ECE375
Introduction to AVR Simulation with Microchip Studio

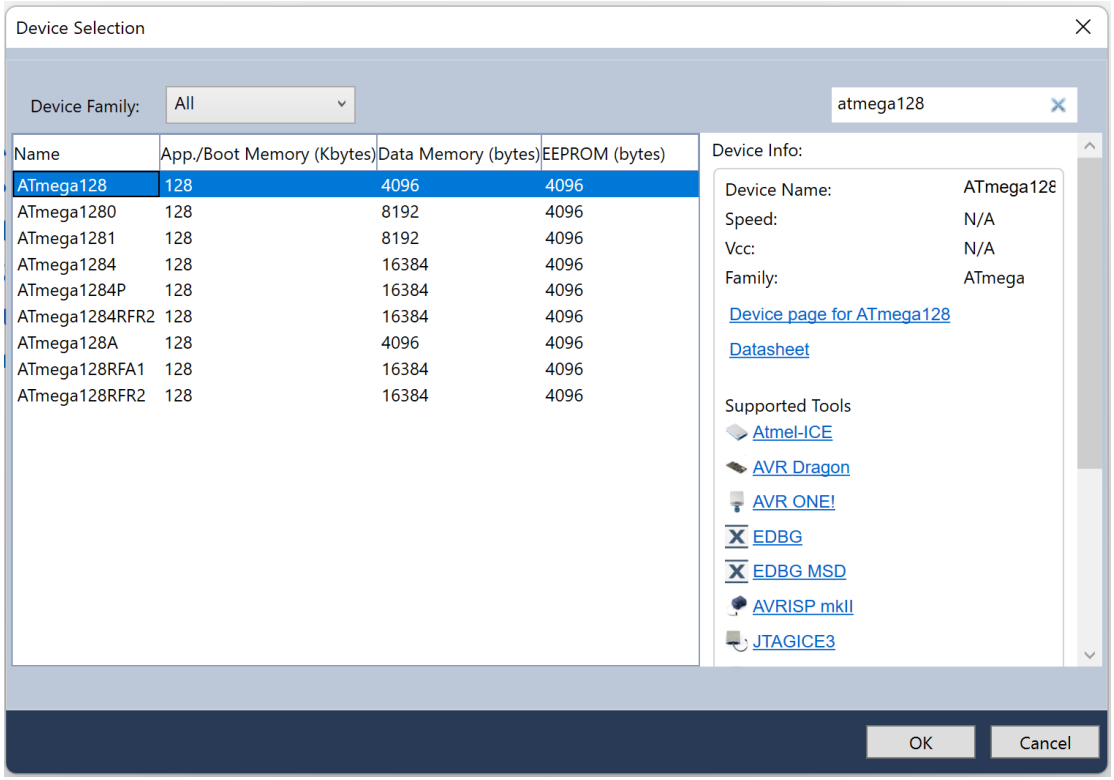
TA:

School of Electrical Engineering and Computer Science

Oregon State University

Supported Architecture

- Only Windows support the Microchip Studio
- Microchip Studio currently does not support a simulator for ATmega32U4. Select the ATmega128 device instead.



Goal of this Pre Lab

- Learn how to use the Microchip Studio simulator
- Set Break Points
- Start Debugging and Break
- Observe data using Processor, I/O port, and Memory window

Break Points

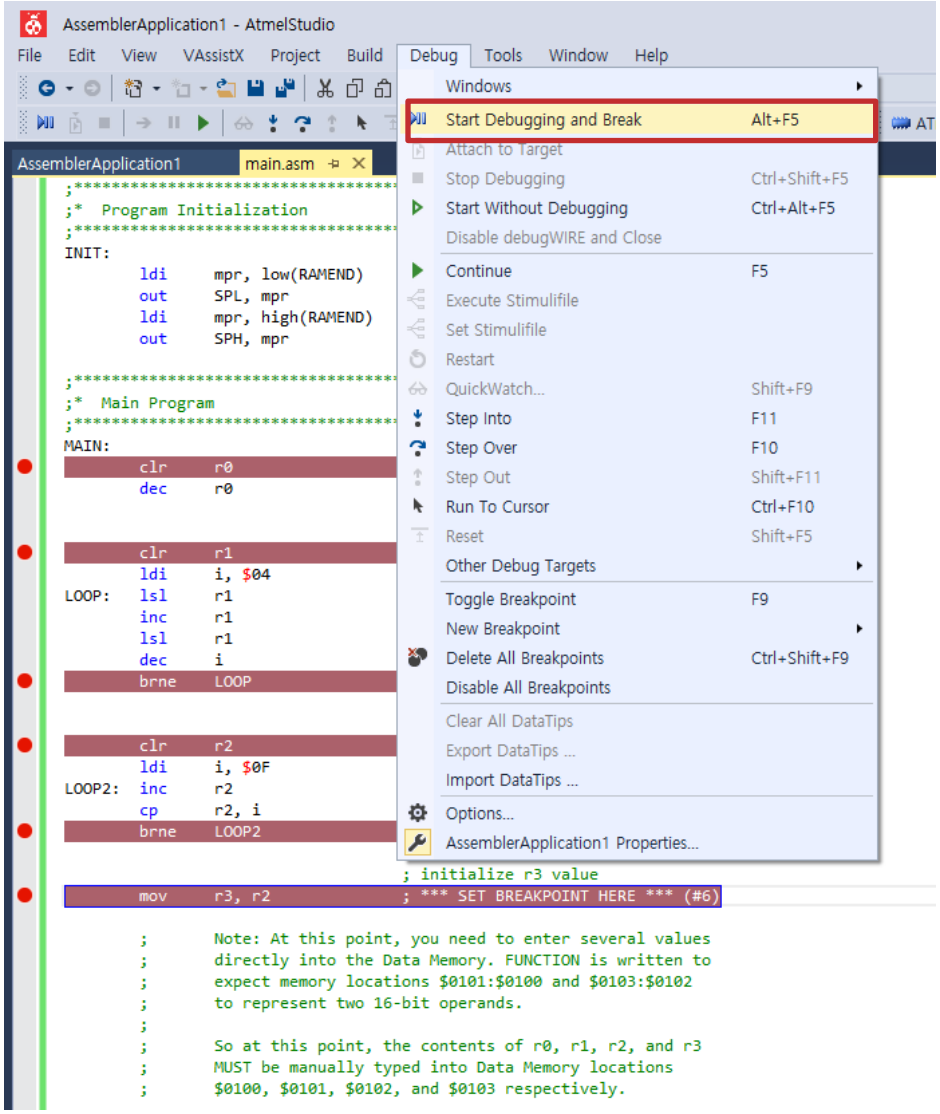
lab3 - AtmelStudio

File Edit View VAssistX ASF Project Build Debug Tools Window Help

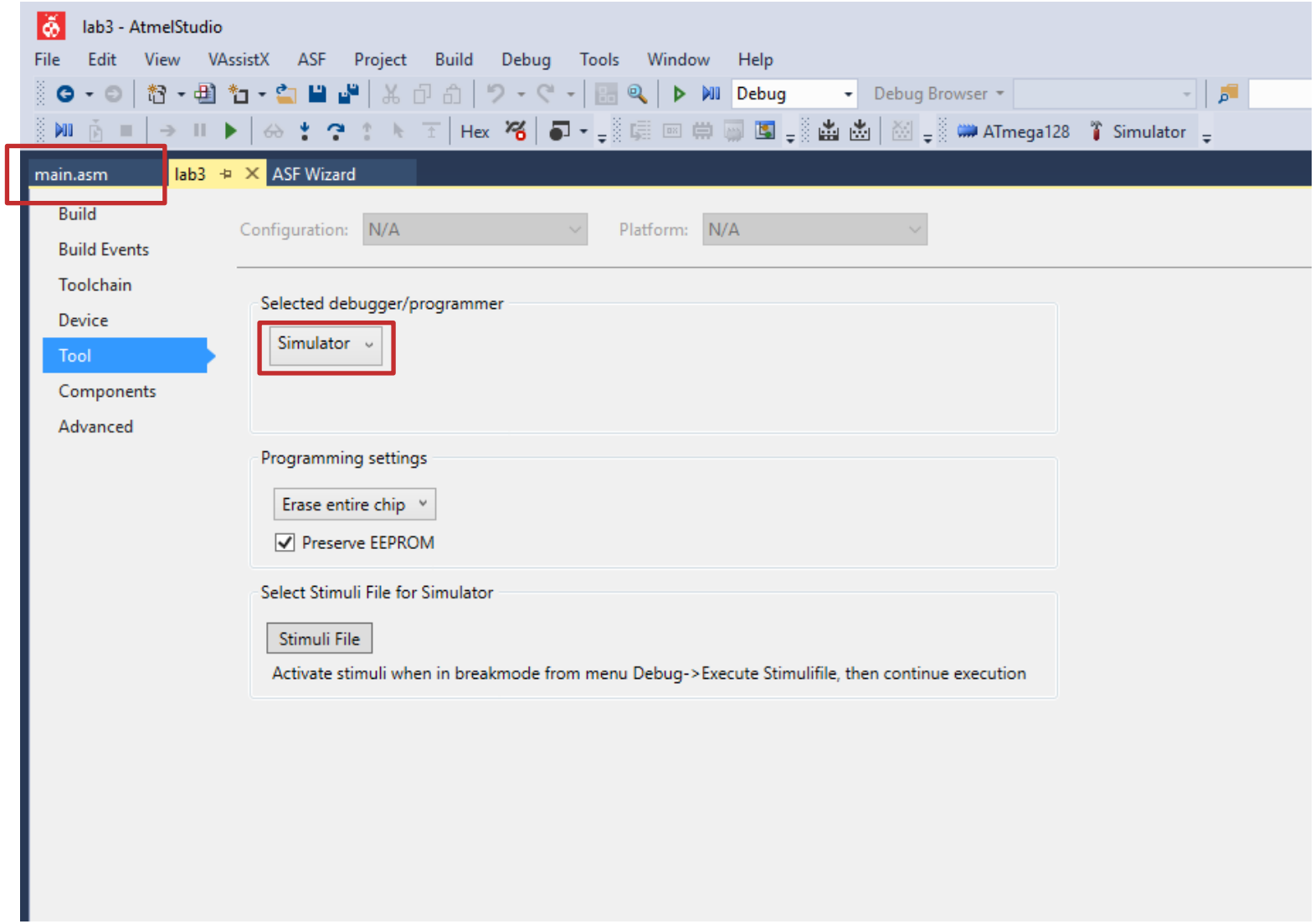
main.asm | lab3 | ASF Wizard

```
;***********************************************************
INIT:                                   ; The initialization routine
        ldi     mpr, low(RAMEND)        ; initialize Stack Pointer
        out     SPL, mpr
        ldi     mpr, high(RAMEND)
        out     SPH, mpr

;***********************************************************
;*  Main Program
;***********************************************************
MAIN:
        clr     r0                      ; *** SET BREAKPOINT HERE *** (#1)
        dec     r0                      ; initialize r0 value


        clr     r1                      ; *** SET BREAKPOINT HERE *** (#2)
        ldi     i, $04
LOOP:   lsl     r1                      ; initialize r1 value
        inc     r1
        lsl     r1
        dec     i
        brne    LOOP                    ; *** SET BREAKPOINT HERE *** (#3)


        clr     r2                      ; *** SET BREAKPOINT HERE *** (#4)
        ldi     i, $0F
LOOP2:  inc     r2                      ; initialize r2 value
        cp      r2, i
        brne    LOOP2                   ; *** SET BREAKPOINT HERE *** (#5)

                                        ; initialize r3 value
        mov     r3, r2                  ; *** SET BREAKPOINT HERE *** (#6)

        ;       Note: At this point, you need to enter several values
        ;       directly into the Data Memory. FUNCTION is written to
        ;       expect memory locations $0101:$0100 and $0103:$0102
        ;       to represent two 16-bit operands.
        ;
        ;       So at this point, the contents of r0, r1, r2, and r3
        ;       MUST be manually typed into Data Memory locations
```

Start Debugging and Break



Simulator

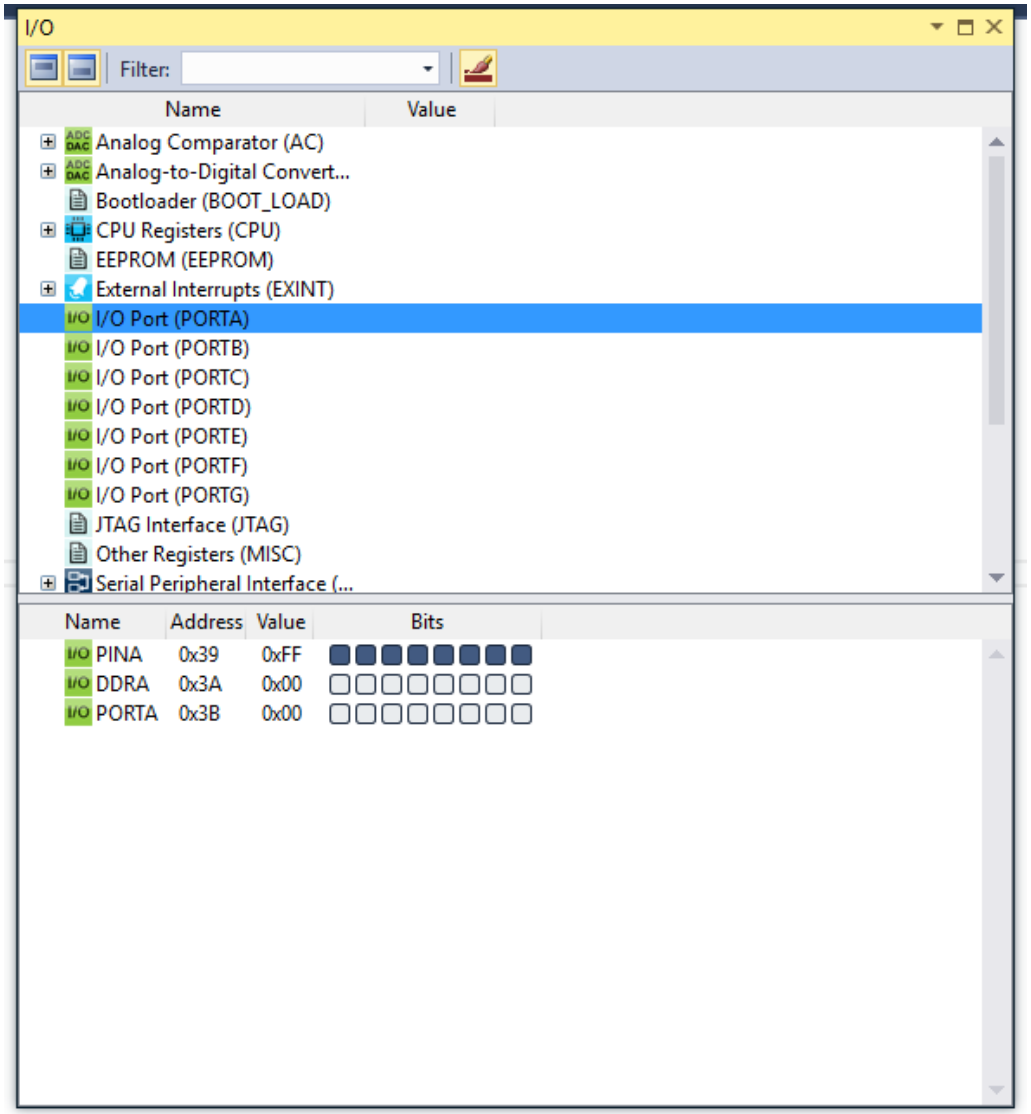

lab3 - AtmelStudio
File
Edit
View
VAssistX
ASF
Project
Build
Debug
Tools
Window
Help
Debug
Debug Browser
Hex
ATmega128
Simulator
main.asm
lab3
ASF Wizard
Build
Build Events
Toolchain
Device
Tool
Components
Advanced
Configuration:
N/A
Platform:
N/A
Selected debugger/programmer
Simulator
Programming settings
Erase entire chip
Preserve EEPROM
Select Stimuli File for Simulator
Stimuli File
Activate stimuli when in breakmode from menu Debug->Execute Stimulifile, then continue execution

Processor Status

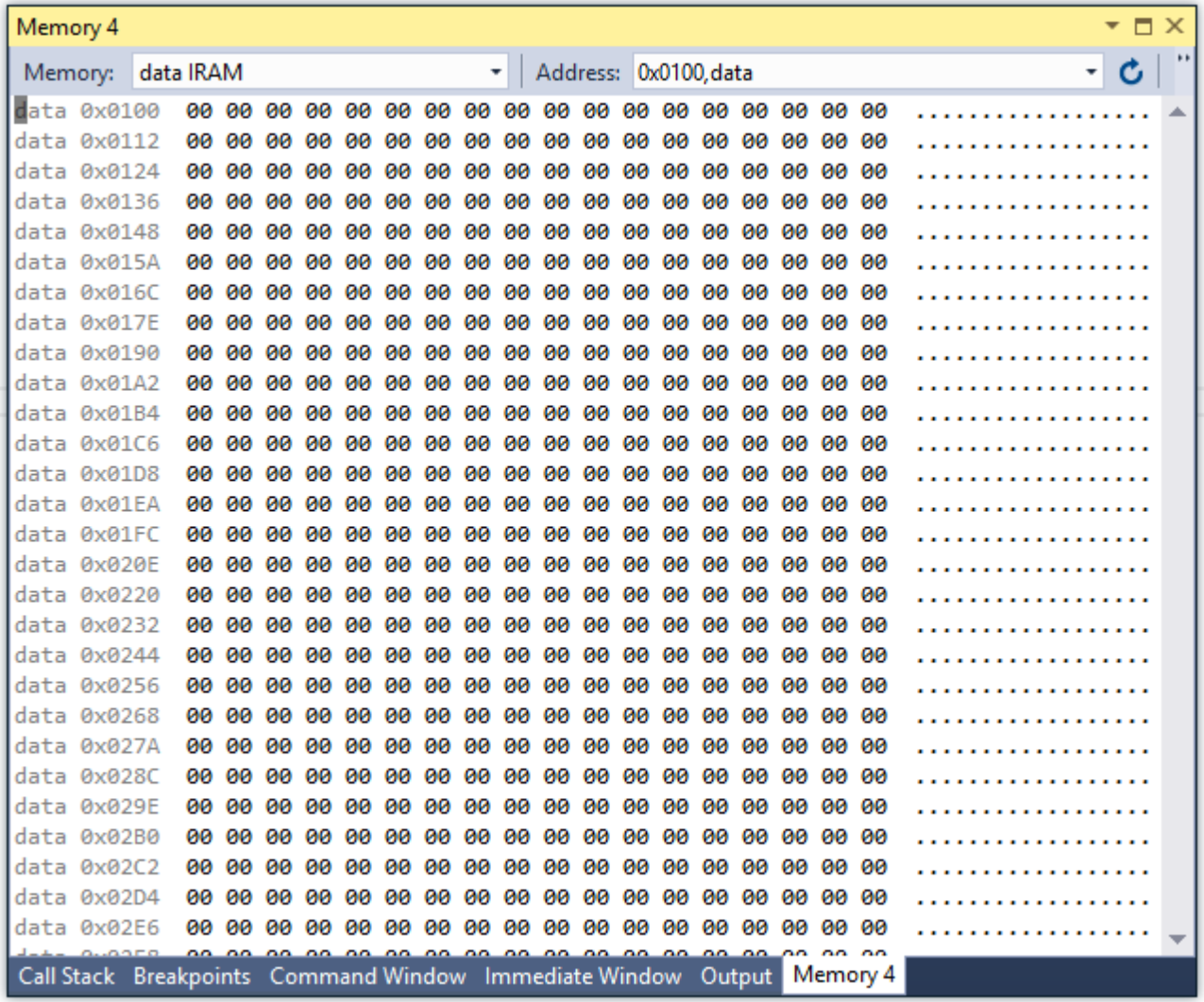
Processor Status

Name	Value
Program Counter	0x00000000
Stack Pointer	0x0000
X Register	0x0000
Y Register	0x0000
Z Register	0x0000
Status Register	I T H S V N Z C
Cycle Counter	0
Frequency	1.000 MHz
Stop Watch	0.00 µs
Registers	
R00	0x00
R01	0x00
R02	0x00
R03	0x00
R04	0x00
R05	0x00
R06	0x00
R07	0x00
R08	0x00
R09	0x00
R10	0x00
R11	0x00
R12	0x00
R13	0x00
R14	0x00
R15	0x00
R16	0x00
R17	0x00

I/O Ports


I/O
Filter:
Name
Value
Analog Comparator (AC)
Analog-to-Digital Convert...
Bootloader (BOOT_LOAD)
CPU Registers (CPU)
EEPROM (EEPROM)
External Interrupts (EXINT)
I/O Port (PORTA)
I/O Port (PORTB)
I/O Port (PORTC)
I/O Port (PORTD)
I/O Port (PORTE)
I/O Port (PORTF)
I/O Port (PORTG)
JTAG Interface (JTAG)
Other Registers (MISC)
Serial Peripheral Interface (...
Name
Address
Value
Bits
PINA 0x39 0xFF
DDRA 0x3A 0x00
PORTA 0x3B 0x00

Memory


Memory 4
Memory: data IRAM
Address: 0x0100,data
data 0x0100 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 .....
data 0x0112 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 .....
data 0x0124 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 .....
data 0x0136 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 .....
data 0x0148 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 .....
data 0x015A 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 .....
data 0x016C 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 .....
data 0x017E 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 .....
data 0x0190 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 .....
data 0x01A2 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 .....
data 0x01B4 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 .....
data 0x01C6 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 .....
data 0x01D8 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 .....
data 0x01EA 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 .....
data 0x01FC 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 .....
data 0x020E 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 .....
data 0x0220 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 .....
data 0x0232 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 .....
data 0x0244 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 .....
data 0x0256 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 .....
data 0x0268 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 .....
data 0x027A 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 .....
data 0x028C 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 .....
data 0x029E 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 .....
data 0x02B0 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 .....
data 0x02C2 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 .....
data 0x02D4 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 .....
data 0x02E6 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 .....
Call Stack
Breakpoints
Command Window
Immediate Window
Output
Memory 4

Announcements

- There is a tutorial video for this assignment.

Questions?