CPE301 – FALL 2019

Design Assignment 1

Student Name: Dillon Archibald
Student #: 5004439916
Student Email: Archid1@unlv.nevada.edu
Primary Github address: https://github.com/Dil-bert/Alabaster.git
Directory: Dil-bert/Alabaster/DA1

1. COMPONENTS LIST AND CONNECTION BLOCK DIAGRAM w/ PINS

List of Components used
Block diagram with pins used in the Atmega328P

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

N/A

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

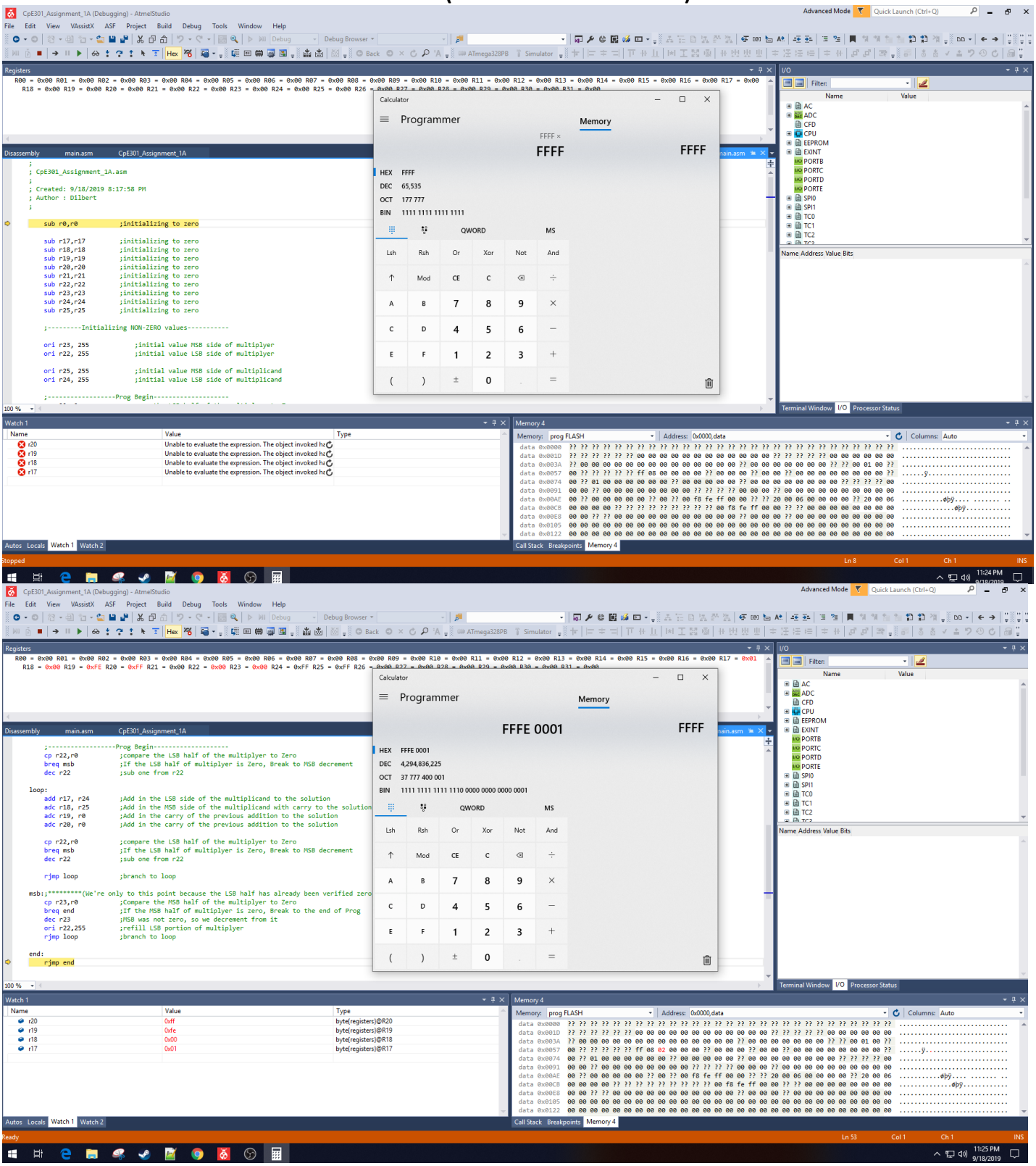
```
;
; CpE301_Assignment_1A.asm
;
; Created: 9/18/2019 8:17:58 PM
; Author : Dilbert
;

sub r0,r0                    ;initializing to zero

sub r17,r17                  ;initializing to zero
sub r18,r18                  ;initializing to zero
sub r19,r19                  ;initializing to zero
sub r20,r20                  ;initializing to zero
sub r21,r21                  ;initializing to zero
sub r22,r22                  ;initializing to zero
sub r23,r23                  ;initializing to zero
sub r24,r24                  ;initializing to zero
sub r25,r25                  ;initializing to zero

;---------Initializing NON-ZERO values-----------

ori r23, 255                 ;initial value MSB side of multiplyer
ori r22, 255                 ;initial value LSB side of multiplyer

ori r25, 255                 ;initial value MSB side of multiplicand
ori r24, 255                 ;initial value LSB side of multiplicand

;-----------------Prog Begin--------------------
```

```
cp r22,r0                       ;compare the LSB half of the multiplyer to Zero
breq msb                ;If the LSB half of multiplyer is Zero, Break to MSB decrement
dec r22                 ;sub one from r22

loop:
add r17, r24            ;Add in the LSB side of the multiplicand to the solution
adc r18, r25            ;Add in the MSB side of the multiplicand with carry to the solution
adc r19, r0             ;Add in the carry of the previous addition to the solution
adc r20, r0             ;Add in the carry of the previous addition to the solution

cp r22,r0               ;compare the LSB half of the multiplyer to Zero
breq msb                ;If the LSB half of multiplyer is Zero, Break to MSB decrement
dec r22                 ;sub one from r22

rjmp loop               ;branch to loop

msb:
;*********(We're only to this point because the LSB half has already been verified
;zero)**********
cp r23,r0               ;Compare the MSB half of the multiplyer to Zero
breq end                ;If the MSB half of multiplyer is zero, Break to the end of Prog
dec r23                 ;MSB was not zero, so we decrement from it
ori r22,255             ;refill LSB portion of multiplyer
rjmp loop               ;branch to loop

end:
rjmp end
```

4. SCHEMATICS

Use fritzing.org

5. SCREENSHOTS OF EACH TASK OUTPUT (ATMEL STUDIO OUTPUT)



6. SCREENSHOT OF EACH DEMO (BOARD SETUP)

N/A

7. VIDEO LINKS OF EACH DEMO

https://youtu.be/TVg_IT_wtfI
Note: the audio is very quiet, this will be remedied in future video submissions

8. GITHUB LINK OF THIS DA

https://github.com/Dil-bert/Alabaster.git

Student Academic Misconduct Policy
http://studentconduct.unlv.edu/misconduct/policy.html

"This assignment submission is my own, original work".
Dillon Archibald