**Tutorial 3**

**Q1.** Find the time delay for the delay subroutine shown below if the system has an AVR with a frequency of 8 MHz:

|  |  |
| --- | --- |
| **Instruction** | **MCs** |
| LDI R16, 200  BACK: LDI R18, 100  HERE: NOP   DEC R18   BRNE HERE   DEC R16   BRNE BACK | 1  1  1  1  1 or 2  1  1 or 2 |

***Ans.***  
1 MC = **1/(8MHz) = 0.125** μ**s**  
Total MCs = Time delay = **1+{(1+[(1+1+2)x100–1]+1+2)x200–1}= 80600 MCs = 10.075 ms**

**Q2.** (10 pts) Using two single bit instructions, configure Port D bit 2 (PD2) as an input without a pull-up resistor.

***Ans.***  
**CBI DDRD,2 // DDRD.2=0 meaning configure PORTD.2 as input**  
**CBI PORTD,2 // PORTD.2 = 1 use pull-up resistor**

**Q3.** (10 pts) Using two single bit instructions, configure and set to one (1) Port D bit 5 (PD5).

***Ans.***  
**SBI DDRD,5 // DDRD.5=1 meaning configure PORTD.5 as output**  
**SBI PORTD,5**

**Q4.** Write corresponding AVR instructions for each operation in following operations  
***Operation AVR instruction(s)***  
R1 ← 60

**LDI R16,60**  
**MOV R1,R16**

D[$19] ← R2 + D[$12]

**LDS R1,$12**  
**ADD R2,R1**  
**STS $19,R1**

IO[$24] ← $96

**LDI R16,$96**

**OUT $24,R16**

D[Z+11] ← $83

**LDI R16,$83**

**STD Z+11, R16**

**Q5.** Write a program to monitor bit PC3. When it is HIGH, send 0x55 to PORTD.

***Ans.***  
 **.include “M32DEF.INC”**  
 **.org 0**  
 **LDI R16, $FF**  
 **OUT DDRD, R16 ; PORTD: Output Port**  
 **CBI DDRC,3 ; PC3 is input pin**  
**AGAIN: SBIC PINC,3 ; Skip next if PC3 is clear**  
 **RJMP OVER ; Jump to OVER if PC3 is high**  
 **RJMP AGAIN**  
**OVER: LDI R16,0x55**  
 **OUT PORTD, R16 ; Send 0x55 to PORTD**  
 **HERE: RJMP HERE**

**Q6.** Write a program to get the status of PD0 and put it on PC0.

***Ans.***  
 **.include “M32DEF.INC”**  
 **.org 0**  
 **CBI DDRD,0 ; PPD0 is input pin**  
 **SBI DDRC,0 ; PC0 is output pin**  
**AGAIN: SBIC PIND,0; Skip next if PD0 is clear**  
 **RJMP OVER ; Jump to OVER if PC3 is high**  
 **CBI PORTC,0 ; PC0 = 0 if PD0 = 0**  
 **RJMP AGAIN**

**OVER: SBI PORTC,0 ; PC0 = 1 if PD0 = 1**  
 **RJMP AGAIN**