Angotti Product Development *Project Improvement*

Accelerating Your High Tech Product Development Projects *Technical Assistance*

“Poor Mans” Configuration Management of Software, Firmware And Hardware Code

Prepared by Carl Angotti

May 8, 2001

Introduction

This memo defines a simple, low cost, method that can be used to control Software, Firmware and Hardware (FPGA or ASIC) Code revisions for a project. It can also be used for schematics and any other project documentation.

There are a number of packages, such as MS Source Safe, that can be used to maintain such code or documentation. On some projects, this can be difficult to implement in practise for a number of reasons, not the least of which is getting everyone on board for the same software and getting access to the database.

This approach illustrates a simple set of methods that can be used to control the Revisions of the code, the inter-operability of the code, and the code that is in process of being developed (WIP). It has been found useful in a number of contexts.

Set Up a Directory with Common Access

These days, with internet based storage facilities, this is not that difficult to accomplish, even for separate teams with widely dispersed geography. Some are free, others have a nominal fee associated with them. An example is “Xdrive.com”. There needs to be a common User Name and Password assigned. One person should be in charge of this. They make sure to enforce the approach and educate others that aren’t following the process. High Speed net access is very helpful, but not absolutely necessary to make this approach effective.

There is some concern, in some companies about security in such an arrangement. It is difficult to imagine someone outside those known to the team trying really hard to get such data, out of context, and not having some previous knowledge about existence of such a directory. My experience has been that this is no worse than any other method that shares data, since the User Name and Password could always get out somehow, with an entire team having access. Such concerns are often what makes it difficult to get such systems set up, “behind the firewall” of the organization. This approach is quick to set up, and workable in many instances.

Storage of Information

All code or other information will be stored on the Common Storage Area under a major functional directory such as “FPGA Code” with the sub-directories as follows:

```
Project Name (Main Folder)
        \FPGA Code
                \Gold Code Directory
                    Gold Log.txt
                    Gold Code1.zip
                    Gold Code2.zip
                        \Archive Directory of Gold Code
                \WIP Directory
                    WIP Log.doc
                    WIP1.zip
                    WIP2.zip
                        \Archive Directory of WIP Code
        \Firmware Code
                \Gold Code Directory
                    Gold Log.doc
                    Gold Code1.zip
                    Gold Code2.zip
                        Archive Directory of Gold Code
                \WIP Directory
                    WIP Log.txt
                    WIP1.zip
                    WIP2.zip
                        \Archive Directory of WIP Code

        \Software Code
                \Gold Code Directory
                    Gold Log.doc
                    Gold Code1.zip
                    Gold Code2.zip
                        Archive Directory of Gold Code
                \WIP Directory
                    WIP Log.txt
                    WIP1.zip
                    WIP2.zip
                        \Archive Directory of WIP Code
```

Other major heading could be: Specifications, Bills of Materials, etc.

Project Name – This can be used to differentiate information stored for each project.

Directory Headings – ie. \Hardware\FPGA are selected to illustrate the code content of the Directory. These are defined in some logical manner related to the organization producing the data, or some easily defined context.

First Sub-Directory Headings – These are divided into two:

> **Gold Directory** – This directory contains the reference code files that have been widely tested and are the most mature.
>
> **WIP (Work in Process)** – The information in this directory is that which being currently worked on. It has not been widely tested. It will contain all code that is not "Gold".
>
> This directory can be used to contain drops of all code being worked on say, once per week, and more often for significant changes, but no more often than say, once per day. This can give managers visability into what is going on in a specific area of the project. I have found this useful by employing "file compare" programs to look at information variations.

Second, Archive, Sub-Directories – These directories contain code that is not currently considered of interest. This is Gold or WIP code that has aged so as not to be useful, but not lost. The person generating the code can use their judgment as to when Code in the First Level directory reaches this status.

Eventually, code in the archive directory will be deleted, at the option of the person generating the code or another designated person.

Log Files – Gold and WIP Directories will contain log files, that briefly describe the files in its directory and Archive sub-directory. This file is updated each time a new file is added to a directory or sub-directory. This file will be a .doc file that describes the file name, date, person generating the code, a brief description of the file content or changes and whether it is in the Gold (G) or WIP (W) directory.

ZIP Files – All files for a particular date are to be zip compressed into a single file using WinZip or compatible compression Utilitiy. The zip file will contain all of the files contained in a "release" of the code.

The zipped file will also contain a **Readme.doc** file that describes the contents of the Zip file. This file should contain, but not be limited to:

1. The Originator of the file

2. The date of the file
3. The revision number of the file

4. Files that are associated with this file, ie FPGA files that must be used with these files, etc. to have the code work properly.
5. A description of the earlier code drop upon which this version is based
6. A description of this file, including changes made, purpose of the change or functionality added.
7. Any other information that might be of interest for later historical purposes.

File Naming Convention

A file naming convention for any project file can be very helpful. In modern projects today, this has literally become a necessity. An example of a file naming convention could be as follows:

Date_Revision_Primary-Type_Description.zip

Date = Date in month, year date format – 040201 for April 2, 2001
Revision = Revision number to two decimal places – XX.XX
Primary Type = FPGA, Firmware, etc.
Description = Fix LED Problem (Minimize No. of Characters)

An example would look like:

040201_Rev 1.00_FPGA_Fix LED Problem.zip or
040201_Rev 2.12_FPGA_Gold.zip

Making the Process Useful

Start the process as early in the project as possible, because it takes time, and there can be considerable resistance to switch once a project is underway. Create a memo, that can be based upon this one, and make it a team requirement early in the project.

Note: You can receive a template of this document, in Word format by requesting it from me at carl@angotti.com.