AB3D Embedded Software Testing / QA Standards

Testing should be designed to be destructive, that is, to prove the program doesn’t work!

All code that complies and is to be either submitted for permanent GitHub commit or used in a hardware testing application is to be ran through CPPCheck, and corrected accordingly.

By placing a trace statement, such as a *printf( ),* at the beginning of every block, you can track when the block—and by implication all the statements in the block—are executed.

All *prinf()* calls are to be *#define*(d) out of (or completely removed from) production code.

Functional Testing “Black Box”: Because black-box tests depend only on the program requirements and its I/O behavior, they can be developed as soon as the requirements are complete. This allows black-box test cases to be developed in parallel with the rest of the system design.

• Stress tests: Tests that intentionally overload input channels, memory buffers, SD card Module, memory management systems, and so on.

• Boundary value tests: Inputs that represent “boundaries” within a particular range (for example, largest and smallest integers together with − 1, 0, + 1, for an integer input) and input values that should cause the output to transition across a similar boundary in the output range.

• Exception tests: Tests that should trigger a failure mode or exception mode.

• Random tests: Generally, the least productive form of testing but still widely used to evaluate the robustness of user-interface code.

• Performance tests: Because performance expectations are part of the product requirement.

Coverage Tests “White box”: Coverage tests are devised with full knowledge of how the software is implemented, that is, with permission to “look inside the box.” White-box tests are designed with the source code handy.

• Statement coverage: Test cases selected because they execute every statement in the program at least once.

• Decision or branch coverage: Test cases chosen because they cause every

branch (both the true and false path) to be executed at least once.

• Condition coverage: Test cases chosen to force each condition (term) in a

decision to take on all possible logic values.