puncturing.

1.1.1.2.3,1 Mother code

The mother code is a rate 3/5 (480,288) binary LDPC code. A parity check matrix of the mother code is listed in Table 7–15, where the sub-matrix size (lifting factor) is L = 48, see Section XXXX(101.xxx FEC encoding process)for the compact definition of parity check matrix. .

Table 7–3 - (480, 288) LDPC Code Parity Check Matrix

16	1	28	9	40	38	16	-	-	-
28	42	36	11	39	9	8	38	-	-
5	2	18	16	25	47	-	2	19	-
18	18	40	18	0	34	-	-	7	32

1.1.1.2.3.2 Shortening an puncturing

Let $(a_0, \cdots, a_{271})$ be the data bits for finite ranging. Pad 16 zeros to this 272 bit sequence, i.e. let $a_{272} = a_{273} = \cdots = a_{287} = 0$. Encode $(a_0, \cdots, a_{287})$ with the mother code encoder. Let the output of the encoder be $(a_0, \cdots, a_{287}, b_{288}, \cdots, b_{479})$, where $b_{288}, \cdots, b_{479}$ parity-check bits are. Then delete the 16 shortening coordinates $a_{272}, a_{273}, \cdots, a_{287}$ (also see Figure XXX2).

The puncturing coordinates are (also see Figure XXX2):

- Period 1: 54 consecutive bits $a_0, a_1, \cdots, a_{53}$
- Period 2: 48 consecutive bits $b_{432}, b_{433}, \cdots, b_{479}$

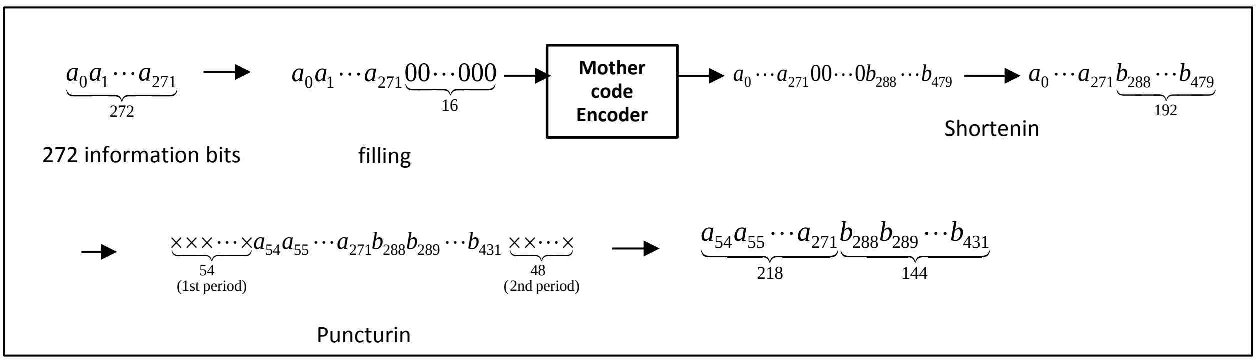

Figure xxx2- Shortening and Puncturing LDPC Encoder for Fine Ranging

1.1.1.2.4 Padding, Randomizing and Interleaving

The CNU MUST calculate the total number of data bits that can be transmitted in the fine ranging signal as follows:

Number_of_allocated_bits = N_{fr}* floor((K-4)/2)*2.

If the number of allocated bits is greater than 362, the CM MUST pad the 362 bits output from the LDPC encoder with ones so that the encoded data and the pad bits equal the Number_of_allocated_bits.

The CNU MUST randomize the data and padding bits as descried in Section ***Data Randomization*** .

The CNU MUST transmit zero valued subcarriers in all symbol times not used for the preamble, data and pad bits.

Note: If K is an even number, the CM transmits K-2 symbols in the fine ranging signal (including the preamble), if K is an odd number, the CM transmits K-3 symbols (including the preamble).

1.1.1.2.5 Power and Time Adjustments

[NOTE: Content of this section would need to be made consistent with EPoC PLC upstream channel.]

Algorithms for power and time adjustments (such as number of fine ranging trials, frequency allocations, etc.) are vendor-specific implementation.

1.2 Constellation structure and mapping for FEC

After LDPC encoding for downstream and upstream transmissions, PLC, initial ranging and fine ranging, the output bits stream of the encoder must be mapped to QAM subcarriers such that first bit is the least-significant bit of the first QAM subcarrier constellation m-tuple, see Figure XX-1

$\longrightarrow$ Time

$encodered\ bit\ stream\ x_0\ x_1\ \ldots\ x_{m-1}\ x_m\ x_{m+1} \cdots\ x_{2m-1}\ x_{2m} \cdots$

	MSB			LSB
QAM m-tuple 1:	x_{m-1}	x_{m-2}	$\cdots$	x_0
QAM m-tuple 2:	x_{2m-1}	x_{2m-2}	$\cdots$	x_m
QAM m-tuple 3:	x_{3m-1}	x_{3m-2}	$\cdots$	x_{2m}

Figure XX-1 – Bitstream to QAM m-tuple mapping

The m-tuples must be modulated onto subcarriers using QAM constellation. As described in the flowing sub-sections, the QAM constellation structure and mappings are defined inductively and use Gray mapping as their base.

1.2.1 One dimensional Gray mapping for m-tuple binary bits

1) When m=1, the Gray mapping is define to be $Gray_1(0)=1$ and $Gray_1(1)=-1$
2) When m>1, the Gray mapping is defined inductively, i.e.

$$Gray_m(x_{m-1}x_{m-2}\cdots x_0)=(1-2x_0)\left(2^{m-1}+Gray_{m-1}(x_{m-1}\cdots x_0)\right)$$

1.1.2. Constellation structure and mapping of BPSK

Let m=1 and a binary bit is x. The BPSK mapping is

$$(I_1(x),Q_1(x))=\left(Gray_1(x),0\right)=\begin{cases}(1,0) & x=0\\(-1,0) & x=1\end{cases}$$

Also see Figure XXX.1

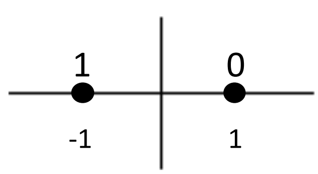

Figure XXX-1 BPSK

1.1.3 Constellation structure and mapping of 2^{2n} –QAM

Let m=2n and the m-tuple binary bits are $x_0,\cdots,x_{n-1},x_n,\cdots,x_{2n-1}$. The mapping from that m-tuple to a 2^m –QAM is defined by

$$(I_{2n}(x_{2n-1},\cdots,x_n,x_{n-1},\cdots,x_0),Q_{2n}(x_{2n-1},\cdots,x_n,x_{n-1},\cdots,x_0))=\left(Gray_n((x_{n-1},\cdots,x_0),Gray_n((x_{2n-1},\cdots,x_{2n}))\right)$$

where the Gray mapping is defined in 1.1.1. Some of the examples are given the following

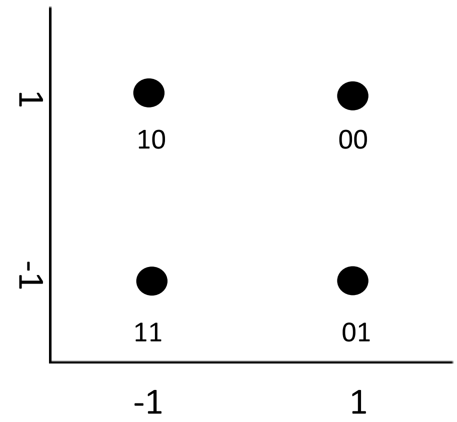

Figure XXX-2 QPSK

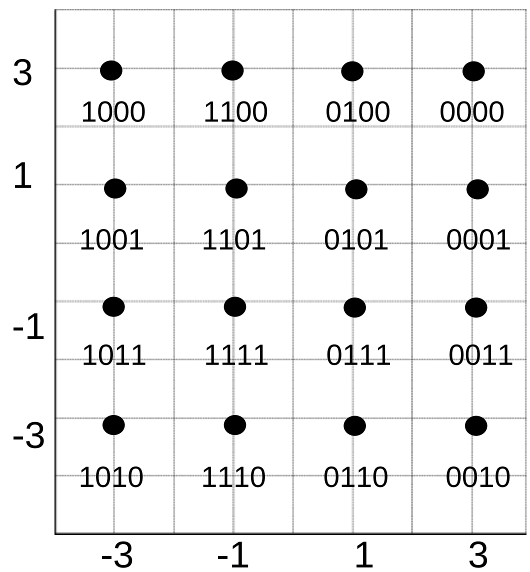

Figure XXX-3 16-QAM

1.1.4 Constellation structure and mapping of 2^{2n+1} –QAM (n>0)

Let m=2n+1 and the m-tuple binary bits are $x_0,\cdots,x_n,x_{n+1},\cdots,x_{2n}$. Firstly, map this m-tuple to a rectangular constellation defined by

$$(I_{rct}(x_{2n},\cdots,x_n,x_{n-1},\cdots,x_0),Q_{rct}(x_{2n},\cdots,x_n,x_{n-1},\cdots,x_0))=\left(Gray_{n+1}(x_{2n}x_{2n-1}\cdots x_n),Gray_n(x_{n-1}x_{n-2}\cdots,x_0)\right)$$

where the Gray mapping is defined in 1.1.1. Then the structures and mappings of cross-constellations are generated in the following sub-sections.

1.1.4.1 Constellation structure and mapping of 8-QAM

Let the constellation signal and its mapping be denoted by $(I_3(x_2x_1x_0),Q_3(x_2x_1x_0))$, then

$$\begin{cases} \begin{cases} I_3(x_2x_1x_0) = I_{rct}(x_2x_1x_0) + 1 \\ Q_3(x_2x_1x_0) = Q_{rct}(x_2x_1x_0) \end{cases} & I_{rct}(x_2x_1x_0) < 3 \\ \begin{cases} I_3(x_2x_1x_0) = 3 - I_{rct}(x_2x_1x_0) \\ Q_3(x_2x_1x_0) = sign(Q_{rct}(x_2x_1x_0))(|Q_{rct}(x_2x_1x_0)| + 2) \end{cases} & \text{otherwise} \end{cases}$$

where the sign function is defined by $sign(a) = \begin{cases} 1 & \text{if } a \geq 0 \\ -1 & \text{if } a < 0 \end{cases}$.

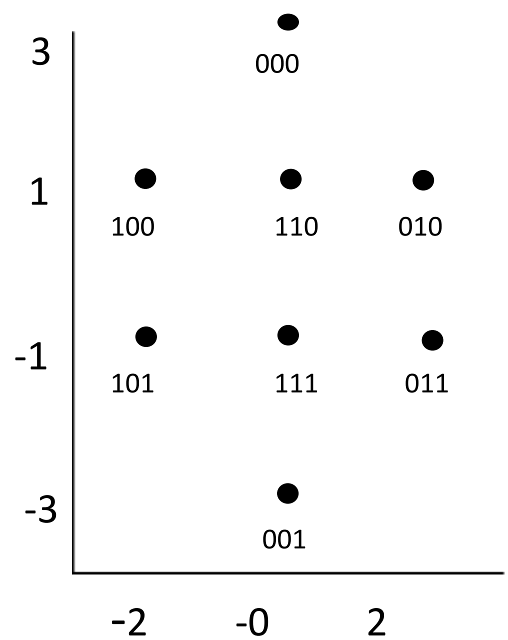

Figure XXX-4 8-QAM

1.1.3.1 Constellation structure and mapping of 2^{2n+1}-QAM with n>1

Let the mapping be denoted by $(I_{2n+1}(x_{2n}x_{2n-1}\cdots x_0), Q_{2n+1}(x_{2n}x_{2n-1}\cdots x_0))$ and let $s = 2^{n-1}$. Then,

when $|I_{rct}(x_{2n}x_{2n-1}\cdots x_0)| < 3s$, $\begin{cases} I_{2n+1}(x_{2n}x_{2n-1}\cdots x_0) = I_{rct}(x_{2n}x_{2n-1}\cdots x_0) \\ Q_{2n+1}(x_{2n}x_{2n-1}\cdots x_0) = Q_{rct}(x_{2n}x_{2n-1}\cdots x_0) \end{cases}$;

when $|I_{rct}(x_{2n}x_{2n-1}\cdots x_0)| \geq 3s$

$$\begin{cases} \begin{cases} I_{2n+1}(x_{2n}x_{2n-1}\cdots x_0) = sign(I_{rec}(x_{2n}x_{2n-1}\cdots x_0))(|I_{rct}(x_{2n}x_{2n-1}\cdots x_0)| - 2s) \\ Q_{2n+1}(x_{2n}x_{2n-1}\cdots x_0) = sign(Q_{rct}(x_{2n}x_{2n-1}\cdots x_0))(4s - |Q_{rct}(x_{2n}x_{2n-1}\cdots x_0)|) \end{cases} & \text{if } |\mathrm{Q}_{\mathrm{rct}}(x_{2n}x_{2n-1}\cdots x_0)| > s \\ \begin{cases} I_{2n+1}(x_{2n}x_{2n-1}\cdots x_0) = sign(I_{rct}(x_{2n}x_{2n-1}\cdots x_0))(4s - |I_{rct}(x_{2n}x_{2n-1}\cdots x_0)|) \\ Q_{2n+1}(x_{2n}x_{2n-1}\cdots x_0) = sign(Q_{rct}(x_{2n}x_{2n-1}\cdots x_0))(|Q_{rct}(x_{2n}x_{2n-1}\cdots x_0)| + 2s) \end{cases} & \text{if } |\mathrm{Q}_{\mathrm{rct}}(x_{2n}x_{2n-1}\cdots x_0)| \leq s \end{cases}$$

The following figure presents the 32-QAM structure and mapping.

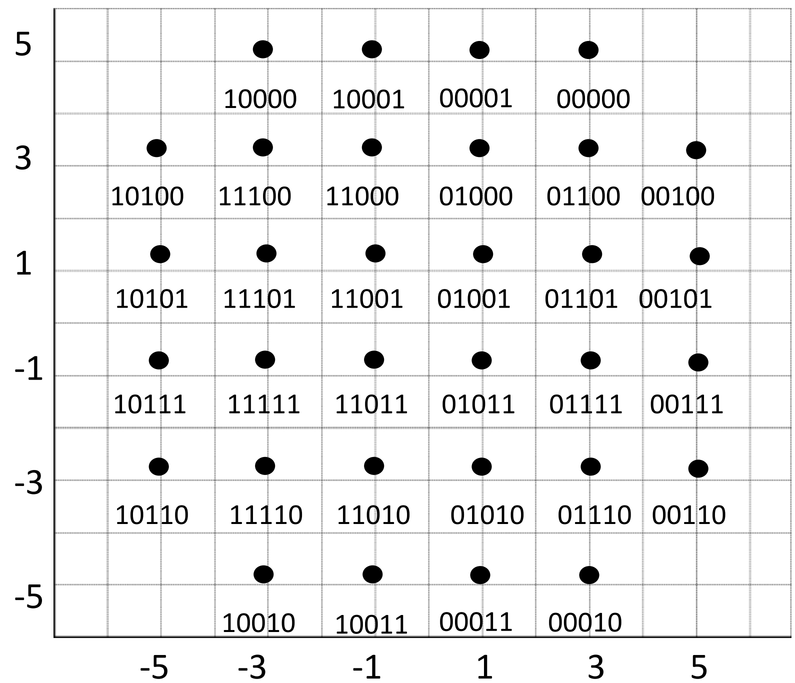

Figure XXX-5 32-QAM

1.2 QAM Constellation Scaling

Both real and imaginary axes of a QAM constellation shall be scaled. The scaling factors given in column 3 of the table below ensure that the mean square value of all QAM constellations are equal to 1.0.

Table XX–1 - QAM Constellation Scaling Factors

QAM Constellation	m Number of bits	Scaling Factor
BPSK	1	1
QPSK	2	$1/\sqrt{2}$
8-QAM	3	$1/\sqrt{5}$
16-QAM	4	$1/\sqrt{10}$
32-QAM	5	$1/\sqrt{20}$
64-QAM	6	$1/\sqrt{42}$
128-QAM	7	$1/\sqrt{82}$
256-QAM	8	$1/\sqrt{170}$
512-QAM	9	$1/\sqrt{330}$
1024-QAM	10	$1/\sqrt{682}$
2048-QAM	11	$1/\sqrt{1322}$
4096-QAM	12	$1/\sqrt{2730}$
8192-QAM	13	$1/\sqrt{5290}$

16384-QAM	14	$1/\sqrt{10922}$

[Note: the following sections are placed here for completeness of this document. They may already be present in other sections of the draft.]

PLC Data Randomization

[NOTE: question of whether EPoC can use a fixed seed or whether it is necessary to have this programmable.]

The CNU MUST implement a randomizer in the upstream modulator shown in Figure **7–4** where the 23-bit seed value is programmable.

At the beginning of each burst, the register is cleared and the seed value is loaded. The CNU MUST use the seed value to calculate the scrambler bit which is combined in an XOR with the first bit of data of each burst.

The CNU MUST configure the randomizer seed value in ~~response to the Upstream Channel Descriptor~~ provided by the CLT.

The CNU MUST use x^23+x^18+1 for the data randomizer polynomial.

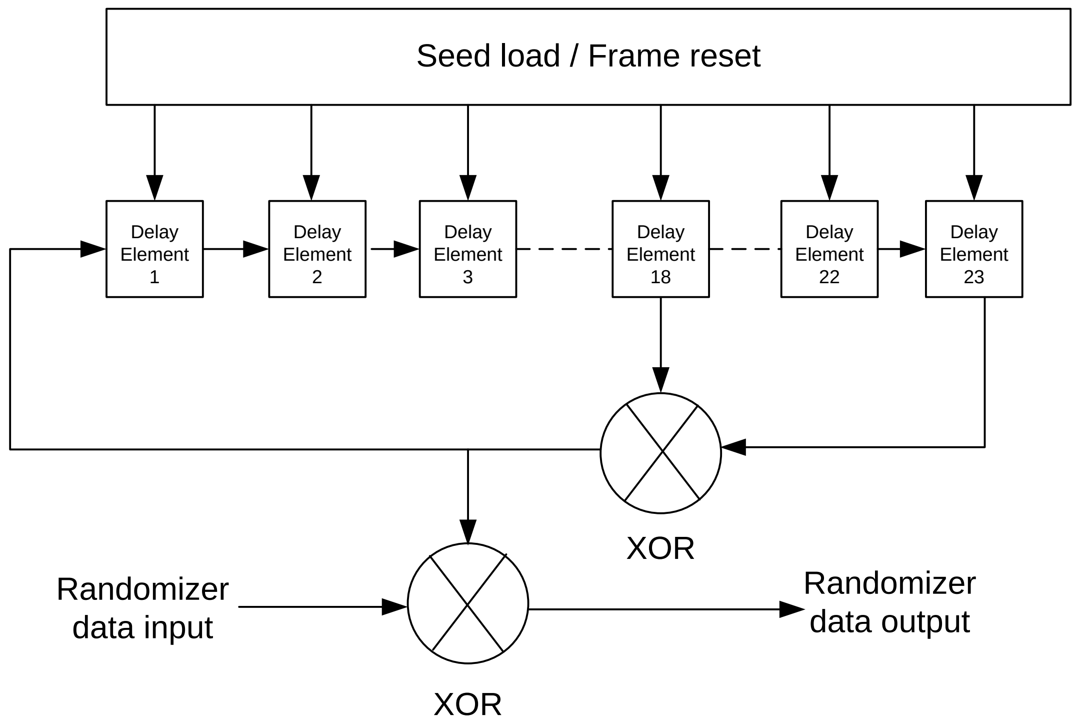

Figure 7–10 - Upstream Data Randomizer

24-bit Cyclic redundancy check (CRC) Code

[Note: this section may be duplicative of other baseline content. It is here for reference/completeness.]

This section contains a 24-bits CRC code encoding, which is used for Initial Ranging..

The CRC encoder generates the 24 bits parity bits denoted by $p_0, p_1, p_2, p_3, ..., p_{23}$ for the input bit stream $b_0, b_1, ..., b_{k-1}$ .using the following generator polynomial:

$$g_{CRC24}(x) = x^{24} + x^{22} + x^{20} + x^{19} + x^{18} + x^{16} + x^{14} + x^{13} + x^{11} + x^{10} + x^{8} + x^{7} + x^{6} + x^{3} + x + 1$$

(127266713 in octal representation) [1], which means in GF(2) the following equation holds

$$b_0 x^{k+23} + b_1 x^{k+22} + ... + b_{k-1} x^{24} + p_0 x^{23} + p_1 x^{22} + ... + p_{22} x^{1} + p_{23} = 0 \bmod g_{CRC24}(x) .$$

This 24-bit CRC polynomial is optimized by G. Castangnoli, S. Bräuer and M. Hermann