

pagina	descrizione
14	Invertire le funzioni NAND (f7) e NOR(f1).
18	rigo 5 dove l'intervallo di i dovrebbe oscillare tra $\{1,2,\dots,k\}$ e non $\{1,2,\dots,n\}$, inoltre nella dimostrazione successiva dal rigo 14 al rigo 17 vengono indicati X_{n-1} e X_{n-2} al posto di X_{k-1} e X_{k-2}
20	secondo membro dell'equivalenza D E al posto di C E
22	Al punto 2, forma canonica congiuntiva, nell'esempio c'è il gruppo (E P !W) che è legato da operatori AND quando invece dovrebbero essere degli OR: E+R+!W
24	Cambiare AND con OR nella tabella
25	$F=(A+!B+C)(!A+B+!C)$ aggiungere negazione su C dopo p15 e P19 eliminare + in ultima riga
27	$F=AB+!AC+!B!C=(A+!B+C)(!A+B+!C)$
28	mappa di Karnaugh in figura 1.15 il secondo gruppo è CB
30	$F=(A+!B+C)(!A+B+!C)$
32	Y=!A+B+!C mancava negazione su C
34	Esercizio 1 (ultimo numero binario), i risultati sono 4555_8 e 2413_{10}
35	esercizio n. 1 di pagina 35 la F vale 1011 e non 0010
35	esercizio n.2 $F=00111111$ $H=A$ $L=00000111$ $M=01111111$
37	fig. 1.17 le mappe K di F, H, L ed M vanno cambiate in funzione della modifica all'esercizio 2 di pag 35
38	esercizio n.5 $L=(A+B+!C)*(!A+!B+C)$

38	esercizio n.5 $I = !A * !B * C + A * !B * C + A * B * !C$
39	esercizio n.7 $L = P1 + P2 + P3 + P4$ $A = P1P2 + P2P3 + P3P4$
40	C = l'indice j all'interno degli elementi di w
40	negare nella soluzione la variabile U
40	eliminare la I nel risultato dell'esercizio 8
42	$F(A,B) = O(E(A;NON(E(NON(A);B)));NON(B))$
43	Soluzione ex.3 $F(A,B,C,D) = O(E(C;D);E(A;NON(B));E(A;NON(O(B;NON(A))));E(C;NON(D)))$
46	aggiungere assegnazione $F = A!C + !X + !Y$
46	modificare 2^4 con 2^3
48	riga 8 sostituire: Per la proprietà dell'AND sui complementi (p7) e dell'OR rispetto all'elemento neutro (p11)
53	esercizio n.6 eliminare il gruppo AB nel terzo e quarto passaggio esercizio n.7 terzo passaggio, proprietà di idempotenza dell'AND (p2) al posto dell'OR
54	manca la B come membro dell'espressione nella 3a e 4a riga,
55	esercizio n.8 sesto passaggio, proprietà di idempotenza dell'AND (p2) al posto dell'OR
59	nello schema con le porte NOR manca una porta NOR che neghi B per la porta NOR a 3 ingressi in basso, in questo modo gli ingressi sono $(A+B+!C)$ anziché $(A+!B+!C)$
62	Quando applica il Teorema di De Morgan (p19) per esprimere F utilizzando solo le NAND manca la negazione alla lettera C della seconda riga dell'espressione
65	invertire posizione dell'inverter sul mux inferiore

81	risultati es. 2.3 $[A+D, !(A!D)]$
82	risultati es. 2.11 soluzione NOR una sola negazione
82	risultati es. 2.9 soluzione NAND $!(!(AB!D)!(BC!D))$
83	$H=A!B+!A B+!A C+B!C$
84	soluzione ex 5 [0011, 1011, 0100] soluzione ex 6 [1110, 1001, 0010] soluzione ex 8 [0100, 0111, 0010]
85	soluzione ex 2.2 [10010, 01111, 01000]
86	soluzione ex 2.4 [10111, 10001, 10000]
87	modificare in: ...di interfacciare l'8522A in modalità <u>memory-mapped</u> I/O a partire dagli indirizzi immediatamente precedenti allo spazio occupato dalla RAM, realizzare il circuito di decodifica ...
95	Cambiare tre con tra (numeri)
95	flag di Saturazione(Q), è asserito quando nel risultato di una operazione è stato inserito un valore limite (estremo superiore o inferiore) a causa del superamento dell'intervallo di rappresentazione del tipo (in eccesso o difetto). Per esempio, considerando il dominio dei numeri interi con segno espressi con 32 bit, l'intervallo di rappresentazione va da -2^{31} (0x80000000) a $+2^{31}-1$ (0x7FFFFFFF), perché il bit 31 è rappresentante del segno (positivo se zero e negativo se uno).
95	Nel set delle istruzioni ARM sono presenti delle istruzioni aritmetiche con saturazione che permettono di bloccare il risultato all'estremo superiore (o inferiore) asserendo il flag Q, che indica appunto la situazione di saturazione. Il seguente esempio, in codice ARM, mostra il comportamento delle diverse operazioni di addizione, senza saturazione (ADD) e con saturazione (QADD): MOV R2, #0x60000000 ; R2=0x60000000 ADD R3, R2, R2 ; R3=R2+R2 =0xC0000000 QADD R4, R2, R2 ; R4=sat(R2+R2)=0x7FFFFFFF
99	tabella 3.6 - la preconditione Sempre (AL) vale 1110
100	riga 9: "Program Statur Register" va modificato in "Program Status Register"; ultima riga: "i registri dei flag CPSR" diventa "i flag del registro CPSR";

109-110	<p>Nel caso di indirizzamento immediato si hanno a disposizione soltanto 12 bit dei 32 che costituiscono la word dell'istruzione, quindi si potrebbero rappresentare soltanto i numeri compresi nell'intervallo 0..+4095 oppure nell'intervallo -2048..+2047, se considerati con segno. Questo limite non sarebbe accettabile poichè il processore ARM, che lavora con registri che memorizzano word (cioè numeri a 32 bit), gestisce valori che hanno un campo di variazione da 0x00000000 a 0xFFFFFFFF.</p> <p>Quindi i progettisti dell'architettura ARM hanno escogitato un modo che permette di esprimere solo alcuni valori dell'intervallo sfruttando i 12 disponibili. L'idea è quella di memorizzare nei 12 bit due campi: posizione (4 bit) e valore (8 bit).</p> <p>Il numero descritto è ottenuto per traslazione, verso destra, dei bit del campo valore per un numero di posizioni pari al doppio del campo posizione (0, 2,..., 30). Per esempio, è possibile caricare un registro con il valore 0x00002040, utilizzando un indirizzamento immediato anche se all'interno dell'istruzione non ci sarebbe abbastanza spazio per scrivere i 14 bit del valore. L'assemblatore memorizzerà all'interno dell'istruzione la coppia di valori <1101_2,10000001_2> che rappresenta il numero ottenuto per traslazione del valore 10000001_2 di 26 posizioni verso destra:</p> <p><figura 3.8></p> <p>Usando questo approccio è evidente una restrizione nella rappresentazione dei numeri, infatti si possono esprimere tutti e soli i numeri che in binario hanno bit 1 a distanza massima minore o, al più, uguale ad 8 che possano essere ottenuti per traslazione in una posizione pari. Rispondono a questo criterio i valori:</p>
110	sostituire nei valori validi 408 con 208 e 102 con 204, mentre quelli non validi 804 con 408
111	<p>riga 10: "SUB R5, R3, #11; R0 = R1 - 11" dovrebbe essere "SUB R5, R3, #11; R5 = R3 - 11";</p> <p>riga 11: "RSB R2, F5" dovrebbe essere "RSB R2, R5";</p> <p>riga 3 della tabella: dovrebbe essere LSR invece di LRR.</p>
114	ultima riga sostituire Rd con Rn
116	eliminare un del
120	<p>cambiare word con halfword</p> <p>eliminare un di</p>
125	la sintassi delle istruzioni MSR devono essere cambiate da MRS a MSR

157	nella tabella al terzo rigo il mnemonico è REVSH anzichè REV16
164	Figura 4.3 far partire l'indice iniziale da 1 e non da 0
171	Fig. 4.11 eliminare blocco K=1, eliminare blocco k=k+1, far partice ciclo for da k=1
172	Fig. 4.12 eliminare blocco K=1, eliminare blocco k=k+1, far partice ciclo for da k=1
184	$V[0] \geq$, va inserito $V[0] \geq 0$
185	espressione logica: $V1[1] V2[1] + \dots + V1[n] V2[n]$ (escludere gli indici 0)
185	LSL R3,R2, #4
187	DNS eliminare blocco K=1, eliminare blocco k=k+1, far partice ciclo for da k=1
190	DNS $R5 = R5 * 4$ (e non $*5$)
214	nella tabella P4 e P5 hanno i tempi $TA(i)$ scambiati
218	ultima frase: il sistema non è sostenibile
219	$C3 = P3 * 0.3299$
221	es. 12: sostituire P4 con P5
222	Svolgimento Es.13 nella seconda bitmap sostituire la riga 3 e 4 in questo modo: 01111, 10001
224	es 15: Nel testo dell'esercizio i contatori sono a 6 bit (e non 8) eliminare i valori di $r = 111111$ e $r = 010101$
232	ex 5.1 soluzioni $Tt=5.83$ e $Tw=2$ ex 5.3 soluzioni $Tt=4$

233	ex 6.3 soluzioni [1 mu s, 10%, 1 mu s, 3.33%, Q2]
235	ex 8.4 soluzioni [80%, 40%, 15%]
236	ex 9.3 P2=50 mu s e C3= 111 mu s soluzioni [15.85%, 66.81%]
237	ex 10.1 C4=547 ns ex 10.2 soluzioni [56.99%, 3.87ms] ex 10.4 soluzioni [53.83%, 230.83 mu s]
239	Soluzioni Esercizio 11.1 [128, 65 o 75, 65 o 76] Soluzioni Esercizio 11.3 [207,67,67] Soluzioni Esercizio 11.4 [126, 70, 66]
239	es. 12.2 Soluzione [1, 8, 1] e non [1, 6, 1] es. 12.4 Soluzioni best-Fit invertite con worst-Fit
240	es. 13.3: Nel testo dell'esercizio il processo P2 ha posizione=7 (e non 2)
242	Nell'esercizio 15 sostituire i valori della tabella: 1) 101001, 000011, 111100, 010110, 100000, 000001, 101000, 111001 [P_4] 2) 000001, 000011, 111100, 110011, 100001, 011110, 111110 [P_5] 3) 100001, 001100, 111100 [P_4] 4) 001001, 000011, 111100, 000001, 100000, 000001, 101000, 011111, 011000 [P_0] 5) 101101, 111111, 100000, 000111, 000100, 010001, 111011, 000111, 000100, 010001, 110010 [P_2]
243	aggiunto esercizio 16.5 [P_1,350,0,0], [P_7,100,1,0], [P_4,300,0,1], [P_5,400,1,0],[P_2,150,0,1], [P_3,350,1,1], [P_0,100,1,0], [P_6,50,1,0] [P_7]