

Jeremiah Webb

CEC320 Workshop 6 Artifacts

Assembly Code Snippet

```
        AREA myfunctions, CODE

        EXPORT determine_data_order

        EXPORT add_uint64_s

        EXPORT sub_uint64_s

        EXPORT tc_uint64_to_int64_s

        ALIGN

determine_data_order PROC

        BX lr

        ENDP

add_uint64_s PROC

        ADDS R0, R2

        ADC R1, R3

        BX lr

        ENDP

sub_uint64_s PROC

        SUBS R0, R2

        SBC R1, R3

        BX lr

        ENDP

tc_uint64_to_int64_s PROC

        MVN R0, R0

        MVN R1, R1

        ADDS R0, #1

        ADC R1, #0

        BX lr
```

ENDP

END

Addresses

```
0x20000000: 255 255 255 255 119 119 119 135 000 000 000 000 050 084 118 152 002 000 000 000
0x20000014: 003 000 000 000 034 034 034 034 017 017 017 017 000 000 000 000 001 000 000 000
0x20000028: 001 000 000 000 123 119 119 135 001 000 000 000 123 119 119 135 222 221 221 221
0x2000003C: 032 067 101 135 222 221 221 221 032 067 101 135 000 000 000 000 255 255 255 255
0x20000050: 000 000 000 000 255 255 255 255 064 120 125 001 104 000 000 032 188 000 000 032
0x20000064: 016 001 000 032 000 000 000 000 000 000 000 000 000 000 000 000 001 002 000 000
0x20000078: 000 000 000 000 001 128 000 000 000 000 000 000 064 000 000 000 000 000 000 000
0x2000008C: 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000
0x200000A0: 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000
0x200000B4: 000 000 000 000 189 000 000 032 000 000 000 000 224 001 000 032 000 000 000 000
0x200000C8: 002 042 065 000 224 001 000 032 002 128 000 000 002 002 000 000 064 000 000 000
0x200000DC: 000 000 000 000 000 000 000 000 000 000 000 000 224 001 000 032 000 000 000 000
0x200000F0: 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000
```

Call Stack - Local | Data Register Window | Memory 1

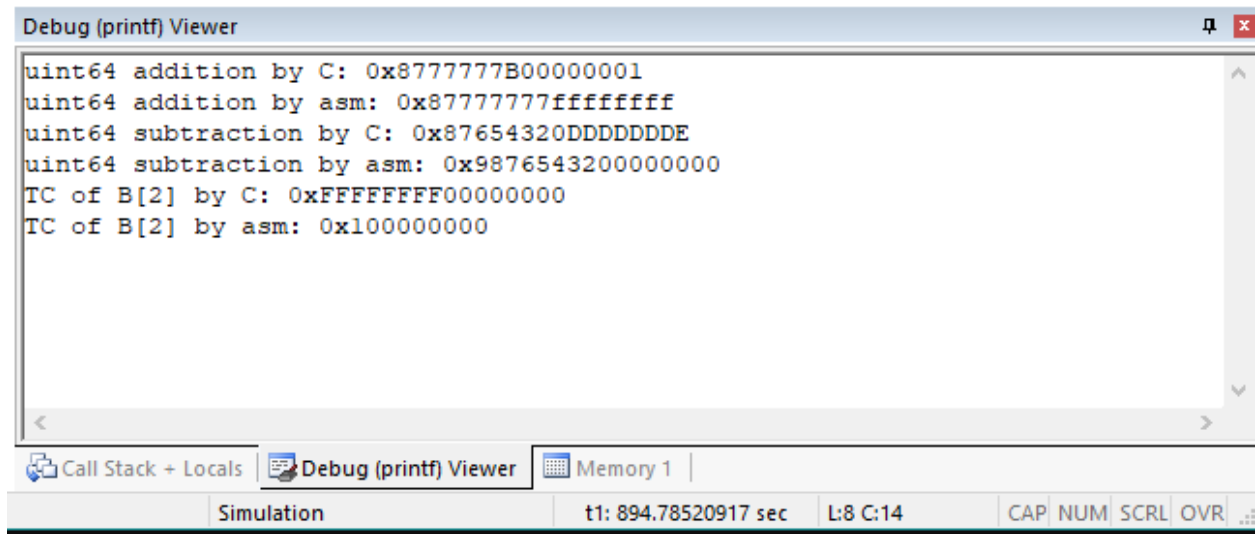
R0 = starts at 0x20000000

R1 = starts at 0x20000008

R2 = starts at 0x20000010

R3 = Starts at 0x20000018

Printf Viewer Task 1 Initial

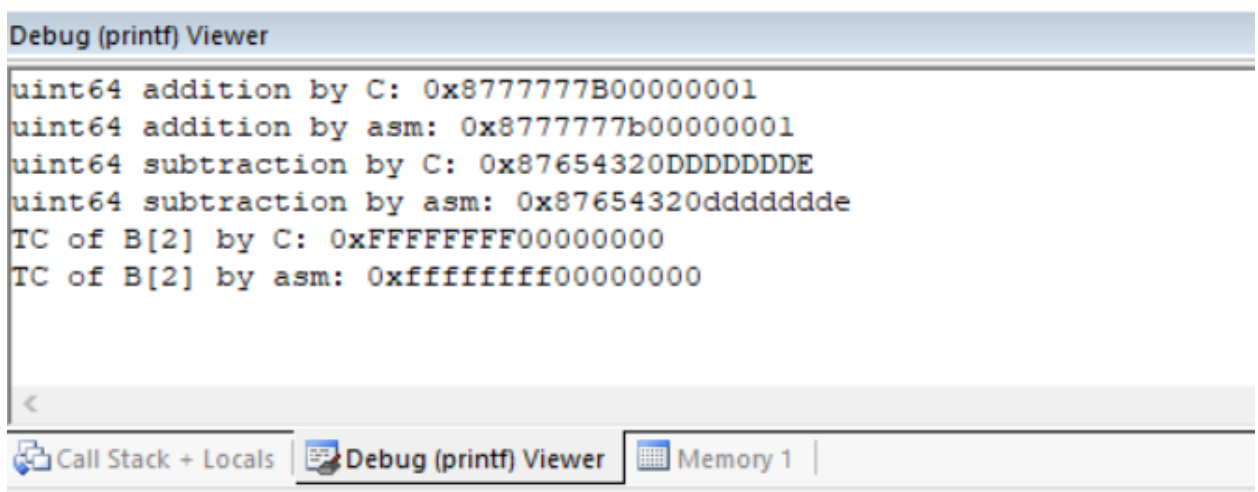


The screenshot shows the 'Debug (printf) Viewer' window with the following text:

```
uint64 addition by C: 0x87777777B00000001
uint64 addition by asm: 0x87777777fffffffff
uint64 subtraction by C: 0x87654320DDDDDDDE
uint64 subtraction by asm: 0x9876543200000000
TC of B[2] by C: 0xFFFFFFFF00000000
TC of B[2] by asm: 0x100000000
```

The window has tabs for 'Call Stack + Locals', 'Debug (printf) Viewer', and 'Memory 1'. The status bar at the bottom shows 'Simulation', 't1: 894.78520917 sec', 'L:8 C:14', and 'CAP NUM SCRL OVR'.

Printf Viewer Final



The screenshot shows the 'Debug (printf) Viewer' window with the following text:

```
uint64 addition by C: 0x87777777B00000001
uint64 addition by asm: 0x87777777b00000001
uint64 subtraction by C: 0x87654320DDDDDDDE
uint64 subtraction by asm: 0x87654320ddddddde
TC of B[2] by C: 0xFFFFFFFF00000000
TC of B[2] by asm: 0xffffffff00000000
```

The window has tabs for 'Call Stack + Locals', 'Debug (printf) Viewer', and 'Memory 1'. The status bar at the bottom shows 'Simulation', 't1: 894.78520917 sec', 'L:8 C:14', and 'CAP NUM SCRL OVR'.