

CORSO DI LAUREA IN INGEGNERIA INFORMATICA

Calcolatori Elettronici — a.a. 2009–2010

**Compito del 12 gennaio 2010 (9 crediti – DM 270)**

Cognome e Nome dello studente: \_\_\_\_\_

**1/** Utilizzando le proprietà dell'algebra booleana, dimostrare che si ha overflow nella somma di due interi con segno espressi in complemento a due  $(a_{k-1}a_{k-2} \dots a_2a_1a_0)_{C_2}$  e  $(b_{k-1}b_{k-2} \dots b_2b_1b_0)_{C_2}$  se e solo se

$$c_{k-1} \neq c_{k-2} \quad ,$$

dove  $c_{k-2}$  e  $c_{k-1}$  sono rispettivamente i riporti all'ingresso e all'uscita dell'ultimo full-adder di un sommatore *ripple carry*.

**2/** Siano date le istruzioni assembly 8086

1. mov AX,VETTORE[BX]
2. add [DI],35
3. loop CICLO
4. push VAR

Indicare per ciascuna di esse una codifica di macchina plausibile, il numero di cicli di bus (distinguendo tra fase di fetch e di esecuzione) e il numero approssimativo di cicli di macchina (fase di esecuzione).

**3/** Scrivere in assembly 8086 una subroutine **SORT\_ARRAY** di tipo NEAR che ordini in modo crescente un array di interi positivi rappresentati su 16 bit. I parametri (puntatore all'array e numero di elementi) siano passati attraverso lo stack.

CORSO DI LAUREA IN INGEGNERIA INFORMATICA

Calcolatori Elettronici — a.a. 2009–2010

**Compito del 12 gennaio 2010 (6 crediti – DM 509)**

Cognome e Nome dello studente:

---

**a/** L'evoluzione dello stato di una macchina sequenziale  $M$  con un ingresso  $x$  e due bit di stato  $y_1$  e  $y_0$  è descritta dalla seguente coppia di equazioni logiche:

$$\begin{aligned}y_1'(x; y_1, y_0) &= x \oplus y_0 \quad ; \\y_0'(x; y_1, y_0) &= y_1 + y_0 \quad .\end{aligned}$$

- a) Riscrivere le equazioni nella forma canonica “somma di prodotti”;
- b) Disegnare il diagramma degli stati della macchina;
- c) Disegnare il circuito usando porte logiche e flip-flop D;
- d) Disegnare il diagramma degli stati della macchina  $M'$  che si otterrebbe sostituendo i flip-flop D del punto precedente con flip-flop JK, aventi come ingressi  $J_1 = K_1 = D_1$ ,  $J_0 = \overline{D_0}$ ,  $K_0 = x$ .

**b/** Scrivere in assembly 8086 una subroutine `SORT_ARRAY` che ordini in modo crescente un array di interi positivi rappresentati su 16 bit. I parametri (puntatore all'array e numero di elementi) siano passati rispettivamente attraverso i registri `BX` e `CX`.