

## 物理信道映射

### 上行

### 下行

### 通用流程

来自 MAC 层/向 MAC 层输出的数据和控制流经过编/解码，通过无线传输链路提供传输和控制服务。信道编码方案是错误检测、错误纠正、速率匹配、交织以及传输信道或控制信息向物理信道映射/从物理信道到传输信道控制信息解析或分离的组合方案。

### CRC 计算

CRC 计算单元的输入比特为  $a_0, a_1, a_2, a_3, \dots, a_{A-1}$ ，奇偶校验比特为  $p_0, p_1, p_2, p_3, \dots, p_{L-1}$ 。AA 是输入序列的长度，LL 是校验比特数目。校验比特由下列循环生成多项式之一产生：

- $g_{\text{CRC24A}}(D) = [D^{24} + D^{23} + D^{18} + D^{17} + D^{14} + D^{11} + D^{10} + D^7 + D^6 + D^5 + D^4 + D^3 + D + 1]$   
 $g_{\text{CRC24A}}(D) = [D^{24} + D^{23} + D^{18} + D^{17} + D^{14} + D^{11} + D^{10} + D^7 + D^6 + D^5 + D^4 + D^3 + D + 1]$ ，CRC 长度  $L = 24$ ；
- $g_{\text{CRC24B}}(D) = [D^{24} + D^{23} + D^6 + D^5 + D + 1]$   
 $g_{\text{CRC24B}}(D) = [D^{24} + D^{23} + D^6 + D^5 + D + 1]$ ，CRC 长度  $L = 24$ ；
- $g_{\text{CRC24C}}(D) = [D^{24} + D^{23} + D^{21} + D^{20} + D^{17} + D^{15} + D^{13} + D^{12} + D^8 + D^4 + D^2 + D + 1]$   
 $g_{\text{CRC24C}}(D) = [D^{24} + D^{23} + D^{21} + D^{20} + D^{17} + D^{15} + D^{13} + D^{12} + D^8 + D^4 + D^2 + D + 1]$ ，CRC 长度  $L = 24$ ；

- $g_{CRC16}(D) = [D^{16} + D^{12} + D^5 + D + 1]$  , CRC 长度  $L=16$

编码以系统方式进行，这意味着在二元域  $GF(2)$  中，多项式：

$$a_0D^{A+L-1} + a_1D^{A+L-2} + \dots + a_{A-1}D^L + p_0D^{L-1} + p_1D^{L-2} + \dots + p_{L-2}D^1 + p_{L-1}D^0$$

除以相应的 CRC 生成多项式时，余数等于 0。

添加 CRC 之后的比特序列表示为  $b_0, b_1, b_2, b_3, \dots, b_{B-1}$ ，其中  $b_0, b_1, b_2, b_3, \dots, b_{B-1}$ 。  $a_k$  和  $b_k$  的关系如下：

$$b_k = a_k, \text{ 对于 } k=0, 1, 2, \dots, A-1$$

$$b_k = p_{k-A}, \text{ 对于 } k=A, A+1, A+2, \dots, A+L-1$$

## 码块分段和 CRC 添加

### LDPC 码

码块分段单元的输入序列为  $b_0, b_1, b_2, b_3, \dots, b_{B-1}$ ，其中  $B > 0$ 。如果  $B$  大于最大码块大小  $K_{cb}$ ，则输入序列要进行分段操作，并且每个分段后的码块要添加一个  $L=24$  的 CRC 序列。最大码块大小为：

- $K_{cb} = 8448$ 。

码块总数  $C$  根据以下方法计算得到：

```

if  $B \leq K_{cb}$ 
     $L = 0$ 
    Number of code blocks:  $C = 1$ 
     $B' = B$ 
else
     $L = 24$ 
    Number of code blocks:  $C = \lceil B / (K_{cb} - L) \rceil$ 
     $B' = B + C \cdot L$ 
end if
    
```

当  $C \neq 0$  时，码块分段的输出比特为  $Cr_0, Cr_1, Cr_2, Cr_3, \dots, Cr_{(Kr-1)}$ ，其中  $0 \leq r < C$  为码块号， $Kr$  是码块  $r$  中的比特数。

每个码块中的比特数为（仅适用于  $C \neq 0$  的情况）：

```

 $K_+ = \lceil B' / C \rceil;$ 
 $K_- = \lfloor B' / C \rfloor;$ 
For LDPC base graph 1,
     $K_b = 22;$ 
For LDPC base graph 2,
    If  $B > 640$ 
         $K_b = 10;$ 
    elseif  $B > 560$ 
         $K_b = 9;$ 
    elseif  $B > 192$ 
         $K_b = 8;$ 
    else
         $K_b = 6;$ 
    end

```

注：蓝色高亮部分是华为批注： $K_+$  and  $K_-$  are kept for now. May update after TBS is finalized.

在 Table 5.3.2-1 中所有列举的集合中找到  $Z$  的最小值，表示为  $Z_c$ 。  $K_b \cdot Z_c \geq K_+ K_b$

·  $Z_c \geq K_+$  且对于 LDPC BG#1 有  $K = 22Z_c K = 22Z_c$ ，对于 LDPC BG#2 有

$$K = 10Z_c K = 10Z_c$$

```

s = 0;
for r = 0 to C-1
    if r < mod(B, C)
        K' = K+;
    else
        K' = K-;
    end
    k = 0;
    while k < K' - L
        crk = bi;
        k = k + 1;
        s = s + 1;
    end while
    if C > 1
        The sequence cr0, cr1, cr2, cr3, ..., cr(K'-L-1) is used to calculate
        the CRC parity bits pr0, pr1, pr2, ..., pr(L-1) according to section
        5.1.1 with the generator polynomial gCRC24B(D).
        while k < K'
            crk = pr(k+L-K');
            k = k + 1;
        end while
    end if
    while k < K          -- Insertion of filler bits
        crk = < NULL >;
        k = k + 1;
    end while
end for

```

## 信道编码

### Polar coding

对于给定的码块，信道编码器的输入序列为  $c_0, c_1, c_2, c_3, \dots, c_{K-1}$ ，

其中  $K$  是编码器的输入比特数。编码后的比特序列表示为  $d_0, d_1, d_2, \dots, d_{N-1}$ ，

其中  $N = 2^n$  并且  $n$  的值由下列方法决定：

$E$  表示速率匹配输出序列长度，由 2.4.1 节给出；

If  $E < (9/8) \cdot 2^{\lceil \log_2 E \rceil - 1}$  and  $K/E < 9/16$

$$n_1 = \lceil \log_2 E \rceil - 1;$$

else

$$n_1 = \lceil \log_2 E \rceil;$$

end if

$$R_{\min} = 1/8;$$

$$n_2 = \lceil \log_2 (K / R_{\min}) \rceil;$$

$$n = \min \{n_1, n_2, n_{\max}\}.$$

比特序列  $c_0, c_1, c_2, c_3, \dots, c_{K-1}$  采用下列方法被交织成比特序列  $C$

$'0, C'1, C'2, C'3, \dots, C'K-1, c_0, c_1, c_2, c_3, \dots, c_{K-1}'$ ;

$$C'_k = C_{\Pi(k)} c'_k = c_{\Pi(k)}, \quad k=0, 1, \dots, K-1$$

其中交织 pattern  $\Pi(k)$  由下列方法确定：

```

if  $I_{IL} = 0$ 
     $\Pi(k) = k, k = 0, 1, \dots, K - 1$ 
else
     $k = 0$ ;
    for  $m = 0$  to  $K_{IL}^{max} - 1$ 
        if  $\Pi_{IL}^{max}(m) \geq K_{IL}^{max} - K$ 
             $\Pi(k) = \Pi_{IL}^{max}(m) - (K_{IL}^{max} - K)$ ;
             $k = k + 1$ ;
        end if
    end for
end if

```

其中  $\Pi_{maxIL}(m)$  由 Table 5.3.1-1 和  $K_{maxIL}$  给定。

华为批注：The value of  $K_{maxIL}$  equals to  $K_{max} = \max(140, \max \text{ DCI payload size in Rel-15} + 20) + 24$ . The value of  $K_{maxIL}$  should be the number of elements in Table 5.3.1-1, in the end. According to the current Table 5.3.1-1,  $K_{maxIL} = 224$ .

Table 5.3.1-1: Interleaving pattern  $\Pi_{\text{IE}}^{\text{max}}(m)$

$m$	$\Pi_{\text{IE}}^{\text{max}}(m)$	$m$	$\Pi_{\text{IE}}^{\text{max}}(m)$	$m$	$\Pi_{\text{IE}}^{\text{max}}(m)$	$m$	$\Pi_{\text{IE}}^{\text{max}}(m)$	$m$	$\Pi_{\text{IE}}^{\text{max}}(m)$	$m$	$\Pi_{\text{IE}}^{\text{max}}(m)$	$m$	$\Pi_{\text{IE}}^{\text{max}}(m)$	$m$	$\Pi_{\text{IE}}^{\text{max}}(m)$
0	0	28	50	56	119	84	173	112	45	140	152	168	90	196	197
1	2	29	54	57	121	85	175	113	48	141	154	169	93	197	203
2	3	30	55	58	122	86	178	114	51	142	156	170	96	198	73
3	5	31	57	59	125	87	179	115	56	143	159	171	104	199	78
4	6	32	59	60	126	88	180	116	58	144	162	172	107	200	98
5	8	33	60	61	127	89	182	117	61	145	165	173	124	201	204
6	11	34	62	62	129	90	183	118	63	146	167	174	134	202	99
7	12	35	64	63	130	91	186	119	65	147	169	175	139	203	205
8	13	36	67	64	131	92	187	120	68	148	172	176	145	204	100
9	16	37	69	65	132	93	189	121	70	149	174	177	157	205	206
10	19	38	74	66	136	94	192	122	75	150	176	178	160	206	101
11	20	39	79	67	137	95	194	123	81	151	181	179	163	207	207
12	22	40	80	68	141	96	196	124	87	152	184	180	177	208	208
13	24	41	84	69	142	97	199	125	89	153	188	181	185	209	209
14	28	42	85	70	143	98	200	126	92	154	190	182	191	210	210
15	32	43	86	71	147	99	1	127	95	155	193	183	196	211	211
16	33	44	88	72	148	100	4	128	103	156	195	184	202	212	212
17	35	45	91	73	149	101	7	129	106	157	201	185	27	213	213
18	37	46	94	74	151	102	9	130	112	158	10	186	31	214	214
19	38	47	102	75	153	103	14	131	115	159	15	187	53	215	215
20	39	48	105	76	155	104	17	132	117	160	18	188	72	216	216
21	40	49	109	77	158	105	21	133	120	161	26	189	77	217	217
22	41	50	110	78	161	106	23	134	123	162	30	190	83	218	218
23	42	51	111	79	164	107	25	135	128	163	52	191	97	219	219
24	44	52	113	80	166	108	29	136	133	164	66	192	108	220	220
25	46	53	114	81	168	109	34	137	138	165	71	193	135	221	221
26	47	54	116	82	170	110	36	138	144	166	76	194	140	222	222
27	49	55	118	83	171	111	43	139	150	167	82	195	146	223	223

华为批注：Contents of this table will be updated. Current working

assumption is the pattern for nFAR=21 in R1-1712167.

Polar 序列  $Q_{N_{\text{max}} - 10} = \{Q_{N_{\text{max}}0}, Q_{N_{\text{max}}1}, \dots, Q_{N_{\text{max}}N_{\text{max}} - 1}\}$   $Q_{0N_{\text{max}} - 1} = \{Q_{0N_{\text{max}}}, Q_{1N_{\text{max}}}, \dots, Q_{N_{\text{max}} - 1N_{\text{max}}}\}$  由 Table 5.3.1-2 给定，其中  $0 \leq Q_{N_{\text{max}}i} \leq N_{\text{max}} - 10 \leq Q_{iN_{\text{max}}} \leq N_{\text{max}} - 1$  表示 Polar 编码前的一个比特索引，其中



$i=0,1,\dots,N-1$   $i=0,1,\dots,N-1$ ,  $N_{\max}=1024$   $N_{\max}=1024$ 。Polar 序列  $Q_{N_{\max}-10Q_{N_{\max}}-1}$  按可靠性升序排列  $W(Q_{N_{\max}0}) < W(Q_{N_{\max}1}) < \dots < W(Q_{N_{\max}N_{\max}-1})$   $W(Q_{0N_{\max}}) < W(Q_{1N_{\max}}) < \dots < W(Q_{N_{\max}-1N_{\max}})$ , 其中  $W(Q_{N_{\max}i})$   $W(Q_{iN_{\max}})$  表示比特索引  $Q_{N_{\max}i}$   $Q_{iN_{\max}}$  的可靠性。

对任意码块编码为  $N$  比特, 都使用相同的 Polar 序列  $Q_{N-10Q_{N-1}} = \{Q_{N0}, Q_{N1}, Q_{N2}, \dots, Q_{N(N-1)}\}$   $Q_{0N-1} = \{Q_{0N}, Q_{1N}, Q_{2N}, \dots, Q_{(N-1)N}\}$ 。Polar 序列  $Q_{N-10Q_{N-1}}$  是 Polar 序列  $Q_{N_{\max}-10Q_{N_{\max}-1}}$  的子集, 即值小于  $N$  的所有  $Q_{N_{\max}i}$   $Q_{iN_{\max}}$  元素, 并且按可靠性升序排列  $W(Q_{N0}) < W(Q_{N1}) < W(Q_{N2}) < \dots < W(Q_{N(N-1)})$   $W(Q_{0N}) < W(Q_{1N}) < W(Q_{2N}) < \dots < W(Q_{(N-1)N})$ 。

$Q_{NI}^-$   $Q_{IN}^-$  是 Polar 序列  $Q_{N-10Q_{N-1}}$  中的一组比特索引,  $Q_{NF}^-$   $Q_{FN}^-$  是 Polar 序列  $Q_{N-10Q_{N-1}}$  中其他比特的索引, 其中  $Q_{NI}^-$   $Q_{IN}^-$  和  $Q_{NF}^-$   $Q_{FN}^-$  由 2.4.1.1 节给定,  $||Q_{NI}^-|| = K + n_{PC}$   $||Q_{IN}^-|| = K + n_{PC}$ ,  $||Q_{NF}^-|| = N - ||Q_{NI}^-||$   $||Q_{FN}^-|| = N - ||Q_{IN}^-||$ ,  $n_{PC}$   $n_{PC}$  是奇偶校验比特数。

$G_N = (G_2)^{\otimes n}$   $G_N = (G_2)^{\otimes n}$  表示矩阵  $G_2$   $G_2$  的  $n$  次克罗内克积, 其中  $G_2 = \begin{bmatrix} 1 & 1 \\ 0 & 1 \end{bmatrix}$   $G_2 = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$ 。

$g_{jj}$  表示矩阵  $G_N$  的第  $j$  行 比特索引  $j=0,1,\dots,N-1$   $j=0,1,\dots,N-1$ 。  $w(g_j)$   $w(g_j)$  表示  $g_{jj}$  的行重, 即  $g_{jj}$  中 1 的个数。奇偶校验比特的索引集合表示为  $Q_{NPC}$   $Q_{PCN}$ , 其中  $||Q_{NPC}|| = n_{PC}$   $||Q_{PCN}|| = n_{PC}$ 。  $(n_{PC} - n_{w_{mPC}})$   $(n_{PC} - n_{PC_{wm}})$  个奇偶校验比特位于  $Q_{NI}^-$   $Q_{IN}^-$  中的  $(n_{PC} - n_{w_{mPC}})$   $(n_{PC} - n_{PC_{wm}})$  个最不可靠的比特索引。其他  $n_{w_{mPC}}$   $n_{PC_{wm}}$  个奇偶校验比特位于  $Q_{NI}^-$   $Q_{IN}^-$  中行重最小的比特索引, 其中

$Q_{\sim NI}Q_{\sim IN}$  表示  $Q_{\sim NI}Q_{\sim IN}$  中  $(||Q_{\sim NI}|| - n_{PC})(|Q_{\sim IN}| - n_{PC})$  个最可靠的比特索引；如果  $Q_{\sim NI}Q_{\sim IN}$  中具有相同最小行重的比特索引大于  $n_{wmPCnPCwm}$ ，则其他  $n_{wmPCnPCwm}$  个奇偶校验比特位于  $Q_{\sim NI}Q_{\sim IN}$  中可靠性最高且行重最小的  $n_{wmPCnPCwm}$  个比特索引。

根据下列方法生成序列  $u = [u_0 \ u_1 \ u_2 \ \dots \ u_{N-1}]$   $u = [u_0 \ u_1 \ u_2 \ \dots \ u_{N-1}]$ ：

```

 $k = 0;$ 
if  $n_{PC} > 0$ 
     $y_0 = 0; y_1 = 0; y_2 = 0; y_3 = 0; y_4 = 0;$ 
    for  $n = 0$  to  $N - 1$ 
         $y_i = y_0; y_0 = y_1; y_1 = y_2; y_2 = y_3; y_3 = y_4; y_4 = y_i;$ 
        if  $n \in \overline{\mathbf{Q}}_I^N$ 
            if  $n \in \mathbf{Q}_{PC}^N$ 
                 $u_n = y_0;$ 
            else
                 $u_n = c_k';$ 
                 $k = k + 1;$ 
                 $y_0 = y_0 \oplus u_n;$ 
            end if
        else
             $u_n = TBD;$ 
        end if
    end for
else
    for  $n = 0$  to  $N - 1$ 
        if  $n \in \overline{\mathbf{Q}}_I^N$ 
             $u_n = c_k';$ 
             $k = k + 1;$ 
        else
             $u_n = TBD;$ 
        end if
    end for
end if

```

注：粉色部分为华为批注：Need further agreements on how to set the values for frozen bits.

通过  $d = uGnd = uGn$  编码器得到输出序列，表示为  $d = [d_0 \ d_1 \ d_2 \ \dots \ d_{N-1}]$ 。编码在二元域  $GF(2)$  进行。

Table 5.3.1-2: Polar sequence  $Q_0^{N_{max}-1}$  and its corresponding reliability  $W(Q_i^{N_{max}})$

$W(Q_i^{N_{max}})$	$Q_i^{N_{max}}$	$W(Q_i^{N_{max}})$	$Q_i^{N_{max}}$	$W(Q_i^{N_{max}})$	$Q_i^{N_{max}}$	$W(Q_i^{N_{max}})$	$Q_i^{N_{max}}$	$W(Q_i^{N_{max}})$	$Q_i^{N_{max}}$	$W(Q_i^{N_{max}})$	$Q_i^{N_{max}}$	$W(Q_i^{N_{max}})$	$Q_i^{N_{max}}$	$W(Q_i^{N_{max}})$	$Q_i^{N_{max}}$
0	0	128	518	256	94	384	214	512	364	640	414	768	819	896	966
1	1	129	54	257	204	385	309	513	654	641	223	769	814	897	755
2	2	130	83	258	298	386	188	514	659	642	663	770	439	898	859
3	4	131	57	259	400	387	449	515	335	643	692	771	929	899	940
4	8	132	521	260	608	388	217	516	480	644	835	772	490	900	830
5	16	133	112	261	352	389	408	517	315	645	619	773	623	901	911
6	32	134	135	262	325	390	609	518	221	646	472	774	671	902	871
7	3	135	78	263	533	391	596	519	370	647	455	775	739	903	639
8	5	136	289	264	155	392	551	520	613	648	796	776	916	904	888
9	64	137	194	265	210	393	650	521	422	649	809	777	463	905	479
10	9	138	85	266	305	394	229	522	425	650	714	778	843	906	946
11	6	139	276	267	547	395	159	523	451	651	721	779	381	907	750
12	17	140	522	268	300	396	420	524	614	652	837	780	497	908	969
13	10	141	58	269	109	397	310	525	543	653	716	781	930	909	508
14	18	142	168	270	184	398	541	526	235	654	864	782	821	910	861
15	128	143	139	271	534	399	773	527	412	655	810	783	726	911	757
16	12	144	99	272	537	400	610	528	343	656	606	784	961	912	970
17	33	145	86	273	115	401	657	529	372	657	912	785	872	913	919
18	65	146	60	274	167	402	333	530	775	658	722	786	492	914	875
19	20	147	280	275	225	403	119	531	317	659	696	787	631	915	862
20	256	148	89	276	326	404	600	532	222	660	377	788	729	916	758
21	34	149	290	277	306	405	339	533	426	661	435	789	700	917	948
22	24	150	529	278	772	406	218	534	453	662	817	790	443	918	977
23	36	151	524	279	157	407	368	535	237	663	319	791	741	919	923
24	7	152	196	280	656	408	652	536	559	664	621	792	845	920	972
25	129	153	141	281	329	409	230	537	833	665	812	793	920	921	761
26	66	154	101	282	110	410	391	538	804	666	484	794	382	922	877
27	512	155	147	283	117	411	313	539	712	667	430	795	822	923	952
28	11	156	176	284	212	412	450	540	834	668	838	796	851	924	495
29	40	157	142	285	171	413	542	541	661	669	667	797	730	925	703
30	68	158	530	286	776	414	334	542	808	670	488	798	498	926	935
31	130	159	321	287	330	415	233	543	779	671	239	799	880	927	978
32	19	160	31	288	226	416	555	544	617	672	378	800	742	928	883
33	13	161	200	289	549	417	774	545	604	673	459	801	445	929	762
34	48	162	90	290	538	418	175	546	433	674	622	802	471	930	503
35	14	163	545	291	387	419	123	547	720	675	627	803	635	931	925

36	72	164	292	292	308	420	658	548	816	676	437	804	932	932	878
37	257	165	322	293	216	421	612	549	836	677	380	805	687	933	735
38	21	166	532	294	416	422	341	550	347	678	818	806	903	934	993
39	132	167	263	295	271	423	777	551	897	679	461	807	825	935	885
40	35	168	149	296	279	424	220	552	243	680	496	808	500	936	939
41	258	169	102	297	158	425	314	553	662	681	669	809	846	937	994
42	26	170	105	298	337	426	424	554	454	682	679	810	745	938	980
43	513	171	304	299	550	427	395	555	318	683	724	811	826	939	926
44	80	172	296	300	672	428	673	556	675	684	841	812	732	940	764
45	37	173	163	301	118	429	583	557	618	685	629	813	446	941	941
46	25	174	92	302	332	430	355	558	898	686	351	814	962	942	967
47	22	175	47	303	579	431	287	559	781	687	467	815	936	943	886
48	136	176	267	304	540	432	183	560	376	688	438	816	475	944	831
49	260	177	385	305	389	433	234	561	428	689	737	817	853	945	947
50	264	178	546	306	173	434	125	562	665	690	251	818	867	946	507
51	38	179	324	307	121	435	557	563	736	691	462	819	637	947	889
52	514	180	208	308	553	436	660	564	567	692	442	820	907	948	984
53	96	181	386	309	199	437	616	565	840	693	441	821	487	949	751
54	67	182	150	310	784	438	342	566	625	694	469	822	695	950	942
55	41	183	153	311	179	439	316	567	238	695	247	823	746	951	996
56	144	184	165	312	228	440	241	568	359	696	683	824	828	952	971
57	28	185	106	313	338	441	778	569	457	697	842	825	753	953	890
58	69	186	55	314	312	442	563	570	399	698	738	826	854	954	509
59	42	187	328	315	704	443	345	571	787	699	899	827	857	955	949
60	516	188	536	316	390	444	452	572	591	700	670	828	504	956	973
61	49	189	577	317	174	445	397	573	678	701	783	829	799	957	1000
62	74	190	548	318	554	446	403	574	434	702	849	830	255	958	892
63	272	191	113	319	581	447	207	575	677	703	820	831	964	959	950
64	160	192	154	320	393	448	674	576	349	704	728	832	909	960	863
65	520	193	79	321	283	449	558	577	245	705	928	833	719	961	759
66	288	194	269	322	122	450	785	578	458	706	791	834	477	962	1008
67	528	195	108	323	448	451	432	579	666	707	367	835	915	963	510
68	192	196	578	324	353	452	357	580	620	708	901	836	638	964	979
69	544	197	224	325	561	453	187	581	363	709	630	837	748	965	953
70	70	198	166	326	203	454	236	582	127	710	685	838	944	966	763



71	44	199	519	327	63	455	664	583	191	711	844	839	869	967	974
72	131	200	552	328	340	456	624	584	782	712	633	840	491	968	954
73	81	201	195	329	394	457	587	585	407	713	711	841	699	969	879
74	50	202	270	330	527	458	780	586	436	714	253	842	754	970	981
75	73	203	641	331	582	459	705	587	626	715	691	843	858	971	982
76	15	204	523	332	556	460	126	588	571	716	824	844	478	972	927
77	320	205	275	333	181	461	242	589	465	717	902	845	968	973	995
78	133	206	580	334	295	462	565	590	681	718	686	846	383	974	765
79	52	207	291	335	285	463	398	591	246	719	740	847	910	975	956
80	23	208	59	336	232	464	346	592	707	720	850	848	815	976	887
81	134	209	169	337	124	465	456	593	350	721	375	849	976	977	985
82	384	210	560	338	205	466	358	594	599	722	444	850	870	978	997
83	76	211	114	339	182	467	405	595	668	723	470	851	917	979	986

121	193	249	770	377	302	505	410	633	690	761	856	889	829	1017	1020
122	152	250	107	378	649	506	231	634	713	762	839	890	965	1018	1007
123	77	251	180	379	771	507	688	635	632	763	725	891	938	1019	1015
124	164	252	151	380	360	508	653	636	482	764	698	892	884	1020	1019
125	768	253	209	381	539	509	248	637	806	765	914	893	506	1021	1021
126	268	254	284	382	111	510	369	638	427	766	752	894	749	1022	1022
127	274	255	648	383	331	511	190	639	904	767	868	895	945	1023	1023

## LDPC 码

### 速率匹配

Polar code 速率匹配包括子块交织、比特收集和比特交织。速率匹配的输入序列为

$d_0, d_1, d_2, \dots, d_{N-1}$ , 输出序列表示为  $f_0, f_1, f_2, \dots, f_{E-1}$

### polar code 速率匹配

#### 子块交织

编码比特  $d_0, d_1, d_2, \dots, d_{N-1}$  就是子块交织器的输入比特。编码比特  $d_0, d_1, d_2, \dots, d_{N-1}$  被分为 32 个子块。子块交织器的输出比特为  $y_0, y_1, y_2, \dots, y_{N-1}$ , 用以下方式生成：

```

for  $n = 0$  to  $N - 1$ 
     $i = \lfloor 32n / N \rfloor$ ;
     $J(n) = P(i) \times (N / 32) + \text{mod}(n, N / 32)$ ;
     $y_n = d_{J(n)}$ ;
end for
    
```

其中子块交织 pattern  $P(i)$  由 Table 5.4.1.1-1 给定。

**Table 5.4.1.1-1: Sub-block interleaver pattern  $P(i)$**

$i$	$P(i)$	$i$	$P(i)$	$i$	$P(i)$	$i$	$P(i)$	$i$	$P(i)$	$i$	$P(i)$	$i$	$P(i)$	$i$	$P(i)$
0	0	4	3	8	8	12	10	16	12	20	14	24	24	28	27
1	1	5	5	9	16	13	18	17	20	21	22	25	25	29	29
2	2	6	6	10	9	14	11	18	13	22	15	26	26	30	30
3	4	7	7	11	17	15	19	19	21	23	23	27	28	31	31

比特索引集合  $Q_{N/32}^{-}$  和  $Q_N^{-}$  由下列方法确定，其中  $K$ ， $n_{PC}$  和  $Q_N$

$-10Q_0N-1$  由 2.3.1 节定义：

```

 $\overline{\mathbf{Q}}_{F,tmp}^N = \emptyset$ 
if  $E \leq N$ 
    if  $K/E < 7/16$  -- puncturing
        for  $n = 0$  to  $N - E - 1$ 
             $\overline{\mathbf{Q}}_{F,tmp}^N = \overline{\mathbf{Q}}_{F,tmp}^N \cup \{J(n)\}$ ;
        end for
    if  $E \geq 3N/4$ 
         $\overline{\mathbf{Q}}_{F,tmp}^N = \overline{\mathbf{Q}}_{F,tmp}^N \cup \{0, 1, \dots, \lceil 3N/4 - E/2 \rceil - 1\}$ ;
    else
         $\overline{\mathbf{Q}}_{F,tmp}^N = \overline{\mathbf{Q}}_{F,tmp}^N \cup \{0, 1, \dots, \lceil 9N/16 - E/4 \rceil - 1\}$ ;
    end if
else -- shortening
    for  $n = E$  to  $N - 1$ 
         $\overline{\mathbf{Q}}_{F,tmp}^N = \overline{\mathbf{Q}}_{F,tmp}^N \cup \{J(n)\}$ ;
    end for
end if
end if
 $\overline{\mathbf{Q}}_{I,tmp}^N = \mathbf{Q}_0^{N-1} \setminus \overline{\mathbf{Q}}_{F,tmp}^N$ ;
 $\overline{\mathbf{Q}}_I^N$  comprises  $(K + n_{PC})$  most reliable bit indices in  $\overline{\mathbf{Q}}_{I,tmp}^N$ ;
 $\overline{\mathbf{Q}}_F^N = \mathbf{Q}_0^{N-1} \setminus \overline{\mathbf{Q}}_I^N$ ;

```

## 比特选择

$E$  表示速率匹配输出序列的长度，速率匹配输出序列为  $e_k$ ,  $k=0,1,2,\dots,E-1$

$k=0,1,2,\dots,E-1$  由以下方法产生：

```

if  $E > N$                                 -- repetition
    for  $k = 0$  to  $E - 1$ 
         $e_k = y_{\text{mod}(k, N)}$  ;
    end for
else
    if  $K/E < 7/16$                         -- puncturing
        for  $k = 0$  to  $E - 1$ 
             $e_k = y_{k+N-E}$  ;
        end for
    else                                    -- shortening
        for  $k = 0$  to  $E - 1$ 
             $e_k = y_k$  ;
        end for
    end if
end if

```

## 比特交织

比特序列  $e_0, e_1, e_2, \dots, e_{E-1}$  通过以下方法被交织为比特序列

$f_0, f_1, f_2, \dots, f_{E-1}$  :

$T_T$  表示满足  $T(T+1)/2 \geq E$  的最小整数 ;



```
 $k = 0;$   
for  $i = 0$  to  $T - 1$   
  for  $j = 0$  to  $T - 1 - i$   
    if  $k < E$   
       $v_{i,j} = e_k;$   
    else  
       $v_{i,j} = \langle NULL \rangle;$   
    end if  
     $k = k + 1;$   
  end for  
end for
```

```
 $k = 0;$   
for  $j = 0$  to  $T - 1$   
  for  $i = 0$  to  $T - 1 - j$   
    if  $v_{i,j} \neq \langle NULL \rangle$   
       $f_k = v_{i,j};$   
       $k = k + 1$   
    end if  
  end for  
end for
```

LDPC code 速率匹配

码块级联

上行传输信道和控制信息

随机接入信道

## 上行共享信道

### 传输块 CRC 添加

每个上行共享信道 (UL-SCH) 传输块都通过一个 CRC 来提供错误检测。

使用整个传输块来计算 CRC 校验比特。送到 L1 的一个传输块的比特表示为  $a_0, a_1, a_2, a_3, \dots, a_{A-1}$ ，校验比特为  $p_0, p_1, p_2, p_3, \dots, p_{L-1}$ ，其中  $A$  是传输块大小， $L$  是校验比特数。按照 TS 38.321 规范中的定义，最低顺序信息比特  $a_0$  被映射到最高有效位。

校验比特按照 2.1 节的描述进行计算并添加到 UL-SCH 传输块中，若  $A > 3824$  则设置  $L$  为 24 比特，使用生成多项式  $g_{CRC24A}(D)$ ；否则设置  $L$  为 16 比特，使用生成多项式  $g_{CRC16}(D)$ 。

### 码块分段及 CRC 添加

码块分段的输入比特流记为  $b_0, b_1, b_2, b_3, \dots, b_{B-1}$ ，其中  $B$  表示传输块的比特数目（包含 CRC）。

码块分段及 CRC 添加根据 2.2.1 节中描述进行。

码块分段之后的比特流记为  $Cr_0, Cr_1, Cr_2, Cr_3, \dots, Cr_{Kr-1}$ ，其中  $r$  是码块号， $Kr$  是码块  $r$  的比特数。

### UL-SCH 的信道编码

码块比特流送至信道编码模块。码块中的比特表示为  $Cr_0, Cr_1, Cr_2, Cr_3, \dots, Cr_{Kr-1}$

1)  $cr_0, cr_1, cr_2, cr_3, \dots, cr_{K_r-1}$  , 其中  $r_r$  是码块号 ,  $K_r$  是码块  $r_r$  的比特数。码块的总数表示为  $C$  , 每个码块根据 2.3.2 节的描述独立地进行 LDPC 编码。

编码之后的比特流表示为  $d_0, d_1, d_2, \dots, d_{N-1}$  , 其中对于 LDPC BG#1 , 有  $N = 66Z_c$  ; 对于 LDPC BG#2 , 有  $N = 50Z_c$  。  $Z_c$  的值由 2.2.1 节给定。

## 速率匹配

## 码块级联

## 数据和控制复用

## 信道交织

## 上行控制信息

## PUCCH 上的控制信息

## UCI 比特序列生成

[Editor notes: This section will capture how to generate the UCI

bit stream  $a_0, a_1, a_2, a_3, \dots, a_{A-1}$ ]

## CRC 添加

如果 UCI 负载大小  $A \geq A_0$  , 整个负载都用于计算 CRC 校验比特。负载比特表示

为  $a_0, a_1, a_2, a_3, \dots, a_{A-1}$ ，校验比特表示为  $p_0, p_1, p_2, p_3, \dots, p_{L-1}$ ，其中  $A$  是 UCI 负载大小， $L$  是校验比特数。

华为注： $A$  的值有待进一步确定。

校验比特按照 2.1 节的描述进行计算并添加，并设置  $L$  为  $X$  比特，得到添加 CRC 后的比特序列  $b_0, b_1, b_2, b_3, \dots, b_{B-1}$ ，其中  $B = A + L$ 。

对于  $X$  的值，华为注：Depending on further agreements, different CRC lengths may be applied for different payload sizes.

[Editor notes: need to define the relationship between  $b_0, b_1, b_2, b_3, \dots, b_{B-1}$  and  $c_0, c_1, c_2, c_3, \dots, c_{K-1}$ .]

## UCI 的信道编码

信息比特流送至信道编码模块。信息比特流表示为  $c_0, c_1, c_2, c_3, \dots, c_{K-1}$ ，其中  $K$  是比特数。

如果  $12 \leq K \leq 22$ ，则信息比特根据 2.3.1 节通过 Polar 进行编码，若  $E - K > 192$ ，则设置  $n_{\max} = 10$ ， $I_{\text{IL}} = 0$ ， $n_{\text{PC}} = 3$ ， $n_{\text{wPC}} = 1$ ；

若  $E - K \leq 192$ ，则设置  $n_{\text{wPC}} = 0$ ，其中  $E$  是由 3.3.1.4 节给定的速率匹配输出序列长度。

关于  $12 \leq K \leq 22$ ，华为批注：Need to capture the agreements for  $K < 12$  in the future.

如果  $K > 22$ ，则信息比特根据 2.3.1 节通过 Polar 进行编码，设置  $n_{\max} = 10$ ， $I_{\text{IL}} = 0$ ， $n_{\text{PC}} = 0$ ， $n_{\text{wmPC}} = 0$ 。

编码之后的比特流表示为  $d_0, d_1, d_2, d_3, \dots, d_{N-1}$ ，其中  $N$  是编码比特数。

## 速率匹配

## PUSCH 上的控制信息

### UCI 比特序列生成

[Editor notes: This section will capture how to generate the UCI

bit stream  $a_0, a_1, a_2, a_3, \dots, a_{A-1}$ ]

### CRC 添加

如果 UCI 负载大小  $A \geq A_0$ ，整个负载都用于计算 CRC 校验比特。负载比特表示为  $a_0, a_1, a_2, a_3, \dots, a_{A-1}$ ，校验比特表示为  $p_0, p_1, p_2, p_3, \dots, p_{L-1}$ ，其中  $A$  是 UCI 负载大小， $L$  是校验比特数。

华为注： $A_0$  的值有待进一步确定。

校验比特按照 2.1 节的描述进行计算并添加，并设置  $L$  为  $X$  比特，得到添加 CRC 后的比特序列  $b_0, b_1, b_2, b_3, \dots, b_{B-1}$ ，其中  $B = A + L$ 。

对于 X 的值，华为注：Depending on further agreements, different CRC lengths may be applied for different payload sizes.

[Editor notes: need to define the relationship between  $b_0, b_1, b_2, b_3, \dots, b_{B-1}$  and  $c_0, c_1, c_2, c_3, \dots, c_{K-1}$ .]

## UCI 的信道编码

信息比特流送至信道编码模块。信息比特流表示为  $c_0, c_1, c_2, c_3, \dots, c_{K-1}$ ，其中  $K$  是比特数。

如果  $12 \leq K \leq 22$ ，则信息比特根据 2.3.1 节通过 Polar 进行编码，若  $E - K > 192$ ，则设置  $n_{\max} = 10$ ， $I_{\text{IL}} = 0$ ， $n_{\text{PC}} = 3$ ， $n_{\text{wmPC}} = 1$ ；若  $E - K \leq 192$ ，则设置  $n_{\text{wmPC}} = 0$ ，其中  $E$  是由 3.3.2.4 节给定的速率匹配输出序列长度。

关于  $12 \leq K \leq 22$ ，华为批注：Need to capture the agreements for  $K < 12$  in the future.

如果  $K > 22$ ，则信息比特根据 2.3.1 节通过 Polar 进行编码，设置  $n_{\max} = 10$ ， $I_{\text{IL}} = 0$ ， $n_{\text{PC}} = 0$ ， $n_{\text{wmPC}} = 0$ 。

编码之后的比特流表示为  $d_0, d_1, d_2, d_3, \dots, d_{N-1}$ ，其中  $N$  是编码比特数。

## 速率匹配

## 下行传输和控制信息

### 广播信息

Figure 7.1-1 给出了 BCH 传输信道的处理结构。到达编码单元的数据，每 80ms 最多有一个传输块，其编码流程如下：

- 向传输块添加 CRC
- 信道编码
- 速率匹配

华为批注：Will add the figure later.

### 传输块 CRC 添加

BCH 传输块都通过 CRC 来进行错误检测。

使用整个传输块来计算 CRC 校验比特。（华为批注：This sentence may be updated depending on further agreements, e.g. if some PBCH payload bits are not used to generate CRC bits.）送到 L1 的一个传输块的比特表示为  $a_0, a_1, a_2, a_3, \dots, a_{A-1}$ ，校验比特为  $p_0, p_1, p_2, p_3, \dots, p_{L-1}$ ，其中  $A$  是传输块大小， $L$  是校验比特数。按照 TS 38.321 规范中的定义，最低顺序信息比特  $a_0$  被映射到最高有效位。

校验比特按照 2.1 节的描述进行计算并添加到 BCH 传输块中，设置  $L$  为 24 比特，使用生成多项式  $g_{CRC24C}(D)$ ，得到添加 CRC 后的序列为  $b_0, b_1, b_2, b_3, \dots, b_{B-1}$ ，其中  $B = A + L$ 。

[Editor notes: need to define the relationship between and ,  
pending on further agreements.]

## 信道编码

信息比特送至信道编码模块。信息比特表示为  $c_0, c_1, c_2, c_3, \dots, c_{K-1}$ ，其中  $K$  为比特数。它们根据 2.3.1 节进行 Polar 编码，设置  $n_{\max} = 9$ ， $I_{\text{IL}} = 1$ ， $n_{\text{PC}} = 0$ ， $n_{\text{wmPC}} = 0$ 。

编码之后的比特序列表示为  $d_0, d_1, d_2, d_3, \dots, d_{N-1}$ ，其中  $N$  为编码比特数。

## 速率匹配

## 下行共享信道和寻呼信道

### 传输块 CRC 添加

每个传输块都通过 CRC 来进行错误检测。

使用整个传输块来计算 CRC 校验比特。送到 L1 的一个传输块的比特表示为  $a_0, a_1, a_2, a_3, \dots, a_{A-1}$ ，校验比特为  $p_0, p_1, p_2, p_3, \dots, p_{L-1}$ ，其中  $A$  是传输块大小， $L$  是校验比特数。按照 TS 38.321 规范中的定义，最低顺序信息比特  $a_0$  被映射到最高有效位。

校验比特按照 2.1 节的描述进行计算并添加到 UL-SCH 传输块中，若  $A > 3824$ ，则设置  $L$  为 24 比特，使用生成多项式



$g_{CRC24A}(D)$ ；否则设置  $LL$  为 16 比特，使用生成多项式  $g_{CRC16}(D)$ 。

## 码块分段及 CRC 添加

码块分段的输入比特流记为  $b_0, b_1, b_2, b_3, \dots, b_{B-1}$ ，其中  $B$  表示传输块的比特数目（包含 CRC）。

码块分段及 CRC 添加根据 2.2.1 节中描述进行。

码块分段之后的比特流记为  $Cr_0, Cr_1, Cr_2, Cr_3, \dots, Cr_{Kr-1}$ ，其中  $r_r$  是码块号， $Kr$  是码块  $r_r$  的比特数。

## 信道编码

码块比特流送至信道编码模块。码块中的比特表示为  $Cr_0, Cr_1, Cr_2, Cr_3, \dots, Cr_{Kr-1}$ ，其中  $r_r$  是码块号， $Kr$  是码块  $r_r$  的比特数。码块的总数表示为  $C$ ，每个码块根据 2.3.2 节的描述独立地进行 LDPC 编码。

编码之后的比特流表示为  $d_0, d_1, d_2, \dots, d_{N-1}$ ，其中对于 LDPC BG#1，有  $N=66Z_c$ ；对于 LDPC BG#2，有  $N=50Z_c$ 。 $Z_c$  的值由 2.2.1 节给定。

## 速率匹配

## 下行控制信息

DCI 传输下行和上行调度信息，对非周期 CQI 报告请求，或对一个小区和 RNTI 的上行功率控制命令。

华为批注：This sentence will be updated later, pending on other agreed usages of DCI.

Figure 7.3-1 给出了 DCI 的处理结构。编码流程如下：

- 信息单元复用
- CRC 添加
- 信道编码
- 速率匹配

华为批注：Will add the figure later.

## DCI 格式

## CRC 添加

DCI 传输通过 CRC 进行错误检测。

使用整个 DCI 负载计算 CRC 校验比特。DCI 负载比特表示为  $a_0, a_1, a_2, a_3, \dots, a_{A-1}$ ，校验比特表示为  $p_0, p_1, p_2, p_3, \dots, p_{L-1}$ ，其中  $A$  为 DCI 负载比特大小， $L$  为校验比特数。

校验比特按照 2.1 节的描述进行计算并添加，并设置  $L$  为 24 比特，使用生成多项式  $g_{CRC24C}(D)$ ，得到添加 CRC 后的比特序列  $b_0, b_1, b_2, b_3, \dots, b_{B-1}$ ，其中  $B = A + L$ 。

[Editor notes: need to define the relationship between  $b_0, b_1, b_2, b_3, \dots, b_{B-1}$  and  $c_0, c_1, c_2, c_3, \dots, c_{K-1}$ , pending on further agreements.]

## 信道编码

信息比特送至信道编码模块。信息比特表示为  $c_0, c_1, c_2, c_3, \dots, c_{K-1}$ ，其中  $K$  是信息比特数，它们根据 2.3.1 节进行 Polar 编码，并设置  $n_{max} = 9$ ， $I_{IL} = 1$ ， $n_{PC} = 0$ ， $n_{wmPC} = 0$ 。

编码之后的比特序列表示为  $d_0, d_1, d_2, d_3, \dots, d_{N-1}$ ，其中  $N$  为编码比特数。

```

s = 0;
for r = 0 to C - 1
    if r < mod(B, C)
        K' = K+;
    else
        K' = K-;
    end
    k = 0;
    while k < K' - L
        crk = bi;
        k = k + 1;
        s = s + 1;
    end while
    if C > 1
        The sequence cr0, cr1, cr2, cr3, ..., cr(K'-L-1) is used to calculate
        the CRC parity bits pr0, pr1, pr2, ..., pr(L-1) according to section
        5.1.1 with the generator polynomial gCRC24B(D).
        while k < K'
            crk = pr(k+L-K');
            k = k + 1;
        end while
    end if
    while k < K          -- Insertion of filler bits
        crk = < NULL >;
        k = k + 1;
    end while
end for

```

## 信道编码

### Polar coding

对于给定的码块,信道编码器的输入序列为  $c_0, c_1, c_2, c_3, \dots, c_{K-1}$ , 其中  $K$  是编码器的输入比特数。编码后的比特序列表示为  $d_0, d_1, d_2, \dots, d_{N-1}$ , 其中  $N=2^n$  并且  $n$  的值由下列方法决定:

微信扫描以下二维码, 免费加入【5G 俱乐部】, 还赠送整套: 5G 前沿、NB-IoT、4G+ (VoLTE) 资料。

