Brandon Wade

CPE301 – SPRING 2016

Design Assignment 1

**DO NOT REMOVE THIS PAGE DURING SUBMISSION:**

The student understands that all required components should be submitted in complete for grading of this assignment.

|  |  |  |  |
| --- | --- | --- | --- |
| **NO** | **SUBMISSION ITEM** | **COMPLETED (Y/N)** | **MARKS**  **(/MAX)** |
| 0. | COMPONENTS LIST AND CONNECTION BLOCK DIAGRAM w/ PINS |  |  |
| 1. | INITIAL CODE OF TASK 1/A |  |  |
| 2. | INCREMENTAL / DIFFERENTIAL CODE OF TASK 2/B |  |  |
| 3. | INCREMENTAL / DIFFERENTIAL CODE OF TASK 3/C |  |  |
| 4. | INCREMENTAL / DIFFERENTIAL CODE OF TASK 4/D |  |  |
| 5. | INCREMENTAL / DIFFERENTIAL CODE OF TASK 5/E |  |  |
| 6. | SCHEMATICS |  |  |
| 7. | SCREENSHOTS OF EACH TASK OUTPUT |  |  |
| 8. | SCREENSHOT OF EACH DEMO |  |  |
| 9. | VIDEO LINKS OF EACH DEMO |  |  |
| 10. | GOOGLECODE LINK OF THE DA |  |  |
|  |  |  |  |
|  |  |  |  |

|  |  |  |  |
| --- | --- | --- | --- |
| 0. | COMPONENTS LIST AND CONNECTION BLOCK DIAGRAM w/ PINS |  |  |

Atmel Studio 7.0

|  |  |  |  |
| --- | --- | --- | --- |
| 1. | INITIAL CODE OF TASK 1/A |  |  |

;

; DA1.asm

;

; Created: 2/25/2016 3:40:18 PM

; Author : Brandon

;

LDI XL, LOW(RAMEND/2)

LDI XH, HIGH(RAMEND/2) ; X gets RAMEND/2

MOVW YL, XL ; Y gets X

LDI R16, 25 ; loop 25 times

storeLoop:

ST Y, YL ; store LOW(RAMEND/2) + n at Y

INC YL ; increment memory location and number

DEC R16 ; decrement counter

CPI R16, 0 ; check if counter is at zero

BRNE storeLoop ; loop 25 times

LDI R16, 25 ; loop 25 times

LDI R18, 0 ; used for carry

LDI R20, 0

LDI R21, 0

LDI R22, 0

LDI R23, 0 ; initialize sum registers

mainLoop:

MOV R17, XL ; R17 gets value at RAMEND/2 + n

check7Loop:

SUBI R17, 7 ; subtract seven from number

BRCS false7 ; jump if not divide by 7

CPI R17, 0 ; check for remainder

BRNE check7Loop ; jump if continue

ADD R20, XL

ADC R21, R18 ; add current number to sum

false7:

MOV R17, XL ; R17 gets value at RAMEND/2 + n

check3Loop:

SUBI R17, 3 ; subtract three from number

BRCS false3 ; jump if not divide by 3

CPI R17, 0 ; check for remainder

BRNE check3Loop ; jump if continue

ADD R22, XL

ADC R23, R18 ; add current number to sum

false3:

INC XL

DEC R16 ; decrement counter

CPI R16, 0 ; check if counter is at zero

BRNE mainLoop ; loop 25 times

CPI R21, 0 ; check if sum is greater than 8 bits

BRNE over8bits ; jump to set bit

CPI R23, 0 ; check if sum is greater than 8 bits

BRNE over8bits ; jump to set bit

RJMP done ; done

over8bits:

SBR R18, 8 ; set third bit in R18

MOV R7, R18 ; move R18 to R7

done:

RJMP done

|  |  |  |  |
| --- | --- | --- | --- |
| 6. | SCHEMATICS |  |  |

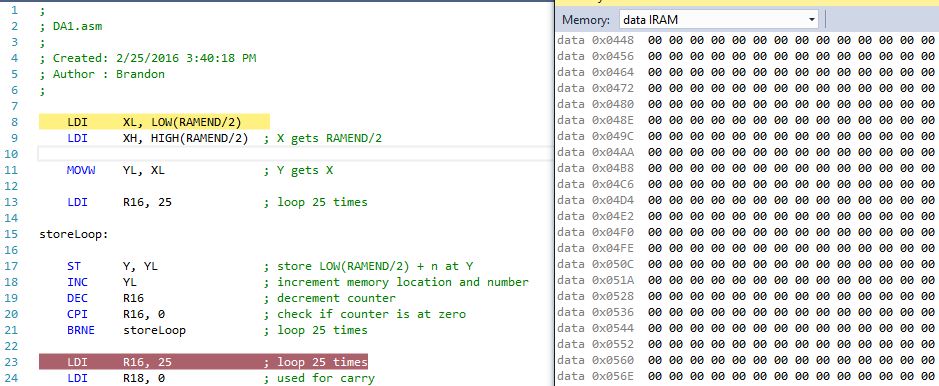
Simluation, no Schematics

|  |  |  |  |
| --- | --- | --- | --- |
| 7. | SCREENSHOTS OF EACH TASK OUTPUT |  |  |

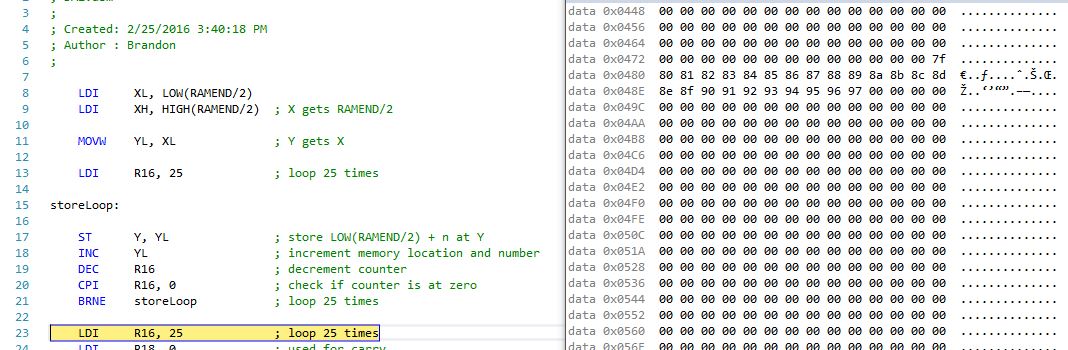
TASK 1/A:

Memory gets filled with incremental numbers starting at RAMEND/2

Before: memory cleared



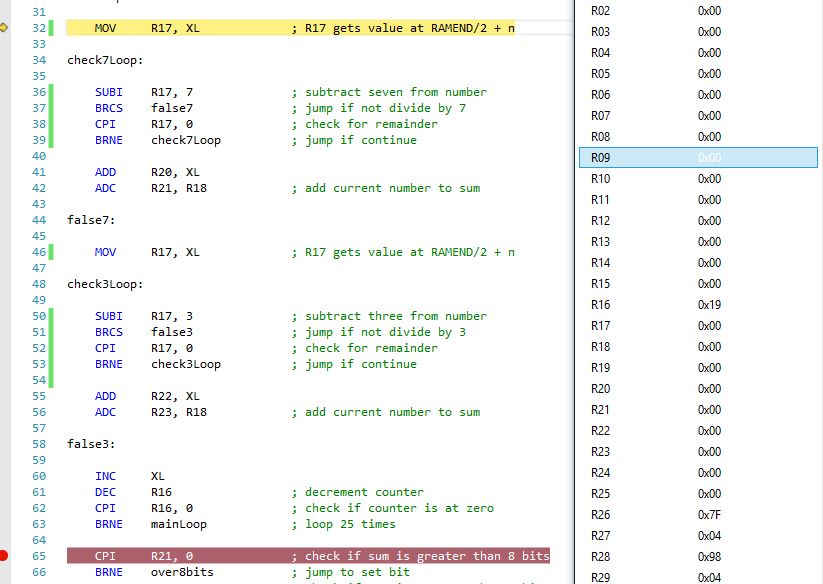
After: memory given incremental values



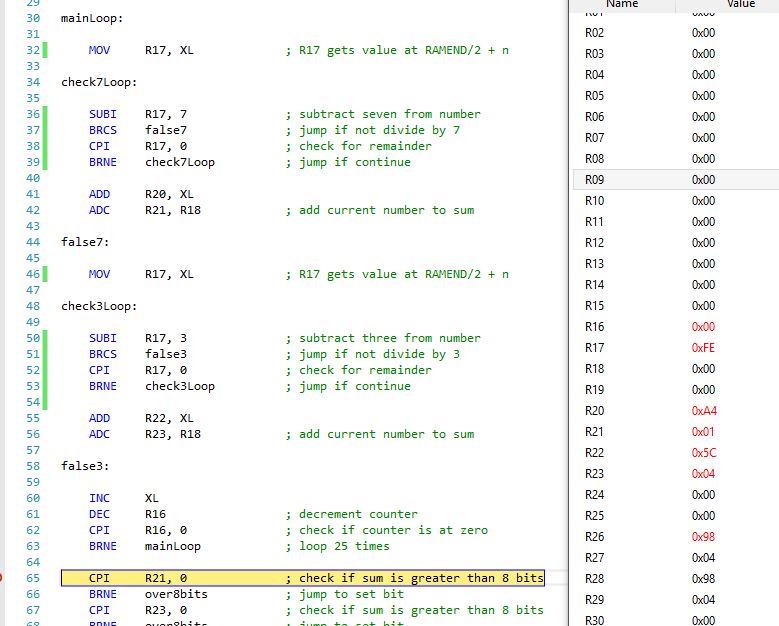
Task 1/B

R20:R21 Registers added with values divisible by 7 and R22:R23 Registers added with values divisible by 3

Before: R20-23 cleared



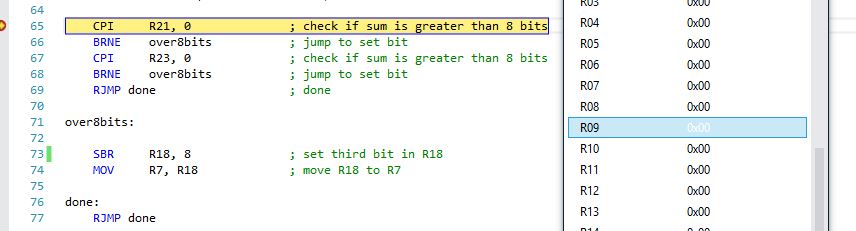
After: R20-R23 filled with values divisible by 7 and 3



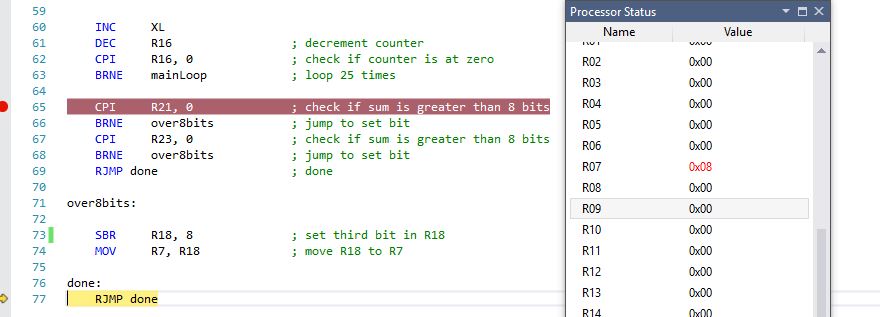
Task 1/D

Because sum is greater than 8 bits, bit three (0x08) gets set in R7

Before: Bit cleared



After: Bit Set

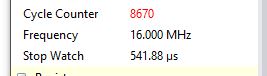


Task 1/E

Execution Time = 541.88 us

Number of Cycles = 8670

Cycle Count/Time to complete @ 16MHz



|  |  |  |  |
| --- | --- | --- | --- |
| 8. | SCREENSHOT OF EACH DEMO |  |  |

TASK 1/A:

Simulation, no demo

|  |  |  |  |
| --- | --- | --- | --- |
| 9. | VIDEO LINKS OF EACH DEMO |  |  |
| N/A | | | |
| 10. | GOOGLECODE LINK OF THE DA |  |  |
| https://github.com/wadeb1/KF3HF6ZFMP.git | | | |

**Student Academic Misconduct Policy**

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

“This assignment submission is my own, original work”.

Brandon Wade