

EE371 Second Semester Test - Thursday October 27, 2005
45 points, 16.67% of Final Grade

Please put your name on the outside of the paper also. Name _____key_____

1. Assume P, Q, and R are **8-bit unsigned integer** variables in 8-bit memory locations named P, Q, and R. Insert assembly language code into the following design statements. Assume P, Q and R are initialized in some other part of the program. (10 points)

```
;IF P >= Q
    ldaa P
    cmpa Q
    blo  elsepart
; THEN R = Q + R
    ldaa Q
    adda R
    staa R
    bra  endif
; ELSE R = 2 * R
elsepart:
    ldaa R      or can do ASL R
    adda R
    staa R
; ENDIF P >= Q
endif:
```

2. Assume P, Q and R are **16-bit signed integer** data in memory locations named P, Q and R.

- a. Show how to allocate memory storage locations for these variables. (3 points)

```
P: DS.B 2
Q: DS.W 1
R: DS.W 2
```

- b. Insert assembly language code into the following design statements. Assume P, Q and R are initialized in some other part of the program: (10 points)

```
;IF Q > P
    ldd Q
    cpd P
    ble elsepart
; THEN P = P + 1
    ldx P
    inx
    stx P
    bra endif
; ELSE P = Q + R
elsepart:
    addd R
    std P
; ENDIF Q > P
endif:
```

3. Why do we insist that a subroutine NOT modify registers unless a register is used to return a value to the calling program? (3 points)

4. Assume accumulator A has \$55, the contents of memory location DATA = \$22 and the instruction CMPA DATA is executed. Fill in the following answering Yes or No whether or not the conditional branch instruction is taken. (5 points)

	BGE	BLE	BGT	BLT	BEQ	BNE
Yes or No ->	Yes	No	Yes	No	No	Yes
	BHS	BLS	BHI	BLO		
Yes or No ->	Yes	No	Yes	No		

5. For the following program, give the results in the affected registers for each program step. (14 points)

Assume the program is located starting at memory location \$C000

```

6      6      ;*****
7      7      ; Constants definitions
8      8      0000 0042 DATA1: EQU 66
9      9      0000 0048 DATA2: EQU DATA1+6
10     10     ;*****
11     11     ; Code Section
12     12     MyCode: SECTION
13     13     Entry:
14     14     main:
15     15     ;*****
16     16     ; Initialize stack pointer register
17     17     000000 CFxx xx      lds  #__SEG_END_SSTACK
18     18     ;*****
19     19     000003 8642          ldaa #DATA1          ; A =  $42
20     20     000005 C648          ldab #DATA2          ; B =  $48
21     21     000007 C606          ldab #(DATA2-DATA1) ; B =  $06
22     22     000009 CE00 42       ldx  #DATA1          ; X =  $0042
23     23     00000C CDxx xx       ldy  #DATA3          ; Y =  $C016
24     24     00000F FExx xx       ldx  DATA3          ; X =  $1122
25     25     000012 B6xx xx       ldaa DATA3          ; A =  $11
26     26     000015 A7           nop
27     27     MyConst:SECTION
28     28     ; Place constant data here
29     29     ;*****
30     30     000000 1122 33       DATA3: DC.B $11,$22,$33

```