CHAPTER 7

CONCLUSION AND SUGGESTION

7. Conclusion and Suggestion

7.1 Conclusion

The study of the quality assurance system for DCS project started from the analysis of the current project execution. The flowchart of the current DCS project execution was drawn and discussed in the chapter 4.1. The data collection of the previous projects has been collected and shown on the chapter 4.2.

The engineering tools such as the failure mode and effect analysis (FMEA) and the fault tree analysis (FTA) were used in the problem analysis in the chapter 4.3 and the result of the analysis was shown on the appendix II. From the analysis, there are 23 high-risk areas in the DCS project execution (Table 4.19). The high-risk area is defined as the process where its risk priority number or RPN has the value exceeds the threshold of examining failure (100).

The main problems in the DCS project execution can be separated into three major items as below.

1. Lack of Procedures to Control the Project Execution
    - Poor Contract Verification
    - Product Defects and Delayed Delivery
    - Incomplete Delivery
2. Lack of the Document and Data Control
    - Poor Document Control
    - Insufficient Procedure to Build Software Specification
    - Design Changes
3. Technical Problems
    - Capacity Problems
    - Poor Design
    - Limited Know-how

The FMEA team has discussed for ways to solve these problems and recorded the solutions into the recommended action column of the FMEA form. There are 28 documents to be established and 1 software tool for keeping the engineering database to be developed (Table 4.20). In addition, the procedures to execute the DCS project have been revised as discussed in the chapter 5: the establishment of the required quality assurance activities. The newly revised procedures as well as the control documents and the software tool (EDP) serve as the quality assurance system for the execution of distributed control system project.

The proposed quality assurance system for the distributed control system has been implemented in one DCS project in July of 1998 (chapter 6.1). The results of the implementation were the improvements in software quality and on-time delivery. 37 software errors were detected during the pre-inspection test and there were only 8 errors during the inspection test with the customer, comparing to the average of 50 to 60 errors found during the inspection test in the past projects. In the view of changes in RPN of each process of the project execution, the percentage of change of RPN comparing between before and after implementation ranges from 33 to 94%.

The evaluation of the quality assurance system was discussed in the chapter 6.2 on the following topics:

- Understand the Scope of Work and Customer Requirements Clearly
- Project Monitoring with the Walkthrough Form: Pros and Cons
- Design Planning: Understand Work to be done and Schedule to finish Clearly
- Validation of Designs: Detect Errors at the Earlier Phases
- Unit Test Records: Ensure that the Application Software has been tested
- Configuration Management Practices: the Process of Change is managed
- The Storage of Test Data: Easy for Searching the Past Software Data
- Rationale for Decision-Making: Deeply Thinking during Software Design Task is developed
- Improvements in the Project Activities
    1. Time Reduction During the Ordering Procedure
    2. Proper Way of the Media Handling Procedure
    3. Time Reduction in the Document Searching
- Engineering Database Pool Software: Reuse Software and Enhance Engineering Knowledge

From the implementation of the proposed quality assurance system in one project, I have found that there still are some points that should be further developed in order to improve the quality assurance system for the DCS project even better. These points are discussed in the following section.

7.2 Suggestion

There are some suggestions to the quality assurance of the DCS project execution and they are as following.

(1) Trade off the Increasing of Paper Work against the Decreasing of Project Work

When implementing the proposed quality assurance system in the RCA2 project, it seems that the project team has to work harder than they have worked before in previous projects. That is, the project team has to do both the project work (design software) and the paper work (all quality document and checklists) during the project execution. The team might think that the quality assurance system for the DCS project has increased their works.

To avoid this problem, it is necessary to explain clearly to the persons who use the checklists about the benefits of the quality assurance system. With the use of newly revised project procedures, the team has to plan before doing jobs, monitor the jobs in detail, verify the work whether they are correct, and validate the work whether they are the same as the customer's requirements. Though all these works have increased their works, but all these quality assurance activities have also decreased their project working time in the view of the redesign and rework due to designing errors. Since the design work is checked at earlier phases of the project execution, it is easier to correct these errors than to correct them at final phases which are complex and more difficult.

In order to use the quality assurance system effectively and avoid resistance of using these checklists, the staffs who use checklists should be involved during designing the quality assurance system and checklists so that they feel that these document are created for themselves and they have their sense of ownership. However, it is important to note that it is quite difficult to apply the quality assurance system at the working place where the rate of turnover is high or where the staffs have negative feeling with the quality assurance system.

(2) To Continually Update the Quality Assurance System

It is important to understand that the quality assurance system is an evolving entity. Since there is no the best quality assurance system that the company can use it forever, the company should evaluate the effectiveness of the quality assurance system periodically. For example, if the quality assurance system has been successfully implemented in the project execution, the threshold of the pursuing failures should be decreased to another level. That is, 95 percent of all failures must be addressed (increased from 90 percent to 95 percent) or, in other words, to address the potential failure that its RPN exceeds 50.

In addition, the company should also evaluate the potential changes that new software development technology and business policy might have on the quality assurance system (Ince, 1994: 35-36). The software errors recorded in the punch lists should also be analysed and used to improve the technical processes and the quality assurance system.

(3) To Apply the Quality Assurance Activity to the Whole DCS Process

The figure below illustrates the whole process of the DCS project execution.

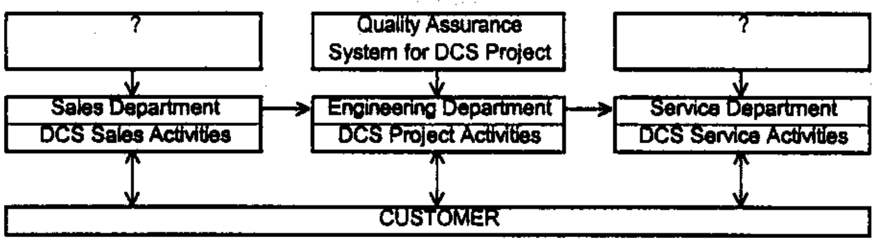

Figure 7.1 Whole Process of the DCS Project

In order to fully satisfy the customer, it is better to apply the FMEA and FTA techniques to the whole process and establish the quality assurance system for the whole DCS project, starting from the sales to the service department. It is not good enough if the engineering department supplies the good DCS product to the customer whereas the service department provides poor maintenance and shutdown service.

For further improvement of the quality assurance of the DCS project in the engineering department, the following items should be developed:

(1) To Establish the Functional Traceability in the Quality System

After the project ends, the customer may require some modification of the DCS software in order to improve the process's capability, reduce bottlenecks in process, or add new functions into the process. The quality assurance system should establish the linkages between programme code and the software functional specification so that the maintenance of the DCS software can be done easily. The functional traceability is the ability to find the relationships between the programme source code and the software functional specification in a short time (Ince, 1994: 16, 34). The idea of relationship between the application software and the software functional specification is shown of the following figure.

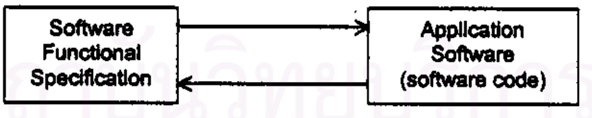

Figure 7.2 Relationship between Specification and Application Software

If a software developer or a system engineer in the engineering department is asked for the self-documentation of the application software (detail programming source code) and one person points to a module in the self-documentation and asks for the relationship of this module with the software functional specification. This means, the software developer is requested to tell him which functions in the software functional specification that module helps to implement. If the software developer takes hours to answer the question, this indicates that the software developer will have problems, during the software maintenance (Ince, 1994: 34).

This function will also help engineers during the project execution. Suppose that there are some changes on the functional specification and the functional traceability is already established in the quality system, the engineer of that project can trace to the related source codes and make the modification easily. This encourages a reduction in errors and an increase in efficiency.

(2) To Plan for System and Acceptance Testing in Advance

In order to ensure that the problems do not exist during the system and acceptance testing, it is better to plan for the testing in advance. As soon as the software functional specification has been validated as being correct, the system and acceptance tests (called FAT in ABC Company) should be developed in outline (Ince, 1994: 36).

There are many reasons for this planning. The project manager wants to know about the level of resources required during the testing and can only calculate this if the project manager has a good idea of what the testing involves. In addition, there may be a need for special-purpose hardware for testing which needs to be fabricated in advance; and there could also be a requirement for special-purpose testing tools which need to be specified and developed in advance (Ince, 1994: 36).

(3) To Establish the Risk Analysis in the Project Execution

To improve the quality assurance system even better, it is necessary to establish the standards and procedures for identifying the possible risks in the project as well as to establish the guidelines which provide advice on how to minimise the effect of those risks into the quality assurance system. The careful planning at the earlier phase of the project will help the project team to mitigate the effect of the risky events (Ince, 1994: 43-46). According to Boehm (1989), the top ten risks in a project are as follows:

- Staff deficiencies
- Unrealistic schedules and budgets
- The wrong software functions may be developed
- The wrong kind of interface may be developed
- Over-engineering
- Requirements volatility
- Deficiencies in externally developed items
- Shortfalls in externally performed tasks
- Performance problems
- Strains on current computer technology

The risk assessment can be carried out using a checklist which would ask questions about the ten areas above. Some questions are shown below (Ince, 1994: 44-45).

- Is this the first time that the project manager has managed a project?
- Is this the first time that the project manager has managed a project in this application area?
- Are any of the items developed by external subcontractors critical items, for example, in performance terms?
- Is there a wide variety of users interfacing with the system?
- Is the hardware technology upon which the application is to be run new?
- Have we achieved a similar response-time on comparable projects in the past?
- Is this the first time that the customer has been exposed to computer technology?

(4) To Prevent Further Errors during the Inspection Test

During the inspection test, the errors in the application software are detected and solved. It is not enough just to rectify those found errors, repeat the test at place where errors were found, and carry on. In practice, there is high probability that the place where we rectify the errors may affect to several places of the application software, and will have caused further errors which can only be discovered by rerunning previous tests. It is necessary to establish the testing procedure, to rerun the previous tests when errors are found, into the testing document. This is to ensure that further errors do not exist.

(5) To Develop Standards and Procedures for Auditing

In order to check how seriously each DCS project team takes quality, it is necessary to have standards, procedures, and guidelines which govern the process of checking that projects follow standards and procedures. These standards, procedures, and guidelines should cover the process of auditing, when to audit, how frequently to audit, what reports are generated from the auditing process, and what actions are taken when a DCS project fails an audit. Since the proposed quality system with good standard for activities and review checklists will not be effective if we omit the auditing task in the project execution (Ince, 1994: 39).