Implementing a Digital Twin, Design and Test, Test and Measurement Strategy

By Duane Lowenstein and Chris Mueth at Keysight Technologies

Explore the history of digital twins and discover how digital twins can and will change the way we develop and implement test strategies in the future. Read the award-winning IEEE article, written by Keysight's Duane Lowenstein and Chris Mueth.

This article was first presented at IEEE AUTOTESTCON 2022, August 29 to September 1.
It is reprinted with permission.

Implementing a Digital Twin, Design and Test, Test and Measurement Strategy

Duane Lowenstein
Keysight Technologies
Solutions Fellow
Andover, MA
Duane_lowenstein@Keysight.com

Chris Mueth
Keysight Technologies
Digital Twin Program Manager
Calabasas, CA
Chris_Mueth@Keysight.com

Abstract—

Accuracy, repeatability, and utilization are used continually when we talk about a test and measurement strategy. These fundamentals allow the balance between the technical and business imperatives that test contributes to a product or program's life cycle. From a design point of view, test is the de facto tool to ensure the theory of design meets the reality of the product or production specifications. In manufacturing, test is a balance between ensuring quality and cost. For support, test is about insight with simplicity of operation. All of these play an integral part of the success of a business, but to make this all happen, there is a fundamental assumption that the test and measurement strategy is implemented and operated as it was designed.

The fact is, not all tests are created equal when looking across an enterprise and/or workflow. Not everyone develops a test strategy and simulates its effectiveness and efficiency on the product the same way. The idea of a digital twin strategy has been around for years in the mechanical world and is starting to gain traction in the electrical world to minimize the gap between theory and reality. These same principals now can be applied to the test and measurement world. Such a strategy can lead to greater accuracy, repeatability, and utilization of test strategy. It will also allow test or design changes to be made before designs are frozen for technical and/or performance reasons. This Design and Test (DaT) process would not only change the way design and test flows work, but how overall programs change the way they do business from concept through support.

This paper will explore the history of digital twins and show how digital twins can and will change the way we develop and implement test strategies in the future. It will detail how the workflow throughout a product/program's life cycle will change to reduce time, resources, and cost while dramatically increasing predictability and repeatability, and ensuring consistency of test strategies. This paper ultimately will give a foundation for a blueprint to develop a test and measurement DaT/digital twin strategy, share examples of use cases today, and outline the business and technical benefits for implementing such a strategy.

I. Introduction

For decades, terms like faster, cheaper, more accurate, lower power, smaller, and so on have been thrown around when developing new electronic products. For many years, we did this through step function innovations like moving from through-hole components to surface mount, manual circuit analysis on spreadsheets to sophisticated Monte Carlo simulations, and manual testing with probes to autonomous custom fixtured test stations. Even with all this progress, on average, most companies are still taking several months to correlate design and test data, according to data collected in a 2018 [1] survey by Dimensional Research (Figure 1). This type of effort, although improvements over the past, still adds significant delay to product launches and strain on engineering resources.

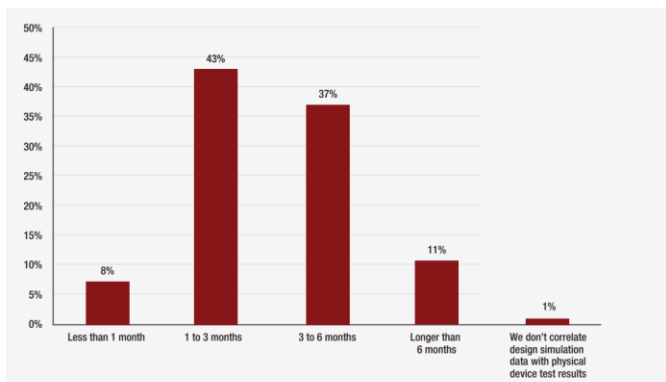

Figure 1. Approximately how long does it take to correlate design simulation test data with actual physical device test results?

So, what is the next revolution? It seems to be the idea of having a digital twin methodology for design and build workflows. Digital twins allow you to simulate a product in such detail, it mimics every aspect of the product as if it was real. In other words, it eliminates the gap between the theory of design and reality.

Although there are several people who have been part of starting the digital twin movement, Michael Grieves is credited most often with building and applying the first model in 2002

and presenting it at a Society of Manufacturing Engineers conference in Troy, Michigan. Since then, the breadth and depth of possibilities and solutions have exploded. By its general definition, a digital twin is a real-time virtual digital representation that mimics a physical object or process. Industries such as the aerospace and defense industry have been developing strategies to implement such solutions for years.

In its simplest form, the digital twin concept seems almost obvious — but as it is unpacked and understood, it becomes deceptively complex. The ability to model a single function over time as it is being stimulated, although it requires deep thought and meticulous construction, is not impossible. How about the possibility of tens of thousands of interdependent functions happening over time with power and frequency changes over various environmental conditions, all happening independently and possibly randomly? That's the definition of complexity.

Without eating the proverbial digital twin elephant in one bite, there is an obvious starting point at the bottom of the product life cycle "V", Figure 2. This is the point that design, and test are closest and have both the greatest ROI and easiest data flow for minimizing the gap between theory and reality, allowing for a realistic digital twin representation.

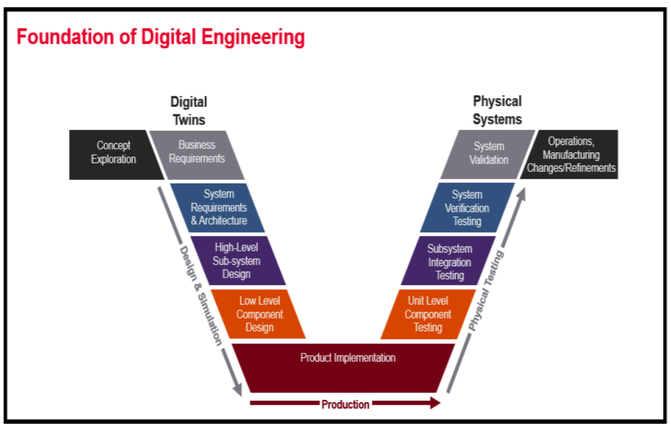

Figure 2. The Digital Engineering "V" Model

II. THE NEED FOR A NEW PRODUCT LIFECYCLE PROCESS

Figure 2 is a high-level view of the interaction between design, manufacturing, and test. Conceptually, the link between design theory and manufacturing reality is test. Test provides feedback to R&D to hone its models and simulations to mimic real life physics so that when built, the product will work as designed. For manufacturing, test is the gate that allows good product to pass on to the next process or end customer and bad product to fail and either be reworked or scrapped. In either case, any and all of this information becomes useful data and the only place where the theory of design is validated against the reality of manufacturing.

Although the data flow seems to be continuous, in virtually all design and build process flows today, only two of the three elements exchange data at best. It is also possible that data flows in only one direction. This leaves huge gaps in data and information for passing any feedback from manufacturing to design and eliminates any possibility for a Digital Twin modeling process. Therefore, the process today as illustrated by Figure 3 needs to be reevaluated and evolved into a new model with its associated need for greater data flow.

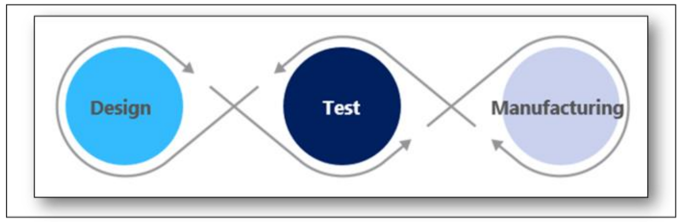

Figure 3. The Relationship Between Design, Test, and Manufacturing

Although many companies have standardized and streamlined their design, test, and build processes, including adding gate processes to ensure repeatability and consistency, there is still complexity in the tools and methods. Figures 4 and 5 again use date from the 2018-Dimensional Research [1] study that shows the multitude of software tools that organizations are using to design, test, and verify their products. The data shows most organizations use three to ten different software tools for each process. This complexity alone can be a symptom, if not the root cause, for the longer product introduction times, product performance issues, and poor quality, both during production and possibly customer support.

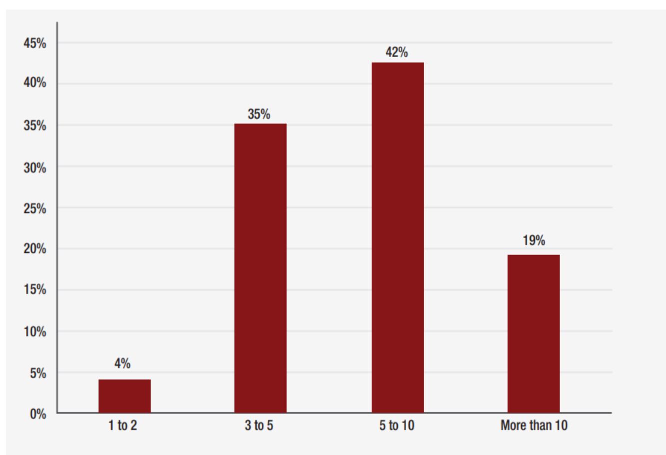

Figure 4. Survey Response: Approximately how many different software tools do you use for design?

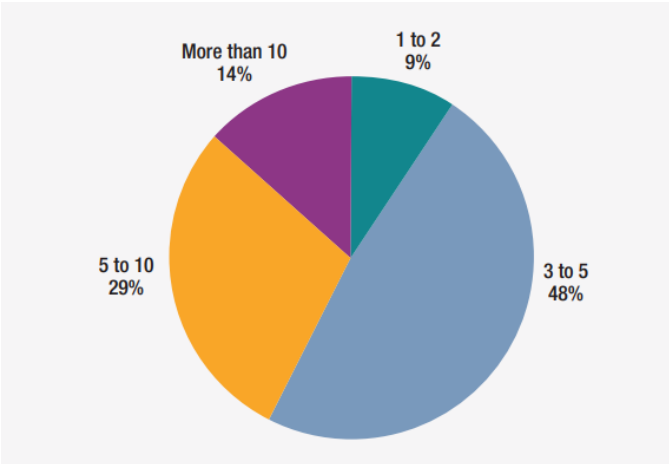

Figure 5. Survey Response: Approximately how many different software tools do you use for testing and verification?

If we could start from a blank piece of paper and design this new process from scratch, of course, we would design a process that relies on minimum software interfaces with free-flowing data that has checks and balances that brings the needed attributes of a digital twin model together. But, of course, few organizations if any have the luxury of stopping everything, redesigning the current processes, and implementing new tools and processes while continuing to move the endless stream of new programs forward. Of course, for any redesign, we will also need to break the old "throw it over the wall" mentality, from design to manufacturing, and reinforce a new culture of continuous flow simultaneously with the other changes.

It will take an evolutionary change rather than a revolutionary change in the current process, but it will require a revolutionary change in the mindset of management to drive and reinforce that the product lifecycle is not a one-of-a-kind process for every new product and program, but rather a repeatable and controlled one. The current "V" model, as seen in Figure 2, best describes the process going from design and simulation to physical production and then physical testing/verification. Although the process looks logical and the flow reasonable, currently, it is still a one-way process that lacks shared data and ownership.

III. Design and Test (DaT) Digital Process Threads

Connecting design and test is the nirvana of what many companies are trying to achieve in order to deal with the complexity of the products being designed today. A modification to the classic "V" model is shown in Figure 6. It utilizes the concept of digital process threads to link the design digital twin with the physical system test. This effectively closes the "V," which has several important advantages. First, IP and data created during the design cycle can be reused in the test phase, which provides improved consistency and correlation between simulated and lab test results. Second, data collected in the test phase can be used to update the digital twin model. Examples of the digital process threads are shown in the gray arrows of Figure 6.

To address the needs of model-based system engineering (MBSE), the digital threads need to link the design and test process from specification requirements through the simulation and test processes to the final data produced by these processes.

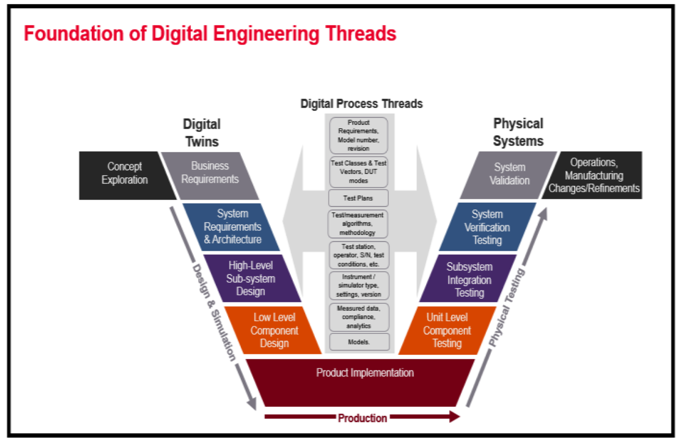

Figure 6. Modified "V" Model

If carefully architected, the solution implementation for such digital threads provides several key advantages. The first advantage is the consistency element mentioned previously. If the processes are digitized, repeatability improves. Inclusion of sources of truth also contribute to consistency. Digitized requirements are a source of truth, as is reference IP (signals, test patterns, digitized methodologies). Second, these threads produce traceability through the process. For any given piece of simulation or test data, there is a record of how it was produced, under what conditions it was produced, and what process, tools, and instruments were utilized, all captured with context in the form of a Digital Twin. The third important advantage is enablement of process automation. This is enabled for a particular task by utilizing information captured in previous parts of the process along with reference IP to automate a particular process. For example, it has been proven that the test automation coding process can be automated to a significant extent, as well as the results compliance process.

The design verification process (simulation) and hardware validation process comprised almost two-thirds of all product development time and is an area the industry is looking to improve. On close examination, these two activities are similar in nature. Both activities attempt to characterize a design to a set of requirements, with one activity being in the virtual digital twin domain, and the other in the physical test domain. This realization provides opportunities to connect these tasks through digital threads to enable the advantages described.

IV. Process and Data Management

An implementation solution for verifying complex products which includes digital threads is based on a process and data

management (PDM) layer for design and test. This concept has been used for years in complex CAE design where mechanical, thermal, fluid dynamics, and multi-physics need to be combined to verify a design, such as for a turbocharger design. In the CAE space, this is called simulation process and data management (SPDM). For electronics design and test, we extend the concept to include test.

The role of the process and data management layer is to define, manage, and execute a simulation or test process and manage the big data produced by these processes. The process definition is really a reusable digitized recipe which can be sequenced to orchestrate the simulation/test automation and is linked via digital threads. Data produced from the process is linked and tagged with all relevant process information and sources of truth. This allows the data to be efficiently processed and enables analytic capabilities. For complex systems, the (PDM) layer can be the interface layer between MBSE and simulation/test tools. The input to the PDM layer is requirements and analysis requests. The output is conformance to the requirements and analytics. A simplified view of the PDM layer is illustrated in Figure 7.

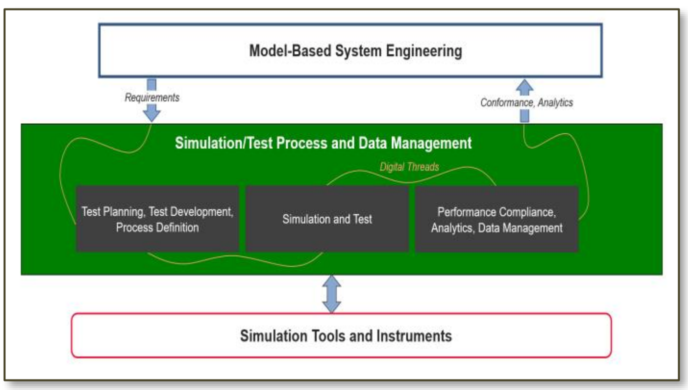

Figure 7. Function of PDM Layer

Verifying and validating a complex electronic product today can require verifying millions of items. The industry trend is to "shift left" to do more of this in the design phase, where design tradeoffs can be managed and the cost to find an issue is much lower.

Gaining confidence in the design phase is important, as design iterations can be very expensive in terms of cost and time. Because designs are becoming more complex, there is a need to do more simulation verification, but that lengthens the design cycle, which is not necessarily expanding for most companies. This verification process can be a significant bottleneck. Therefore, a balance needs to be struck on what is practical to simulate. Techniques such as corner analysis can provide insights into the robustness of a design. High performance computing (HPC) can shrink the time it takes to do a corner analysis, as well as the overall verification time, by parallelizing computing tasks. This can be significant – up to an order of magnitude in improvement. HPC combined with a well architected PDM layer can significantly reduce this bottleneck.

Considerable time in the design simulation verification step and the hardware test validation step is spent deciding what to test, how to test, and developing tests. These tasks can be sped up through the use of reference IP and digital threads, as mentioned. While on the surface this might look like an additional step, the effort to implement is reduced through proper selection of tools that help bridge the two.

How does this work within a PDM layer? It starts with a set of requirements and vectors which are usually defined in a digital document or SysML model that are provided as part of the requirements flow down which are digitally mapped into the PDM process (usually a mapping file). The requirement is a unique specification to be tested or simulated (for example, gain of an amplifier). The vector describes the test class and parameters by which the item might be tested (for example, the frequency range and power level to be tested at). The test class specifies a particular class of instrument or simulator to be used. When this information is utilized with a reference IP library which supports that particular test class, the recipe for conducting the simulation or lab test can be automatically created. The recipe can be further refined and customized as needed, but the root method for automating through this troublesome step is now consistent and functional. This process could potentially shave weeks off test development time.

The resulting recipe and associated simulator/instrument drivers, which are an output of the test development step, are utilized directly to automate the simulations or test instruments. This generically is called the data acquisition step as the virtual or physical DUT is being exercised to produce data for analysis. As the data is produced, a metadata tagging process is used to incorporate the digital threads. Because data from different tool and instrument vendors utilize different formats, data needs to be adapted to a common format like HDF5 using a hub and spoke technique. This technique reduces the overall task of creating translators from one tool to another. Instead, an adapter is utilized to connect a particular data format to the hub.

PDM serves an important role in data management. Collecting, organizing, and analyzing data which covers all requirements, operating modes, voltages, temperatures, and the DUT sample lot in the physical test domain is a daunting task. There can be literally millions of data points to examine. However, using PDM and digital threads, this process can be automated. It is as easy as loading datasets into a compliance tool that is part of the PDM layer and having the tool do the work of creating a compliance report summary. Because the data is in a common format, design and test data can be examined together, providing insights which can be used for correlation of test results to expected results, design centering, and troubleshooting. The use of metadata allows sorting of data by any metadata type. The metadata can also be used to enable machine learning if the sample lot is large enough.

V. Streamlined Verification Workflow

Now that the importance of the PDM layer is understood, how does this work in a workflow? Figure 8 illustrates the workflow. The process starts with the digital document and the design and test engineers working together to decide what requirements need to be tested. The requirements/vectors, along with the test libraries, automate the test development step through the use of a software plugin designed for this task. The output is a test plan and IP that can drive the simulator or test instruments directly as part of the data acquisition step. The sequencing of the recipe can be done in a number of ways, but when the data is produced, it is tagged through the use of a results listener then stored in a repository. At this point, the data can be accessed by a number of people with different roles through a compliance analytics software plugin. The compliance analytics tool automated the compliance process and provides important analytics capabilities.

The overall workflow provides a collaborative environment where people from different disciplines can work together. The process helps parallelize the design and test development efforts, which for many companies is still largely serial. It is not uncommon for a test engineer to comment about the design being “thrown over the wall” where there is little IP to leverage and little understanding of what the expected results should look like.

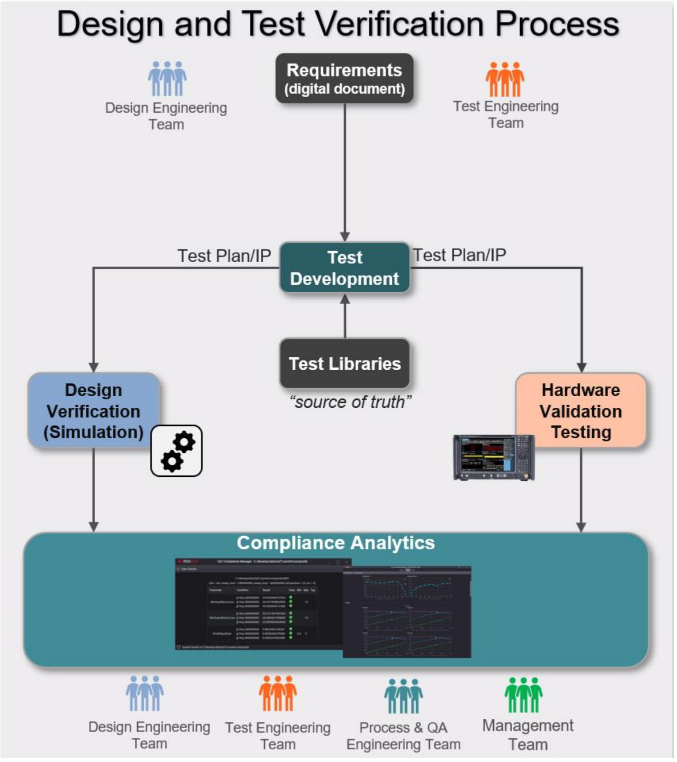

Figure 8. Design and Test Verification Workflow

VI. Other Requirements

Implementing a PDM-enabled workflow is not easy. The EDA and test and measurement industry has not done enough to enable customers to adopt PDM, and customers in the past have had to create custom in-house software solutions themselves, with varying success and little support from the EDA and T&M industry.

One of the key requirements is a vendor agnostic approach. There is a natural reluctance for vendors to work together and this compromises interoperability. However, if the PDM is architected properly, the adaptation work needed for a vendor’s product to interface with the PDM is reasonable.

While most customers face common challenges, each customer has their own unique combination of processes and systems. Therefore, the PDM utilized must have open APIs for the software plug-ins and utilize a framework that is extensible. Programming languages C# and C++ are commonly utilized, but Python is growing in popularity, as it is ideal for application support and is commonly taught in many schools. The use of open APIs and Python are a great combination for framework extensibility.

Cloud enablement is another important consideration. We already discussed HPC for design verification, and parallel computing in the cloud is an attractive way of scaling for these peak simulation needs. Besides the large cloud compute providers, there are one-stop-shop cloud hosting providers that can provide a more turnkey cloud compute experience. But cloud HPC is not only limited to simulation. Large measurement disaggregation techniques can be used to process large data acquisitions. Parallel processing not only speeds up the computational speed but pushing the data into the cloud for this task can free up valuable test CPU time to increase overall throughput.

Tying together design and test into a common PDM process also presents some unique challenges. The test IP and measurement science utilized needs to be as consistent as possible between the design and test domains. Software architectures designed to service one domain may not work with the other domain, and IP developed in the design test domains may be constructed differently using different software languages. To deal with these issues, the PDM architecture and significant CS skills required for this effort must be carefully considered. Both design and test domain knowledge as well as application knowledge also need to be factored in. A vendor with both design and test expertise supplying a PDM to the industry will have an advantage.

VII. Is it Practical?

There are a number of challenges that have been outlined for an automated verification workflow, so the question is, is it

practical? The short answer is yes. Within the industry, customers have implemented (albeit with much effort) their own solutions which cover portions of the workflow. There is a realization that digital transformation and PDM is a must if companies are going to be successful. Recently, the full workflow outlined here has been demonstrated and is gaining interest from companies struggling with traditional processes. As always, one of the fundamental questions becomes where to start. One strategy is to begin with the biggest return on investment.

DaT has gotten a lot of attention and traction. Some of the tools and understanding that are needed are in place. With the additional data transformation and simulation tools that will be available, the methods and process changes will be the key to success. Although this may be harder for engineers who have been using the traditional tools, newer engineers will naturally be attracted to a DaT flow and the future PDM workflow.

VIII. Conclusion

This paper has outlined the challenges facing product development teams today in designing complex products. The lines between a system and a component are being blurred and today's products are so complex that traditional ways of doing things no longer work. The concept of digital transformation and the "V" model help customers deal with this complexity.

To service the "V" model, the concept of a process and data management layer was introduced. The functionality of the PDM was examined as well as key requirements. Applying the PDM to an automated verification workflow concept was outlined. This workflow has been demonstrated within the industry.

Ultimately, we need to move from a linear process for developing products that does not reuse, feedback, or share data (as shown in Figure 9) to one that data is useable, shared, consistent and can be used efficiently and effectively by all, from cradle to grave of the products on an ongoing basis. Figure 10 shows the first step in the journey. Although still linear in flow, the idea of sharing data freely becomes part of the process. The goal, as shown in Figure 11, is the ability to have data and knowledge not only used and exchanged for one process, but all components, subsystems, and systems within and across products.

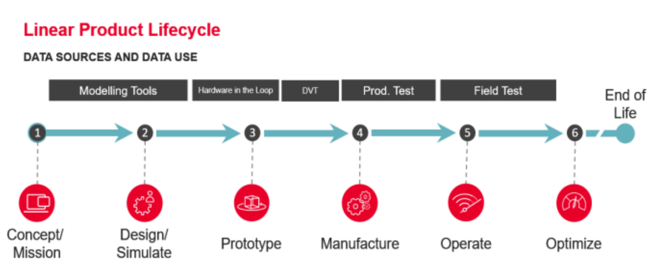

Figure 9. Traditional Linear Product Lifecycle

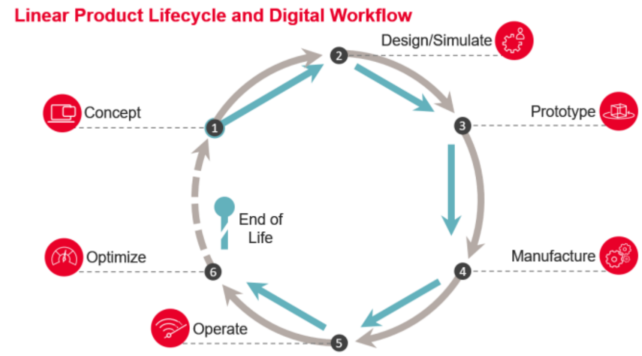

Figure 10. Linear Digital Product Lifecycle

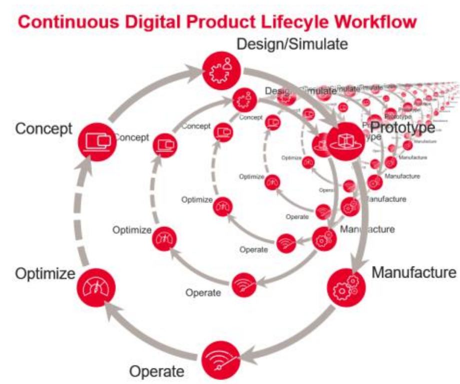

Figure 11. A Continuous Digital Product Lifecyle Workflow

There is never a good time for change, even when it is too late. The optimal time is now, with something that is manageable and leads to greater change in the future. Digital twin models will be a competitive advantage for those who can implement them in the short term, before they become necessary for everyone in the near future. The first step in this process is a PDM strategy with the first step of implementation being a DaT process.

References

[1] Removing Time-To-Market Barriers for Design and Test Engineers: Keysight Technologies, pub 5992-3814EN