

Lab 1 Basic Debugging with Keil: Lab Report

Student: Jeremiah Webb

Student ID: 2545328

Instructor: Dr. Jianhua Liu

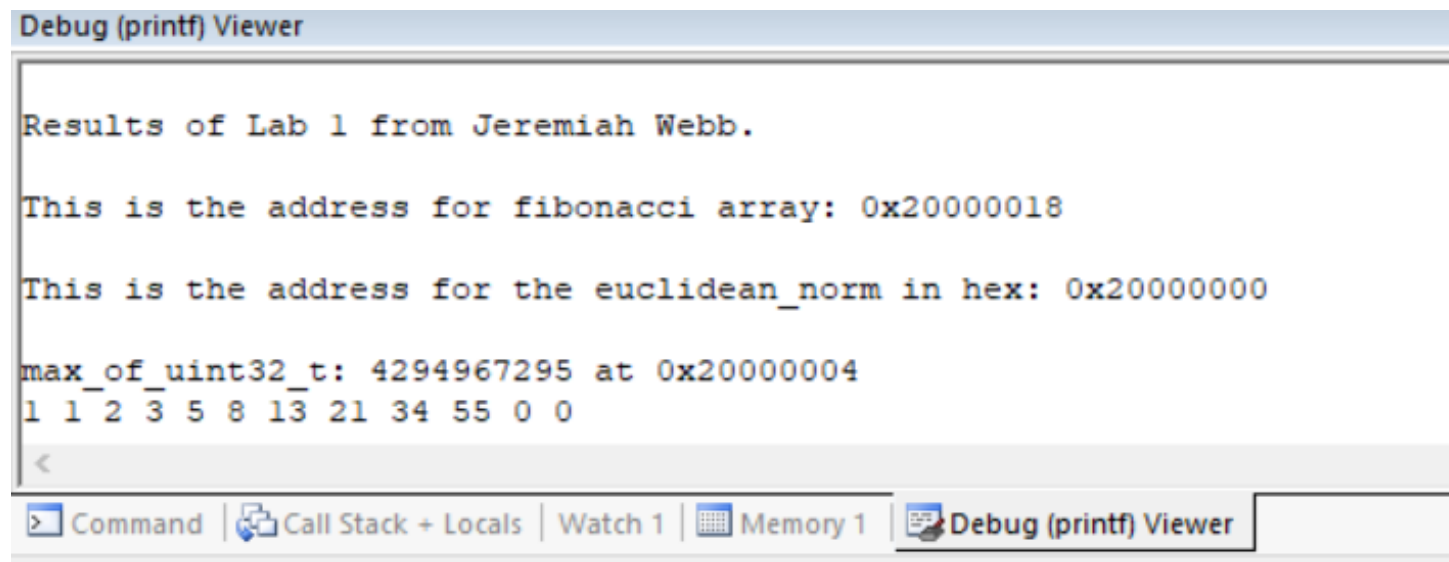
Section #2

Introduction

This lab was used to analyze and understand how students can debug their C programs in Keil MDK-ARM simulator. Students practiced basic debugging techniques such as using the debug (printf) viewer window, the Watch window to display local variables without the use of printing, and the Memory window to display global variables without the use of printing.

Report Artifacts

Report Artifact 1



```
Results of Lab 1 from Jeremiah Webb.

This is the address for fibonacci array: 0x20000018

This is the address for the euclidean_norm in hex: 0x20000000

max_of_uint32_t: 4294967295 at 0x20000004
1 1 2 3 5 8 13 21 34 55 0 0
<
```

For Artifact 1, we can see that the values of fibonacci_array[4] which is 5. However, for the euclidean_norm we cannot see the value, as instructed by form we were not required to print the value of the euclidean_norm. We can see the value of max_of_uint32_t, which is 4294967295.

Report Artifact 2a

Name	Value	Type
my_team	0x20001188 "Jeremiah Webb"	uchar[14]
norm	<cannot evaluate>	uchar
<Enter expression>		

The value of “my_team” is Jeremiah Webb, its type is a character array.

Report Artifact 2b

Watch 1		
Name	Value	Type
my_team	<not in scope>	uchar[14]
norm	69.9642792	float
<Enter expression>		

Command | Call Stack + Locals | Watch 1 | Memory 1 | Debug (printf) Viewer

The value of “norm” is 69.9642792, its type is a float.

Report Artifact 3

Memory 2									
Address: 0x20000000									
0x20000000:	000000069	4294967295	0025000000	0536870976	0536871060	0536871144	0000000001	0000000001	
0x20000020:	0000000002	0000000003	0000000005	0000000008	0000000013	0000000021	0000000034	0000000055	
0x20000040:	0000000000	0000000000	0000000000	0000000000	0000000000	0000000000	0000000000	0000000000	
0x20000060:	0000000000	0000000000	0000000000	0000000000	0000000000	0000000000	0000000000	0000000000	
0x20000080:	0000000000	0000000000	0000000000	0000000000	0536871060	0000000000	0000000000	0000000000	
0x200000A0:	0000000000	0000000000	0000000000	0000000000	0000000000	0000000000	0000000000	0000000000	
0x200000C0:	0000000000	0000000000	0000000000	0000000000	0000000000	0000000000	0000000000	0000000000	
0x200000E0:	0000000000	0536871144	0000000000	0000000000	0000000000	0000000000	0000000000	0000000000	
0x20000100:	0000000000	0000000000	0000000000	0000000000	0000000000	0000000000	0000000000	0000000000	
0x20000120:	0000000000	0000000000	0000000000	0000000000	0000000000	0000000000	0000000000	0000000000	
0x20000140:	0000000000	0536871328	0000000000	0000000000	0000000000	0000000000	0000000000	0000000000	
0x20000160:	0000000000	0000000000	0000000000	0000000000	0000000000	0000000000	0000000000	0000000000	
0x20000180:	0000000000	0000000000	0000000000	0000000000	0000000000	0000000000	0000000000	0000000000	
0x200001A0:	0000000000	0536871348	0536871328	0000000000	0000000000	0000003052	0000000000	0540221490	
0x200001C0:	0540549173	0840971057	0875765809	0540357920	0540024880	0807433313	0808465016	0808464432	
0x200001E0:	1763707444	1701322862	0807418488	0808465016	0808464432	0000002608	0000000000	0000002980	
0x20000200:	0000000000	0000000000	0000000000	0000000000	0000000000	0000000000	0000000000	0000000000	
0x20000220:	0000000000	0000000000	0000000000	0000000000	0000000000	0000000000	0000000000	0000000000	

In the yellow box one can see that the fibonacci_array[4] value is indeed 5. Euclidean_norm’s value is red box, which indicates it is 69. The max_ofuint32_t is 4294967295. These values check out as they match with Report Artifact’s 1 values.

Report Artifact 4

Memory 2									
Address: 0x20000000									
0x20000000:	0000000069	4294967295	0025000000	0536870976	0536871060	0536871144	0000000001	0000000001	
0x20000020:	0000000002	0000000003	0000000005	0000000008	0000000013	0000000021	0000000034	0000000055	
0x20000040:	0000000000	0000000000	0000000000	0000000000	0000000000	0000000000	0000000000	0000000000	
0x20000060:	0000000000	0000000000	0000000000	0000000000	0000000000	0000000000	0000000000	0000000000	
0x20000080:	0000000000	0000000000	0000000000	0000000000	0536871060	0000000000	0000000000	0000000000	
0x200000A0:	0000000000	0000000000	0000000000	0000000000	0000000000	0000000000	0000000000	0000000000	
0x200000C0:	0000000000	0000000000	0000000000	0000000000	0000000000	0000000000	0000000000	0000000000	
0x200000E0:	0000000000	0536871144	0000000000	0000000000	0000000000	0000000000	0000000000	0000000000	
0x20000100:	0000000000	0000000000	0000000000	0000000000	0000000000	0000000000	0000000000	0000000000	
0x20000120:	0000000000	0000000000	0000000000	0000000000	0000000000	0000000000	0000000000	0000000000	
0x20000140:	0000000000	0536871328	0000000000	0000000000	0000000000	0000000000	0000000000	0000000000	
0x20000160:	0000000000	0000000000	0000000000	0000000000	0000000000	0000000000	0000000000	0000000000	
0x20000180:	0000000000	0000000000	0000000000	0000000000	0000000000	0000000000	0000000000	0000000000	
0x200001A0:	0000000000	0536871348	0536871328	0000000000	0000000000	0000003052	0000000000	0540221490	
0x200001C0:	0540549173	0840971057	0875765809	0540357920	0540024880	0807433313	0808465016	0808464432	
0x200001E0:	1763707444	1701322862	0807418488	0808465016	0808464432	0000002608	0000000000	0000002980	
0x20000200:	0000000000	0000000000	0000000000	0000000000	0000000000	0000000000	0000000000	0000000000	

What one can see is that despite the changing of the order of the global variables they are all at the same addresses, as seen in Report Artifact 4 and in Report Artifact 3.

Code Snippets that were changed between Report Artifacts 3 & 4.

Initially:

```
//Stores the 1-11th values of the fibranachi sequence
uint32_t fibonacci_array[10];

//Stores the euclidean norm of the fibranachi sequence
uint32_t euclidean_norm = 0;

//Stores the max value of a uint32
uint32_t max_of_uint32_t = 0;
```

After:

```
//Stores the euclidean norm of the fibranachi sequence
uint32_t euclidean_norm = 0;

//Stores the max value of a uint32
uint32_t max_of_uint32_t = 0;

//Stores the 1-11th values of the fibranachi sequence
uint32_t fibonacci_array[10];
```

Narrative

Overall, the lab went well, there was some difficulty in ensuring the Euclidean function was working, however with help from the TA I was able to solve it. Otherwise, this lab was not too difficult and helped me understand the objectives. I enjoyed that we are being eased back into C programming and looking into the hard data and coding of C is helpful in visualizing how a program is running on a computer.

Results

In C programming we can see that debugging using Keil MDK-ARM simulator is extremely helpful in seeing how the computer runs a program from the high-level programming down to the assembly and binary. I learned how to use Keil effectively to do basic debugging for C. I now understand how to read the data outputted from a program based on addresses in hexadecimal.

Numerical Results:

Fibonacci_array[4] value is 5.

Value of max_of_uint32_t is 4294967295.

The value of "my_team" is Jeremiah Webb.

The value of "norm" is 69.9642792.

Euclidean_norm's value is 69.