TSG-RAN Working Group 1 (Radio) meeting #7 ***TSGR1#7(99)c34***
Hannover, Germany, Aug. 30-Sep. 3, 1999

Agenda Item : (Physical ad-hoc 1 meeting)

Source : Shinsegi Telecomm, Inc.[i], Hyundai Electronics Industries, Co., Ltd.

Title **TDD Cell Search Text Proposals for 25.221, 25.223 and 25.224 (CPM Based Cell Search Scheme)**

Document for : Approval

---

1. Introduction

A new cell search scheme based on the CPM and its simulation results are presented in Tdoc c33. The performances and complexities are better than the other proposals. Therefore, we propose this scheme as the cell search of TDD mode. In this document required changes are given as follows.

Text proposal for 25.221

5.4 The physical synchronisation channel (PSCH)

[Editors Note : The detailed scheme of CCCH pointing by SCH is FFS.]

The PSCH is similar to the FDD SCH. In order not to limit the UL/DL asymmetry the PSCH is mapped on one or two DL slots per frame only.

There are three cases of SCH and CCCH allocation as follows:

Case 1) SCH and CCCH allocated in TS#k, k=0….15~~4~~

Case 2) SCH in two TS and CCCH in the same two TS: TS#k and TS#k+8, k=0…~~7~~6

Case 3) SCH in two TS, TS#k and TS#k+8, k=0…7, and the primary CCCH TS#i, i=0…15~~4~~, pointed by SCH

The position of SCH (value of k) in frame can change on a long term basis in any case.

Figure 12 is one example, k=0, of Case 2 or Case 3. In this case, the PSCH uses system-wide always the same two DL slots, which are slot 0 and slot 8.

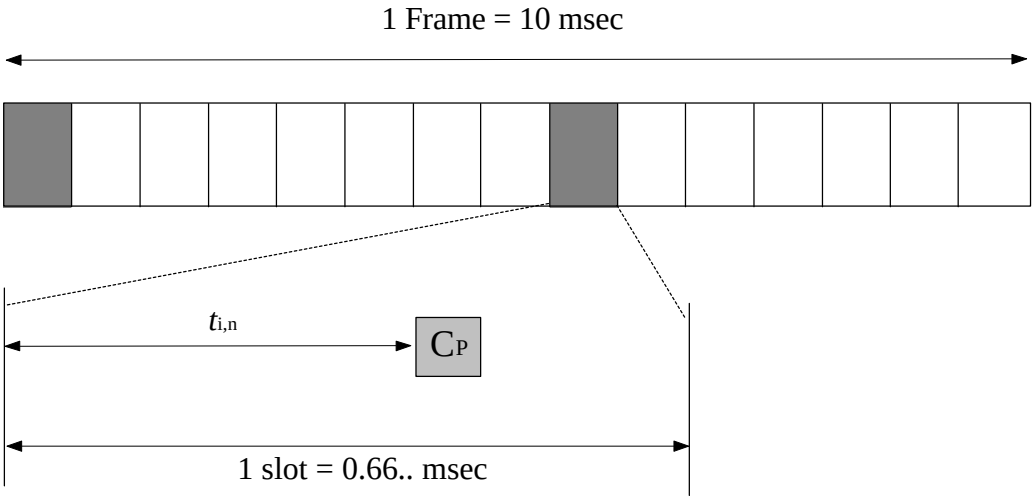

Figure 11~~12~~ Scheme for Synchronisation channel SCH consisting of one primary sequence C_p ~~and one secondary sequence C_s~~ per slot (example of Case 2 or Case 3)

As depicted in Figure 12, the PSCH consists of a primary ~~and secondary~~ code sequence with 256 chips length. The used sequence~~s~~ C_p ~~and C_s are~~ is the same as in FDD-Mode~~, see [2]~~.

~~The time offset t_{gap} is the time between the primary synchronisation code and the secondary synchronisation code. It provides enough time for calculations and a better interference distribution, since the codes do not superimpose. The exact value is to be determined.~~

~~Due to mobile to mobile interference, it is mandatory for public TDD systems to keep synchronisation between base stations. As a consequence of this, a capture effect concerning PSCH can arise. The time offset t_{offset} enables the system to overcome the capture effect~~.

The ~~time offset t_{offset}~~ code position $t_{i,n}$ is one of 32 values, depending on the cell parameter, thus on the code group of the cell, cf. 'Table 9 Mapping scheme for Cell Parameters, Code Groups, Scrambling Codes, Midambles ~~and t_{offset}~~' in 'TS25.223 Spreading and modulation (TDD)'. The exact value for ~~t_{offset}~~ $t_{i,n}$, regarding column ~~'Associated t_{offset}'~~ 'Associated Hopping Code H_i' .in Table 9 from TS25.221, is given by:

~~$$t_n = t_{offset,n} = n \cdot T_C \cdot \left\lfloor \frac{2560 - 96 - 512 - \frac{t_{gap}}{T_C}}{31} \right\rfloor \quad ; n = 0 \ldots 31$$~~

$$t_{i,n} = C_{i,n} \cdot T_c \cdot \left\lfloor \frac{2560 - 96 - 256}{31} \right\rfloor = C_{i,n} \cdot 71T_c \qquad ; i = 1, \ldots, 256 \quad ; n = 1, \ldots, 8$$

where, $C_{i,n}$ is the code element of a length 8 hopping code sequence which corresponding to each Hopping Code H_i, and the alphabet size of the hopping code is 32, that is,

$C_{i,n} \in \{0,1,...,31\}$

~~We recommend to include this equation into TS25.221 specification document, as then t_{offset} can be derived immediately when t_{gap} is given.~~

Text Proposal for 25.223

7. Synchronisation codes

7.1 Code Generation

The code generation for synchronisation codes is handled in the same way as in FDD Mode. Thus we refer to TS 25.213, chapter '5.2.3 Synchronisation Codes'. From this procedure we obtain one primary synchronisation code $C_p = C_{SCH,0}$ ~~and seventeen different secondary synchronisation codes $C_{S,i} = C_{SCH,i}$ with i=1…17.~~

To avoid misunderstandings when documents are reorganised in the future, we repeat the actual content of this chapter below using small font.

The Primary code sequence, C_p is constructed as a so-called generalised hierarchical Golay sequence. The Primary SCH is furthermore chosen to have good aperiodic auto correlation properties.
Letting a = < x_1, x_2, x_3, …, x_{16} > = < 0, 0, 0, 0, 0, 0, 1, 1, 0, 1,0, 1, 0, 1, 1, 0 > and
b = < x_1, x_2, x_3, …, x_8, x_1, x_2, x_3, …, x_8 >.
The PSC code is generated by repeating sequence 'a' modulated by a Golay complementary sequence.
Letting $y = < a, a, a, \bar{a}, \bar{a}, a, \bar{a}, \bar{a}, a, a, a, \bar{a}, a, \bar{a}, a, a >$
The definition of the PSC code word C_p follows (the left most index corresponds to the chip transmitted first in each time slot):
C_p=< y(0),y(1),y(2),...,y(255) >.

~~Let the sequence $z = < b, b, b, b, b, b, b, b, b, b, b, b, b, b, b, b >$. Then the Secondary Synchronization code words, $<C_1,\ldots,C_{17}>$ are constructed as the position wise addition modulo 2 of a Hadamard sequence and the sequence z.~~

~~The Hadamard sequences are obtained as the rows in a matrix H_8 constructed recursively by:~~

$$H_0 = (0)$$
$$H_k = \begin{pmatrix} H_{k-1} & H_{k-1} \\ H_{k-1} & \overline{H_{k-1}} \end{pmatrix}, \quad k \geq 1$$

~~The rows are numbered from the top starting with row 0 (the all zeros sequence).~~

~~The Hadamard sequence h depends on the chosen code number n and is denoted h_n in the sequel.~~

~~This code word is chosen from every 8^{th} row of the matrix H_8. Therefore, there are 32 possible codewords out of which n = 1, 2,…,17 are used.~~

~~Furthermore, let $h_n(i)$ and $z(i)$ denote the i:th symbol of the sequence h_n andz, respectively.~~

~~Then h_n is equal to the row of H_8 numbered by the bit reverse of the 8 bit binary representation of n.~~

~~The definition of the n:th SCH code word follows (the left most index correspond to the chip transmitted first in each slot):~~

~~$C_{SCH,n} = < h_n(0)+ z(0), h_n(1)+ z(1), h_n(2)+ z(2), \ldots, h_n(255)+ z(255) >,$~~

~~All sums of symbols are taken modulo 2.~~

These PSC ~~and SSC~~ binary code words are converted to real valued sequences by the transformation '0' -> '+1', '1' -> '-1'.

~~The Secondary SCHcode words are defined in terms of $C_{SCH,n}$ and the definition of $< C_1,\ldots,C_{17} >$ now follows as:~~

~~$C_i = C_{SCH,i}$, i=1,…,17~~

7.2 Code Allocation

~~Sequences of 8 secondary SCH codes, thus composed of $C_{S,i}$ from chapter 7.1 above, are used to transmit information on the PSCH.~~ In general the information on the code group of a cell and on the frame timing (see TS 25.224, Section '6.6.1 Cell Search') is transmitted in the PSCH. According to TS 25.221 section '7.4 The Physical Synchronisation Channel (PSCH)', there is case (3) where additional information from SCH transport channel is to be transmitted in the PSCH.

~~The sequences of secondary SCH codes are constructed such that their cyclic-shifts are unique, i.e. a non-zero cyclic shift less than 8 of any of the sequences is not equivalent to some cyclic shift of any other of the sequences. Also, a non-zero cyclic shift less than 8 of any of the sequences is not equivalent to itself with any other cyclic shift less than 8. This property is used to uniquely determine the transmitted sequence in the receiver.~~

The evaluation of transmitted information on code group and frame timing is shown in table 9, where the 32 code groups are listed. Each code group is containing 4 specific scrambling codes, each scrambling code associated with a specific short and long basic midamble code.

Each code group is additionally linked to a specific ~~t_{Offset}~~Hopping Code H_j, thus to a specific frame timing. By using this scheme, the UE can derive the position of the frame border due to the position of the SCH sequence and the knowledge of ~~t_{Offset}~~ Hopping Code H_j. ~~Positioning of the secondary SCH codes is depicted in the last line of table 10 and 11.~~

The complete mapping of Code Group to Scrambling Code, Midamble Codes and ~~t_{Offset}~~ Hopping Code H_j is depicted in table 9, cf. also TS 25.231.

CELL PARA-METER	Code Group	Associated Codes			Associated ~~t_{Offset}~~ Hopping Code
		Scrambling Code	Long Basic Midamble Code	Short Basic Midamble Code	
0	Group 1	Code 0	m_{PL0}	m_{SL0}	~~t_0~~ H_1-H_8
1		Code 1	m_{PL1}	m_{SL1}	
2		Code 2	m_{PL2}	m_{SL2}	
3		Code 3	m_{PL3}	m_{SL3}	
4	Group 2	Code 4	m_{PL4}	m_{SL4}	~~t_1~~ H_9-H_{16}
5		Code 5	m_{PL5}	m_{SL5}	
6		Code 6	m_{PL6}	m_{SL6}	
7		Code 7	m_{PL7}	m_{SL7}	
. . . .					
124	Group 32	Code 124	m_{PL124}	m_{SL124}	~~t_{31}~~ H_{249}-H_{256}
125		Code 125	m_{PL125}	m_{SL125}	
126		Code 126	m_{PL126}	m_{SL126}	
127		Code 127	m_{PL127}	m_{SL127}	

Table 9 Mapping scheme for Cell Parameters, Code Groups, Scrambling Codes, Midambles and ~~t_{Offset}~~ Hopping Code H_i.

For basic midamble codes m_P cf.TS 25.221, section '7.2.3.1 & 7.2.3.2 Midamble

For CELL PARAMETERS also cf. TS 25.231.

The following subchapters 7.2.1 and 7.2.2 are referring to the three cases of PSCH/CCPCH usage as described in TS 25.221 section 7.4.

7.2.1 Code allocation for case 1 and 2

In table 10 the 32 hopping code sequences used in the cases 1 and 2 of PSCH/CCPCH scheme are listed. Again, these are used to encode the 32 different code groups.

~~It should be mentioned that the sequences used here can be derived from FDD sequences by puncturing every 2nd position, thus a UE can use same database for FDD and TDD.~~

Code Group	~~Secondary~~ Primary SCH Code Position								~~Associated t_{Offset}~~ Hopping Code H_j
	#1	#2	#3	#4	#5	#6	#7	#8	
Group1	~~C1~~ 20	~~C2~~ 7	~~C6~~ 28	~~C15~~ 2	~~C8~~ 26	~~C7~~ 28	~~C3~~ 13	~~C11~~ 8	~~t_0~~ H_1
Group2	~~C1~~ 18	~~C9~~ 2	~~C10~~ 22	~~C13~~ 31	~~C11~~ 6	~~C3~~ 2	~~C2~~ 26	~~C16~~ 8	~~t_1~~ H_9
Group 3	~~C1~~ 1	~~C16~~ 8	~~C14~~ 25	~~C11~~ 25	~~C14~~ 3	~~C16~~ 31	~~C1~~ 12	~~C4~~ 27	~~t_2~~ H_{17}
Group 4	~~C1~~ 22	~~C6~~ 3	~~C1~~ 24	~~C9~~ 28	~~C17~~ 1	~~C12~~ 15	~~C17~~ 6	~~C9~~ 3	~~t_3~~ H_{25}
Group 5	~~C1~~ 29	~~C13~~ 26	~~C5~~ 8	~~C7~~ 4	~~C3~~ 9	~~C8~~ 4	~~C16~~ 17	~~C14~~ 18	~~t_4~~ H_{33}
Group 6	~~C1~~ 14	~~C3~~ 12	~~C9~~ 5	~~C5~~ 28	~~C6~~ 31	~~C4~~ 4	~~C15~~ 4	~~C2~~ 20	~~t_5~~ H_{41}
Group 7	~~C1~~ 0	~~C10~~ 3	~~C13~~ 13	~~C3~~ 31	~~C9~~ 7	~~C17~~ 28	~~C14~~ 18	~~C7~~ 1	~~t_6~~ H_{49}
Group 8	~~C1~~ 23	~~C17~~ 31	~~C17~~ 10	~~C1~~ 31	~~C12~~ 9	~~C1~~ 24	~~C13~~ 0	~~C12~~ 27	~~t_7~~ H_{57}
Group 9	~~C1~~ 16	~~C7~~ 21	~~C4~~ 25	~~C16~~ 11	~~C15~~ 31	~~C9~~ 24	~~C12~~ 18	~~C17~~ 2	~~t_8~~ H_{65}
Group 10	~~C1~~ 4	~~C14~~ 6	~~C8~~ 0	~~C14~~ 26	~~C1~~ 4	~~C5~~ 27	~~C11~~ 19	~~C5~~ 27	~~t_9~~ H_{73}
Group 11	~~C1~~ 30	~~C4~~ 26	~~C12~~ 25	~~C12~~ 22	~~C4~~ 11	~~C1~~ 27	~~C10~~ 1	~~C10~~ 24	~~t_{10}~~ H_{81}
Group 12	~~C1~~ 9	~~C11~~ 2	~~C16~~ 29	~~C10~~ 29	~~C7~~ 12	~~C14~~ 13	~~C9~~ 1	~~C15~~ 3	~~t_{11}~~ H_{89}
Group 13	~~C1~~ 19	~~C1~~ 6	~~C3~~ 30	~~C8~~ 11	~~C10~~ 3	~~C10~~ 28	~~C8~~ 23	~~C3~~ 21	~~t_{12}~~ H_{97}
Group 14	~~C1~~ 4	~~C8~~ 14	~~C7~~ 31	~~C6~~ 2	~~C13~~ 31	~~C6~~ 30	~~C7~~ 9	~~C8~~ 18	~~t_{13}~~ H_{105}
Group 15	~~C1~~ 24	~~C15~~ 1	~~C11~~ 4	~~C4~~ 3	~~C16~~ 28	~~C2~~ 19	~~C6~~ 1	~~C13~~ 12	~~t_{14}~~ H_{113}
Group 16	~~C1~~ 29	~~C5~~ 5	~~C15~~ 14	~~C2~~ 3	~~C2~~ 11	~~C15~~ 25	~~C5~~ 0	~~C1~~ 26	~~t_{15}~~ H_{121}
Group 17	~~C1~~ 30	~~C12~~ 22	~~C2~~ 2	~~C17~~ 4	~~C5~~ 16	~~C11~~ 22	~~C4~~ 19	~~C6~~ 13	~~t_{16}~~ H_{129}
Group 18	~~C2~~ 28	~~C11~~ 8	~~C14~~ 7	~~C4~~ 12	~~C10~~ 12	~~C1~~ 0	~~C15~~ 31	~~C8~~ 21	~~t_{17}~~ H_{137}
Group 19	~~C2~~ 10	~~C1~~ 23	~~C1~~ 1	~~C2~~ 1	~~C13~~ 5	~~C14~~ 26	~~C14~~ 29	~~C13~~ 18	~~t_{18}~~ H_{145}
Group 20	~~C2~~ 14	~~C8~~ 18	~~C5~~ 29	~~C17~~ 16	~~C16~~ 1	~~C10~~ 9	~~C13~~ 28	~~C1~~ 5	~~t_{19}~~ H_{153}
Group 21	~~C2~~ 29	~~C15~~ 0	~~C9~~ 28	~~C15~~ 23	~~C2~~ 30	~~C6~~ 0	~~C12~~ 4	~~C6~~ 16	~~t_{20}~~ H_{161}
Group 22	~~C2~~ 8	~~C5~~ 1	~~C13~~ 36	~~C13~~ 6	~~C5~~ 2	~~C2~~ 16	~~C11~~ 20	~~C11~~ 30	~~t_{21}~~ H_{169}
Group 23	~~C2~~ 0	~~C12~~ 7	~~C17~~ 11	~~C11~~ 17	~~C8~~ 3	~~C15~~ 29	~~C10~~ 29	~~C16~~ 31	~~t_{22}~~ H_{177}
Group 24	~~C2~~ 24	~~C2~~ 19	~~C4~~ 2	~~C9~~ 28	~~C11~~ 5	~~C11~~ 2	~~C9~~ 12	~~C4~~ 15	~~t_{23}~~ H_{185}

Group 25	~~C2~~ 8	~~C9~~ 9	~~C8~~ 27	~~C7~~ 27	~~C14~~ 26	~~C7~~ 22	~~C8~~ 1	~~C9~~ 23	~~t_{24}~~ H_{193}
Group 26	~~C2~~ 31	~~C16~~ 28	~~C12~~ 14	~~C5~~ 27	~~C17~~ 1	~~C3~~ 4	~~C7~~ 10	~~C14~~ 17	~~t_{25}~~ H_{201}
Group 27	~~C2~~ 31	~~C6~~ 27	~~C16~~ 25	~~C3~~ 0	~~C3~~ 21	~~C16~~ 7	~~C6~~ 11	~~C2~~ 8	~~t_{26}~~ H_{209}
Group 28	~~C2~~ 25	~~C13~~ 19	~~C3~~ 12	~~C1~~ 21	~~C6~~ 29	~~C12~~ 31	~~C5~~ 3	~~C7~~ 10	~~t_{27}~~ H_{217}
Group 29	~~C2~~ 0	~~C3~~ 30	~~C7~~ 18	~~C16~~ 12	~~C9~~ 25	~~C8~~ 18	~~C4~~ 13	~~C12~~ 9	~~t_{28}~~ H_{225}
Group 30	~~C2~~ 27	~~C10~~ 7	~~C11~~ 2	~~C14~~ 3	~~C12~~ 10	~~C4~~ 1	~~C3~~ 10	~~C17~~ 21	~~t_{29}~~ H_{233}
Group 31	~~C2~~ 7	~~C17~~ 1	~~C15~~ 28	~~C12~~ 0	~~C15~~ 6	~~C17~~ 30	~~C2~~ 30	~~C5~~ 20	~~t_{30}~~ H_{241}
Group 32	~~C2~~ 12	~~C7~~ 23	~~C2~~ 2	~~C10~~ 17	~~C1~~ 13	~~C13~~ 30	~~C1~~ 2	~~C10~~ 0	~~t_{31}~~ H_{249}
Frame position	Frame #1		Frame #2		Frame #3		Frame #4		

Table 10 Spreading Code allocation for ~~Secondary~~ Primary SCH Code, case 2) of PSCH/CCPCH scheme

7.2.2 Code allocation for case 3

In table 11 the 256 hopping code sequences used in case 3 of PSCH/CCPCH scheme are listed. In addition to the information on code group three bits from SCH transport channel are transmitted to the UE with these codes.

<Editors note: The usage of CCPCH pointing is for further study (cf. TDoc R1#2(99) 74)>

Code Group	~~Secondary~~ Primary PSCH Code at Position								Additional Bits from SCH Transport Channel	~~Associated~~ t_{Offset} Hopping Code H_i
	#1	#2	#3	#4	#5	#6	#7	#8		
Group 1	~~C2~~ 20	~~C14~~ 7	~~C6~~ 28	~~C8~~ 2	~~C4~~ 26	~~C9~~ 28	~~C17~~ 13	~~C15~~ 8	000	~~t_0~~ H_1, H_8
	~~C2~~ 16	~~C4~~ 16	~~C10~~ 28	~~C6~~ 14	~~C7~~ 5	~~C5~~ 8	~~C16~~ 19	~~C3~~ 17	001	
	~~C3~~ 13	~~C3~~ 11	~~C5~~ 29	~~C10~~ 12	~~C12~~ 18	~~C12~~ 22	~~C10~~ 29	~~C5~~ 6	010	
	~~C3~~ 20	~~C10~~ 25	~~C9~~ 6	~~C8~~ 28	~~C15~~ 21	~~C8~~ 1	~~C9~~ 27	~~C10~~ 28	011	
	~~C3~~ 21	~~C17~~ 31	~~C13~~ 11	~~C6~~ 5	~~C1~~ 13	~~C4~~ 4	~~C8~~ 27	~~C15~~ 7	100	
	~~C3~~ 2	~~C7~~ 31	~~C17~~ 21	~~C4~~ 29	~~C4~~ 228	~~C17~~ 1	~~C7~~ 28	~~C3~~ 4	101	
	~~C3~~ 4	~~C14~~ 4	~~C4~~ 6	~~C2~~ 13	~~C7~~ 29	~~C13~~ 25	~~C6~~ 23	~~C8~~ 0	110	
	~~C3~~ 18	~~C4~~ 24	~~C8~~ 3	~~C17~~ 0	~~C10~~ 7	~~C9~~ 29	~~C5~~ 13	~~C13~~ 24	111	
Group 2	~~C3~~ 18	~~C11~~ 2	~~C12~~ 22	~~C15~~ 31	~~C13~~ 6	~~C5~~ 2	~~C4~~ 26	~~C1~~ 8	000	~~t_1~~ H_9, H_{16}
	~~C3~~ 6	~~C1~~ 11	~~C16~~ 5	~~C13~~ 23	~~C16~~ 17	~~C1~~ 27	~~C3~~ 6	~~C6~~ 30	001	
	~~C3~~ 15	~~C8~~ 15	~~C3~~ 30	~~C11~~ 25	~~C2~~ 31	~~C14~~ 29	~~C2~~ 1	~~C11~~ 26	010	
	~~C3~~ 9	~~C15~~ 30	~~C7~~ 9	~~C9~~ 5	~~C5~~ 11	~~C10~~ 3	~~C1~~ 15	~~C16~~ 22	011	
	~~C3~~ 5	~~C5~~ 31	~~C11~~ 13	~~C7~~ 14	~~C8~~ 3	~~C6~~ 10	~~C17~~ 7	~~C4~~ 13	100	
	~~C3~~ 4	~~C12~~ 22	~~C15~~ 20	~~C5~~ 14	~~C11~~ 1	~~C2~~ 28	~~C16~~ 1	~~C9~~ 2	101	
	~~C3~~ 2	~~C2~~ 19	~~C2~~ 18	~~C3~~ 8	~~C14~~ 31	~~C15~~ 17	~~C15~~ 15	~~C14~~ 28	110	
	~~C3~~ 24	~~C9~~ 31	~~C6~~ 15	~~C1~~ 30	~~C17~~ 30	~~C11~~ 22	~~C14~~ 11	~~C2~~ 25	111	
Group 3	~~C3~~ 1	~~C16~~ 8	~~C10~~ 25	~~C16~~ 25	~~C3~~ 3	~~C7~~ 31	~~C13~~ 12	~~C7~~ 27	000	~~t_2~~ H_{17}, H_{24}
	~~C3~~ 30	~~C6~~ 7	~~C14~~ 31	~~C14~~ 5	~~C6~~ 2	~~C3~~ 11	~~C12~~ 5	~~C12~~ 10	001	
	~~C3~~ 12	~~C13~~ 21	~~C1~~ 11	~~C12~~ 12	~~C9~~ 24	~~C16~~ 17	~~C11~~ 20	~~C17~~ 18	010	
	~~C4~~ 16	~~C12~~ 0	~~C13~~ 12	~~C16~~ 29	~~C14~~ 2	~~C6~~ 30	~~C5~~ 12	~~C2~~ 8	011	
	~~C4~~ 17	~~C2~~ 8	~~C17~~ 26	~~C14~~ 6	~~C17~~ 0	~~C2~~ 6	~~C4~~ 29	~~C7~~ 25	100	
	~~C4~~ 23	~~C9~~ 21	~~C4~~ 23	~~C12~~ 4	~~C3~~ 30	~~C15~~ 14	~~C3~~ 6	~~C12~~ 0	101	

	C4 23	C16 8	C8 3	C10 19	C6 6	C11 21	C2 7	C17 14	110	
	C4 1	C6 3	C12 3	C8 30	C9 2	C7 10	C1 31	C5 26	111	
Group 4	C4 22	C13 3	C16 24	C6 28	C12 1	C3 15	C17 6	C10 3	000	t_3 H_{25}, H_{32}
	C4 8	C3 13	C3 15	C4 29	C15 6	C16 31	C16 19	C15 19	001	
	C4 5	C10 25	C7 27	C2 17	C1 8	C12 14	C15 25	C3 22	010	
	C4 17	C17 10	C11 7	C17 9	C4 28	C8 2	C14 3	C8 31	011	
	C4 3	C7 30	C15 21	C15 23	C7 3	C4 17	C13 24	C13 17	100	
	C4 22	C14 21	C2 2	C13 31	C10 2	C17 26	C12 15	C1 19	101	
	C4 13	C4 4	C6 3	C11 14	C13 30	C13 20	C11 22	C6 1	110	
	C4 8	C11 24	C10 22	C9 19	C16 10	C9 0	C10 0	C11 13	111	
Group 5	C4 29	C1 26	C14 8	C7 4	C2 9	C5 4	C9 17	C16 28	000	t_4 H_{33}, H_{40}
	C4 21	C8 0	C1 1	C5 4	C5 18	C1 1	C8 3	C4 25	001	
	C4 3	C15 28	C5 20	C3 10	C8 16	C14 10	C7 30	C9 1	010	
	C4 11	C5 30	C9 31	C1 14	C11 17	C10 29	C6 0	C14 14	011	
	C5 14	C4 22	C4 14	C5 1	C16 0	C17 23	C17 21	C16 30	100	
	C5 31	C11 12	C8 9	C3 9	C2 27	C13 1	C16 8	C4 3	101	
	C5 16	C1 29	C12 19	C1 5	C5 21	C9 2	C15 22	C9 11	110	
	C5 27	C8 18	C16 27	C16 7	C8 28	C5 25	C14 12	C14 2	111	
Group 6	C5 14	C15 12	C3 5	C14 28	C11 31	C1 4	C13 4	C2 20	000	t_5 H_{41}, H_{48}
	C5 28	C5 27	C7 14	C12 22	C14 31	C14 13	C12 29	C7 27	001	
	C5 36	C12 23	C11 8	C10 15	C17 20	C10 8	C11 11	C12 28	010	
	C5 6	C2 26	C15 31	C8 29	C3 14	C6 0	C10 22	C17 26	011	
	C5 27	C9 31	C2 31	C6 16	C6 8	C2 31	C9 3	C5 24	100	
	C5 0	C16 15	C6 28	C4 19	C9 27	C15 7	C8 16	C10 15	101	
	C5 30	C6 11	C10 18	C2 1	C12 31	C11 18	C7 26	C15 4	110	
	C5 31	C13 7	C14 4	C17 10	C15 30	C7 4	C6 2	C3 18	111	

Group7	C5 0	C3 3	C1 13	C15 31	C1 7	C3 28	C5 18	C8 1	000	t_6 H_{49}- H_{56}
	C5 9	C10 25	C5 2	C13 21	C4 29	C16 16	C4 2	C13 3	001	
	C5 1	C17 28	C9 14	C11 17	C7 0	C12 9	C3 0	C1 31	010	
	C5 5	C7 6	C13 1	C9 0	C10 19	C8 1	C2 23	C6 31	011	
	C5 2	C14 1	C17 27	C7 8	C13 16	C4 22	C1 22	C11 6	100	
	C6 14	C13 29	C12 1	C11 20	C 15	C11 5	C12 15	C13 11	101	
	C6 31	C3 21	C16 1	C9 1	C4 20	C7 8	C11 7	C1 0	110	
	C6 38	C10 24	C3 6	C7 26	C7 1	C3 0	C10 22	C6 7	111	
Group 8	C6 23	C17 31	C7 10	C5 31	C10 9	C16 24	C9 0	C11 27	000	t_7 H_{57}- H_{64}
	C6 27	C7 5	C11 29	C3 16	C13 26	C12 24	C8 1	C16 1	001	
	C6 8	C14 14	C15 19	C1 1	C16 13	C8 26	C7 18	C4 19	010	
	C6 16	C4 25	C2 0	C16 20	C2 26	C4 3	C6 19	C9 2	011	
	C6 26	C11 18	C6 24	C14 27	C5 29	C17 13	C5 14	C14 1	100	
	C6 24	C1 25	C10 1	C12 23	C8 1	C13 12	C4 2	C2 30	101	
	C6 5	C8 4	C14 0	C10 28	C11 25	C9 15	C3 13	C7 18	110	
	C6 17	C15 16	C1 4	C8 5	C14 19	C5 22	C2 3	C12 9	111	
Group 9\	C6 16	C5 21	C5 25	C6 11	C17 31	C1 24	C1 18	C17 2	000	t_8 H_{65}- H_{72}
	C6 14	C12 31	C9 1	C4 17	C3 1	C14 26	C17 9	C5 3	001	
	C6 8	C2 14	C13 27	C2 2	C6 1	C10 31	C16 24	C10 12	010	
	C6 3	C9 7	C17 26	C17 19	C9 26	C6 1	C15 23	C15 14	011	
	C6 1	C16 28	C4 4	C15 22	C12 24	C2 20	C14 14	C3 4	100	
	C6 12	C6 3	C8 20	C13 31	C15 23	C15 29	C13 1	C8 3	101	
	C7 27	C5 16	C3 8	C17 1	C3 2	C5 11	C7 21	C10 3	110	
	C7 0	C12 10	C7 1	C15 16	C6 27	C1 30	C6 9	C15 25	111	
Group 10	C7 4	C2 6	C11 0	C13 26	C9 4	C14 27	C5 19	C3 27	000	t_9 H_{73}- H_{80}
	C7 24	C9 20	C15 10	C11 10	C12 1	C10 11	C4 22	C8 16	001	

	C7 0	C16 22	C2 7	C9 27	C15 27	C6 3	C3 1	C13 18	010	
	C7 1	C6 27	C6 22	C7 25	C1 15	C2 2	C2 31	C1 25	011	
	C7 10	C13 12	C10 1	C5 20	C4 1	C15 13	C1 8	C6 23	100	
	C7 5	C3 14	C14 29	C3 0	C7 1	C11 20	C17 14	C11 27	101	
	C7 9	C10 18	C1 17	C1 6	C10 1	C7 31	C16 31	C16 25	110	
	C7 0	C17 27	C5 25	C16 3	C13 5	C3 30	C15 3	C4 26	111	
Group 11	C7 30	C7 26	C9 25	C14 22	C16 11	C16 27	C14 1	C9 24	000	t_{10} H_{81}- H_{88}
	C7 16	C14 6	C13 26	C12 23	C2 19	C12 27	C13 1	C14 28	001	
	C7 9	C4 17	C17 31	C10 0	C5 1	C8 19	C12 12	C2 24	010	
	C7 3	C11 23	C4 0	C8 9	C8 16	C4 17	C11 12	C7 0	011	
	C7 1	C1 10	C8 2	C6 25	C11 2	C17 17	C10 14	C12 22	100	
	C7 5	C8 18	C12 3	C4 1	C14 29	C13 20	C9 0	C17 0	101	
	C7 20	C15 25	C16 2	C2 25	C17 12	C9 0	C8 3	C5 1	110	
	C8 13	C14 20	C11 5	C6 1	C5 30	C16 0	C2 26	C7 12	111	
Group 12	C8 9	C4 2	C15 29	C4 29	C8 12	C12 13	C1 1	C12 3	000	t_{11} H_{89}- H_{96}
	C8 5	C11 10	C2 3	C2 5	C11 21	C8 16	C17 1	C17 19	001	
	C8 3	C1 13	C6 0	C17 17	C14 8	C4 23	C16 3	C5 6	010	
	C8 0	C8 13	C10 5	C15 1	C17 19	C17 18	C15 21	C10 5	011	
	C8 15	C15 21	C14 1	C13 29	C3 21	C13 18	C14 23	C15 2	100	
	C8 7	C5 2	C1 1	C11 1	C6 26	C9 19	C13 28	C3 5	101	
	C8 9	C12 15	C5 11	C9 4	C9 17	C5 4	C12 1	C8 25	110	
	C8 11	C2 5	C9 31	C7 29	C12 25	C1 13	C11 28	C13 1	111	
Group 13	C8 19	C9 6	C13 30	C5 11	C15 3	C14 28	C10 23	C1 21	000	t_{12} H_{97}- H_{104}
	C8 1	C16 23	C17 30	C3 10	C1 3	C10 20	C9 1	C6 19	001	
	C8 11	C6 25	C4 1	C1 25	C4 23	C6 30	C8 26	C11 9	010	
	C8 28	C13 29	C8 1	C16 24	C7 12	C2 9	C7 25	C16 1	011	

	C8 6	C3 6	C12 30	C14 3	C10 0	C15 16	C6 31	C4 10	100	
	C8 24	C10 28	C16 27	C12 5	C13 26	C11 27	C5 8	C9 1	101	
	C8 30	C17 21	C3 0	C10 0	C16 24	C7 4	C4 9	C14 13	110	
	C8 16	C7 1	C7 13	C8 2	C2 22	C3 1	C3 18	C2 20	111	
Group 14	C9 4	C6 14	C2 31	C12 2	C7 31	C10 30	C14 9	C4 18	000	t_{13} H_{105}-H_{112}
	C9 11	C13 0	C6 29	C10 11	C10 2	C6 10	C13 21	C9 30	001	
	C9 15	C3 13	C10 15	C8 8	C13 28	C2 1	C12 17	C14 27	010	
	C9 12	C10 22	C14 25	C6 30	C16 1	C15 15	C11 23	C2 17	011	
	C9 2	C17 30	C1 23	C4 6	C2 1	C11 31	C10 30	C7 1	100	
	C9 31	C7 22	C5 27	C2 1	C5 9	C7 1	C9 19	C12 0	101	
	C9 3	C14 23	C9 0	C17 11	C8 1	C3 31	C8 19	C17 20	110	
	C9 20	C4 1	C13 4	C15 20	C11 6	C16 5	C7 26	C5 18	111	
Group 15	C9 24	C11 1	C17 4	C13 3	C14 28	C12 19	C6 1	C10 12	000	t_{14} H_{113}-H_{120}
	C9 26	C1 5	C4 25	C11 15	C17 4	C8 1	C5 28	C15 5	001	
	C9 20	C8 30	C8 13	C9 28	C3 4	C4 24	C4 6	C3 1	010	
	C9 4	C15 27	C12 10	C7 31	C6 0	C17 30	C3 25	C8 1	011	
	C9 13	C5 6	C16 31	C5 15	C9 18	C13 6	C2 1	C13 22	100	
	C9 17	C12 127	C3 6	C3 28	C12 1	C9 12	C1 30	C1 15	101	
	C9 19	C2 2	C7 16	C1 18	C15 9	C5 21	C17 1	C6 2	110	
	C9 5	C9 26	C11 1	C16 26	C1 15	C1 17	C16 7	C11 29	111	
Group 16	C9 29	C16 5	C15 14	C14 3	C4 11	C14 25	C15 0	C16 26	000	t_{15} H_{121}-H_{128}
	C10 0	C15 28	C10 17	C1 20	C9 29	C4 7	C9 14	C1 14	001	
	C10 16	C5 31	C14 10	C16 4	C12 1	C17 29	C8 30	C6 4	010	
	C10 0	C12 15	C1 16	C14 0	C15 22	C13 1	C7 19	C11 26	011	
	C10 20	C2 1	C5 7	C12 4	C1 29	C9 4	C6 9	C16 22	100	
	C10 17	C9 31	C9 3	C10 15	C4 0	C5 31	C5 2	C4 30	101	

	C10 2	C16 29	C13 17	C8 1	C7 5	C1 27	C4 26	C9 6	110	
	C10 18	C6 0	C17 11	C6 8	C10 29	C14 27	C3 1	C14 31	111	
Group 17	C10 30	C13 22	C4 2	C4 4	C13 16	C10 22	C2 19	C2 13	000	t_{16} H_{129}- H_{136}
	C10 0	C3 6	C8 20	C2 1	C16 28	C6 24	C1 9	C7 3	001	
	C10 12	C10 1	C12 31	C17 2	C2 1	C2 16	C17 30	C12 26	010	
	C10 13	C17 16	C16 12	C15 1	C5 28	C15 12	C16 5	C17 2	011	
	C10 1	C7 28	C3 5	C13 10	C8 6	C11 27	C15 27	C5 18	100	
	C10 6	C14 13	C7 11	C11 28	C11 25	C7 1	C14 5	C10 10	101	
	C10 0	C4 30	C11 1	C9 29	C14 1	C3 23	C13 28	C15 8	110	
	C10 7	C11 30	C15 27	C7 1	C17 11	C16 16	C12 7	C3 3	111	
Group 18	C10 28	C1 8	C2 7	C5 12	C3 12	C12 0	C11 31	C8 21	000	t_{17} H_{137}- H_{144}
	C10 15	C8 30	C6 1	C3 6	C6 27	C8 11	C10 1	C13 17	001	
	C11 1	C7 26	C1 3	C7 25	C11 17	C15 28	C4 30	C15 11	010	
	C11 29	C14 17	C5 25	C5 12	C14 29	C11 30	C3 15	C3 30	011	
	C11 23	C4 22	C9 15	C3 3	C17 23	C7 7	C2 25	C8 7	100	
	C11 28	C11 24	C13 28	C1 29	C3 27	C3 2	C1 17	C13 25	101	
	C11 13	C1 6	C17 22	C16 13	C6 20	C16 31	C12 24	C1 10	110	
	C11 22	C8 27	C4 10	C14 20	C9 19	C12 20	C16 23	C6 15	111	
Group 19	C11 10	C15 23	C8 1	C12 1	C12 5	C8 26	C15 29	C11 18	000	t_{18} H_{145}- H_{152}
	C11 11	C5 24	C12 21	C10 9	C15 18	C4 23	C14 7	C16 27	001	
	C11 19	C12 10	C16 6	C8 17	C1 29	C17 29	C13 11	C4 19	010	
	C11 17	C2 23	C3 14	C6 5	C4 16	C13 27	C12 23	C9 18	011	
	C11 21	C9 25	C7 4	C4 27	C7 25	C9 13	C11 19	C14 14	100	
	C11 14	C16 8	C11 6	C2 20	C10 5	C5 18	C10 22	C2 16	101	
	C11 10	C6 6	C15 2	C17 18	C13 12	C1 21	C9 2	C7 11	110	
	C11 5	C13 10	C2 19	C15 6	C16 12	C14 25	C8 31	C12 30	111	

Group 20	C11 14	C3 18	C6 29	C13 16	C2 1	C10 9	C7 28	C17 5	000	t_{19} H_{153}, H_{160}
	C11 21	C10 6	C10 5	C11 19	C5 16	C6 11	C6 28	C5 2	001	
	C11 26	C17 17	C14 12	C9 23	C8 9	C2 18	C5 19	C10 12	010	
	C12 30	C16 5	C9 24	C13 11	C13 14	C9 14	C16 17	C12 30	011	
	C12 20	C6 28	C13 13	C11 27	C16 20	C5 26	C15 9	C17 26	100	
	C12 23	C13 30	C17 2	C9 20	C2 21	C1 5	C14 4	C5 30	101	
	C12 24	C3 13	C4 25	C7 26	C5 16	C14 29	C13 5	C10 20	110	
	C12 18	C10 26	C8 0	C5 20	C8 3	C10 15	C12 11	C15 7	111	
Group 21	C12 29	C17 0	C12 28	C3 23	C11 30	C6 0	C11 4	C3 16	000	t_{20} H_{161}, H_{168}
	C12 9	C7 13	C16 29	C1 8	C14 20	C2 23	C10 8	C8 25	001	
	C12 15	C14 8	C3 1	C16 31	C17 13	C15 0	C9 10	C13 18	010	
	C12 31	C4 9	C7 7	C14 17	C3 24	C11 3	C8 7	C1 26	011	
	C12 16	C11 14	C11 3	C12 20	C6 8	C7 12	C7 20	C6 21	100	
	C12 23	C1 25	C15 19	C10 23	C9 15	C3 30	C6 8	C11 1	101	
	C12 9	C8 11	C2 10	C8 25	C12 8	C16 4	C5 5	C16 11	110	
	C12 16	C15 12	C6 14	C6 31	C15 23	C12 22	C4 13	C4 31	111	
Group 22	C12 8	C5 1	C10 30	C4 6	C1 2	C8 16	C3 20	C9 30	000	t_{21} H_{169}, H_{176}
	C12 0	C12 26	C14 20	C2 22	C4 23	C4 23	C2 31	C14 0	001	
	C12 21	C2 21	C1 22	C17 24	C7 19	C17 12	C1 5	C2 4	010	
	C12 7	C9 8	C5 27	C15 29	C10 4	C13 13	C17 11	C7 21	011	
	C13 5	C8 18	C17 28	C2 20	C15 17	C3 4	C11 30	C9 19	100	
	C13 11	C15 9	C4 20	C17 2	C1 4	C16 8	C10 21	C14 9	101	
	C13 13	C5 25	C8 0	C15 10	C4 12	C12 9	C9 12	C2 3	110	
	C13 28	C12 17	C12 6	C13 16	C7 2	C8 4	C8 1	C7 1	111	
Group 23	C13 0	C2 7	C16 11	C11 17	C10 3	C4 29	C7 29	C12 31	000	t_{22} H_{177}, H_{184}
	C13 11	C9 17	C3 2	C9 3	C13 20	C17 16	C6 18	C17 4	001	

	C13 23	C16 28	C7 4	C7 30	C16 27	C13 11	C5 13	C5 18	010	
	C13 6	C6 1	C11 6	C5 22	C2 26	C9 21	C4 23	C10 5	011	
	C13 26	C13 5	C15 6	C3 29	C5 7	C5 3	C3 25	C15 27	100	
	C13 31	C3 6	C2 12	C1 2	C8 21	C1 0	C2 31	C3 25	101	
	C13 15	C10 14	C6 21	C16 28	C11 9	C14 3	C1 2	C8 8	110	
	C13 11	C17 28	C10 22	C14 18	C14 0	C10 17	C17 20	C13 20	111	
Group 24	C13 24	C7 19	C14 2	C12 28	C17 5	C6 2	C16 12	C1 15	000	t_{23} H_{185}, H_{192}
	C13 23	C14 17	C1 31	C10 25	C3 11	C2 26	C15 17	C6 15	001	
	C13 27	C4 4	C5 24	C8 25	C6 7	C15 1	C14 24	C11 14	010	
	C13 11	C11 19	C9 23	C6 7	C9 19	C11 18	C13 25	C16 1	011	
	C13 14	C1 11	C13 8	C4 9	C12 14	C7 19	C12 27	C4 19	100	
	C14 16	C17 15	C8 11	C8 18	C17 28	C14 23	C6 6	C6 1	101	
	C14 0	C7 13	C12 12	C6 28	C3 3	C10 21	C5 13	C11 4	110	
	C14 3	C14 22	C16 18	C4 27	C6 15	C6 1	C4 26	C16 22	111	
Group 25	C14 8	C4 9	C3 27	C2 27	C9 26	C2 22	C3 1	C4 23	000	t_{24} H_{193}, H_{200}
	C14 3	C11 8	C7 7	C17 5	C12 9	C15 12	C2 9	C9 13	001	
	C14 7	C1 11	C11 20	C15 15	C15 3	C11 25	C1 25	C14 5	010	
	C14 21	C8 4	C15 23	C13 8	C1 1	C7 21	C17 15	C2 23	011	
	C14 17	C15 11	C2 11	C11 28	C4 15	C3 4	C16 22	C7 13	100	
	C14 23	C5 11	C6 21	C9 9	C7 22	C16 1	C15 3	C12 21	101	
	C14 9	C12 16	C10 5	C7 18	C10 13	C12 31	C14 19	C17 11	110	
	C14 21	C2 16	C14 8	C5 12	C13 5	C8 1	C13 1	C5 15	111	
Group 26	C14 31	C9 28	C1 14	C3 27	C16 1	C4 4	C12 10	C10 17	000	t_{25} H_{201}, H_{208}
	C14 8	C16 5	C5 9	C1 21	C2 26	C17 0	C11 6	C15 9	001	
	C14 28	C6 18	C9 11	C16 19	C5 5	C13 11	C10 17	C3 7	010	
	C14 5	C13 8	C13 25	C14 2	C8 10	C9 17	C9 19	C8 11	011	

	C14 3	C3 17	C17 26	C12 29	C11 1	C5 28	C8 17	C13 13	100	
	C14 2	C10 28	C4 3	C10 1	C14 4	C1 29	C7 10	C1 14	101	
	C15 18	C9 14	C16 28	C14 1	C2 11	C8 3	C1 1	C3 20	110	
	C15 3	C16 20	C3 0	C12 3	C5 30	C4 12	C17 1	C8 7	111	
Group 27	C15 31	C6 27	C7 25	C10 0	C8 21	C17 7	C16 11	C13 8	000	t_{26} H_{209}, H_{216}
	C15 1	C13 19	C11 6	C8 1	C11 21	C13 2	C15 3	C1 24	001	
	C15 1	C3 3	C15 25	C6 6	C14 17	C9 18	C14 9	C6 30	010	
	C15 3	C10 1	C2 17	C4 0	C17 25	C5 29	C13 4	C11 21	011	
	C15 28	C17 24	C6 19	C2 28	C3 11	C1 9	C12 1	C16 27	100	
	C15 1	C7 9	C10 3	C17 22	C6 22	C14 19	C11 18	C4 10	101	
	C15 25	C14 21	C14 18	C15 16	C9 11	C10 16	C10 1	C9 5	110	
	C15 3	C4 14	C1 4	C13 1	C12 19	C6 9	C9 14	C14 7	111	
Group 28	C15 25	C11 19	C5 12	C11 21	C15 29	C2 31	C8 3	C2 10	000	t_{27} H_{217}, H_{224}
	C15 15	C1 7	C9 12	C9 6	C1 13	C15 1	C7 8	C7 31	001	
	C15 19	C8 13	C13 3	C7 10	C4 13	C11 31	C6 1	C12 16	010	
	C15 18	C15 0	C17 23	C5 4	C7 23	C7 2	C5 21	C17 3	011	
	C15 8	C5 26	C4 17	C3 5	C10 16	C3 7	C4 21	C5 1	100	
	C15 3	C12 9	C8 24	C1 1	C13 5	C16 20	C3 1	C10 31	101	
	C15 10	C2 26	C12 29	C16 30	C16 29	C12 15	C2 1	C15 11	110	
	C16 16	C1 14	C7 29	C3 27	C4 28	C2 1	C13 19	C17 23	111	
Group 29	C16 0	C8 30	C11 18	C1 12	C7 25	C15 18	C12 13	C5 9	000	t_{28} H_{225}, H_{232}
	C16 29	C15 1	C15 11	C16 27	C10 27	C11 2	C11 18	C10 19	001	
	C16 20	C5 1	C2 17	C14 5	C13 1	C7 10	C10 29	C15 15	010	
	C16 20	C12 21	C6 12	C12 2	C16 13	C3 13	C9 1	C3 20	011	
	C16 18	C2 24	C10 14	C10 2	C2 0	C16 12	C8 6	C8 8	100	
	C16 0	C9 18	C14 26	C8 21	C5 12	C12 1	C7 12	C13 20	101	

	C16 1	C16 1	C1 9	C6 0	C8 3	C8 17	C6 29	C1 25	110	
	C16 7	C6 9	C5 15	C4 24	C11 22	C4 1	C5 8	C6 18	111	
Group 30	C16 27	C13 7	C9 2	C2 3	C14 10	C17 1	C4 10	C11 21	000	t_{29} H_{233}, H_{240}
	C16 30	C3 1	C13 4	C17 11	C17 24	C13 27	C3 20	C16 11	001	
	C16 24	C10 9	C17 30	C15 1	C3 11	C9 8	C2 25	C4 7	010	
	C16 8	C17 14	C4 1	C13 24	C6 15	C5 1	C1 12	C9 4	011	
	C16 2	C7 16	C8 14	C11 6	C9 1	C1 24	C17 28	C14 21	100	
	C16 4	C14 23	C12 29	C9 24	C12 28	C14 6	C16 1	C2 20	101	
	C16 19	C4 30	C16 11	C7 11	C15 24	C10 1	C15 28	C7 9	110	
	C16 15	C11 2	C3 3	C5 18	C1 17	C6 12	C14 24	C12 1	111	
Group 31	C17 7	C10 1	C15 28	C9 0	C6 6	C13 30	C8 30	C14 20	000	t_{30} H_{241}, H_{248}
	C17 4	C17 6	C2 1	C7 25	C9 18	C9 17	C7 30	C2 0	001	
	C17 8	C7 0	C6 23	C5 23	C12 21	C5 5	C6 3	C7 4	010	
	C17 30	C14 6	C10 16	C3 20	C15 8	C1 1	C5 8	C12 14	011	
	C17 15	C4 14	C14 27	C1 0	C1 24	C14 1	C4 27	C17 18	100	
	C17 17	C11 11	C1 2	C16 15	C4 26	C10 2	C3 27	C5 0	101	
	C17 1	C1 14	C5 26	C14 8	C7 31	C6 1	C2 11	C10 31	110	
	C17 4	C8 24	C9 7	C12 30	C10 13	C2 3	C1 0	C15 2	111	
Group 32	C17 12	C15 23	C13 2	C10 17	C1 13	C15 30	C17 2	C3 0	000	T_{31}, H_{249}, H_{256}
	C17 20	C5 6	C17 25	C8 9	C16 6	C11 7	C16 1	C8 30	001	
	C17 10	C12 22	C4 17	C6 1	C2 24	C7 0	C15 22	C13 23	010	
	C17 11	C2 15	C8 19	C4 2	C5 24	C3 21	C14 1	C1 31	011	
	C17 27	C9 3	C12 8	C2 27	C8 30	C16 16	C13 24	C6 1	100	
	C17 6	C16 18	C16 27	C17 13	C11 27	C12 15	C12 31	C11 24	101	
	C17 4	C6 12	C3 27	C15 24	C14 1	C8 25	C11 30	C16 27	110	
	C17 4	C13 5	C7 9	C13 27	C17 31	C4 2	C10 0	C4 22	111	

Frame position	Frame #1	Frame #2	Frame #3	Frame #4

Table 11 Spreading Code allocation for ~~Secondary~~Primary SCH Code, case 3) of PSCH/CCPCH scheme

Text Proposal for 25.224

4.5 Synchronisation and Cell Search Procedures

4.5.1 Cell Search

During the initial cell search, the UE searches for a cell. It then determines the midamble, the downlink scrambling code and frame synchronisation of that cell. The initial cell search uses the Physical Synchronisation Channel (PSCH) described in ~~S1.21~~25.221. The generation of synchronisation codes is described in ~~S1.23~~25.223
This initial cell search is carried out in three steps:

Step 1: Slot synchronisation

During the first step of the initial cell search procedure the UE uses the primary synchronisation code c_p to acquire slot synchronisation to the strongest cell. Furthermore, frame synchronisation with the uncertainty of 1 out of 2 is obtained in this step. A single matched filter (or any similar device) is used for this purpose, that is matched to the primary synchronisation code which is common to all cells. The procedure is according to the description for the FDD mode in ~~S1.14~~25.214.

Step 2: Frame synchronisation and code-group identification

The Step 2 is described for the case where PSCH and CCPCH are in timeslot k and k+8 with k=0…~~7~~6.

During the second step of the initial cell search procedure, the UE uses the ~~sequence of Secondary~~ code positions of Primary Synchronisation Codes to find frame synchronisation and identify one of 32 code groups. Each code group is linked ~~to a specific t_{Offset}, thus~~ to a specific ~~frame timing~~time hopping code sequence, and is containing 4 specific scrambling codes. Each scrambling code is associated with a specific short and long basic midamble code.

~~The detection of secondary synchronisation sequence is done by correlating the received signal at the positions of the Secondary Synchronisation Code with all possible sequences of Secondary Synchronisation Codes, similar to FDD Mode.~~ The detection of time hopping code sequence is done by testing the matched filter output of the received signal with all possible code positions shown in Table 10 and 11.

After four frames a sequence of eight codes is available providing all necessary information described above. Nevertheless, it should be noted that due to the special coding already three codes show the sequence unambiguously, i.e. a UE can determine the whole sequence when three codes have been received.~~.~~

~~-~~**Step 3: Scrambling code identification**

During the third and last step of the initial cell-search procedure, the UE determines the exact basic midamble code and the accompanying scrambling code used by the found cell. They are identified through correlation over the CCPCH with all four midambles of the code group identified in the second step . Thus the third step is a one out of four decision.
This step is taking into account that the CCPCH containing the BCH is transmitted using the first spreading

code ($a_{Q=16}^{(h=1)}$ in figure 2 of ~~S1.23~~ 25.223 section '6.2 Spreading Codes') and using the first midamble $\mathbf{m}^{(1)}$ (derived from basic midamble code $\mathbf{m}_P$, cf. ~~S1.21~~ 25.221 section '7.2.3 Training sequences for spread bursts'). Thus CCPCH code and midamble can be immediately derived when knowing scrambling code and basic midamble code.