

Architettura dei Calcolatori

Prova scritta – 11 febbraio 2020 – 1h30

PARTE 1 – RISPOSTA SINGOLA - Ogni domanda ha una sola risposta VERA.

- Una risposta esatta fa acquisire il punteggio positivo riportato a fianco della domanda
 - Una risposta errata fa perdere il punteggio negativo riportato a fianco della domanda
 - Una risposta lasciata in bianco viene valutata 0
1. (2, -.5) Quale e' il numero massimo rappresentabile in logica complemento a due con 4 byte?
 - a) Un numero vicino a 4 miliardi
 - b) Un numero vicino a 2 miliardi
 - c) $2^{(4*10)}$
 - d) Nessuna delle precedenti risposte

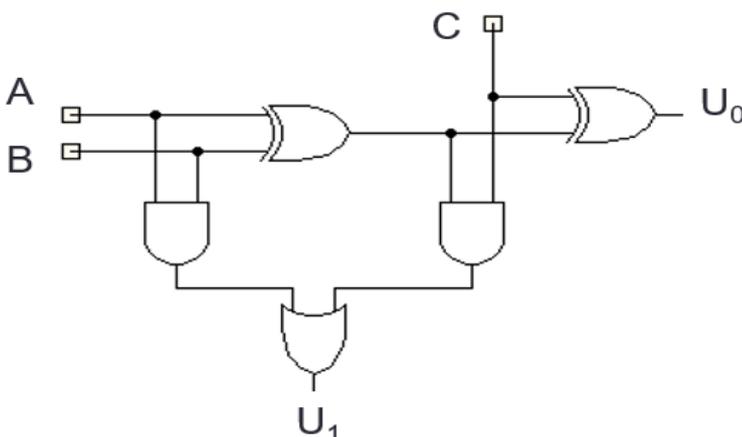
 2. (2, -.5) Assumendo interi a 8 bit, la rappresentazione in complemento a due del numero decimale -12 e':
 - a) 10001100
 - b) 11110011
 - c) 11110110
 - d) Nessuna delle precedenti risposte

 3. (2, -.5) Nel formato IEEE 754, la notazione denormalizzata consente di:
 - a) Decidere arbitrariamente il numero di bit da dedicare alla mantissa e all'esponente
 - b) Ridurre la perdita di precisione in caso di underflow
 - c) Rappresentare il numero infinito
 - d) Nessuna delle precedenti risposte

PARTE 2 – (POSSIBILI) RISPOSTE MULTIPLE -

Ogni domanda può avere da una a quattro risposte CORRETTE.

- Ogni risposta esatta viene calcolata: +1
 - Ogni risposta errata viene calcolata: -0.5
 - Una risposta lasciata in bianco viene calcolata: 0
4. In caso di chiamata di una funzione foglia (i.e., che non chiama altre funzioni):
- a) È possibile evitare di salvare sullo stack l'indirizzo di ritorno del chiamante
 - b) È possibile sporcare i "saved registers" senza preoccuparsi di ripristinarli al termine della funzione chiamata
 - c) In generale, è possibile evitare di salvare sullo stack il valore degli argomenti passati dal chiamante al chiamato
 - d) È consigliabile ritornare alla funzione chiamante attraverso l'istruzione `jal x1, ProcedureLabel`
5. Nel tipico memory layout di una architettura RISC-V:
- a) Figura lo spazio per i dati (statici o dinamici), ma non per il codice
 - b) I dati statici e i dati dinamici sono entrambi allocati nello spazio riservato allo heap
 - c) Stack e memoria dinamica crescono in direzioni opposte contendendosi lo stesso spazio di memoria
 - d) Si trovano indirizzi puntati da frame pointer e stack pointer
6. Per una cache set associative a 16 vie:
- a) Ogni locazione della cache ha un campo per l'indice di 5 bit
 - b) Richiede più circuiteria di comparazione rispetto a una cache fully associative
 - c) Permette in genere di diminuire il numero dei miss rispetto a una cache direct mapped
 - d) Richiede un numero di comparatori 16 volte maggiore rispetto a una cache direct mapped
7. Si consideri la seguente rete logica:



Quali affermazioni sono vere?

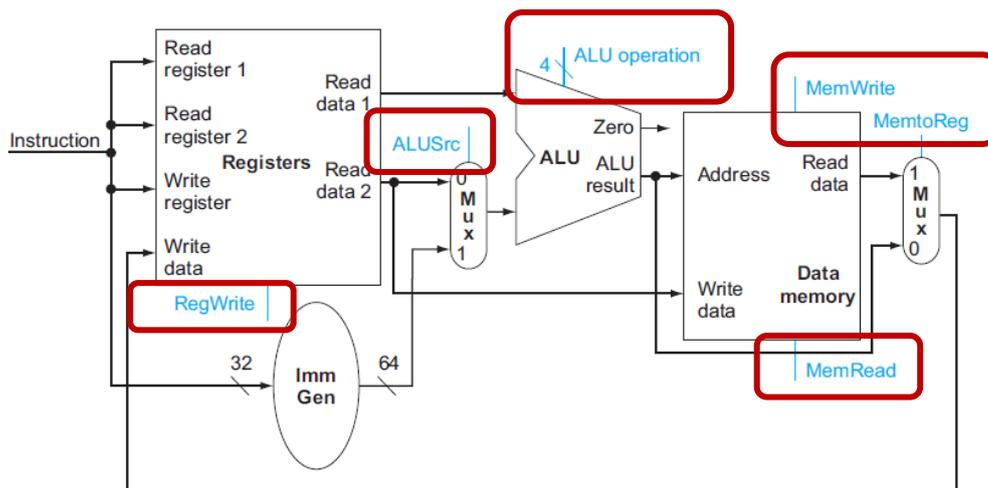
- a) Rappresenta un half adder dove non viene riportato in uscita il valore del carry-out
- b) Rappresenta un full adder con valore del carry-out sull'uscita in alto
- c) Eseguisce l'addizione tra i bit A, B e C, fornendo il valore di uscita in U1U0
- d) Non è un componente combinatorio notevole

PARTE 3 – DOMANDE APERTE

- Una risposta esatta fa acquisire il punteggio positivo riportato a fianco della domanda
- Una risposta errata può eventualmente causare una penalità che dipende dalla gravità dell'errore
- Una risposta lasciata in bianco viene calcolata: 0
- SI RICORDA CHE L'UNICO FOGLIO DA CONSEGNARE E' IN CALCE AL COMPITO. QUESTO FOGLIO, PUO' SERVIRE ESCLUSIVAMENTE COME "BRUTTA COPIA". EVENTUALI RISPOSTE SCRITTE IN QUESTO FOGLIO NON VERRANNO PRESE IN CONSIDERAZIONE

8. (6 pt) Si consideri la seguente istruzione assembly RISC-V:

and rd, rs1, rs2



ALU control lines	Function
0000	AND
0001	OR
0010	add
0110	subtract

a. Si dica che valori assumono i segnali di controllo generati per questa istruzione (cerchiati in figura) [2 pt]

- Regwrite = 1
- ALUSrc = 0
- ALU operation = 0000
- MemWrite = 0
- MemRead = 0
- MemtoReg = 0

b. quali blocchi funzionali del datapath effettuano lavoro utile per questa istruzione? [2 pt]

Register file, ALU

c. quali blocchi funzionali del datapath NON producono output per questa istruzione o

producono un output che non è utilizzato? [2 pt]

ImmGen, DataMem

9. (6 pt) Si consideri il seguente programma C

```
clear1 (int array[], int size) {
    int i;
    for (i= 0; i< size; i+= 1)
        array[i] = 0;
}
```

a. Si scriva il corrispondente programma assembly RISC-V, assumendo che gli argomenti vengano forniti in x10 e x11. [3 pt]

b. Si consideri un datapath privo di logica di rilevamento degli hazard, di stallo e di forwarding. Si modifichi manualmente il programma perché esegua correttamente su tale datapath [3 pt]

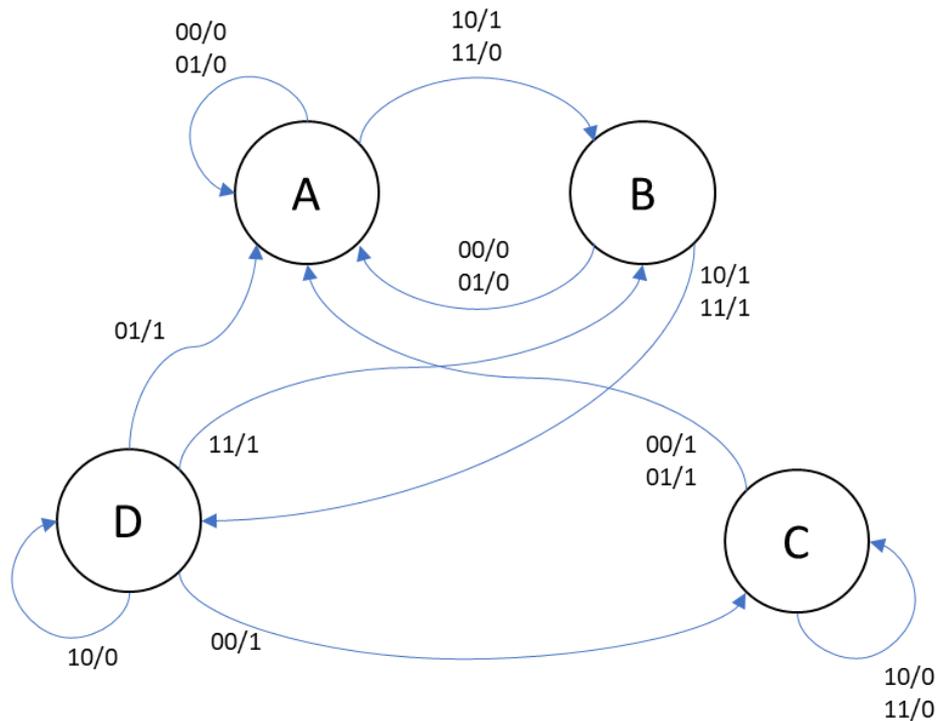
a.

```
li x5,0 // i= 0
loop1:
    slli x6,x5,3 // x6 = i * 8
    add x7,x10,x6 // x7 = address of array[i]
    sd x0,0(x7) // array[i] = 0
    addi x5,x5,1 // i= i+ 1
    blt x5,x11,loop1 // if (i<size)
// go to loop1
```

b.

li x5,0	IF	ID	EXE	MEM	WB (x5)														
nop		X	X	X	X	X													
nop			X	X	X	X	X												
slli x6,x5,3				IF	ID (x5)	EXE	MEM	WB (x6)											
nop					X	X	X	X	X										
nop						X	X	X	X	X									
add x7,x10,x6							IF	ID (x6)	EXE	MEM	WB (x7)								
nop								X	X	X	X	X							
nop									X	X	X	X	X						
sd x0,0(x7)										IF	ID (x7)	EXE	MEM	WB (x0)					
addi x5,x5,1											IF	ID (x5)	EXE	MEM	WB (x5)				
nop											X	X	X	X	X				
nop												X	X	X	X	X			
blt x5,x11,loop1														IF	ID (x5)	EXE	MEM	WB	

10. (7 pt) Si consideri il diagramma a stati di un circuito sequenziale riportato in figura:



- Dire se si tratta di un diagramma di Mealy o di Moore. Ricavare la tabella di verità. [2 pt]
- Assegnare una codifica binaria per gli stati e aggiornare la tabella di verità. [2 pt]
- Minimizzare le reti di stato futuro e delle uscite tramite mappe di Karnaugh. Disegnare il circuito finale. [3 pt]

s ₁ s ₂	S	x ₁ x ₂	S'	s' ₁ s' ₂	Z
00		00	A	00	0
00	A	01	A	00	0
00		10	B	01	1
00		11	B	01	0
01		00	A	00	0
01	B	01	A	00	0
01		10	D	11	1
01		11	D	11	1
10		00	A	00	1
10	C	01	A	00	1
10		10	C	10	0
10		11	C	10	0
11		00	C	10	1
11	D	01	A	00	1
11		10	D	11	0
11		11	B	01	1

$s_1s_2 \backslash x_1x_2$	00	01	11	10
00	0	0	0	0
01	0	0	1	1
11	0	0	1	1
10	0	0	1	1

$$s'_1 = s_1x_1 + s_2x_1 = x_1(s_1 + s_2)$$

$s_1s_2 \backslash x_1x_2$	00	01	11	10
00	0	0	1	1
01	0	0	1	1
11	0	0	1	1
10	0	0	0	0

$$s'_2 = s'_1x_1 + s_2x_1 = x_1(s'_1 + s_2)$$

$s_1s_2 \backslash x_1x_2$	00	01	11	10
00	0	0	0	1
01	0	0	1	1
11	1	1	1	0
10	1	1	0	0

$$Z = s'_1x_1x'_2 + s_2x_1x_2 + s_1x'_1$$

Architettura dei Calcolatori

Prova scritta – 11 febbraio 2020 – 1h30

Nome: _____ Cognome: _____

Matricola: _____

Indicare le risposte corrette apponendo una croce nella casella corrispondente. Per superare la prova bisogna aver raggiunto almeno 9 punti nelle domande a risposta singola/multipla, ed almeno 15 complessivamente. Questa pagina e la successiva sono una copia di quelle da consegnare, utile per tenere traccia delle risposte da voi date e calcolare il voto ottenuto durante la correzione.

	Risposte				Punti/ Penalità	
	A	B	C	D		
1					2	-0.5
2					2	-0.5
3					2	-0.5
4						
5						
6						
7						

Risposta alla domanda 8 (6 pt):

Risposta alla domanda 9 (6 pt):

Risposta alla domanda 10 (7 pt):

Architettura dei Calcolatori

Prova scritta – 11 febbraio 2020 – 1h30

Nome: _____ Cognome: _____

Matricola: _____

Indicare le risposte corrette apponendo una croce nella casella corrispondente. Per superare la prova bisogna aver raggiunto almeno 9 punti nelle domande a risposta singola/multipla, ed almeno 15 complessivamente. Questa pagina e la successiva sono le uniche pagine che dovete consegnare. Per comodità avete anche una copia di questa pagina per calcolare il voto da voi ottenuto durante la correzione.

	Risposte				Punti/ Penalità	
	A	B	C	D		
1					2	-0.5
2					2	-0.5
3					2	-0.5
4						
5						
6						
7						

Risposta alla domanda 8 (6 pt):

Risposta alla domanda 9 (6 pt):

Risposta alla domanda 10 (7 pt):