Brian Medrano Kiaer

CPE301 – SPRING 2018

Design Assignment 1

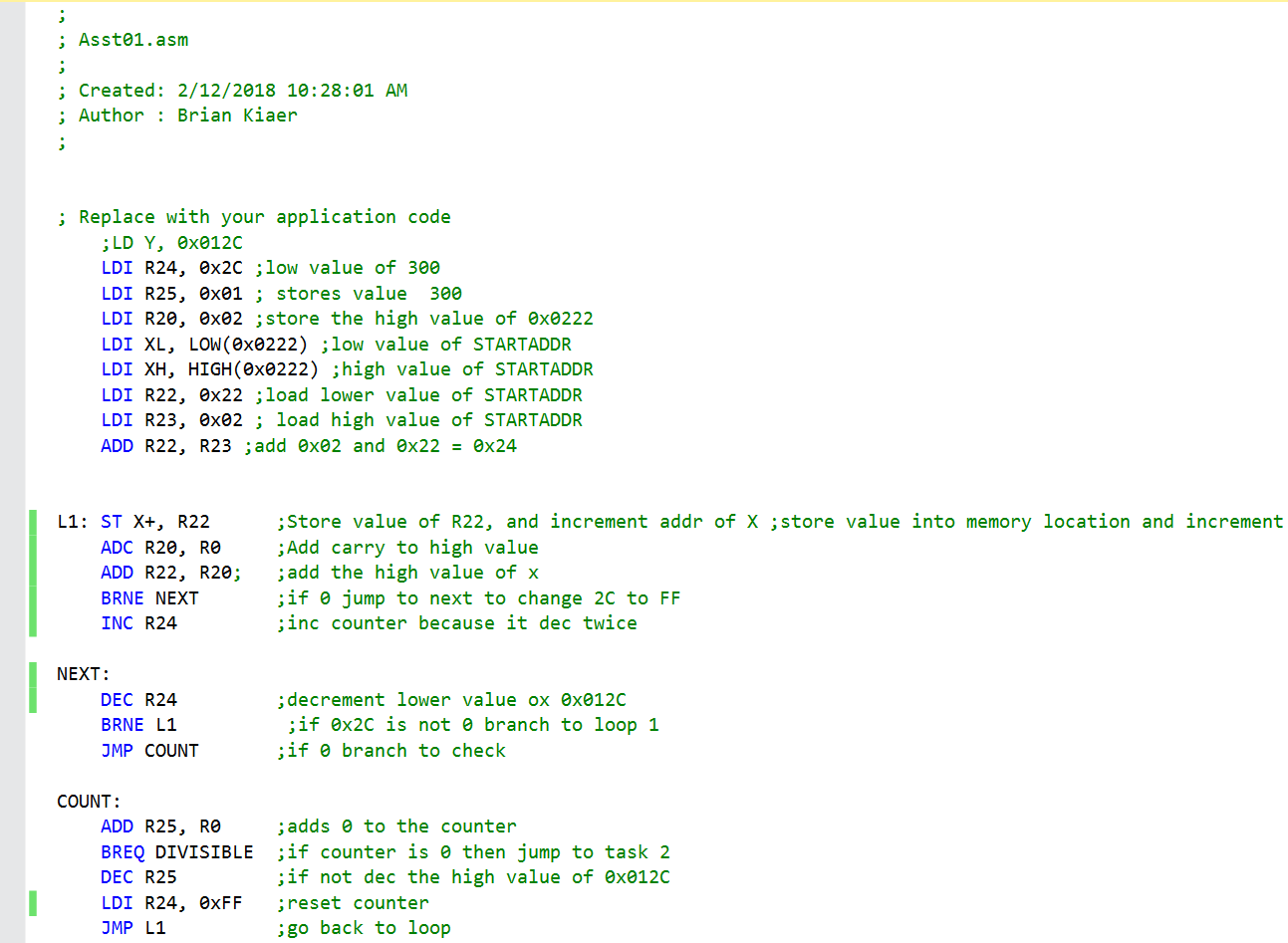
**DO NOT REMOVE THIS PAGE DURING SUBMISSION:**

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)** |
| 1 | COMPONENTS LIST AND CONNECTION BLOCK DIAGRAM w/ PINS |  |  |
| 2. | INITIAL CODE OF TASK 1/A |  |  |
| 3. | INCREMENTAL / DIFFERENTIAL CODE OF TASK 2/B |  |  |
| 3. | INCREMENTAL / DIFFERENTIAL CODE OF TASK 3/C |  |  |
| 3. | INCREMENTAL / DIFFERENTIAL CODE OF TASK 4/D |  |  |
| 3. | INCREMENTAL / DIFFERENTIAL CODE OF TASK 5/E |  |  |
| 4. | SCHEMATICS |  |  |
| 5. | SCREENSHOTS OF EACH TASK OUTPUT |  |  |
| 5. | SCREENSHOT OF EACH DEMO |  |  |
| 6. | VIDEO LINKS OF EACH DEMO |  |  |
| 7. | GOOGLECODE LINK OF THE DA |  |  |
|  |  |  |  |
|  |  |  |  |

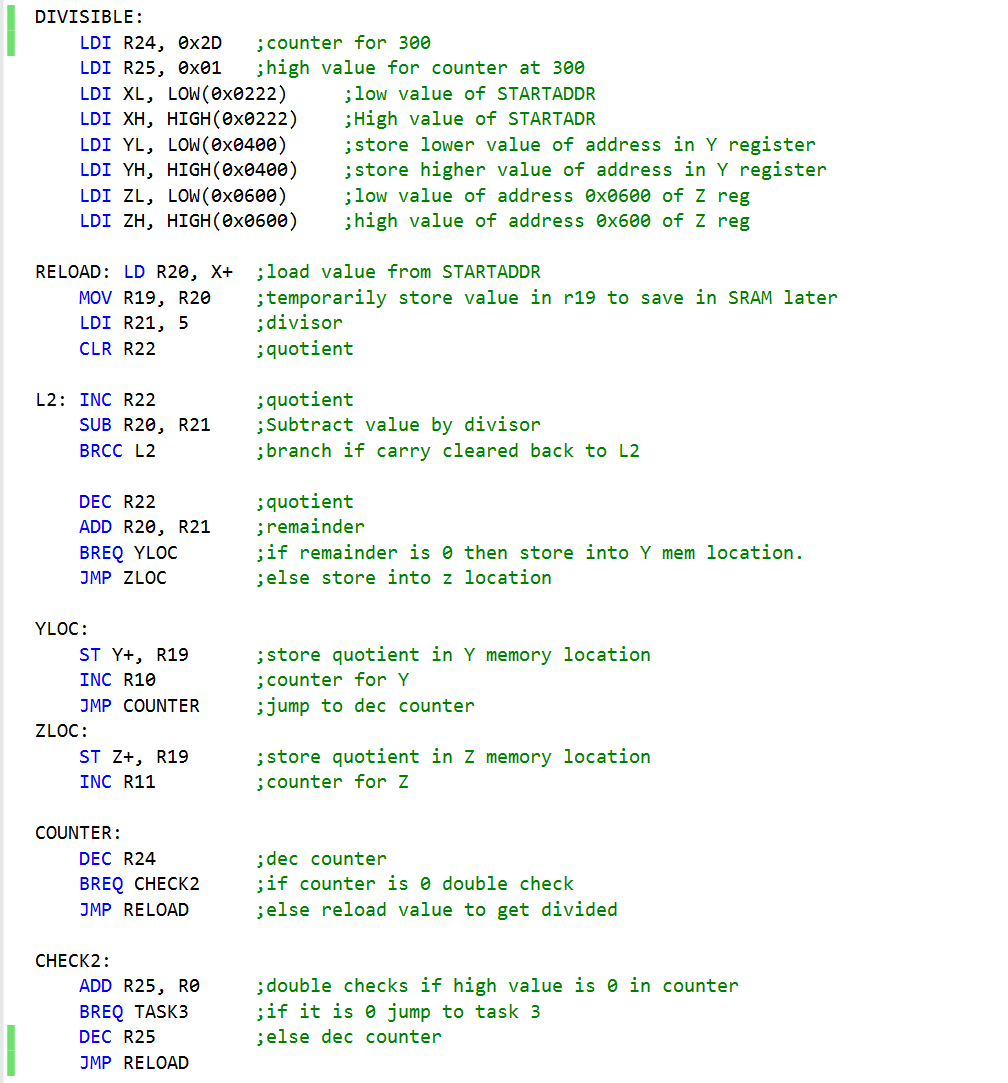
1. **INITIAL/DEVELOPED CODE OF TASK 1**

Code Segment that is responsible for storing 300 numbers starting with the sum of low(STARTADDR) and high(STARTADDR) at memory location 0x0222.



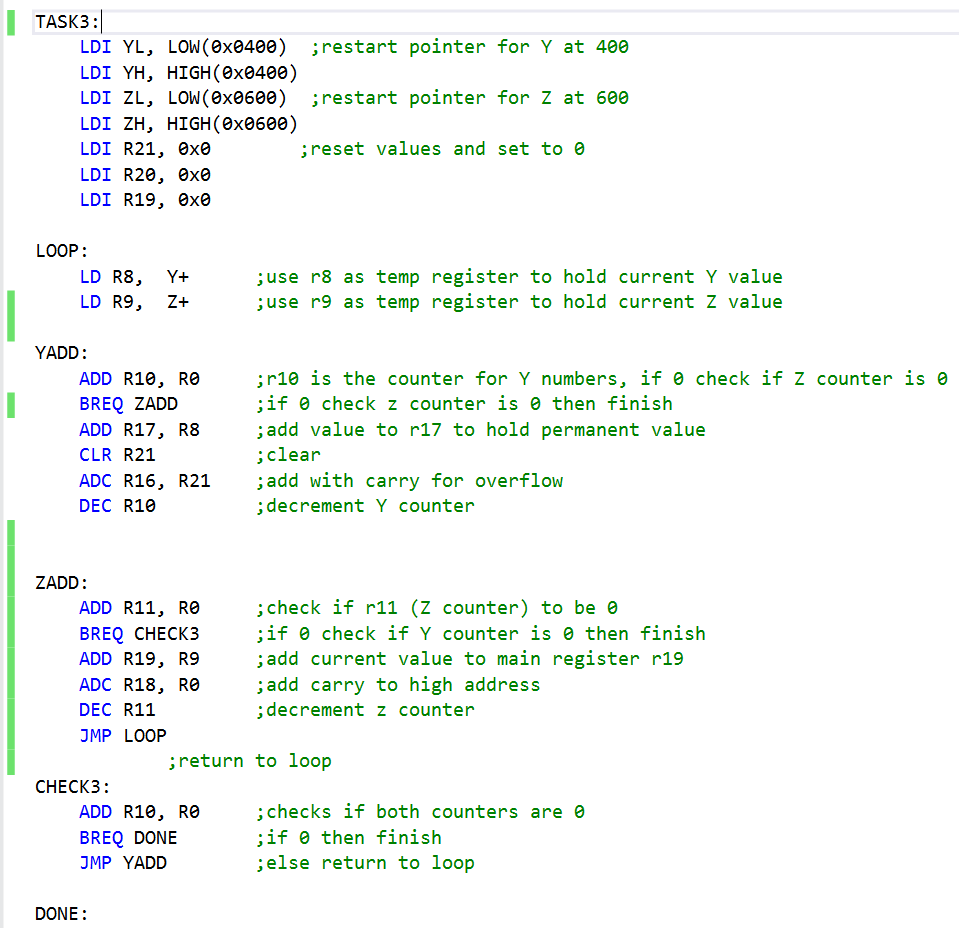
1. **INITIAL CODE OF TASK 2**

Code Segment responsible for dividing the stored 300 values by 5 and appropriately storing them in memory location 0x0400 if divisible by 5 and memory location 0x0600 if not divisible by 5. This is managed by using the X/Y/Z Registers as pointers.

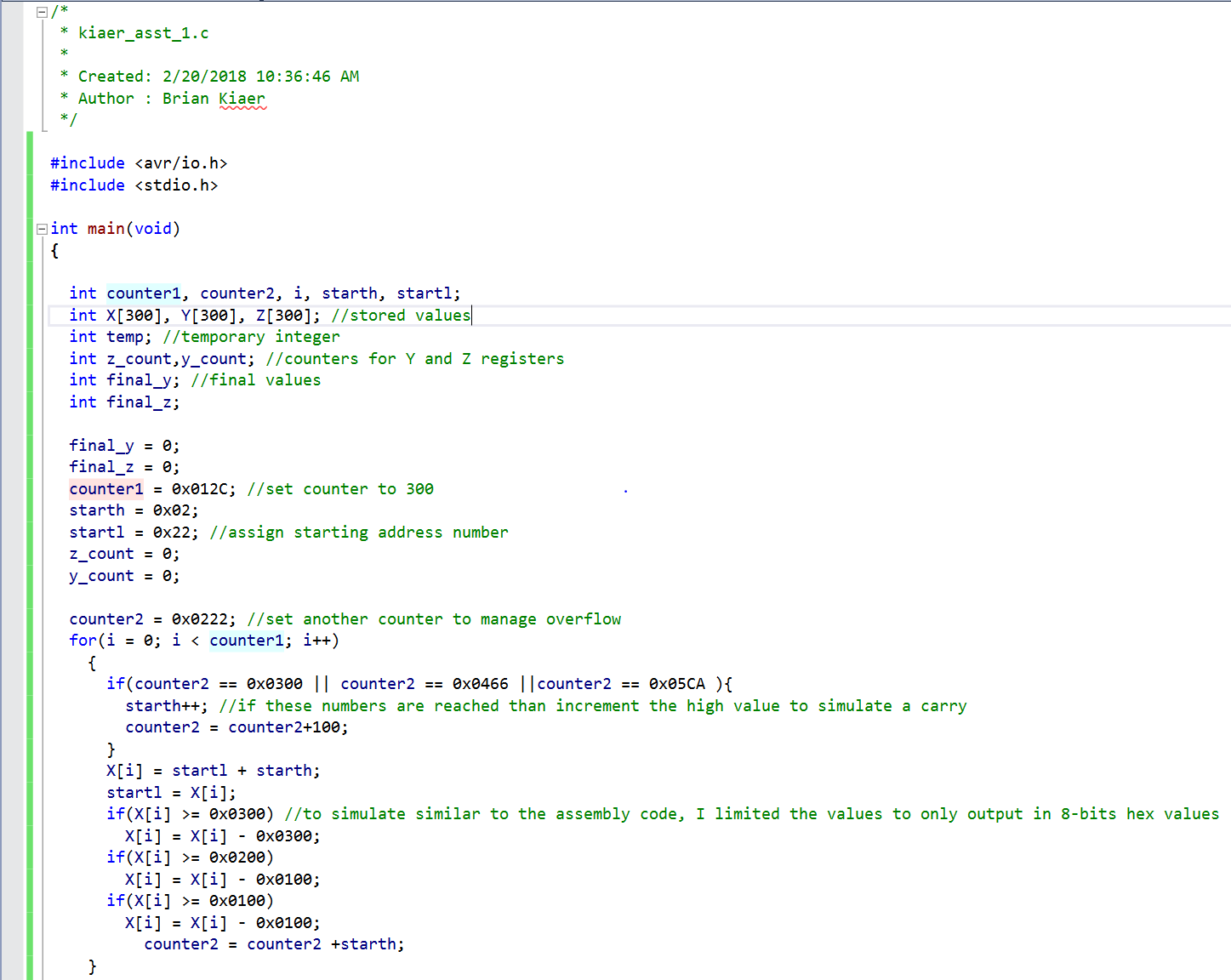


1. **INITIAL CODE OF TASK 3**

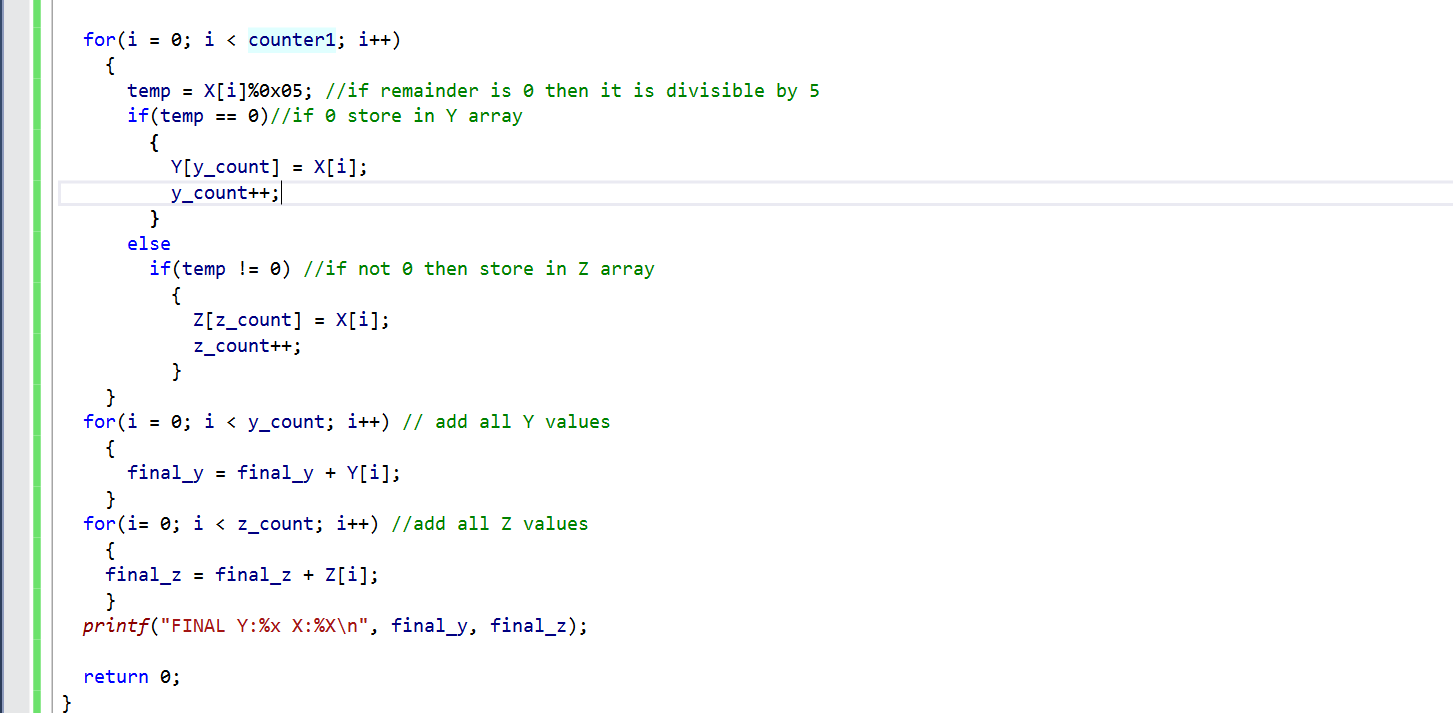
This code segment is responsible for simultaneously adding the values stored in 0x0400 and the values stored in 0x0600 into the respective registers R16\*R17 and R18\*R19.



1. **COMPLETE CODE FOR TASK 4**

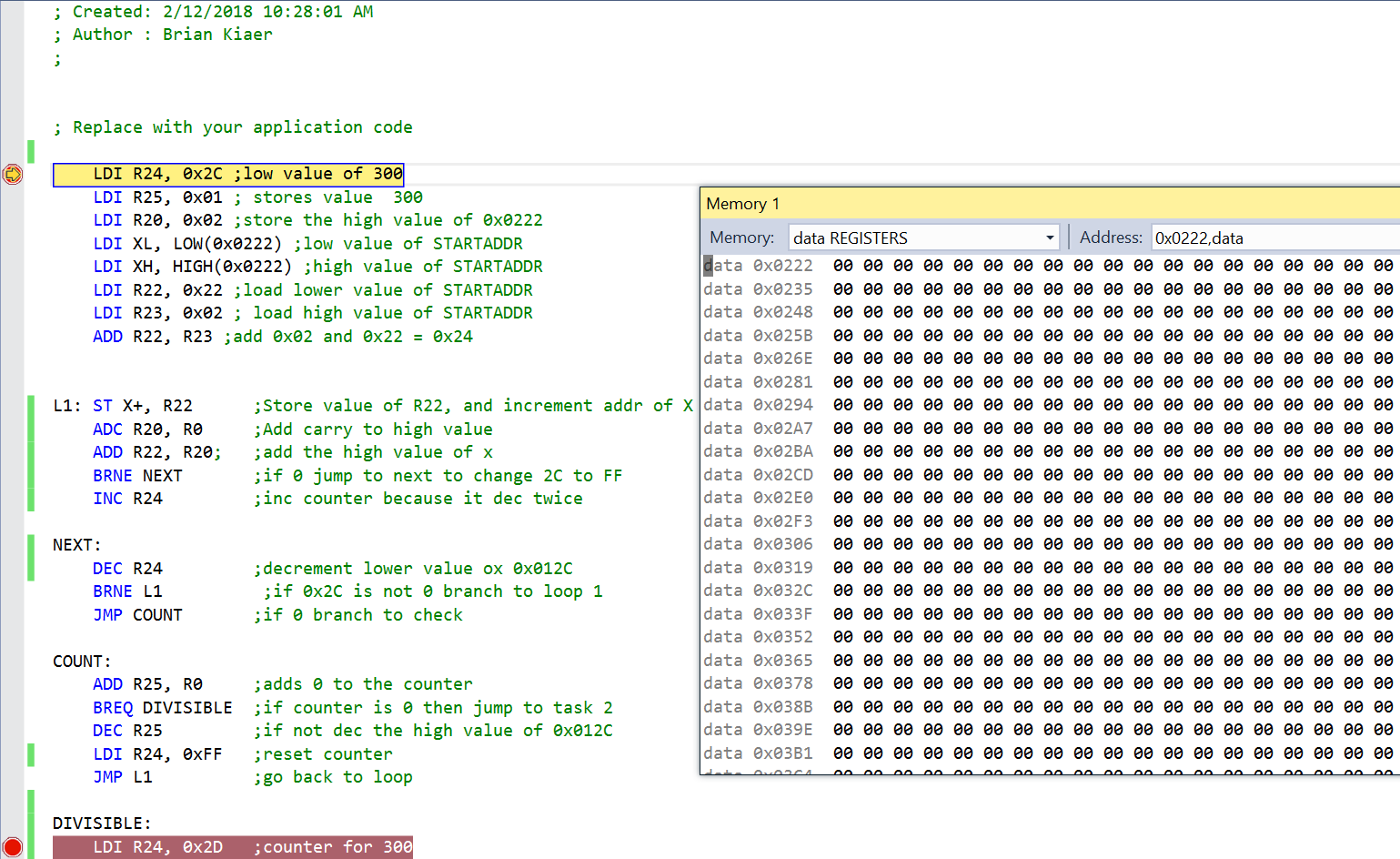
Task 1 in C

Task 2/3 in C

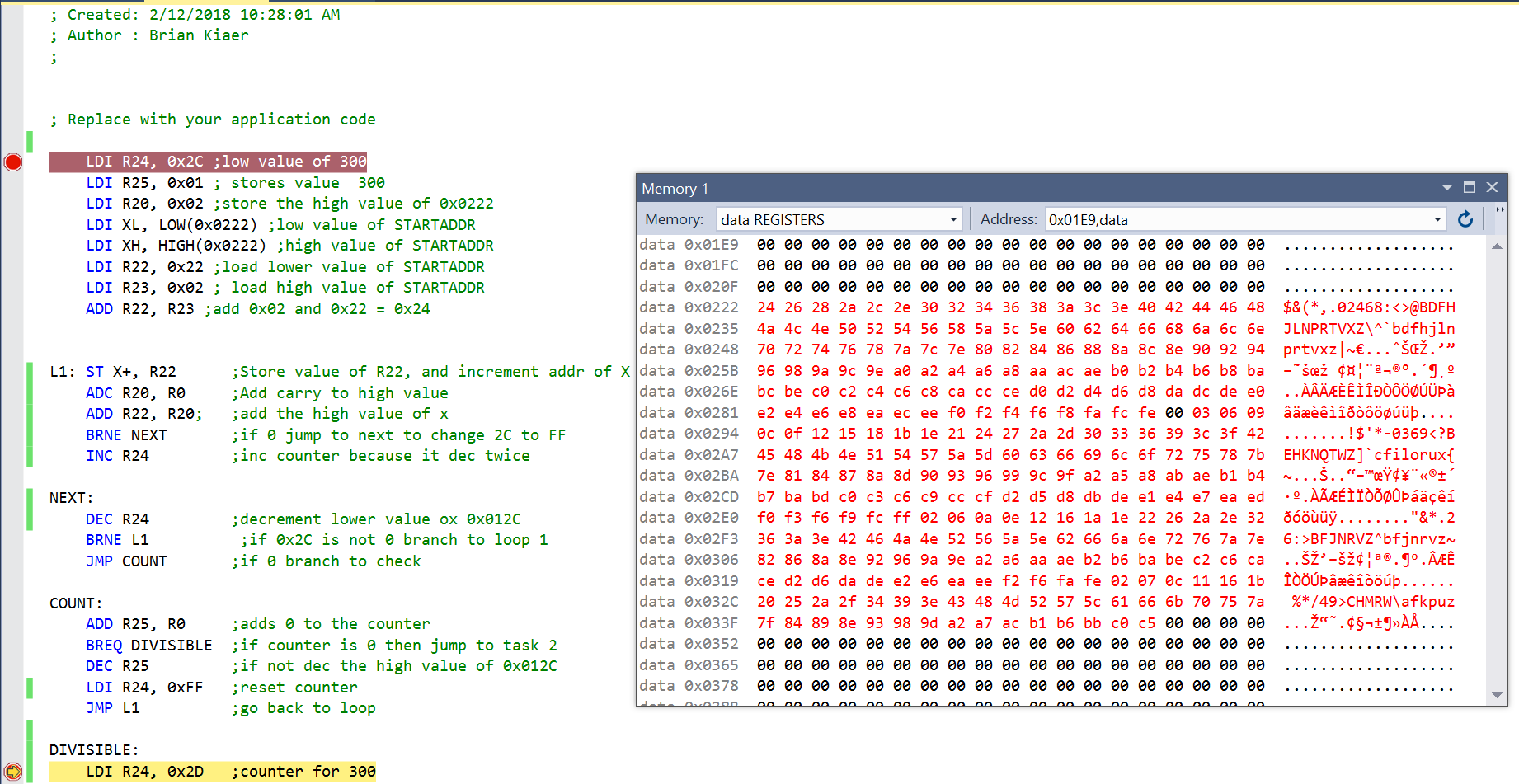


1. **SCREENSHOTS OF EACH TASK OUTPUT (ATMEL STUDIO OUTPUT)**

TASK 1

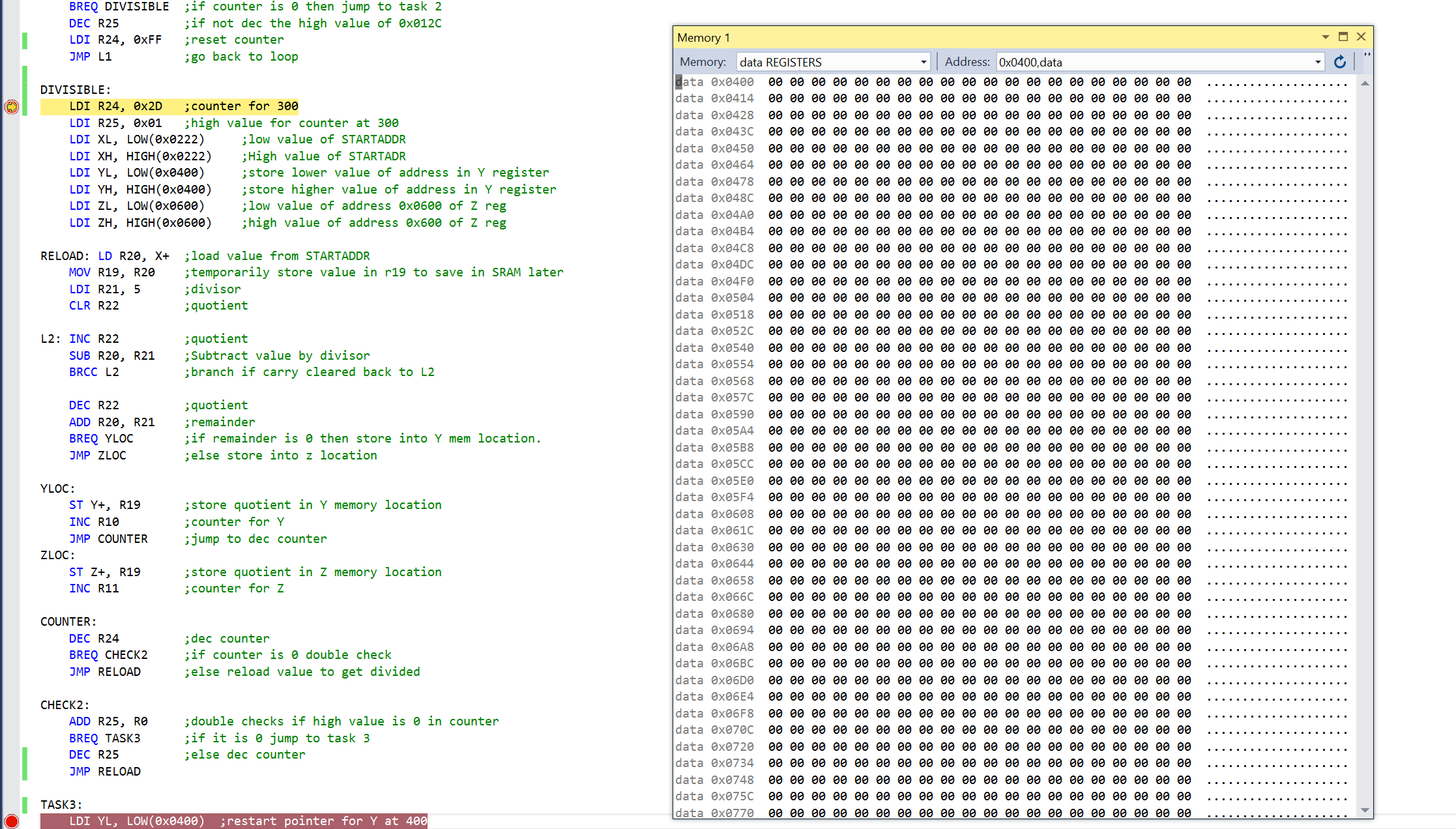


*Figure 1. Before storing values*

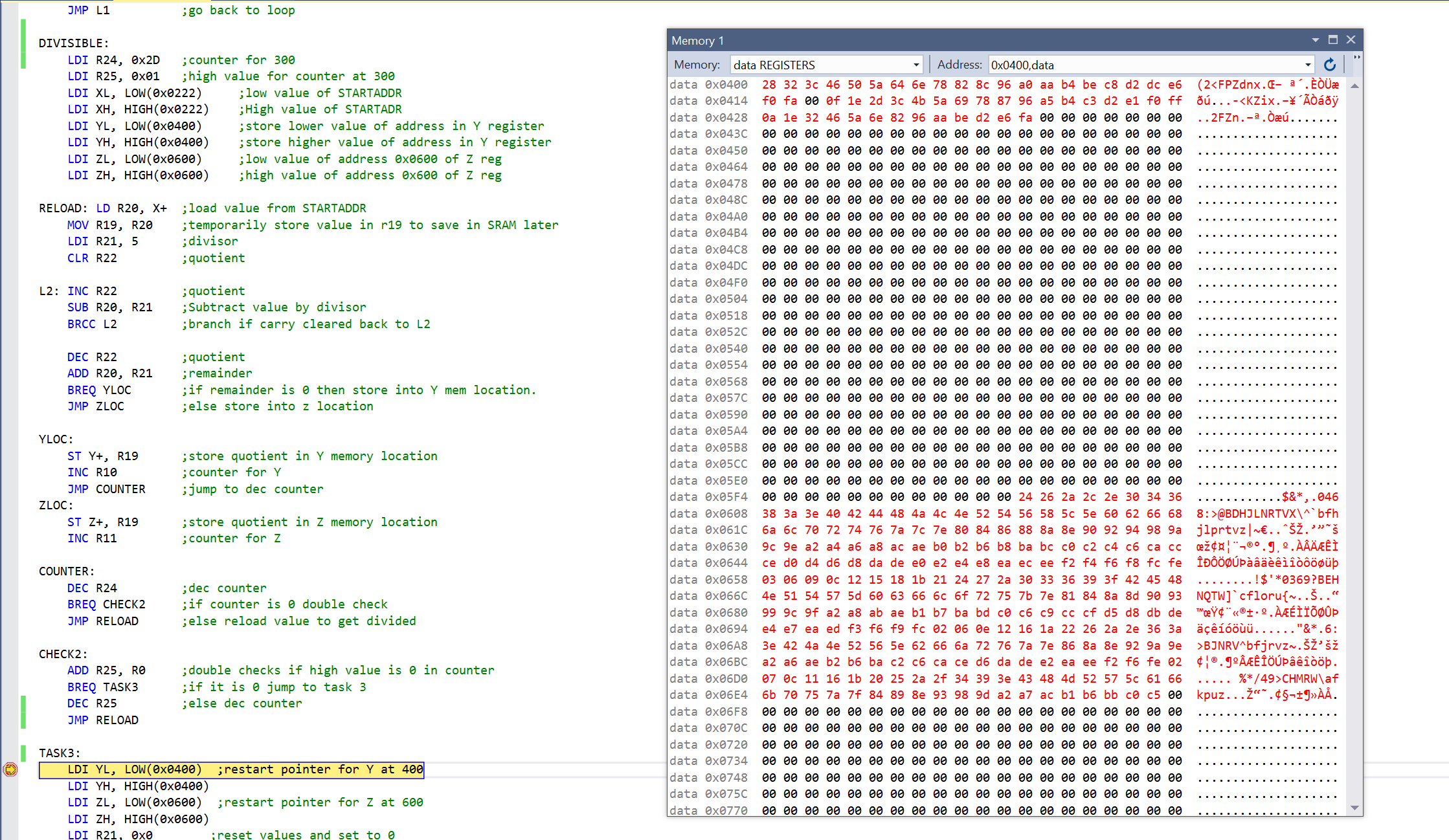


*Figure 2. After storing values*

TASK 2

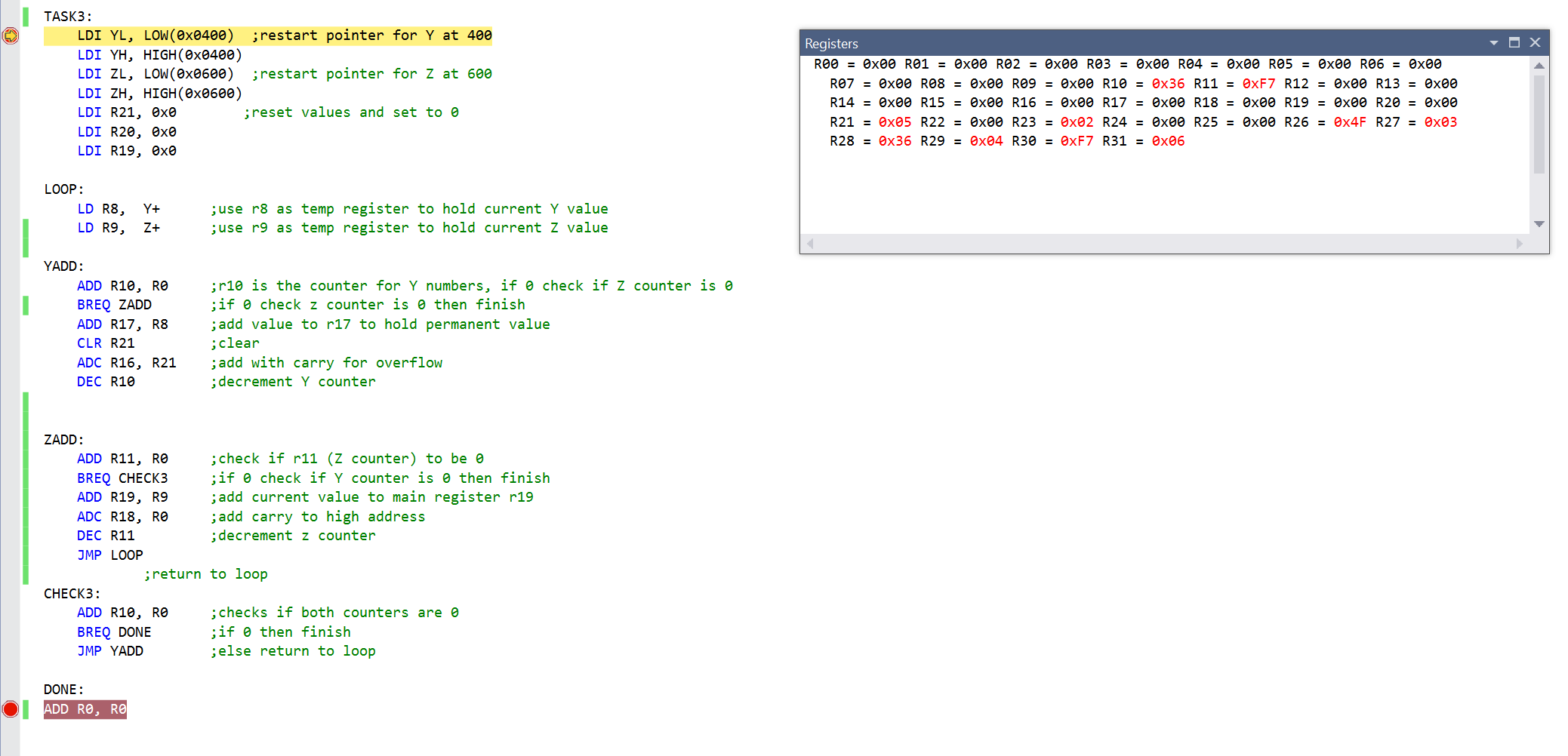


*Figure 3. Before dividing values and storing into 0x0400 and 0x0600*

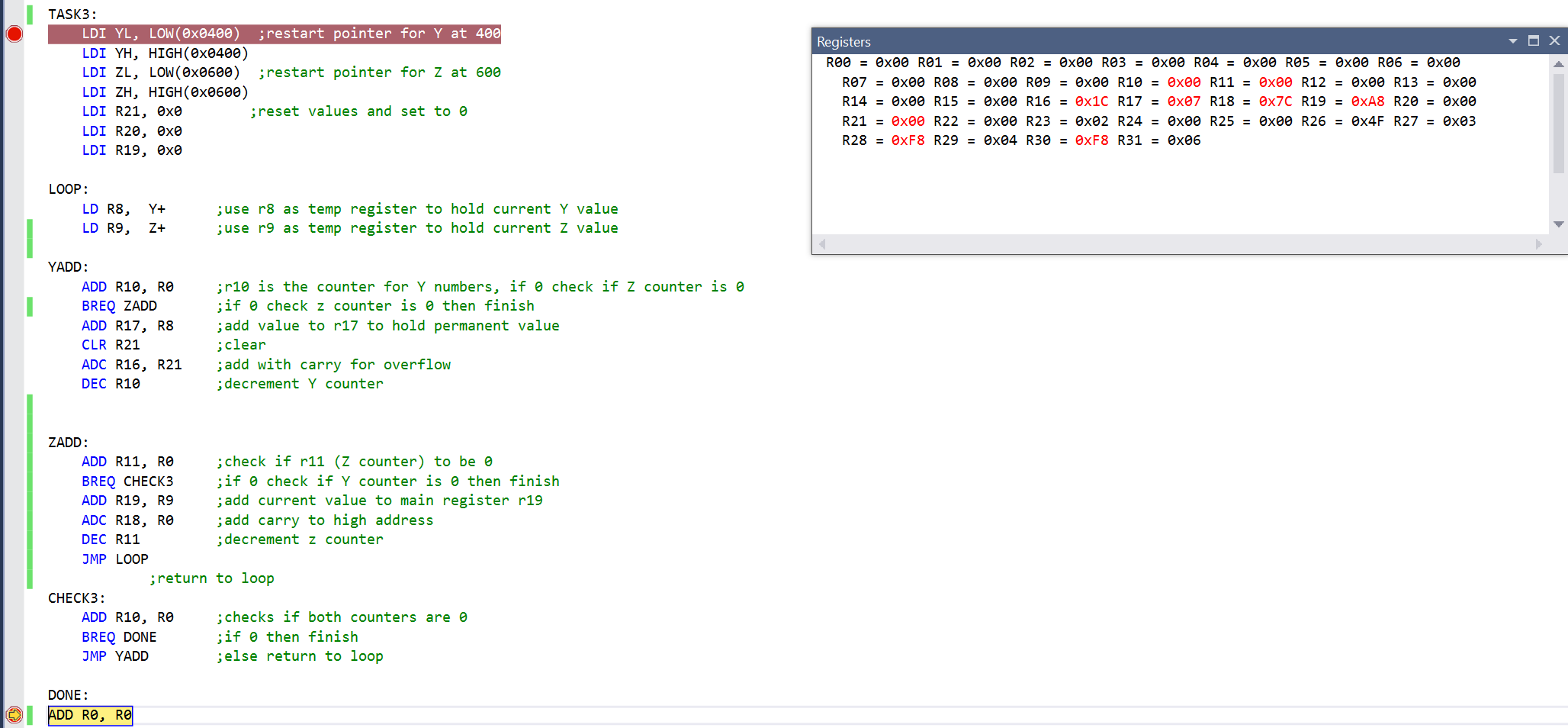


*Figure 4. After parsing values and storing in memory locations 0x0400 and 0x0600 simultaneously*

TASK 3

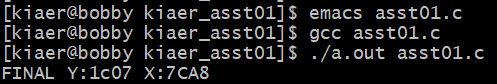


*Figure 5. Before adding values*



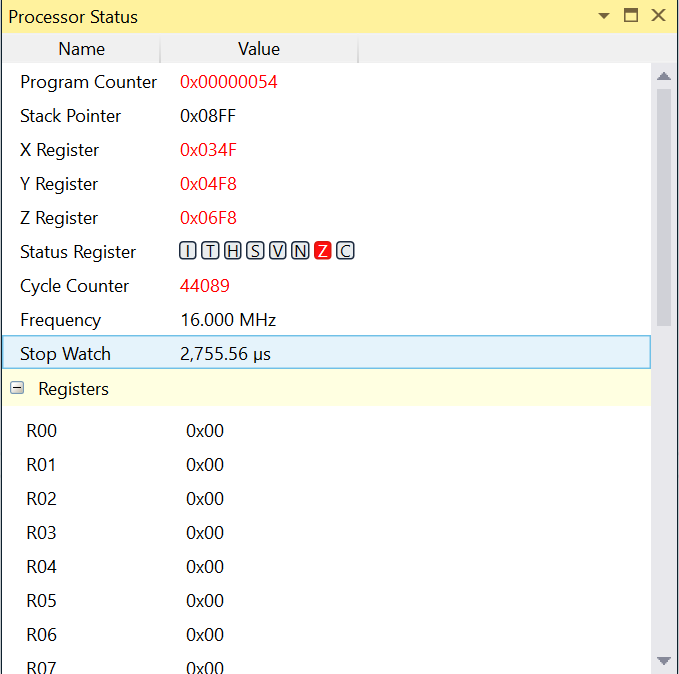
*Figure 6. Final Values in R16\*R17 and R18\*R19*

TASK 4



*Figure 7. Output of C code of Final Values through GIT BASH (Mintty)*

TASK 5



*Figure 8. Clock at 16MHz, execution done in 2755.56 microseconds*

1. **VIDEO LINKS OF EACH DEMO**

<https://youtu.be/v5MTSM_Fx3M>

1. **GITHUB LINK OF THIS DA**

https://github.com/bkiaer/DesignA1

**Student Academic Misconduct Policy**

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

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

Brian Medrano Kiaer