***Analysis and Discussion:***

1) In what way is it good design of fixed.c that there is no arrow directly from the fixed.c module to the rit128x96x4.c module in the call graph for your system?

*A call to the rit128x96x4 module would be subverting the main function and the OLED output functions. Ignoring these controlling programs and interacting directly with the OLED driver could possibly corrupt a something that the driver is already trying to output under direction from the out functions.*

2) Why is it important for the decimal point to be in the exact same physical position independent of the number being displayed?

*When the decimal is in the same point, it makes the display “pretty” and readable because all the numbers are lined up.*

3) When should you use fixed-point over floating point? When should you use floating-point over fixed-point?

*You should use fixed point over floating point when the range of values is known and relatively small. Floating point should be used to accommodate a large range of values.*

4) When should you use binary fixed-point over decimal fixed-point? When should you use decimal fixed-point over binary fixed-point?

*You should use binary fixed-point when you are not expecting system interaction with a human operator. The Arm processor is very efficient in bit-shifting so binary-fixed point is much faster to process. When expecting system interaction with humans, you should use decimal fixed-point.*

5) Give an example application (not mentioned in this lab assignment) for fixed-point. Describe the problem, and choose an appropriate fixed-point format. (no software implementation required).

*Fixed point could be used to implement a simple kitchen timer. The range would be from 0:00 to 59:59 (Minutes:Seconds). For this application you would need a resolution of 0.01 and a way to alter basic arithmetic calculations to accommodate the 60-place value for seconds. The most appropriate format would be 32bit fixed-point decimal.*

6) Can we use floating point on the Arm Cortex M3? If so, what is the cost?

*No we cannot use floating point on the Arm Cortex M3.*

Extra credit) Is fixed-point or floating-point arithmetic faster on the Pentium w/MMX?

*Fixed-point arithmetic is faster on the Pentium w/MMX. Using fixed-point arithmetic (integer commands), we can execute two instructions at a time as opposed to only one floating point instruction.*