

# Midterm Exam

Youngjoo Lee  
Assistant Professor,  
POSTECH EE

**Modify your previous UART slave module to support the following encoding modes**

- ✓ Binary mode (the same as your previous works)
- ✓ 8b/10b encoding mode

**Design your API functions for the following operations**

- ✓ RX, TX mode setting
- ✓ Baudrate setting
- ✓ Encoding mode setting
- ✓ Data transmission

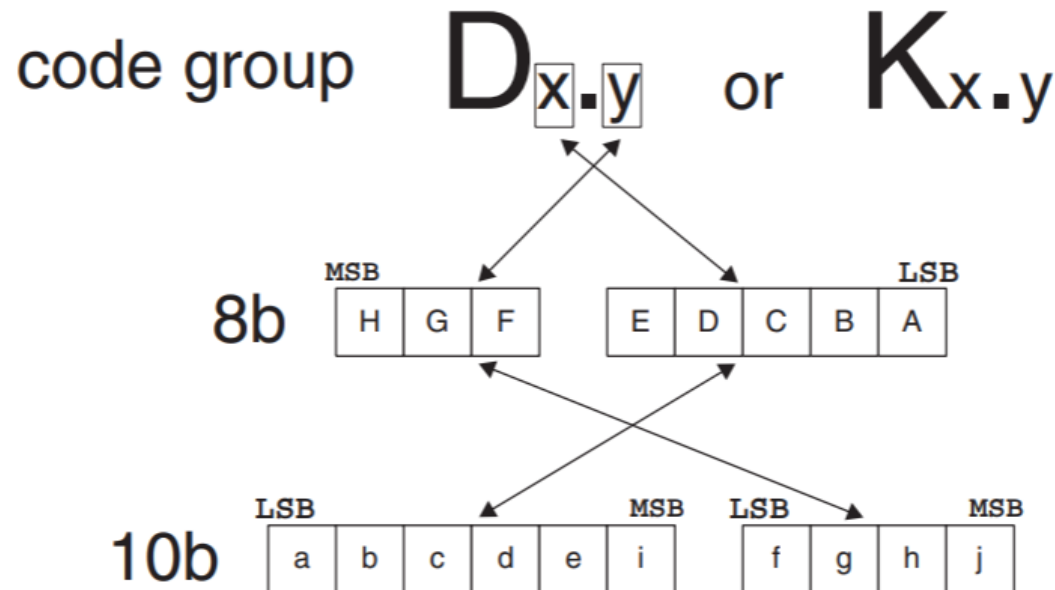
**8b/10b is a line code that maps 8-bit words to 10-bit symbols to achieve DC-balance and bounded disparity.**

- ✓ This helps to reduce the demand for the lower bandwidth limit of the channel necessary to transfer the signal.
- ✓ The difference between the counts of ones and zeros in a string of at least 20 bits is no more than two.
- ✓ There are not more than five ones or zeros in a row.

**Firstly developed by IBM for fiber channel communications (ANSI X3T11/Project 755D/Rev 4.3), and now widely applied to various protocols.**

**The 8b/10b encoder converts 8-bit code groups into 10-bit codes.**

- ✓ The code groups include 256 data characters named  $D_{x.y}$  and 12 control characters named  $K_{x.y}$ .
  - $x$ : 5 least significant bits  $\rightarrow$  encoded into 6 bits
  - $y$ : 3 most significant bits  $\rightarrow$  encoded into 4 bits



## 3b-to5b encoding table

3b Decimal	3b Binary (HGF)	4b Binary (fghi)
0	000	0100 or 1011
1	001	1001
2	010	0101
3	011	0011 or 1100
4	100	0010 or 1101
5	101	1010
6	110	0110
7	111	0001 or 1110 or 1000 or 0111

## 5b-to-6b encoding table

5b Decimal	5b Binary (EDCBA)	6b Binary (abcdei)
0	00000	100111 or 011000
1	00001	011101 or 100010
2	00010	101101 or 010010
3	00011	110001
4	00100	110101 or 001010
5	00101	101001
6	00110	011001
7	00111	111000 or 000111
8	01000	111001 or 000110
9	01001	100101
10	01010	010101
11	01011	110100
12	01100	001101
13	01101	101100
14	01110	011100
15	01111	010111 or 101000
16	10000	011011 or 100100
17	10001	100011
18	10010	010011
19	10011	110010

## 5b-to-6b encoding table (cont.)

20	10100	001011
21	10101	101010
22	10110	011010
23	10111	111010 or 000101
24	11000	110011 or 001100
25	11001	100110
26	11010	010110
27	11011	110110 or 001001
28	11100	001110
29	11101	101110 or 010001
30	11110	011110 or 100001
31	11111	101011 or 010100

## **1. UART module basic operations with APB interfaces (50)**

- ✓ Binary TX/RX (10)
- ✓ Register map defines (10)
- ✓ 8b/10b TX (10)
- ✓ 8b/10b RX (10)
- ✓ Changing baudrates (10)

## **2. C programs for Cortex-M0 SoC (30)**

- ✓ Single C program for testing all the UART operations (15)
- ✓ Valid operations in waveform viewer (15)

## **3. APIs for UART operations (20)**

- ✓ API definitions for controlling UART modules (10)
- ✓ Single C program using API functions (10)



**You have to show your design results at the computer room (LG bldg.) on Oct. 30 (14:00~16:30)**

- ✓ Please prepare the verification scenarios so that we can save the checking time by repeatedly recompiling a whole system.
- ✓ It might be better to prepare multiple Quartus-II projects and pre-compiled HEX files.
- ✓ If you have any urgent schedule on the demonstration day, please let me know immediately.