ORG test 2 13 Nov 85 by _____

1-4 10 pts each 5-8 15 pts each Good Luck!

A 1010
B 1011
C 1100
D 1101
E 1110
F 1111

1. In an 8-bit machine $3C_{16}$ is subtracted from BD_{16}. Give the contents of the flags C, S, V, Z and PE (parity even) after this operation.

$= 0011\ 1100$

$$\begin{array}{r} {\scriptstyle 1\ 1\ 1\ 1\ 1} \\ 1011\ 1101 \\ 1100\ 0100 \\ \hline 1\ 0000001 \end{array}$$

C = 1 S = 1 V = 0 Z = 0 PE = 0

2. AB. Formulate a mapping process from (macro) op-codes to (μ) control memory addresses given that μ-memory has 1K words, op-codes are 6 bits wide and 4 control words are needed for each macro-op

00 X X X X X X 00

C. Illustrate your map with the op-code 42_{10}, give your address in binary

00 101010 00

D. What is the size of the CAR? 10

3. AB. The text divides all interrupts into 3 classes. Give these 3 classes and an example of each.

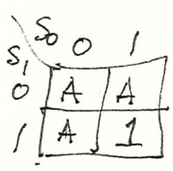
ext (asyc) I/O input
int (trap) ÷0
software (sys call) output

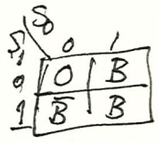
C. What is an interrupt vector and how is it used?

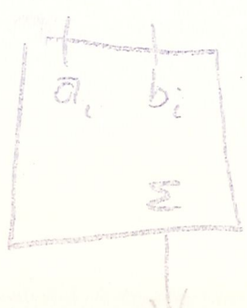
an address in memory assoc w/ part interrupt
either loc of interrupt routine or pointer to same

4. Design a circuit for ALU which yields (Actually we need only one "stage" of the logic which is "fed" into a parallel adder.) The decoder will ease the pain

S_1S_0	$C_{in}=0$	$C_{in}=1$
00	A	A+1
01	A+B	A+B+1
10	A+$\overline{B}$	A+$\overline{B}$+1
11	B−1	B







```
ORG 36
NOP  I  CALL INDIRECT
DECSP, PCTBR, BRTPC  U JMP NEXT
SPTAR      U JMP NEXT
WRITE      U JMP FETCH
```

```
ORG 48
SPTAR      U JMP NEXT
INCSP, READ  U JMP NEXT
BRTPC      U JMP FETCH
```

$R3 \leftarrow x$

$M[x] \leftarrow M[R2]$

$M[R2-1] \leftarrow M[PC+x]$ & $R2 \leftarrow R2-1$

$M[M[x]] \leftarrow M[R0]$ & $R0 \leftarrow R0+1$

$M[R3+y] \leftarrow M[R2+x]$

(μ-ops) ORG/T2/FPS

7. This version of the chapter 5 computer has both an SP (points to TOS, grows toward low memory) and an FP (a "frame pointer"). The LINK ass-instruction reserves some stack space (for subroutine variables). The amount of space is given by the address field (it's a memory reference instruction) It does PUSH FP, FP←SP, SP←SP+constant. UNLINK undoes this (but it doesn't know the "constant" (nor does it need it)) Write the execution cycles for these with controls given they have "op-codes" r & s respectively. (Additional Adders are available)

A. LINK

$c_2 rt_0$: SP ← SP−1, MBR ← FP

t_1: MAR ← SP, FP ← SP

t_2: M ← MBR, SP ← SP + constant (where)

t_3: change states

B UNLINK

$c_2 st_0$: SP ← FP, MAR ← FP

t_1: MBR ← M, SP ← SP+1

t_2: FP ← MBR

t_3: change states

SP ← SP−1

MAR ← SP, MBR ← FP, FP ← SP, SP ← SP + MBR(AD)

M ← MBR

8. (parameter passing via the stack) [SP points to TOS, grows toward low memory]

A compiler would generate the code

```
push 10
push A
push 1
call P
```

for a procedure call P(1, A, 10) if the procedure body is as in the box.

```
Procedure P (A, B, C: integer
[local variables]
begin
 .
 .
end
```

A. Suppose integers and addresses fit into 1 word of memory. At the begining of P the variables A, B, C are easy to obtain via using the SP as an index register. what are the offsets from SP for Var A? +1 Var B? +2 Var C? +3

B. If Procedure P has local variables it will call LINK (Prob 7) to create stack space for them. The variables A, B, C are still easy to obtain but this time via indexing from the FP. What are the offsets from FP for Var A? +2 Var B? +3 Var C? +4

CDE. This stack use allows for procedures to be called with a different number of parameters each time it is called. We push the variables onto the stack in backwards order (right to left) then push the size of the variables onto the stack (i.e. reals take 2 words) then push the number of variables onto the stack and call the routine. Like writeln(integer, real, integer)

C. what is the offset of the first variable from SP in general?

M[SP] +1 +1

D. what is the offset of the second variable from SP?

M[SP] + M[SP+1] +1+1

E. the offset for 3rd

M[SP] + M[SP+1] + M[SP+2] +1+1

ORG test 2 13 Nov 85 by _____

1-4 10 pts each 5-8 15 pts each Good Luck!

1. In an 8-bit machine $3C_{16}$ is subtracted from BD_{16}. Give the contents of the flags C, S, V, Z and PE (parity even) after this operation.

50
56
24
4
1
135

135/180 ≈ 75.0%

5/0/7/0/4/0/1/0/0/1/0

2. AB. Formulate a mapping process from (macro) op-codes to (μ) control memory addresses given that μ-memory has 1K words, op-codes are 6 bits wide and 4 control words are needed for each macro-op

140
8
6
4
2
160

88.9%

14/0/1/0/1/0/1/0/1/0/0

C. Illustrate your map with the op-code of 42_{10}, give your address in binary

D. What is the size of the CAR?

3. AB. The text divides all interrupts into 3 classes. Give those 3 classes and an example of each.

100
18
8
14
12
4
156

86.7%

10/2/1/2/2/0/1/0→

C. What is an interrupt vector and how is it used?

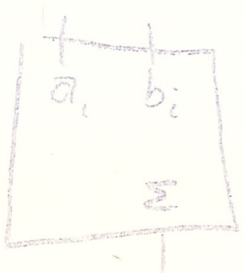
4. Design a circuit for ALU which yields (Actually we need only one "stage" of the logic which is "fed" into a parallel adder.) The decoder will ease the pain

S_1S_0	$C_{in}=0$	$C_{in}=1$
00	A	A+1
01	A+B	A+B+1
10	A+$\overline{B}$	A+$\overline{B}$+1
11	B−1	B

8/2/2/0/2/0/0/0/3/0/1

80
18
16
12
6
0
152

73.3%



2/0/0/2/0/3/0/1/4/1/2/0/1/2/0/0

180
90

30
24
30
8
28
6
10
3
4
143

$\frac{143}{270} \approx 53.0\%$

15
14
24
22
30
8
7
6
5
16
2
149

1/1/0/2/2/3/0/1/1/1/1/4/0/1/0/0

55.2%

0/0/0/4/2/1/2/1/4/0/1/0/0/1/2/0

48
22
10
18
8
8
28
5
2
2

143

53.0%

0/0/2/2/3/1/2/0/1/3/2/1/0/1/0/0

26
24
33
10
18
7
18
10
4
2

152

$\frac{9}{270}$ $\frac{1}{30}$

$30\overline{)1.00}$ 033
90

56.3%