Microprocessors Fall 2005

Homework 7
Due Wednesday, November 2

THE DUART

1. Explain why the output register of the DUART was designed with the separate OPR_SET and OPR_CLR instead of just one address that stores the output bits?

2. Write code to set bits 0-2 of the DUART output register to %101 without affecting bits 3-7.

3. Explain the difference between serial and parallel communication.

4. Suppose memory contains:
`$9000: 31 A0 73 82 BE FF 00`
Suppose D0 contains $1234 and A0 contains $9000.
a. What are the contents of memory after the instruction `MOVEP.W D0,3(A0)`
b. What are the contents of D0 after the instruction `MOVEP.L 0(A0),D0`

POSITION-INDEPENDENT CODE

Read 2.4.10, 2.4.11, 4.8
Do Chapter 2 # 30df, 19

1. Why is it desirable to write position-independent code?

2. Compare JSR, BSR, and TRAP as methods to call a subroutine.

3. The code below reads hex characters from the keyboard and outputs to a 7 segment LED connected to the 68KMB output port. Find four places where the code fails to be position independent and fix them.

```
DUART     EQU       $00C001      ;68681 base address
IPR       EQU       13*2         ;input port register
OPR_SET   EQU       14*2         ;set bit command reg.
OPR_CLR   EQU       15*2         ;clear bit command reg.
ISHEX     EQU       $0009B2
ATOH      EQU       $000982

          ORG       $8000
LED7      MOVEA.L   #DUART,A0    ;A0 points to 68681
LOOP      TRAP      #0           ;read key
          BSR       ISHEX        ;is it a hex digit?
          BCC       LOOP         ;no.. try again
          JSR       ATOH         ;convert to nibble
          JSR       CONVERT      ;convert to 7-seg code
          BSR       OUT681       ;update LEDs
          JMP       LOOP         ;repeat

* convert a nibble to a 7-segment code *
CONVERT   MOVE.L    #TABLE,A1
          MOVE.B    (A1,D0),D0   ;table look-up
          RTS
TABLE     DC.B      $7E,$30,$6D,$79,$33,$5B,$5F,$70   ;7 seg led
          DC.B      $7F,$73,$77,$1F,$4E,$3D,$4F,$47   ;patterns

* OUT681 - output data to 68681 output port
OUT681    SET.B     OPR_CLR(A0)           ;clear all bits
          MOVE.B    D0,OPR_SET(A0)        ;set bits from D0
          RTS
```