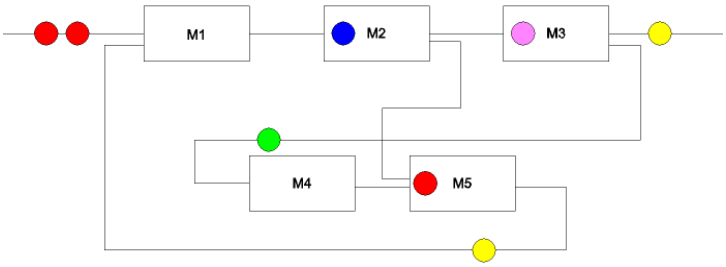

FabSim

A (semiconductor) factory simulator

FabSim Structure

FabSim Interactive is a compact fab simulator contained in a single C++ Windows dynamic link library. The dll is controlled by a supervisor program written in Borland Delphi. The simulator structure is depicted in figure 1. The first function called by the supervisor is *Init*, which reads toolset data from a file, generates toolset objects from a toolset class (one per toolset), tool (machine) objects from a machine class and sets all start values. Function *MachineCount* is used to change tool count values, function *Sim* starts the simulation run, after simulation function *GetOutWafCount* returns simulation data to the supervisor. During the simulation FabSim.dll continuously sends data to the supervisor via the DLL_MESSAGE procedure. Currently 24 functions are available for interactive simulation control.

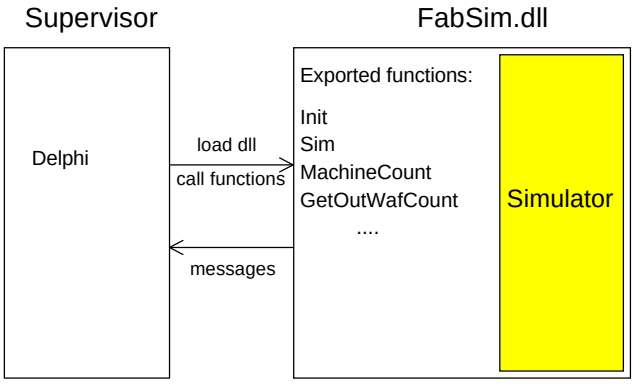

Figure 1: Structure of FabSim Interactive

Simulation Engine

FabSim, in contrast to most of the general purpose simulation programs, applies “fixed-increment time advance” as a discrete-event simulation model. Simulation proceeds in user definable fixed time steps, e.g. 1 minute. During each time step all state changes are recalculated, triggered by an event happening. Events may be loading or unloading wafer lots from a tool, start of a new lot or release of a lot from the fab. Machines change state, e.g. decrement the time counter if a process runs or go from IDLE to PROCEES, from PROCESS

to DOWN etc.. If no state change is required, the respective function call returns immediately. Wafer lots are exchanged between the machines as objects derived from a lot class. A scheduler distributes all lots to the toolsets as prescribed by the process flow chart.

Borrowed from SystemC is the way to exchange data between all objects. Only when all calculations for the current time step inside all objects are finished, quasi in an infinitesimal short time period, immediately before the next one minute time step commences, all data are exchanged. Thus all calculations are always based on the data available at the beginning of the time step. This method relives the simulator from deciding which event to process first. The approach resembles an electronic system which runs synchronized by a steady clock.

Input, Output and Options

Input (minimum requirement)

List of toolsets (including tool data, 128 toolsets, 50 tools each)

Run sheets (one per process, unlimited number of processes)

Lot start sequence

Input (optional)

From-to matrix of lot transport time

Tool count optimization data

Operator data

Output (standard)

Lots processed

Lots scrapped

Toolset and tool usage

Output (optional)

Log of each process step

WIP versus time

Tools available and tools active versus time

Occupation of toolset buffers versus time

Optimization log and tool count

Output data evaluation and visualization

EXCEL macros are available to plot data

Toolset Options

Batch operation (size, minimum lot size, maximum lot waiting time)

Setup (implant, lithography, CMP, dispatching with setup avoidance option)

Mtbf, mttr with exponential or gamma time distribution

Scheduled downtimes

Lot Flow Options

Constant WIP (CONWIP) by controlled lot release

Lot release control by watching overflow of a selected toolset buffer

Three priority levels (standard, move to front of buffer, reserve tool)

Internal dispatching rules first in, first out (FIFO), shortest processing time first (SPTF), operation due date (ODD), earliest due date (EDD), weighted shortest processing time (WSPT)

Rework with dedicated rework run sheets

Fab yield (lots ready versus lots started within fab) and lot yield (wafers ready versus wafers started within lot) with statistically determined loss

Monitoring of time constraints between process steps

Automated selection of alternative tools

Data evaluation may start only after fab stabilization period has passed

Processing time may vary statistically

Customer defined dispatching rules in Cdispatch.dll

Simulation modes

Batch mode: Run a sequence of simulation jobs, option entries read from a batch file

Interactive mode: stop simulation on predetermined events, change parameters and continue simulation

Supervisor controlled simulation: Implement control or simulation strategies into a supervisor program, which controls the simulator and evaluates the simulation data

Start simulation with empty fab or after preloading an actual fab status

For further information please contact:

Prof. Dr. Holger Vogt
Fraunhofer-Institut IMS
Finkenstr. 61
47057 Duisburg, Germany
phone: 49 203 3783 100
email: info@fabsim.com,
web site: www.fabsim.com

Version 3.0, November 2004