

*** Solution for Homework 7 ***

CSC270 Homework7

Zoey Jingyi Sun

April 4, 2016

The 6800 program I got through reverse engineering is shown in the following code session.

```
-----  
;;; hw7.asm  
;;; Zoey Jingyi Sun  
;;;  
;;; This program compares the data stored in address 0001 and 0002 and  
;;; store the bigger number in address 0003 and the smaller one to  
;;; address 0004.  
;;;  
-----  
                include      Macros.asm  
  
                data        SEGMENT  
                ORG          0000 ;specifies starting address 0000  
a                DB          3 ; 3 is stored in 0000  
b                DB          4 ; 4 is stored in 0001  
c                DB          ?  
d                DB          ?  
e                DB          ?  
f                DB          ?  
                ENDS  
  
                code        SEGMENT  
                ORG          0006 ;specifies starting address 0006  
start:          LDAA         b ; ACCA = mem[0001] (3 cycles)  
                CMPA         c ; compare ACCA to mem[0002] (3 cycles)  
                BLE          to ; go to the location of "to" if AccA is  
                ;less than or equal to the value in mem[0002] (3 cycles)  
                STAA         d ; Mem[0003]<-AccA (3 cycles)  
                LDAB         c ;ACCB = mem[0002] (3 cycles)  
                STAB         e ; mem[0004]<-AccB (3 cycles)  
                JMP          next ; jump to the "next" position  
                ;(equivalent to return to the start of program) (3 cycles)  
to:             STAA         e ; mem[0004]<-AccA (3 cycles)  
                LDAB         c ; AccB = mem[0002] (3 cycles)  
                STAB         d ; mem[0003]<- AccB (3 cycles)  
next:          JMP          start ; jump back to the start of the program (3 cycles)  
  
                ENDS  
                END
```

Assume that the clock of the 6811 is 1 MHz, regardless of the unknown value c, each loop at most will go through $3*8 = 24$ cycles. Time for 1 loop is : $24 * 1 * 10^{-6} = 2.4 * 10^{-5}s$ In one second, there will be $1/(2.4 * 10^{-5}) = 41666$ cycles.

Similar Program in Pentium assembly language:

```
;;; pentium.asm
;;; Zoey Jingyi Sun
;;;
;;; Simulation of hw7 6811 program in pentium version.
;;; to assemble and run:
;;;
;;; nasm -f elf -F stabs pentium.asm
;;; ld -melf_i386 -o pentium pentium.o
;;; ./pentium
;;; -----

;;; -----
;;; data areas
;;; -----

                section .data

a      db      3
b      db      4
c      db      0      ; unkown value
d      db      0      ; unknown value
e      db      0      ; unknown value
f      db      0      ; unknown value
                        ; (randomly assigned to make sure the program compile)

;;; -----
;;; code area
;;; -----

                section .text
                global _start

_start:
                mov     eax, b
                cmp     eax, dword[c]
                jbe     other ;jump if b is below or equal to c
                mov     dword[d], eax
                mov     ebx, c
                mov     dword[e], ebx
                jmp     done
other:  mov     ebx, c
                mov     dword[d], ebx
                mov     dword[e], eax
;;; exit()
done:   mov     eax, 1
                mov     ebx, 0
                int     0x80      ; final system call
```

There are 13 lines of instructions. Assume each instruction is 1 cycle, and assume runs at 1MHz per cycle. Each loop will take $13 * 1 * 10^{-6} = 1.3 * 10^{-5}$ seconds. There will be roughly $1s / (1.3 * 10^{-5}) = 76923$ loops per second.

Using the real frequency of Pentium which is 3.2 GHz. Each cycle will take $1/3.2GHz = 3.125 * 10^{-8}$ seconds. Each loop will take $13 * 3.125 * 10^{-8} = 4.0625 * 10^{-7}$ seconds. There will be roughly $1s / (4.0625 * 10^{-7}) = 2.46 * 10^6$ loops per second.