CE2800
Lab week 3

Basic output and looping techniques

This lab session introduces the student to basic output and looping techniques.

Fibonacci Numbers are defined by the following definition:
Fib(1)=1
Fib(2)=1
Fib(n)=Fib(n-1)+Fib(n-2) for n>=3
i.e Fibonacci sequence is 1, 1, 2, 3, 5, 8, 13, 21...

Write a program which calculates the first 10 Fibonacci numbers and displays them on the LED display. The calculations must be done in a loop. The LED display should be connected to PORTB. Below you will find a portion of the program. This is intended to get you started.

```
.NOLIST
#include <avr/io.h>
.LIST
.segment .bss
fib:
      .byte 10
.segment .code
start:
; The following four lines are needed for use with subroutines. This
; will be discussed in an upcoming lecture. Just include for now.
      ldi r16, hi8(RAMEND-0x20)
      out SPH, r16
      ldi r16, lo8(RAMEND-0x20)
      out SPL, r16
      ldi r16, 0xff     ; Program PORTB for use as an output port
      out DDRB, r16
      ldi r17, 8        ; r17 is used as the loop counter
      ldi XH, hi8(fib)
      ldi XL, lo8(fib)  ; X now points to where the Fib nos. will be
;stored
      ldi r16, 1
      st X+, r16
      st X+, r16        ; Store the first two fib nos. into SRAM
      mov r1, r16       ; r1 contains fib(n-1)
      mov r2, r16       ; r2 contains fib(n-2)
loop:
```

Use r3 to hold fib(n). R1 and r2 should always contain the two previous Fibonacci numbers. As you calculate the new Fibonacci number, store it into SRAM. Use r17 as your loop counter. After all 10 Fibonacci numbers have been calculated and stored to SRAM, read them back from SRAM one at a time and send them to PORTB (This is done by using the instruction **out PORTB, r3.** This of course assumes that r3 contains the number**)**. This should also be done in a loop. Do not forget to insert a delay after sending the number to PORTB.

A waiting loop between each output ensures enough delay to allow the numbers to be viewed. This loop is provided below. Include work to calculate the duration of this loop starting at the label L1: and ending at the instruction **brne L1**.

```
delay:
      push r16
      push r17
      push r18
      ldi r16, 20
L1:
      ldi r17, 255
L2:
      ldi r18, 255
L3:
      dec r18
      brne L3
      dec r17
      brne L2
      dec r16
      brne L1
      pop r18
      pop r17
      pop r16
      ret
```

This delay loop is written as a subroutine. We have not discussed this yet in lecture. The subroutine should be placed **after** your program. Following each output to PORTB, you can insert the delay simply by using the instruction, **rcall delay**.

Pre-laboratory investigation and report:

- Calculate the duration of the waiting loop provided. Assume that the Atmega32 frequency is 8MHz.
- Draw a flowchart of your entire program using Visio.
- Your program must be written before coming to lab.
- Hand-simulate your program as was done in lecture (Only the part that calculates the Fibonacci numbers).

The pre-laboratory is due at the beginning of the lab.

Laboratory Investigation:

- Debug your program.
- Execute to test the program.
- Obtain a checkout from the instructor.

Due Date: Before or at the beginning of laboratory session of the 4th week.

- A copy of the documented source program
- A copy of the flow-chart
- Checkout sheet

Checkout _____