



DATE: 05/12/2023

IA TEST: I

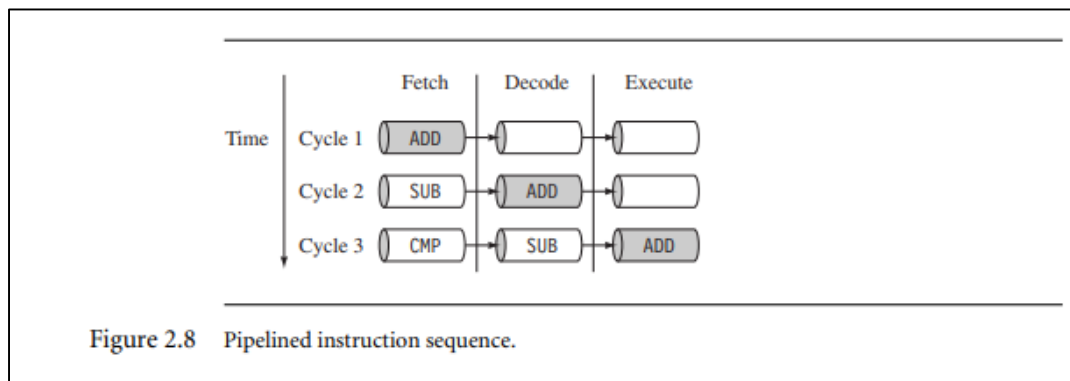
MARKS

SUBJECT: MICROCONTROLLERS AND EMBEDDED SYSTEMS

SUBJECT CODE: 21CS53

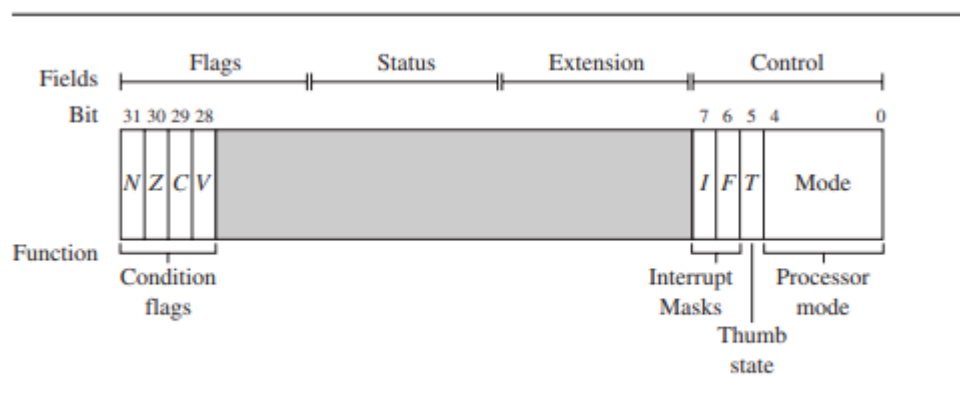
PART A

1. A pipeline is the mechanism a RISC processor uses to execute instructions. Using a pipeline speeds up execution by fetching the next instruction while other instructions are being decoded and executed.



Explanation

2.

A generic program status register (*psr*).

- SUBS R0, R1, R1; Zero flag is affected as R1-R1 is zero
- CMP R2, R3; R2=0x20 and R3=0x50; Negative flag is affected as R2-R3 is

1M+1M

		negative	
PART B			
3.		<pre> AREA TWO, CODE, READONLY ENTRY MOV R0, #5 LDR R1, = FBLOCK LDR R2, = SBLOCK GOTO LDRH R3,[R1],#2 STRH R3,[R2],#2 SUBS R0,#1 BNE GOTO L B L FBLOCK DCW 0X1234,0X4567,0X7865,0X6633,0X1987 AREA MYDATA,DATA, READWRITE SBLOCK DCW 0 END ; Mark end of file </pre>	5 M
4.		<pre> 1 AREA NEW, CODE, READONLY 2 ENTRY 3 MOV R0, #1 4 MOV R1, #1 5 MOV R2, #1 6 MOV R3, #1 7 MOV R4, #6 8 LOOP MLA R5, R0, R1, R2 9 ADD R3, R3, R5 10 SUBS R4, #1 11 BNE LOOP 12 L B L 13 END </pre>	5 M
5.		<pre> AREA SIX, CODE, READONLY ENTRY MOV R1, #0X12 MOV R2, #0X00 ORR R0, R1, R2 AND R0, R1, R2 EOR R0, R1, R2 L B L END </pre>	5 M
6.		<pre> AREA FOUR, CODE, READONLY ENTRY MOV R0, #7 MOV R1, R0 ; Mark first instruction to execute ; STORE FACTORIAL NUMBER IN R0 ; MOVE THE SAME NUMBER IN R1 FACT SUBS R1, R1, #1 CMP R1, #1 BEQ L MUL R3, R0, R1; MOV R0, R3 BNE FACT L B L END ; SUBTRACTION ; COMPARISON ; MULTIPLICATION ; Result ; BRANCH TO THE LOOP IF NOT EQUAL ; Mark end of file </pre>	5 M

PART C

7.	<pre> 1 AREA T1, CODE, READONLY 2 ENTRY 3 MOV R0, #0XC0000003 ; Content of R0=? 4 MOV R1, #0XE0000001 ; Content of R1=? 5 MOVS R2, R0,ROR #1 ; Content of R2=? R0=? 6 ; status of NZCV? 7 CMP R2,R1 ;Content of R2=? R1=? 8 ; status of NZCV? 9 L B L 10 END </pre> <p> R0=0XC0000003 R1=0XE0000001 R2=0XE0000001, R0=0XC0000003, N=1, Z=0, C=1, V=0 R2=0XE0000001, R1=0XE0000001, N=0, Z=1, C=1,V=0 </p>	<p>0.5 M</p> <p>0.5 M</p> <p>2 M</p> <p>2 M</p>
8.	<pre> 1 AREA T31, CODE, READONLY 2 ENTRY 3 MOV R0, #0XF0000007 ; content of R0=? 4 MOV R1, #0XE0000008 ; content of R1=? 5 AND R2,R0,R1 ; content of R2=? 6 ORR R3,R0,R1 ; content of R3=? 7 EOR R4,R0,R0 ; content of R4=? 8 TEQ R4, #00 ; content of R4=? 9 ; comment on the status of N, Z, C, V flag bits 10 L B L 11 END </pre> <p> R0= 0XF0000007 R1=0XE0000008 R2=0XE0000000 R3=0XF000000F R4=0X00000000 R4=0X00000000, N=0,Z=1,C=0,V=0 </p>	<p>0.5 M</p> <p>0.5 M</p> <p>0.5 M</p> <p>0.5 M</p> <p>0.5 M</p> <p>2.5 M</p>