40001 Intro to Computer Systems 2021 Exam Sample Solution

Disclaimer: This is by no means an official answer key, please correct any mistakes you find and keep in mind there could be several possible solutions

Section A

1 a. $(A' \bullet (B + A)) + (B' \bullet (A + A'))$

$\equiv \quad (A' \bullet (B + A)) + (B' \bullet 1)$ **[$A + A' = 1$]**

$\equiv \quad (A' \bullet (B + A)) + B'$ **[$A \bullet 1 = A$]**

$\equiv \quad (A' \bullet B + A' \bullet A) + B'$ **[Distributivity of •]**

$\equiv \quad (A' \bullet B + 0) + B'$ **[$A \bullet A' = 0$]**

$\equiv \quad (A' \bullet B) + B'$ **[$A + 0 = A$]**

$\equiv \quad (B' + A') \bullet (B + B')$ **[Distributivity of +]**

$\equiv \quad (B' + A') \bullet 1$ **[$A + A' = 1$]**

$\equiv \quad (B' + A')$ **[$A \bullet 1 = A$]**

$\equiv \quad (A \bullet B)'$ **[De Morgan's Law]**

b. i)

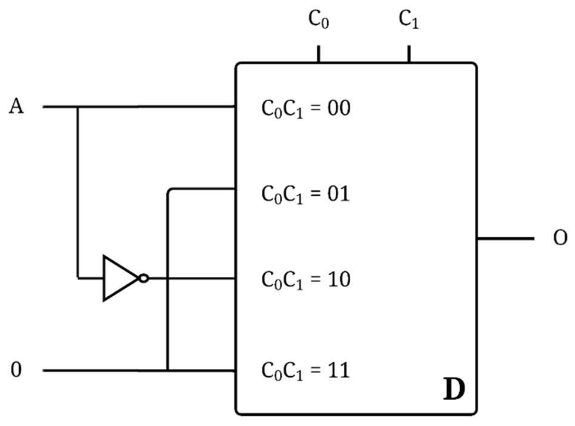
A	C_0	C_1	O	Minterm
0	0	0	0	
0	0	1	0	
0	1	0	1	$A' \bullet C_0 \bullet C_1'$
0	1	1	0	
1	0	0	1	$A \bullet C_0' \bullet C_1'$
1	0	1	0	
1	1	0	0	
1	1	1	0	

Canonical Minterm Form:

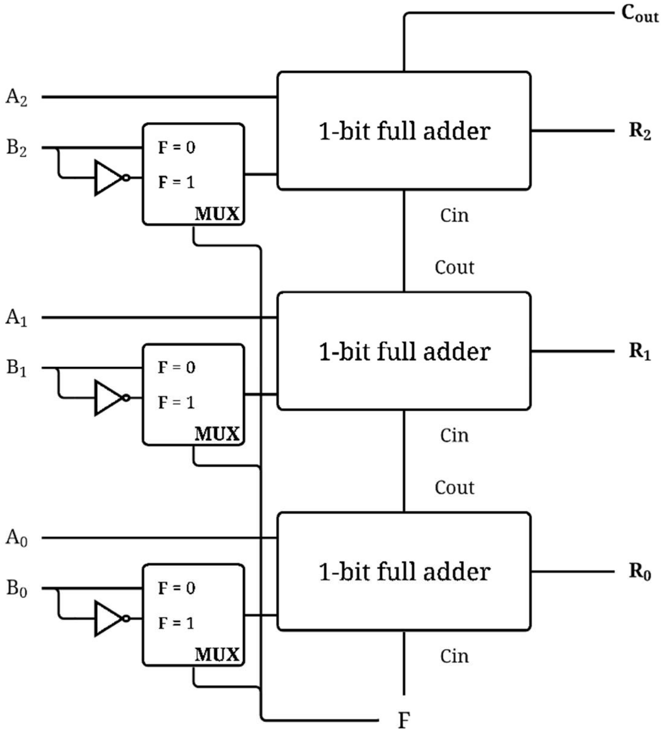
$(A' \bullet C_0 \bullet C_1') + (A \bullet C_0' \bullet C_1')$

ii) 3 NOT gates, 4 AND gates and 1 OR gate

iii) D is a 4-to-1 multiplexer:



c



d. i) 0xCAFE0000 = 1100 1010 1111 1110 0000 0000 0000 0000

Sign Bit: 1 = Negative

Exponent: 1001 0101 = 149 – 127 = 22

Mantissa: 111 1110 0000 0000 0000 0000 = 0.984375

= 1.984375 (Restore hidden bit)

= -1.984375×2^{22}

ii) = -8.323072×10^{6}

Section B

2 a.

Instruction	Cycle	Transfers	Path
ADD Rdest, Rsrc	E1	A ← Rsrc	
	E2	B ← Rdest	
	E3	Rdest ← ALUres; C ← ALUcout	ALU=A+B; Cin=0
COMPARE Rdest, Rsrc	E1	A ← Rsrc	
	E2	B ← Rdest	
	E3	C ← ALUcout	ALU=A-B; Cin=0

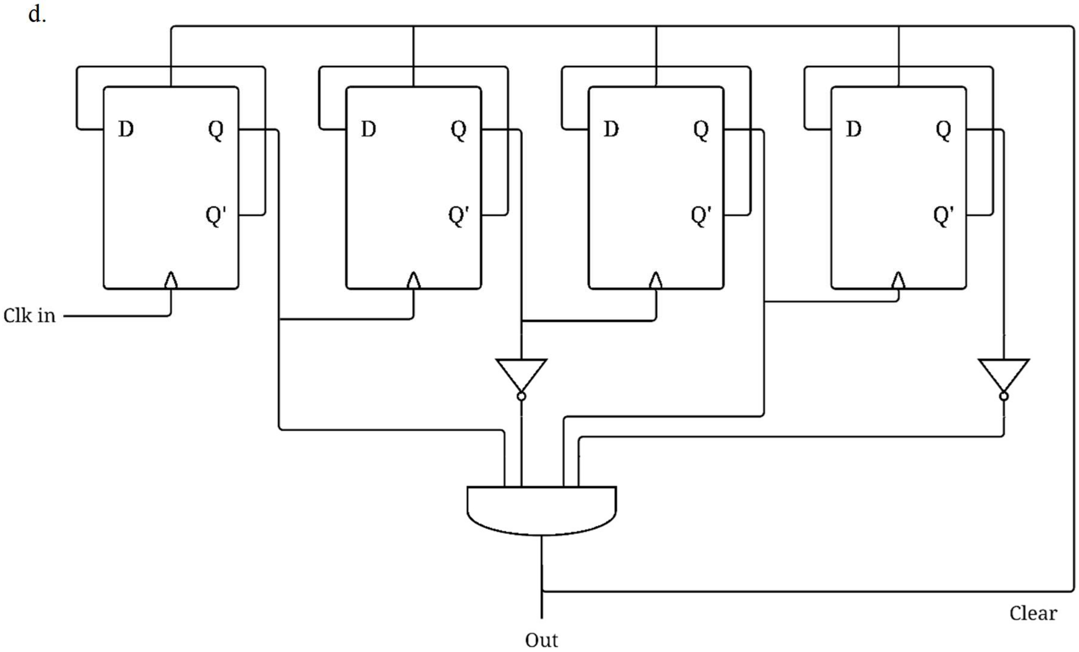
b. SUBTRACT, AND, OR and XOR are all two-register instructions that are very similar to ADD, just with different ALU settings c.

Instruction	Cycle	Transfers	Path
SKIP	E1	PC ← PC + 1	

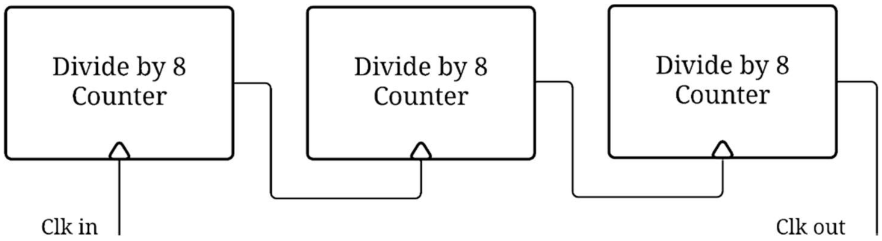
The COMPARE instruction is used by SKIP, as the carry register is updated after COMPARE is executed which is then used as an input to the controller

The NOP instruction requires 0 cycles and has no register transfers. It could be usefully applied for timing purposes, such as producing a delay for an idle processor

d.



e.



1/512 = 1/8 * 1/8 * 1/8, with the resulting circuit being asynchronous