**AIISRR** Artificial Intelligence Infinity Software Reliability Regulation

AIISRR-C-: 2010

Regulations for the use of the C language in critical system

1. Development Tools
   1. Code Editor

Currently AII recommends use the latest versions of Microsoft Visual Studio and WTS Visual Assist X for code edit; use Source Insight for code browse but not for code edit. Other editors such as VIM, EMACS, eclipse and Ultra Editor, etc. are also available, but must be configured to compliant with MS VS + WTS VAX. For efficiency and project compatible consideration, Microsoft Visual Studio 2008 + WTS VAX are the standard IDE for most of projects of AII.

* 1. Compiler

**Windows Platform:** Only MS Visual Studio Compiler and Intel Compiler are regularly available for common projects. Any version and edition of MS Visual Studio and Intel Compiler must be verified by AII Technology Committee regularly. Other compiler must be verified by AII Technology Committee if any other special project requires.

**Linux Desktop Platform:** Only GCC are regularly verified by AII Technology Committee and are available for common projects. Intel Compiler are also available for Intel based platform.

**Linux Embedded Platform:** Only commercial regulated compilers or IDE integrated compilers verified by AII Technology Committee are available for projects of AII. Currently Windriver System and QNX are available for projects of AII.

**vxWorks Embedded Platform:** Only vxWorks 6.0 edition or above it are available for projects of AII.

**MAC OS X Platform:** Only xcode IDE are available for projects of AII.

* 1. Project version management

AII recommends Subversion for small scale projects and git for large scale projects. AII will introduce commercial version management solution later.

1. Project Structure
   1. General Folder Structure

The project folder should contain at least three directories: build, src, doc, whatever it is used for any type of project and in any development phase.

The folder “build” contains project definition files which are solution setting files in Visual Studio of Windows platform, Makefile in GCC platform or other project setting files of other platform. Different platforms must settle on different folders and name them with platform name. Standard name must be “linux”, “win32”, “macos”, “vxWorks”.

The folder “src” contains all the C source code and header files. The internal structure of “src” folder are slight different in different project type.

The folder “doc” contains all the development documents of the project. The development documents should be cataloged into different folder. Only MS office format and PDF are acceptable .

* 1. Application Integration Development

The folder structure of Application Integration Development must compliant with General Folder Structure. Other directories can be added depends on project requirements.

In folder “src”, creates an individual sub-folder for each module in which contains C code file and internal header files. The header files (in most of the cases just one head file for each module) which define interfaces and open data structure definition or objects must settle on public “include” folder of such project. The project only has one public “include” folder which contains the open header files of all the modules.

* 1. Library Development

The Library Development Project contains folders: “build”, “src”, “include”, “lib”, “doc”. The structure of folder “build” and “doc” are compliant with General Folder Structure.

The folder “src” contains all the C source code files and internal header files of each module.

The folder “include” contains all the public header files of each module.

The folder “lib” contains all the compiled dynamic link library files on every possible supported platform. The DLL files of every platform settle on individual sub-folder with platform name which must be standardized as: “linux”, “win32”, “macos”, “vxWorks”.

* 1. Module Development

The Module Development Project contains folder: “build”, “src”, “doc”.

1. Name Regulation

The module name which generally represents as the module folder name must be identical with module open header file.