OHD General Programming Standards and Guidelines Peer Review Checklist

Reviewer's Name:	Charlie Becker	**Peer Review Date:**	02/14/18
Project Name:	**Module 2**	**Project ID:** Enter if applicable	
Developer's Name:	Taylor Dudunake	**Project Lead:**	
Review Files & Source code			
Code Approved	Yes		

This checklist is to be used to assess source code during a peer review. Items which represent the code being reviewed should be checked off.

Refer to the *OHD General Programming Standards and Guidelines* document for more complete descriptions and examples of the items listed below.

1. Internal Documentation

__X__ Comment block exists at the beginning of the source file containing at least the following information: original author's name, file creation date, development group, and a brief statement of the purpose of the software in the file.

__X__ Each subroutine or function in the file is preceded by a comment block which provides the following information: routine name, original author's name, routine's creation date, purpose of the routine, a list of the calling arguments

(their types and what they do), a list of required files and/or database tables, the routine's return value, error codes and exceptions processed by the routine, and a modification history indicating when and by whom changes were made.

2. Programming Standards

__-_ Consistent indentation of at least 3 spaces is used to distinguish conditional or control blocks of code. TABS NOT USED FOR INDENTATION.

__X_ Inline comments are frequently used and adequately describe the code.

__X__Structured programming techniques are adhered to.

__X__Subroutines, functions, and methods are reasonably sized.

__-__ The routines in each source file shall have a common purpose or have interrelated functionality. Methods in a class support its functionality.

__-__ The name of the file, script or class represents its function.

__-__ Function and method names contain a verb, that is, they indicate an action.

__X__Meaningful variable names are used.

__no__ Variables are initialized prior to use.

__X__ Braces are used consistently

__X__ There are no compiler warnings

3. Programming Guidelines

__X__Source file line lengths are 80 characters or less.

__X__Spacing is used correctly to enhance the source code's readability.

__-__ When continuing lines of code on new lines, they are broken after a comma or an operator. Higher level breaks are used instead of lower level breaks.

__-__ Wrapped lines of code are aligned with the beginning of the expression at the same level on the previous line.

__-__ Multiple line variable declarations are preceded by a type.

__X__Program statements are limited to one per line.

__X__Nested program statements are avoided.

__X__ Parentheses are used to remove operator precedence ambiguity and to improve code readability.

__X__ Inline comments constitute approximately 20% of the total lines of code in the program, excluding the file and routine documentation blocks.

__X__ The software reflects a balance of coding for efficiency and coding for readability.

__-__ Meaningful error messages are used.

__-__ System calls which acquire system resources are tested for error returns.

__X__ Routines and methods are contain no more than 200 executable lines.

__-__ The number of routines in a source file is kept to a minimum for procedural languages.

Reviewer's Comments:

Very nicely commented with a combination of markdown and in-line. Excellent "task flow" from plotting data -> histogram -> key stats -> trends -> autocorrelation -> overview. Perhaps some interpretation of autocorrelation coefficient values at example lags would be beneficial? Nice connection pointed out between trend and decaying autocorrelation function. Well done.