

QUALITY ASSURANCE MANUAL

**MEng Year 3
Department of Electronics
University of York**

Software Engineering Group Project

The Really Exciting Company

Document Control

Version	Modified by	Date	Section modified	Remarks
3.2	AMT	Oct 2012	All	Updated
3.3	GT	Oct 2012	All	Minor changes and updates
3.4	SJP	Jan 2013	Template	Adapted from Digital Systems Engineering Template

CONTENTS

1. INTRODUCTION

1.1. Company Profile	1
1.2. Vision	1

2. ROLES & RESPONSIBILITIES

2.1. Organizational Structure.....	2
2.2. Group Leader	3
2.2.1. Role Description.....	3
2.2.2. Risk Management.....	3
2.2.3. QA Metrics.....	4
2.3. QA & Documentation Manager	5
2.3.1. Role Description.....	5
2.3.2. Risk Management.....	6
2.3.3. QA Metrics.....	6
2.4. Design & Specification Manager	7
2.4.1. Role Description.....	7
2.4.2. Risk Management.....	7
2.4.3. QA Metrics.....	8
2.5. Software Manager	9
2.5.1. Role Description.....	9
2.5.2. Risk Management.....	9
2.5.3. QA Metrics.....	10
2.6. Hardware Manager	11
2.6.1. Role Description.....	11
2.6.2. Risk Management.....	11
2.6.3. QA Metrics.....	12
2.7. Testing & Integration Manager	13
2.7.1. Role Description.....	13
2.7.2. Risk Management.....	13
2.7.3. QA Metrics.....	14
2.8. Finance Manager	15
2.8.1. Role Description.....	15
2.8.2. Risk Management.....	15
2.8.3. QA Metrics.....	15
2.9. Marketing Manager	17
2.9.1. Role Description	17
2.9.2. Risk Management	17

2.9.3. QA Metrics	17
3. DELIVERABLES	18
4. PROJECT MANAGEMENT METHODOLOGY.....	20
4.1. Requirements and Specifications Phase	20
4.2. Design Phase	21
4.3. Implementation Phase	22
4.4. Testing and Integration Phase	22
4.5. Quality Auditing Reviews	23
5. Appendix A: Documents Templates	25
Meeting Minutes	26
Manager Reports	27
QA Metric List	28
Implementation Progress Report	29
Test Report	30
6. Appendix B: Design Methodologies	31
State Diagrams	31
UML Class Diagrams	31
Use Case Diagram	32
Gantt Chart	33
7. Appendix C: Implementation Examples	34

1. INTRODUCTION

1.1. Company Profile

The company was set up as a provider of entertainment systems solutions. Our major business focus is on the design, production and support of portable consumer electronics using FPGA devices. Our design specifications comply with industry standards and our products emphasis is on innovative solutions as well as dedicated product support.

Based in York, we are a team of highly qualified engineers constantly researching new ways of providing high quality products with strong attention to reusability and sustainability. Our new state-of-the-art design and manufacturing laboratories includes support for integrated circuit development.

1.2. Vision

- ✓ “To be the foremost manufacturer of FPGA-based devices, creating innovative solutions and supporting environmental sustainability”
- ✓ “To bring reliable consumer devices to every home”
- ✓ “To integrate modern technologies and design into systems”
- ✓ “To be the world leader in FPGA technologies for consumer products, driving advancement in all areas of life”
- ✓ “To provide affordable quality products in a green environment”

2. ROLES AND RESPONSIBILITIES

2.1. Organizational Structure

In order for any group of workers to achieve their set objectives, responsibilities and roles must be well identified. Figure 1 shows the structure of the company (although not all of these posts may be occupied at any one time¹). The Group Leader holds the responsibility to ensure the harmony between group members and that company policies and procedures are adhered to by all members of the company. As one goes down in the figure, the roles change into more operational nature, through which the actual products are designed and strategies are applied. Communication between team members is achieved mainly through weekly meetings and review/progress reports. Reviews at each milestone are also critical to evaluate the quality procedures that were followed, and to inspect deliverable for errors and defects.

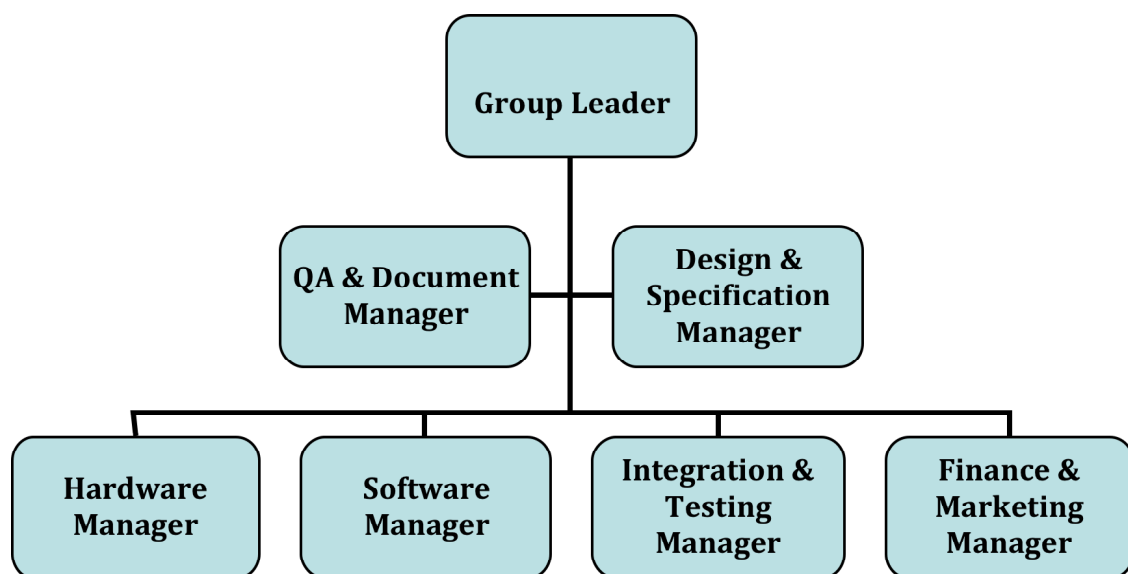


Figure 1: The structural organization of project group

¹ It should be realized that whilst not all of these posts need to be occupied all the time, the functions as described in the role descriptions of each post do need to be kept in mind in all aspects of the project.

2.2. Group Leader

2.2.1. Role Description

The group leader is responsible for taking an overseeing role within the company and organising the working of the team toward the desired goals. He/she is accountable for insuring that the design and development process runs smoothly during the project life cycle (section 4) until the final product is presented to the customer. Accomplishing this involves several tasks that include:

- ❖ Arranging for regular meetings with the team and assigning the Chair in each meeting.
- ❖ Provide meeting agendas (see Appendix A for templates).
- ❖ Assigning the different roles required in the project in a way that ensures the optimal use of every member's qualifications is reached.
- ❖ Maintaining an encouraging and cooperative environment where every individual contribution is appreciated and respected.
- ❖ With the help of the Design & Specification Manager, provide the plans for the project development and monitor the progress.
- ❖ With the help of the appropriate managers, provide the timetable plan and ensure it's met, or make appropriate modifications as required.
- ❖ Make sure that the requirements and standards are met at every stage whether during the development process itself or when the final product is near completion.
- ❖ Provide solutions and backup strategies as required.
- ❖ Grant final approval of all the deliverables provided by the team.
- ❖ Deliver reports to customers and act as the connection between them and the project team.
- ❖ Produce the overall project plan and time schedule.

2.2.2. Risk Management

Risk	Possible Solution
Some group members don't get along with each other.	Try to resolve any issues that group members might have. If this does not work, consider assigning tasks to minimise disruption within the group as a whole.
The prolonged absence of a member in the group.	Have more than one team member trained to do any task. Be ready to adjust schedules as appropriate.
Deadline overdue.	Have regular project plan review meetings

	where possible overruns can be identified and rescheduling can be considered.
Failure to meet one of the requirements.	Have regular review meetings where requirement conformity is checked. Make sure that the design and the development of the product is organized in a way that simplifies rollback and re-implements the missing requirements with as little modification to other parts as possible.

2.2.3. QA Metrics

Metric	How measured
Client requirements clarity.	Holding regular meetings ensuring that the client understands his/her own requirement statements and that these have been interpreted correctly by the company and finally signed off by both sides.
Complete specification.	Checking the specifications against the requirements statement to make sure all requirements are covered by the specifications team, during appropriate review meeting(s).
Deadlines met.	Deliverables submitted as timetable and recorded.

2.3. QA & Documentation Manager

2.3.1. Role Description

Quality Assurance works through procedures and processes to ensure the products produced by the company are of high quality that meets the customer requirements. The QA manager is responsible for developing processes and procedures that can be followed throughout the complete life cycle of a product to ensure the quality of the product through the adherence of these processes and procedures, including setting up, and monitoring, an appropriate document change management system (see appendix A). Accurately carrying out these sets of processes and checks requires maintaining regular interactions and communications between the QA & Documentation manager and other key members in the project team.

The communication between different members of the group is done mainly through review meetings, regular group meetings, various types of documents and reports that are designed to help members follow the company's QA procedures and allow the QA & Documentation manager to check such procedures are being followed. As projects progress and experience is obtained QA procedures and process may change, but these changes will only be made after a QA review meeting, the approval of the QA & Documentation manager and Group Leader and following the updating of the appropriate procedure document. Tasks for the QA & Documentation manager include:

- ❖ In conjunction with the Group Leader define the quality standards for the company.
- ❖ Provide the QA plan that explains what processes should be followed in any project undertaken by the company, who should do it, what procedures and methods should be used, and what standards of quality and performance to follow.
- ❖ Specify the QA metrics against which the quality of the result must be measured in each stage of the product development.
- ❖ Monitor the use of the QA processes through the collection of QA metrics.
- ❖ Provide updated reports to the Group Leader about the system progress and the efficiency in following the quality procedures.
- ❖ Carry out Review processes at the end of each development stage to ensure the quality and address any shortfalls in process.
- ❖ Specify the different types of documents and reports to be delivered throughout the product life cycle and specify when these should be produced.
- ❖ Ensure that all documents follow the standards set by the company, e.g. in term of their style, format, drawing and figure clarity, etc.

- ❖ Archive all records related to a project.
- ❖ Track all the updates on company documents.
- ❖ Monitor documents delivery deadlines.
- ❖ Ensure that the most up-to-date versions of all documents are available and used.
- ❖ Ensure all meetings are appropriately managed with records taken of proceedings.
- ❖ Ensure all actions are completed in a timely and qualitatively acceptable way.

2.3.2. Risk Management

Risk	Possible Solution
Missing or corrupted documents.	Have a backup copy whenever any document is added or updated.
Failure to carry out a procedure.	Hold regular review meetings to consider QA procedures. Consider QA procedures on a regular basis and assess their appropriateness, updating as required.
QA metric not met.	Set up continued monitoring of staff's adherence to QA procedures and regular, monitored, deadlines for metrics.
Incompatibility between two developed modules.	Ensure the most up-to-date versions of all documents are available and used.

2.3.3. QA Metrics

Metric	How measured
Documents delivered.	Number of documents expected Vs delivered.
QA metric collection.	Number of QA metrics met Vs expected.
Deadlines met.	Deliverable submission Vs timetabled.

2.4. Design & Specification Manager

2.4.1. Role Description

The need to focus on a detailed product design and specification is paramount to the delivery of a high quality end product. To ensure this, a large proportion of the company's budget goes into the research of the product's design. Our design team develops novel design concepts and collaborates with the design teams of some of our major clients. We also pride ourselves in knowledge partnerships with the Electronics Department of the University of York. Specific responsibilities of the Design and Specification Manager include:

- ❖ Use specialist knowledge to establish requirements for the product from the customer specification.
- ❖ Develop design procedures and guidelines to ensure consistency of all design work against these procedures and guidelines.
- ❖ Establish procedures for product design, evaluate and develop improved techniques for the control of product designs, quality, reliability and safety.
- ❖ Lead/manage delivery of all design work for the company, providing design guidance and feedback to other company departments involved.
- ❖ Plan, monitor and control the efficient and cost effective flow of orders through the department and ensure deadlines are met by taking into account design flow priorities, budget and time of production.
- ❖ Develop comprehensive creative design development processes, and roll these out across the company, and with suppliers, including clear briefing and signoff guidelines.
- ❖ Manage and oversee all roles and activities in the design area of the company.
- ❖ In association with the Group Leader, delegate duties when appropriate and monitor individual's performance so staff are able to perform to a high level of attainment and satisfaction.
- ❖ Meet on a weekly basis with the other managers of the company to discuss project progress.

2.4.2. Risk Management

Risk	Possible Solution
Delay in design completion.	Report to the group working on next stage of the project. Consider re-allocation of tasks to conclude critical parts of the design flow. Review why delay has occurred and identify the reasons.

Unacceptable error margins on product design specifications.	Communicate with the production team, and recheck the design margins adjusting these where appropriate.
Non-conformance of product to stipulated standards.	Identify which part of the product is non-conformant. Analyse and rectify according to the standards.

2.4.3. QA Metrics

Metric	How measured
Design Cost.	Suitable measure should be chosen, such as time, man-power etc.
Design Appropriateness.	Correct translation of specification.
Module/Component Interoperability.	Estimate of the cohesion among different components in the design.
Defects.	Number of defects identified and stage when identified.
Deliverables Submitted/ Reviewed/Accepted/Rejected.	Number and percentage of design elements that are released and related to number that are accepted and rejected.

2.5. Software Manager

2.5.1. Role Description

The software manager is responsible for planning, designing and coordinating the implementation of the software components of a project. He/she should have a full understanding of the specifications and the requirements of the project in order to develop the desired product. Also, he/she should be in regular communication with the QA manager to ensure the quality of the software design, as well as with the Hardware Manager to ensure the compatibility between the hardware and software components of the product. The Software Manager tasks should include:

- ❖ Develop a clear overall software design based on the specifications and the customer requirements.
- ❖ Decide on the methods and the techniques to be used in the design phase.
- ❖ Identify the details of the specification, design and implementation, such as the programming language, interfaces, modules, objects, data encapsulation, and hierarchy.
- ❖ Monitor the implementation and the coding process.
- ❖ Deliver appropriate documents regarding the high-level and low-level designs.
- ❖ Make sure the design standards are followed and the procedures are applied to keep consistency between different parts of code generated by different members of the implementation team.

2.5.2. Risk Management

Risk	Possible Solution
Failure of code at some point.	Consider during regular review meeting and assign appropriate manpower to correct failure, based on criticality of code. Cooperate with test and integration manager to provide complete set of test cases.
Not compatible with hardware or with other parts of system.	Develop the design in collaboration with specification/design manager and hardware manager to ensure compatibility.
Code inconsistency.	Provide specific standards for coding in terms of format, language, naming conventions etc., and keep a reference of all the additions to the code and the relation between them (e.g. in the form of charts and/or class diagrams).

These issues are to be communicated with the Group Manager and discussed with group members by the earliest to coordinate any rescheduling. The Testing and Integration Manager is also to be notified.

2.5.3. QA Metrics

Metric	How measured
Actual coding time Vs planned.	Register the start and end of coding.
Number of actual code lines Vs planned.	Keep count of code lines produced by every member in the implementation team.
Comments in code.	Consider line of comments compared to lines of executable code.
Compiling errors.	Produce a compilation report to indicate the number of errors during compilation and whether code is error-free and tested.

2.6. Hardware Manager

2.6.1. Role Description

The Hardware Manager is responsible for planning, designing and coordinating the implementation of the hardware development for products. He/she should have a full understanding of the specifications and the requirements of the project in order to develop the desired product. Also, he/she should be in regular communication with the QA manager to ensure the quality of the hardware design, as well as with the Software manager to ensure the compatibility between the hardware and software components of the product. Key responsibilities of the Hardware Manager include:

- ❖ Develop a clear overall hardware design based on the specifications and the customer requirements.
- ❖ Decide on the methods and the techniques to be used in the design phase.
- ❖ Identify the details of the specification, design and the implementation details such as the hardware models, interfaces, sub-sections and hierarchy.
- ❖ Monitor the implementation and the hardware implementation.
- ❖ Assess the quality of the produced hardware.
- ❖ Deliver appropriate documents regarding the high-level and low-level designs.
- ❖ Make sure the design standards are followed and the procedures are applied to keep consistency between different parts of hardware produced by different members of the implementation team.

2.6.2. Risk Management

Risk	Possible Solution
Failure of hardware at some point.	Consider during regular review meeting and assign appropriate manpower to correct failure, based on criticality of the hardware. Cooperate with Test and Integration manager to provide complete set of test cases.
Not compatible with software or with other parts of system.	Develop the design in collaboration with specification/design manager and software manager to ensure compatibility.
Hardware inconsistency.	Provide specific standards for hardware in terms of format, design models, naming conventions etc., and keep a reference of all the additions to the hardware design and the relation between them (e.g. in the form of state machine).

These issues are to be communicated with the Group Manager and discussed with group members by the earliest to coordinate any rescheduling. The Testing and Integration Manager are also to be notified.

2.6.3. QA Metrics

Metric	How measured
Number of modules successfully implemented as compared to the total number of modules.	Keep record of number of modules implemented successfully.
Number of specifications met as compared to specifications assigned.	Measure specifications successfully met, appropriately, and compare to original specification.
Number of bugs and errors in hardware compilation.	Measure during implementation.
Actual coding time (FPGA's for example) as a percentage of planned coding time.	Keep record of coding time and compare when modules completed with planned time.

2.7. Testing & Integration Manager

2.7.1. Role Description

The responsibility of the Testing and Integration manager is to manage the procedures involved with the testing and integration phases of a project. He/She should have a holistic view of the testing and integration phase in order to competently perform their responsibility. The main role of the integration and testing manager is to make an effective plan for integration and testing of the project. Integration and testing Gantt charts must be produced to show how the plans are scheduled. An example of integration and testing Gantt chart is shown in Appendix B. In relation with the Gantt chart, the tasks can be assigned to group members and results from team members collected at the appropriate deadline. The results will be collected by the integration and testing manager for them to produce testing and integration reports. The integration and testing manager should also produce test sheets for all team members to use during implementation stages of a project. These should be collected by the integration and testing manager and reports produced based on these completed sheets. An example testing report is included in Appendix A. The report will be delivered to the Group Leader for input to review meetings. The key responsibilities of the integration and testing manager include:

- ❖ Manage the procedures in the testing and integration phases.
- ❖ Make an effective integration and testing plan.
- ❖ Represent and schedule the plan by producing integration and testing Gantt chart.
- ❖ Assign tests to team members according to schedules as outlined in the project Gantt chart.
- ❖ Collect results from team members at the deadlines request.
- ❖ Produce testing and integration reports.

2.7.2. Risk Management

Risk	Possible Solution
Implementation phase overdue.	Reschedule integration and testing plan.
Any part of testing and integration overdue.	Report the reasons for overrun and also reschedule integration and testing plan.
Error occurs, but cannot find which part of design is the problem.	Communicate with Software and Hardware managers to identify the exact problem.
Significant issues related to the integration of one or more modules.	Communicate issues with Software and/or Hardware managers to fix problem.
Modules developed to out-of-date	Ensure the most up-to-date versions of all

module specification and/or design documents.	documents are available and used.
---	-----------------------------------

These issues are to be communicated with the Group Leader and discussed with group members by the earliest to coordinate any rescheduling. The Software and Hardware managers are also to be notified.

2.7.3. QA Metrics

Metric	How measured
Integration plan followed.	Monitor and assess integration throughout the project time.
Testing plan followed.	Monitor and assess testing procedures followed throughout the project time.
Number of modules tested.	Collected from test reports.
Errors and time to fix these.	Collected from test reports.

2.8. Finance Manager

2.8.1. Role Description

Key to the success of the company is a competent finance team. The finance team will take responsibility for organizing the financial and accounting affairs. Some of the key duties of the Finance Manager include:

- ❖ Provide and interpret financial information.
- ❖ Monitor and interpret cash flows and predict future trends.
- ❖ Analyse change and advise Group Leader accordingly.
- ❖ Formulate strategic and long-term business plans.
- ❖ Research and report on factors influencing business performance.
- ❖ Analyse competitors and market trends.
- ❖ Develop financial management mechanisms that minimise financial risk.
- ❖ Conduct reviews and evaluations for cost-reduction opportunities.
- ❖ Manage a company's financial accounting, monitoring and reporting systems.
- ❖ Liaise with auditors to ensure annual monitoring is carried out.
- ❖ Develop external relationships with appropriate contacts e.g. auditors, solicitors, bankers and statutory organizations such as the Inland Revenue.
- ❖ Produce accurate financial reports to specific deadlines.
- ❖ Manage budgets.
- ❖ Arrange new sources of finance for a company's debt facilities.
- ❖ Keep abreast of changes in financial regulations and legislation.

2.8.2 Risk Management

Risk	Possible Solution
Credit risk.	Make detailed contracts with customers.
Liquidity risk.	Make an accurate budget and monitor during project lifecycle.
Poor budgeting.	Monitor at regular intervals and adjust as necessary.

2.8.3 QA Metrics

Metric	How measured
Assets turnover.	Sales/Average total assets. Measure the efficiency of the company's use of its money.

Economic value added.	Net operating profit after taxes – cost. Help determine created value.
Debt to equity ratio.	Debit/Equity. Limit the ratio to less than 1 helps earn profit.
Return on Investment (ROI).	Optimise the efficiency at each stage.
Total Cost of Ownership (TCO).	Consider all costs, including energy, training, maintenance, rent etc.

2.9. Marketing Manager

2.9.1. Role Description

A dynamic marketing plan is crucial in keeping the company brand in demand and staying ahead of the competition. The marketing arm will concern itself with promoting the sales of our product range. Key roles of the Marketing Manager include:

- ❖ Research and evaluate new product opportunities, demand for potential products, and customer needs and insights.
- ❖ Define overall marketing strategy and execution of plans for the existing products.
- ❖ Work with the product development team to manage new product development.
- ❖ Manage launch campaigns for new products.
- ❖ Manage distribution channels for products.
- ❖ Ensure effective, branded marketing communications including the company website, print communication, and advertising.
- ❖ Manage media and marketing staff and external PR agencies.
- ❖ Analyse the effectiveness of all marketing efforts.
- ❖ Define the target market for new products and set the pricing.
- ❖ Explore appropriate business models for sales of new products.

2.9.2. Risk Management

Risk	Possible Solution
Market access.	Gain authorisation from certification bodies as required. Check whether end users are those with purchasing authority, model selling strategy accordingly.
Pricing issues related to core products.	Find trusted sources with appropriate pricing.
Competition.	Ensure quality of products. Produce appropriate advertising campaign. Provide excellent after-sales service.

2.9.3. QA Metrics

Metric	How measured
Brand awareness.	Market survey.

Customer satisfaction.	Customer feedback.
Share of the market.	Market research.

3. DELIVERABLES

Table 1: list of deliverables expected throughout the project

Deliverable	Producer	Recipient	Due
QA Plan	QA manager + managers	All company personnel	On company establishment
Tender Document	Group Leader + managers	Client	Beginning of project
Requirements Statement	Client	Group Leader + Design & Specifications manager + QA & Documentation manager	Beginning of project
Product Specification	Design & Specification manager	Project team	End of specifications phase
Schedule Plan	Group Leader	Project team	End of specifications phase
General Design Specifications (GDS)	Software manager + Hardware manager	Specification manager + QA manager	Design phase
Internal Design Specifications (IDS)	Software manager + Hardware manager	Implementation team	End of design phase
Integration & Testing Plan	Testing & Integration manager	Testing team	Before implementation starts (Testing plan) & Implementation phase (Integration plan)
QA Auditing Reports	QA & Documentation manager	Group Leader	Throughout development process
Codes & Constructed	Implementation team	Software manager	Specified milestones

HW components		+ Hardware manager	
Test Results Reports	Testing & Integration manager	Design & Specification manager + QA & Documentation manager	Integration phase
Acceptance Test Reports	Client	Group Leader	End of project development
QA Metrics	QA & Documentation manager	Project team	End of Design phase
Budget Analysis Reports	Finance manager	Group Leader	Beginning of project
Costs, expenses and resource reports	Finance manager	Group Leader	Throughout the project
Progress reports	Project managers	Group Leader	Throughout the project
Meeting agenda	Group Leader	Project managers	Prior to meeting
Meeting minutes	Meeting chairman	Project managers	After meeting

4. PROJECT MANAGEMENT METHODOLOGY

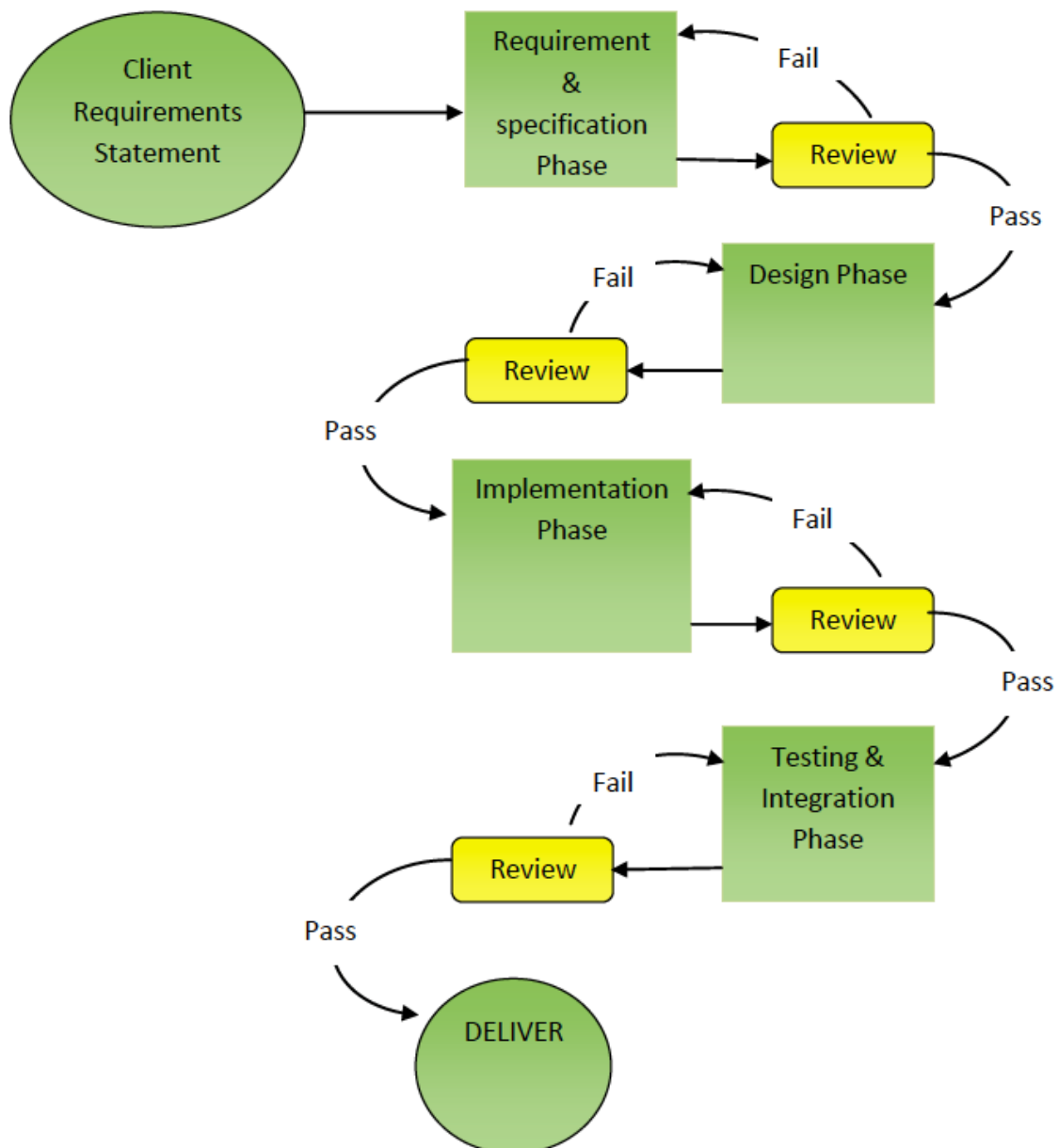


Figure 2: Project Life Cycle

4.1. Requirements and Specifications Phase

Projects will usually start with a statement made by the customer to explain what he/she requires of a product. This is then followed by a thorough requirements capture analysis,

undertaken by the company, to produce a detailed document of the specifications of the project. Any inconsistencies or inaccuracies between the specification and requirements should be identified at this stage and discussed with the customer as this will form the basis to start the design phase of the project. To summarise the basic steps:

- ❖ The customer provides a statement to the company of what the product must do.
- ❖ The Group Leader with the Design & Specifications manager and the QA & Documentation manager will analyze the requirements statement. They will outline the functionality of the product and identify whether all the requirements are specified and appropriate. They will check for any inconsistent or incorrect requirements as well as any unnecessary ones that can be eliminated without affecting the functionality. An estimated budget can also be derived at this stage and the complete product is predicted to be feasible or not.
- ❖ The Group Leader should report to the customer and discuss any changes made to the requirements.
- ❖ Once a contract is agreed, the Design & Specification manager generates a specification document, and the Group Leader, in consultation with their managers and Client will create the project time schedule.
- ❖ The Design & Specification manager should then meet with the managers of each stage of the development process (usually through Review meetings) and provide them with the specification document and start the implementation phase of the project.

The techniques to be used to generate the specifications document are mix of the *state diagram* and the *UML model* (see Appendix B for example).

4.2. Design Phase

As soon as the specifications document is finalised, the Hardware and Software managers, under the supervision of the Design & Specification manager, should start to design the system. They should think of *how* the requirements must be implemented. Since the project might have both hardware and software components, both managers will work together to produce a compatible overall design. They must then generate a General Design Specification document. This should explain the major components of the system and the interfaces between them without diving deeply into the internal implementation of these. The hardware and software managers should then provide Internal Design Specification documents, which are aimed at their respective teams. These should explain in more detail the specifications and the functionalities of the components involved in the system and

sub-systems and how they should be implemented. Specific components used by the company include:

- ❖ Use of Xilinx Spartan 3E FPGA.
- ❖ Use of VHDL to implement the hardware functionality into the board.
- ❖ Use Java for any software application.

Some of the techniques used in the documentation during this phase of a project include: *finite state machine diagrams FSM, UML class and sequence diagrams, design structure diagrams DSD, and pseudo code (see Appendix B for examples)*

4.3. Implementation Phase

Related to the General Design Specification and Internal Design Specification documents, the Software and Hardware managers will analyse the design and consider appropriate time schedules and resource requirements for both software and hardware designs. The Hardware and Software managers will assign tasks to their hardware and software team members respectively. The software team will work on the various programs required using the specified programming language; the hardware team will design appropriate PCBs according to needs taken from the specification. In this implementation phase, both Software and Hardware managers will be responsible for the control of the whole implementation progress as scheduled, and this will feed into regular review meetings. If unforeseen problems occur, the managers may need to reschedule the implementation plan and communicate with team members to fix these problems. When each individual component is finished, the managers are responsible for testing the component and delivering it to the Testing and Integration manager.

- ❖ Software team works on transferring pseudo code to executable code.
- ❖ Hardware team design PCBs and circuit to be implemented in the FPGA according to specification.
- ❖ Hardware manager is responsible for sending the designed PCBs for fabrication.
- ❖ The fabrication factory sends the PCBs to Testing & Integration manager for testing.
- ❖ The Finance manager makes the budget monitoring spend and takes responsibility for the overall cost.

4.4. Testing and Integration Phase

After the product is designed and implemented, the individual components of the project should be tested and integrated during the testing and integration phase. This is a critical phase in the project lifecycle. The testing and integration manager is responsible for the procedures in this phase of the product cycle. For hardware components, each individual device should be tested by executing specified test procedures to ensure basic functionality. For software components, testing and integration must be scheduled both for individual components and as components are integrated together. The recommended strategy is bottom up integration. White box testing strategy is recommended during testing. Final testing of a module/component should not be carried out by the designer of that particular module/component during this phase. Finally the software will be verified in tandem with the hardware. The testing and integration manager has responsibility for producing testing and integration reports which record the details of successful components and any errors encountered (example in appendix B). The errors will likely involve iteration back to an earlier phase of the cycle. Regression test must be adopted at the end of this phase.

- ❖ Throughout the test and integration phase, every member should follow integration and testing schedule and report the errors encountered.
- ❖ For each component in hardware, a test program should be available.
- ❖ The software will be simulated in IDE first, and then integrated by using sideways integration method.
- ❖ Finally the software will be verified with the hardware.

4.5. Quality Auditing Reviews

Generating quality product can't be accomplished without a production process of the same high quality. The procedures used to conduct each of the aforementioned design and implementation phases are designed to meet these standards. A systematic method of evaluating the quality of every deliverable and output, as well as supervising QA metric collection, is one of the main purposes of the review sessions shown in Figure 2. Passing a review session is a *MUST* to carry out the next step in a project. The review teams in the company consist of:

- ❖ Deliverer: The person whose product is under inspection.
- ❖ Review Chairman: Sets the review time, co-ordinates the flow of the review.
- ❖ Secretary: Recording the minutes and outcome of the review.
- ❖ QA representative: Checks the application of the quality assurance procedures.
- ❖ Inspectors: usually 3 individuals. The first one from a prior phase that provided the input for the inspected product, to make sure these are met. The second is from the same team that delivered the product, to provide support and justification of

the actions taken during production. Finally, the third inspector is from the next phase team, making sure that the product actually has what's needed for the upcoming work.

- ❖ On occasions a representative of the customer might be present, if the deliverable is directly related to the final product.

In the review session, the team will use various resources such as: *Testing Report*, *QA Metrics list*, and *Progress Report* (*Appendix A* for templates) to thoroughly inspect a deliverable for any errors, inconsistencies, incompatibility etc. The output of the review session is the *Quality Auditing Report*, written by the QA & Documentation manager to submit to the Group Leader. No further development can take place without the approval of the QA & Documentation manager and Group Leader and to the satisfaction of the review team. If the review team is not happy, the process will be rolled back to the last successful review to find the problem and an appropriate solution.

Appendix A: Documents Templates

The documents in any project are as important as any other part of it. Therefore, they must be standardized to insure better quality. Some of the standards used in the company are:

- ❖ All documents must have cover page, with the title of the document, the date, the author, and the company logo.
- ❖ All pages must have the company logo in the upper right corner.
- ❖ All pages must be numbered in the lower right corner, and the lower left corner must contain the document title.
- ❖ Titles and subtitles must be of bold font size 14.
- ❖ Text must be of font size 12.
- ❖ All documents must include an updated history table, specifying version number, modification date, modifier name, sections and a remarks section.

Configuration Change Management

It is crucial in any project, and even more so in a project involving more than one person, that the most up-to-date documents are used when changes are made and that it is possible to follow the history of a particular document (document here can mean reports, audits, software code, design documents, hardware designs, in fact anything the company produces). To do this a company will put in place a configuration change management system. In its simplest form this will mean that every document will have a *Name* (in the case of this document 'QAM') and one or more *Numbers* after this name representing the version of that particular document. In the case of this document this is '3.2' where '3' indicates the third major release of the document 'QAM' and '2' indicates the second minor revision of release '3'. When changes are made to documents the name will stay the same but the numbers will change. In addition, a short note at the start of the document should summarise what has been changed between the current and previous versions (see the beginning of this document under Document Control).

Meeting Minutes

Meeting :#

Week:

Date:

Course: MSc in Digital Systems Engineering

Group :

Members:

Agenda

1. Apologies for absence:
2. Actions from last meeting:
3. Specific items for this meeting:
4. Next meeting date & time:

[Manager] Reports

Report number:

Circulation list:

Report author:

Date:

1. Description and purpose of the report
2. Body of report
3. Comments/discussion points/recommendations

QA Metric List

Project:

Report Title:

Circulation list:

Author:

Date:

Metric	How measured	Produced by	Date	Results & remarks

Comments/discussion points/recommendations

Implementation Progress Report

Project:

Report Title:

Circulation list:

Author:

Date:

Component or task	Start date	Progress details	% complete	Expected finish date
Insert here the name of the component you are working on	The date you started the work	Details of what you did, any problems encountered, what still needs to be done		

Comments/discussion points/recommendations

Test Report

Project:
Report Title:
Circulation list:

Author:
Date:

Hardware/Software (delete one) test

Test Component:

Test Case:

Test Results:

Error Correction:

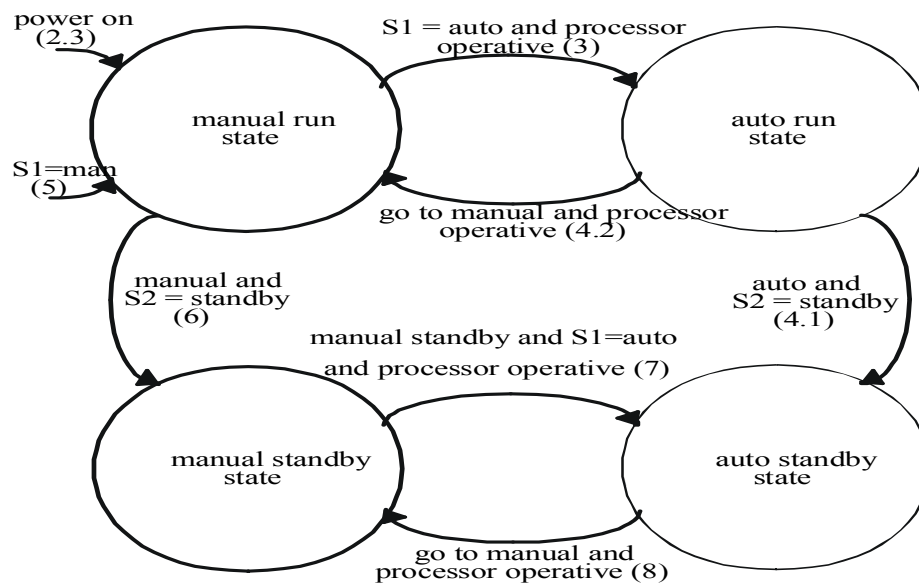
Comments:

Test Programme:

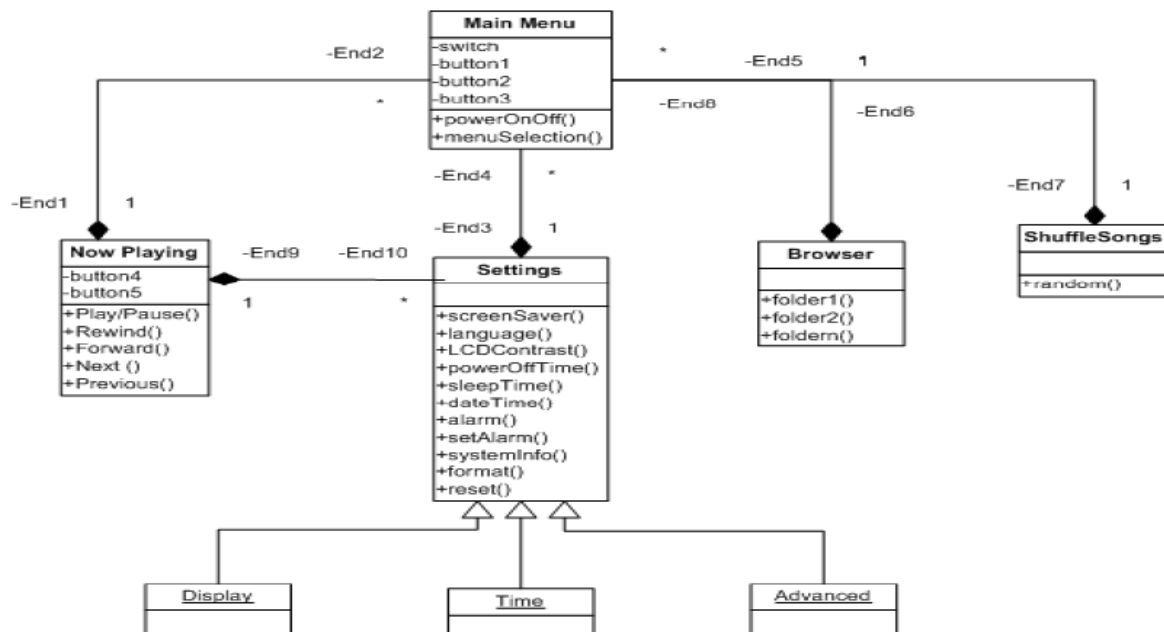
Tester signature:

Appendix B: Design methodologies

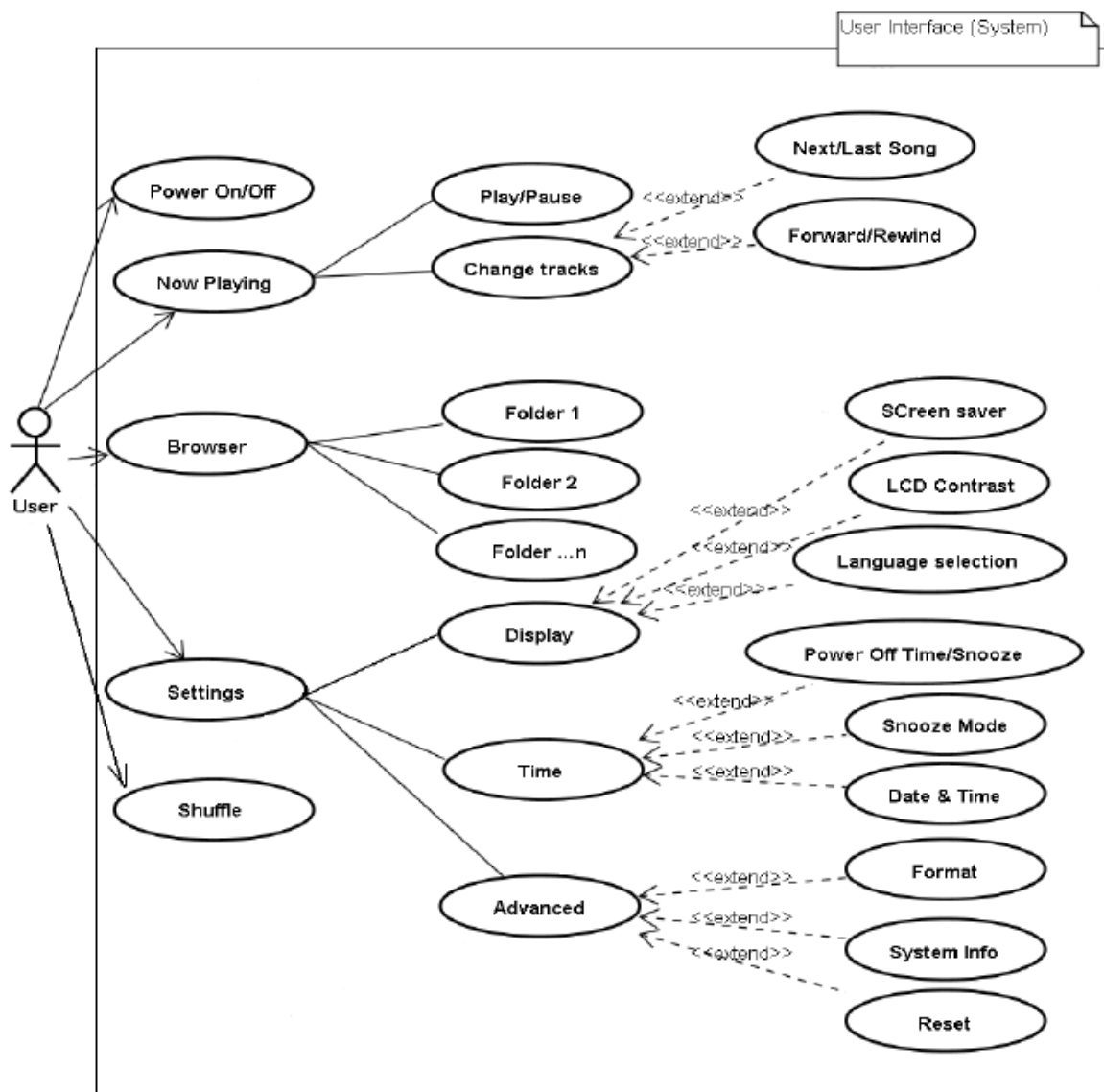
B.1 State Diagrams



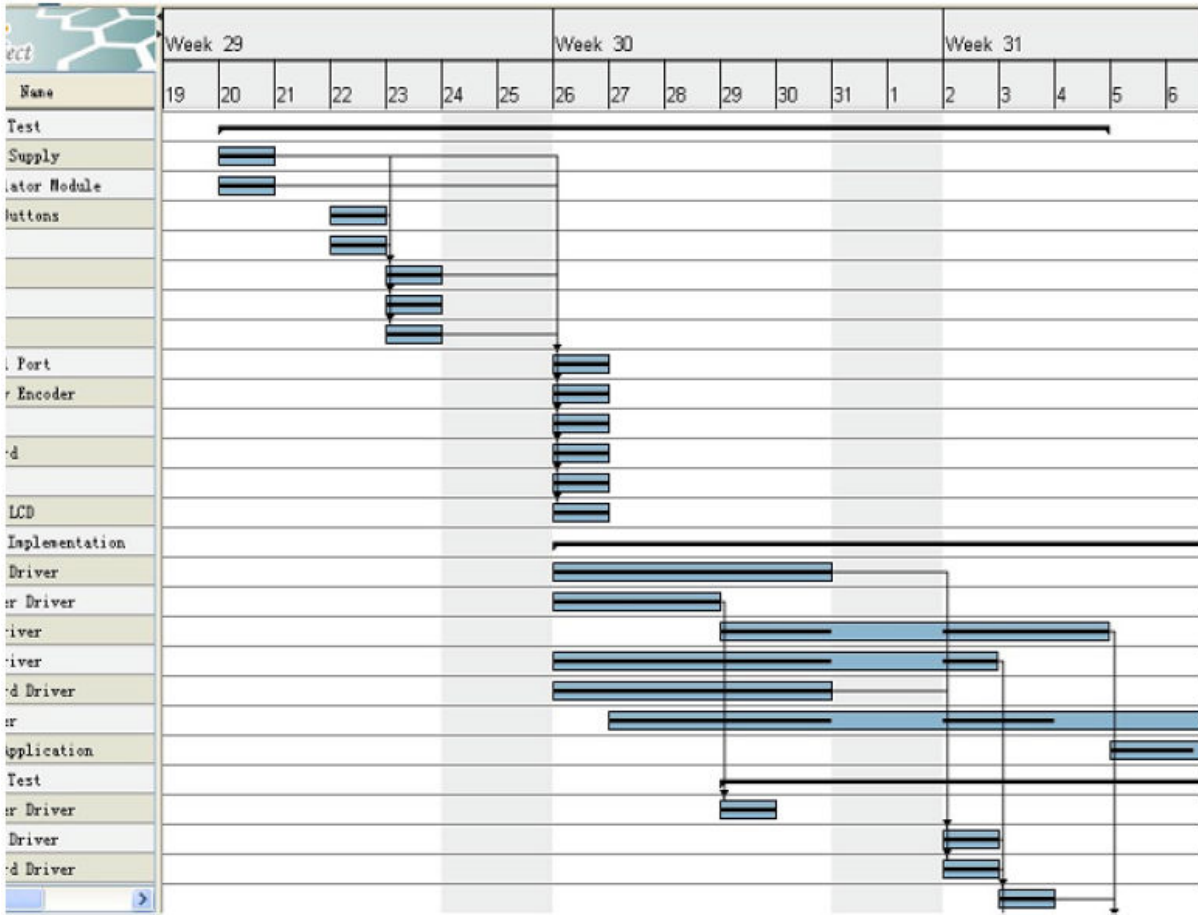
B.2 UML Class Diagrams.



B.3 Use Case Diagram.



B.4 Gantt Chart.



Appendix C: Implementation Examples.

Many electronic system projects might include some software programs written in a conventional or OO language such as java or C, as well as having some hardware components implemented using VHDL and/or assembly language. The important thing when writing code is to keep it as clear and as readable as possible for further modifications and for the integration phase of the project. This is usually done through extensive comments and by keeping version control on the code. Here is an example of the standard format that should be followed (used here in VHDL, but the same system is followed in all other languages).

```
-- Introduction to VHDL
-- Digital Design Assessment
-- By
-- Version 3.2
-- Date: 1/12/2009

-- VHDL libraries for standard logic
library ieee;
use ieee.std_logic_1164.all;

-- Defining a new entity for the single Bit Full Adder
entity Bit_FA is
    port (a, b : in std_logic;
          cin : in STD_LOGIC;
          sum : out std_logic;
          cout : out STD_LOGIC);
end Bit_FA;

-- beginning of the actual deisgn in the architecture
architecture behavioural of Bit_FA is

    begin

-- the description of the FA functionality,
-- using the logical operators that represent different logic gates.
--
end behavioural;
```