

REAL TIME DISPATCHING - A CATALYST TO ASSEMBLY TEST MANUFACTURING EXECUTION AUTOMATION

Bala Iyer

ATTD Automation
Intel Corporation
Chandler, AZ 85226, USA

Binay Dash

ATTD Automation
Intel Corporation
Bangalore, India

ABSTRACT

Intel's Assembly Test (AT) factories have long relied on legacy applications for lot kitting out of inventory locations. The accuracy, usability and product enhancement cycle times of these applications were constrained by data latency, inflexibility in business rule implementation and maintenance. To address these issues we successfully leveraged the Real Time Dispatching (RTD) application framework to provide a variety of kitting capabilities. We describe these projects and the new capabilities enabled through the use of the RTD framework in this paper.

1 INTRODUCTION

Integrated circuits such as Intel's CPU and Chipset products go through several major sets of manufacturing operations. First the chip is patterned on a wafer in the Fab process. Then there is a wafer level test step which Sorts the chips into bins of different performance characteristics. Then in a Die Preparation (DP) step wafers are diced and the individual die mounted on a tape which is wound around a reel. In the Assembly process, the dies are mounted and soldered on to a substrate package. The next major step is Unit Test where the performance of the packaged unit is validated. The final step is the Finish operation where the units are fused and marked to meet the configuration specified in a customer order.

In Intel's manufacturing and execution planning paradigm the Fab, Sort, and Die Preparation factories manufacture to forecast while the Assembly, Test and Finish (AT) operations build to order. This ensures that the very capital intensive Fabs operate at maximum capacity utilization while the AT factories maximize responsiveness to changing customer preferences. In order to accomplish this each AT factory maintains two major Work in Process inventory locations. The first is at the start of the assembly process and is called the Tape and Reel Die Inventory or TRDI for short. The second is after the Unit Test process and just before the Finish steps and is called the Semi Finished Goods Inventory or SFGI for short. Following the Finish operations, the units are packed into customer orders and sent to component warehouses and from there dispatched to customer sites. Figure 1 shows the flow of material through the various manufacturing steps as described above.

As shown in Figure 1, sorted wafers from multiple Fabs go to a smaller number of Die Preparation factories. Material in tape and reel format is sent from the Die Preparation factories to one of several AT factories. The material is released from TRDI when there is demand pull. It is assembled and tested and temporarily stored in SFGI. To allow maximize responsiveness to customer order changes, the final configuration to a specific customer order is done post SFGI.

The focus of this paper is the automation of the kitting execution process out of the TRDI and SFGI inventory locations in the AT factories. Intel's AT Factories had been for the last several years relying on legacy applications for their kitting business process automation. Work in Process (WIP) status and other Master Data like Product, Route and Operations residing in shop floor manufacturing execution system

WorkStream are key inputs to these business decisions. One of the major constraints with the legacy applications was the latency of 15-30 minutes in getting the WorkStream data feed. This latency adversely affects the accuracy and usability of kitting applications especially with low volume, high mix products – for example, lot batches suggested for release would sometimes become invalid or unavailable by the time the Manufacturing Specialist (MS) picked them or available lots would not be suggested. Invalid batches caused compliance and lot mixing issues as MS by-passed the system trying to move the lots quickly through the line. The second drawback of the legacy applications was that business rule development was complicated and managed by the software vendors thereby making product enhancements time consuming and expensive.

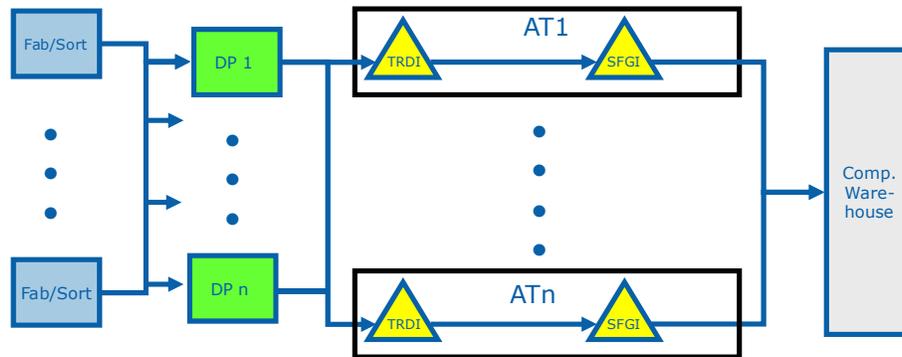


Figure 1: 10000 ft. view of Intel's Manufacturing Supply Chain

In order to tackle these issues, the Intel team decided to adopt RTD (Applied Materials 2011), a well established vendor product (Li et al. 2005; Jindal et al. 2001), as the framework for implementing kitting applications in AT. The key drivers for selecting RTD were that it had a built-in real time data tapping mechanism into WorkStream and other databases to source necessary data and a fast and flexible rule engine to encode complex operating logic. The Intel team also adopted a new user interface framework which brought in added advantages of development efficiency, enhanced look and feel, improved logging with external communication mechanism and improved security features. Leveraging these extensible frameworks, the Intel team has been successful in implementing kitting solutions across all AT factories. We describe these projects and the new capabilities enabled through the use of the RTD framework in the following sections of this paper. We start in Section 2 with an explanation of the common platform on which these capabilities were delivered. Sections 3 and 4 each cover a major factory kitting capability and explain how the RTD framework was leveraged in the design.

2 PLATFORM AND ARCHITECTURE DESIGN

A key objective adopted early in the design of the RTD projects was that they would be delivered on a common, low cost, scalable, reliable, available, and maintainable platform that would be standardized across all the AT implementations. First, the team decided to co-host all RTD projects at a factory site on a single cluster of servers. The team's previous experience in RTD and platform scalability analysis from Fab implementations provided enough confidence to justify this co-hosting strategy even though there were still many unknowns with the usage model in AT. Co-hosting significantly reduced the capital and support cost associated with providing these solutions across the AT factory network. Second, the team decided to architecturally model the RTD application, the optimization solver engine, and associated user interfaces (UIs) as a black box communicating via a middleware integration adaptor (ATMIA) to the rest of the AT automation suite. Third, an Oracle™ Database (DB) was used as the configuration and data pre-stage location for non-WorkStream data to enable high availability and standard data integration for all the projects. Fourth, a standard UI Framework was adopted. This allowed all project's UIs to standardize look and feel, as well as application integration. Fifth, a handful of new adapter components (OPL™

Wrapper, Timer Service, Dispatch Adapter, Service Manager) were developed to enable a flexible system that could support multiple projects running in parallel on a co-hosted platform. Finally, the solver, Oracle DB, RTD application, and adapter components architecture were optimized for high availability by hosting all components that were Microsoft™ Cluster Service compliant on a two node cluster, and hosting RTD components on a separate two node primary/backup failover pair. The resulting architecture, shown in Figure 2, is highly flexible and scalable to support multiple kitting capabilities across multiple ATM factories on a single site.

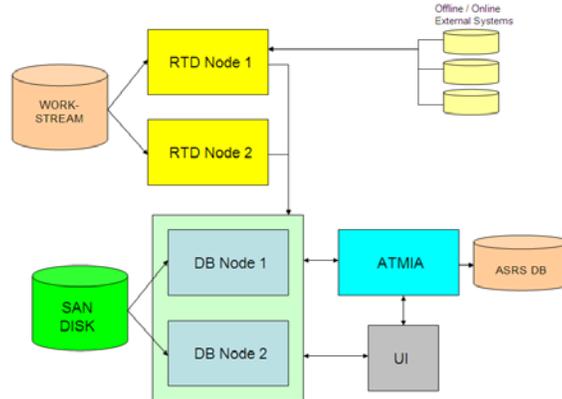


Figure 2: RTD system architecture for AT

3 TAPE REEL DIE INVENTORY (TRDI) STORE KITTING

As shown in Figure 1, TRDI is the first WIP inventory location within AT manufacturing. Dies produced by slicing the wafers are stored in reels and staged before the assembly route. The first operation after TRDI is the chip-attachment module where the die is attached to the substrate, before further assembly and test operations. There are several decision options for meeting the customer demand requests. The product-mix has to be decided to manage the loading across alternate routes based on testing needs and appropriate cycle times to meet customer demand in a timely manner. Material release from TRDI must take into account customer demand and due dates, WIP in the line, and cycle time through the different routes. It is also important to ensure high utilization of the expensive chip attachment equipment. The Chip Attachment Module (CAM) equipment is designed to process the TRDI reels in batches of several parallel lanes which have to be operated simultaneously. For optimal utilization of the equipment, the reels need to be of equal lengths. The MS in TRDI store can take certain maximum number of TRDI reels and splice them to get desired reel lengths. The TRDI reels which are spliced into the lanes of a batch have to follow certain merge rules based on their characteristics and are consumed only in a certain priority order to meet customer demand requests.

The TRDI store kitting capability automates the complex selection, merging and kitting decision processes and recommends batches of reels to be kitted out for splicing or loading into the chip attachment module. Once the kitting instruction has been printed, the TRDI personnel will retrieve the required reels from the TRDI shelves and scan the reel identification (ID) and do the necessary splicing within the reels to create the Assembly lot physically ready to be loaded in CAM. After this the Material Handling personnel will move the Assembly lot to the desired CAM equipment queue.

The main components of TRDI kitting solution are User Interface, RTD rules and reports and an optimization based solver. We describe each of these in subsequent sections.

3.1 TRDI Store Kitting User Interface

The TRDI store kitting User Interface provides options for factory personnel to configure the kitting system based on their needs. The users are provided with features to configure parameters like:

- Product kitting requirements like lot sizes, reel sizes, lane variances, etc.
- Effective Sort to Assembly product mapping
- Merge rules based on product attributes and maximum number of lots that can be merged
- Kitting related configuration like quantities, splicing options, etc., and generation of the prioritized kitting list. There are options to view the kitting list and print it out.

The User Interface stores these configurations on the configuration Operations Database (OPS DB).

3.2 TRDI RTD Rules and Reports

RTD reports are used to pull all the necessary data like demand, ship outs, inventory, product configurations, etc. The inventory and WIP data is available in the RTD repository on a real time basis from WorkStream. However for the other data items RTD needs to interact with external data systems to get relevant data.

After extraction, the data is processed by RTD reports which generate the data input for the ILOG™ batching solver and data quality reports. The RTD reports primarily consist of:

- Solver data feed rule which pre-processes all the relevant lots mapping to the demands and groups as per merge constraints, so that the solver can independently run the demand netting versus batch creation iterations
- A rule which calculates the Quantity To Go (QTG) at TRDI
- Data quality reports for problems like missing attributes, broken mappings, missing rank-lists and guideline in the source data

3.3 TRDI Solver

The TRDI optimization solver determines how best the candidate reels can be spliced and/or batched together to load the four lanes on the CAM equipment to meet Customer Demand. The solver is constrained by parameters such as optimum and minimum lot size, and allowable lane variances across lanes, lane pairs, etc. which are entered by the factory personnel. The Solver will try to maximize the total die in the batch and simultaneously minimize the differences of die quantities in the lanes and lane pairs and avoid remnant lot sizes less than the minimum lot size. The logic includes prioritization based on FIFO. If there are TRDI reels of same size but different age, the solver will try to pick the older reels to create the batch.

4 SEMI FINISHED GOODS INVENTORY (SFGI) STORE KITTING

SFGI is the last store location within AT manufacturing and is located before the Finish product route segment. Release of material out of SFGI therefore plays a critical role in meeting customer demand in a timely manner. While there are many feasible options for material release to meet demand, choosing the option that minimizes manufacturing costs requires complex logic. For certain products, configure-to-order decisions have to be made while releasing material from this store. For example a product of higher configuration may need to be fused to a lower configuration to meet the customer demand in a timely manner. Release of lots also has to take into consideration the alternate route loading requirements from a sampling perspective for specific products. Although a majority of the demands have a due date in line with the factory cut-off date of Saturday 7:00pm, there are certain portions of the demand that have due dates on various other days in the week due to product and customer priorities.

For meeting certain demands there can be different feasible combinations of finish products and test products and hence the selection logic has to comprehend route cycle times. The SFGI store kitting capability automates the complex selection, merging and kitting decision processes. The decision in terms of a kitting list of lot batches prioritized based on different selection, merging and kitting conditions and aggregated to meet demand line items is sent to the controls system of the Automated Storage and Retrieval System (ASRS) which holds the physical inventory of the material. Whenever there is a need to kit semi

finished goods out of the SFGI store, the MS at ASRS looks up the kitting list and selects required lots and subsequently retrieves these lots from their physical storage, collects these lots and takes care of ticketing and release to the Finish route.

The main components of automated SFGI store kitting solution are the User Interface, RTD rules and reports, ASRS and its Controls. We describe each of these in subsequent sections.

4.1 SFGI Store Kitting User Interface

The SFGI store kitting User Interface provides options for factory personnel to configure the kitting system based on their needs. The users are provided with features to configure product level parameters like:

- Lot sizes and SFGI cutoff day/time
- Moisture exposure time limit rules
- Merge rules based on product attributes and maximum number of lots that can be merged
- Platform validation sampling percentage overrides and priorities
- Critical product priority (CPP) requests to close products early ahead of original schedules

The User Interface stores these configurations in the OPS DB.

4.2 SFGI RTD Rules and Reports

RTD reports pull all the necessary data like demand, ship outs, WIP, store inventory, product configurations, etc. After extraction, the data is processed by RTD reports which generate the kitting suggestion. The RTD reports primarily consist of:

- Demand Netting rule which comprehends the demand at SFGI from the factory demand, the ship outs and WIP already in the Finish route
- Lot Generation rule which looks into demand at SFGI level and the lots available in ASRS to generate a list of lots that can be kitted out to the manufacturing line
- Data Quality report for missing attributes, broken mappings, etc. in the source data

4.3 SFGI ASRS Controls

The ASRS is a vendor supplied stocking unit, containing multiple crane aisles, capable of automatically or manually storing and retrieving semi finished goods. The controls system that drives the ASRS has a database component that contains current and historical bin and tray stack information, as well as information related to the current status of the individual crane aisles. It receives information regarding requests to retrieve required products through a message queue interface to the kitting rules. It responds to these retrieval requests by identifying the lots and sending them to a collection unit using a conveyor.

5 CONCLUSION

The Intel team has successfully implemented both TRDI and SFGI kitting applications across all of Intel AT factories. The applications continue to evolve to meet business needs with additional capabilities while meeting the factories reliability, availability, and performance targets. A key learning was that the RTD framework is well suited to deliver kitting applications for AT factories. The built in capabilities for real time data sourcing and the flexible rule engine helped the team deliver highly customized business logic like multi-chip package kitting, even consumption of lots of different sizes, meeting Sampling volume needs in SFGI, easily and quickly without requiring any long development from the supplier. RTD based solutions can also be easily adapted as business needs change. A major benefit of using RTD as a development platform is that it is easy for industrial engineers who are familiar with factory business process to learn and use. By re-using common macros and rules across multiple projects, development teams were also able to leverage each other's work to quickly deliver solutions. The real time data feed from

MES ensured that the lot list presented to manufacturing staff was up to date with inventory on the floor thereby driving a high degree of compliance with system recommendations.

These capabilities of RTD together with the ability to integrate external math optimization and simulation engines provide Intel developers a powerful framework to address emerging needs in the AT production scheduling roadmap.

REFERENCES

- Applied Materials. 2011. "APF RTD/ APF Reporter." Accessed September 23. <http://appliedmaterials.com/services-software/library/apf-rtd-apf-reporter>.
- Jindal, A., P. A. Flores, M. L. Cain, and M. Krishna. 2001. "Using RTD For Manual Order Queue Administration." Presented at *Brooks Automation Conference*.
- Li, B., J. Wu, M. P. Herring, and C. Mouli. 2005. "Improving Factory Productivity through Agile WIP Scheduling and Dispatching in 300mm Wafer Manufacturing." International Sematech Manufacturing Initiative.

AUTHOR BIOGRAPHIES

BALA IYER is a Principal Engineer in Intel's Assembly Test Technology Development Group. His current responsibility includes defining and delivering automation and MHS solutions to Intel's Assembly Test factories. He has a B. Tech from Indian Institute of Technology, M.S from Carnegie Mellon University, and a Ph.D. from Massachusetts Institute of Technology, all in Mechanical Engineering. His email address is bala.s.iyer@intel.com.

BINAY DASH is an Automation Engineer in Intel's Assembly Test Technology Development Group. His current responsibility includes designing and delivering Factory Scheduling automation solutions to Intel's Assembly Test factories. He graduated in mechanical engineering from National Institute of Technology, India (Bhopal), post-graduated in Industrial Engineering from National Institute of Industrial Engineering, India (Mumbai) and is currently pursuing his Ph.D. from Indian Institute of Technology. His email address is binay.dash@intel.com.