### **CPE301 - SPRING 2019**

# Design Assignment 1A

Student Name: Dan Ray Dimapilis

Student #: 5002763761

Student Email: dimapd1@unlv.nevada.edu

Primary Github address: https://github.com/dimapd1/ASN\_subs

Directory: dimapd1/ASN\_subs

### 1. COMPONENTS LIST AND CONNECTION BLOCK DIAGRAM w/ PINS

NA

### 2. INITIAL/MODIFIED/DEVELOPED CODE OF TASK 1/A

mov r0, r16

```
;AssemblerApplication1.asm
manually mul 32b by 32b to get 64b product while not using mul instruction;
; Created: 2/13/2020 6:06:59 AM
; Author: dimapd1
;r[19,18,17,16] - multiplicand
;r[23,22,21,20] - multiplier
;r[Z, Y, X, 25 24] -product
Z(H/L) = r31/30
                    Y=29/28
                                          X=27/26
.DEF ANS1 = r24
                     ;64'b product
.DEF ANS2 = r25
.DEF ANS3 = r26
.DEF ANS4 = r27
.DEF ANS5 = r28
.DEF ANS6 = r29
.DEF ANS7 = r30
.DEF ANS8 = r31
.DEF A1 = R16
                    ;32'b Multiplicand
.DEF A2 = R17
.DEF A3 = R18
.DEF A4 = R19
.DEF B1 = R20
                    ;32'b Multiplier
.DEF B2 = R21
.DEF B3 = R22
.DEF B4 = R23
  ; Ctr = R0
                 ;Lp Counter
initializations:
              ldi r16, 33
```

;Set Loop Counter to 33

```
LDI A1, LOW($190f19a0)
                                 ;load imm, low byte of ?420 420 000? decimal
   LDI A2,BYTE2($190f19a0)
                                 ;load imm, 2nd byte of ?420 420 000? decimal
   LDI A3,BYTE3($190f19a0)
                                 ;load imm, 3rd byte of ?420 420 000? decimal
   LDI A4,BYTE4($190f19a0)
                                 ;load imm, 4th byte of ?420 420 000? decimal
   LDI B1, LOW($0000208)
                                 ;load imm, low byte of 520 decimal
   LDI B2,BYTE2($00000208)
                                 ;load imm, second byte of 520 decimal
   LDI B3,BYTE3($00000208)
                                 ;load imm, third byte of 520 decimal
   LDI B4,BYTE4($00000208)
                                 ;load imm, fourth byte of 520 decimal
calculations:
   CLR ANS1
                  ;Initialize Answer to zero
   CLR ANS2
   CLR ANS3
   CLR ANS4
   CLR ANS5
   CLR ANS6
   CLR ANS7
   SUB ANS8, ANS8 ; Clear ANS8 and Carry Flag
   MOV ANS1,B1
                    ;Copy Multiplier to Answer
   MOV ANS2,B2
   MOV ANS3,B3
    MOV ANS4,B4
LOOP:
   ROR ANS4
                  ;Shift Multiplier to right
   ROR ANS3
   ROR ANS2
   ROR ANS1
   DEC r0
               ;Decrement Loop Counter
   BREQ DONE
                   ;Check if all bits processed
    BRCC SKIP add
                   ;If Carry Clear skip addition
   ADD ANS5,A1
                    ;Add Multipicand into Answer
   ADC ANS6,A2
   ADC ANS7,A3
                   ;
   ADC ANS8,A4
SKIP add:
    ROR ANS8
                  ;Shift high bytes of Answer
   ROR ANS7
   ROR ANS6
   ROR ANS5
    RJMP LOOP
DONE: jmp DONE
```

#### 3. **DEVELOPED MODIFIED CODE OF TASK 2/A from TASK 1/A**

NA

#### 4. **SCHEMATICS**

0x00

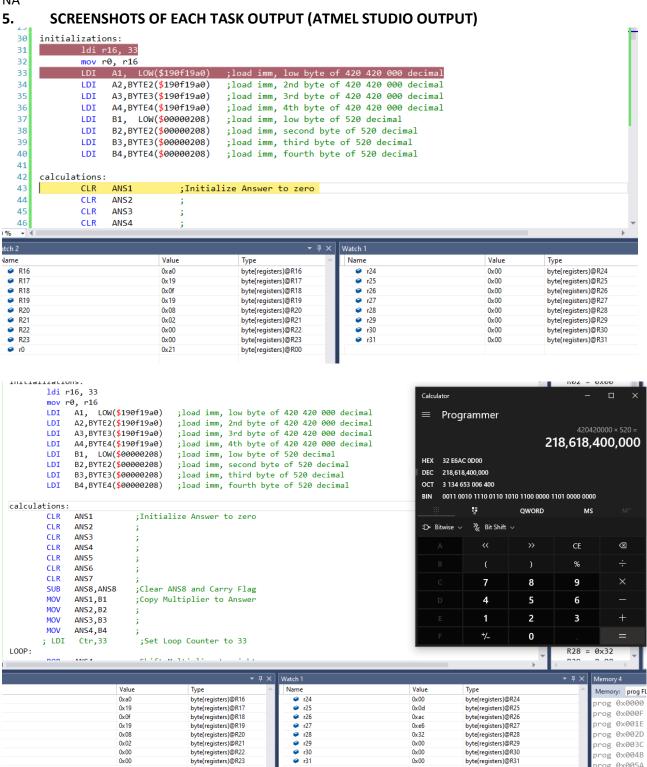
0x00

0x00

byte{registers}@R22

byte{registers}@R00

NA



0x00

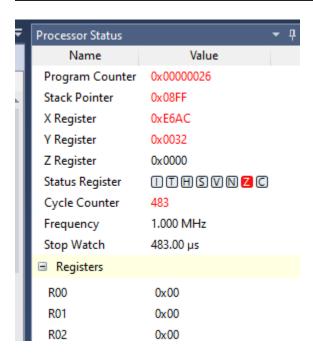
byte(registers)@R30

byte{registers}@R31

prog 0x004B

rog 0x005A

Watch 1		
Name	Value	Туре
	0x00	byte{registers}@R24
	0x0d	byte{registers}@R25
	0xac	byte{registers}@R26
	0хеб	byte{registers}@R27
	0x32	byte{registers}@R28
	0x00	byte{registers}@R29
	0x00	byte{registers}@R30
	0x00	byte{registers}@R31
		-



CYCLE COUNT = 483 FOR THE ENTIRE PROCESS OF THE CODE/DA1

# 6. SCREENSHOT OF EACH DEMO (BOARD SETUP)

NA

### 7. VIDEO LINKS OF EACH DEMO

NA

### 8. GITHUB LINK OF THIS DA

## **Student Academic Misconduct Policy**

http://studentconduct.unlv.edu/misconduct/policy.html

"This assignment submission is my own, original work". NAME OF THE STUDENT