

Elementi di Architettura
Elementi di Architettura e Sistemi Operativi - Modulo I
Laurea in Bioinformatica

Tiziano Villa

23 Febbraio 2021

Nome e Cognome:

Matricola:

Posta elettronica:

| problema | punti massimi | i tuoi punti |
|------------|---------------|--------------|
| problema 1 | 10 | |
| problema 2 | 10 | |
| problema 3 | 10 | |
| totale | 30 | |

1. (a) Si scriva la rappresentazione nel formato IEEE per la virgola mobile del numero decimale 3,75.

Si mostrino i passi per ottenerlo.

Traccia di soluzione.

Rappresentando $3,75_{10}$ in base 2 si ottiene $11,11_2$ che normalizzato diventa $1,111_2 \times 2^1$.

Il campo dell'esponente e' 10000000 ottenuto da $x - 127 = 1$, cioè $x = 128$ ($128 - 127 = 1$).

Eliminando la cifra 1 iniziale, il campo della parte frazionaria e' 1110000000000000000000.

La rappresentazione IEEE e' 0 10000000 111000000000000000000000.

- (b) Si scriva il numero decimale rappresentato nel formato IEEE per la virgola mobile dalla seguente codifica:

0 10000000 000000000000000000000000.

Si mostrino i passi per ottenerlo.

Traccia di soluzione.

L'esponente e' $128 - 127 = 1$.

il numero e' $+1,0...0 \times 2^1 = 2_{10}$.

- (c) Si consideri la rappresentazione dei numeri in base 4.

Qual e' il massimo decimale senza segno che si puo' rappresentare con 3 cifre in base 4 ?

Traccia di soluzione.

$63_{10} = (333_4)$.

- (d) Qual e' il massimo decimale senza segno che si puo' rappresentare con n cifre in base 4 ? La risposta deve essere una funzione di n .

Traccia di soluzione.

$4^n - 1$.

2. Si progetti un circuito sequenziale con due variabili binarie d'ingresso A, B e una variabile binaria d'uscita U che produce all'istante t la funzione congiunzione (AND) logica degli ingressi all'istante $t-2$, cioè $U_t = AND(A_{t-2}, B_{t-2})$.

(a) Si disegni il grafo delle transizioni di una macchina a stati finiti di tipo Moore che corrisponde alla specifica. S'indichi lo stato iniziale. Si supponga che sia $U = 0$ agli istanti $t = 0, t = 1$.

(b) Si scriva la tavola delle transizioni con gli stati futuri e le uscite, e la si minimizzi.

(c) Si codifichi la tavola delle transizioni minimizzata.

Traccia di soluzione.

Tavola delle transizioni della MSF prima della codifica (stato iniziale $s00$).

| A | B | Sp | Sf | U |
|---|---|-----|-----|---|
| 0 | 0 | s00 | s00 | 0 |
| 0 | 1 | s00 | s00 | 0 |
| 1 | 0 | s00 | s00 | 0 |
| 1 | 1 | s00 | s10 | 0 |

| | | | | |
|---|---|-----|-----|---|
| 0 | 0 | s01 | s00 | 1 |
| 0 | 1 | s01 | s00 | 1 |
| 1 | 0 | s01 | s00 | 1 |
| 1 | 1 | s01 | s10 | 1 |

| | | | | |
|---|---|-----|-----|---|
| 0 | 0 | s10 | s01 | 0 |
| 0 | 1 | s10 | s01 | 0 |
| 1 | 0 | s10 | s01 | 0 |
| 1 | 1 | s10 | s11 | 0 |

| | | | | |
|---|---|-----|-----|---|
| 0 | 0 | s11 | s01 | 1 |
| 0 | 1 | s11 | s01 | 1 |
| 1 | 0 | s11 | s01 | 1 |
| 1 | 1 | s11 | s11 | 1 |

Tavola delle transizioni della MSF codificata.

| A | B | P0 | P1 | F0 | F1 | U |
|---|---|----|----|----|----|---|
| 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 1 | 0 | 0 | 0 | 0 | 0 |
| 1 | 0 | 0 | 0 | 0 | 0 | 0 |
| 1 | 1 | 0 | 0 | 1 | 0 | 0 |
| | | | | | | |
| 0 | 0 | 0 | 1 | 0 | 0 | 1 |
| 0 | 1 | 0 | 1 | 0 | 0 | 1 |
| 1 | 0 | 0 | 1 | 0 | 0 | 1 |
| 1 | 1 | 0 | 1 | 1 | 0 | 1 |
| | | | | | | |
| 0 | 0 | 1 | 0 | 0 | 1 | 0 |
| 0 | 1 | 1 | 0 | 0 | 1 | 0 |
| 1 | 0 | 1 | 0 | 0 | 1 | 0 |
| 1 | 1 | 1 | 0 | 1 | 1 | 0 |
| | | | | | | |
| 0 | 0 | 1 | 1 | 0 | 1 | 1 |
| 0 | 1 | 1 | 1 | 0 | 1 | 1 |
| 1 | 0 | 1 | 1 | 0 | 1 | 1 |
| 1 | 1 | 1 | 1 | 1 | 1 | 1 |

- (d) Supponendo di usare bistabili di tipo D, si derivino le equazioni minimizzate di eccitazione degl'ingressi dei bistabili e le equazioni minimizzate delle uscite.

Traccia di soluzione.

$$F0 = AB, F1 = P0, U = P1.$$

- (e) Si realizzi il circuito sequenziale corrispondente con bistabili di tipo D campionati sul fronte di salita, invertitori e porte NAND (a 2, 3, o 4 ingressi). Si etichettino con chiarezza i segnali.

Traccia di soluzione.

Il circuito sequenziale e' costituito da una porta AND con ingressi A e B e uscita $F0$ in ingresso a un bistabile con uscita $P0$ che coincide con l'ingresso $F1$ a un secondo bistabile in serie la cui uscita $P1$ coincide con l'uscita U .

Ovviamente si puo' esprimere la porta AND con una porta NAND negata (la negazione a sua volta si esprime con un NAND con un unico ingresso).

3. Si analizzi il codice seguente, sia commentando le singole istruzioni, sia spiegando la funzionalità complessiva.

```

                                LDR    R0, R5, #0
                                BRz    SALTA

                                LDR    R0, R5, #-1
                                ADD    R0, R0, #1
                                STR    R0, R5, #-1

                                LDR    R0, R5, #-2
                                ADD    R0, R0, #-1
                                STR    R0, R5, #-2
                                BR      Fine

SALTA:                        LDR    R0, R5, #-1
                                ADD    R0, R0, #-1
                                STR    R0, R5, #-1

                                LDR    R0, R5, #-2
                                ADD    R0, R0, #1
                                STR    R0, R5, #-2

Fine:                        ...
                                ...
```

Traccia di soluzione.

Si supponga che l'indirizzo di x si trovi in $R5$, quelli di y e z siano rispettivamente una e due posizioni prima di quello di x .

```

        LDR    R0, R5, #0      ; carica x in R0
                                ; R0 <- mem[R5]
        BRz    SALTA          ; se x=0 va a salta

        LDR    R0, R5, #-1     ; carica y in R0
                                ; R0 <- mem[R5-1]
        ADD    R0, R0, #1
        STR    R0, R5, #-1     ; y++
                                ; mem[R5-1] <- R0

        LDR    R0, R5, #-2     ; carica z in R0
                                ; R0 <- mem[R5-2]
        ADD    R0, R0, #-1
        STR    R0, R5, #-2     ; z--
                                ; mem[R5-2] <- R0
        BR     Fine

SALTA:   LDR    R0, R5, #-1     ; carica y in R0
                                ; R0 <- mem[R5-1]
        ADD    R0, R0, #-1
        STR    R0, R5, #-1     ; y--
                                ; mem[R5-1] <- R0

        LDR    R0, R5, #-2     ; carica z in R0
                                ; R0 <- mem[R5-2]
        ADD    R0, R0, #1
        STR    R0, R5, #-2     ; z++
                                ; mem[R5-2] <- R0
Fine:    ...
        ...
```

Nel commento le istruzioni LDR e STR (rispettivamente lettura e scrittura della memoria) devono essere spiegate esplicitamente distinguendo l'indirizzo

e il contenuto.

Corrisponde al seguente codice C

```
if (x) {  
    y++;  
    z--;  
}  
else {  
    y--;  
    z++;  
}
```