From DFT to Systems Test-
A Model Based Cost Optimisation Tool

Michael G. Wahl*, Tony Ambler+, Christoph Maaß*, Mohammed Rahman‡

*Universität Siegen, +University of Texas at Austin, ‡Southwest Texas State University

*m.g.wahl@ieee.org, ambler@ece.utexas.edu, mohammed_rahman@vtel.com

Abstract

Long lasting systems like airplanes have a cost structure where the maintenance costs are larger than the purchasing costs. Testing is required, both for preventive maintenance as well as repair and a majpor source for cost. Previously we have analysed test and Design for Testability for digital systems, covering ASICs, boards and systems. Besides, the continuous development of technology requires cost models that can grow dynamically and, because we will never have all information, can work with incomplete data sets. In this paper we present a tool that is well suited for a wide range of applications. Previously developed cost models can be incorporated and new elements can be added to the model as needed. Due to the generic approach the tool allows modelling general systems. It is not bound to the digital domain, although it has a strong background there.

1 Introduction

Testing is an issue not only for consumer products. Long lasting products have their own rules, starting with the same requirements as consumer products, but then adding a lot of complexity. The environmental requirements are harder, and the testing does not happen once only. Long lasting systems are in a continuous process of maintenance to ensure availability and security. Tests are performed during normal maintenance and in the case of a failure.

If we keep an airplane in mind we all know that they have a lifetime of 20 years and more. The must be maintained as well all equipment required for maintenance. The number of test systems can exceed 50 for a fighter airplane. Obviously test is significant part of the huge maintenance overhead and cost optimisation. DfT is method for optimising test cost, but at this time there is no way of quantifying the benefit.

Our new approach links design, test and maintenance costs so that it becomes possible to get an overall picture. A software tool is presented, which supports a distributed model. Which makes it possible to use the cost modelling approach effectively in the system design flow.

2 Design for testability

The first question is: What is Design for Testability (DFT)? The second question is: Do we really need it? Here we try to answer these questions, at least from a global point of view.

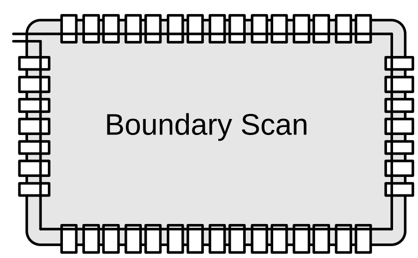

Figure 1 Chip with Boundary Scan

Adding DFT methods adds to the cost of design but increases testability. Adding a scan path requires more complex flipflops and additional overhead for building the scan register.

Subsequently two approaches developed: inside the devices self test became a standard, beginning with BILBOs[6] and later on more sophisticated and efficient methods. To the outside world the Boundary Scan standard gained acceptance (IIEEE 1149.1[7]). This standard significantly improved the testability on the board level.

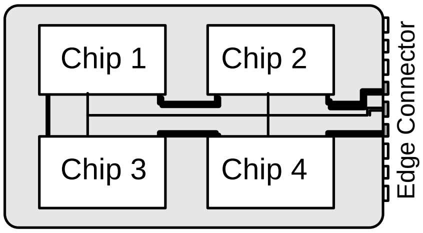

Figure 2 Board with Boundary Scan

Today modules are assembled on chips that have the complexity of yesterdays chips. The test problem remains, simply geometrically scaled down to submicron structures. based on the Boundary scan standard, IEEE P 1500 tackles the test problem on the chip level.

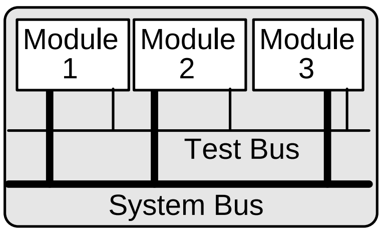

Figure 3 System with test bus

For analogue and mixed signal applications IEEE 1149.4[10] was developed. This standard introduces the analogue test bus, expanding the work on the 1149.1 standard. On the system level yet other testability methods are know, introducing more overhead.

Looking at all these different methodologies for testing and DfT the question arises: Do we really need all this? The answer is: It depends.

Of course, all these methods have not been developed only for the sake of the researchers. The driving forces were high quality requirements, limitations of tester abilities and cost reduction. For today's processors the test costs can make up to 50% of the total manufacturing costs and this is the motivation for looking at testing from the economic viewpoint.

3 Test & Maintenance

Long lasting systems create quite different requirements than consumer products. Reliability and security become the key issue in high safety critical applications. Different rules apply, and concepts developed for consumer products are possibly not the best solution, in particular with respect to the long term cost. We know e.g., that the purchasing costs for helicopters can be less than 10% of the overall costs occurring during the complete lifetime.

The number of subsystems increases and usually each subsystem requires its individual test environment, summing up to around 50 for a fighter airplane. The effective use of testability standards like boundary scan and test busses like 1149.4/1149.5 becomes essential for handling this complexity. In addition, maintenance strategies and the influences from, personnel skills, documentation, and logistic come into play.

At this point it becomes clear that manufacturing test, DFT, and maintenance are closely linked together and sum up to the overall lifetime cost of a system.

Manufacturer	User
Design Manufacturing	Maintenance Disposal

Figure 4 Cost distribution

It would be ideal to know during the design phase of a system how much money will have to be spent for DFT as well as for general and specific test systems to find a cost optimum. For a complete answer we would need all system design information. This is not possible to achieve, so we must be able to survive with an incomplete knowledge. Besides, we have to integrate the information and the process of how to make use of it into the design flow.

4 Cost Models

Cost modelling is the key approach where the cost structure is complex. First, the models themselves must be generated. The origin is always a data analysis which serves as the basis for the developments of analytical models. This is called data mining, because the required data are available, but usually they are buried in distributed data repositories in different formats.

In literature several test cost models were developed. The models described in [1] and many related publications cover the chip, board and system level for digital circuits. A significant part of this work was done as part of the Everest project of the CEC. These models as well as the associated software were closely linked to no longer existing CAD design environment.

In [5] system tests are described. The key issue was the test cost optimisation of minicomputers before and after delivering to the customer site. The result of this work created substantial savings in the system test phase.

Results from a large project that was launched in the USA several years later were published by Maly et al. They discussed chip test cost issues in more depth [13]. Other activities used simple cost modelling tool to justify DfT methods implemented on the IC level. The tools were not suited for public and there was never the intention to use such tools for cost optimisation.

Currently the system test costs in a huge PC manufacturing environment are analysed and the results have been presented in [8] and later publications. We have successfully analysed system test and maintenance costs [11][12] of long lasting products in the recent years.

Summarising we can say that some areas in the test domain are covered by appropriate models (e.g. chip test), while other areas are no longer up to date (e.g. ASIC) or not even approached. Our next step will be an update of the digital system model to incorporate the requirements coming up from the intensive use of IPs and the test problems associated with it. Keeping the huge area of testing in mind it becomes obvious that each model for itself will never be usable in a system environment, not to mention long lasting systems that add the time dimension to the model.

Figure 5 Section of maintenance model

As mentioned before, software was produced for some models. Simple approaches are programmed as spreadsheets, but spreadsheets do not allow dynamic changes. Besides, spreadsheets create huge security problems, because all cost critical information is contained in a file that often is transferred via the Internet.

Some of the software is now unusable due to a close link to a no longer existing design system, while other models have no implementation and exist only in paper form. This situation implied the development of a powerful general modelling tool so that the existing models as well as the new developments can easily be linked together.

5 Cost modelling techniques

The scenario described above makes it clear that we cannot simply link the models to get the overall costs. We need a unified modelling approach as well as a tool suited for a complex model in a distributed environment.

Modelling can be done on quite different levels of abstraction. Figure 5 shows a top level view of our maintenance cost model. This kind of model is appropriate for discussions but not for a tool. The modelling technique we employed is based an entity relationship diagrams and the Express-G representation. We decided to create a standard cost bearing entity according to Figure 6. The top part shows some general information, like the cell type and reliability information an of course a link to one or more cost bearing entities. This is the mechanism we need for creating the cost tree.

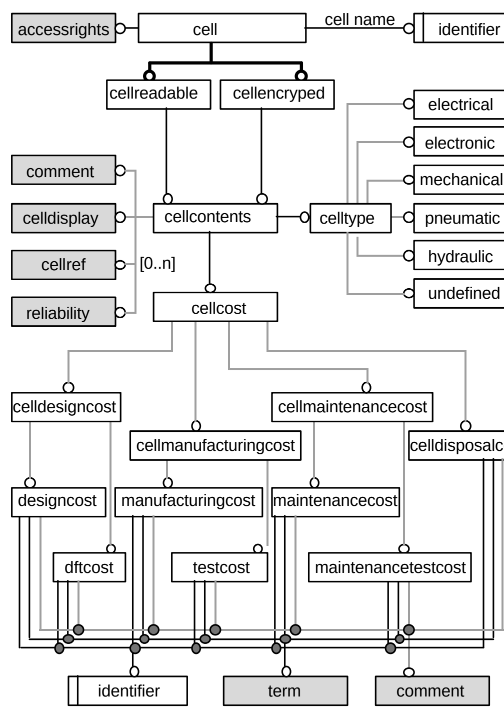

Figure 6 Diagram of the cost bearing cell

The cost themselves are categorized according their occurrence during the lifetime: design, manufacturing, maintenance and disposal. Test costs are always kept in a different category than the other costs. This quite general model provides an ideal framework. It is a compromise between the aim of modelling everything up to the last detail and the time and information available. The example in Figure 7 is derived from [13] and shows a simple part of the model tree. The white boxes are the roots for more detailed sub trees. Each of the entities in the example would be represented by a generic cost bearing entity as described above.

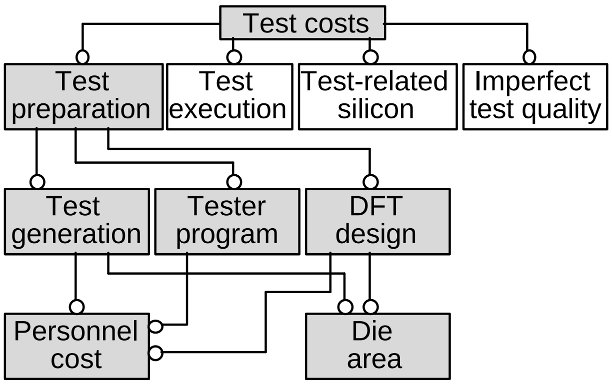

Figure 7 Example derived from [13]

On the lowest level, cost equations are described which allow the evaluation of the costs. The equation below (from [2]) shows the test pattern generation cost TG depending on the number of gates (*bgate*) in block *n*.

$$TG_n = kc_n \bullet bgate_n^{PC_n}$$

Of course, one of the key issues in modelling is the creation of a data dictionary that describes all cost bearing entities as well as the parameters used in the model. Without the data dictionary a model is unusable. This is, as we all know from standardisation, a cumbersome task, but essential.

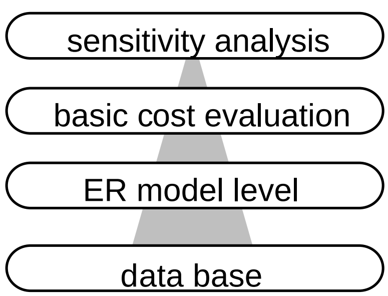

Figure 8 Data and cost evaluation

Now we can have a complex model at hand and we get to the evaluation part. First, we want to have the overall cost. This is a straight forward procedure: You simply have to sum up the cost items and that's it. The model of the cost bearing entity above supports different alternatives, too. In the first analysis step the model can be searched for alternatives and the optimal alternative can be shown. In the third step, those parameters that can be modified are evaluated with respect to their influence on the cost (sensitivity analysis).

6 The modelling tool

The modelling tool is a state of the art tool, based on the Internet as the underlying architecture. The tool is implemented in a three tier architecture: The user (client) communicates with system using a web standard web browser supporting Java. The web sever will upload an applet to his machine which is the user interface and which performs the necessary editing functions.

The data resides on the computing server in an SQL database. The computing server is not visible on the Internet, only accessible by the web server. It performs the evaluation of the model and prepares the results for the user. The results are sent to the user via the web server. This architecture ensures that the data is well protected.

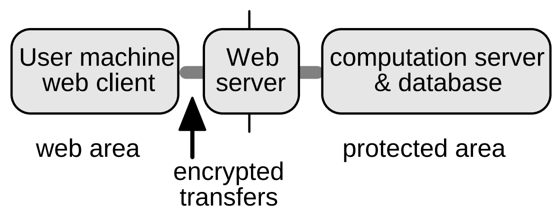

Figure 9 Three tier architecture

The connection itself must be encrypted, because cost information is highly sensitive data and must not be available to the public. Security methods are necessary for the model itself. There different access levels have been defined: the owner can modify the model. Others have either the privilege of reading, which means that they can browse through the model, but cannot make changes, or they can evaluate the model without being able to see the structure. Only the evaluation software on the computation server accesses the models.

The key point of the system is the evaluation. In previous implementations a Monte Carlo simulation was performed to get the sensitivity data for the parameters. We will follow this path first, but we now analysing neural networks for solving this problem. In large data mining projects this approach has been proven very successful. Another approach we will analyse is the use of genetic algorithms for finding the cost optimal parameter configuration. Further research is necessary to find an optimal solution for this np-hard problem.

7 Tool application

Finally, we have to deal with the question of who is the user of cost modelling approach and the cost modelling tool. Obviously the growing competition forces the industry to a continuous improvement process for product quality and productivity. There are areas where productivity is handled very effectively, e.g. the automotive industry. Here the basic product does not change very much within a year, whereas a year in the digital domain is a real long time span.

The tool users will be found in different parts of a company. The developers e.g. can use it to justify their decisions concerning DfT methods. The engineering management level can have a look at the individually planned test support and optimise from a more global view. Finally, other departments can incorporate the model and use it in their plans for a complete system.

The cost modelling tool currently under development will, together with the appropriate is targeted at the engineering management level. One definition of Engineering Management has been written down in a military standard [8]. It is still a valid statement: *"3.1 Engineering Management – The management of the engineering and technical effort required to transform a military requirement into an operational system. It includes the system engineering required to define the system performance parameters and preferred system configuration to satisfy the requirement, the planning and control of technical program tasks, integration of the engineering specialities, and the management of a totally integrated effort of design engineering, speciality engineering, test engineering, logistic engineering, and production engineering to meet cost, technical performance and schedule objectives."*

The cost modelling approach linked with the tool will provide a well suited tool for fulfilling the requirements described in this ancient standard. Recent feedback from industry showed that in many cases the application of cost models is not a standard procedure. One of the reasons for this is the lack of a tool that enables the user to describe and use cost models effectively.

8 Summary

This paper describes a consistent cost modelling approach. It is the first time that conceptionally the range from digital design to complex heterogenous systems has been addressed. We have shown the steps that are necessary for effective cost modelling, such as data dictionary, er-model, optimisation algorithms and sensitivity analysis. We have also shown that for many domains models are at hand, but these models are hardly used in industry. The tool under development now provides the means for effective use of the models and allows the integration into the engineering/design flow. We are co-operating with industry concerning in the tool specification and we are running a pilot project where we create a model and directly use the tool in a read industrial environment.

Our future work will be to homogenize the models and extend them to current problems, e.g. IP based designs. Besides, we need more effective algorithms for fast online evaluation of the model.

9 References

[1] Dear, I; Dislis, C; Dick, J.; Ambler, A.: Test Strategy Planning Using Economic Analysis. In: Economics of Electronic Design, Manufacture and Test, M. & A. Ambler (Eds.); Kluwer Academic Publishers, 1994

[2] Dislis, C.: A financially based automated cost advisor for Design for Test strategy generation. PhD Thesis, Brunel University, 1992

[3] “EIA 632 Processes for Engineering a System”. Electronic Industry Alliance, 1999

[4] “EIA 731.1/2 System Engineering Capability Model”. Electronic Industry Alliance, 1999

[5] Farren, Des; Ambler, A.: System Test Cost Modelling based on Event Rate Analysis. Proceedings of the 3rd Intl. Conference on Economics of Design, Test & Manufacturing, Austin, 1994, IEEE CS Press 1996.

[6] Koenemann, B.; Mucha, J.; Zwiehoff, G.: Built-in Logic Block Observation Techniques. IEEE Test Conference, Cherry Hill, 1979, pp. 37-41

[7] Maunder, C.; Tulloss, R.: The Boundary Test Access Port and Boundary Scan Architecture. IEEE Computer SC Press, Los Alamitos, CA , 1990

[8] "MIL-STD-499A Engineering Management"; Department of Defense; 1974

[9] Orton, R.; Ambler, A.: A System Test Cost Model as Influenced by Business Plan. 2nd IEEE International Workshop on System Test and Diagnosis, Atlantic City, 1999

[10] Osseiran, A.: Analogue and Mixed-Signal Boundary-Scan- A Guide to the IEEE 1149.4 Test Standard. Kluwer Academic Press, Dordrecht, 1999

[11] Riedel, T.; Wahl, M.; Ambler, A.; "Cost Analysis linking Deign, Test & Maintenance". European Test Workshop, Constance, 1999

[12] Riedel, T.; Ambler, A.; Wahl, M.; "LCCA: Life Cycle Cost Analysis". AUTOTESTCON, Salt Lake City, 1998

[13] Wei, S.; Nag, P.K.; Blanton, R.D.; Gattiker, A.; Maly, W.: To DFT or not to DFT? IEEE ITC 1997, Washington, 1997, pp. 557 - 566