DEVELOPMENT OF A SEAMLESSLY INTEGRATED FACTORY PLANNING SOFTWARE TOOL (PROTOTYPE) TO EVALUATE AND OPTIMIZE SURFACE MOUNT MANUFACTURING LINES

By:
Vijay Mehra
B.S. Operations Research and Industrial Engineering -- Cornell University (1990)
B.A. Economics -- Cornell University (1990)

Submitted to the Departments of Management and Civil & Environmental Engineering (Intelligent Engineering Systems Laboratory)
In Partial Fulfillment of the Requirements for the Degrees of
MASTER OF SCIENCE IN MANAGEMENT and MASTER OF SCIENCE IN CIVIL AND ENVIRONMENTAL ENGINEERING

at the MASSACHUSETTS INSTITUTE OF TECHNOLOGY
May 1995
© Massachusetts Institute of Technology, 1995. All rights reserved

Signature of Author
/MIT Sloan School of Management (Manufacturing Track)
Department of Civil & Environmental Engineering
(Intelligent Engineering Systems Laboratory)
May, 12 1995

Certified by
Professor John Williams
Department of Civil & Environmental Engineering
(Intelligent Engineering Systems Laboratory)
Thesis Supervisor

Certified by
Professor Larry Wein
MIT Sloan School of Management

Accepted by
Professor Joseph M. Sussman
Graduate Officer
Department of Civil and Environmental Engineering

Accepted by
Jeffrey A. Barks
Associate Dean, Sloan Masters and Bachelor's Program

JUN 21 1995
Abstract

DEVELOPMENT OF A SEAMLESSLY INTEGRATED FACTORY PLANNING SOFTWARE TOOL (PROTOTYPE) TO EVALUATE AND OPTIMIZE SURFACE MOUNT MANUFACTURING LINES

By

Vijay Mehra

Submitted to the Departments of

Management

and

Civil & Environmental Engineering (Intelligent Engineering Systems Laboratory)

on May 12, 1995

In Partial Fulfillment of the Requirements for the Degrees of

Master of Science in Management

and

Master of Science in Civil and Environmental Engineering

ABSTRACT

Surface Mount Technology (SMT) has revolutionized manufacturing in the electronics industry and lies at the core of most electronics component assembly lines today. An SMT line might comprise of a combination of the following operations (which are typically linked by an automated conveyor):

• Solder paste application
• High-speed automated assembly
  -- "Common" size/shape components
  -- "Odd" size/shape components
• Manual assembly
• Thermal treatment
• Component test
• Unit assembly
• Unit test
• Unit pack
• Unit shipment

Each of the above operations might be accomplished by using one or more types of equipment and each operation on each piece of equipment could typically constitute from anywhere between one to ten and one to a thousand sub-steps.

Needless to say, the sheer number of total steps and sub-steps (and their interdependencies) that need to be accounted for from the beginning to the end of the line is overwhelming and any attempt to evaluate the performance of a given line (in terms of expected throughput, expected bottlenecks, expected WIP, etc.) will have to use sophisticated stochastic simulation tools in order to generate accurate and meaningful results. These tools emulate important attributes of an SMT line such as:

• The steps and sub-steps of each operation.
• The interdependencies between each of the operations.
• The expected failures/down times corresponding to each operation and the subsequent impacts on upstream and downstream operations.
Abstract

- The flow of product from one operation to another.
- The setup times (for each operation) when the line has to change from one product to another product.

However, even the most popular commercially available simulation tools that are used in the industry today have significant learning curves associated with them. To climb these learning curves takes a significant amount of time and effort and hence it is typical for a select group of people within an organization to develop this functional expertise and become "simulation experts". The line managers and engineers -- who need to periodically assess the performance of their lines and determine what changes need to be made to line layouts, product mixes and process flows to enhance line performance -- have to therefore hire the experts in order to use simulation tools to evaluate their manufacturing environments. This process takes time (each simulation effort is undertaken as a "mini-project"), is costly (the manager has to bear the project costs) and it can even come in the way of managers resorting to the use of simulation tools to evaluate their manufacturing environments.

This thesis aims at enabling line managers/engineers to directly evaluate the performance of their lines by providing them with a software tool to seamlessly configure, execute and evaluate complex stochastic simulation models representing their manufacturing environments. This tool (referred to as the Prototype Factory Planning Tool) will thereby transparently bring stochastic simulation into frequent use. It will transform stochastic simulation into an enabling technology used by managers when they are faced with critical decisions (pertaining to line layouts, product mixes, process flows, etc.) in order to evaluate and optimize the performance of their manufacturing lines.

In addition, this thesis also addresses the software requirements and software architecture pertaining to the enhancement of the Prototype Factory Planning Application into a full-blown version. The latter application will permit the user to evaluate the line performance against product cost and product quality metrics in addition to the throughput related metrics available in the prototype application. (The latter application was not developed during the author's internship since it is a multi-person, multi-year effort and is hence beyond the scope of his thesis).

Thesis Supervisor: Professor John Williams

Title: Associate Professor of Civil & Environmental Engineering
TABLE OF CONTENTS

1.0 INTRODUCTION .................................................................................................................. 14
  1.1 Problem Description ......................................................................................................... 14
  1.2 Thesis Goal ....................................................................................................................... 19
  1.3 Thesis Layout -- "Top-Down" Approach ........................................................................... 20

2.0 ACKNOWLEDGMENTS ........................................................................................................ 22

3.0 THESIS METHODOLOGY ................................................................................................. 24
  3.1 Chapter Overview .............................................................................................................. 24
  3.2 Gathering the Voice of the Customer ................................................................................ 24
  3.3 Literature Search/Industry Benchmark ............................................................................. 25
    3.3.1 Research Work at Some Universities/Corporations .................................................. 25
    3.3.2 Commercially Available Software Applications ....................................................... 27
    3.3.3 SEMATECH Efforts Relevant to Thesis Topic .............................................................. 29
  3.4 Determination of Pilot Customer Site ............................................................................... 30
  3.5 Overview of Primary Iterative Processes ......................................................................... 31
    3.5.1 Periodic Meetings -- Continuous Feedback from Customers .................................... 31
    3.5.2 Weekly Team Meetings -- "Yo One" ............................................................................ 33
    3.5.3 Dissemination of Project Information ........................................................................ 33
    3.5.4 Project Schedule Methodology .................................................................................. 34

4.0 THESIS RESULTS ............................................................................................................... 36
  4.1 Chapter Overview .............................................................................................................. 36
  4.2 Manufacturing Scenario Planning .................................................................................... 36
    4.2.1 Four Scenarios Seamlessly Configured and Evaluated .............................................. 37
    4.2.2 Cycle Time Reduction ................................................................................................ 39
  4.3 Platform for Future Factory Planning Software Tools ....................................................... 39

5.0 SOFTWARE REQUIREMENTS (HIGH LEVEL) ............................................................... 40
  5.1 Chapter Overview .............................................................................................................. 40
  5.2 Introduction ....................................................................................................................... 40
    5.2.1 Purpose ....................................................................................................................... 40
    5.2.2 Scope ........................................................................................................................... 41
    5.2.3 Definitions, Acronyms and Abbreviations ................................................................. 41
    5.2.4 References .................................................................................................................. 42
    5.2.5 Overview .................................................................................................................... 42
  5.3 General Description ......................................................................................................... 43
    5.3.1 Product Perspective .................................................................................................... 43
    5.3.2 Product Functions ...................................................................................................... 43
    5.3.3 User Characteristics ................................................................................................. 44
    5.3.4 General Constraints .................................................................................................. 44
    5.3.5 Assumptions and Dependencies ............................................................................... 45
<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>5.4 Specific Requirements</td>
<td>45</td>
</tr>
<tr>
<td>5.4.1 Functional Requirements</td>
<td>45</td>
</tr>
<tr>
<td>5.4.1.1 Line Layout</td>
<td>45</td>
</tr>
<tr>
<td>5.4.1.1.1 Introduction</td>
<td>45</td>
</tr>
<tr>
<td>5.4.1.1.2 Inputs</td>
<td>45</td>
</tr>
<tr>
<td>5.4.1.1.3 Processing</td>
<td>46</td>
</tr>
<tr>
<td>5.4.1.1.4 Outputs</td>
<td>46</td>
</tr>
<tr>
<td>5.4.1.2 Line Evaluation</td>
<td>46</td>
</tr>
<tr>
<td>5.4.1.2.1 Introduction</td>
<td>46</td>
</tr>
<tr>
<td>5.4.1.2.2 Inputs</td>
<td>47</td>
</tr>
<tr>
<td>5.4.1.2.3 Processing</td>
<td>49</td>
</tr>
<tr>
<td>5.4.1.2.4 Outputs</td>
<td>49</td>
</tr>
<tr>
<td>5.4.1.3 Board Assignment to Lines</td>
<td>50</td>
</tr>
<tr>
<td>5.4.1.3.1 Introduction</td>
<td>50</td>
</tr>
<tr>
<td>5.4.1.3.2 Inputs</td>
<td>50</td>
</tr>
<tr>
<td>5.4.1.3.3 Processing</td>
<td>50</td>
</tr>
<tr>
<td>5.4.1.3.4 Outputs</td>
<td>51</td>
</tr>
<tr>
<td>5.4.1.4 Board Family Generation</td>
<td>51</td>
</tr>
<tr>
<td>5.4.1.4.1 Introduction</td>
<td>51</td>
</tr>
<tr>
<td>5.4.1.4.2 Inputs</td>
<td>51</td>
</tr>
<tr>
<td>5.4.1.4.3 Processing</td>
<td>51</td>
</tr>
<tr>
<td>5.4.1.4.4 Outputs</td>
<td>52</td>
</tr>
<tr>
<td>5.4.1.5 Board Sequencing</td>
<td>52</td>
</tr>
<tr>
<td>5.4.1.5.1 Introduction</td>
<td>52</td>
</tr>
<tr>
<td>5.4.1.5.2 Inputs</td>
<td>52</td>
</tr>
<tr>
<td>5.4.1.5.3 Processing</td>
<td>52</td>
</tr>
<tr>
<td>5.4.1.5.4 Outputs</td>
<td>52</td>
</tr>
<tr>
<td>5.4.1.6 Front End Throughput Optimization</td>
<td>52</td>
</tr>
<tr>
<td>5.4.1.7 Scenario and Data Editing</td>
<td>53</td>
</tr>
<tr>
<td>5.4.1.7.1 Introduction</td>
<td>53</td>
</tr>
<tr>
<td>5.4.1.7.2 Inputs</td>
<td>53</td>
</tr>
<tr>
<td>5.4.1.7.3 Processing</td>
<td>54</td>
</tr>
<tr>
<td>5.4.1.7.4 Outputs</td>
<td>54</td>
</tr>
<tr>
<td>5.4.2 External Interface Requirements</td>
<td>55</td>
</tr>
<tr>
<td>5.4.2.1 User Interfaces</td>
<td>55</td>
</tr>
<tr>
<td>5.4.2.2 Software Interfaces</td>
<td>55</td>
</tr>
<tr>
<td>5.4.2.3 Communications Interfaces</td>
<td>55</td>
</tr>
<tr>
<td>5.4.3 Design Constraints</td>
<td>56</td>
</tr>
<tr>
<td>5.4.4 Other Requirements</td>
<td>56</td>
</tr>
</tbody>
</table>
# Table of Contents

6.0 SOFTWARE ARCHITECTURE (HIGH LEVEL) ................................................................. 57
   6.1 Chapter Overview ........................................................................................................ 57
   6.2 Architecture Requirements ...................................................................................... 57
   6.3 Architecture Assumptions ....................................................................................... 57
   6.4 Software Modules ...................................................................................................... 58
      6.4.1 Main User Interface ......................................................................................... 60
      6.4.2 Editor Modules ............................................................................................... 60
      6.4.3 Line Evaluator/Optimizer Interface ................................................................. 62
      6.4.4 Task Manager .................................................................................................. 62
      6.4.5 Board Assignment to Lines .............................................................................. 63
      6.4.6 Board Family Generation ............................................................................... 63
      6.4.7 Board Sequencing ........................................................................................... 64
      6.4.8 Component Allocation ..................................................................................... 64
      6.4.9 Placement Sequence Optimizer ......................................................................... 65
      6.4.10 Simulation Driver .......................................................................................... 65
      6.4.11 Simulation Model (Witness) ............................................................................ 66
      6.4.12 Line Evaluation Report Generator ................................................................. 67
      6.4.13 Cost Evaluation Tool ...................................................................................... 67
      6.4.14 Quality Evaluation Tool .................................................................................. 68
      6.4.15 Integer Programming Solver .......................................................................... 68
      6.4.16 NC Generation ............................................................................................... 69
      6.4.17 Setup Generation ............................................................................................ 69
      6.4.18 Data Transfer Interface ................................................................................... 70
      6.4.19 Factory Planning Data Base ............................................................................ 70
   6.5 Software Modules Developed for Pilot Application .................................................. 71
   6.6 Architectural Issues Pertaining to the Factory Planning Data Base .......................... 71

7.0 SOFTWARE REQUIREMENTS (SIMULATION DRIVER) ............................................. 77
   7.1 Chapter Overview ...................................................................................................... 77
   7.2 Introduction .............................................................................................................. 77
      7.2.1 Purpose ........................................................................................................... 77
      7.2.2 Scope .............................................................................................................. 77
      7.2.3 Definitions, Acronyms and Abbreviations ......................................................... 78
      7.2.4 References ....................................................................................................... 78
      7.2.5 Overview ......................................................................................................... 78
   7.3 General Description ................................................................................................... 79
      7.3.1 Product Perspective ......................................................................................... 79
      7.3.2 Product Functions ............................................................................................ 79
      7.3.3 User Characteristics ......................................................................................... 82
      7.3.4 General Constraints ....................................................................................... 82
      7.3.5 Assumptions and Dependencies ..................................................................... 82
   7.4 Specific Requirements ............................................................................................... 82
      7.4.1 Functional Requirements ................................................................................. 83
          7.4.1.1 Line Configuration Data Reading Function .............................................. 83
              7.4.1.1.1 Introduction ...................................................................................... 83
              7.4.1.1.2 Inputs .............................................................................................. 83
              7.4.1.1.3 Processing ........................................................................................ 83
              7.4.1.1.4 Outputs ............................................................................................ 83
Development of a Seamlessly Integrated Factory Planning Software Tool (Prototype) to Evaluate and Optimize Surface Mount Manufacturing Lines
<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>7.4.1.2 Production Data Definition Function</td>
<td>84</td>
</tr>
<tr>
<td>7.4.1.2.1 Introduction</td>
<td>84</td>
</tr>
<tr>
<td>7.4.1.2.2 Inputs</td>
<td>84</td>
</tr>
<tr>
<td>7.4.1.2.3 Processing</td>
<td>86</td>
</tr>
<tr>
<td>7.4.1.2.4 Outputs</td>
<td>86</td>
</tr>
<tr>
<td>7.4.1.3 Simulation Model Configuration &amp; Execution Function</td>
<td>86</td>
</tr>
<tr>
<td>7.4.1.3.1 Introduction</td>
<td>86</td>
</tr>
<tr>
<td>7.4.1.3.2 Inputs</td>
<td>86</td>
</tr>
<tr>
<td>7.4.1.3.3 Processing</td>
<td>90</td>
</tr>
<tr>
<td>7.4.1.3.4 Outputs</td>
<td>90</td>
</tr>
<tr>
<td>7.4.1.4 Line Evaluation Report Configuration Function</td>
<td>90</td>
</tr>
<tr>
<td>7.4.1.4.1 Introduction</td>
<td>91</td>
</tr>
<tr>
<td>7.4.1.4.2 Inputs</td>
<td>91</td>
</tr>
<tr>
<td>7.4.1.4.3 Processing</td>
<td>91</td>
</tr>
<tr>
<td>7.4.1.4.4 Outputs</td>
<td>91</td>
</tr>
<tr>
<td>7.4.1.5 Oracle Communication Logon Function</td>
<td>91</td>
</tr>
<tr>
<td>7.4.1.5.1 Introduction</td>
<td>91</td>
</tr>
<tr>
<td>7.4.1.5.2 Inputs</td>
<td>91</td>
</tr>
<tr>
<td>7.4.1.5.3 Processing</td>
<td>91</td>
</tr>
<tr>
<td>7.4.1.5.4 Outputs</td>
<td>91</td>
</tr>
<tr>
<td>7.4.1.6 Oracle Communication Logoff Function</td>
<td>92</td>
</tr>
<tr>
<td>7.4.1.6.1 Introduction</td>
<td>92</td>
</tr>
<tr>
<td>7.4.1.6.2 Inputs</td>
<td>92</td>
</tr>
<tr>
<td>7.4.1.6.3 Processing</td>
<td>92</td>
</tr>
<tr>
<td>7.4.1.6.4 Outputs</td>
<td>92</td>
</tr>
<tr>
<td>7.4.1.7 Integer Conversion to ASCII String Function</td>
<td>92</td>
</tr>
<tr>
<td>7.4.1.7.1 Introduction</td>
<td>92</td>
</tr>
<tr>
<td>7.4.1.7.2 Inputs</td>
<td>92</td>
</tr>
<tr>
<td>7.4.1.7.4 Processing</td>
<td>93</td>
</tr>
<tr>
<td>7.4.1.7.4 Outputs</td>
<td>93</td>
</tr>
<tr>
<td>7.4.1.8 Reverse String Function</td>
<td>93</td>
</tr>
<tr>
<td>7.4.1.8.1 Introduction</td>
<td>93</td>
</tr>
<tr>
<td>7.4.1.8.2 Inputs</td>
<td>93</td>
</tr>
<tr>
<td>7.4.1.8.3 Processing</td>
<td>93</td>
</tr>
<tr>
<td>7.4.1.8.4 Outputs</td>
<td>93</td>
</tr>
<tr>
<td>7.4.1.9 Retrieve Route Cycle Time Data Structure Information Function</td>
<td>94</td>
</tr>
<tr>
<td>7.4.1.9.1 Introduction</td>
<td>94</td>
</tr>
<tr>
<td>7.4.1.9.2 Inputs</td>
<td>94</td>
</tr>
<tr>
<td>7.4.1.9.3 Processing</td>
<td>94</td>
</tr>
<tr>
<td>7.4.1.9.4 Outputs</td>
<td>95</td>
</tr>
</tbody>
</table>
# Table of Contents

7.4.1.10 Comparison of Records (by production sequence) in the Product Information Data Structure .......... 95
  7.4.1.10.1 Introduction...................................... 95
  7.4.1.10.2 Inputs........................................... 95
  7.4.1.10.3 Processing........................................ 95
  7.4.1.10.4 Outputs........................................... 95

7.4.1.11 Comparison of Records (by process step) in the Temporary Data Structures................................ 95
  7.4.1.11.1 Introduction...................................... 95
  7.4.1.11.2 Inputs........................................... 96
  7.4.1.11.3 Processing........................................ 96
  7.4.1.11.4 Outputs........................................... 96

7.4.1.12 Build Routing/Cycle Time File Function.......... 96
  7.4.1.12.1 Introduction...................................... 96
  7.4.1.12.2 Inputs........................................... 96
  7.4.1.12.3 Processing........................................ 96
  7.4.1.12.4 Outputs........................................... 97

7.4.1.13 Build Production Order File Function.............. 97
  7.4.1.13.1 Introduction...................................... 97
  7.4.1.13.2 Inputs........................................... 97
  7.4.1.13.3 Processing........................................ 98
  7.4.1.13.4 Outputs........................................... 98

7.4.1.14 Build ASCII File Function.......................... 98
  7.4.1.14.1 Introduction...................................... 98
  7.4.1.14.2 Inputs........................................... 99
  7.4.1.14.3 Processing........................................ 99
  7.4.1.14.4 Outputs........................................... 100

7.4.1.15 Run Simulation Function............................ 100
  7.4.1.15.1 Introduction...................................... 100
  7.4.1.15.2 Inputs........................................... 100
  7.4.1.15.3 Processing........................................ 101
  7.4.1.15.4 Outputs........................................... 101

7.4.1.16 Extract Results Data Function....................... 101
  7.4.1.16.1 Introduction...................................... 101
  7.4.1.16.2 Inputs........................................... 101
  7.4.1.16.3 Processing........................................ 101
  7.4.1.16.4 Outputs........................................... 101

7.4.1.17 Store Report Data in Data Base Function.......... 102
  7.4.1.17.1 Introduction...................................... 102
  7.4.1.17.2 Inputs........................................... 102
  7.4.1.17.3 Processing........................................ 102
  7.4.1.17.4 Outputs........................................... 102
7.4.2 External Interface Requirements........................................... 112
   7.4.2.1 User Interfaces....................................................... 112
   7.4.2.2 Software Interfaces.............................................. 112
   7.4.2.3 Communications Interfaces................................. 113

7.4.3 Design Constraints............................................................... 113

8.0 SOFTWARE REQUIREMENTS (TASK MANAGER)................................. 114
8.1 Chapter Overview........................................................................ 114
8.2 Introduction............................................................................... 114
   8.2.1 Purpose........................................................................ 114
   8.2.2 Scope........................................................................... 114
   8.2.3 Definitions, Acronyms and Abbreviations..................... 115
   8.2.4 References................................................................. 115
   8.2.5 Overview....................................................................... 115
8.3 General Description..................................................................... 115
   8.3.1 Product Perspective..................................................... 115
   8.3.2 Product Functions....................................................... 116
   8.3.3 User Characteristics.................................................. 116
   8.3.4 General Constraints.................................................. 116
   8.3.5 Assumptions and Dependencies................................ 117
8.4 Specific Requirements.................................................................. 117
   8.4.1 Functional Requirements.............................................. 117
      8.4.1.1 Software Execution Function............................... 117
         8.4.1.1.1 Introduction............................................. 117
         8.4.1.1.2 Inputs.................................................... 117
         8.4.1.1.3 Processing............................................... 117
         8.4.1.1.4 Outputs.................................................. 119
      8.4.1.2 Count Scenarios Function.................................. 120
         8.4.1.2.1 Introduction............................................ 120
         8.4.1.2.2 Inputs.................................................... 120
         8.4.1.2.3 Processing............................................... 120
         8.4.1.2.4 Outputs.................................................. 121
      8.4.1.3 Read Scenarios Function.................................... 120
         8.4.1.3.1 Introduction............................................ 120
         8.4.1.3.2 Inputs.................................................... 120
         8.4.1.3.3 Processing............................................... 120
         8.4.1.3.4 Outputs.................................................. 121
6.4.2 External Interface Requirements....................................... 121
   8.4.2.1 User Interfaces...................................................... 121
   8.4.2.2 Software Interfaces.............................................. 121
   8.4.2.3 Communications Interfaces..................................... 121
8.4.3 Design Constraints............................................................... 121
Table of Contents

9.0 SOFTWARE REQUIREMENTS (USER INTERFACE) ........................................... 123
  9.1 Chapter Overview ................................................................................ 123
  9.2 Introduction ......................................................................................... 123
    9.2.1 Purpose ......................................................................................... 123
    9.2.2 Scope ............................................................................................ 123
    9.2.3 Definitions, Acronyms and Abbreviations ...................................... 124
    9.2.4 References .................................................................................... 124
    9.2.5 Overview ....................................................................................... 124

  9.3 General Description ............................................................................. 124
    9.3.1 Product Perspective ...................................................................... 124
    9.3.2 Product Modules, Screens and Their Functions ......................... 125
    9.3.3 User Characteristics .................................................................... 127
    9.3.4 General Constraints ...................................................................... 127
    9.3.5 Assumptions and Dependencies .................................................. 128

  9.4 Specific Requirements ........................................................................ 128
    9.4.1 Functional Requirements ............................................................. 128
      9.4.1.1 Main User Interface Module ................................................... 128
        9.4.1.1.1 Main Menu Screen ......................................................... 128
        9.4.1.1.1.1 Introduction ............................................................... 128
        9.4.1.1.1.2 Inputs ........................................................................... 131
        9.4.1.1.1.3 Processing ................................................................. 132
        9.4.1.1.1.4 Outputs ....................................................................... 132
      9.4.1.2 Line Evaluator/Optimizer Module ....................................... 132
        9.4.1.2.1 Scenario and Report Selection Screen ......................... 132
          9.4.1.2.1.1 Introduction ............................................................... 132
          9.4.1.2.1.2 Inputs ....................................................................... 134
          9.4.1.2.1.3 Processing ................................................................. 134
          9.4.1.2.1.4 Outputs ..................................................................... 134
      9.4.1.3 Line Evaluation Report Generator Module ........................... 134
        9.4.1.3.1 Line Throughput Results Screen .................................. 134
          9.4.1.3.1.1 Introduction ............................................................... 134
          9.4.1.3.1.2 Inputs ....................................................................... 137
          9.4.1.3.1.3 Processing ................................................................. 137
          9.4.1.3.1.4 Outputs ..................................................................... 137
        9.4.1.3.2 Product Related Results Screen ................................... 137
          9.4.1.3.2.1 Introduction ............................................................... 137
          9.4.1.3.2.2 Inputs ....................................................................... 140
          9.4.1.3.2.3 Processing ................................................................. 140
          9.4.1.3.2.4 Outputs ..................................................................... 140
        9.4.1.3.3 Machine Related Results Screen ................................... 141
          9.4.1.3.3.1 Introduction ............................................................... 141
          9.4.1.3.3.2 Inputs ....................................................................... 143
          9.4.1.3.3.3 Processing ................................................................. 143
          9.4.1.3.3.4 Outputs ..................................................................... 143

Development of a Seamlessly Integrated Factory Planning Software Tool (Prototype) to Evaluate and Optimize Surface Mount Manufacturing Lines
9.4.1.3.4 WIP Related Results Screen ........................................ 144
  9.4.1.3.4.1 Introduction .................................................... 144
  9.4.1.3.4.2 Inputs .................................................................. 146
  9.4.1.3.4.3 Processing ............................................................ 146
  9.4.1.3.4.4 Outputs ............................................................... 146

9.4.1.4 Editor Module .................................................................. 147
  9.4.1.4.1 Scenario Selection Screen .......................................... 147
    9.4.1.4.1.1 Introduction ....................................................... 147
    9.4.1.4.1.2 Inputs .................................................................. 149
    9.4.1.4.1.3 Processing ............................................................ 149
    9.4.1.4.1.4 Outputs ............................................................... 149
  9.4.1.4.2 Scenario Data Editor Screen ......................................... 149
  9.4.1.4.3 Report Selection Screen .............................................. 151
    9.4.1.4.3.1 Introduction ....................................................... 151
    9.4.1.4.3.2 Inputs ................................................................. 153
    9.4.1.4.3.3 Processing ............................................................ 153
    9.4.1.4.3.4 Outputs ............................................................... 153
  9.4.1.4.4 Line Evaluation Report Configuration Editor Screen ........... 153
    9.4.1.4.4.1 Introduction ....................................................... 153
    9.4.1.4.4.2 Inputs .................................................................. 156
    9.4.1.4.4.3 Processing ............................................................ 156
    9.4.1.4.4.4 Outputs ............................................................... 156
  9.4.1.4.5 Line Layout Editor Screen ........................................... 156
    9.4.1.4.5.1 Introduction ....................................................... 156
    9.4.1.4.5.2 Inputs .................................................................. 159
    9.4.1.4.5.3 Processing ............................................................ 159
    9.4.1.4.5.4 Outputs ............................................................... 161
  9.4.1.4.6 Board Assignment to Lines Editor Screen ....................... 161
    9.4.1.4.6.1 Introduction ....................................................... 161
    9.4.1.4.6.2 Inputs .................................................................. 163
    9.4.1.4.6.3 Processing ............................................................ 163
    9.4.1.4.6.4 Outputs ............................................................... 163
  9.4.1.4.7 Board Family Setup Editor Screen .................................. 163
    9.4.1.4.7.1 Introduction ....................................................... 163
    9.4.1.4.7.2 Inputs .................................................................. 165
    9.4.1.4.7.3 Processing ............................................................ 165
    9.4.1.4.7.4 Outputs ............................................................... 165
  9.4.1.4.8 Board Sequencing Editor Screen ..................................... 165
    9.4.1.4.8.1 Introduction ....................................................... 165
    9.4.1.4.8.2 Inputs .................................................................. 167
    9.4.1.4.8.3 Processing ............................................................ 167
    9.4.1.4.8.4 Outputs ............................................................... 167
Table of Contents

9.4.2 External Interface Requirements......................... 167
  9.4.2.1 Software Interfaces.............................. 167
  9.4.2.2 Communications Interfaces....................... 167

9.4.3 Design Constraints........................................ 167

10.0 INTEGRATED DATA MODEL FOR PILOT APPLICATION......... 168
  10.1 Chapter Overview.......................................... 168
  10.2 Business Model (Normalized)................................ 169
      10.2.1 Entity Attribute Descriptions..................... 170

  10.3 Data Model for Prototype Application (De-normalized)..... 174
      10.3.1 Entity Attribute Descriptions (Database Schema)...... 175

11.0 PROJECT REVIEW AND NEXT STEPS............................. 179
  11.1 Chapter Overview.......................................... 179
  11.2 Strengths and Weaknesses of Project Methodology .......... 179
  11.3 Next Steps................................................ 180
FIGURES

Figure 1 Causal Loop Diagram Depicting Problem Domain and Interacting Phenomenons............................................................ 16
Figure 2 Feedback at Each production Step............................................... 32
Figure 3 Continuous Feedback During the Software Development Process................................................................. 33
Figure 4 Typical Scheduling Methodology for Software Development Projects................................................................. 35
Figure 5 Software Architecture (High Level)........................................ 59
Figure 6 Factory Planning Application Editor Modules........................ 61
Figure 7 Software Modules Developed for Pilot Application.................. 73
Figure 8 Representation of the Application Specific Tables (ASTs) Approach for the Factory Planning Database Implementation... 74
Figure 9 Representation of the Independent Database Approach for the Factory Planning Database Implementation........................................ 75
Figure 10 Main Menu Screen.......................................................... 130
Figure 11 Scenario and Report Selection Screen................................ 133
Figure 12 Line Throughput Results Screen......................................... 136
Figure 13 Product Related Results Screen.......................................... 139
Figure 14 Machine Related Results Screen........................................ 142
Figure 15 WIP Related Results Screen.............................................. 145
Figure 16 Scenario Selection Screen.................................................. 148
Figure 17 Scenario Data Editor Screen................................................ 150
Figure 18 Report Selection Screen...................................................... 152
Figure 19 Line Evaluation Report Configuration Editor Screen.............. 155
Figure 20 Line Layout Editor Screen................................................... 158
Figure 21 Board Assignment to Lines Editor Screen........................... 162
Figure 22 Board Family Setup Editor Screen...................................... 164
Figure 23 Board Sequencing Editor Screen......................................... 166
Figure 24 Business Model (Normalized) ............................................. 169
Figure 25 Data Model for Prototype Application (De-normalized)............ 174

TABLES

Table 1 Evaluation of (Weighted) Key Characteristics Associated With Pilot Customer Site......................................................................... 31
Table 2 Comparison of the two Approaches for the Factory Planning Data Base....................................................................................... 76
Table 3 Entity Attribute Descriptions (Business Model).......................... 170
Table 4 Entity Attribute Descriptions (Database Schema).......................... 175

END NOTES........................................................................................................... 181

BIBLIOGRAPHY.................................................................................................. 182
1.0 INTRODUCTION

This thesis adheres to the Leaders For Manufacturing (LFM) Program thesis guidelines and fulfills the author's thesis requirements pertaining to the following two degrees:

- Master of Science in Management (Manufacturing Track)
- Master of Science in Civil and Environmental Engineering (Intelligent Engineering Systems Laboratory)

This thesis is written on the basis of the author's six and a half month LFM internship at Motorola Manufacturing Systems located in Schaumburg, IL.

1.1 Problem Description

As Motorola continues to try and play catch up to rapidly increasing consumer demand, it resorts to both short term solutions -- such as purchasing faster automated chip placement equipment and replacing selective manual manufacturing operations with automation -- and long term solutions such as constructing additional factories. Each of these approaches creates a dynamically evolving manufacturing environment wherein manufacturing operations are continuously subject to product and process changes. As a result, line and factory managers are confronted with the complex task of optimizing line throughput, minimizing costs and ensuring quality products in an environment which is constantly subject to change.

One of the most useful methodologies that is used to conduct quantitative analyses to determine optimal line layouts, process configurations, etc. is that of stochastic simulation. It is considered to be highly reliable in predicting expected line throughput, identification of the bottleneck work center, build up of WIP on the line and additional important parameters because the uncertainties that exist on the manufacturing floor (such as machine breakdowns, variations in time to fix the machines, variations in machine processing cycle times, etc.) can be modeled using random variables and distribution functions which map the expected behavior of such uncertainties.

However, although stochastic simulation in the above context is known to be a powerful tool to conduct analyses of changing manufacturing environments, its use is often limited because of the inherent complexity in building a simulation model to represent a given
manufacturing environment. Such models can take from between a few weeks to a few months to construct and evaluate. Moreover, the problem is further compounded by the fact that each of the commercially available stochastic simulation applications is fairly intricate to use and a substantial amount of learning is required before these applications can be used in a meaningful manner. Hence, stochastic simulation tools are probably not being used as widely and as frequently as they should be in order to address the needs of managers who have to make critical decisions in rapidly changing manufacturing environments.

The problem is explicitly described by the causal loop diagram (adopted from the discipline of Systems Dynamics) depicted in Figure 1 on the following page.
Demand Rate Increase

Marketing Pressure

Rate of Optimal Integration of New Equipment

Need to Quickly Conduct "What-if" Analyses

Use of Stochastic Simulation-based tools

Need to use Stochastic Simulation-based tools

Demand for Simulation Experts

Time Required to Construct Simulation Models

Structure Representing Problem Domain

Chances of Change in the Fundamental Problem

Need to Conduct "What-if" Analyses (New Factories)

Frequency of Use of Stochastic Simulation

Popularity of Stochastic Simulation-based Analysis

Rate of Long Term Capacity Expansion

Figure 1 Causal Loop Diagram Depicting Problem Domain and Interacting Phenomenons
The variables that are used to describe the problem are defined as follows:

<table>
<thead>
<tr>
<th>Variable Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Demand Rate Increase</td>
<td>Rate of increase in market demand</td>
</tr>
<tr>
<td>Marketing Pressure</td>
<td>Marketing pressures on factories to meet customer orders</td>
</tr>
<tr>
<td>Rate of Purchase of New Machines</td>
<td>Rate of purchase of newer and faster manufacturing equipment (at existing factories)</td>
</tr>
<tr>
<td>Need to Quickly Conduct &quot;What-if&quot; Analyses</td>
<td>Need to quickly construct different scenarios to determine how to optimally integrate new equipment into existing manufacturing environments</td>
</tr>
<tr>
<td>Need to use Stochastic Simulation-based Tools</td>
<td>Need to use stochastic simulation-based tools to conduct analyses described above</td>
</tr>
<tr>
<td>Use of Stochastic Simulation-based Tools</td>
<td>Use of stochastic simulation-based tools to conduct analyses described above</td>
</tr>
<tr>
<td>Rate of Optimal Integration of New Equipment</td>
<td>Rate of optimal integration of new equipment into existing manufacturing environments</td>
</tr>
<tr>
<td>Demand for Simulation Experts</td>
<td>Need (demand) for experts in stochastic simulation</td>
</tr>
<tr>
<td>Time Required to Construct Simulation Models</td>
<td>Time required to construct elaborate, sophisticated and representative stochastic simulation models</td>
</tr>
<tr>
<td>Remaining Time on Hand</td>
<td>Time remaining to address other manufacturing/organizational needs</td>
</tr>
<tr>
<td>Chances of Change in the Fundamental Problem</td>
<td>Chances of the nature of the problem changing by the time the solution is generated (this is conceivable especially if formulating the solution takes a few months)</td>
</tr>
<tr>
<td>Popularity of Stochastic Simulation-based Analysis</td>
<td>Popularity of stochastic simulation as an tool (at a given factory)</td>
</tr>
<tr>
<td>Frequency of use of Stochastic Simulation</td>
<td>Frequency of use of stochastic simulation as an analysis tool (at a given factory)</td>
</tr>
<tr>
<td>Rate of Long Term Capacity Expansion</td>
<td>Rate of construction of new factories</td>
</tr>
</tbody>
</table>
Figure 1 Explanation
The arcs in Figure 1 indicate the links between the different variables described above. The polarity of each link is indicated near the arrowhead for each arc. The polarity depicts the relationship between the direction of change of the two variables connected by the corresponding arc. For example, when marketing pressure increases, the rate of purchase of new machines increases whereas when the rate of optimal integration of new equipment increases, the marketing pressure decreases. (The "+" indicates that a change in the preceding variable in a given direction causes a change in the following variable in the same direction. The "-" indicates that a change in the preceding variable in a given direction causes a change in the following variable in the opposite direction).

There are five feedback loops depicted in the diagram -- each of which indicates a cause and effect relationship that arises as a result of a number of interacting variables. There are four Balancing Loops and one Reinforcing Loop. A Balancing Loop depicts behavior wherein the net result of traversing the loop is to weaken or lessen the stimulus that is depicted at the start of the loop. For instance, an increase in marketing pressure impacts a number of other variables in the top most loop structure which eventually cause the rate of optimal integration of new equipment to increase which causes the marketing pressure to decrease. A Reinforcing Loop depicts behavior wherein the net result of traversing the loop is to strengthen or increase the stimulus that is depicted at the start of the loop. For instance, an increase in the need to quickly conduct "what-if" analyses causes an increase in the need to use stochastic simulation-based tools which in turn causes an increase in demand for simulation experts. This causes an increase in the time required to construct simulation models which causes a decrease in the remaining time on hand which in turn further aggravates the need to quickly conduct "what-if" analyses.

One of the most important aspects of Figure 1 is that of the different delays that are depicted along certain arcs. Each "DELAY" label indicates that there is a significant relative delay between the cause and effect relationship of the two connecting variables (relative compared to the other arcs in the diagram). Furthermore, in one instance, there are actually two "DELAY" labels depicted side by side to indicate that this delay is much longer than the other delays depicted in the diagram. As described below, it is important
to note the existence of these delays in order to understand the overall impact they have in the system depicted in Figure 1.

The diagram depicts that there is at least one delay in each of the Balancing Loops whereas there is no delay in the Reinforcing Loop. This is important because it indicates that the cause and effect relationships of the Reinforcing Loop will be felt much sooner than those of the Balancing Loops. When we examine the cause and effect relationships of the Reinforcing and Balancing Loops, we realize that this means that in the short run, an increase in marketing pressure will (subsequently) cause an increase in the need to quickly conduct "what-if" analyses which (since it is at the start of the Reinforcing Loop) will eventually generate a snow-ball effect and further aggravate the need to conduct "what-if" analyses. In other words, the simulation experts can be expected to be "resourced-out" in the short run. However, as the effects of the balancing loops play out over the long run, the frequency of use of stochastic simulation analyses at individual factory sites could be expected to wane -- not because there is no need for such analyses, but because there is not enough time to construct elaborate models required to conduct such analyses. (This implies that the amount of repeat business of the simulation experts would be expected to decrease in the long run). The most important outcome implied by Figure 1, however, is that in the long run, managers would be deprived of making as informed decisions as they might make on the basis of the use of stochastic simulation models.

1.2 Thesis Goal

This thesis project will significantly reduce the lengthy cycle time of constructing stochastic simulation models to evaluate surface mount manufacturing environments and will thereby increase the frequency, in the long run, of utilizing stochastic simulation-based analyses to evaluate different manufacturing environments.

The aim of this thesis is to develop a software application that will enable managers to seamlessly configure, execute and evaluate sophisticated, representative and elaborate stochastic simulation models (in Witness) to analyze most surface mount manufacturing environments. Managers will be free to evaluate different manufacturing environments against any set of metrics chosen from those listed below. In addition, this software application will serve as the core of a larger Factory Planning software application wherein the user can analyze a given manufacturing environment according to cost and
quality metrics in addition to the metrics listed below. (Additional tools will have to be integrated into this software application for this purpose and this thesis addresses the primary architectural issues that will facilitate this integration).

Metrics Against Which a Given Manufacturing Environment Can be Evaluated

* WHAT IS THE EXPECTED LINE THROUGHPUT?

- Printed Circuit Board statistics for each product
  -- Number entering production
  -- Number shipped
  -- Number scrapped
  -- Number assembled
  -- Number rejected
  -- Average Cycle time (entire production process)

- Device Statistics
  -- Bottleneck machine identification
  -- For each machine
    -- % Busy
    -- % Idle
    -- % Stopped
      * Downstream blocking
      * Setup
      * Machine Error
      * Repair
  -- Number of Operations completed

- WIP Statistics
  -- Average WIP on the line for each type of product (printed circuit board)
  -- WIP level between front end and test
  -- Maximum WIP for a given scenario
  -- Minimum WIP for a given scenario
  -- Which product in the scenario corresponds to the maximum WIP?
  -- Which product in the scenario corresponds to the minimum WIP?

1.3 Thesis Layout -- "Top-Down" Approach

The core of this thesis (chapters five through nine) follows a "top-down" approach. The reader is introduced to the various functions of the Factory Planning Application (at a high level) in Chapter Five. (These cover all the proposed functions, over and above those that will be implemented for the prototype application). Thereafter, in Chapter Six, the reader is presented with an overview of the various software modules that will provide these functionalities. (These include all the proposed software modules, over and above those that will be developed for the prototype application).
It is expected that after reading Chapter Five and Chapter Six, the reader will have a holistic understanding of the different functions and corresponding software modules pertaining to the Factory Planning Application. Subsequently, chapters Seven through Nine journey the reader through the detailed requirements for the software modules developed for the Prototype application.
2.0 ACKNOWLEDGMENTS

The author would like to offer special thanks to the following people whose continual support and guidance proved invaluable in this project.

* Peggy Eastwood
* Alex Lach
* Bob Rush

In addition, the author wishes to thank Dave Liljegren of Motorola Manufacturing Systems whose expertise in the field of stochastic simulation was paramount in making this project succeed.

Furthermore, the author wishes to explicitly thank the following members of the Factory Planning software development team, without whose help this thesis would not have been possible.

* Sharath Kalvakota
* Roger Larson
* Mike Sorkis
* Karen Vilt

Next, the author is extremely grateful to all the cross-functional team members at the pilot factory site (Motorola Wireless Data Group). These include:

* Laurinda Bellinger
* Chuck Coughlin
* Pete Hassler
* Deishin Lee
* Thilo Semmelbauer

In addition, the author is grateful to all the staff members at Motorola Manufacturing Systems and Motorola Wireless Data Group who assisted in this effort at different points in time.

Next, the author would like to acknowledge MIT's Leaders For Manufacturing Program for being instrumental in making this thesis project come to fruition.

Finally, the author would like to acknowledge all individuals who have contributed in this effort and have not been listed above.
TO ROSINA AND SIDDHARTH FOR THE WARMTH AND COMFORT YOU HAVE GIVEN ME DURING THIS PROJECT
Chapter 3

3.0 THESIS METHODOLOGY

3.1 Chapter Overview

This chapter describes the key elements of the processes used to complete this thesis project. Section 3.2 highlights the means by which the voice of the customer was gathered. Section 3.3 outlines some of the findings from an effort to conduct a literature search/industry benchmark pertaining to the functionalities envisioned in the Factory Planning Application. The next section describes the method by which the pilot factory site was selected. Finally, Section 3.5 describes some of the key iterative efforts that were carried out throughout the project and were instrumental in ensuring the on-time completion of the project.

3.2 Gathering the Voice of the Customer

In order to gather the voice of the customer in as efficient a manner as possible, the following steps were undertaken:

- A project initialization meeting was held between key members of the project team, some of their colleagues who had worked with them on significant projects in the past, a manager who was identified as one of the key "Lead Users" for the project and the project manager. During this meeting, all the information known about the project at that time was discussed and a very rough outline of the next few steps was determined.

- The above meeting was followed by a series of "brain storming" sessions by the core members of the project team wherein the different types of primary scenarios that occur on the shop floor were envisioned, the needs arising from these scenarios were outlined and the specific functions that the Factory Planning software would have to include to address these needs were identified.

- Information generated as a result of the above steps was refined and condensed into a three page document which provided:
  -- A brief description of the proposed Factory Planning software.
  -- A list of the problems encountered on the shop floor that the Factory Planning software was intended to address.
  -- A brief description of the primary functions that would be included in the Factory Planning application.
  -- A list of metrics that would be used to format the output of the Factory Planning software.

This document was then discussed/circulated (with a cover letter outlining the overall project) amongst seven "Lead Users" at Motorola factories located in different parts of the United States.
Feedback from the aforementioned "Lead Users" was used to rank the proposed functions of the Factory Planning software in order of importance. This feedback also served as a "sanity check" to ensure that the project team was on the right track. The team used the information generated by the steps described above to make a presentation to the potential pilot factory site and highlight the benefits of the Factory Planning software. On the basis of this presentation, the potential pilot factory site subsequently volunteered to function as the pilot site. The software requirements for the Factory Planning Application were then iteratively refined (see Section 3.5.1) by the cross functional team comprising of the project development team, users and other staff from the pilot factory site.

3.3 Literature Search/Industry Benchmark

Prior to commencing the development of the software requirements for the Factory Planning software, an informal literature search was conducted to determine the scope of the work that has been done (if any) in the realm of Factory Planning as defined by the functionalities desired by our "Lead Users". For each of the papers that were found to be relevant to some degree, a brief synopsis is provided in the following text.

3.3.1 Research Work at Universities/Corporations

Optimization of SMT Systems by Computer-Aided Planning, Simulation and Monitoring

This paper describes some of the work done at the University of Erlangen-Nuremberg where an entire SMT line is built in a model factory to further the purposes of research and education. The portion of the paper of relevance to this thesis pertains to the use of computer aided simulation techniques to optimize the throughput of the SMT line and to also determine buffer sizes, idle times of machines, occupied feeders and other parameters of the SMT line. However, there are no details pertaining to the nature of the simulation models generated, their ease of use, degree of complexity, accuracy, etc.

Factory Control Integrated With Operational Planning

This paper is part of a research project conducted at the Asea Brown Boveri Corporate Research Laboratory in Baden, Switzerland and describes the integration of a factory control system with a decision support system to facilitate operational planning.

The relevant portion of this paper describes the use of a graphical and interactive simulation system which is used as a decision support tool. The simulator enables the user to simulate a given factory and monitor important line performance parameters such
as machine statistics (% of time the machine is busy or idle), queue lengths in front of every machine, etc. However, the relevance of this application to surface mount manufacturing operations cannot be determined from the paper. Furthermore, the ease of use of the simulator as a decision support tool is unclear. (One of the goals of this thesis is to seamlessly integrate a simulator into the Factory Planning environment so that managers/engineers can benefit from the use of a simulator without having to climb the learning curve associated with the intricacies of the simulator.

**SEMATECH's Semiconductor Workbench for Integrated Modeling (SWIM)**

SWIM is a modeling environment currently being developed by the Semiconductor Manufacturing Technology (SEMATECH) consortium of companies. It will enable users to construct and evaluate diverse semiconductor manufacturing models from the same platform using a common user interface. Users will be able to construct the following types of models (developed at SEMATECH) via the SWIM environment:

- Cost/Resource Model
- Cost of Ownership Model

The SWIM environment also incorporates ManSim (a commercially available factory simulation package) and STATA (a commercially available statistics package) which will be used from within the SWIM environment to conduct analyses at the fab, cell or equipment levels. These and other applications will be automatically executed from within the SWIM environment wherein the required data will be seamlessly transferred between applications. As a result, it is hoped, the non-productive engineering activity of entering the same data in different models will be eliminated and the total time required to construct and evaluate elaborate models for semiconductor manufacturing will be significantly reduced.

Owing to the fact that there is little commonality between the semiconductor fabrication process and the printed circuit board assembly process from a manufacturing standpoint, the SWIM tools cannot be directly incorporated into the Factory Planning environment. However, some of the core philosophies behind the SWIM environment have been incorporated into the Factory Planning environment and in this context, a perusal of SEMATECH's efforts regarding the development of the SWIM environment has proven to be valuable.
3.3.2 Commercially Available Software Applications

Our benchmarking efforts also enabled us to determine if there were any commercially available software applications that could be used to serve the needs of this thesis project. Furthermore, this search also enabled the author and the project team to generate an approximate assessment of the degree of sophistication of the relevant commercially available software applications and benchmark the functionalities envisioned to be a part of the Factory Planning Application with those of commercially available software applications in the industry.

Although the literature search did not reveal any existing software applications that suited our needs, a few of the applications had some degree of overlap with the functionalities identified by our "Lead Users" and hence deserve mention here -- they are briefly described below in alphabetical order. (In addition, some of the philosophies behind the Factory Planning Application and SEMATECH's efforts to develop the SWIM platform are very closely linked and deserve special mention. These are described in Section 3.3.3 below).

Achilles
Manufacturer Information: IMT Systems, Inc.
444 Castro Street, Suite 711
Mountain View, CA 94041

This tool provides "what-if" analyses capabilities and control capabilities for semiconductor wafer fabrication. It can be used for mid-shift problem solving and simulates in detail, the processes, machines and wafer lots of a specific factory environment. The output includes throughput reports by process step and workstation, equipment statistics such as utilization rates, queue lengths, cycle times, bottleneck identification, etc. This tool can also be initialized by integrating it to an existing CAM system to ensure accurate model initialization and calibration. However, from the depth of the information found on this tool, it appears that it is geared specifically towards semiconductor fabrication operations as opposed to surface mount assembly operations for printed circuit board assembly.
CIMBridge
Manufacturer Information: Mitron Corporation
15256 N.W. Greenbrier Parkway
Beaverton, Oregon 97006

CIMBridge is an integrated printed circuit board design and manufacturing tool which links board design data with manufacturing systems. The tool accepts data from diverse CAD systems, converts the data to a neutral format and thereafter generates machine placement NC programs for various machines on the assembly line. The tool has limited functionality pertaining to estimating the cycle time impacts (on a given station/machine) of changing printed circuit board layouts.

However, the tool does not have the capability to enable the user to conduct sophisticated "what-if" analyses in terms of allocating different products to different lines, changing the line layouts, process flows, product lot sizes, etc. Hence, it does not adequately serve the needs of Factory Planning as described in this thesis. Furthermore, there is a significant amount of learning that has to be accomplished prior to using this tool effectively, and this contradicts the ease of use requirement of the Factory Planning Application.

Paragon
Manufacturer Information: Tecnomatix Technologies Inc.
39830 Grand River Avenue
Suite B-3, Novi, MI 48375

This tool provides Computer Aided Process Engineering capabilities to assist decision making pertaining to the printed circuit board assembly process. Although this tool provides some key functions that aid the design and implementation of the printed circuit board assembly process (such as graphically viewing the board layout, graphically defining the line layout that will be used to assemble the board, automatically creating some of the machine NC programs to place components on the board, etc.), these functions are either already provided by software previously developed in-house at Motorola or are envisioned to be included in some of the applications currently under development in the company.

Most importantly, the level of sophistication of the line operation simulations that can be generated via Paragon are not considered to be nearly as advanced as those that can be
generated via the Factory Planning prototype application. Furthermore, there is a significant learning curve associated with using the Paragon computer aided process planning tools and this defeats one of the core purposes of the Factory Planning Application -- to provide managers with the capability of quickly configuring and evaluating elaborate "what-if" manufacturing scenarios by transparently driving sophisticated stochastic simulation models.

**Simfactory 2.5/Simprocess**
Manufacturer Information: CACI products Co.
(Subsidiary of CACI International, Inc.)
3333 N. Torrey Pines Court
La Jolla, CA 92037

This software application is used to conduct simulation-based analyses for Factory Planning and business process re-engineering. It addresses issues pertaining to layout design, capacity planning, rework and repair analyses, material handling and maintenance planning. It provides graphical outputs with critical parameters such as equipment utilization, queue levels, transporter utilization, raw material consumption and throughput. It is not clear from the depth of literature search conducted whether or not this software application would be ideally suitable to simulate surface mount manufacturing operations. Furthermore, the defacto software tool used for constructing stochastic simulation models at Motorola Manufacturing Systems (the author's internship site) is Witness and hence by default, Witness would be used as the tool in which simulation models are seamlessly configured by the Factory Planning software application.

### 3.3.3 SEMATECH Efforts Relevant to Thesis Topic

SEMATECH has been developing an integrated software environment for evaluating and optimizing Semiconductor manufacturing processes. This environment -- also referred to above -- is called the Semiconductor Workbench for Integrated Modeling (SWIM) and some of the philosophies behind the SWIM architecture are similar to those of the Factory Planning Application.

Most notably, there are two significant commonalities behind the approaches adopted in the SWIM and Factory Planning architectures. Firstly, both architectures allow for third
party software tools to be exchanged for specific software modules. The Factory Planning architecture accommodates this by:

- A highly modular structure
- Requiring that global variables are not use across different software modules
- Stipulating the specific data flows to and from each software module -- a module from the Factory Planning Application can thereby be swapped with another software module which provides the same functionality by ensuring that the data flows before and after the swap remain unchanged.

The SWIM architecture allows for such an exchange by including a detailed, integrated data model which describes the relationships between the various entities/attributes associated with the different software modules within the SWIM environment. Any module within the environment can be interchanged with a module external to the environment so long as the relationships between the various entities/attributes that are affected by the exchange are preserved.

Secondly, both the SWIM application and the Factory Planning Application support data source independence in that they can be ported from one factory to another with relative ease, in spite of the fact that the data repositories at the two factories might be dramatically different. This is accomplished in each case by tying the different software applications to an auxiliary database which in turn is linked to the local data repository at a given factory by means of what the author refers to as a data transfer interface. Hence, when the SWIM/Factory Planning Applications are ported from one factory to another, only the data transfer interface has to be changed to match the local data repository at a given factory -- the individual software modules (that make up the SWIM/Factory Planning Applications) remain unchanged.

3.4 Determination of Pilot Customer Site

As described above, members of the project team had contacted "Lead Users" to develop an initial assessment of the voice of the customer for the Factory Planning Application. These users were located at different Motorola factories across the United States and our project team evaluated each of these factories as a potential pilot customer site according to the key characteristics described below. Each of these characteristics was assigned a weight according to the degree of importance the project team associated with that characteristic and the sum of the weights was made to equal 100. (The seven sites
considered are listed as S1 through S7 in order to hide their real identity). According to this ranking system, Motorola's Wireless Data Group (WDG) was selected as the potential pilot customer site. (This was only confirmed after the project team made a presentation to WDG regarding the capabilities envisioned for the Factory Planning Application and obtained buy-in from WDG as a result).

### Key Characteristics Related to Pilot Customer Site

<table>
<thead>
<tr>
<th></th>
<th>Weight</th>
<th>S1</th>
<th>S2</th>
<th>S3</th>
<th>S4</th>
<th>S5</th>
<th>S6</th>
<th>S7</th>
</tr>
</thead>
<tbody>
<tr>
<td>Proximity</td>
<td>25</td>
<td>8</td>
<td>9</td>
<td>9</td>
<td>4</td>
<td>4</td>
<td>10</td>
<td>4</td>
</tr>
<tr>
<td>Availability of Leading Edge Engineers (Feedback / Testing)</td>
<td>20</td>
<td>9</td>
<td>7</td>
<td>8</td>
<td>9</td>
<td>7</td>
<td>10</td>
<td>8</td>
</tr>
<tr>
<td>Availability of Expertise is Front End, Back End, and Test</td>
<td>5</td>
<td>7</td>
<td>7</td>
<td>9</td>
<td>8</td>
<td>7</td>
<td>8</td>
<td>8</td>
</tr>
<tr>
<td>Support of Engineering Management</td>
<td>23</td>
<td>10</td>
<td>8</td>
<td>7</td>
<td>7</td>
<td>4</td>
<td>6</td>
<td>7</td>
</tr>
<tr>
<td>EI environment (DB, Lack of Tool Restrictions)</td>
<td>2</td>
<td>7</td>
<td>7</td>
<td>7</td>
<td>5</td>
<td>6</td>
<td>8</td>
<td>6</td>
</tr>
<tr>
<td>Importance to Motorola</td>
<td>5</td>
<td>10</td>
<td>9</td>
<td>9</td>
<td>5</td>
<td>10</td>
<td>3</td>
<td>6</td>
</tr>
<tr>
<td>Fit with Existing Tools</td>
<td>5</td>
<td>9</td>
<td>10</td>
<td>10</td>
<td>9</td>
<td>2</td>
<td>7</td>
<td>4</td>
</tr>
<tr>
<td>Low Overhead</td>
<td>10</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>9</td>
<td>6</td>
<td>9</td>
<td>8</td>
</tr>
<tr>
<td>Fit of Needs with Tool Vision</td>
<td>5</td>
<td>8</td>
<td>6</td>
<td>9</td>
<td>9</td>
<td>8</td>
<td>8</td>
<td>9</td>
</tr>
<tr>
<td>TOTAL</td>
<td>100</td>
<td>8</td>
<td>7.3</td>
<td>7.6</td>
<td>7</td>
<td>5.4</td>
<td>8.2</td>
<td>6.5</td>
</tr>
</tbody>
</table>

Table 1 Evaluation of (Weighted) Key Characteristics Associated With Pilot Customer Site

### 3.5 Overview of Main Processes

The following sub-sections document some of the key processes that were followed during the course of this thesis project. The processes were followed keeping in mind the business aspects of this effort, the thesis requirements of the project and the project management practices followed at Motorola Manufacturing Systems at the time of undertaking this thesis project.

#### 3.5.1 Periodic Meetings -- Continuous Feedback From Customers

Throughout the project life cycle (after the pilot customer site had been confirmed), periodic meetings were held which comprised of cross-functional members from the project development and customer site teams. The composition of the cross-functional groups that met during each of these meetings was determined prior to each meeting according to the meeting agenda. For instance, if database integration issues needed to be addressed, members from the customer site's CIM group attended the meetings. On the other hand, if the design of the user interface was to be reviewed, potential users from the...
pilot customer site were invited to the meeting. The frequency of these meetings was
determined according to the constantly evolving project schedule (discussed below). On
average, these meetings were held on a weekly basis.

The goal of the periodic meetings was to ensure that the software development team was
on the right track at all times. Owing to the extremely tight project schedule (discussed
below), it was imperative to obtain frequent feedback from the customer site and to take
corrective action accordingly. This process is akin to that of inspection at every stage of
an in-line manufacturing process wherein the worker at each station ensures that his/her
output caters to the needs of his/her most immediate downstream customer. (This
process is depicted in Figure 2 below).

![Diagram of Feedback at Each Production Step](image)

**Figure 2** Feedback at Each Production Step

In the case of the software development process used in this project, the only difference is
that the stages of software development (analogous to the different manufacturing stages)
are primarily separated by time instead of by physical space as is the case with
manufacturing processes on the shop floor. Hence, the phenomenon of continuous
feedback employed in the software development process for this project can be depicted
as shown in Figure 3 below.
3.5.2 Weekly Team Meetings -- "Yo One"

Weekly team meetings were held by the software development team to discuss and address any issues internal to the team. Although we did not follow a formal "Yo-One" ceremony, each of these meetings served the purpose of the team coming to agreement on important issues pertaining to the project development effort. In some cases, the team agreed upon the actions taken for the completion of certain tasks and agreed to move on. In other cases, the team agreed on what approaches it should adopt to tackle particular problems.

3.5.3 Dissemination of Project Information

In order to ensure that project information was disseminated in a timely and comprehensive manner, two policies were established.

Firstly, the project team submitted a single weekly report to the project manager, senior management, customer site and all other personnel who needed to be kept up to date with the project's progress. These reports highlighted major accomplishments for the given week, major efforts under way at the time and significant problems that the project team was encountering. The reports also documented the estimated completion dates for milestones coming up in the following week.
Chapter 3

Secondly, all significant documents (such as software architecture and requirements documents, activity logs, major issues/to dos, etc.) were maintained on a file server which could be accessed by all members of the project team who needed to have access to such information.

3.5.4 Project Schedule Methodology

Since the author's internship had a specific end date, this project did not enjoy the degree of flexibility that is inherent to most thesis projects. Furthermore, since the author felt very strongly that to ensure that the project benefits were fully realized it was imperative to complete and implement the prototype application prior to the end of his internship, the project was essentially scheduled backwards from the completion date.

In the case of most software projects, the completion dates for individual tasks and sub-tasks are constantly revised (and often pushed further out in time) with the progression of the project, thereby resulting in delays in the target project completion date. Figure 4 is a simple schematic depicting this phenomenon. As the figure shows, a single initial delay in a task which is on the project's critical path maps onto the other critical paths of the project and causes the original completion date to be pushed out further in time. Of course, in addition to the delays depicted in the schematic, there will be other delays owing to the fact that the critical path often changes as individual tasks and sub-tasks get re-scheduled, thereby exacerbating the overall project delay even further.
However, in this project, the due date was considered fixed and the completion times for the individual tasks and sub tasks were constantly re-evaluated in order to ensure that the due date did not get postponed. As a result, the prototype Factory Planning Application was implemented on time in spite of repeated concerns raised by team members that it simply could not be done according to the proposed schedule. (This scheduling technique is often used by aggressive software and systems integration consulting firms. These firms nurture this capability into a core competence upon which they compete with larger firms in their industry).

Figure 4 Typical Scheduling Methodology for Software Development Projects
4.0 THESIS RESULTS

4.1 Chapter Overview

This chapter outlines the results of the author's thesis project. It describes the four scenarios that were used to transparently configure, execute and evaluate a stochastic simulation model (using Witness). Furthermore, it highlights the benefits of the Factory Planning (Prototype) Application in terms of:

- The transformation of stochastic simulation-based modeling into an enabling technology which managers can transparently use for manufacturing scenario planning to make more accurate and informed decisions.

- The estimated cycle time savings of transparently configuring and evaluating intricate Witness models.

- The development of a software platform into which additional software applications can be integrated in the future in order to evaluate surface mount lines according to cost and quality metrics in addition to throughput metrics.

4.2 Manufacturing Scenario Planning

The Factory Planning software can completely configure (seamlessly to the user) most printed circuit board assembly environments (front end and back end). It seamlessly models:

- A single line

- An entire factory consisting of one or more independent lines

- An entire factory consisting of many front ends which feed into one or more common back ends

- The manufacturing processes for single/double sided boards as well as panelized boards.

- The manufacturing processes where two or more boards assembled on the line are eventually assembled into one product prior to shipping.

- The manufacturing processes where one side of a given board is produced on one line and the other side is produced on the same or a different line.

In each of the above cases, the modeling is done seamlessly to the user -- the user does not have to run the software in any different way when modeling one situation versus
(The Factory Planning Application constructs simulation models based on the process routes for each product as stipulated by the user). For each of the sets of above mentioned manufacturing situations, the Factory Planning software allows the user to select combinations of metrics from a pre-determined list of metrics (described in Section 1.2 above) to evaluate the different manufacturing situations. For the selected metrics, the Factory Planning software displays comparative graphs so that the user can easily evaluate the impacts of each of the different situations modeled.

The only information required as input from the user is:

- The line configuration for a given line/factory
- The product mix and production sequence for the products
- The process routes for each of the products and the corresponding cycle time at each station in the process route.
- The setup times for chip placement equipment on the line.
- The mean time between failures and mean time of assists for machines that the user wants to model with breakdowns.
- The yield % for each test/inspection station on the line (by product)

For the prototype application, the input data is entered into the Factory Planning data base via SQL scripts. The next phase of software development will include the completion of the user interface to allow for this data to be input directly from the interface via a "point and click" environment.

4.2.1 Four Scenarios Seamlessly Configured and Evaluated

For the prototype Factory Planning Application, the following scenarios (which are explained in the following text) were evaluated by transparently configuring, executing and evaluating Witness models:

- Current State
- SWAP 1
- SWAP 2
- SWAP 3

In each scenario, the number of minutes available in a shift, the number of shifts per day and the number of working days per week were modeled according to the existing norms.
in the factory. In addition, the following parameters were modeled for various stations on the line:

- Setup time
- Mean time to failure
- Mean time of assist
- Yield percentage and rework device ID (for test stations)
- Capacity (for buffers)

**Current State**

This scenario depicted the actual line configurations, process routes and lot sizes that were being used in the factory at the time of this thesis project. This consisted of dedicating each of the two lines in the factory to one product: One of the products (Product A) had an RF and a Logic board whereas the other product (Product B) had a single two-sided board (having a Top and Bottom side). Both the RF and Logic boards of Product A were assembled on Line A and both the Top and Bottom sides of product B were assembled on Line B.

The manufacturing process for product A spanned thirty-three stations (such as solder paste application, automated chip placement, manual chip insertion, inspection, etc.), each of which was transparently modeled by the Factory Planning Prototype application. The manufacturing process for Product B spanned twenty-three stations (similar to those listed above) which were also transparently modeled by the Factory Planning Application.

**SWAP 1**

In this scenario, the manufacturing lines were modeled such that the RF board of Product A and the Top side of Product B were processed on Line A whereas the Logic board of product A and the Bottom side of Product B were processed on Line B. This scenario was compared to the Current State according to the metrics listed in Section 1.2 above.

**SWAP 2**

In this scenario, the manufacturing lines were modeled such that the Logic board of Product A and the Top side of Product B were processed on Line A whereas the RF board of product A and the Bottom side of Product B were processed on Line B. This scenario was compared to the Current State according to the metrics listed in Section 1.2 above.
Chapter 4

**SWAP 3**

In this scenario, the manufacturing lines were modeled such that the Logic board of Product A, the Top side of Product B and the Bottom side of Product B were processed on Line A whereas Line B was dedicated to manufacturing only the RF board of Product A. This scenario was compared to the Current State according to the metrics listed in Section 1.2 above.

### 4.2.2 Cycle Time Reduction

Each of the four different scenarios listed above required approximately two hours to configure, execute and evaluate using the Factory Planning Prototype Application. For the same four scenarios to be modeled and evaluated in Witness (without the use of the Factory Planning tool), a simulation expert would require approximately one week\(^6\). A person proficient in Witness but not an expert would take approximately two months\(^7\). In other words, the use of the Factory Planning (Prototype) Application constitutes a cycle time savings between 5X and 40X.

Most of the eight hours that were required to use the Factory Planning Application (for the above four scenarios) were used to enter manufacturing related data into the data base by means of SQL scripts. After the user interface is completed, as mentioned above, this data will be easily entered by the user via a "point and click" environment and hence the time required to evaluate different manufacturing scenarios via the Factory Planning Application can be expected to be even less than it is today.

### 4.3 Platform for Future Factory Planning Software Tools

The software architecture for the Factory Planning Prototype application allows for the future integration of additional software tools which will be used to estimate the performance of one or manufacturing lines according to cost and quality metrics, in addition to the throughput metrics listed in Section 1.2. These tools will either be developed in-house at Motorola -- some tools have already been developed by different groups within the company and these will be evaluated as potential candidates for future integration into the Factory Planning environment -- or they might be procured from outside vendors. The idea is to provide the user with a single platform in the long run upon which the user can configure "what-if" scenarios and estimate the performance of the line in terms of throughput, cost and quality metrics.
5.0 SOFTWARE REQUIREMENTS (HIGH LEVEL)

5.1 Chapter Overview

This chapter provides the reader with a high level description of the various functionalities that are to be provided by the Factory Planning Application. The different sections of this chapter are written in a format prescribed by the "Software Engineering Standards" -- a document compiled by the Institute of Electrical and Electronics Engineers (October 1989).

5.2 Introduction

The Factory Planning application is intended to significantly enhance the decision making process when the production environment is subject to product and process changes. For example, when a factory updates a chip shooter from a Fuji CP III to Panasonic MSH-2, or when a new line is added to a factory, some of the critical decisions that need to be made include:

- The new line configuration.
- The allocation of products and components to the new machine(s).
- The batch sizes in which products should be made.
- The setup policies for each machine and product -- common vs dedicated setups.
- The buffer size between different segments of the line (example front end and test).

These and other decisions will be supported by the functions of the software tool. It will also evaluate the line in terms of throughput and WIP and will feed "low-level" component placement, machine setup and cycle time data to its different modules in order to address issues such as those listed above.

5.2.1 Purpose

This chapter provides an overview of the various functionalities of the Factory Planning Application. It provides the software development team with a starting point upon which to develop the detailed software requirements for the individual modules of the Factory Planning Application.
Chapter 5

5.2.2 Scope

The Factory Planning Application will include the following software functions:

- Line Layout Function
- Line Evaluation
- Board Assignment to Lines
- Board Family Generation
- Board Sequencing
- Front End Throughput Optimization
- Scenario and Data Editing

These modules will collectively enable the user to accomplish the following tasks:

- Determine which products should be assigned to which lines for throughput optimization.

- For each line:
  -- Determine which products should be grouped into a common family.
  -- Determine the sequence in which products should be built to minimize total setup time.

- Iterate through different scenarios to determine:
  -- What batch sizes to use.
  -- Setup policies by machine and product (common vs dedicated setups).
  -- Buffer sizes between any two stations (and between front end and test).

- Generate NC programs and setup files for the front-end of the line to ensure that the front end is balanced and optimized for throughput maximization.

5.2.3 Definitions, Acronyms and Abbreviations

BOM Bill Of Materials

CAD Computer Aided Design

Chip Shooter Common name for a high speed chip placement machine (such as a Fuji CP IV, Sanyo TCM 60 etc.).

CIM Computer Integrated Manufacturing

CP Chip Placer (same as chip shooter)

FP Factory Planning Tool

GUI Graphical User Interface

IEEE Institute of Electrical and Electronics Engineers

Line Balancing The process of evenly distributing workload to various...
manufacturing cells on a production line

MMS  Motorola Manufacturing Systems -- Illinois

MPC  Motorola Multi Personal Computer

PDG  Part Data Generator

Placement Sequence Optimization  Process of altering the order in which components are placed by a chip shooter so as to minimize the time taken to place these components.

Setup Optimization  Process of assigning components to feeder slots on a chip shooter in such a manner so as to minimize the feeder travel time for a given product build.

SMT  Surface Mount Technology

SMTBal  Surface Mount Technology Balance Software

SPC  Statistical Process Control

Stochastic Process  A process that can be modeled using a family of random variables that describe the evolution of the process over time.

Throughput  The rate at which products come through the end of the manufacturing line.

WIP  Work in process.

5.2.4 References


5.2.5 Overview

The remainder of this chapter is organized into two sections. The first one is titled "General Description" and describes the factors that impact the FP software requirements. The second section is titled "Specific Requirements" and is the most important section in this chapter. This section includes a detailed description of the inputs and outputs corresponding to each of the software functions.
5.3 General Description

5.3.1 Product Perspective

The FP will comprise of a number of different software applications, some of which have already been developed at Motorola. Others will either be developed within Motorola or will be procured from outside vendors.

5.3.2 Product Functions

Line Layout
The Line Layout function will enable the user to manually configure the front and back ends of a production line using pre-defined station icons. These icons will represent the different types of auto/manual placement, test and other equipment that can be included on the line. The Line Layout function will also enable the user to utilize pre-defined templates to configure a given line. These templates will include sets of front end and back end configurations and will enable the user to re-use portions of previously defined layouts.

Line Evaluation
The line evaluation function will enable the user to measure the line performance in terms of throughput and WIP. It will enable the user to automatically configure a simulation model using low level CAD and BOM data to generate accurate estimates of line throughput. The line evaluation function will enable the user to iterate through different scenarios and thereby determine the settings of the following parameters so as to maximize the overall line throughput:

- Batch sizes to use for each product.
- Setup policies by machine and product.
- Buffer sizes between front-end and test.

Board Assignment to Lines
This function will generate a mapping of board types to manufacturing lines in order to minimize the production time on each line.
Board Family Generation
This function will take the output (board assignments to lines) from the above function and will group a number of board types from the same line into families for common setups in order to minimize the total setup time for the line.

Board Sequencing
The board sequencing function will provide the user with a sequence in which a family of boards is to be run on a given line so as to minimize the total setup time incurred.

Front End Throughput Optimization
The front end throughput optimization function will enable the user to allocate components (for an entire product mix) to front-end machines so as to balance the workload on them. It will also generate optimized NC programs and optimized setup files for each product and machine.

Scenario and Data Editing
The scenario/data editor will enable the user to alter data attributes used by the simulation, optimization and other software modules of the FP. The scenario/data editor will also allow the user to configure and save a particular data set as a scenario in order to use the FP later on without having to re-enter the data.

5.3.3 User Characteristics

The FP will be used by different types of users depending upon the problems being addressed. It will be used by managers when they are planning the deployment of a new line or factory. Managers will also use the tool to facilitate the decision making process for assigning products to existing lines. The FP will also be used by line engineers when throughput impacts of BOM and process changes have to be evaluated prior to their implementation on the line.

5.3.4 General Constraints

The FP will have to be architected in a manner so that it can later be deployed to other Motorola factories without significant alterations. It will have to drive a Witness simulation model running in a UNIX environment.
5.3.5 Assumptions and Dependencies

- The FP software will be developed in a UNIX environment.

5.4 Specific Requirements

This section describes details that the software development team needs to know in order to complete the software design for the application. Section 5.4.1 describes the functional requirements by individual function. Section 5.4.2 describes the preliminary external interface requirements of the software. The last two sections respectively list design constraints that are currently known and miscellaneous requirements.

5.4.1 Functional Requirements

For all the functional requirements, the following format is used to describe the sources of inputs for each of the functions. For inputs that are provided by the user and are formatted (seamlessly) by the Simulation Driver, the sources are described as "User (Simulation Driver)". If these inputs are also retrievable from the FP data base, their sources are listed as "User (Simulation Driver)/FP Data base". For Inputs initially provided by the user, which are later retrievable from the FP data base, the sources are listed as "User/FP Data base". (The user provided inputs are listed in bold simply to allow the reader to quickly assess the amounts of data the user needs to input to execute each of the software functions.)

5.4.1.1 Line Layout

5.4.1.1.1 Introduction

The Line Layout function enables the user to manually configure the front and back end of a manufacturing line using graphical templates representing different machine types and line layouts.

5.4.1.1.2 Inputs

Listed below are the required inputs and their sources.

<table>
<thead>
<tr>
<th>Inputs</th>
<th>Sources</th>
</tr>
</thead>
<tbody>
<tr>
<td>Line ID</td>
<td>User</td>
</tr>
<tr>
<td>Graphical template for each machine on the line</td>
<td>FP Data base</td>
</tr>
</tbody>
</table>
Chapter 5

-- Screen Printer ---- FP Data base
-- Chip Shooter ---- FP Data base
-- Odd Parts Placer ---- FP Data base
-- Reflow Oven ---- FP Data base
-- Manual Stations (Insertion, Inspection, etc.) ---- FP Data base
-- Other Equipment (Shields, Shear, Heat Seal, etc.) ---- FP Data base
-- Test Stations ---- FP Data base
-- Repair Stations ---- FP Data base
-- Assembly Stations ---- FP Data base
-- Packing Stations ---- FP Data base

- Graphical template for buffers ---- FP Data base
- Machine ID for each selected machine ---- User/FP Data base
- Buffer ID for each selected buffer ---- User/FP Data base
- Machine Type ---- User/FP Data base

5.4.1.1.3 Processing

The user would input the line ID for which the layout is being configured. He/she would configure the line by any combination of the following three ways:

* Selecting each of the stations that are to be included in the line.
* Selecting front/back end templates from the data base.
* Modifying front/back end templates selected from the data base.

5.4.1.4 Outputs

The line layout is displayed according to the machines and sources of input and output for each machine selected by the user.

5.4.1.2 Line Evaluation

5.4.1.2.1 Introduction

The Line Evaluation function will enable the user to iterate through different scenarios to determine what the batch sizes should be for each product, what the buffer sizes should be between the front end and test sections of the line and whether common or dedicated setups should be used for various products on different machines. This function will seamlessly tie into a stochastic simulation model which will use low level BOM and CAD data to generate line throughput estimates.
5.4.1.2.2 Inputs
Listed below are the required inputs and their sources.

**Inputs**

- Product List
- Production Order
- Sequencer
- Lot size by product
- For each station on the line (except reflow ovens)
  -- Station Name
  -- Quantity
  -- Type
  -- Cycle Time by product (except chip placement machines)
  -- Actions
  -- Input Rule
  -- Output Rule
  -- Mean time between failure
  -- Mean time of assist
  -- Reporting
  -- Setup Detail
    * Setup number
    * Mode
    * Setup Time
    * Description
    * Station Number
  -- Push or Pull Rules (preconditions, sources, destinations)
- For each reflow oven (modeled as a conveyor)
  -- Conveyor Name
  -- Quantity
  -- Type
  -- Part Length
  -- Maximum Capacity

**Sources**

- User/FP Data base
- User/FP Data base/Board
- User/FP data base
- FP Data base (Line Layout)
- FP Data base (Line Layout)
- FP Data base (Simulation Driver)
- User/FP Data base
- FP Data base (Simulation Driver)
- (Simulation Driver)/FP Data base
- User (Simulation Driver)/FP Data base
- User (Simulation Driver)/FP Data base
- FP Data base (Simulation Driver)
- User/FP Data base
- FP Data base (Simulation Driver)
- (Simulation Driver)/FP Data base
- FP Data base (Simulation Driver)
- FP Data base (Simulation Driver)
- FP Data base (Simulation Driver)
- FP Data base (Simulation Driver)
- FP Data base (Simulation Driver)
-- Input Rule
-- Output Rule
-- Cycle Time by product
-- Mean time between failure
-- Mean time of assist
-- Setup time for profile changes
-- Reporting

* For each buffer
  -- Buffer Name
  -- Buffer Quantity
  -- Capacity
  -- Input Position
  -- Output Scan From
  -- Select Rules
  -- Actions
  -- Reporting

* For attribute definitions
  -- Attribute Name
  -- Quantity

* For variable definitions
  -- Name
  -- Quantity
  -- Reporting

* For part definitions
  -- Part Name
  -- Type
  -- Actions
    * P_Type
    * N_Passes
* Pass
* Panel

- Product route branch condition (if one exists) selected from a pre-defined list of options
- Estimated placement rate by chip placement machine setup and placement sequence
- Estimated change out time by feeder & chip placement machine
- Estimated setup time by feeder & machine

5.4.1.2.3 Processing
The above listed data attributes will be automatically used to configure and run a simulation model and a simulation report will be generated for the user to analyze.

5.4.1.2.4 Outputs
The Line Evaluation function will generate a standard Witness simulation report. The user will have the option of generating a full blown report or selecting a set of specific metrics which are to be included in the report. The metrics would include the following:

- Estimated line capacity for a given line configuration, product mix and production order.
- WIP levels between front end and test and between stations.
- Machine metrics (for all machines, including bottleneck):
  -- % actual production time
  -- % time waiting for work
  -- % setup time
  -- % planned down time
  -- % down time

In addition, when the user chooses to evaluate different production scenarios, the Line Evaluation function will also generate graphs to allow the user to compare the results of different scenarios. Finally, all report data will be stored in the data base so that the user can retrieve this data at later time if required.
5.4.1.3 Board Assignment to Lines

5.4.1.3.1 Introduction
This function will assign different boards to various lines in order to minimize the
production time on each line. The assignment solution will maintain constraints wherein
some boards must be assigned to either the same or different lines. The function will be
implemented in two phases -- the first phase will form subgroups based on part
similarities across BOM files. The second phase will assign subgroups to lines so as to
balance the workload on each line and minimize the production time on each line.

5.4.1.3.2 Inputs
Listed below are the required inputs and their sources.

Inputs -- Phase I
- BOM file name for each product or
- Part numbers used by each product
- Board processing time
- Production quantity/board
- Number of feeder slots/part
- Subgroup maximum capacity
- Pairs of boards that need to be assigned to different/same subgroups

Sources
User/FP data base
FP data base
Placement Sequence
Optimizer/FP data base
User/FP data base
FP data base
User/FP data base
FP data base

Inputs -- Phase II
- Number of lines
- Number of subgroups
- Processing time for each subgroup
- Lists of subgroups that need to be fixed on given lines
- Lists of subgroup pairs that cannot be on the same line

Sources
User/FP data base
FP data base
FP data base
FP data base
FP data base

5.4.1.3.3 Processing
This function will determine the which board families are to be assigned to which lines
based on criteria such as the following:
Chapter 5

- Part commonalities (based on the bill of materials for each board family)
- Estimated processing time per board per machine (to balance the workload on the line)
- The feeder capacity for each machine
- Constraints which require some boards to be paired up on the same line
- Mutually exclusive constraints which require some boards to be processed on different lines

5.4.1.3.4 Outputs

Phase I
Number of subgroups formed, production time and conveyor width for each subgroup, part list for each subgroup.

Phase II
Subgroup assignments to lines, estimated production times for given lot sizes on each line.

5.4.1.4 Board Family Generation

5.4.1.4.1 Introduction
This function will generate common setups for various boards that have been assigned to a given line (by the above function) in order to minimize the total setup time for the line.

5.4.1.4.2 Inputs
Listed below are the required inputs and their sources.

<table>
<thead>
<tr>
<th>Inputs</th>
<th>Sources</th>
</tr>
</thead>
<tbody>
<tr>
<td>Line ID for each line</td>
<td>User/FP data base</td>
</tr>
<tr>
<td>Subgroups for each line</td>
<td>User/FP data base</td>
</tr>
<tr>
<td>Parts required for each subgroup</td>
<td>User/FP data base</td>
</tr>
<tr>
<td>Conveyor width for each subgroup</td>
<td>FP data base</td>
</tr>
<tr>
<td>Parts setup time for each line</td>
<td>FP data base</td>
</tr>
<tr>
<td>Production batch size for all subgroups</td>
<td>FP data base</td>
</tr>
<tr>
<td>(across lines)</td>
<td>FP data base</td>
</tr>
<tr>
<td>Feeder capacity for each line</td>
<td>FP data base</td>
</tr>
</tbody>
</table>

5.4.1.4.3 Processing
The board families will be generated by evaluating certain parameters such as:

- Bill of material commonality between boards
- Time required to setup the machines on each line
- Lot sizes for production
- Total feeder capacity of a given line (based on the line configuration)
- Conveyor width required for a given board
5.4.1.4.4 Outputs
Groups that comprise common setups on each line, total setup time for each line.

5.4.1.5 Board Sequencing

5.4.1.5.1 Introduction
This function will generate an optimal sequence in which to release boards to the line in order to minimize setup changeovers between builds. This feature will be particularly useful if the line is being run in a high-mix mode. The inputs, data processing and outputs corresponding to the board sequencing function are listed below.

5.4.1.5.2 Inputs
Listed below are the required inputs and their sources.

<table>
<thead>
<tr>
<th>Inputs</th>
<th>Sources</th>
</tr>
</thead>
<tbody>
<tr>
<td>• Line ID</td>
<td>User/FP data base</td>
</tr>
<tr>
<td>• BOM file name for each product to be sequenced or Part numbers used by each product</td>
<td>User/FP data base</td>
</tr>
<tr>
<td>• Machine Feeder Capacity</td>
<td>User/FP data base</td>
</tr>
</tbody>
</table>

5.4.1.5.3 Processing
The Board Sequencing function will use an optimization heuristic (such as one based on the shortest path problem) and improvement techniques (such as two and three opting) to determine the order in which to produce boards so that the total number of feeder change outs is minimized.

5.4.1.5.4 Outputs
Suggested sequence of building products

5.4.1.6 Front End Throughput Optimization
The front end throughput optimizer enables the user to balance and optimize front end manufacturing processes. It provides the user with optimized NC programs for each machine and product, optimized setup files for each machine and product and a high level...
summary report containing important information such as estimated machine setup times, estimated build times by lot etc. The front end throughput optimization function will be provided by integrating the FP with SMTBal (a software tool previously developed in-house at Motorola) and hence all the line throughput optimization functions that are contained in SMTBal, their inputs, data processing and outputs are not discussed here.

5.4.1.7 Scenario and Data Editing

5.4.1.7.1 Introduction
The Scenario and Data Editing function enables the user to alter data attributes used by other modules of the software (see Line Layout, Line Evaluation, Board Assignment to Lines, Board Family Generation, Board Sequencing and Front End Throughput Optimization function descriptions). In addition, it enables the user to save or edit scenarios in accordance with particular data configurations. The Scenario and Data Editor will also allow the user to update data in the factory data base if the user decides to do so.

5.4.1.7.2 Inputs
The inputs include all the variables that are used by the other functions listed in the above paragraph. Listed below are the inputs and their sources.

Inputs Sources

- Line ID User/FP database
- Machine ID for each machine or station User/FP Data base
- Buffer ID for each buffer User/FP Data base
- Product List User/FP database
- Production Order User/FP Data base/Board Sequencer
- Lot size by product User/FP database
- For each station on the line (except reflow ovens)
  -- Cycle Time by product (except chip placement machines) User/FP Data base
  -- Input Rule (Simulation Driver)/FP Data base
  -- Output Rule (Simulation Driver)/FP Data base
  -- Priority User (Simulation Driver)/FP Data base
Chapter 5

-- Breakdowns

-- Setup Detail
   * Mode

   * Setup Time

-- Push or Pull Rules (preconditions, sources, destinations)

* For each reflow oven (modeled as a conveyor)
   -- Input Rule

   -- Output Rule

   -- Cycle Time by product

* For each buffer
   -- Capacity

* Product route branch condition (if one exists) selected from a pre-defined list of options

* Estimated placement rate by chip placement machine setup and placement sequence

* Estimated change out time by feeder & chip placement machine

* Estimated setup time by feeder & machine

* Estimated conveyor travel rate

* BOM file name for each product to be sequenced or

* Part numbers used by each product

* Machine Feeder Capacity

5.4.1.7.3 Processing
This function will serve primarily as an editing function and will not involve any data processing.

5.4.1.7.4 Outputs
Messages to the user that the data or scenario edits have been successfully completed.
Chapter 5

5.4.2 External Interface Requirements

5.4.2.1 User Interfaces

A Motif GUI will be developed for the application in order to enhance its ease of use. It will be mouse-driven and will be X compatible. The GUI will be developed using TclTk to reduce product cost and enhance the FP application portability from one factory to another.

5.4.2.2 Software Interfaces

The FP will interface with the following software applications as described below:

<table>
<thead>
<tr>
<th>Software Application</th>
<th>Mechanism of Interface</th>
</tr>
</thead>
<tbody>
<tr>
<td>• IP Solver</td>
<td>ASCII file</td>
</tr>
<tr>
<td>• Simulation software (Witness)</td>
<td>ASCII file in a specific format required and specified by Witness</td>
</tr>
<tr>
<td>• FP data base</td>
<td>Data base tables, data base scripts. The FP data base schema will remain unchanged when the Factory Planning software is transferred from one factory to another.</td>
</tr>
<tr>
<td>• Factory data base</td>
<td>Data base tables, data base scripts. (The Factory data base will interface with the FP data base via a data transfer interface.</td>
</tr>
<tr>
<td>• Data Transfer Interface</td>
<td>This piece of software will enable data to be uploaded from the Factory data base to the FP data base. It will also enable the user to make edits in the Factory data base. The Data transfer interface is the only piece of software that will have to change when porting the Factory Planning Application from one site to another.</td>
</tr>
</tbody>
</table>

5.4.2.3 Communications Interfaces

To be determined according to the local network protocols that exist in the customer's factory environment.
5.4.3 Design Constraints

- The software architecture must support data source independence.

- The software architecture must support application independence. It must be designed to allow functional blocks of the system to be interchanged with other applications which provide the same functionality.

- The architecture will not introduce any requirements for redundant data entry.

- As far as possible, duplication of effort will be avoided -- hence, previously developed tools that provide functionalities targeted in the FP will be integrated into the FP.

5.4.4 Other Requirements

Although the pilot application will be developed for the initial customer site, the software will have to be modular, easy to change and portable to other Motorola facilities in the future.

For instance, although the initial application is being developed for a factory using primarily Panasonic placement machines, the application will be easily portable to other facilities using different machines (such as Fuji, Sanyo, Universal, TDK, etc.).
Chapter 6

6.0 SOFTWARE ARCHITECTURE (HIGH LEVEL)

6.1 Chapter Overview

This chapter outlines the architectural requirements of the Integrated Factory Planning Software and describes the roles played by and the data flows between the main software modules.

6.2 Architecture Requirements

- The architecture must support data source independence. It must be architected to minimize a port to a new data environment (e.g., flat files, different data base schema).

- The architecture must support application independence. It must be designed to allow functional blocks of the system to be interchanged with other applications which provide the same functionality.

  -- Modules of the software will also be independent in that they will not require other software modules in order to perform their function. Data will be passed among modules but global variables will not be used to share data among modules.

- The architecture will not introduce any requirements for redundant data entry.

- The system will be able to be ported to any UNIX platform with a relatively small effort.

6.3 Architecture Assumptions

(Please refer to Figure 5 below).

- Arrows depict data flows and/or control flows.

- Data error message flows not shown.

- The Factory Data Base is a generic term used to denote an existing data base in a given factory environment.

- The Factory Planning Data Base is specific to the Factory Planning Application and contains all the data required by the application.
6.4 Software Modules

This section describes the primary software modules of the application. It does not describe the Build Aid Generation and Auto Data Update Tools (please see Section 6.4.1) since these modules have not yet been clearly defined.
Figure 5  Software Architecture (High Level)

Development of a Seamlessly Integrated Factory Planning Software Tool (Prototype) to Evaluate and Optimize Surface Mount Manufacturing Lines

Page 59
6.4.1 Main User Interface

Purpose
This serves as the interface between the user and the Factory Planning Application and connects to the following other functions:

• Editor Modules
• Line Evaluation Report Generator
• Line Evaluator/Optimizer Interface
• Auto Data Update Tool
• Build Aid Generation Tool

Figure 5 shows all the required connections between the software modules.

6.4.2 Editor Modules

(Please refer to Figure 6 below)

Purpose
These modules enable the user to edit data required by various different functions of the Factory Planning software.

Inflows and Outflows
A list of sample inflows and outflows are listed under each of the editor modules in Figure 6. The Editor modules will obtain inflows from and give outflows to the Factory Planning data base.
Chapter 6

Figure 6
Factory Planning Application Editor Modules

Legend
- New Functionality
- Incorporation into current factory tools
- Main Logic Flow

Legend
- Machine Type
- Setup Data Editor
- Placements Data Editor
- Parts Restriction Data Editor

Scenario Selector
- Enter Scenario Name
- Define Scenario

Machine Attribute Editor
- Machine Type
- # of Feeders
- Board Load/Unload Time
- Preload Read Time
- Preload Change Times
- Avg Place Rate
- Max Place Rate
- Geometry to Folders/Associations
- Geometry to Cam Speeds
- Bond Times
- Bond Temperatures

Lot Attribute Editor
- # of Lots
- # of Panels/Lot

Setup Data Editor
- Machine Setups by Slot

Placement Data Editor
- Component Rotations
- Polarized Components
- Force Parts off Slot
- Group Setup by Machine & Part #

Panel Layout CAD EF
- Panel X Offset
- Panel Y Offset
- Panel Scale

Board Layout CAD EF
- Board X Offset
- Board Y Offset
- Board Rotation
- # Board/Panel

Component Data Editor
- Component Data
- Dimension
- CAD Attributes
- Part # Ref Des
- Associations
- Ref Des
- X,Y Locations
- CAD/MCS Load

CAD Data EF
- Vendor Data
- Vendor Name
- Vendor Prices
- Reel Size
- Component/Reel
- Component Cost

BOM Data EF
- Component Cost
- Component Quantity
- Component Quality
- Component Update Parameter Editor

Simulation Parameter Editor
- Distribution Functions
- Cycle Times

Simulation Rule Editor
- Model Rules
- Practices

Simulation Model Editor
- Line Layout
- Direct Labor Rate
- Capital Expense/Machine

Factory Cost Data Editor
- Historic Yield Info by Process
- Component
- Unit/Week
- Unit/Direct Labor
- Units/Sq Ft

Process Quality Data EF
- Historic Yield
- Info by Process
- Component
- Unit/Week
- Unit/Direct Labor
- Units/Sq Ft

Component Quality Data EF
- Component
- Unit/Week
- Unit/Direct Labor
- Units/Sq Ft

Auto Data Update Parameter Editor
- Component
- Unit/Week
- Unit/Direct Labor
- Units/Sq Ft
6.4.3 Line Evaluator/Optimizer Interface

**Purpose**
To allow the user to enter the criteria upon which to evaluate a given line configuration against throughput, cost and quality metrics and to allow a user to seamlessly call appropriate modules to generate optimized setups and placement programs for the front end of the line.

**Inflows**
* From User
  * User-specified questions/inputs which describe the attributes of the line that the user wants to evaluate or optimize.

* From Task Manager
  * Messages indicating which software modules are executing and whether or not the entire evaluation has been completed.

**Outflows**
* To Factory Planning Data Base
  * Queries to retrieve scenario information

* To Task Manager
  * User-specified data which describes the attributes of the line that the user wants to evaluate or optimize.

6.4.4 Task Manager

**Purpose**
To sequence the appropriate software module(s) to answer the questions that the user asks or configures.

**Inflows**
* From Line Evaluator/Optimizer Interface
  * User-specified questions/inputs which describe the attributes of the line that the user wants to evaluate or optimize.

**Outflows**
* To Line Evaluator/Optimizer Interface
  * Messages indicating which software modules are executing and whether or not the entire evaluation has been completed.
Chapter 6


Control flows (messages/commands) to execute appropriate software modules (with appropriate lists of attributes that need to be retrieved from the data base) in order to answer the questions posed by the user.

6.4.5 Board Assignment to Lines

Purpose
Generates board and line relationships.

Inflows
From Factory Planning Data Base
- BOM data by product
- Material handling data (conveyor width, speed, type, machine compatibility, etc.)
- Line configuration data -- which machines are on the line and in what order.

Outflows
To Factory Planning Data Base
- Board assignments to each line.

6.4.6 Board Family Generation

Purpose
Generates board family groups for each machine on the line.

Inflows
From Factory Planning Data Base
- BOM data by product
- Machine Attributes (Number of feeders, component placement capability, etc.)

Outflows
To Factory Planning Data Base
- Board family groups for each machine on the line.
6.4.7 Board Sequencing

Purpose
Generates a sequence in which to build different products so as to minimize the total setup changes incurred.

Inflows

* From Factory Planning Data Base
  - Machine setup information by product and line.
  - Board family groups by line and machine.
  - BOM data by product.

Outflows

* To Factory Planning Data Base
  - Order in which different products should be made on the line.

6.4.8 Component Allocation

Purpose
Determines which components are to be placed by which machines so as to minimize the total manufacturing time (placement plus setup) for each board.

Inflows

* From Factory Planning Data Base
  - Sequence in which to build products.
  - Line layout
  - Machine configuration (feeder capacity, average feeder change times, maximum placement rates etc.).
  - Board family groups by line and machine (generated by Board Family Grouping software).
  - Lot information (lot sizes, number of lots etc.).
  - Source file information (CAD or NC, file format information, etc.)
  - BOM data by product.

* From IP Solver
  - MPS file containing information on component-machine relationships.
Chapter 6

Outflows

To IP Solver
- MPS file containing line layout, machine configuration, board family and lot size constraints.

To Factory Planning Data Base
- List of component-machine allocations by product and line.

6.4.9 Placement Sequence Optimizer

Purpose
Minimizes the placement cycle time for a given board on a given machine by altering the order in which the placements are made. Also estimates placement cycle time corresponding to any given board, machine setup, placement sequence and BOM.

Inflows

From Factory Planning Data Base
- List of component-machine allocations by product and line.
- Component location and orientation (CAD information).
- BOM by product.
- Machine configuration (machine setup, maximum number of feeders etc.).
- Product/process specific information (PDG -- Part number associations, cam speeds by part number etc.).
- Constraint information (which parts must always be placed last in a sequence of placements, etc.)

Outflows

To Factory Planning Data Base
- NC file names or placement sequence information.
- Estimated cycle times for placement sequences corresponding to a given product and machine setup.

6.4.10 Simulation Driver

Purpose
To configure and run a simulation model according to user defined parameters.
Chapter 6

**Inflows**

* From Factory Planning Data Base
  * Description of line layout
  * Simulation parameters
  * Model rules

**Outflows**

* To Simulation software
  * ASCII files or messages/commands to drive the simulation.

* To Factory Planning Data Base
  * Simulation driver configurations.

6.4.11 Simulation Model (Witness)

**Purpose**

To generate accurate estimates of line throughput on the basis of a stochastic model that factors machine breakages, maintenance, variations in cycle times, etc.

**Inflows**

* From Simulation Driver
  * Estimated cycle times for placement sequences corresponding to a given product and machine setup.
  * Simulation parameters (model rules, distribution functions representing machine interruptions etc.)
  * Simulation object information (icons representing machines, conveyors, assembly stations etc.).
  * Line configuration information (line layout, machine configurations etc.)
  * Estimated placement times corresponding to given products and machine setups.
  * Other estimated process times (setup times by machine and feeder type, feeder change out times by machine and feeder type etc.).
  * Build data (number of boards per panel, panel lot sizes etc.)

**Outflows**

* To Factory Planning Data Base
  * Simulation Results
6.4.12 Line Evaluation Report Generator

**Purpose**
To generate reports which will answer the particular questions being raised by the user.

**Inflows**
* From Factory Planning Data Base
  - List of metrics against which the user wants to evaluate the line (throughput, quality, etc.)

**Outflows**
* To Line Evaluator/Optimizer Interface
  - Line performance information (in the form of text, graphs etc.) according to the metrics selected by the user.
* To Factory Planning Data Base
  - Line performance information pertaining to a given report.

6.4.13 Cost Evaluation Tool

**Purpose**
Determines the product cost in terms of the cost metrics selected by the user (direct materials, direct labor, total cost etc.).

**Inflows**
* From Factory Planning Data Base
  - Metrics against which costs are to be computed (direct materials, total cost, labor dollars per unit etc.).
  - BOM by product.
  - Build data (lot sizes, number of panels per lot etc.)
  - Factory cost data (direct labor rate, capital expenditure/machine, depreciation/machine, etc.)
  - Component costs.
6.4.14 Quality Evaluation Tool

**Purpose**
Determines the product/process quality in terms of the quality metrics selected by the user (ppm by process, defect yields by component etc.).

**Inflows**
* From Factory Planning Data Base
  - Metrics against which quality is to be computed (sigma levels by process, etc.).
  - Product information (BOM).
  - Process information (line configuration, etc.).
  - Historical yield information by process.
  - Historical yield information by component.

**Outflows**
* To Factory Planning Data Base
  - Quality data generated by the Quality Evaluation software.

6.4.15 Integer Programming Solver

**Purpose**
Determines component-machine associations to minimize the total production time (placement and setup) for a given product and line configuration.

**Inflows**
* From Auto Component Allocation Software
  - MPS file containing line layout, machine configuration, board family and lot size constraints.

**Outflows**
* To Auto Component Allocation Software
  - MPS file containing information on component-machine relationships.
6.4.16 NC Generation

Purpose
To generate NC files for placement machines (corresponding to a given machine setup, product BOM and CAD file).

Inflows
* From Factory Planning Data Base
  * Machine configuration (feeder setups, etc.).
  * Placement sequence.
  * File format type
  * Component information (locations, degree rotations, cam speeds, etc.)
  * Component -- part number associations.

Outflows
* To Factory Planning Data Base
  * NC information for different placement machines.

* To User-Specified Directory
  * NC Files for different placement machines

6.4.17 Setup Generation

Purpose
To generate machine setup information (setup files, setup sheets).

Inflows
* From Factory Planning Data Base
  * Machine configuration (number of feeders, etc.)
  * Part number -- feeder associations.
  * File format type.

Outflows
* To Factory Planning Data Base
  * Setup information by machine and product

* To User-Specified Directory
  * Setup files.
  * Setup sheets (change over aids).
6.4.18 Data Transfer Interface

**Purpose**
To enable the Factory Planning database to upload data from a given factory database and to minimize the amount of rework that has to be done when porting the Factory Planning software from one factory environment to another.

**Inflows**
* From Factory Data Base
  - All the data attributes required by the above modules (see inputs for each of the modules).

* From Factory Planning Data Base
  - Data Attributes that the user wants to modify in the Factory Data Base.

**Outflows**
* To Factory Planning Data Base
  - Data attributes required by the above software modules.

* To Factory Data Base
  - Data Attributes that the user wants to modify in the Factory Data Base.

6.4.19 Factory Planning Data Base

**Purpose**
To enable the Factory Planning software modules to be independent of any particular factory database schema and to make task of porting the Factory Planning software from one factory to another easier.

**Inflows**
* From Data Formatter/Translator
  - All the data attributes required by the above modules (see inputs for each of the modules).

* From Editor Modules
  - All user-specified edits to data in the Factory Planning database.

* From the Board Assignment to Lines, Board Family Generator, Board Sequencing, Placement Sequence Optimizer, Simulation Driver, Simulation Model, Component Allocation, Setup Generation, NC Generation, Cost Evaluation, Quality Evaluation
Chapter 6


- All the outputs (data attributes) generated by the above modules (see outputs for each of the modules).

Outflows

To Editor Modules

- All data attributes required by the editor modules


- Data attributes required by the above software modules.

6.5 Software Modules Developed for Pilot Application

Figure 7 below indicates the subset of software modules that were developed for the Factory Planning Prototype Application. These modules were targeted because:

- They make up the core of the overall Factory Planning Application -- modules which will be developed in the future will either utilize data generated by these modules or they will be integrated with these modules.

- They represent one of the most difficult parts of the Factory Planning Application. At the start of the project, it was uncertain whether the Simulation Driver module could be developed to provide all the functionalities that it currently provides. This uncertainty coupled with the fact that the Simulation Driver was considered to be absolutely essential to the success of the Factory Planning application led to the decision to include this module into the set of modules targeted to be developed for the prototype application.

(The development of the user interface was not challenging in itself but was included in the first phase so as to provide the user with a ready-to-use prototype application).

6.6 Architectural Issues Pertaining to the Factory Planning Data Base

One of the fundamental issues that had to be addressed while establishing the Factory Planning software architecture pertained to the design of the Factory Planning data base. One approach was to add tables to manufacturing databases at individual factory sites so
Chapter 6

that they would be able to store data required by the Factory Planning Application. This would be done by adding Application Specific Tables (ASTs) to existing data bases. The pros and cons of this approach were analyzed by team members and other Motorola personnel responsible for maintaining manufacturing databases at different Motorola factories. The findings of this analysis are outlined below.
Figure 7 Software Modules Developed for Pilot Application
Chapter 6

Factory Planning Database Architecture Alternatives

Option 1 - Application Specific Tables

Figure 8 Representation of the Application Specific Tables (ASTs) Approach for the Factory Planning Database Implementation

The existing Factory Database is enhanced and is used to store all Factory Planning data. Tables will need to be segregated for data that remains common for both the factory CIM system and the Factory Planning application. A local copy of manufacturing data will be accessed by the Factory Planning Application for simulation and "what if" analyses. Each software module of the Factory Planning Application would be isolated from the actual database schema by a data translator/formatter. Data retrieval and storage requests would be made independent of the database schema and the translators would perform the actual data request. A data transfer interface would update the appropriate data base tables prior to and after executions of the Factory Planning software modules.
Option 2 - Self Contained FP Database

Figure 9  Representation of the Independent Database Approach for the Factory Planning Database Implementation

All information needed by the Factory Planning Application is transferred from the Factory database to an independent Factory Planning data base. Simulations and "what if" scenarios are constructed from data residing in the Factory Planning data base. Factory Planning software modules retrieve and store data directly from/to the Factory Planning data base, which is identical across all Factory Planning installations in different factories. A data transfer interface would update the appropriate Factory data base tables if and when the user desires to do so.
### Comparison of the two Approaches

<table>
<thead>
<tr>
<th>Characteristic</th>
<th>AST Implementation</th>
<th>Self Contained DB</th>
</tr>
</thead>
<tbody>
<tr>
<td>Application Dependency</td>
<td>Isolated to the formatter interface.</td>
<td>Isolated to the relevant database schema.</td>
</tr>
<tr>
<td>Update to Factory Database</td>
<td>Update Data Xfer Interface, Update Data Translators</td>
<td>Update Data Xfer Interface</td>
</tr>
<tr>
<td>Editor Implementation</td>
<td>Existing Editors updated and enhanced for management of new data attributes. Screens can be made to reflect same layout (to prevent operator confusion). Care would need to be taken to ensure modification of correct data repositories from identical looking screens.</td>
<td>Separate set of editors for each database/system. Layout of Factory Planning screens designed for common look and feel. Although screens are available for modification of any data element, configuration updates made automatically by Xfer Interface.</td>
</tr>
<tr>
<td>Customer Integration Responsibility</td>
<td>Create/Update database tables, assist in development of Data Xfer Interface, create/Update data editors.</td>
<td>Generate/read flat files (database tables) as per standard interface documentation.</td>
</tr>
<tr>
<td>Data Redundancy</td>
<td>none</td>
<td>Static Configuration Data (automatic updates)</td>
</tr>
<tr>
<td>Data Validation</td>
<td>Limited up front checking based only on schema layout and enhancements made to existing data editors.</td>
<td>Built in validation made at data Xfer and local editor updates.</td>
</tr>
<tr>
<td>Factory Roll out Requirements</td>
<td>Analyze Factory database schema, update local copy tables, update Application Specific Tables, update Data Xfer Interface, update data translators, update data editors.</td>
<td>Update Factory database side of Xfer Interface.</td>
</tr>
<tr>
<td>Application Enhancement Requirements</td>
<td>Update Factory database, update data translators.</td>
<td>Update Factory Planning database</td>
</tr>
<tr>
<td>Database Maintenance</td>
<td>One database</td>
<td>Two databases</td>
</tr>
</tbody>
</table>

Table 2: Comparison of the two Approaches for the Factory Planning Database
Chapter 7

7.0 SOFTWARE REQUIREMENTS (SIMULATION DRIVER)

7.1 Chapter Overview

This chapter provides the reader with a detailed description of the various functionalities that are to be provided by the Simulation Driver Module of the Factory Planning Application. The different sections of this chapter are written in a format prescribed by the "Software Engineering Standards" -- a document compiled by the Institute of Electrical and Electronics Engineers (October 1989).

7.2 Introduction

In order to address the detailed requirements of the Simulation Driver software module, the inputs, outputs and data processing associated with each of the functions that comprise the Simulation Driver module listed in the following sections.

7.2.1 Purpose

This chapter is intended to describe the software requirements of the Simulation Driver in enough detail so as to enable the software development team to commence working on the software design and development for this module. The level of detail will be such that the software design and implementation can be carried out simultaneously with development activities of other software modules that make up the Factory Planning Application.

7.2.2 Scope

The Simulation Driver will include the following primary software functions

- Line Configuration Data Reading
- Production Data Definition
- Simulation Model Configuration and Execution
- Line Evaluation Report Configuration

In addition, the Simulation Driver will also have the following functions which play a supporting role to the primary functions and are called by the primary functions.

- Oracle Communication Logon
Chapter 7

- Oracle Communication Logoff
- Integer Conversion to ASCII String
- Reverse String
- Retrieve Route Cycle Time Data Structure Information
- Comparison of Records (by board side) in the Product Information Data Structure
- Comparison of Records (by production sequence) in the Product Information Data Structure
- Comparison of Records in the Temporary Data Structures
- Build Routing Cycle Time File
- Build Production Order File
- Build ASCII File
- Run Simulation
- Extract Results Data
- Store Report Data in Data Base

The above functions will collectively enable the user to transparently configure and execute a Witness simulation model corresponding to a given line configuration, product mix and production sequence.

7.2.3 Definitions, Acronyms and Abbreviations

LCDR Function Line Configuration Data Reading Function
PDD Function Production Data Definition Function
SMCE Function Simulation Model Configuration and Execution Function
LERC Function Line Evaluation Report Configuration Function
FP Factory Planning

7.2.4 References


7.2.5 Overview

The remainder of this chapter is organized into two sections. The first one is titled “General Description” and describes the factors that impact the Simulation Driver software requirements. The second section is titled “Specific Requirements” and is the most important section in this chapter. This section includes a detailed description of the inputs, outputs and data processing associated with each of the functions in the Simulation Driver.
This chapter is organized on the basis of the IEEE Guide to Software Requirements Specifications.

7.3 General Description

7.3.1 Product Perspective

The Simulation Driver is a module of the Integrated Factory Planning Software application and enables the user to transparently configure and execute a Witness simulation model representing a given line/factory scenario without requiring the user to know any of the Witness modeling rules or the required Witness syntax. It also enables the user to customize (in a "point and click" environment) graphical simulation reports according to specific metrics that the user wishes to measure the line/factory against.

7.3.2 Product Functions

Line Configuration Data Reading Function
This Function automatically obtains (from the Factory Planning data base) the line/factory configuration information for a given line/factory. This function loads this data into the appropriate data structures which are subsequently accessed by other functions within the Simulation Driver to configure the ASCII files that drive the Witness simulation. The Line Configuration Data Reading Function is transparent to the user.

Production Data Definition
This function retrieves from the Factory Planning data base the production data pertaining to the list of products to be included in a scenario, the lot sizes, the sequence of production, etc. and loads this data into appropriate data structures. These data structures are subsequently accessed by other functions within the Simulation Driver to generate the required Witness ASCII file to drive the simulation. The Production Data Definition function is transparent to the user.

Simulation Model Configuration and Execution
This function completely configures a Witness simulation model (and then executes it) by coupling inputs from the above two functions with other data (transparent to the user) required to define a Witness simulation. For each of the scenarios being evaluated by the user, this function defines and executes a different simulation model.
Line Evaluation Report Configuration
This function extracts the Witness simulation results data from the Witness ASCII reports file and stores this data in the data base. These results are subsequently retrieved from the data base by the Line Evaluation Report Generator to display various graphs according to the metrics selected by the user.

Oracle Communication Logon
This function logs onto the Factory Planning data base and is called by the different functions within the Simulation Driver that retrieve and/or store data in the data base.

Oracle Communication Logoff
This function logs off from the Factory Planning data base and is called by the different functions within the Simulation Driver that retrieve and/or store data in the data base.

Integer Conversion to ASCII String
This function converts an integer value to its corresponding ASCII string.

Reverse String
This function reverses a string -- it is used in conjunction with the Integer Conversion to ASCII String function described above.

Retrieve Route Cycle Time Data Structure Information
This function retrieves the cycle time and process flow related data from the data base for the different products included in a given scenario and loads this data in the appropriate data structure.

Comparison of Records (by production sequence) in the Product Information Data Structure
This function compares the records (board name, board side, production sequence, number of panels, etc.) in the data structure containing product information on the basis of the production sequence of the given product.

Comparison of Records (by process step) in the Temporary Data Structures
This function compares the records (process step, device ID, cycle time) in the temporary data structures containing product information on the basis of the process step number corresponding to each record.
Chapter 7

Build Routing Cycle Time File
This function creates the Routing Cycle Time file which is one of the required files to configure and run the Witness simulation model. This file contains information pertaining to the process route and the cycle time (at each of the steps in the route) for each of the products being modeled in the Witness simulation.

Build Production Order File
This function creates the Production Order file which is one of the files required to configure and run the Witness simulation model. This file contains information pertaining to the order in which the different batches of printed circuit boards are released to the shop floor.

Build ASCII File
This function creates two ASCII files (which drive the Witness simulation model) for each of the scenarios being modeled. The first one is used to run the simulation once and determine the time that the simulation model should be re-run without causing starvation on the line. This time is then written to the appropriate section of the second file which is used to iteratively run the simulation for a given number of runs and generate reports after the model has reached a steady state.

Run Simulation
This function makes a system call to execute the Witness simulation model.

Extract Results Data
This function extracts the results data from the standard Witness ASCII report file and loads it into appropriate data structures.

Store Report Data in Data Base
This routine stores the results data (retrieved by the Extract Results Data function) in the data base. This data is later displayed to the user by the Line Evaluation Report Generator.
7.3.3 User Characteristics

The Simulation Driver will be accessed by people who want to evaluate the performance of a given line/factory by simulating a particular line/factory configuration and production environment.

7.3.4 General Constraints

The Simulation Driver will be accessed by the user in a transparent manner. The user will not have to be concerned with any of the simulation model-specific parameters required to configure a Witness model as the software will handle these transparently. The inputs that the user provides to define the line/factory configuration and production environment will be automatically formatted into the required formats by the Simulation Driver.

7.3.5 Assumptions and Dependencies

The Simulation Driver will run on UNIX. The Witness simulation model will also run on UNIX. (At the time of this project, Witness was running on a DOS platform and hence, the ASCII files created by the Simulation Driver were copied over to a DOS platform in order to execute a Witness model. Thereafter, the Witness report files were copied over to UNIX so that the Simulation Driver could extract the results data from these files and store this data in the Factory Planning data base).

7.4 Specific Requirements

This section describes details required to complete the software design and code generation for the Simulation Driver. Section 7.4.1 describes the functional requirements by individual function. Section 7.4.2 describes the external interface requirements of the software. The last two sections respectively list design constraints that are currently known and miscellaneous requirements.
7.4.1 Functional Requirements

7.4.1.1 Line Configuration Data Reading Function

7.4.1.1.1 Introduction
The Line Configuration Data Reading function automatically reads the line configuration data from the database corresponding to user-defined line layout for each scenario. As a result, the user is not required to re-enter line configuration data when evaluating the line (the user has previously entered this data using the Line Layout Editor (please see Section 9.4.1.4.5).

7.4.1.1.2 Inputs
Listed below are the required inputs and their sources.

<table>
<thead>
<tr>
<th>Inputs</th>
<th>Sources</th>
</tr>
</thead>
<tbody>
<tr>
<td>Factory ID</td>
<td>Task Manager</td>
</tr>
<tr>
<td>Line ID</td>
<td>Task Manager</td>
</tr>
<tr>
<td>Scenario Name</td>
<td>Task Manager</td>
</tr>
<tr>
<td>Device ID for each device on the line</td>
<td>FP Data base</td>
</tr>
<tr>
<td>Device production type (used for configuring the Witness model)</td>
<td>FP Data base</td>
</tr>
<tr>
<td>Order of each device on the line</td>
<td>FP Data base</td>
</tr>
</tbody>
</table>

7.4.1.1.3 Processing
This function generates a query to the database using the Factory ID, Line ID and Scenario Name as keys and obtains all the inputs from the database that are listed above in Section 7.4.1.1.2. These inputs are loaded into the appropriate data structures so that they can be subsequently accessed by other functions.

7.4.1.1.4 Outputs
Error codes if execution fails within this function (if there are no errors, the error code is defaulted to zero).
Chapter 7

7.4.1.2 Production Data Definition

7.4.1.2.1 Introduction
This function retrieves production environment data from the data base for each of the scenarios that the user is evaluating. If the user has defined multiple scenarios and wants to compare the performance of each, this function will be called iteratively. The data retrieved by this function is loaded into appropriate data structures and is subsequently accessed by other functions within the Simulation Driver.

7.4.1.2.2 Inputs
Listed below are the required inputs and their sources.

<table>
<thead>
<tr>
<th>Inputs</th>
<th>Sources</th>
</tr>
</thead>
<tbody>
<tr>
<td>Scenario name</td>
<td>Task Manager</td>
</tr>
<tr>
<td>Reporting Interval (data base stores this data in a measuring unit of days and the Simulation Driver converts this data to minutes)</td>
<td>User/FP Data base</td>
</tr>
<tr>
<td>Product List</td>
<td>User/FP Data base</td>
</tr>
<tr>
<td>Production Order</td>
<td>User/FP Data base</td>
</tr>
<tr>
<td>Lot size by product</td>
<td>User/FP Data base</td>
</tr>
<tr>
<td>Number of Panels/Board</td>
<td>User/FP Data base</td>
</tr>
<tr>
<td>Production Route (Including Buffers) for each product</td>
<td>User/FP Data base</td>
</tr>
<tr>
<td>For Chip Placement machines</td>
<td></td>
</tr>
<tr>
<td>-- Production Type (either chip placement, buffer, reflow, batch, assembly, singulation, test or other)</td>
<td>FP Data base</td>
</tr>
<tr>
<td>-- Cycle Time</td>
<td>FP Data base</td>
</tr>
<tr>
<td>-- Feeder replenishment time (per feeder)</td>
<td>FP Data base</td>
</tr>
<tr>
<td>-- Feeder setup time between products</td>
<td>FP Data base</td>
</tr>
<tr>
<td>-- Mean time between failure</td>
<td>User/FP Data base</td>
</tr>
<tr>
<td>-- Mean time of assist</td>
<td>User/FP Data base</td>
</tr>
<tr>
<td>For Buffers</td>
<td></td>
</tr>
<tr>
<td>-- Production Type (either chip placement, buffer, reflow, batch, assembly, singulation, test or other)</td>
<td>FP Data base</td>
</tr>
<tr>
<td>-- Capacity</td>
<td>User/FP Data base</td>
</tr>
</tbody>
</table>
Chapter 7

* For Reflow Ovens (modeled as conveyors)
  -- Production Type (either chip placement, buffer, reflow, batch, assembly, singulation, test or other)  
  -- Cycle Time
  -- Setup time
  -- Capacity (Number of boards)
  -- Mean time between failure
  -- Mean time of assist

* For Batch Machines
  -- Production Type (either chip placement, buffer, reflow, batch, assembly, singulation, test or other)  
  -- Cycle Time
  -- Setup time
  -- Mean time between failure
  -- Mean time of assist
  -- Batch min
  -- Batch max

* For Assembly Machines
  -- Production Type (either chip placement, buffer, reflow, batch, assembly, singulation, test or other)  
  -- Cycle Time
  -- Setup time
  -- Mean time between failure
  -- Mean time of assist
  -- Assembly Quantity (the number

* For Singulation Machines
  -- Production Type (either chip placement, buffer, reflow, batch, assembly, singulation, test or other)  
  -- Cycle Time
  -- Setup time
  -- Mean time between failure
  -- Mean time of assist

* For Test Machines
  -- Production Type (either chip placement, buffer, reflow, batch, assembly, singulation, test or other)  
  -- Cycle Time
  -- Setup time
  -- Mean time between failure
  -- Mean time of assist
  -- Yield
  -- Rework Loop (Rework location)

* For Other Machines
  -- Production Type (either chip placement, buffer, reflow, batch, assembly,
7.4.1.2.3 Processing
This function generates a query to the data base using the Scenario Name as a key and obtains all the inputs from the data base that are listed above in Section 7.4.1.2.2.

7.4.1.2.4 Outputs
The inputs obtained from the data base are passed onto the Simulation Model Configuration and Execution Function.

7.4.1.3 Simulation Model Configuration and Execution

7.4.1.3.1 Introduction
For each scenario being evaluated by the user, this function completely configures sets of (two) ASCII files which drive a Witness simulation model (subsequently, this function also executes the Witness model). In other words, if the user wishes to evaluate various different scenarios, this function creates separate sets (of two) ASCII files for each of the different scenarios. The first ASCII file in each set is used to run the simulation model once and determine the time interval at which the second ASCII file has to be re-read by the model (the file has to be re-read a number of times to run the model iteratively so as to ensure that the model is in a steady state when data is being collected to generate the simulation reports). This time interval is then automatically stored in a variable in the second ASCII file and this variable then governs the time intervals at which this second ASCII file is re-read iteratively.

Note: All time units that are listed in the ASCII file will be in minutes.

7.4.1.3.2 Inputs
Listed below are the required inputs and their sources.

<table>
<thead>
<tr>
<th>Inputs</th>
<th>Sources</th>
</tr>
</thead>
<tbody>
<tr>
<td>• Line ID</td>
<td>LCDR Function</td>
</tr>
<tr>
<td>• Device ID for each device on the line</td>
<td>LCDR Function</td>
</tr>
</tbody>
</table>
Chapter 7

- Quantity of each device
  Default Value
- Order of each device on the line
  LCDR Function
- Scenario name
  PDD Function
- Product List
  PDD Function
- Production Order
  PDD Function
- Time to be released (for each product)
  Default Value
- Production Order File Name
  Default Value
- Routing / Cycle Time File Name
  Default Value
- Lot size by product
  PDD Function

- For Chip Placement machines
  -- Production Type (either chip placement, buffer, reflow, batch, assembly, singulation, test or other)
    PDD Function
  -- Cycle Time
  -- Feeder replenishment time (per feeder)
    FP Data base
  -- Feeder setup time between products
    FP Data base
  -- Mean time between failure
    PDD Function
  -- Mean time of assist
  -- Input Rule
    Default Value
  -- Output Rule
    Default Value
  -- Actions
    Default Value
  -- Reporting
    Default Value
  -- Setup Detail
    * Setup Number
    * Mode
    * Setup Time
    * Description
    * Station Number
    Default Value
  -- Push or Pull Rules

- For Buffers
  -- Production Type (either chip placement, buffer, reflow, batch, assembly, singulation, test or other)
    PDD Function
  -- Capacity
    PDD Function
  -- Input Position
    Default Value
  -- Output Scan From
    Default Value
  -- Select Rules
    Default Value
  -- Actions
    Default Value
  -- Reporting
    Default Value

- For Reflow Ovens (modeled as conveyors)
  -- Production Type (either chip placement, buffer, reflow, batch, assembly,
singulation, test or other)  PDD Function
-- Cycle Time  PDD Function
-- Setup time  PDD Function
-- Capacity (Number of boards)  PDD Function
-- Mean time between failure  PDD Function
-- Mean time of assist  PDD Function
-- Input Rule  Default Value
-- Output Rule  Default Value
-- Setup Number  Default Value
-- Actions  Default Value
-- Reporting  Default Value

* For Batch Machines
-- Production Type (either chip placement, buffer, reflow, batch, assembly, singulation, test or other)  PDD Function
-- Cycle Time  PDD Function
-- Setup time  PDD Function
-- Mean time between failure  PDD Function
-- Mean time of assist  PDD Function
-- Batch min  PDD Function
-- Batch max  PDD Function
-- Input Rule  Default Value
-- Output Rule  Default Value
-- Actions  Default Value
-- Reporting  Default Value

* For Assembly Machines
-- Production Type (either chip placement, buffer, reflow, batch, assembly, singulation, test or other)  PDD Function
-- Cycle Time  PDD Function
-- Setup time  PDD Function
-- Mean time between failure  PDD Function
-- Mean time of assist  PDD Function
-- Assembly Quantity (the number  PDD Function
-- Input Rule  Default Value
-- Output Rule  Default Value
-- Actions  Default Value
-- Reporting  Default Value

* For Singulation Machines
-- Production Type (either chip placement, buffer, reflow, batch, assembly, singulation, test or other)  PDD Function
-- Cycle Time  PDD Function
-- Setup time  PDD Function
-- Mean time between failure  PDD Function
-- Mean time of assist  PDD Function
-- Input Rule  Default Value
-- Output Rule  Default Value
-- Actions  Default Value
-- Reporting  Default Value
Chapter 7

* For Test Machines
  -- Production Type (either chip placement, buffer, reflow, batch, assembly, singulation, test or other)
    PDD Function
  -- Cycle Time
    PDD Function
  -- Setup time
    PDD Function
  -- Mean time between failure
    PDD Function
  -- Mean time of assist
    PDD Function
  -- Yield
    PDD Function
  -- Rework Loop (Rework location)
    PDD Function
  -- Input Rule
    Default Value
  -- Output Rule
    Default Value
  -- Actions
    Default Value
  -- Reporting
    Default Value

* For Other Machines
  -- Production Type (either chip placement, buffer, reflow, batch, assembly, singulation, test or other)
    PDD Function
  -- Cycle Time
    PDD Function
  -- Setup time
    PDD Function
  -- Mean time between failure
    PDD Function
  -- Mean time of assist
    PDD Function
  -- Input Rule
    Default Value
  -- Output Rule
    Default Value
  -- Actions
    Default Value
  -- Reporting
    Default Value

* For attribute definitions
  -- Attribute Name
    Default Value
  -- Quantity
    Default Value
  -- Data Type
    Default Value
  -- Group Number
    Default Value

* For variable definitions
  -- Name
    Default Value
  -- Quantity
    Default Value
  -- Reporting
    Default Value

* For part definitions
  -- Part Name
    PDD Function (same as Product Name)
  -- Type
    Default Value
  -- Actions
    Default Value
    PDD Function (Number of panels/board)
    Default Value
    Default Value (Operation Pointer)
    Defaults to 1
    Default Value (Test Stations Only) -- defaults to zero.
7.4.1.3.3 Processing
This function makes function calls to a series of other functions in order to:

- Generate a Routing/Cycle Time File (see section 7.4.1.12)
- Generate a Production Order File (see section 7.4.1.13)
- Generate two ASCII Files (see section 7.4.1.14)
- Run the Witness Simulation model (see section 7.4.1.15)

7.4.1.3.4 Outputs
The outputs as described in the following sections:
Section 7.4.1.12
Section 7.4.1.13
Section 7.4.1.14
Section 7.4.1.15

7.4.1.4 Line Evaluation Report Configuration

7.4.1.4.1 Introduction
The Line Evaluation Report Configuration Function extracts the results data from the Witness ASCII report and stores this data in the Factory Planning database. This data is later graphically displayed to the user according to the metrics that the user has chosen to evaluate the line/factory. These metrics are listed below: (They are also listed in Section 1.2).

- WHAT IS THE EXPECTED LINE THROUGHPUT?
- Printed Circuit Board statistics for each product
  -- Number entering production
  -- Number shipped
  -- Number scrapped
  -- Number assembled
  -- Number rejected
  -- Average Cycle time (entire production process)
- Device Statistics
  -- Bottleneck machine identification
  -- For each machine
    -- % Busy
    -- % Idle
    -- % Stopped
      * Downstream blocking
      * Setup
      * Machine Error
      * Repair
    -- Number of Operations completed
• WIP Statistics
  -- Average WIP on the line for each type of product (printed circuit board)
  -- WIP level between front end and test
  -- Maximum WIP for a given scenario
  -- Minimum WIP for a given scenario
  -- Which product in the scenario corresponds to the maximum WIP?
  -- Which product in the scenario corresponds to the minimum WIP?

7.4.1.4.2 Inputs
Listed below are the required inputs and their sources.

* Inputs
  • Scenario Name: Task Manager

7.4.1.4.3 Processing
This function calls the Extract Results Data and the Store Report Data in Database functions. (Please refer to Section 7.4.1.16 and 7.4.1.17 below).

7.4.1.4.4 Outputs
Function calls to the Extract Results Data and the Store Report Data in Database functions. (Please refer to Section 7.4.1.16 and 7.4.1.17 below).

7.4.1.5 Oracle Communication Logon

7.4.1.5.1 Introduction
This function is used to log onto the Factory Planning data base and is called by the different functions within the Simulation Driver that retrieve and/or store data in the data base.

7.4.1.5.2 Inputs
There are no inputs (passed to this function as arguments) since the log-in name and password for logging into the data base (for the prototype software application) is predetermined.

7.4.1.5.3 Processing
None.

7.4.1.5.4 Outputs
Error messages to inform the user of a data base logon error if one occurs.
7.4.1.6 Oracle Communication Logoff

7.4.1.6.1 Introduction
This function is used to log off from the Factory Planning data base and is called by the different functions within the Simulation Driver that retrieve and/or store data in the data base.

7.4.1.6.2 Inputs
There are no inputs (passed to this function as arguments).

7.4.1.6.3 Processing
Data base updates will be committed to the data base if no data base logoff errors occur. If a data base logoff error occurs, the data base will be rolled back to its previous state.

7.4.1.6.4 Outputs
Error messages to inform the user of a data base logoff error if one occurs.

7.4.1.7 Integer Conversion to ASCII String

7.4.1.7.1 Introduction
This function converts an integer value to its corresponding ASCII string. It is used when the data storage format in the data base or in a data structure differs from the inherent format of a data element and hence the format of the data element needs to be transformed from an integer to a string format.

7.4.1.7.2 Inputs
Listed below are the required inputs and their sources.

Inputs          Sources
---            ---
• Integer value to be converted to an ASCII string  Function which calls this function

• String name which is to represent the corresponding integer value  Function which calls this function
7.4.1.7.4 Processing
This function converts each digit of the integer value passed to it as an argument into the corresponding character of an ASCII string. This string is terminated by a null character. Since this function processes data for each of the digits in the integer starting from the rightmost digit, the ASCII string created is in reverse order as compared to the initial integer. This string is reversed by a function call to the Reverse String function described in 7.4.1.8 below.

7.4.1.7.4 Outputs
The ASCII string which corresponds to the integer value passed to this function as an argument.

7.4.1.8 Reverse String

7.4.1.8.1 Introduction
This function reverses a string (which is passed to it as an argument). That is, the string "tset", for example, will be reversed to become "test".

7.4.1.8.2 Inputs
Listed below are the required inputs and their sources.

<table>
<thead>
<tr>
<th>Inputs</th>
<th>Sources</th>
</tr>
</thead>
<tbody>
<tr>
<td>* The string which is to be reversed</td>
<td>Function which calls this function</td>
</tr>
</tbody>
</table>

7.4.1.8.3 Processing
This function reverses the order of characters in any string which is passed to it as an argument.

7.4.1.8.4 Outputs
The reversed string as described above.
Chapter 7

7.4.1.9 Retrieve Route Cycle Time Data Structure Information

7.4.1.9.1 Introduction
This Function retrieves from the data base information pertaining to the process route and cycle time for each product in a given scenario and stores this information in the appropriate data structure.

7.4.1.9.2 Inputs
Listed below are the required inputs and their sources.

<table>
<thead>
<tr>
<th>Inputs</th>
<th>Sources</th>
</tr>
</thead>
<tbody>
<tr>
<td>* Board Name</td>
<td>PDD Function</td>
</tr>
<tr>
<td>* Board Side</td>
<td>PDD Function</td>
</tr>
<tr>
<td>* Number of Process Steps (for this board-side combination)</td>
<td>PDD Function</td>
</tr>
<tr>
<td>* Unique Code corresponding to a given board-side combination and the corresponding data set for the process steps and cycle times for that board-side combination</td>
<td>PDD Function</td>
</tr>
</tbody>
</table>

7.4.1.9.3 Processing
This function retrieves the following information from the data base (on the basis of the inputs described above):

- Process Step Number for each process step for this board-side combination
- Device Number for devices included in the process steps for this board-side combination
- Device ID for each device included in the process steps for this board-side combination
- Cycle time for each process step
- Setup time (if any) for each process step
- Feeder setup time (if any) for each process step
- Feeder Replenishment time (if any) for each process step
- Average time between failures (if any) for each process step
- Average time of each failure (if any) for each process step
- Buffer capacity (if applicable)
- Minimum Batch size for batch operations
- Maximum batch size for batch operations
- Assembly quantities for each assembly operation involving sub-assemblies
- Yield value (for test operations)
- Test rework location (for test operations)
This information is thereafter stored in the appropriate data structure for subsequent use by other functions within the Simulation Driver.

### 7.4.1.9.4 Outputs

The values retrieved from the data base as described above.

### 7.4.1.10.1 Comparison of Records (by Production Sequence) in the Product Information Data Structure

#### 7.4.1.10.1 Introduction

This function re-arranges the records in the Product Information data structure in ascending order according to the production sequence corresponding to each record.

#### 7.4.1.10.2 Inputs

Listed below are the required inputs and their sources.

<table>
<thead>
<tr>
<th>Inputs</th>
<th>Sources</th>
</tr>
</thead>
<tbody>
<tr>
<td>Pointer to the Product Information data structure</td>
<td>Function which calls this function</td>
</tr>
</tbody>
</table>

#### 7.4.1.10.3 Processing

This function compares on the basis of the production sequence, the two records that are passed to it as parameters. It is used in conjunction with the qsort function call to arrange the records in the Product Information data structure in ascending order according to the production sequence corresponding to each record.

#### 7.4.1.10.4 Outputs

The re-arranged Product Information data structure as described above.

### 7.4.1.11.1 Comparison of Records (by Process Step) in the Temporary Data Structures

#### 7.4.1.11.1 Introduction

This function re-arranges the records in the temporary data structures in ascending order according to the process step corresponding to each record. These data structures are referred to as temporary because they are created and dissolved each time their encompassing function is called.
7.4.11.2 Inputs
Listed below are the required inputs and their sources.

*Inputs* Source
- Pointer to the temporary data structures Function which calls this function

7.4.11.3 Processing
This function compares on the basis of the process step the two records that are passed to it as parameters, the records of the temporary data structures. It is used in conjunction with the qsort function call to arrange the records in the temporary data structures in ascending order according to the process step corresponding to each record.

7.4.11.4 Outputs
The re-arranged temporary data structures as described above.

7.4.12 Build Routing/Cycle Time File

7.4.12.1 Introduction
This function creates the routing/cycle time file which is essential for generating and running a Witness simulation model for each of the scenarios being evaluated by the user. This file contains the process route information and device cycle time information for each of the products (by board name and board side) that are included in the given scenario.

7.4.12.2 Inputs
Listed below are the required inputs and their sources.

*Inputs* Source
- Scenario Name: Simulation Model

7.4.12.3 Processing
For each scenario being evaluated by the user, the routing/cycle time file contains the process route information and device cycle time information for each of the products (by board name and board side) that are included in the given scenario. The Routing/Cycle Time file will have the following data:
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Part 2 ID</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Part 3 ID</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Part 4 ID</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Part 5 ID</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Part 6 ID</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

The actual file format will be comma delimited as follows:

- Part 1 ID, Machine ID 1, Machine ID 2, Cycle Time 1, ...
- Part 2 ID, Machine ID 1, Machine ID 2, Cycle Time 2, ...
- Part 3 ID, Machine ID 1, Machine ID 2, Cycle Time 3, ...
- Part 4 ID, Machine ID 1, Machine ID 2, Cycle Time 4, ...
- Part 5 ID, Machine ID 1, Machine ID 2, Cycle Time 5, ...
- Part 6 ID, Machine ID 1, Machine ID 2, Cycle Time 6, ...

If part routes vary in the number process types, blank fields need to be populated with "NONE" for Machine ID fields, and "0.0" for Cycle Time fields. Normally, the last step in any process route is "SHIP". If, however, the product is a subassembly to be assembled into a final product, the last step in the route should be the buffer preceding the assembly operation.

### 7.4.1.12.4 Outputs

The Routing/Cycle Time file as described above.

### 7.4.1.13 Build Production Order File

#### 7.4.1.13.1 Introduction

For each of the scenarios being evaluated by the user, a Production Order File is created which depicts the order of production of the different products (by board name and side) that are included in the scenario being evaluated by the user.

#### 7.4.1.13.2 Inputs

Listed below are the required inputs and their sources.

**Inputs**
- Scenario Name: Configuration

**Sources**
- Simulation Model
- and Execution
7.4.1.13.3 Processing

The Production Order File must have the following data:

<table>
<thead>
<tr>
<th>Product Name</th>
<th>Batch Quantity</th>
<th>Time to be Released</th>
</tr>
</thead>
<tbody>
<tr>
<td>Product ID 1</td>
<td>200</td>
<td>0.00</td>
</tr>
<tr>
<td>Product ID 2</td>
<td>150</td>
<td>0.00</td>
</tr>
<tr>
<td>Product ID 3</td>
<td>225</td>
<td>0.00</td>
</tr>
<tr>
<td>Product ID 4</td>
<td>100</td>
<td>0.00</td>
</tr>
</tbody>
</table>

(The numbers used for Batch Quantity above are for illustrative purposes only -- the batch quantity for a given product can be set to whatever the user wishes to simulate). The actual file format will be comma delimited as follows:

<Product ID 1>,<200>,<0.00>
<Product ID 2>,<150>,<0.00>
<Product ID 3>,<225>,<0.00>
<Product ID 4>,<100>,<0.00>

The time to be released is the decimal time equivalent to release the product to the simulation model. Time 0.00 indicates that all products are available to produce at the beginning of the run. If the entry to the simulation is to be delayed, the corresponding decimal time has to be used. For example, if we assume that the simulation time units are minutes, 4 hours into the run is represented by 240.00.

7.4.1.13.4 Outputs

The Production Order File as described above.

7.4.1.14 Build ASCII File

7.4.1.14.1 Introduction

For each scenario being evaluated by the user, this function completely configures an ASCII file, which is used to run the simulation model. The Build ASCII file function is invoked twice for each of the scenarios being evaluated by the user -- the file generated the first time is used to determine the time interval at which the second ASCII file has to be re-read by the model and the file generated the second time is used to actually run the simulation model iteratively.
7.4.1.14.2 Inputs
Listed below are the required inputs and their sources.

<table>
<thead>
<tr>
<th>Inputs</th>
<th>Sources</th>
</tr>
</thead>
<tbody>
<tr>
<td>• Scenario Name:</td>
<td>Simulation Model</td>
</tr>
<tr>
<td></td>
<td>Configuration and Execution</td>
</tr>
<tr>
<td>• Variable indicating whether this is the first time or the second time the ASCII file is being generated for the given scenario.</td>
<td>Simulation Model</td>
</tr>
</tbody>
</table>

7.4.1.14.3 Processing
For each of the scenario names the user wishes to evaluate, this function will generate an ASCII file in the format described below.

If any of the configurable fields are not defined for a particular parameter in the file -- such as the <Setup Number> field for the MACHINE definition -- these fields are simply not included in the file, i.e., the default is to exclude the undefined fields. Comments for particular cases that the file format must adhere to are listed within curly {} brackets.

In order to generate the complete ASCII file to drive the simulation, the various variable names -- indicated in <> brackets -- will have to be automatically assigned to each ASCII file according to the scenario data entered by the user. This functionality is accomplished by the Build ASCII File Function.

Depending upon whether the ASCII file is the first or the second file being generated for a given scenario, the ASCII file format will have to be changed. The following text describes these and other details according to which the ASCII files will have to be (transparently) formatted according to the data sets entered by the user when using the Factory Planning Application.

ASCII File Formats

! WITNESS MODEL: <Scenario Name>
DEFINE

PART: <Product Name 1>, Variable attributes;
    (Product Name in ASCII file <=> Board Name in data base)
PART: <Product Name 2>, Variable attributes;
    •
    •

Development of a Seamlessly Integrated Factory Planning Software Tool (Prototype) to Evaluate and Optimize Surface Mount Manufacturing Lines Page 99
MACHINE: <Machine ID 1>, <Quantity>, <Type>, 0,0,<Setup Number>;
MACHINE: <Machine ID 1>, <Quantity>, <Type>, 0,0,<Setup Number>;

(The above lines must end in a ":"). If the Setup Number is not defined, the preceding "", is replaced by a ":")

CONVEYOR <Conveyor ID 1>, <Quantity>, Queuing, <Capacity in Number of Parts>;
CONVEYOR <Conveyor ID 2>, <Quantity>, Queuing, <Capacity in Number of Parts>;

BUFFER: <Buffer ID 1>, <Quantity>, <Capacity>;
BUFFER: Top,1,10000;
BUFFER: Bottom,1,10000;

(The preceding two buffers are always defined)

ATTRIBUTE: P_Type, l,Integer,1;
ATTRIBUTE: Panel,1,Integer,1;
ATTRIBUTE: S_Time,1,Real,1;
ATTRIBUTE: OP,1,Integer,1;
ATTRIBUTE: Fail, 1,Integer,1;

(The preceding attributes are always defined)

VARIABLE: C_P_Type,1,1,Integer;
VARIABLE: K,1,1,Integer;
VARIABLE: J,1,1,Integer;
VARIABLE: RTE,2,<Number of Products in this scenario>,<Maximum No. of Steps in process route (from Route/Cycle Time file)>,Name;
VARIABLE: Cyc_Time,2,<Number of Products in this scenario>,<Maximum No. of Steps in process route (from Route/Cycle Time file)>,Real;
VARIABLE: T_Sum,1,<Number of Products in this scenario>,Real;
VARIABLE: Count,1,<Number of Products in this scenario>,Integer;
VARIABLE: T_Avg,1,<Number of Products in this scenario>,Real;
VARIABLE: Total_P,1,1,Integer;
VARIABLE: N_Ship,l,l,Integer;
VARIABLE: TPass,1,l,Real; {Must be defined if a Production Type is defined}
VARIABLE: File_RE,1,l,Real;

(The preceding variables are always defined)

FILE: Input_F,Read;
PART_FILE: Schedule, Read;

(The file and part_file are always defined)

DETAIL

SELECT

<PRODUCT NAME 1>

NAME OF PART: <PRODUCT NAME 1>;
TYPE: Variable attributes;
GROUP NUMBER: 1;
MAXIMUM ARRIVALS: 0;

ACTIONS, Create
Add
P_Type = <P_Type>
Panel = <Panel>
OP = 1

End Actions
ACTIONS, Leave
Add
T_Sum (P_Type) = T_Sum (P_Type) + (TIME - S_Time)
Count (P_Type) = Count (P_Type) + 1
T_Avg (P_Type) = T_Sum (P_Type) / Count (P_Type)

End Actions
OUTPUT RULE: Wait;
PART ROUTE: None
REPORTING: Yes;
CONTAINS FLUIDS: No;
SHIFT: Undefined;

END <PRODUCT NAME 1>

<PRODUCT NAME 2>

NAME OF PART: <PRODUCT NAME 2>;
TYPE: Variable attributes;
GROUP NUMBER: 1;
MAXIMUM ARRIVALS: 0;

ACTIONS, Create
Add
P_Type = <P_Type>
Panel = <Panel>
OP = 1

End Actions
ACTIONS, Leave
Add

(The appropriate value of P_Type obtained from the Production Order File -- Product ID 1 has P_Type1, Product ID 2 has P_Type 2, etc.)
(Obtained from the data base)
\[
\begin{align*}
T_{\text{Sum}}(P_{\text{Type}}) &= T_{\text{Sum}}(P_{\text{Type}}) + (TIME - S_{\text{Time}}) \\
\text{Count}(P_{\text{Type}}) &= \text{Count}(P_{\text{Type}}) + 1 \\
T_{\text{Avg}}(P_{\text{Type}}) &= \frac{T_{\text{Sum}}(P_{\text{Type}})}{\text{Count}(P_{\text{Type}})}
\end{align*}
\]

End Actions

OUTPUT RULE: Wait;
PART ROUTE: None
REPORTING: Yes;
CONTAINS FLUIDS: No;
SHIFT: Undefined;

END <Product Name 2>

\[
\begin{align*}
\text{NAME OF MACHINE: } &<\text{Machine ID 1}>; \\
\text{QUANTITY: } &<\text{Quantity}>; \\
\text{TYPE: } &\text{Single}; \\
\text{PRIORITY: } &\text{Undefined}; \\
\text{LABOR: } &\text{Repair: None}
\end{align*}
\]

LABOR: Cycle: None

DISCRETE LINKS: Fill: None

DISCRETE LINKS: Empty: None

CYCLE TIME: Cyc_Time(P_{\text{Type}}, OP);
BREAKDOWNS: No;
ACTIONS, Start
Add
\[
\begin{align*}
C_{\text{P}_{\text{Type}}} &= P_{\text{Type}} \\
N_{\text{Ship}} &= N_{\text{Ship}} + 1 \\
\text{IF } N_{\text{Ship}} &= \text{Total}_P \\
\text{File}_R &= \text{TIME} \\
\text{STOP} \\
\text{ENDIF}
\end{align*}
\]

End Actions
ACTIONS, Finish
Add
\[
\begin{align*}
\text{OP} &= \text{OP} + 1
\end{align*}
\]

End Actions
INPUT RULE: IF NPARTS(Input_P) = 0
PULL from Bottom
ELSEIF NPARTS(Bottom) >= 1 AND Input_P:P_{\text{Type}} <> C_{\text{P}_{\text{Type}}}
PULL From Bottom
ELSE
PULL from Input_P
ENDIF;
OUTPUT RULE: PUSH to RTE(P_{\text{Type}}, OP)
REPORTING: Individual;
SHIFT: Undefined, 0, 0;

END < Machine ID 1>

<Machine ID 2> (This machine data format is for a Batch Machine)

NAME OF MACHINE: <Machine ID 2>
QUANTITY: <Quantity>
TYPE: Batch
* Batch min: <Value>
* Batch max: <Value>
PRIORITY: Undefined
LABOR:
  Repair: None
END
LABOR:
  Cycle: None
END
DISCRETE LINKS:
  Fill: None
END
DISCRETE LINKS:
  Empty: None
END
CYCLE TIME: Cyc_Time (P_Type, OP);
BREAKDOWNS: No;
ACTIONS, Finish
Add
  OP = OP + 1
End Actions
INPUT RULE: <Input Rules>
[See below for the different formats that can be used for the input rules. The appropriate format must be transparently mapped to each machine as is listed in the

INPUT RULE Formats section below.]

OUTPUT RULE: <Output Rules>  (See below for the different formats that can be used for the output rules. The appropriate format must be transparently mapped to each machine as is listed in the

OUTPUT RULE Formats section below.)

REPORTING: Individual;
SHIFT: Undefined, 0, 0;
END < Machine ID 2>

<Machine ID 3> (This machine data format is for an Assembly operation)

NAME OF MACHINE: <Machine ID 3>
QUANTITY: <Quantity>
TYPE: Assembly
* Assembly quantity: <Value>
[This is the number of products that the assembly station pulls from a given buffer at a given time. It is mostly set to 1]
PRIORITY: Undefined
LABOR:
Chapter 7

Repair: None
END
LABOR:
  Cycle: None
END
DISCRETE LINKS:
  Fill: None
END
DISCRETE LINKS:
  Empty: None
END

CYCLE TIME: NORMAL(Cyc_Time (P_Type, OP),<Standard Deviation>,PRN);

(Standard deviation can be .25 * route/cycle time file value -- based on experience of "simulation expert" at different SMT factories across Motorola)

BREAKDOWNS: No;
ACTIONS,
Finish
Add
  OP = OP + 1
End Actions

INPUT RULE: <Input Rules>; (See below for the different formats that can be used for the input rules. The appropriate format must be mapped to each machine as is listed in the INPUT RULE Formats section below.)

OUTPUT RULE: <Output Rules> (See below for the different formats that can be used for the output rules. The appropriate format must be mapped to each machine as is listed in the OUTPUT RULE Formats section below.)

REPORTING: Individual;
SHIFT: Undefined, 0, 0;
END < Machine ID 3>

<Machine ID 4> (This machine data format is for a test operation)

NAME OF MACHINE: <Machine ID 4>;
QUANTITY: <Quantity>;
TYPE: Single;
PRIORITY: Undefined;
LABOR:
  Repair: None
END
LABOR:
  Cycle: None
END
DISCRETE LINKS:
  Fill: None
END
DISCRETE LINKS:
  Empty: None
END

CYCLE TIME: Cyc_Time (P_Type, OP);
BREAKDOWNS: No;
ACTIONS, Finish
Add
IF RANDOM(99) > <Yield Value>
Fail = 1
ELSE
OP = OP + 1
ENDIF
End Actions
INPUT RULE: <Input Rules>;  

OUTPUT RULE: IF Fail = 1
PUSH to <Rework Location>
ELSE
PUSH to RTE (P_TYPE, OP)
ENDIF;
REPORTING: Individual;
SHIFT: Undefined, 0, 0;

END <Machine ID 4>

<Machine ID 5> [This machine data format is for an singulation operation]

NAME OF MACHINE: <Machine ID 5>;
QUANTITY: <Quantity>;
TYPE: Production;
* Part type: Same;
* Production qty: Panel - 1;
PRIORITY: Undefined;
LABOR:
  Repair: None
END
LABOR:
  Cycle: None
END
DISCRETE LINKS:
  Fill: None
END
DISCRETE LINKS:
  Empty: None
END
CYCLE TIME: Cyc_Time (P_Type, OP);
BREAKDOWNS: No;
ACTIONS, Start
Add
  OP = OP + 1
  T_Pass = S_Time
End Actions
ACTIONS, Finish
Add
  S_Time = T_Pass
End Actions
INPUT RULE: <Input Rules>;

OUTPUT RULE: <Output Rules>  

for the output rules. The appropriate format must be mapped to each machine as is listed in the OUTPUT RULE Formats section below.)

REPORTING: Individual;
SHIFT: Undefined, 0, 0;

END < Machine ID 5>

<Machine ID 6> (This machine data format is for any device excluding the above)

NAME OF MACHINE: <Machine ID 6>;
QUANTITY: <Quantity>;
TYPE: Single;
PRIORITY: Undefined;
LABOR:
    Repair: None
END

LABOR:
    Cycle: None
END

DISCRETE LINKS:
    Fill: None
END

DISCRETE LINKS:
    Empty: None
END

SETUPDETAIL (Setup detail can be added as needed. The lines displayed must be inserted, and only setup time included)

* Description: Setup number 1;
* Station number: Station Number 1;
LABOR:
    Setup: None;
END

SETUPDETAIL

CYCLE TIME: Cyc_Time (P_Type, OP);

BREAKDOWNS: Busy Time;
    * Down interval: MIN(NEGEXP(<Mean Value>,<PRN>),<5 * Mean Value>);
    * Repair time: NORMAL(<Mean Value>,<Standard Deviation>,<PRN>);
    * Scrap part: No;
    * Setup on repair: No;

ACTIONS, Finish
Add
    OP = OP + 1
End Actions

INPUT RULE: <Input Rules>;

(See below for the different formats that can be used for the input rules. The appropriate format must be mapped to each machine as is listed in the INPUT RULE Formats section below.)
Chapter 7

OUTPUT RULE: <Output Rules> (See below for the different formats that can be used for the output rules. The appropriate format must be mapped to each machine as is listed in the OUTPUT RULE Formats section below.)

REPORTING: Individual;
SHIFT: Undefined, 0, 0;

END <Machine ID 6>

<Conveyor ID 1> (This conveyor data format is for all reflow ovens)

NAME OF CONVEYOR: <Conveyor ID 1>;
QUANTITY: <Quantity>;
TYPE: Queueing;
PART LENGTH: <Capacity in number of boards>;
MAX CAPACITY: <PART LENGTH>;
INPUT RULE: <Input Rules>; [See below for the different formats that can be used for the input rules. The appropriate format must be mapped to each machine as is listed in the INPUT RULE Formats section below.]

OUTPUT RULE: <Output Rules> (See below for the different formats that can be used for the output rules. The appropriate format must be mapped to each machine as is listed in the OUTPUT RULE Formats section below.)

CYCLE TIME: Reflow Time / PART LENGTH;
BREAKDOWNS: No;
PRIORITY: Undefined;
LABOR: Repair: None;
END
REPORTING: Individual;
SHIFT: Undefined;

END <Conveyor ID 1>

Top (This buffer data format is for buffers that are at the front of the line and that supply the top side of the board to the first machine on the line.)

NAME OF BUFFER: Top;
QUANTITY: <Quantity>;
CAPACITY: 10000;
DELAY TIME: Undefined;
INPUT POSITION: Rear;
OUTPUT SCAN FROM: Front;
* Select: First;
REPORTING: Individual;
SHIFT: Undefined, 0;
END Top

Bottom  

(This buffer data format is for buffers that are at the front of the line and that supply the bottom side of the board to the first machine on the line.)

NAME OF BUFFER: Bottom;  
QUANTITY: <Quantity>;  
CAPACITY: 10000;  
DELAY TIME: Undefined;  
INPUT POSITION: Rear;  
OUTPUT SCAN FROM: Front;  
* Select: First;  
ACTIONS, Out  
Add  
\[ \text{OP} = \text{OP} + 1 \]  
End Actions  
REPORTING: Individual;  
SHIFT: Undefined, 0;  

END Bottom

<BUFFER ID 1>  

(This buffer data format is for buffers that are at the front of the line and that supply the bottom side of the board to the first machine on the line.)

NAME OF BUFFER: <BUFFER ID 1>;  
QUANTITY: <Quantity>;  
CAPACITY: <Capacity>;  
DELAY TIME: Undefined;  
INPUT POSITION: Rear;  
OUTPUT SCAN FROM: Front;  
* Select: First;  
ACTIONS, Out  
Add  
\[ \text{OP} = \text{OP} + 1 \]  
End Actions  
REPORTING: Individual;  
SHIFT: Undefined, 0;  

END <BUFFER ID 1>

Schedule  

(First run only)

NAME OF PART FILE: Schedule;  
FILES ACTUAL NAME: <Production Order File Name>; (Can be full path name)  
TYPE: Read;  
TIME: Absolute;  
RESTART: No;  
START OFFSET: 0.00;  
DELAY: 0.00;  
OUTPUT RULE: PUSH to Top  
SHIFT: Undefined;  

END Schedule;
Chapter 7

Schedule (Second run only!)

NAME OF PART FILE: Schedule;
FILES ACTUAL NAME: <Production Order File Name>; (Can be full path name)
TYPE: Read;
TIME: Absolute;
RESTART: Yes;
START OFFSET: 0.00;
DELAY: <File_RE Value obtained from first run>;
OUTPUT RULE: PUSH to Top
SHIFT: Undefined;

END Schedule;

Input_F (File to read in production route and cycle time)

NAME OF FILE: Input_F
FILES ACTUAL NAME: <Routing/Cycle TimeFile Name>;{Can be full path name}
TYPE: Read;
RESTART: Yes;

End Input_F

END SELECT

END DETAIL

INITIALISE

Add

FOR K = 1 to <Number of products (board names) defined in this scenario>
FOR J = 1 to <Maximum Number of Process Steps defined in Route/Cycle Time file>
READ Input_F RTE (J, K), Cyc_Time (J, K);
NEXT
NEXT
Total_P = ∑ All products(Boards/Panel* Batch Size)>

End Actions

END INITIALISE

(The following statements provide run information for each simulation run)

BATCH TIME, 1000000
REPORTS FILE, <Temporary File> (First run to obtain file re-read time)

BATCH TIME, <Reporting Inverval>
RESET ALL
BATCH TIME, <Reporting Inverval>
REPORTS FILE, <Scenario Name_1(Run Number)>
RESET ALL
BATCH TIME, <Reporting Inverval>
REPORTS FILE, <Scenario Name_2(Run Number)>

(The model will be run for a predetermined amount of time that must be entered by the user. The reports will be reset with no output for the first interval. The BATCH TIME, REPORTS, and RESET commands need to be repeated at least 10 times to produce valid results.)

* ...
* ...
QUIT
Chapter 7

(The ASCII File would end with the "QUIT" statement. Listed below are the possible pre-defined formats for input rules and output rules that can be attributed to any device on the line. The correct format will have to be automatically mapped to each of the devices on the line in accordance with the user-defined scenario data and the guidelines described below.)

**INPUT RULE Formats**

INPUT RULE: Pull From <Buffer ID>;  
(When buffer precedes any device)

INPUT RULE: Wait;  
(Normal Rule)

INPUT RULE: SEQUENCE /Wait <Buffer ID 1> #<Quantity>, <Buffer ID 2> #<Quantity>, • • •  
(Used for Assembly devices)

**OUTPUT RULE Formats**

OUTPUT RULE: PUSH to RTE (P_Type, OP);  
(Normal Output Rule)

OUTPUT RULE: IF Fail = 1  
PUSH to <Rework Location>  
ELSE  
PUSH to RTE (P_TYPE, OP)  
ENDIF;  
(Test device with rework loop)

7.4.1.14.4 Outputs  
The ASCII file as described in the above section.

7.4.1.15 Run Simulation

7.4.1.15.1 Introduction  
This function sends commands to Witness to source each of the ASCII files and run (execute) a Witness simulation model corresponding to each ASCII file.

7.4.1.15.2 Inputs  
Listed below are the required inputs and their sources.

**Inputs**
- Scenario Name:  
- Variable indicating whether this is the first time or the second time the ASCII file is being generated for the given scenario.

**Sources**  
Simulation Model  
Configuration and Execution
7.4.1.15.3 Processing
This function issues a system call to execute the Witness simulation model corresponding to the two ASCII files associated with each scenario.

7.4.1.15.4 Outputs
Systems calls to execute Witness simulation models as described above.

7.4.1.16 Extract Results Data

7.4.1.16.1 Introduction
This function extracts the results data from the ASCII results file generated by Witness and stores this data in corresponding data structures. The data is stored according to the scenario name corresponding to a particular data set.

7.4.1.16.2 Inputs
Listed below are the required inputs and their sources.

<table>
<thead>
<tr>
<th>Inputs</th>
<th>Sources</th>
</tr>
</thead>
<tbody>
<tr>
<td>Scenario Name:</td>
<td>Simulation Model</td>
</tr>
<tr>
<td></td>
<td>Configuration and Execution</td>
</tr>
</tbody>
</table>

7.4.1.16.3 Processing
This function retrieves the results data from the standard Witness ASCII report that corresponds to the entire set of questions that the user can configure from the user interface. Although the length of the Witness report and the number of data fields it contains varies from one scenario to another, the format of each report is the same. Hence, this function retrieves this information from the Witness report file by looking for certain key format identifiers which enable it to automatically determine where the appropriate data fields will be in the Witness results file. Hence, this function is able to extract data from any Witness reports file, regardless of the size of the file and the diversity of data contained within it.

7.4.1.16.4 Outputs
There are no outputs corresponding to this function.
7.4.1.17 Store Report Data in Data Base

7.4.1.17.1 Introduction
This function stores in the data base the results data retrieved by the function described in 7.4.1.16.

7.4.1.17.2 Inputs
Listed below are the required inputs and their sources.

<table>
<thead>
<tr>
<th>Inputs</th>
<th>Sources</th>
</tr>
</thead>
<tbody>
<tr>
<td>Scenario Name:</td>
<td>Simulation Model</td>
</tr>
<tr>
<td></td>
<td>Configuration and Execution</td>
</tr>
</tbody>
</table>

7.4.1.17.4 Processing
This function inserts into the data base the results data which has been stored in data structures as described in section 7.4.1.16. The data is stored in the data base in accordance with the scenario name that corresponds to each of the data sets. If the data base already contains results data corresponding to a particular scenario, this data is deleted from the data base prior to the new data set being inserted.

7.4.1.17.4 Outputs
Queries to the data base to store the results data as described above.

7.4.2 External Interface Requirements

7.4.2.1 User Interfaces
The Simulation Driver will not directly interface with the user. All user inputs will be obtained via the Line Evaluator/Optimizer Interface or via the different data editors and will be stored in the Factory Planning data base.

7.4.2.2 Software Interfaces
The Simulation Driver will interface with the Factory Planning data base, the Task Manager and the Witness Simulation software as described in the above text.
7.4.2.3 Communications Interfaces

To be determined according to the local network protocols that exist in the customer's factory environment.

7.4.3 Design Constraints

- The Simulation Driver must support data source independence. It must be architected to minimize a port to a new data environment (e.g., flat files, different schema).

- The Simulation Driver must not need to access any global variables that are accessed by other software modules of the Factory Planning software application. Data can be passed to the Simulation Driver by other modules (as function calls with arguments).

- The Simulation Driver must be transparent to the user.

- The Simulation Driver must transparently configure and execute a Witness simulation model (provided Witness has been installed in a UNIX environment and the path name for the Witness software is known) for the different types of data sets that the user can configure in a scenario. There will be no need to edit/re-compile the source code to handle different data sets representing the following environments:

  -- A single line

  -- An entire factory consisting of one or more independent lines

  -- An entire factory consisting of many front ends which feed into one or more common back ends

  -- The manufacturing processes for single/double sided boards as well as panelized boards.

  -- The manufacturing processes where two or more boards assembled on the line are eventually assembled into one product prior to shipping.

  -- The manufacturing processes where one side of a given board is produced on one line and the other side is produced on the same or a different line.
Chapter 8

8.0 SOFTWARE REQUIREMENTS (TASK MANAGER)

8.1 Chapter Overview

This chapter provides the reader with a detailed description of the various functionalities that are to be provided by the Task Manager Module of the Factory Planning Application. The different sections of this chapter are written in a format prescribed by the "Software Engineering Standards"—a document compiled by the Institute of Electrical and Electronics Engineers (October 1989).

8.2 Introduction

In order to describe the requirements for the Task Manager in enough detail, the inputs, outputs and data processing associated with each of the functions that comprise the Task Manager module are listed in the sections below.

8.2.1 Purpose

This chapter is intended to describe the software requirements of the Task Manager in enough detail so as to enable the software development team to commence working on the software design and development for this module. The level of detail will be such that the software design and implementation can be carried out independently of development activities of other software modules that make up the Factory Planning Application.

8.2.2 Scope

The Task Manager will consist of one main function (the main routine) and two supporting functions. The main function will issue execution commands (with arguments) to the different software modules (such as the Simulation Driver) that have to be executed to answer the questions configured by the user. The software modules will use the arguments to retrieve the appropriate information from the data base.

In this manner, the Task Manager will enable the user to evaluate and optimize a line by transparently driving various software modules to provide solutions to the user's specific inputs.
8.2.3 Definitions, Acronyms and Abbreviations

<table>
<thead>
<tr>
<th>Acronym</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>LEO</td>
<td>Line Evaluator/Optimizer Interface</td>
</tr>
<tr>
<td>SE</td>
<td>Software Execution Function (main routine)</td>
</tr>
</tbody>
</table>

8.2.4 References


8.2.5 Overview

The remainder of this chapter is organized into two sections. The first one is titled "General Description" and describes the factors that impact the Task Manager software requirements. The second section is titled "Specific Requirements" and is the most important section in this chapter. This section includes a detailed description of the inputs, outputs and data processing associated with each of the functions in the Task Manager.

This chapter is organized on the basis of the IEEE Guide to Software Requirements Specifications.

8.3. General Description

8.3.1 Product Perspective

The Task Manager is a module of the Integrated Factory Planning Software application which enables the user to transparently execute different software modules to perform line evaluation or optimization tasks. The Task Manager automatically executes various software modules according to the inputs made by the user via the LEO Interface.

(For a complete list of all the software functions and modules included in the Integrated Factory Planning software, please refer to the following chapters:)

- Integrated Factory Planning Software Requirements (Chapter 5)
- Integrated Factory Planning Software Architecture (Chapter 6)
8.3.2 Product Functions

**Software Execution Function (the main routine)**
This function sends execution commands (with argument lists) to different software modules of the Factory Planning Application. It is transparent to the user and executes these modules according to the inputs made by the user from the LEO Interface. The different software modules use the argument list passed to them by this function to retrieve appropriate data from the data base.

**Count Scenarios Function**
This function counts the total number of scenarios listed in the “scenario_info” file which is created by the interface in accordance with the inputs made by the user. The “scenario_info” file is transparent to the user.

**Read Scenarios Function**
This function reads the scenario information (Scenario Name, Factory ID and Line ID) from the “scenario_info” file and loads this information into the “scenario_data” data structure.

8.3.3 User Characteristics

The Task Manager will be transparently accessed by the user when the Factory Planning Application is used to evaluate or optimize a given line. The user will not have to know anything about the inputs and outputs associated with the Task Manager since it will be invisible to the user.

8.3.4 General Constraints

The Task Manager will be accessed by the user in a transparent manner. The user will not have to be concerned with any of the inputs and outputs associated with the different functions that make up the Task Manager. These will be configured automatically according to the choices made by the user from the Line Evaluation and Optimization Interface.
8.3.5 Assumptions and Dependencies

The Task Manager will run on UNIX.

8.4 Specific Requirements

This section describes details required to complete the software design and code generation for the Task Manager. Section 8.4.1 describes the functional requirements by individual function. Section 8.4.2 describes the external interface requirements of the software and Section 8.4.3 lists some preliminary design constraints.

8.4.1 Functional Requirements

8.4.1.1 Software Execution Function

8.4.1.1.1 Introduction

The Software Execution Function (main routine) passes execution messages and parameters to different functions within different modules of the Factory Planning software application and causes the functions within these modules to execute on the basis of these parameters.

8.4.1.1.2 Inputs

Listed below are the required inputs and their sources.

Inputs

<table>
<thead>
<tr>
<th>Inputs</th>
<th>Sources</th>
</tr>
</thead>
<tbody>
<tr>
<td>Auto vs Manual Board Assignment to Lines</td>
<td>User/LEO Interface</td>
</tr>
<tr>
<td>Auto vs Manual Board Family Generation</td>
<td>User/LEO Interface</td>
</tr>
<tr>
<td>Auto vs Manual Board Sequencing</td>
<td>User/LEO Interface</td>
</tr>
<tr>
<td>Scenario Name</td>
<td>User/LEO Interface</td>
</tr>
<tr>
<td>Factory ID</td>
<td>User/LEO Interface</td>
</tr>
<tr>
<td>Line ID</td>
<td>User/LEO Interface</td>
</tr>
<tr>
<td>Line Evaluation Report Configuration ID</td>
<td>User/LEO Interface</td>
</tr>
</tbody>
</table>

8.4.1.1.3 Processing

For each of the first three inputs described in 8.4.1.1.2, this function decides whether or not a software module (and corresponding functions) needs to be executed. When the options chosen by the user (from the LEO Interface) are such that two or more software
modules (other than the Task Manager) have to be executed, the Software Execution Function will execute these modules in the following order:

1) Board Assignment to Lines
2) Board Family Generation
3) Board Sequencing
4) Placement Sequence Optimizer
5) Simulation Driver
6) Line Evaluation Report Generator

The inputs described in 8.4.1.1.2 are passed to various software modules as follows:
Whenever the user chooses to automatically assign boards to lines from the Line Evaluation Optimization Interface, this function issues a command to execute the Board to Line Assignment software and includes the Factory ID and Line ID as parameters. The Board Assignments to Lines software uses the Factory ID and Line ID as keys and generates a query to retrieve the values of required variables from the data base.

Whenever the user chooses to automatically generate board families for a given set of lines in a factory, this function issues a command to execute the Board Family Generation software and includes the Factory ID and Line ID as parameters. The Board Family Generation software uses the Factory ID and Line IDs as keys and generates a query to retrieve the values of required variables from the data base.

Whenever the user chooses to automatically sequence boards for a given line in the factory, this function issues a command to execute the Board Sequencing software and includes the Factory ID, Line ID and Product ID as parameters. The Board Sequencing software uses the Factory ID, Line ID and Product IDs as keys and generates a query to retrieve the values of required variables from the data base.

Whenever the user wants to evaluate different scenarios, the LEO Interface iteratively executes the following functions within the Simulation Driver for each of the scenarios being evaluated:

- Line Configuration Data Reading
- Production Data Definition
- Simulation Model Configuration and Execution
- Line Evaluation Report Configuration
The Scenario Name, Factory ID and Line ID are used as keys by the Line Configuration and Data Reading Function in the Simulation Driver to generate a query to the data base and retrieve the required data elements (see Section 8.4.1.1).

The Scenario Name is used as a key by the Production Data Definition Function in the Simulation Driver to generate a query to the data base and retrieve the required data elements (see Section 8.4.1.2).

The Scenario Name is used by the Simulation Model Configuration and Execution function in the Simulation Driver to assign the appropriate names to the different files that are created by the Simulation Model Configuration and Execution function and are required to completely configure and execute a Witness model.

The Scenario Name is used by the Line Evaluation Report Configuration Function in the Simulation Driver to store in the data base the simulation results data obtained from the ASCII results file created by the Witness simulation model. (The Line Evaluation Report Configuration Function first calls the Extract Results Data function which retrieves the appropriate results data from the ASCII file -- see Section 8.4.1.4).

The Software Execution Function also calls the Count Scenarios and Read Scenarios Functions within the Task Manager. The Count Scenarios function is called with the scenario file ("scenario_info") as an argument and the Read Scenarios function is called with two arguments: the number of scenarios and the name of the scenario file.

8.4.1.4 Outputs
The outputs are the execution commands (with respective parameters) to the various functions within the different software modules as described above. In addition, if any of the functions called by the Software Execution Function (such as those within the Task Manager, Simulation Driver and other software modules) do not successfully execute, this function prints an error message to the screen along with a unique error code corresponding to the unique place in the source code where the error was encountered. This provision helps future trouble shooting efforts, should they arise.
8.4.1.2 Count Scenarios Function

8.4.1.2.1 Introduction
The Count Scenarios function opens the “scenario_info” file and determines the total number of scenarios being compared by the user.

8.4.1.2.2 Inputs
The name of the scenario file is passed to this function as input. Since this file is transparent to the user, the name is fixed (“scenario_info”).

8.4.1.2.3 Processing
This function opens the scenario file and counts the number of lines in this file (there is one line of information for each scenario selected by the user).

8.4.1.2.4 Outputs
If this function executes successfully, the output is the number of scenarios counted. If there is an error in execution, an error value is returned instead.

8.4.1.3 Read Scenarios Function

8.4.1.3.1 Introduction
The Read Scenarios function loads the information contained in the scenario file (“scenario_info”) into the “scenario_data” data structure. For each scenario, this information consists of the Scenario Name, Factory ID and Line ID).

8.4.1.3.2 Inputs
Listed below are the required inputs and their sources.

Inputs  Sources
* Scenario file name         SE Function
* Number of scenarios       SE Function

8.4.1.3.3 Processing
This function first allocates enough memory to store all the scenario information entered by the user. It then opens the scenario file, reads in the information created according to the selections made by the user from the user interface and stores this information in the “scenario_data” data structure.
8.4.1.3.4 Outputs
This function returns an error value if it does not execute successfully. Otherwise it returns a value of zero.

8.4.2 External Interface Requirements

8.4.2.1 User Interfaces

The Task Manager will not directly interface with the user. All user inputs will be obtained via the LEO Interface and will be input transparently to the Task Manager.

8.4.2.2 Software Interfaces

The Task Manager will Interface with the following software modules:

- LEO Interface
- Board Assignment to Lines
- Board Family Generation
- Board Sequencing
- Placement Sequence Optimizer
- Simulation Driver
- Line Evaluation Report Generator

For specific information pertaining to the data flows between the Task Manager and each of the software modules, please refer to Chapter 6.

8.4.2.3 Communications Interfaces

To be determined according to the local network protocols that exist in the customer’s factory environment.

8.4.3 Design Constraints

The Task Manager must be able to drive the various software modules described above without incurring any significant delays due to transmission of data/messages to the different modules. Furthermore, when more than one software module is executed by the Task Manager, all data base transactions that occur as a result of execution of any one
module must be completed before the next module is executed to ensure data correctness and avoid data corruption.
Chapter 9

9.0 SOFTWARE REQUIREMENTS (USER INTERFACE)

9.1 Chapter Overview

This chapter describes the various functionalities that are to be provided by the User Interface for the Factory Planning Application. The different sections of this chapter are written in a format prescribed by the "Software Engineering Standards" -- a chapter compiled by the Institute of Electrical and Electronics Engineers (October 1989).

9.2 Introduction

In order to address the software requirements for the User Interface module, the inputs, outputs and data processing associated with each of the functions that comprise the User Interface module is described in the following sections.

9.2.1 Purpose

This chapter is intended to describe the software requirements of the User Interface in enough detail so as to enable the software development team to commence working on the software design and development for this module. The level of detail will be such that the software design and implementation can be carried out independently of development activities of other software modules that make up the Factory Planning Application.

9.2.2 Scope

The User Interface will provide the link between the users and the software modules of the Factory Planning Application that require input from or output to the users. The User Interface will include the following interface modules:

- Main User Interface Module
- Line Evaluator/Optimizer Module
- Line Evaluation Report Generator Module
- Editor Module
Chapter 9

9.2.3 Definitions, Acronyms and Abbreviations

FPA Factory Planning Application
LEO Line Evaluator/Optimizer Module
LERG Line Evaluation Report Generator Module
MUI Main User Interface Module

9.2.4 References


9.2.5 Overview

The remainder of this chapter is organized into two sections. The first one is titled “General Description” and describes the factors that impact the User Interface software requirements. The second section is titled “Specific Requirements” and is the most important section in this chapter. This section includes a detailed description of the inputs, outputs and data processing associated with each of the modules in the User Interface.

This chapter is organized on the basis of the IEEE Guide to Software Requirements Specifications.

9.3 General Description

9.3.1 Product Perspective

The User Interface is a module of the Integrated Factory Planning Software application and enables the user to navigate between, provide input to and receive output from the functions requiring user interaction.

For a complete list of all the software functions and modules included in the Integrated Factory Planning software, refer to the following chapters:

- Integrated Factory Planning Software Requirements (Chapter 5)
- Integrated Factory Planning Software Architecture (Chapter 6)
9.3.2 Product Modules, Screens and Their Functions

- Main User Interface Module
- Line Evaluator/Optimizer Module
- Line Evaluation Report Generator Module
- Editor Module

Main User Interface Module

This module has the following screen:

Main Menu
This screen allows the user to access different functions and data editors associated with the Factory Planning software. In cases where the user must follow a specific sequence between screens, this screen allows the user to only access the first screen of a given sequence and does not allow the user to access any other screen within the sequence.

Line Evaluator/Optimizer Module

This module has the following screen:

Scenario and Report Selection Screen
This screen enables the user to select one or more scenarios to be compared and the Report ID against which these scenarios are to be evaluated. (Each Report ID corresponds to a particular configuration of the available metrics against which different scenarios are compared). If the user wishes to use a new Report ID to configure a new report for a given selection of scenarios (instead of using an existing Report ID retrieved from the data base), this screen provides the user with this option as well.

Line Evaluation Report Generator Module

This module has the following screens:

Line Throughput Results Screen
This screen displays line throughput results for the selected scenarios.
Product Related Results Screen
This screen displays product related results for each scenario (% of product A shipped in Scenario 1, % of product B shipped in Scenario 1, % of product A shipped in Scenario 2, % of product B shipped in Scenario 2, etc.)

Machine Related Results Screen
This screen depicts the machine related results for each scenario (% busy, % idle, % waiting, etc. for each machine on the line for each scenario).

WIP Related Results Screen
This screen depicts the WIP related results for each product in each scenario (Max/Min WIP for product A in Scenario 1, Avg. WIP for product A in Scenario 1, Max/Min WIP for product B in Scenario 1, Avg. WIP for product B in Scenario 1, Max/Min WIP for product A in Scenario 2, Avg. WIP for product A in Scenario 2, Max/Min WIP for product B in Scenario 2, Avg. WIP for product B in Scenario 2, etc.)

Editor Module
This module has the following screens:

Scenario Selection Screen
This screen allows the user to either select a scenario from amongst a list of scenarios contained in the data base or it allows the user to select a new scenario.

Scenario Data Editor Screen
This screen allows the user to edit data contained in a particular scenario.

Report Selection Screen
This screen allows the user to either select a Report ID from amongst a list of Report IDs contained in the data base or it allows the user to create a new Report ID.

Line Evaluation Report Configuration Editor Screen
This screen shows the user the metrics that correspond to a given Report ID. If the user is configuring a set of metrics corresponding to a new Report ID, this screen enables the user to select the metrics that will be associated with the new Report ID.
Chapter 9

Line Layout Editor Screen
This screen allows the user to graphically define or alter the layout for a particular line and also determine the process routes for the different products that are made on that line.

Board Assignment to Lines Editor Screen
This screen allows the user to assign a list of boards to a given line.

Board Family Setup Editor Screen
This screen allows the user to assign a group of boards (that have been assigned to a particular line) to the same setup, thereby generating a board family having the same setup.

Board Sequencing Editor Screen
This screen enables the user to generate a sequence in which different board types (belonging to the same board family and across board families) are to be produced.

9.3.3 User Characteristics

The User Interface will be accessed by all people who want to use the Integrated Factory Planning Tool.

9.3.4 General Constraints

The User Interface will be architected in such a manner so as to avoid redundant data entry by the user. If data input by the user in one screen is also required in another screen which the user invokes subsequently, this data will be copied to the subsequent screen and the user will not have to re-enter the data.

In cases where the user must follow a particular sequence when going from one screen to another, the User Interface will enforce this sequence and will not allow the user to access the given set of screens in any order other than the required sequence.
Chapter 9

9.3.5 Assumptions and Dependencies

The User Interface will run on UNIX in an X Windows environment. This environment will allow for bit-mapped graphical presentation of information on a monitor. The present assumption is that all screens will be developed using Tcl Tk.

9.4 Specific Requirements

This section describes details required to complete the software design and code generation for the User Interface. Section 9.4.1 describes the functional requirements by individual function. Section 9.4.2 describes the external interface requirements of the software. The last two sections respectively list design constraints that are currently known and miscellaneous requirements.

9.4.1 Functional Requirements

9.4.1.1 Main User Interface Module

This module has only one screen -- the Main Menu Screen.

9.4.1.1.1 Main Menu Screen

9.4.1.1.1.1 Introduction

The Main Menu screen displays a list of options for the user to choose from. From this screen, the user can access the different software tools, data editors and the scenario editor contained in the Factory Planning software. This screen is mouse driven. The user only has to point to the option and click the mouse button to execute the option. The options will be displayed by pull down menus and are listed below:

File
* Exit Application

Tools
* Line Evaluator / Optimizer
  Cascade Menu
  -- Scenario and Report Selection
  -- Line Evaluation Report Configuration (this option is greyed out -- see 9.4.1.1.1.3)

* Tool 2
Editors

* Data Editors

    Cascade Menu
    -- Line Layout Data
    -- Board Assignment to Lines Data
    -- Board Family Setup Data
    -- Board Sequencing Data
    -- Line Evaluation Report Configuration Data

* Scenario Editor

A schematic of the Main Menu is depicted in Figure 10 below.
Development of a Seamlessly Integrated Factory Planning Software Tool (Prototype) to Evaluate and Optimize Surface Mount Manufacturing Lines
9.4.1.1.1.2 Inputs
The inputs are received from the user and are mouse-clicks on any of the options depicted above in Section 9.4.1.1.1.

9.4.1.1.1.3 Processing
When the user selects any of the options described in 9.4.1.1.1 above, the options are processed as follows:

<table>
<thead>
<tr>
<th>Option</th>
<th>Processing</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>File</strong></td>
<td></td>
</tr>
<tr>
<td>• Exit Application</td>
<td>Exits FPA</td>
</tr>
<tr>
<td><strong>Tools</strong></td>
<td></td>
</tr>
<tr>
<td>• Line Evaluator / Optimizer</td>
<td></td>
</tr>
<tr>
<td>* Cascade Menu</td>
<td></td>
</tr>
<tr>
<td>-- Scenario and Report Selection</td>
<td>Display Scenario and Report Selection Screen</td>
</tr>
<tr>
<td>-- Line Evaluation Report Configuration</td>
<td>Greyed out i.e., not accessible from this level. The user must first enter the Scenario and Report Selection Screen and select at least one scenario and one Report ID (or a new Report ID) before being able to access the Line Evaluation and Report Configuration Screen</td>
</tr>
<tr>
<td><strong>Editors</strong></td>
<td></td>
</tr>
<tr>
<td>• Data Editors</td>
<td></td>
</tr>
<tr>
<td>* Cascade Menu</td>
<td></td>
</tr>
<tr>
<td>-- Line Layout Data</td>
<td>Display Line Layout Editor Screen</td>
</tr>
<tr>
<td>-- Board Assignment to Lines Data</td>
<td>Display Board Assignment to Lines Editor Screen</td>
</tr>
<tr>
<td>-- Board Family Setup Data</td>
<td>Display Board Family Setup Editor Screen</td>
</tr>
<tr>
<td>-- Board Sequencing Data</td>
<td>Display Board Sequencing Editor Screen</td>
</tr>
<tr>
<td>-- Line Evaluation Report Configuration Data</td>
<td>Display Report Selection Screen followed by appropriate Line Evaluation Report Configuration Editor Screen</td>
</tr>
<tr>
<td><strong>Scenario Editor</strong></td>
<td>Display Scenario Selection Screen</td>
</tr>
</tbody>
</table>
9.4.1.1.4 Outputs

Messages/commands to carry out the appropriate processing as described above in Section 9.4.1.1.3.

9.4.1.2 Line Evaluator/Optimizer Module

This module has one screen -- the Scenario and Report Selection Screen.

9.4.1.2.1 Scenario and Report Selection Screen

9.4.1.2.1.1 Introduction

The Scenario and Report Selection Screen enables the user to select one or more scenarios and the Report ID against which they are to be compared. If the user does not wish to select a Report ID from amongst those existing in the database, but wants to create a new report configuration, this screen provides the user with this option. This screen is mouse driven. The user only has to point to the option and click the mouse button to execute the option.

The options available to the user are:

Select Scenarios:
* Scenario 1
* Scenario 2

Select a Report ID:
* NEW
* Report ID 1
* Report ID 2

* Cancel

* OK

A schematic of the Scenario and Report Selection Screen is depicted in Figure 11 below.
Figure 11   Scenario and Report Selection Screen
9.4.1.2.1.2 Inputs
The inputs are received from the user and are mouse-clicks on one or more scenarios and a click on one Report ID (or NEW) -- please refer to 9.4.1.2.1.1 above.

9.4.1.2.1.3 Processing
For each of the mouse-clicks the user makes within the scenario selection and report selection boxes (please refer to schematic in Figure 11), the row within the corresponding box is highlighted. (Hence, the user can view his/her selection of Scenarios and the Report ID by looking at the highlighted rows in the Scenario selection and Report ID box respectively.)

For the other options on this screen the processing is as follows:

<table>
<thead>
<tr>
<th>Option</th>
<th>Processing</th>
</tr>
</thead>
<tbody>
<tr>
<td>* Cancel</td>
<td>Display Main Menu Screen</td>
</tr>
<tr>
<td>* OK</td>
<td>Display Line Evaluation Report</td>
</tr>
<tr>
<td></td>
<td>Configuration Editor Screen</td>
</tr>
</tbody>
</table>

9.4.1.2.1.4 Outputs
Messages/commands to carry out the appropriate processing as described above in Section 9.4.1.2.1.3.

9.4.1.3 Line Evaluation Report Generator Module
This module has the following screens:

* Line Throughput Results Screen
* Product Related Results Screen
* Machine Related Results Screen
* WIP Related Results Screen

9.4.1.3.1 Line Throughput Results Screen
9.4.1.3.1.1 Introduction
The Line Throughput Results Screen displays a comparison of the expected line throughput for the different scenarios being evaluated. The throughput results are depicted in terms of the number of products produced/week.
The options available to the user are:

* **File**
  * Save
  * Print
  * Return to Main Menu

* **Results**
  * Line Throughput Results
  * Product Related Results
  * Machine Related Results
  * WIP Related Results

A schematic of the Line Throughput Results Screen is depicted in Figure 12 below.
Figure 12  Line Throughput Results Screen

Development of a Seamlessly Integrated Factory Planning Software Tool (Prototype) to Evaluate and Optimize Surface Mount Manufacturing Lines
9.4.1.3.1.2 Inputs
The user selects from the options described above in 9.4.1.3.1.1 by positioning the mouse over the appropriate option -- please refer to 9.4.1.3.1.1 above.

9.4.1.3.1.3 Processing
For each of the user-provided inputs, the processing is as follows:

<table>
<thead>
<tr>
<th>Option</th>
<th>Processing</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>File</strong></td>
<td></td>
</tr>
<tr>
<td>• Save</td>
<td>Display a pop-up window to enable the user to enter a Session ID under which the results of the current session (i.e. current selection of scenarios) is saved. The session ID can be up to 100 characters long so that the user can save sessions under descriptive names if desired.</td>
</tr>
<tr>
<td>• Print</td>
<td>Print the results (all four categories mentioned in 9.4.1.3.1.1) to a printer.</td>
</tr>
<tr>
<td>• Return to Main Menu</td>
<td>Return to Main Menu.</td>
</tr>
<tr>
<td><strong>Results</strong></td>
<td></td>
</tr>
<tr>
<td>• Line Throughput Results</td>
<td>Display Line Throughput Results Screen</td>
</tr>
<tr>
<td>• Product Related Results</td>
<td>Display Product Related Results Screen</td>
</tr>
<tr>
<td>• Machine Related Results</td>
<td>Display Machine Related Results Screen</td>
</tr>
<tr>
<td>• WIP Related Results</td>
<td>Display WIP Related Results Screen</td>
</tr>
</tbody>
</table>

9.4.1.3.1.4 Outputs
Messages/commands to carry out the appropriate processing as described above in Section 9.4.1.3.1.3.

9.4.1.3.2 Product Related Results Screen
9.4.1.3.2.1 Introduction
The Product Related Results Screen displays a comparison of the product related results for the different scenarios being evaluated. For each product, this screen displays the comparative results (listed below) for each scenario being evaluated. A pop-up window displays the products that are produced in the scenarios being evaluated and for each...
product selected by the user, this screen displays the comparative results of different scenarios.

**Product Related Results**
- The % of boards that are:
  - shipped
  - scrapped
  - assembled
  - rejected
- Average production cycle time.

The options available to the user are:

**File**
- Save
- Print
- Return to Main Menu

**Results**
- Line Throughput Results
- Product Related Results
- Machine Related Results
- WIP Related Results

A schematic of the Product Related Results Screen is depicted in Figure 13 below.
Figure 13  Product Related Results Screen

Development of a Seamlessly Integrated Factory Planning Software Tool (Prototype) to Evaluate and Optimize Surface Mount Manufacturing Lines
Chapter 9

9.4.1.3.2.2 Inputs

The user selects from the File and Results options described above in 9.4.1.3.2.1 by positioning the mouse over the appropriate option -- please refer to 9.4.1.3.2.1 above.

The user selects the product names from the pop-up window described in 9.4.1.3.2.1 by positioning the mouse over each product name and clicking once.

9.4.1.3.2.3 Processing

For each of the user-provided inputs, the processing is as follows:

<table>
<thead>
<tr>
<th>Option</th>
<th>Processing</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>File</strong></td>
<td><strong>Processing</strong></td>
</tr>
<tr>
<td>* Save</td>
<td>Display a pop-up window to enable the user to enter a Session ID under which the results of the current session (i.e. current selection of scenarios) is saved. The session ID can be up to 100 characters long so that the user can save sessions under descriptive names if desired.</td>
</tr>
<tr>
<td>* Print</td>
<td>Print the results (all four categories mentioned in 9.4.1.3.2.1) to a printer.</td>
</tr>
<tr>
<td>* Return to Main Menu</td>
<td>Return to Main Menu.</td>
</tr>
<tr>
<td><strong>Results</strong></td>
<td></td>
</tr>
<tr>
<td>* Line Throughput Results</td>
<td>Display Line Throughput Results Screen</td>
</tr>
<tr>
<td>* Product Related Results</td>
<td>Display Product Related Results Screen</td>
</tr>
<tr>
<td>* Machine Related Results</td>
<td>Display Machine Related Results Screen</td>
</tr>
<tr>
<td>* WIP Related Results</td>
<td>Display WIP Related Results Screen</td>
</tr>
</tbody>
</table>

When the user selects a product name from the pop-up box described in 9.4.1.3.2.1, the product related results listed in 9.4.1.3.2.1 are displayed on the screen.

9.4.1.3.2.4 Outputs

Messages/commands to carry out the appropriate processing as described above in Section 9.4.1.3.2.3.
9.4.1.3.3 Machine Related Results Screen

9.4.1.3.3.1 Introduction
The Machine Related Results Screen displays a comparison of the machine related results for the different scenarios being evaluated. For each machine in each scenario, this screen displays the comparative results (listed below) for each scenario being evaluated. A pop-up window displays the Machine IDs that are used in each of the scenarios being evaluated and for each Machine ID selected by the user, this screen displays the comparative results of different scenarios.

Machine Related Results

- The % of time that the machine is:
  -- busy
  -- idle
  -- stopped

- What % of the idle time is due to:
  -- Downstream blocking
  -- Setup
  -- Machine Error
  -- Repair

- Which machine will be the bottleneck on the line (in terms of processing time)

- How many operations will each machine complete

The options available to the user are:

File
- Save
- Print
- Return to Main Menu

Results
- Line Throughput Results
- Product Related Results
- Machine Related Results
- WIP Related Results

A schematic of the Machine Related Results Screen is depicted in Figure 14 below.
Figure 14  Machine Related Results Screen

Development of a Seamlessly Integrated Factory Planning Software Tool (Prototype) to Evaluate and Optimize Surface Mount Manufacturing Lines
9.4.1.3.3.2 Inputs
The user selects from the File and Results options described above in 9.4.1.3.3.1 by positioning the mouse over the appropriate option -- please refer to 9.4.1.3.3.1 above. The user selects the Machine IDs from the pop-up window described in 9.4.1.3.3.1 by positioning the mouse over each Machine ID and clicking once.

9.4.1.3.3.3 Processing
For each of the user-provided inputs, the processing is as follows:

<table>
<thead>
<tr>
<th>Option</th>
<th>Processing</th>
</tr>
</thead>
<tbody>
<tr>
<td>File</td>
<td>Display a pop-up window to enable the user to enter a Session ID under which the results of the current session (i.e. current selection of scenarios) is saved. The session ID can be up to 100 characters long so that the user can save sessions under descriptive names if desired.</td>
</tr>
<tr>
<td>* Save</td>
<td>Print the results (all four categories mentioned in 9.4.1.3.3.1) to a printer.</td>
</tr>
<tr>
<td>* Print</td>
<td>Return to Main Menu.</td>
</tr>
<tr>
<td>Results</td>
<td>Display Line Throughput Results Screen</td>
</tr>
<tr>
<td>* Line Throughput Results</td>
<td>Display Product Related Results Screen</td>
</tr>
<tr>
<td>* Product Related Results</td>
<td>Display Machine Related Results Screen</td>
</tr>
<tr>
<td>* Machine Related Results</td>
<td>Display WIP Related Results Screen</td>
</tr>
<tr>
<td>* WIP Related Results</td>
<td>When the user selects a Machine ID from the pop-up box described in 9.4.1.3.3.1, the machine related results listed in 9.4.1.3.3.1 are displayed on the screen.</td>
</tr>
</tbody>
</table>

9.4.1.3.3.4 Outputs
Messages/commands to carry out the appropriate processing as described above in Section 9.4.1.3.3.3.
9.4.1.3.4 WIP Related Results Screen

9.4.1.3.4.1 Introduction
The WIP Related Results Screen displays a comparison of the WIP related results for the different scenarios being evaluated. For each product in each scenario, this screen displays the comparative WIP related results (listed below) for each scenario being evaluated. A pop-up window displays the product names that are used in each of the scenarios being evaluated and for each product name selected by the user, this screen displays the WIP related results of different scenarios.

WIP Related Results

- WIP level between de-panelization and test
- Maximum WIP
  -- Device at which maximum WIP is reached
- Minimum WIP
  -- Device at which minimum WIP is reached

The options available to the user are:

File
- Save
- Print
- Return to Main Menu

Results
- Line Throughput Results
- Product Related Results
- Machine Related Results
- WIP Related Results

A schematic of the WIP Related Results Screen is depicted in Figure 15 below.
Figure 15

WIP Related Results Screen
9.4.1.3.4.2 Inputs
The user selects from the *File* and *Results* options described above in 9.4.1.3.4.1 by positioning the mouse over the appropriate option -- please refer to 9.4.1.3.4.1 above. The user selects the product names from the pop-up window described in 9.4.1.3.4.1 by positioning the mouse over each product name and clicking once.

9.4.1.3.4.3 Processing
For each of the user-provided inputs, the processing is as follows:

<table>
<thead>
<tr>
<th><strong>Option</strong></th>
<th><strong>Processing</strong></th>
</tr>
</thead>
<tbody>
<tr>
<td><em>File</em></td>
<td></td>
</tr>
<tr>
<td><em>Save</em></td>
<td>Display a pop-up window to enable the user to enter a Session ID under which the results of the current session (i.e. current selection of scenarios) is saved. The session ID can be up to 100 characters long so that the user can save sessions under descriptive names if desired.</td>
</tr>
<tr>
<td><em>Print</em></td>
<td>Print the results (all four categories mentioned in 9.4.1.3.4.1) to a printer.</td>
</tr>
<tr>
<td><em>Return to Main Menu</em></td>
<td>Return to Main Menu.</td>
</tr>
<tr>
<td><em>Results</em></td>
<td></td>
</tr>
<tr>
<td><em>Line Throughput Results</em></td>
<td>Display Line Throughput Results Screen</td>
</tr>
<tr>
<td><em>Product Related Results</em></td>
<td>Display Product Related Results Screen</td>
</tr>
<tr>
<td><em>Machine Related Results</em></td>
<td>Display Machine Related Results Screen</td>
</tr>
<tr>
<td><em>WIP Related Results</em></td>
<td>Display WIP Related Results Screen</td>
</tr>
</tbody>
</table>

When the user selects a product name from the pop-up box described in 9.4.1.3.4.1, the WIP related results listed in 9.4.1.3.4.1 are displayed on the screen.

9.4.1.3.4.4 Outputs
Messages/commands to carry out the appropriate processing as described above in Section 9.4.1.3.4.3.
9.4.1.4 Editor Module

This module has the following screens:

- Scenario Selection Screen
- Scenario Data Editor Screen
- Report Selection Screen
- Line Evaluation Report Configuration Editor Screen
- Line Layout Editor Screen
- Board Assignment to Lines Editor Screen
- Board Family Setup Editor Screen
- Board Sequencing Editor Screen

9.4.1.4.1 Scenario Selection Screen

9.4.1.4.1.1 Introduction

The Scenario Selection Screen enables the user to either select one or more scenarios from the database or select a new scenario if the user chooses to define a new scenario. This screen is mouse driven. The user only has to point to the option and click the mouse button to execute the option.

The options available to the user are:

Select Scenarios:
- NEW
- Scenario 1
- Scenario 2
- ...
- ...
- Cancel
- OK

A schematic of the Scenario Selection Screen is depicted in Figure 16 below.
Figure 16  Scenario Selection Screen
Chapter 9

9.4.1.4.2 Inputs
The inputs are received from the user and are mouse-clicks on one or more scenarios (or NEW) -- please refer to 9.4.1.4.1 above.

9.4.1.4.1.3 Processing
For each of the mouse-clicks the user makes within the scenario selection box (please refer to schematic in Figure 16), the row within the corresponding box is highlighted. (Hence, the user can view his/her selection of Scenarios by looking at the highlighted rows in the Scenario selection box.)

For the other options on this screen the processing is as follows:

<table>
<thead>
<tr>
<th>Option</th>
<th>Processing</th>
</tr>
</thead>
<tbody>
<tr>
<td>- Cancel</td>
<td>Display Main Menu Screen</td>
</tr>
<tr>
<td>- OK</td>
<td>If the user chooses an existing scenario from the data base, display the Scenario Data Editor Screen. If the user selects a new scenario, display the Board Assignment to Lines Editor Screen. When the user clicks “OK” on this screen, display the Board Family Setup Editor Screen. When the user clicks “OK” on this screen, display the Board Sequencing Editor Screen.</td>
</tr>
</tbody>
</table>

9.4.1.4.4 Outputs

Messages/commands to carry out the appropriate processing as described above in Section 9.4.1.4.1.3.

9.4.1.4.2 Scenario Data Editor Screen

The specific requirements for this screen were left for the next phase of implementation of the Factory Planning Application. A schematic is depicted in Figure 17 below, but this is subject to change in the implementation of the next phase.
Figure 17  Scenario Data Editor Screen

Development of a Seamlessly Integrated Factory Planning Software Tool (Prototype) to Evaluate and Optimize Surface Mount Manufacturing Lines
Chapter 9

9.4.1.4.3 Report Selection Screen

9.4.1.4.3.1 Introduction

The Report Selection Screen enables the user to either select a Report ID from the data base or select a new Report ID if the user chooses to define a new Report ID. This screen is mouse driven. The user only has to point to the option and click the mouse button to execute the option.

The options available to the user are:

Select a Report ID
- NEW
- Report ID 1
- Report ID 2
- Cancel
- OK

A schematic of the Report Selection Screen is depicted in Figure 18 below.
Figure 18  Report Selection Screen

Development of a Seamlessly Integrated Factory Planning Software Tool (Prototype) to Evaluate and Optimize Surface Mount Manufacturing Lines
9.4.1.4.3.2 Inputs
The inputs are received from the user and are mouse-clicks on one Report ID (or NEW) -- please refer to 9.4.1.4.3.1 above.

9.4.1.4.3.3 Processing
When the user makes a mouse click within the report selection box (please refer to schematic in Figure 18), the corresponding row within the box is highlighted. (Hence, the user can view his/her selected Report ID by looking at the highlighted row in the report selection box.)

For the other options on this screen the processing is as follows:

<table>
<thead>
<tr>
<th>Option</th>
<th>Processing</th>
</tr>
</thead>
<tbody>
<tr>
<td>• Cancel</td>
<td>Display Main Menu Screen</td>
</tr>
<tr>
<td>• OK</td>
<td>If the user chooses an existing Report ID from the data base, display the Line Evaluation Report Configuration Editor Screen with the options corresponding to the given Report ID selected (by coloring in the &quot;buttons&quot; associated with each of the options). If the user selects a new Report ID, display the Line Evaluation Report Configuration Editor Screen with none of the options selected.</td>
</tr>
</tbody>
</table>

9.4.1.4.3.4 Outputs
Messages/commands to carry out the appropriate processing as described above in Section 9.4.1.4.3.3.

9.4.1.4.4 Line Evaluation Report Configuration Editor Screen
9.4.1.4.4.1 Introduction
The Line Evaluation Report Configuration Editor Screen allows the user to select the metrics which are to be associated with a given Report ID. The set of metrics from which the user can select from are as follows:
Chapter 9

• Expected Line Throughput

• Product Related Metrics
  -- How many boards will enter production?
  -- What percentage will be
    -- Shipped  -- Scrapped  -- Assembled  -- Rejected
  -- How much WIP can we expect on the line on the average?
  -- What will the average cycle time be for the production process?

• Machine Related Questions (Front End Only)
  -- Which machine will be the bottleneck?
  -- For each machine, what is the % of time the machine is
    -- Busy  -- Idle  -- Stopped
  -- What % of the idle time is due to
    -- Downstream  -- Setup  -- Machine  -- Repair
    Blocking  Error
  -- How many operations has each of the machines completed?

• WIP Related Questions
  -- What are the WIP levels between front end and test?
  -- What is the maximum WIP for this scenario?
  -- What is the minimum WIP for this scenario?
  -- At which machine/station is the maximum reached?
  -- At which machine/station is the minimum reached?

A schematic of the Line Evaluation Report Configuration Editor Screen is depicted in Figure 19 below.
Select Questions You Wish to Have Answered:

0  WHAT IS THE EXPECTED LINE THROUGHPUT?

Product Related Questions
  o  How many boards will enter production?
    o  What percentage will be
      o  Shipped  o  Scrapped  o  Assembled  o  Rejected
  o  How much WIP can we expect on the line on the average?
  o  What will the average cycle time be for the production process?

Machine Related Questions (Front End Only)
  o  Which machine will be the bottleneck?
  o  For each machine, what is the % of time the machine is
    o  Busy  o  Idle  o  Stopped
    o  What % of the idle time is due to
      o  Downstream  o  Setup  o  Machine  o  Repair
      o  Blocking  o  Error
  o  How many operations has each of the machines completed?

WIP Related Questions
  o  What are the WIP levels between front end and test?
  o  What is the maximum WIP for this scenario?
  o  What is the minimum WIP for this scenario?
  o  At which machine/station is the maximum reached?
  o  At which machine/station is the minimum reached?
9.4.1.4.4.2 Inputs
The user points and clicks with the mouse on the buttons associated with each of the metrics described in 9.4.1.4.4.1.

9.4.1.4.4.3 Processing
For each of the metrics selected by the user, the corresponding button is filled in with a color to indicate that the corresponding metric has been selected. If a given metric is already selected and the user points and clicks the mouse on the button next to it, the button is de-selected (it is filled in with its original color again).

<table>
<thead>
<tr>
<th>Option</th>
<th>Processing</th>
</tr>
</thead>
<tbody>
<tr>
<td>• Cancel</td>
<td>Display Main Menu Screen</td>
</tr>
<tr>
<td>• OK</td>
<td>If the user has entered the Line Evaluation Report Configuration Editor Screen from the Report Selection Screen clicking on the OK button causes the metrics selected by the user to be stored in the database (along with the corresponding Report ID) and then causes the Main Menu to be displayed.</td>
</tr>
<tr>
<td>• Run Simulation</td>
<td>If the user has entered the Report Configuration Editor Screen from the Scenario and Report Selection Screen, the OK button is replaced by the Run Simulation Button. Clicking on this button causes the Software Execution Function within the Task Manager to be executed (see Chapter 8).</td>
</tr>
</tbody>
</table>

9.4.1.4.4.4 Outputs
Messages/commands to carry out the appropriate processing as described above in Section 9.4.1.4.4.3

9.4.1.4.5 Line Layout Editor Screen
9.4.1.4.5.1 Introduction
This screen enables the user to graphically define layout for a given line as well as a process route for each of the products that is made on the given line. The user enters the Line ID for which the layout is being established and then defines a line layout by dragging and dropping icons from a pre-determined list of icons (described below).
The options available to the user are:

- Add
- Delete
- Move

**Pre-Defined Icons**
- Screen Printers
- Chip Shooters
- Odd Parts Placers
- Reflow Ovens
- Manual Stations
- Other Equipment
- Test Stations
- Repair Stations
- Assembly Stations
- Packing Stations
- Buffers
- Templates

**Templates**
- Template 1
- Template 2
- ...
- ...

**View Process Routes by Product (and Side)**
- Product 1 (Side 1)
- Product 1 (Side 2)
- Product 2 (Side 2)
- ...
- ...

**Define Process Routes by Product**
- Product 1 (Side 1)
- Product 1 (Side 2)
- Product 2 (Side 2)
- ...
- ...

- Cancel
- OK

A schematic of the Line Layout Editor Screen is depicted in Figure 20 below.
Figure 20 Line Layout Editor Screen

Development of a Seamlessly Integrated Factory Planning Software Tool (Prototype) to Evaluate and Optimize Surface Mount Manufacturing Lines
9.4.1.4.5.2 Inputs

The user enters the Line ID and then points and clicks on the options described above in 9.4.1.4.5.1 in order to create a layout corresponding to the Line ID.

9.4.1.4.5.3 Processing

For each of the user-provided inputs, the processing is as follows:

<table>
<thead>
<tr>
<th>Option</th>
<th>Processing</th>
</tr>
</thead>
<tbody>
<tr>
<td>Add</td>
<td>Clicking on this button puts the user in “Add” mode so that the user can now add icons to the line by clicking on a given icon (see Pre-defined icon list), dragging the icon to the desired location on the line and dropping the icon (by releasing the mouse button).</td>
</tr>
<tr>
<td>Delete</td>
<td>Clicking on this button puts the user in “Delete” mode so that when the user clicks on one or more icons on the line, these icons are deleted from the line layout.</td>
</tr>
<tr>
<td>Move</td>
<td>Clicking on this button puts the user in “Move” mode so that the user can click on any icon on the line and move it to another location by moving the mouse to the desired location and clicking on the mouse again.</td>
</tr>
</tbody>
</table>

Pre-Defined Icons. These icons are manipulated according to the above descriptions of the “Add”, “Delete” and “Move” options.

- Screen Printers
- Chip Shooters
- Odd Parts Placers
- Reflow Ovens
- Manual Stations
- Other Equipment
- Test Stations
- Repair Stations
- Assembly Stations
- Packing Stations
- Buffers
- Templates

Templates When the user clicks on the Templates option, a list of templates is retrieved from the data base. (A template is a generic configuration.)
of a segment of a line -- e.g. a back end template would correspond to a typical back end configuration of a line at a given factory. If the user selects any one of these templates, the corresponding layout for the template is displayed on the screen. The user has to then move the icons around to prevent overlapping which might occur when a template is retrieved from the data base and displayed on the screen.

* Template 1
* Template 2
  

**View Process Routes by Product (and Side)**

Clicking on this option displays a pop-up window with a list of board names and sides (which are retrieved from the data base) that are produced on the given line. When the user selects a given board name and board side, the corresponding process route for the given line and board name (and board side) is displayed on the screen.

* Product 1 (Side 1)
* Product 1 (Side 2)
* Product 2 (Side 2)
  

**Define Process Routes by Product**

Clicking on this option displays a pop-up window with a list of board names and sides (which are retrieved from the data base) that are produced on the given line. The user selects a given board name and board side, and then points and clicks on the different devices on the line to define the product route for that board name and that board side.
Chapter 9

- Product 1 (Side 1)
- Product 1 (Side 2)
- Product 2 (Side 2)
- 
- 
- Cancel

Return to Main Menu

- OK

Store user-defined line layout and process route information to the data base.

9.4.1.4.5.4 Outputs
Messages/commands to carry out the appropriate processing as described above in Section 9.4.1.4.5.3

9.4.1.4.6 Board Assignment to Lines Editor Screen

9.4.1.4.6.1 Introduction
The Board Assignment to Lines Editor Screen enables the user to assign a number of boards and sides (top, bottom or both sides) to a given line. The user enters the Board to Line Assignment ID, the Line ID and the list of board names and their sides (top, bottom or both).

The options available to the user are:

- Cancel
- OK

A schematic of the Board Assignment to Lines Editor Screen is depicted in Figure 21 below.
Figure 21    Board Assignment to Lines Editor Screen

<table>
<thead>
<tr>
<th>Side</th>
<th>Board Name</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Board Assignment to Lines Editor Screen
9.4.1.4.6.2 Inputs
The inputs are received from the user are text entries as described above in 9.4.1.4.6.1 as well as mouse clicks on either the OK or Cancel buttons.

9.4.1.4.6.3 Processing
For the other options on this screen the processing is as follows:

<table>
<thead>
<tr>
<th>Option</th>
<th>Processing</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cancel</td>
<td>Display Main Menu Screen</td>
</tr>
<tr>
<td>OK</td>
<td>The list of boards (and their corresponding sides), the Line ID and the Board to Line Assignment ID is stored in the database and the Main Menu is displayed thereafter.</td>
</tr>
</tbody>
</table>

9.4.1.4.6.4 Outputs
Queries to store the information (described in 9.4.1.4.6.3) in the database and messages/commands to carry out the appropriate processing as described above in Section 9.4.1.4.6.3.

9.4.1.4.7 Board Family Setup Editor Screen
9.4.1.4.7.1 Introduction
The Board Family Setup Editor Screen enables the user to assign a number of boards (that have already been assigned to a given line -- the user inputs the Board to Line Assignment ID) to a particular Board Family Setup ID. The board names (and their corresponding sides) that are assigned to a particular Board Family Setup ID will be targeted for a common machine setup during production (if the total number of different components for all the boards corresponding to the same Board Family Setup ID is greater than the feeder capacity of the machine, some feeder change outs might be incurred when producing different boards (and board sides) within the same family).

The options available to the user are:

| Option | |
|--------||
| Cancel | |
| OK     | |

A schematic of the Board Family Setup Editor Screen is depicted in Figure 22 below.
Figure 22  Board Family Setup Editor Screen
9.4.1.4.7.2 Inputs
The inputs are received from the user are text entries as described above in 9.4.1.4.7.1 as well as mouse clicks on either the OK or Cancel buttons.

9.4.1.4.7.3 Processing
For the other options on this screen the processing is as follows:

<table>
<thead>
<tr>
<th>Option</th>
<th>Processing</th>
</tr>
</thead>
<tbody>
<tr>
<td>* Cancel</td>
<td>Display Main Menu Screen</td>
</tr>
<tr>
<td>* OK</td>
<td>The list of boards (and their corresponding sides) and the Board Family Setup ID is stored in the database and the Main Menu is displayed thereafter.</td>
</tr>
</tbody>
</table>

9.4.1.4.7.4 Outputs

Queries to store the information (described in 9.4.1.4.7.3) in the database and messages/commands to carry out the appropriate processing as described above in Section 9.4.1.4.7.3.

9.4.1.4.8 Board Sequencing Editor Screen

9.4.1.4.8.1 Introduction
The Board Sequencing Editor Screen enables the user to input the production sequence for a number of board types/names (and their corresponding sides) both within a given board family (corresponding to a particular Board Family Setup ID) and across different board families. The user assigns a Board Sequence ID for the sequence entered through this screen.

The options available to the user are:
- * Cancel
- * OK

A schematic of the Board Sequencing Editor Screen is depicted in Figure 23 below.
Figure 23  Board Sequencing Editor Screen

<table>
<thead>
<tr>
<th>Production Sequence</th>
<th>Board Name</th>
<th>Side</th>
<th>Board Family Setup ID</th>
<th>Board to Line Assignment ID:</th>
<th>Board Sequence ID:</th>
</tr>
</thead>
</table>

Development of a Seamlessly Integrated Factory Planning Software Tool (Prototype) to Evaluate and Optimize Surface Mount Manufacturing Lines
9.4.1.4.8.2 Inputs
The inputs are received from the user are text entries as described above in 9.4.1.4.8.1 as well as mouse clicks on either the OK or Cancel buttons.

9.4.1.4.8.3 Processing
For the other options on this screen the processing is as follows:

<table>
<thead>
<tr>
<th>Option</th>
<th>Processing</th>
</tr>
</thead>
<tbody>
<tr>
<td>• Cancel</td>
<td>Display Main Menu Screen</td>
</tr>
<tr>
<td>• OK</td>
<td>The list of boards, their corresponding sides and Board Family Setup IDs as well as their production sequence as input by the user is stored in the data base and the Main Menu is displayed thereafter.</td>
</tr>
</tbody>
</table>

9.4.1.4.8.4 Outputs
Queries to store the information (described in 9.4.1.4.8.3) in the data base and messages/commands to carry out the appropriate processing as described above in Section 9.4.1.4.8.3.

9.4.2 External Interface Requirements

9.4.2.1 Software Interfaces
The User Interface will interface with the Task Manager and the Factory Planning data base.

9.4.2.2 Communications Interfaces
To be determined according to the local network protocols that exist in the customer’s factory environment.

9.4.3 Design Constraints
The User Interface will have a Motif look and feel and will be developed in Tcl Tk.
Chapter 10

10.0 INTEGRATED DATA MODEL FOR PILOT APPLICATION

10.1 Chapter Overview

This chapter describes the Integrated Data Model for the Factory Planning Prototype Application. The reader is first provided with a schematic of the various entities that make up the data model (referred to as the Business Model -- normalized). Thereafter, each of these entities is broken down in terms of the attributes contained in it and each of these attributes are formally defined.

Subsequently, the reader is provided with a schematic of the de-normalized data model as well as a listing of the de-normalized entity-attribute relationships/definitions. This de-normalized listing essentially constitutes the database schema for the Factory Planning Prototype application.

Please Note:
The Business Model (Normalized) and corresponding Entity Attribute Description was developed by Roger Larson (from Source Consulting Inc.) under the guidance of the author. The business model was subsequently de-normalized by Roger Larson (who also developed the corresponding Entity Attribute Description).
10.2 Business Model (Normalized)

Development of a Seamlessly Integrated Factory Planning Software Tool (Prototype) to Evaluate and Optimize Surface Mount Manufacturing Lines
### 10.2.1 Entity Attribute Descriptions

<table>
<thead>
<tr>
<th>ATTRIBUTE</th>
<th>NAME</th>
<th>BOARD TYPE KEY</th>
<th>BOARD LINE SEQ</th>
<th>BOARD FAMILY KEY</th>
<th>FACTORY</th>
</tr>
</thead>
<tbody>
<tr>
<td>NAME</td>
<td>DESCRIPTIVE NAME FOR A BOARD TYPE</td>
<td>NUMBER</td>
<td>YES</td>
<td>YES</td>
<td>YES</td>
</tr>
<tr>
<td>N FABER</td>
<td>A BRANCH OF A BOARD FAMILY</td>
<td>NUMBER</td>
<td>YES</td>
<td>YES</td>
<td>YES</td>
</tr>
<tr>
<td>BOARD LINE SEQ</td>
<td>A UNIQUE IDENTIFIER FOR EACH INSTANCE OF A BOARD LINE</td>
<td>NUMBER</td>
<td>NO</td>
<td>NO</td>
<td>NO</td>
</tr>
<tr>
<td>BOARD FAMILY KEY</td>
<td>A UNIQUE IDENTIFIER FOR EACH INSTANCE OF A BOARD FAMILY</td>
<td>NUMBER</td>
<td>NO</td>
<td>NO</td>
<td>NO</td>
</tr>
<tr>
<td>FACTORY</td>
<td>A UNIQUE IDENTIFIER FOR EACH INSTANCE OF A FACTORY</td>
<td>NUMBER</td>
<td>NO</td>
<td>NO</td>
<td>NO</td>
</tr>
</tbody>
</table>

**Table 3** Entity Attribute Descriptions (Business Model)

Development of a Seamlessly Integrated Factory Planning Software Tool (Prototype) to Evaluate and Optimize Surface Mount Manufacturing Lines  
Page 170
<table>
<thead>
<tr>
<th>ID</th>
<th>Description</th>
<th>Type</th>
<th>Size</th>
<th>Null</th>
</tr>
</thead>
<tbody>
<tr>
<td>LINE</td>
<td>A CRYPTOGRAPHIC NAME ASSOCIATED WITH A FACTORY</td>
<td>CHAR</td>
<td>10</td>
<td>NO</td>
</tr>
<tr>
<td>KEY</td>
<td>A unique identifier for each instance of a LINE</td>
<td>NUMBER</td>
<td>4</td>
<td>NO</td>
</tr>
<tr>
<td>ID</td>
<td>A CRYPTOGRAPHIC NAME ASSOCIATED WITH A LINE</td>
<td>NUMBER</td>
<td>4</td>
<td>NO</td>
</tr>
<tr>
<td>BOARD_LINE_ASSIGNMENT_KEY</td>
<td>FOREIGN KEY TO THE BOARD TO LINE ASSIGNMENT ENTITY.</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>DEVICE</td>
<td>A PHYSICAL DEVICE IN A FACTORY</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>KEY</td>
<td>A unique identifier for each instance of a DEVICE</td>
<td>NUMBER</td>
<td>4</td>
<td>NO</td>
</tr>
<tr>
<td>ID</td>
<td>A CRYPTOGRAPHIC NAME ASSOCIATED WITH A DEVICE</td>
<td>NUMBER</td>
<td>4</td>
<td>NO</td>
</tr>
<tr>
<td>DEVICE_TYPE</td>
<td>FOREIGN KEY TO THE DEVICE TYPE ENTITY.</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>KEY</td>
<td>THE MODEL NAME FOR A DEVICE</td>
<td>NUMBER</td>
<td>2</td>
<td>NO</td>
</tr>
<tr>
<td>NAME</td>
<td>A CRYPTOGRAPHIC NAME ASSOCIATED WITH A DEVICE TYPE</td>
<td>NUMBER</td>
<td>4</td>
<td>NO</td>
</tr>
<tr>
<td>DEVICE_PRODUCTION_TYPE</td>
<td>FOREIGN KEY TO THE DEVICE PRODUCTION TYPE ENTITY.</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>KEY</td>
<td>A classification of device used by &quot;WITNESS&quot; only</td>
<td>NUMBER</td>
<td>4</td>
<td>NO</td>
</tr>
<tr>
<td>NAME</td>
<td>A CRYPTOGRAPHIC NAME ASSOCIATED WITH A DEVICE PRODUCTION TYPE ENTITY</td>
<td>NUMBER</td>
<td>4</td>
<td>NO</td>
</tr>
<tr>
<td>RPT_SCENARIO_ASSIGNMENT</td>
<td>A CROSS-REFERENCE ENTITY BETWEEN REPORT CONFIGURATION AND SCENARIO.</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>KEY</td>
<td>A unique identifier for each instance of a report to scenario assignment</td>
<td>NUMBER</td>
<td>4</td>
<td>NO</td>
</tr>
<tr>
<td>RPT_CONFIG</td>
<td>FOREIGN KEY TO THE REPORT CONFIGURATION ENTITY.</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>SCENARIO</td>
<td>FOREIGN KEY TO THE SCENARIO ENTITY.</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>KEY</td>
<td>A unique identifier for each instance of a report configuration</td>
<td>NUMBER</td>
<td>4</td>
<td>NO</td>
</tr>
<tr>
<td>DEFAULT_VALUE_ID</td>
<td>A CRYPTOGRAPHIC NAME ASSOCIATED WITH THE REPORT CONFIGURATION.</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>LINETHROUGHPUT_EST_FLAG</td>
<td>ESTIMATED LINE THROUGHPUT.</td>
<td>NUMBER</td>
<td>2</td>
<td>NO</td>
</tr>
<tr>
<td>BOARD_PRODUCTION_FLAG</td>
<td>QUANTITY OF BOARDS ENTERING PRODUCTION FLAG.</td>
<td>NUMBER</td>
<td>2</td>
<td>NO</td>
</tr>
<tr>
<td>BDGT_PCTSHIP_FLAG</td>
<td>PERCENTAGE OF BOARDS SHIPPED.</td>
<td>NUMBER</td>
<td>2</td>
<td>NO</td>
</tr>
<tr>
<td>BDGT_PCTSCRAP_FLAG</td>
<td>PERCENTAGE OF BOARDS SCRAPPED.</td>
<td>NUMBER</td>
<td>2</td>
<td>NO</td>
</tr>
<tr>
<td>BDGT_PCTASSEMBLY_FLAG</td>
<td>PERCENTAGE OF BOARDS ASSEMBLED.</td>
<td>NUMBER</td>
<td>2</td>
<td>NO</td>
</tr>
<tr>
<td>BDGT_PCTREJECT_FLAG</td>
<td>PERCENTAGE OF BOARDS REJECTED.</td>
<td>NUMBER</td>
<td>2</td>
<td>NO</td>
</tr>
<tr>
<td>AVG_WP_BETLAG</td>
<td>AVERAGE TIME IN PROCESS FLAG.</td>
<td>NUMBER</td>
<td>2</td>
<td>NO</td>
</tr>
<tr>
<td>AVG_PROD_CYCLE_TIME_FLAG</td>
<td>AVERAGE TIME FOR THE PRODUCTION PROCESS FLAG.</td>
<td>NUMBER</td>
<td>2</td>
<td>NO</td>
</tr>
<tr>
<td>DEV_BOTTLENECK_FLAG</td>
<td>DEVICE BOTTLENECK FLAG.</td>
<td>NUMBER</td>
<td>2</td>
<td>NO</td>
</tr>
<tr>
<td>DEV_PCT_BUSY_FLAG</td>
<td>PERCENTAGE OF TIME DEVICE IS BUSY</td>
<td>NUMBER</td>
<td>2</td>
<td>NO</td>
</tr>
<tr>
<td>DEV_PCT_IDLE_FLAG</td>
<td>PERCENTAGE OF TIME DEVICE IS IDLE.</td>
<td>NUMBER</td>
<td>2</td>
<td>NO</td>
</tr>
<tr>
<td>DEV_PCT_STOP_FLAG</td>
<td>PERCENTAGE OF TIME DEVICE IS STOPPED</td>
<td>NUMBER</td>
<td>2</td>
<td>NO</td>
</tr>
<tr>
<td>IDLE_PCT_US_BLOCKED_FLAG</td>
<td>PERCENTAGE OF IDLE TIME DUE TO DOWNSTREAM BLOCKING FLAG.</td>
<td>NUMBER</td>
<td>2</td>
<td>NO</td>
</tr>
<tr>
<td>IDLE_PCT_SETUP_FLAG</td>
<td>PERCENTAGE OF IDLE TIME DUE TO SETUP FLAG.</td>
<td>NUMBER</td>
<td>2</td>
<td>NO</td>
</tr>
<tr>
<td>IDLE_PCT_DEVERROR_FLAG</td>
<td>PERCENTAGE OF IDLE TIME DUE TO DEVICE ERROR FLAG.</td>
<td>NUMBER</td>
<td>2</td>
<td>NO</td>
</tr>
<tr>
<td>IDLE_PCT_REPAIR_FLAG</td>
<td>PERCENTAGE OF IDLE TIME DUE TO REPAIR FLAG.</td>
<td>NUMBER</td>
<td>2</td>
<td>NO</td>
</tr>
<tr>
<td>DEV_OPER_TIME_FLAG</td>
<td>QUANTITY OF OPERATIONS DEVICE HAS COMPLETED.</td>
<td>NUMBER</td>
<td>2</td>
<td>NO</td>
</tr>
<tr>
<td>WIP_LEVEL_FLAG</td>
<td>CALCULATE WIP LEVELS.</td>
<td>NUMBER</td>
<td>2</td>
<td>NO</td>
</tr>
<tr>
<td>SCENARIO_OUTPUT</td>
<td>A unique identifier for each instance of a scenario report output.</td>
<td>NUMBER</td>
<td>4</td>
<td>NO</td>
</tr>
<tr>
<td>RPT_SCENARIO_ASSIGNMENT_KEY</td>
<td>FOREIGN KEY TO THE SCENARIO ASSIGNMENT ENTITY.</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>LINE_THROUGHPUT_EST</td>
<td>ESTIMATED LINE THROUGHPUT.</td>
<td>NUMBER</td>
<td>4</td>
<td>NO</td>
</tr>
<tr>
<td>BOTTLENECK_DEV_ID</td>
<td>A CRYPTOGRAPHIC NAME ASSOCIATED WITH THE BOTTLENECK DEVICE FOR THIS SCENARIO.</td>
<td>CHAR</td>
<td>10</td>
<td>YES</td>
</tr>
<tr>
<td>AVG_WP_BETLAG</td>
<td>AVERAGE TIME IN PROCESS FLAG.</td>
<td>NUMBER</td>
<td>2</td>
<td>NO</td>
</tr>
<tr>
<td>AVG_PROD_CYCLE_TIME_FLAG</td>
<td>AVERAGE TIME FOR THE PRODUCTION PROCESS.</td>
<td>NUMBER</td>
<td>2</td>
<td>NO</td>
</tr>
<tr>
<td>BOARD_OUTPUT</td>
<td>A unique identifier for each instance of a board report output.</td>
<td>NUMBER</td>
<td>4</td>
<td>NO</td>
</tr>
<tr>
<td>REPORT_BOARD_KEY</td>
<td>FOREIGN KEY TO THE REPORT BOARD ASSIGNMENT ENTITY.</td>
<td>NUMBER</td>
<td>4</td>
<td>NO</td>
</tr>
<tr>
<td>BOARD_PRODUCTION_FLAG</td>
<td>QUANTITY OF BOARDS ENTERING PRODUCTION.</td>
<td>NUMBER</td>
<td>2</td>
<td>NO</td>
</tr>
<tr>
<td>BDGT_NUM_SHIPPED</td>
<td>NUMBER OF BOARDS SHIPPED.</td>
<td>NUMBER</td>
<td>2</td>
<td>NO</td>
</tr>
<tr>
<td>BDGT_NUM_SCRAPPED</td>
<td>NUMBER OF BOARDS SCRAPPED.</td>
<td>NUMBER</td>
<td>2</td>
<td>NO</td>
</tr>
<tr>
<td>BDGT_NUM_ASSEMBLED</td>
<td>NUMBER OF BOARDS ASSEMBLED.</td>
<td>NUMBER</td>
<td>2</td>
<td>NO</td>
</tr>
</tbody>
</table>
### Table 3: Entity Attribute Descriptions (Business Model) -- contd.

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Description</th>
<th>Type</th>
<th>Nullable</th>
</tr>
</thead>
<tbody>
<tr>
<td>BDRS NUM RCT</td>
<td>Number of boards rejected</td>
<td>NUMBER</td>
<td>NO</td>
</tr>
<tr>
<td>AVG WIP</td>
<td>Average WIP between front end and test</td>
<td>NUMBER</td>
<td>NO</td>
</tr>
<tr>
<td>WIP MAX</td>
<td>Scenario maximum WIP</td>
<td>NUMBER</td>
<td>NO</td>
</tr>
<tr>
<td>WIP MIN</td>
<td>Scenario minimum WIP</td>
<td>NUMBER</td>
<td>NO</td>
</tr>
<tr>
<td>WIP MAX DEV</td>
<td>Device where max WIP occurs</td>
<td>NUMBER</td>
<td>NO</td>
</tr>
<tr>
<td>WIP MIN DEV</td>
<td>Device where min WIP occurs</td>
<td>NUMBER</td>
<td>NO</td>
</tr>
<tr>
<td>DEVICE RPT OUTPUT</td>
<td>Key</td>
<td>A unique identifier for each instance of a device report output</td>
<td>NUMBER</td>
</tr>
<tr>
<td>RPT DEV ASGN KEY</td>
<td>Key</td>
<td>Foreign key to the report device assignment entity</td>
<td>NUMBER</td>
</tr>
<tr>
<td>DEV FCT BUSY</td>
<td>Percentage of time device is busy</td>
<td>NUMBER</td>
<td>NO</td>
</tr>
<tr>
<td>DEV FCT IDLE</td>
<td>Percentage of time device is idle</td>
<td>NUMBER</td>
<td>NO</td>
</tr>
<tr>
<td>DEV FCT STOP</td>
<td>Percentage of time device is stopped</td>
<td>NUMBER</td>
<td>NO</td>
</tr>
<tr>
<td>DEV FCT WAIT</td>
<td>Percentage of time device is waiting</td>
<td>NUMBER</td>
<td>NO</td>
</tr>
<tr>
<td>IDS FCT DS BLOCK</td>
<td>Percentage of idle time due to downstream blocking</td>
<td>NUMBER</td>
<td>NO</td>
</tr>
<tr>
<td>IDS FCT SETUP</td>
<td>Percentage of idle time due to setup</td>
<td>NUMBER</td>
<td>NO</td>
</tr>
<tr>
<td>IDS FCT DEV ERR</td>
<td>Percentage of idle time due to device error</td>
<td>NUMBER</td>
<td>NO</td>
</tr>
<tr>
<td>IDS FCT REPAIR</td>
<td>Percentage of idle time due to repair</td>
<td>NUMBER</td>
<td>NO</td>
</tr>
<tr>
<td>DEV OPER CMPLT</td>
<td>Quantity of operations device has completed</td>
<td>NUMBER</td>
<td>NO</td>
</tr>
<tr>
<td>RPT DEV ASGN</td>
<td>Key</td>
<td>A unique identifier for each instance of a report device assignment</td>
<td>NUMBER</td>
</tr>
<tr>
<td>RPT CONFG KEY</td>
<td>Key</td>
<td>Foreign key to the report configuration entity</td>
<td>NUMBER</td>
</tr>
<tr>
<td>DEVICE KEY</td>
<td>Key</td>
<td>Foreign key to the device entity</td>
<td>NUMBER</td>
</tr>
<tr>
<td>RPT BOARD ASGN</td>
<td>Key</td>
<td>A unique identifier for each instance of a report board assignment</td>
<td>NUMBER</td>
</tr>
<tr>
<td>REPORT CONFG KEY</td>
<td>Key</td>
<td>Foreign key to the report configuration entity</td>
<td>NUMBER</td>
</tr>
<tr>
<td>BOARD TYPE KEY</td>
<td>Key</td>
<td>Foreign key to the board type entity</td>
<td>NUMBER</td>
</tr>
<tr>
<td>BOARD ASGN</td>
<td>Key</td>
<td>A unique identifier for each instance of a board production route</td>
<td>NUMBER</td>
</tr>
<tr>
<td>SIDE LINE ASGN KEY</td>
<td>Key</td>
<td>Foreign key to the side line assignment entity</td>
<td>NUMBER</td>
</tr>
<tr>
<td>DEVICE KEY</td>
<td>Key</td>
<td>Foreign key to the device sequence entity</td>
<td>NUMBER</td>
</tr>
<tr>
<td>CHIP PLACE DTL</td>
<td>Key</td>
<td>Foreign key to the board product route entity</td>
<td>NUMBER</td>
</tr>
<tr>
<td>BOARD PROD ROUTE KEY</td>
<td>Key</td>
<td>Foreign key to the device entity</td>
<td>NUMBER</td>
</tr>
<tr>
<td>CYCLE TIME</td>
<td>Time</td>
<td>Replenishment time per feeder</td>
<td>NUMBER</td>
</tr>
<tr>
<td>FEEDER SETUP TIME</td>
<td>Time</td>
<td>Setup time between boards</td>
<td>NUMBER</td>
</tr>
<tr>
<td>MEAN TIME TO ASSIST</td>
<td>Time</td>
<td>Mean time to assist</td>
<td>NUMBER</td>
</tr>
<tr>
<td>MEAN TIME OF ASSIST</td>
<td>Time</td>
<td>Mean time of assist</td>
<td>NUMBER</td>
</tr>
<tr>
<td>BUFFER DTL</td>
<td>Key</td>
<td>Foreign key to the board product route entity</td>
<td>NUMBER</td>
</tr>
<tr>
<td>BOARD PROD ROUTE KEY</td>
<td>Key</td>
<td>Foreign key to the device entity</td>
<td>NUMBER</td>
</tr>
<tr>
<td>CAPACITY</td>
<td>Number</td>
<td>Maximum number of boards a buffer can handle</td>
<td>NUMBER</td>
</tr>
<tr>
<td>REFLOW DTL</td>
<td>Key</td>
<td>Foreign key to the board product route entity</td>
<td>NUMBER</td>
</tr>
<tr>
<td>BOARD PROD ROUTE KEY</td>
<td>Key</td>
<td>Foreign key to the device entity</td>
<td>NUMBER</td>
</tr>
<tr>
<td>CAPACITY</td>
<td>Number</td>
<td>Maximum number of boards a reflow oven can handle</td>
<td>NUMBER</td>
</tr>
<tr>
<td>CYCLE TIME</td>
<td>Time</td>
<td></td>
<td>NUMBER</td>
</tr>
<tr>
<td>SETUP TIME</td>
<td>Time</td>
<td></td>
<td>NUMBER</td>
</tr>
<tr>
<td>MEAN TIME TO ASSIST</td>
<td>Time</td>
<td></td>
<td>NUMBER</td>
</tr>
<tr>
<td>MEAN TIME OF ASSIST</td>
<td>Time</td>
<td></td>
<td>NUMBER</td>
</tr>
<tr>
<td>BATCH MACH DTL</td>
<td>Key</td>
<td>Foreign key to the board product route entity</td>
<td>NUMBER</td>
</tr>
<tr>
<td>BOARD PROD ROUTE KEY</td>
<td>Key</td>
<td>Foreign key to the device entity</td>
<td>NUMBER</td>
</tr>
<tr>
<td>CYCLE TIME</td>
<td>Time</td>
<td></td>
<td>NUMBER</td>
</tr>
<tr>
<td>SETUP TIME</td>
<td>Time</td>
<td></td>
<td>NUMBER</td>
</tr>
</tbody>
</table>
### Table 3: Entity Attribute Descriptions (Business Model) -- contd.

<table>
<thead>
<tr>
<th>Entity</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ASY MACH DTL</td>
<td>BOARD PROD ROUTE KEY: FOREIGN KEY TO THE BOARD PROD ROUTE ENTITY.</td>
</tr>
<tr>
<td></td>
<td>DEVICE KEY: FOREIGN KEY TO THE DEVICE ENTITY.</td>
</tr>
<tr>
<td></td>
<td>CYCLE TIME</td>
</tr>
<tr>
<td></td>
<td>SETUP TIME</td>
</tr>
<tr>
<td></td>
<td>MEAN TIME TO ASSIST</td>
</tr>
<tr>
<td></td>
<td>MEAN TIME OF ASSIST</td>
</tr>
<tr>
<td>SINGL MACH DTL</td>
<td>BOARD PROD ROUTE KEY: FOREIGN KEY TO THE BOARD PROD ROUTE ENTITY.</td>
</tr>
<tr>
<td></td>
<td>DEVICE KEY: FOREIGN KEY TO THE DEVICE ENTITY.</td>
</tr>
<tr>
<td></td>
<td>CYCLE TIME</td>
</tr>
<tr>
<td></td>
<td>SETUP TIME</td>
</tr>
<tr>
<td></td>
<td>MEAN TIME TO ASSIST</td>
</tr>
<tr>
<td></td>
<td>MEAN TIME OF ASSIST</td>
</tr>
<tr>
<td>TEST MACH DTL</td>
<td>BOARD PROD ROUTE KEY: FOREIGN KEY TO THE BOARD PROD ROUTE ENTITY.</td>
</tr>
<tr>
<td></td>
<td>DEVICE KEY: FOREIGN KEY TO THE DEVICE ENTITY.</td>
</tr>
<tr>
<td></td>
<td>CYCLE TIME</td>
</tr>
<tr>
<td></td>
<td>SETUP TIME</td>
</tr>
<tr>
<td></td>
<td>MEAN TIME TO ASSIST</td>
</tr>
<tr>
<td></td>
<td>MEAN TIME OF ASSIST</td>
</tr>
<tr>
<td>OTHER MACH DTL</td>
<td>BOARD PROD ROUTE KEY: FOREIGN KEY TO THE BOARD PROD ROUTE ENTITY.</td>
</tr>
<tr>
<td></td>
<td>DEVICE KEY: FOREIGN KEY TO THE DEVICE ENTITY.</td>
</tr>
<tr>
<td></td>
<td>CYCLE TIME</td>
</tr>
<tr>
<td></td>
<td>SETUP TIME</td>
</tr>
<tr>
<td></td>
<td>MEAN TIME TO ASSIST</td>
</tr>
<tr>
<td></td>
<td>MEAN TIME OF ASSIST</td>
</tr>
</tbody>
</table>
10.3 Data Model For Prototype Application (De-normalized)

Figure 25  Data Model for Prototype Application (De-normalized)
## 10.3.1 Entity Attribute Descriptions (Database Schema)

<table>
<thead>
<tr>
<th>WPG Pilot Application</th>
<th>Attribute</th>
<th>Description</th>
<th>Data Type</th>
<th>Length</th>
</tr>
</thead>
<tbody>
<tr>
<td>IB</td>
<td>Number</td>
<td>Instance of a type of printed wiring board</td>
<td>NUMBER</td>
<td>4</td>
</tr>
<tr>
<td>IB</td>
<td>Char</td>
<td>Instance name of a type of printed wiring board</td>
<td>CHAR</td>
<td>20</td>
</tr>
<tr>
<td>IB</td>
<td>Number</td>
<td>Number of boards of one side</td>
<td>NUMBER</td>
<td>2</td>
</tr>
<tr>
<td>IB</td>
<td>Char</td>
<td>Board type name</td>
<td>CHAR</td>
<td>2</td>
</tr>
<tr>
<td>IB</td>
<td>Char</td>
<td>Description of a type of printed wiring board</td>
<td>CHAR</td>
<td>3</td>
</tr>
<tr>
<td>IB</td>
<td>Char</td>
<td>A unique identifier for each instance of a board type</td>
<td>CHAR</td>
<td>4</td>
</tr>
<tr>
<td>IB</td>
<td>Char</td>
<td>A specific side of a type of printed wiring board</td>
<td>CHAR</td>
<td>4</td>
</tr>
<tr>
<td>IB</td>
<td>Char</td>
<td>A sequence of board type side and board line assignment</td>
<td>CHAR</td>
<td>2</td>
</tr>
<tr>
<td>IB</td>
<td>Char</td>
<td>A quantity of board type side</td>
<td>CHAR</td>
<td>2</td>
</tr>
<tr>
<td>IB</td>
<td>Char</td>
<td>A name for a sequence of board type side</td>
<td>CHAR</td>
<td>2</td>
</tr>
<tr>
<td>IB</td>
<td>Char</td>
<td>A foreign key to the board line type</td>
<td>CHAR</td>
<td>2</td>
</tr>
<tr>
<td>IB</td>
<td>Char</td>
<td>A foreign key to the board type table</td>
<td>CHAR</td>
<td>2</td>
</tr>
<tr>
<td>IB</td>
<td>Char</td>
<td>A foreign key to the sequence of board type side and board line assignment</td>
<td>CHAR</td>
<td>2</td>
</tr>
<tr>
<td>IB</td>
<td>Char</td>
<td>A foreign key to the quantity of board type side</td>
<td>CHAR</td>
<td>2</td>
</tr>
<tr>
<td>IB</td>
<td>Char</td>
<td>A name for a sequence of board type side</td>
<td>CHAR</td>
<td>2</td>
</tr>
<tr>
<td>IB</td>
<td>Char</td>
<td>A foreign key to the board line type</td>
<td>CHAR</td>
<td>2</td>
</tr>
<tr>
<td>IB</td>
<td>Char</td>
<td>A foreign key to the board type table</td>
<td>CHAR</td>
<td>2</td>
</tr>
<tr>
<td>IB</td>
<td>Char</td>
<td>A foreign key to the sequence of board type side and board line assignment</td>
<td>CHAR</td>
<td>2</td>
</tr>
<tr>
<td>IB</td>
<td>Char</td>
<td>A foreign key to the quantity of board type side</td>
<td>CHAR</td>
<td>2</td>
</tr>
<tr>
<td>IB</td>
<td>Char</td>
<td>A name for a sequence of board type side</td>
<td>CHAR</td>
<td>2</td>
</tr>
</tbody>
</table>

Table 4    Entity Attribute Descriptions (Database Schema)
| Name          | Entity Type | Key   | Device Type | Device Prod Type | Factory (Line) | Component Type | Device Prod Qty | Line Comp Qty | Report Qty | PK    | FK    | FK    | FK    | FK    | FK    |
|---------------|-------------|-------|-------------|------------------|---------------|----------------|----------------|---------------|------------|--------|-------|-------|-------|-------|-------|-------|
| Device Seq ID | LINE COMP ID | ID    | DEVICE ID   | DEVICE PROD ID   | FACTORY ID    | COMPONENT ID   | DEVICE PROD ID | Line Comp ID | REPORT ID | YES   | NO    | NO    | YES   | NO    | NO    |
| Board Line Seq ID | LINE COMPONENT | ID    | DEVICE ID   | DEVICE PROD ID   | FACTORY ID    | COMPONENT ID   | DEVICE PROD ID | Line Comp ID | REPORT ID | YES   | NO    | NO    | YES   | NO    | NO    |
| Board Line Seq ID | LINE COMPONENT | ID    | DEVICE ID   | DEVICE PROD ID   | FACTORY ID    | COMPONENT ID   | DEVICE PROD ID | Line Comp ID | REPORT ID | YES   | NO    | NO    | YES   | NO    | NO    |
| Board Line Seq ID | LINE COMPONENT | ID    | DEVICE ID   | DEVICE PROD ID   | FACTORY ID    | COMPONENT ID   | DEVICE PROD ID | Line Comp ID | REPORT ID | YES   | NO    | NO    | YES   | NO    | NO    |
| Board Line Seq ID | LINE COMPONENT | ID    | DEVICE ID   | DEVICE PROD ID   | FACTORY ID    | COMPONENT ID   | DEVICE PROD ID | Line Comp ID | REPORT ID | YES   | NO    | NO    | YES   | NO    | NO    |
| Board Line Seq ID | LINE COMPONENT | ID    | DEVICE ID   | DEVICE PROD ID   | FACTORY ID    | COMPONENT ID   | DEVICE PROD ID | Line Comp ID | REPORT ID | YES   | NO    | NO    | YES   | NO    | NO    |
| Board Line Seq ID | LINE COMPONENT | ID    | DEVICE ID   | DEVICE PROD ID   | FACTORY ID    | COMPONENT ID   | DEVICE PROD ID | Line Comp ID | REPORT ID | YES   | NO    | NO    | YES   | NO    | NO    |
| Board Line Seq ID | LINE COMPONENT | ID    | DEVICE ID   | DEVICE PROD ID   | FACTORY ID    | COMPONENT ID   | DEVICE PROD ID | Line Comp ID | REPORT ID | YES   | NO    | NO    | YES   | NO    | NO    |
| Board Line Seq ID | LINE COMPONENT | ID    | DEVICE ID   | DEVICE PROD ID   | FACTORY ID    | COMPONENT ID   | DEVICE PROD ID | Line Comp ID | REPORT ID | YES   | NO    | NO    | YES   | NO    | NO    |
| Board Line Seq ID | LINE COMPONENT | ID    | DEVICE ID   | DEVICE PROD ID   | FACTORY ID    | COMPONENT ID   | DEVICE PROD ID | Line Comp ID | REPORT ID | YES   | NO    | NO    | YES   | NO    | NO    |
| Board Line Seq ID | LINE COMPONENT | ID    | DEVICE ID   | DEVICE PROD ID   | FACTORY ID    | COMPONENT ID   | DEVICE PROD ID | Line Comp ID | REPORT ID | YES   | NO    | NO    | YES   | NO    | NO    |
| Board Line Seq ID | LINE COMPONENT | ID    | DEVICE ID   | DEVICE PROD ID   | FACTORY ID    | COMPONENT ID   | DEVICE PROD ID | Line Comp ID | REPORT ID | YES   | NO    | NO    | YES   | NO    | NO    |
| Board Line Seq ID | LINE COMPONENT | ID    | DEVICE ID   | DEVICE PROD ID   | FACTORY ID    | COMPONENT ID   | DEVICE PROD ID | Line Comp ID | REPORT ID | YES   | NO    | NO    | YES   | NO    | NO    |
| Board Line Seq ID | LINE COMPONENT | ID    | DEVICE ID   | DEVICE PROD ID   | FACTORY ID    | COMPONENT ID   | DEVICE PROD ID | Line Comp ID | REPORT ID | YES   | NO    | NO    | YES   | NO    | NO    |
| Board Line Seq ID | LINE COMPONENT | ID    | DEVICE ID   | DEVICE PROD ID   | FACTORY ID    | COMPONENT ID   | DEVICE PROD ID | Line Comp ID | REPORT ID | YES   | NO    | NO    | YES   | NO    | NO    |
| Board Line Seq ID | LINE COMPONENT | ID    | DEVICE ID   | DEVICE PROD ID   | FACTORY ID    | COMPONENT ID   | DEVICE PROD ID | Line Comp ID | REPORT ID | YES   | NO    | NO    | YES   | NO    | NO    |
| Board Line Seq ID | LINE COMPONENT | ID    | DEVICE ID   | DEVICE PROD ID   | FACTORY ID    | COMPONENT ID   | DEVICE PROD ID | Line Comp ID | REPORT ID | YES   | NO    | NO    | YES   | NO    | NO    |
| Board Line Seq ID | LINE COMPONENT | ID    | DEVICE ID   | DEVICE PROD ID   | FACTORY ID    | COMPONENT ID   | DEVICE PROD ID | Line Comp ID | REPORT ID | YES   | NO    | NO    | YES   | NO    | NO    |
| Board Line Seq ID | LINE COMPONENT | ID    | DEVICE ID   | DEVICE PROD ID   | FACTORY ID    | COMPONENT ID   | DEVICE PROD ID | Line Comp ID | REPORT ID | YES   | NO    | NO    | YES   | NO    | NO    |
| Board Line Seq ID | LINE COMPONENT | ID    | DEVICE ID   | DEVICE PROD ID   | FACTORY ID    | COMPONENT ID   | DEVICE PROD ID | Line Comp ID | REPORT ID | YES   | NO    | NO    | YES   | NO    | NO    |
| Board Line Seq ID | LINE COMPONENT | ID    | DEVICE ID   | DEVICE PROD ID   | FACTORY ID    | COMPONENT ID   | DEVICE PROD ID | Line Comp ID | REPORT ID | YES   | NO    | NO    | YES   | NO    | NO    |
| Board Line Seq ID | LINE COMPONENT | ID    | DEVICE ID   | DEVICE PROD ID   | FACTORY ID    | COMPONENT ID   | DEVICE PROD ID | Line Comp ID | REPORT ID | YES   | NO    | NO    | YES   | NO    | NO    |
| Board Line Seq ID | LINE COMPONENT | ID    | DEVICE ID   | DEVICE PROD ID   | FACTORY ID    | COMPONENT ID   | DEVICE PROD ID | Line Comp ID | REPORT ID | YES   | NO    | NO    | YES   | NO    | NO    |

Table 4  
Entity Attribute Descriptions (Database Schema) -- contd.
### Table 4

**Entity Attribute Descriptions (Database Schema) -- contd.**

<table>
<thead>
<tr>
<th>RPT_CONFIG</th>
<th>KEY</th>
<th>A UNIQUE IDENTIFIER FOR EACH INSTANCE OF A REPORT CONFIGURATION.</th>
<th>NUMBER</th>
<th>4</th>
<th>NO</th>
</tr>
</thead>
<tbody>
<tr>
<td>DEFAULT VALUE</td>
<td>ID</td>
<td>A CRYPTIC NAME ASSOCIATED WITH THE REPORT CONFIGURATION.</td>
<td>CHAR</td>
<td>20</td>
<td>NO</td>
</tr>
<tr>
<td>0</td>
<td>LINE_THRU_PUT_FLAG</td>
<td>ESTIMATED LINE THROUGHPUT FLAG.</td>
<td>NUMBER</td>
<td>2</td>
<td>NO</td>
</tr>
<tr>
<td>0</td>
<td>BOARD_PROD_QTY_FLAG</td>
<td>QUANTITY OF BOARDS ENTERING PRODUCTION FLAG.</td>
<td>NUMBER</td>
<td>2</td>
<td>NO</td>
</tr>
<tr>
<td>0</td>
<td>BRDS_PCT_SHIP_FLAG</td>
<td>PERCENTAGE OF BOARDS SHIPPED FLAG.</td>
<td>NUMBER</td>
<td>2</td>
<td>NO</td>
</tr>
<tr>
<td>0</td>
<td>BRDS_PCT_SCRAP_FLAG</td>
<td>PERCENTAGE OF BOARDS SCRAPPED FLAG.</td>
<td>NUMBER</td>
<td>2</td>
<td>NO</td>
</tr>
<tr>
<td>0</td>
<td>BRDS_PCT_ASSM_FLAG</td>
<td>PERCENTAGE OF BOARDS ASSEMBLED FLAG.</td>
<td>NUMBER</td>
<td>2</td>
<td>NO</td>
</tr>
<tr>
<td>0</td>
<td>BRDS_PCT_REJCT_FLAG</td>
<td>PERCENTAGE OF BOARDS REJECTED FLAG.</td>
<td>NUMBER</td>
<td>2</td>
<td>NO</td>
</tr>
<tr>
<td>0</td>
<td>AVG_PROD_CYCLE_TIME_FLAG</td>
<td>AVERAGE CYCLE TIME FOR THE PRODUCTION PROCESS FLAG.</td>
<td>NUMBER</td>
<td>2</td>
<td>NO</td>
</tr>
<tr>
<td>0</td>
<td>DEV_BOTTLENECK_FLAG</td>
<td>DEVICE BOTTLENECK FLAG.</td>
<td>NUMBER</td>
<td>2</td>
<td>NO</td>
</tr>
<tr>
<td>0</td>
<td>DEV_PCT_BUSY_FLAG</td>
<td>PERCENT OF TIME DEVICE IS BUSY FLAG.</td>
<td>NUMBER</td>
<td>2</td>
<td>NO</td>
</tr>
<tr>
<td>0</td>
<td>DEV_PCT_IDLE_FLAG</td>
<td>PERCENT OF TIME DEVICE IS IDLE FLAG.</td>
<td>NUMBER</td>
<td>2</td>
<td>NO</td>
</tr>
<tr>
<td>0</td>
<td>DEV_PCT_STOP_FLAG</td>
<td>PERCENT OF TIME DEVICE IS STOPPED FLAG.</td>
<td>NUMBER</td>
<td>2</td>
<td>NO</td>
</tr>
<tr>
<td>0</td>
<td>IDLE_PCT_DS_BLOCK_FLAG</td>
<td>PERCENT OF IDLE TIME DUE TO DOWNSTREAM BLOCKING FLAG.</td>
<td>NUMBER</td>
<td>2</td>
<td>NO</td>
</tr>
<tr>
<td>0</td>
<td>IDLE_PCT_SETUP_FLAG</td>
<td>PERCENT OF IDLE TIME DUE TO SETUP FLAG.</td>
<td>NUMBER</td>
<td>2</td>
<td>NO</td>
</tr>
<tr>
<td>0</td>
<td>IDLE_PCT_DEV_ERR_FLAG</td>
<td>PERCENT OF IDLE TIME DUE TO DEVICE ERROR FLAG.</td>
<td>NUMBER</td>
<td>2</td>
<td>NO</td>
</tr>
<tr>
<td>0</td>
<td>IDLE_PCT_REPAIR_FLAG</td>
<td>PERCENT OF IDLE TIME DUE TO REPAIR FLAG.</td>
<td>NUMBER</td>
<td>2</td>
<td>NO</td>
</tr>
<tr>
<td>0</td>
<td>DEV_OPER_CMPRT_FLAG</td>
<td>QUANTITY OF OPERATIONS DEVICE HAS COMPLETED FLAG.</td>
<td>NUMBER</td>
<td>2</td>
<td>NO</td>
</tr>
<tr>
<td>0</td>
<td>AVG_WIP_QTY_FLAG</td>
<td>AVERAGE WORK IN PROGRESS FLAG.</td>
<td>NUMBER</td>
<td>2</td>
<td>NO</td>
</tr>
<tr>
<td>0</td>
<td>WIP_LEVEL_FLAG</td>
<td>CALCULATE WIP LEVELS FLAG.</td>
<td>NUMBER</td>
<td>2</td>
<td>NO</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>RPT_DEV_ASSIGN</th>
<th>KEY</th>
<th>A CROSS REFERENCE ENTITY BETWEEN REPORT CONFIGURATION AND DEVICE.</th>
<th>NUMBER</th>
<th>4</th>
<th>NO</th>
</tr>
</thead>
<tbody>
<tr>
<td>* RPT_CONFIG_ID</td>
<td>FOREIGN KEY TO THE REPORT CONFIGURATION TABLE.</td>
<td>CHAR</td>
<td>20</td>
<td>NO</td>
<td></td>
</tr>
<tr>
<td>* SCENARIO_NAME</td>
<td>FOREIGN KEY TO THE SCENARIO TABLE.</td>
<td>CHAR</td>
<td>20</td>
<td>NO</td>
<td></td>
</tr>
<tr>
<td>* DEVICE_ID</td>
<td>FOREIGN KEY TO THE DEVICE TABLE.</td>
<td>CHAR</td>
<td>20</td>
<td>NO</td>
<td></td>
</tr>
<tr>
<td>DEV_PCT_BUSY</td>
<td>PERCENT OF TIME DEVICE IS BUSY.</td>
<td>NUMBER</td>
<td>2</td>
<td>NO</td>
<td></td>
</tr>
<tr>
<td>DEV_PCT_IDLE</td>
<td>PERCENT OF TIME DEVICE IS IDLE.</td>
<td>NUMBER</td>
<td>2</td>
<td>NO</td>
<td></td>
</tr>
<tr>
<td>DEV_PCT_STOP</td>
<td>PERCENT OF TIME DEVICE IS STOPPED.</td>
<td>NUMBER</td>
<td>2</td>
<td>NO</td>
<td></td>
</tr>
<tr>
<td>STOP_PCT_DS_BLOCK</td>
<td>PERCENT OF STOPPED TIME DUE TO DOWNSTREAM BLOCKING.</td>
<td>NUMBER</td>
<td>2</td>
<td>NO</td>
<td></td>
</tr>
<tr>
<td>STOP_PCT_SETUP</td>
<td>PERCENT OF STOPPED TIME DUE TO SETUP.</td>
<td>NUMBER</td>
<td>2</td>
<td>NO</td>
<td></td>
</tr>
<tr>
<td>STOP_PCT_DEV_ERR</td>
<td>PERCENT OF STOPPED TIME DUE TO DEVICE ERROR.</td>
<td>NUMBER</td>
<td>2</td>
<td>NO</td>
<td></td>
</tr>
<tr>
<td>STOP_PCT_REPAIR</td>
<td>PERCENT OF STOPPED TIME DUE TO REPAIR.</td>
<td>NUMBER</td>
<td>2</td>
<td>NO</td>
<td></td>
</tr>
<tr>
<td>DEV_OPER_CMPRT</td>
<td>QUANTITY OF OPERATIONS DEVICE HAS COMPLETED.</td>
<td>NUMBER</td>
<td>2</td>
<td>NO</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>RPT_BOARD_ASSIGN</th>
<th>KEY</th>
<th>A CROSS REFERENCE ENTITY BETWEEN REPORT CONFIGURATION AND BOARD TYPE ENTITY.</th>
<th>NUMBER</th>
<th>4</th>
<th>NO</th>
</tr>
</thead>
<tbody>
<tr>
<td>* RPT_CONFIG_ID</td>
<td>FOREIGN KEY TO THE REPORT CONFIGURATION TABLE.</td>
<td>CHAR</td>
<td>20</td>
<td>NO</td>
<td></td>
</tr>
<tr>
<td>* BOARD_TYPE_NAME</td>
<td>FOREIGN KEY TO THE BOARD TYPE TABLE.</td>
<td>CHAR</td>
<td>20</td>
<td>NO</td>
<td></td>
</tr>
<tr>
<td>* SCENARIO_NAME</td>
<td>FOREIGN KEY TO THE SCENARIO TABLE.</td>
<td>CHAR</td>
<td>20</td>
<td>NO</td>
<td></td>
</tr>
<tr>
<td>BOARD_PROD_QTY</td>
<td>QUANTITY OF BOARDS ENTERING PRODUCTION.</td>
<td>NUMBER</td>
<td>2</td>
<td>NO</td>
<td></td>
</tr>
<tr>
<td>BRDS_NUM_SHIP</td>
<td>NUMBER OF BOARDS SHIPPED.</td>
<td>NUMBER</td>
<td>2</td>
<td>NO</td>
<td></td>
</tr>
<tr>
<td>BRDS_NUM_SCRAP</td>
<td>NUMBER OF BOARDS SCRAPPED.</td>
<td>NUMBER</td>
<td>2</td>
<td>NO</td>
<td></td>
</tr>
<tr>
<td>BRDS_NUM_ASSM</td>
<td>NUMBER OF BOARDS ASSEMBLED.</td>
<td>NUMBER</td>
<td>2</td>
<td>NO</td>
<td></td>
</tr>
<tr>
<td>BRDS_NUM_REJCT</td>
<td>NUMBER OF BOARDS REJECTED.</td>
<td>NUMBER</td>
<td>2</td>
<td>NO</td>
<td></td>
</tr>
<tr>
<td>AVG_CYCLE_TIME</td>
<td>AVERAGE CYCLE TIME FOR BOARD IN THIS SCENARIO.</td>
<td>NUMBER</td>
<td>2</td>
<td>NO</td>
<td></td>
</tr>
<tr>
<td>Attribute</td>
<td>Description</td>
<td>Type</td>
<td>Nullable</td>
<td></td>
<td></td>
</tr>
<tr>
<td>--------------------</td>
<td>-----------------------------------------------------------</td>
<td>-------</td>
<td>----------</td>
<td></td>
<td></td>
</tr>
<tr>
<td>AVG_WIP</td>
<td>AVERAGE WIP BETWEEN FRONT END AND TEST.</td>
<td>NUMBER</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>WIP_LEVEL</td>
<td>WIP LEVEL BETWEEN FRONT END AND TEST.</td>
<td>NUMBER</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>BOARD_PROD_ROUTE</td>
<td>CROSS REFERENCE BETWEEN SIDE TO LINE ASSGN AND DEVICE SEQUENCE ENTITIES</td>
<td>NUMBER</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>KEY</td>
<td>A UNIQUE IDENTIFIER FOR EACH INSTANCE OF A BOARD PRODUCTION ROUTE</td>
<td>CHAR</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>BOARD_LINE_ASSGN_ID</td>
<td>FOREIGN KEY TO THE BOARD LINE ASSIGNMENT TABLE</td>
<td>CHAR</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>BOARD_TYPE_NAME</td>
<td>FOREIGN KEY TO THE BOARD TYPE TABLE</td>
<td>CHAR</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>BOARD_TYPE_SIDE</td>
<td>FOREIGN KEY TO THE BOARD TYPE SIDE TABLE WHEN USED WITH THE BOARD_TYPE_NAME</td>
<td>NUMBER</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>DEVICE_SEQUENCE</td>
<td>FOREIGN KEY TO THE DEVICE SEQUENCE TABLE.</td>
<td>NUMBER</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>SETUP_TIME</td>
<td></td>
<td>NUMBER</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>CYCLE_TIME</td>
<td></td>
<td>NUMBER</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>FEEDER_REPLEN_TIME</td>
<td>FEEDER REPLENISHMENT TIME</td>
<td>NUMBER</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>FEEDER_SETUP_TIME</td>
<td>FEEDER SETUP TIME BETWEEN BOARDS.</td>
<td>NUMBER</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>MEAN_TIME_TO_ASSIST</td>
<td></td>
<td>NUMBER</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>MEAN_TIME_OF_ASSIST</td>
<td></td>
<td>NUMBER</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>CAPACITY</td>
<td>MAXIMUM NUMBER OF BOARDS A REFLOW OVEN CAN HANDLE.</td>
<td>NUMBER</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>BATCH_MIN</td>
<td></td>
<td>NUMBER</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>BATCH_MAX</td>
<td></td>
<td>NUMBER</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>ASSY_QTY</td>
<td></td>
<td>NUMBER</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>YIELD</td>
<td></td>
<td>CHAR</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>REWORK_DEV_ID</td>
<td></td>
<td>CHAR</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
11.0 PROJECT REVIEW AND NEXT STEPS

11.1 Chapter Overview
This chapter reflects upon the methods used during this six and a half month internship and outlines some of the strengths and weaknesses of this effort. It also provides a brief list of recommendations for future improvement and outlines some of the next steps that should be taken in order to carry the project forward into its next phase.

11.2 Strengths and Weaknesses of Project Methodology

Strengths

• The project team followed an extremely aggressive schedule and was able to deliver the prototype application on the initial target date in spite of the fact that there was skepticism from both the project team and the team from the pilot site as to whether this due date could be met. This success was largely due to the fact that team members recognized the importance of meeting this aggressive schedule and made personal commitments up front to put their best foot forward.

• A fair amount of time was spent initially in identifying the high level software requirements for the project (even before the pilot customer site was identified). After identifying the pilot customer site, a significant amount of time was spent in refining and enhancing these requirements via iterative meetings with core team members from the pilot customer site. As a result, there was very little ambiguity once the software development process was begun and there was no wastage of effort in terms of software re-design or rework for the prototype software application.

• The project team was kept fairly small and this made management of team efforts easier. The team comprised of:
  -- Three software developers (the author functioned as one of the software developers and also as team leader)
  -- One project manager
  -- Four core team members from the pilot factory site (two lead users and two CIM engineers).

• The importance of the project was highlighted up front by upper management. Hence the task of obtaining buy in from project team members (both at Motorola Manufacturing Systems and at the Wireless Data Group) was made easier.

Weaknesses

• The project got off to a slow start owing to the fact that the author's initial LFM internship was not related to Factory Planning. Hence, instead of commencing the Factory Planning project at the start of the author's internship (June 1994), this project was initiated only in the first week of July 1994.

• The pilot customer site was yet to ramp up production to full capacity and hence certain characteristics of this site were not representative of the wider customer base.
within Motorola that the Factory Planning Application can serve. As a result, there was some amount of tension between the functionalities that the pilot customer wished to see in the prototype application and the functionalities that the project team felt would be essential for a wider customer base within Motorola.

- The software development environment was sometimes unstable due to the lack of a systems administrator to manage the software development platform. Furthermore, project activities were interrupted several times due to the physical movement of the team from one location to another (There were a total of four moves during the six month period. These relocations could not be avoided, however, because the department was undergoing rapid expansion).

- There was sometimes a struggle between the look and feel of the prototype application desired by the pilot customer and the functionalities that could be included in the prototype application in order to cater these requirements. These functionalities were limited owing to the limitations of the environment chosen to develop the user interface (Tcl Tk) and the fact that the software team was climbing the learning curve in developing software in this environment. Hence, in such struggles between the "market-in" and "product-out" phenomenons, the "product-out" perspective was sometimes chosen.

11.3 Next Steps

The steps that should be followed to carry this project to its next phase include the following:

- Use the prototype application to analyze different "what-if" scenarios for new Motorola factories (such as the Motorola Cellular Subscriber Group's new factory at Harvard, IL). This effort will serve to further validate the prototype application.

- Prioritize the next steps pertaining to the integration of other software applications into the Factory Planning Application (other throughput optimization, cost analysis and quality analysis tools).

- Determine the applicability of SEMATECH's Semiconductor Workbench for Integrated Modeling (SWIM) cost and quality analyses tools to the Factory Planning software.

- Determine a project plan for the overall multi-year, multi-person Factory Planning project at Motorola.
END NOTES


2. "Yo-One" is a ceremony taught at MIT's Leaders For Manufacturing Program. It is used to signify completion and agreement of critical or significant tasks and is typically invoked when a phase of team activity has been finished. Also see Shoji Shiba, Alan Graham and David Walden, *A New American TQM -- Four Practical Revolutions in Management*, (Cambridge, MA: Productivity Press, 1993).


4. Some of this information was obtained from the following CD ROM disc: Data Sources Report -- Ziff Communications Company.

5. Prior to his joining Motorola, the author was a Strategic Consultant at one such firm where this technique was aggressively followed.

6. This estimate was made by David Liljegren of Motorola Manufacturing Systems, who is widely regarded as a simulation expert within Motorola.

7. This estimate was also made by David Liljegren of Motorola Manufacturing Systems.

8. The diagrams and tables in this section (Section 6.6) were drawn by Mike Sorkis (of Motorola Manufacturing Systems, Schaumburg, IL). These were constructed on the following sources:

   -- Mike Sorkis' knowledge of the subject matter.

   -- A one page document on this subject written by the author (on the basis of discussions with Mike Sorkis and Chip Dawes of Motorola Manufacturing Systems located in Schaumburg, IL).
Bibliography

BIBLIOGRAPHY


• Roberts B., Edward (editor); Managerial Applications of System Dynamics, Productivity Press, MA (1978).


• Rumbaugh, James; Blaha, Michael; Premerlani, William; Eddy, Frederick; Lorensen, William; Object Oriented Modeling and Design, Prentice Hall, NJ (1991).


• Software Engineering Standards, The Institute of Electrical and Electronics Engineers, Inc. (October 1989).

• “Motorola Cellular Subscriber Group AXIS++ Data Dictionary and Usage Version 5.02.02” (October 1993).

• “Semiconductor Workbench for Integrated Modeling -- Integrated Data Model 5.0, SEMATECH Technology Transfer No. 92041103B-ENG” (May 1993).


