

Bộ môn: Cơ Điện Tử
Khoa: Cơ khí

ĐÁP ÁN GIỮA HK II/10-11

Môn: **TỔ CHỨC VÀ CẤU TRÚC MÁY TÍNH**
Lớp: **CK07CD1**
Ngày thi: **10/04/2011**
Thời gian: **45 phút**

(Sinh viên ĐƯỢC sử dụng tài liệu + Laptop)

Câu 1) (2 điểm)

8 bits được sử dụng để hiện thực số nguyên có dấu theo 3 cách là **two's complement**, **8-bit signed magnitude** và **8-bit excess 127**. Điền các mẫu 8 bits tương ứng vào bảng sau.

Signed Integer	Two's Complement	8-bit Signed Magnitude	8-bit Excess 127
- 1	1111 1111	1000 0001	0111 1110
+ 128			1111 1111
0	0000 0000	0000 0000	0111 1111
-127	1000 0001	1111 1111	0000 0000

Câu 2) (2.5 điểm)

Single precision IEEE 754 (base 2 scientific notation) sử dụng **32 bits** để hiện thực các số thực. Trong đó gồm **1 bit** dấu, **8 bits** để hiện thực số mũ (exponent) theo phương pháp mã hoá **excess 127**, và **23 bits** dành cho số lẻ (fraction) ở dạng chuẩn hoá (**normalized form**).

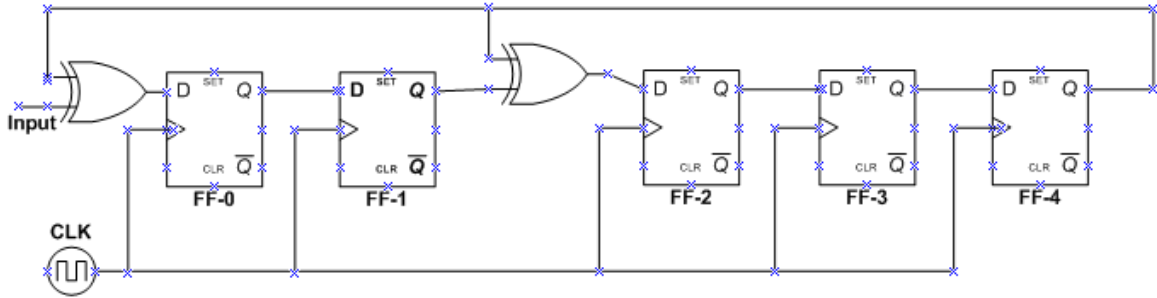
Chú ý:

- Hidden bit 1 nằm bên trái dấu chấm thập phân. Số **1.0101** là đã chuẩn hoá. Số **.101** là không chuẩn hoá.

Hãy chuyển đổi và điền cho đầy đủ thông tin vào bảng sau:

Value	Floating point representation		
	Sign	Exponent	Fraction
+99	0	1000 0101	100 0110 0000 0000 0000 0000
-125.625	1	1000 0101	111 1011 0100 0000 0000 0000
-259.875	1	1000 0111	000 0001 1111 0000 0000 0000
-2055.0	1	1000 1010	000 0000 0111 0000 0000 0000
1	0	0111 1111	000 0000 0000 0000 0000 0000

Câu 3) (3 điểm)



Cho mạch điện để tạo ra CRC code với polinomial generator **P** là: $x^5 + x^2 + 1$

Giả sử dữ liệu cần truyền là: **1001 0011** Hãy:

- a) Xác định CRC bằng phương pháp đại số (Modulo 2 division) (1 điểm)
- b) Xác định **CRC** (1 điểm)
- c) Xác định chuỗi bit sẽ truyền và kiểm tra (1 điểm)

Chú ý: Câu 3b) và 3c) là chạy từng bước và điền thông tin theo bảng bên:

Time	Input	FF-4	FF-3	FF-2	FF-1	FF-0
1	1					
2	0					
3	0					
4	1					
5	0					
6	0					
7	1					
8	1					

a)										
1	0	0	1	0	1	1	0	0	1	1
						1	0	0	0	0
						0	0	0	0	1
						1	0	0	1	0
						0	1	1	1	0
						1	0	0	1	0
						0	1	1	1	1
						1	0	0	1	0
						0	1	1	0	1

b)						
Time	Input	FF-4	FF-3	FF-2	FF-1	FF-0
1	1	0	0	0	0	0
2	0	0	0	0	0	1
3	0	0	0	0	1	0
4	1	0	0	1	0	0
5	0	0	1	0	0	1
6	0	1	0	0	1	0
7	1	0	0	0	0	1
8	1	0	0	0	1	1
9	0	0	0	1	1	1
10	0	0	1	1	1	0
11	0	1	1	1	0	0
12	0	1	1	1	0	1
13	0	1	1	1	1	1
14		1	1	0	1	1

c)						
Time	Input	FF-4	FF-3	FF-2	FF-1	FF-0
1	1	0	0	0	0	0
2	0	0	0	0	0	1
3	0	0	0	0	1	0
4	1	0	0	1	0	0
5	0	0	1	0	0	1
6	0	1	0	0	1	0
7	1	0	0	0	0	1
8	1	0	0	0	1	1
9	1	0	0	1	1	1
10	1	0	1	1	1	1
11	0	1	1	1	1	1
12	1	1	1	0	1	1
13	1	1	0	0	1	0
14		0	0	0	0	0

Câu 4) (2.5 điểm)

Sử dụng Hamming code và even parity để mã hoá chuỗi ký tự “**Hamming**”. Điền tất cả code word vào bảng sau: (**P_i** là các even parity bits)

Chú ý: Điền theo đúng thứ tự bits được cho ở bảng.

Char	ASCII							Hamming Code										
	b6	b5	b4	b3	b2	b1	b0	1	2	3	4	5	6	7	8	9	10	11
								P1	P2	b6	P4	b5	b4	b3	P8	b2	b1	b0
H (48H)	1	0	0	1	0	0	0	0	0	1	1	0	0	1	0	0	0	0
a (61H)	1	1	0	0	0	0	1	1	0	1	1	1	0	0	1	0	0	1
m (6DH)	1	1	0	1	1	0	1	1	1	1	0	1	0	1	0	1	0	1
m (6DH)	1	1	0	1	1	0	1	1	1	1	0	1	0	1	0	1	0	1
i (69H)	1	1	0	1	0	0	1	0	1	1	0	1	0	1	1	0	0	1
n (6EH)	1	1	0	1	1	1	0	0	1	1	0	1	0	1	0	1	1	0
g (67H)	1	1	0	0	1	1	1	0	1	1	1	1	0	0	1	1	1	1