

The Role of ATE in the Semiconductor Ecosystem

by John Morris and Roy Chorev, Teradyne Semiconductor Test Division

The semiconductor ecosystem spans the design, test, and manufacturing environments. As a part of this ecosystem, ATE can do much more than test and bin devices. It can be an integral part of an overall process to identify design defects, improve yield, and reduce defect escapes. A well-designed system can continuously reduce manufacturing test costs and reduce the time-to-revenue for a new product.

At a high level, the ecosystem the ATE platform exists within is attempting to supply three distinct functional domains: design, test, and manufacturing (Figure 1). Each of these can have different needs and priorities:

- The design domain must quickly assess the integrity of new devices when initial silicon becomes available and determine any potential weaknesses of the device design across its intended operating conditions.

- The test domain initially is interested in quickly developing quality screening and characterization test programs and interface hardware to be in place for initial silicon. Once silicon arrives, the test domain focuses on using those programs to provide device functionality feedback to the design domain and screen early samples for target customers. Longer term, the test domain must deliver highly efficient, robust test solutions to be used in volume manufacturing.

- The manufacturing organization continuously works to minimize a device's overall cost while maintaining high quality. It requires a stable process which can be easily replicated and maintained as needed at other manufacturing facilities inside or outside the company. The manufacturing domain also works to maximize the overall utilization of equipment, including ATE systems.

Because of the different needs and priorities of each group, it is typical that each domain requires specialized tools to produce, process, and analyze data. However, much of the ecosystem data moving from domain to domain ultimately is consumed or generated by the ATE system, making it a critical component of the overall ecosystem.

The efficient transfer of data between design and test domains can allow design data to be easily utilized to generate initial test programs. The data from early ATE test and characterization results can help identify underlying design or processing issues that impact the initial device bring-up or limit the longer-term device yield during volume manufacturing.

Data transfer from the tester to the manufacturing domain can lower the cost of test, improve quality, and increase the utilization of the equipment used in the process. Automated analysis tools within the ecosystem—interpreting a rich set of data and events coming from multiple monitors, screeners, and databases in real time—can modify the ATE program to opti-

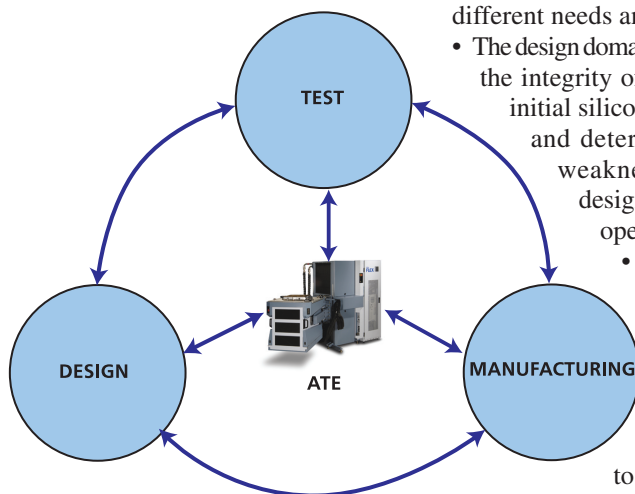


Figure 1. The Test Cell Ecosystem

mize the test flow without sacrificing quality. These same data monitoring tools also can detect deviations in the expected results of the manufacturing process allowing underlying causes to be quickly identified and addressed, improving the overall system and test floor utilization.

Historically, semiconductor manufacturers have strived to build such test-cell ecosystems by growing them organically, trying to cobble together multiple software products and in-house tools to address specific end-use requirements. The various tools used within the process often utilize custom data formats, which lead to system implementations requiring heavy use of real-time or post-processing data converters. They often must have storage of the same data in multiple formats to satisfy each tool's specific data-format constraints.

The implementation of such systems can require development of a significant amount of custom code, which is expensive to develop and maintain. The implementation often is constrained by the least common denominator of the components being connected within the system.

Because of the significant investment in developing software for these types of systems, it is common to add a component or tool that is forced to conform to an existing format used in the system. This conformance constraint may introduce inefficiencies or unnecessarily limit the functionality of the new component. Because so much of the implementation of such a system has been left to the system owner, the full benefit of the system is seldom realized. The system users, developers, and owners spend significantly more time on implementation and integration specifics instead of using the system for its intended purpose.

Simplifying the Ecosystem

One solution to simplifying the development and maintenance of the ecosystem is to use components from a single supplier, assuming that it would already have solved the component-to-component data-flow issues. While this may be true, a survey of most systems

in use today will show they utilize a diverse mix of tools from multiple suppliers or custom in-house tools designed to meet a focused need or application.

Additionally, it is not reasonable to expect that even the ATE component of the system is a single-platform model or type. Most system owners do not expect a single supplier to be able to provide the ideal solution for every step in the process for every device, at any point in time.

A more realistic solution is making sure the integration effort for each data generator or consumer component in the system is minimized. One method to minimize the integration time for each component in the system is to utilize industry-standard data formats whenever available.

The use of standard data formats between nodes of the system allows the data pipe between process nodes to be component agnostic and eliminates the need for development of data translation infrastructure. This permits the end system user the flexibility to choose the mix of tools that provides the best alignment of features to specific needs and to add or subtract capability or components over time. Some examples of currently available standards used in typical test-cell ecosystems include STDF for tested device result data, STIL (IEEE 1450) for digital CAD-to-test information, and SECS/GEM for equipment-to-host communications (**Figure 2**).

Standards

New or updated standards should focus on defining a minimum set of functionality required by all components including key event notification, input formats, and output formats. The standards should not limit the component supplier from offering differentiating functionality or place constraints on the implementation technology, given this is how suppliers differentiate their

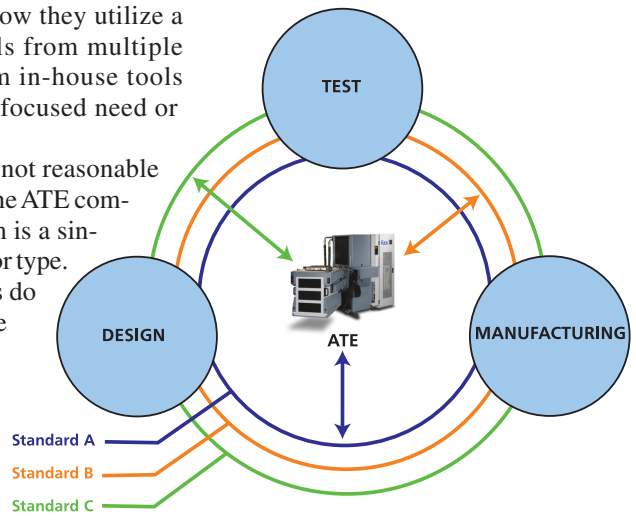


Figure 2. The Ideal, Well-Designed Ecosystem

components and provide real value to their customers.

The benefit of using standards within the ecosystem can be significantly reduced if there is lack of participation by the component supplier base. For example, last year the STDF steering committee approved extensions to the standard to support scan test records with the intent of improving integration of ATE results to third-party EDA scan diagnostic tools. While a number of ATE vendors now support the extension, not a single EDA vendor currently is showing plans to extend their products to utilize the standardized STDF data for the purpose of failure analysis. As a result, semiconductor manufacturers still will be required to maintain and develop custom software solutions to convert STDF data to proprietary data formats.

To provide the end user with a broad set of options for improving the efficiency and to add value to their products and processes, it is important for the continued adoption of standards by all tool vendors. ATE vendors cannot do this alone. They need the participation of their customers' design, test, and manufacturing organizations as well as all component suppliers in the ecosystem.

With the launch of SEMI's Collaborative Alliance for Semiconductor Test (CAST) in 2008, the industry has a chance to make progress in this area.

CAST should take on initiatives to support standardization of some trends that are driving new requirements in the ecosystem such as Protocol Aware ATE and Adaptive Test.

Extensibility via Application Programming Interfaces

Standards simply provide the overall construct for efficiently moving data and event information among components of the ecosystem. Standards do not define the functionality of the end component. While most component suppliers will provide a broad set of integrated capabilities to generate or consume data and events per the standard, there are occasions where the customer will need to customize the generation or consumption functionality of the component.

Some standards have predefined constructs and methods for the user to do this customization. For example, STDF includes Datalog Text Records (DTRs) and Generic Data Records (GDRs) which can be used to incorporate application-specific data within the standardized STDF data. To provide maximum flexibility to a component user, the supplier of that component should offer an extensive set of application programming interfaces (APIs) for interacting with the standardized data interfaces. These APIs should allow the user full access to the standardized data generated or consumed by the component.

The combination of these standardized APIs with other component-specific APIs provides a powerful means of extending the functionality of any component and then integrating that functionality back into the overall ecosystem. For example, a manufacturing domain yield monitoring tool may use its native functionality to monitor results in real time on one or more testers. That monitor tool then could be programmed to detect certain yield-loss scenarios and, if they occur, send instructions to the ATE to modify its test flow. That test-flow modification could include instructing the ATE to add a characterization test to the flow that is targeted at a specific set of conditions based on past results from the lot.

The ATE would use one or more component-specific APIs to set the characterization test setup based on the data provided by the manufacturing monitoring tool. The test program then may use its STDF API to place the results of the characterization test into the ATE result data stream using customized STDF records.

A second manufacturing analysis tool may use its STDF API to process the characterization data in the custom records of the STDF stream. That analysis tool also can read in data from the design domain or other manufacturing data to improve its capability to identify root cause of the failures.

When providing APIs to enable the user to extend or customize the component, it is important that the full functionality, normally available via the native capability of the component, be fully exposed via the API and that the basic use model of the platform be maintained. This not only should include the direct functional behavior of the operation being customized, but also any associated tools such as debug displays.

To minimize the end user's implementation effort, the task of the component supplier is to make sure the APIs for its component are well documented and support industry standards where they exist. Depending on the complexity of the interface and the functionality it controls, it may be important to provide reference designs or even more robust Software Developers Kits which might include additional debug costs.

When APIs are provided by the component supplier for either the standardized data interface or other component-specific functionality, it is important that these APIs do not impact the underlying performance of the component. For example, if the API were used to implement a native functionality offered by the component, the execution time performance of both solutions should be comparable.

Conclusion

An ATE platform is a key component within the test-cell ecosystem that spans design, test, and manufactur-

ing domains. For the ecosystem to provide maximum benefit to its users, it needs to support a variety of components from multiple suppliers and allow data to easily flow among those components. The data flow should not burden the ecosystem owner with developing extensive data converters and customized tools to support proprietary data formats, nor should it limit the user from using tools they desire.

One means of simplifying the data flow and providing a component-agnostic system is for each component to adopt industry-standard data formats. Each component should provide an extensive set of APIs so the native functionality of that component can be extended to meet user-specific needs while maintaining the industry data interface with other components. Such an implementation enables the user to integrate specialized third-party applications and develop specific customer tools. ATE vendors need to offer customers comprehensive solutions that complete the ecosystem by supplying extensible systems and partnering with industry-leading third parties.

About the Authors

John Morris is a senior application engineer in the Semiconductor Test Division of Teradyne. He has 25 years of experience in the ATE and semiconductor industries spanning design, test, and manufacturing. Mr. Morris attended Christian Brothers University and holds four U.S. patents. Teradyne Semiconductor Test Division, 38 River Rd., Suite 2, Essex Junction, VT 05452, 802-288-7426, e-mail: john.morris@teradyne.com

Roy Chorev is the software marketing manager in the Semiconductor Test Division of Teradyne. Prior to joining marketing, he was a senior software designer for several Teradyne ATE products. Mr. Chorev attended McGill University and Babson's Olin Graduate School of Business. Teradyne Semiconductor Test Division, MS 600-3, 600 Riverpark Dr., North Reading, MA 01864, 978-370-8450, e-mail: Roy.Chorev@teradyne.com