

AD-A253 332



2

CDRL No. 0002AB-2

Quarterly Technical Report - Report No. 2

April 1, 1992 - June 30, 1992

DARPA DICE Manufacturing Optimization

**Linda J. Lapointe
Robert V.E. Bryant**



Raytheon Company

1992

**This document has been approved
for public release and sale; its
distribution is unlimited.**

**DARPA
Defense Advanced Research
Projects Agency**



92 7 24 013

CDRL No. 0002AB-2

Quarterly Technical Report - Report No. 2
April 1, 1992 - June 30, 1992
DARPA DICE Manufacturing Optimization

Prepared by
Linda J. Lapointe
Robert V.E. Bryant

DTIC QUALITY INSPECTED 4

Raytheon Company
Missile Systems Laboratories
Tewksbury, MA 01876

July 1992

ARPA Order No. 8363/02
Contract MDA972-92-C-0020

Prepared for

DARPA
Defense Advanced Research
Projects Agency

Contracts Management Office
Arlington, VA 22203-1714

Accession For	
NTIS CRA&I	<input checked="" type="checkbox"/>
DTIC TAB	<input type="checkbox"/>
Unannounced	<input type="checkbox"/>
Justification	
By <i>pn A251363</i>	
Distribution /	
Availability Codes	
Dist	Avail and/or Special
<i>A-1</i>	

Contents

1. Summary	1
2. Introduction	3
3. Methods, Assumptions, and Procedures	4
3.1 High Level PWB Design Flow	4
3.2 Typical PWB Manufacturing Flow	5
4. Results and Discussion	8
4.1 External Interface Requirements	8
4.1.1 Product - Process Team Communication	8
4.1.2 CAD Database Interface	9
4.2 Capability Requirements	9
4.2.1 Process Analyzer	9
4.2.2 Guidelines Analyzer	10
4.2.3 Yield & Rework Analyzer	10
4.2.4 Cost Estimator	11
4.2.5 Manufacturing Advisor	11
5. Conclusions	13
6. References	15
7. Notes	16
7.1 Acronyms	16
Appendix I - Raytheon PWB EXPRESS Model	17
10. PWB EXPRESS Schemas	17
10.1 rpd_design.exp	17
10.2 rpdtypes.exp	18
10.3 rpd_header.exp	19
10.4 alias.exp	20
10.5 annotation.exp	21
10.6 cari.exp	21
10.7 class.exp	21
10.8 comment.exp	22
10.9 dr_block.exp	22
10.10 gate.exp	25
10.11 net.exp	26
10.12 metal_area.exp	27

10.13 part.exp.....	28
10.14 pin.exp.....	30
10.15 route.exp.....	31
10.16 via.exp.....	32
10.17 xref.exp.....	33
10.18 shape.exp.....	33
10.19 stackup.exp.....	34
10.20 model.exp.....	34
Appendix II - MO OOA Model.....	37
Appendix III - Second Quarterly Review Presentation Slides.....	38
Figure 3-1. Printed Wiring Board Design Flow	4
Figure 3-2. Printed Wiring Board Manufacturing Flow	6
Figure II-1. MO OOA Diagram	37

1. Summary

This is the Quarterly Technical Report for the DARPA DICE Manufacturing Optimization. The goal of the Manufacturing Optimization (MO) system is to facilitate a two tiered team approach to the product/process development cycle where the product design is analyzed by multiple manufacturing engineers, and the product/process changes are traded concurrently in the product and process domains. The system will support Design for Manufacturing and Assembly (DFMA) with a set of tools to model the manufacturing processes, and manage tradeoffs across multiple processes. The subject of this report is the technical work accomplished during the second quarter of the contract. This report describes the development efforts of the MO Functional Requirements and Measure of Performance Report. The PWB EXPRESS Models are based on the Raytheon Automated Placement and Interconnect Design System (RAPIDS) Structured Database (RSD) and the RAPIDS Library Database (RLD).

The methodology used to acquire and define the functional requirements for the Manufacturing Optimization (MO) System consisted of analyzing a typical design and manufacturing cycle for standard through-hole printed wiring boards, including both assembly and fabrication. This technique highlighted the inter-relationships among design and manufacturing processes, and established a baseline for examining the role of DFMA in a concurrent engineering environment. After analyzing the selected manufacturing areas, with respect to the design flow, the internal capabilities and external interfaces were determined.

The external interface requirements include communications for the two tiered virtual tiger team and an interface to the CAD database. The two tiered approach consists of a cross functional product team linked to teams within each of the functions, in this case a manufacturing process team. To implement this approach there must be communication among the members of each team, and between the product and process team; therefore, product-to-process and process-to-product team communication will be supported. The CAD database interface will be between the Raytheon Automated Placement and Interconnect Design System (RAPIDS) and the ROSE database. RAPIDS will provide a graphical CAD environment for displaying and manipulating the PWB product design. The interface will be bi-directional to support manipulation of the data within ROSE and subsequent re-use by RAPIDS. An information model, representing PWB design and manufacturing data, was developed with the

EXPRESS information modeling language used by the PDES/STEP standards. The existing RAPIDS data dictionary was used as the basis for this PWB EXPRESS model.

Using Object-Oriented Analysis (OOA) techniques, the internal requirements were modeled and organized into five major subject areas: process analyzer, guidelines analyzer, yield & rework analyzer, cost estimator, and manufacturing advisor. The process analyzer selects or determines the process sequence required to manufacture the product design. The capability to select the process sequence is based on the evaluation of product design parameters or process parameters. The guideline analyzer will evaluate a design against a set of design for manufacturing guidelines. These manufacturing guidelines may delineate quantitative and/or qualitative manufacturability issues. The yield and rework analyzer calculate yield and rework rates for a selected process sequence, associated with a product design, on an operation by operation basis. The cost estimator calculates the recurring manufacturing cost for each operation of the process sequence. The manufacturing advisor allows viewing of the results produced by each process participating in an analysis.

Raytheon will continue development of MO during the next quarter based on the "Functional Requirements and Measure of Performance For the Manufacturing Optimization (MO) System" document developed during the reporting period. An initial prototype will be developed during the next quarter with a target demonstration date in September 1992. Raytheon is also in the process of developing the Design Specification which will be delivered during the fourth quarter of 1992.

2. Introduction

This is the Quarterly Technical Report for the DARPA DICE Manufacturing Optimization. The concept behind the Manufacturing Optimization (MO) system is to facilitate a two tiered team approach to the product/process development cycle where the product design is analyzed by multiple manufacturing engineers, and the product/process changes are traded concurrently in the product and process domains. The system will support DFMA with a set of tools to model the manufacturing processes, and manage tradeoffs across multiple processes. The subject of this report is on the technical work accomplished during the second quarter of the contract. This report describes the development efforts of the MO Functional Requirements and Measure of Performance Report. The PWB EXPRESS Models are based on the Raytheon Automated Placement and Interconnect Design System (RAPIDS) Structured Database (RSD) and the RAPIDS Library Database (RLD).

The primary objective of the MO system is in researching and developing a DFMA environment capable of modeling diverse manufacturing processes. This environment will allow multiple manufacturing engineers to simultaneously analyze a design, compare or merge results, and conduct trade-off studies, which will result in a consolidated and comprehensive evaluation of the manufacturability of a product. To fulfill this mission and gather the system requirements, analysis of specific manufacturing areas, with respect to the high level design cycle, was performed. Since MO is to be demonstrated using a printed wiring board (PWB), the selected areas included Printed Wiring Assembly (PWA) and Fabrication (PWF). After analyzing the selected manufacturing areas, the MO system capabilities and external interface requirements were determined. The remainder of this report is devoted to the methods, assumptions, and procedures, as well as, the results, conclusions, and recommendations regarding the definition of the MO functional requirements.

3. Methods, Assumptions, and Procedures

The methodology used to acquire and define the functional requirements for the Manufacturing Optimization (MO) System consisted of analyzing a typical design and manufacturing cycle for standard through hole printed wiring boards, including both assembly and fabrication. This technique highlighted the inter-relationships among design and manufacturing processes, and established a baseline for examining the role of Design for Manufacturing and Assembly (DFMA) in a concurrent engineering environment. After analyzing the selected manufacturing areas, with respect to the design cycle, the MO system capabilities and external interface requirements were determined. Provided below is an overview of the typical design and manufacturing cycle that was analyzed in order to achieve the resulting functional requirements.

3.1 High Level PWB Design Flow

The high level design flow for printed wiring boards, excluding revision cycles, consists of the following steps: packaging concept, design capture, design analysis/verification, component placement, interconnect routing, and documentation/transition to manufacturing. Figure 3-1 depicts the high level printed wiring board design flow. Each step is described in the Functional Requirements and Measure of Performance Report section 4.2 (reference 4) in terms of the functions performed and the design features, or attributes, determined. The packaging concept and component placement design steps are included in this report as a sample of what type of functions are performed and what kind of design features and attributes are determined.

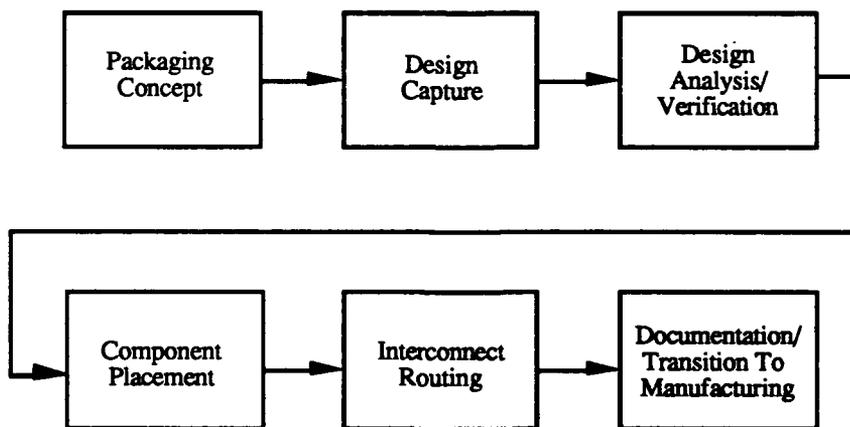


Figure 3-1. Printed Wiring Board Design Flow

During the concept phase, the product team establishes a packaging concept based on the system requirements specification. The concept is reviewed for functionality, performance, fit, power and thermal, cost and schedule, reliability, manufacturability, and test. Among the attributes determined at this stage are: number of required board types, board geometries (area, length, width, aspect ratio), number of interconnect and power/ground layers, layer stackup, layer-to-layer spacing and tolerances, PWB materials, design rules (trace widths, spacings, and via sizes), component family and attachment method, preferred parts, thermal management, test strategy, and cost, performance and reliability budgets. Thus, a packaging baseline is established, although many of these parameters may be challenged and re-evaluated during the detail design process.

Component placement is often initiated by the circuit design engineer, and later refined by the layout draftsman. Approval by the circuit designer is always required prior to initiating detailed layout. Crucial to the placement function is the need to achieve an effective balance between circuit performance and thermal behavior. Thus, critical paths must be kept short, while reactive and hot components must be kept apart. Density, timing simulation, and thermal analysis tools are frequently invoked to identify conflicts even prior to detailed interconnect routing. Manufacturability analysis tools are also employed to assess compatibility with automated component insertion equipment, and to evaluate the cost/yield implications of alternative layer stackups and interconnect design rules. During this process, part substitutions may be evaluated in order to resolve packaging, reliability, and manufacturability problems. Upon conclusion of the placement process, component locations and orientation have been determined and the required parts list substitutions have been implemented.

3.2 Typical PWB Manufacturing Flow

Figure 3-2 depicts the typical manufacturing assembly and fabrication flow for printed wiring boards. At each step in the flow there are design attributes which influence the manufacturing process. The four manufacturing processes highlighted in the figure below is described in the Functional Requirements and Measure of Performance Report section 4.3 (reference 4). The auto insert/pick and place and the flow solder/reflow clean processes are included in this report as samples to show the type of design attributes that influence a manufacturing process.

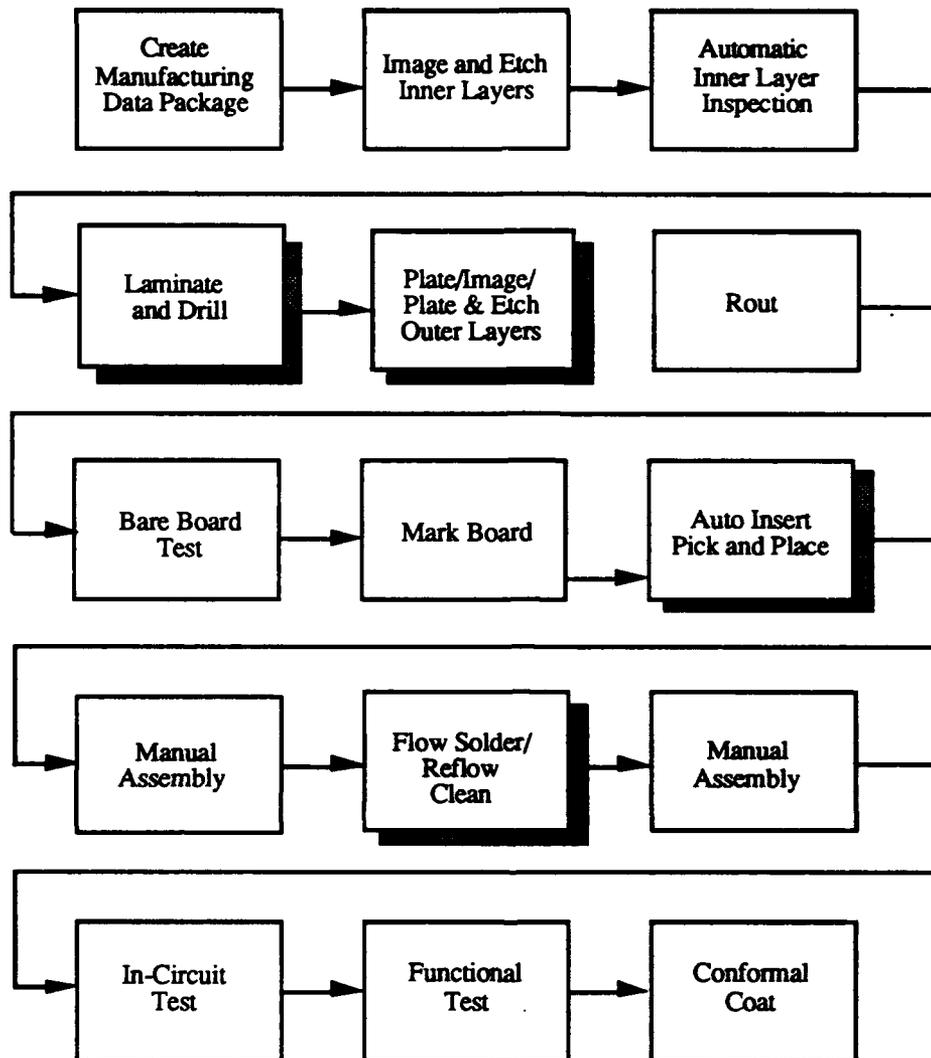


Figure 3-2. Printed Wiring Board Manufacturing Flow

The Auto Insertion/Pick and Place processes involve automatic insertion of through-hole components and the attachment of surface mount devices. The critical design attributes, which influence the auto insertion process, include: insertability of each component type, number of components by insertion type, component orientation by component type, component-component spacing, component-obstruction spacing, board thickness versus component lead length, lead diameter versus hole diameter, static sensitivity, sequencer compatibility of components, and component bonding/attachment method.

Fundamental to automated insertion is the inherent insertability of specific component types. A secondary, but similarly important criteria is the number of components of a given

class (e.g., dual in-line IC or surface mount packages) that ultimately determines the economic feasibility of allocating and setting up a specialized insertion machine for a limited number of components. Even were the component type and count meet the required criteria, component orientation and component spacing become important factors. The best case occurs where all components are aligned in the same direction, less attractive is where components of a given type occur at zero and ninety degree rotations, and worst case is where components are aligned off-axis (i.e., 45 degrees or other).

The Flow Solder process is where through-hole components are soldered onto the board using a "wave" of molten solder and surface mount components are reflowed using a vapor phase, IR, convection, or combination. The critical design attributes, which influence the flow solder process include: board thickness/lead protrusion, thermal sensitivity of components, metal balance, component orientation, board geometry, presence of interconnect or ground plane on solder side, pad geometry, plated through-hole lead diameter, and aspect ratio of through-hole.

Thick boards, coupled with short lead protrusion, and improperly sized/shaped pads on the solder side create lead clinching problems. Pads that do not have proper thermal relief with respect to copper power and ground planes become heat sinks and promote poor solder joints. Conductor paths on the solder side cause solder bridging. Heat sensitive components require special handling and solder-side ground planes cause a variety of problems associated with heat absorption.

4. Results and Discussion

The primary objective of the MO system is in researching and developing a DFMA environment capable of modeling diverse manufacturing processes, which allows manufacturing specialists to participate in the product/process development activities concurrently. To fulfill this mission and gather the system requirements, analysis of specific manufacturing areas, with respect to the high level design cycle, was performed. Since MO is to be demonstrated using a printed wiring board (PWB), the selected areas included the Printed Wiring Assembly (PWA) and Fabrication (PWF). After analyzing the selected manufacturing areas, with respect to the design cycle, the MO system capabilities and external interface requirements were determined. Provided below is an overview of the resulting requirements, which are specified in the Functional Requirements and Measure of Performance Report (reference 4).

4.1 External Interface Requirements

4.1.1 Product - Process Team Communication

MO introduces the concept of a two tiered virtual tiger team. The two tiered approach consists of a cross functional product team linked to teams within each of the functions, in this case a manufacturing process team. To implement this approach there must be communication among the members of each team, and between the product and process team. The following capabilities are required to support this communication:

- **Product - to - Process Team Communication**
 - Notification of design task completed or other pertinent status information.
 - Notification and issuance of database available for analysis.
 - Notification of alternative designs or trade-off decisions under consideration.

- **Process - to - Product Team Communication**
 - Notification and issuance of analysis results.
 - Notification and issuance of modified database with recommended changes.
 - Notification of changes to the process, guidelines, cost or yield models.

4.1.2 CAD Database Interface

MO will use PWB design data, stored in a ROSE database, as input. At Raytheon, PWB data is currently stored in two related file based databases called the RAPIDS Structured Database (RSD) and the RAPIDS Library Database (RLD), where RAPIDS stands for Raytheon Automated Placement and Interconnect Design System. MO will support an interface between the RAPIDS and ROSE database. RAPIDS will provide a graphical CAD environment for displaying and manipulating the PWB product design. The interface will be bi-directional to support manipulation of the data within ROSE and subsequent re-use by RAPIDS. Since ROSE is a neutral database that uses the PDES/STEP standards, interfacing MO to other commercial PWB CAD systems is possible.

An information model, representing PWB design and manufacturing data, was developed with the EXPRESS information modeling language used by the PDES/STEP standards. The existing RAPIDS data dictionary was used as the basis for this PWB EXPRESS model. Appendix I contains all the EXPRESS schemas which constitute the Raytheon PWB EXPRESS model.

4.2 Capability Requirements

The requirements were modeled using Object-Oriented Analysis (OOA) techniques where the required MO capabilities were organized into five major subject areas: process analyzer, guidelines analyzer, yield & rework analyzer, cost estimator, and manufacturing advisor. Each of these areas will be described in the following subsections. Appendix II contains the complete OOA diagram for the MO system. Refer to reference 4 Appendix I for an explanation of the OOA notations used in MO OOA diagram.

4.2.1 Process Analyzer

The process analyzer provides the capability to select or determine the process sequence required to manufacture the product design. The manufacturing process will be represented by three levels of abstraction: the process, operation, and operational step. The process is an organized group of manufacturing operations, the operation is a common unit of work that is performed on the part, and the operational step is an elemental unit of work within an operation.

The process operation sequence for a given product design will be selected from a list of all available operations within the process. The operation steps sequence within an operation will be selected from a list of all available steps within an operation.

The process analyzer will be able to select a process sequence based on the evaluation of product design parameters or process parameters. The evaluation function will use an "if-then" structure. An "if" statement that evaluates to true will result in the execution of the "then" statement. The "then" statement will add an operation or step to the process sequence list.

4.2.2 Guidelines Analyzer

The guideline analyzer will provide the capability to evaluate a design against a set of design for manufacturing guidelines. Manufacturing guidelines may delineate quantitative and/or qualitative manufacturability issues. The guidelines will be structured with an "if" statement that defines the parameters the guideline evaluates, and a "then" statement which will be the recommended action or caution related to the evaluated guideline.

4.2.3 Yield & Rework Analyzer

The yield and rework analyzer will provide the capability to calculate yield and rework rates for a selected process sequence associated with a product design. This capability must provide for calculation of the yield or rework rate on an operation level within the process sequence. The rate will be calculated based on the design features' influence on the operation. The yield or rework rate for each design feature associated with an operation will be calculated using either of the following techniques:

- A look-up table that will select the rate based on the value of a design feature. The table will be structured to include the operation number, the design feature, the feature value, and the scrap rate. The scrap rate is equal to $(1 - \text{yield})$.
- Through the evaluation of an equation to calculate the rate where the equation may include product design parameters. The slope of yield curve could be expressed as an equation for example.

The total yield or rework rate for an operation will be calculated by treating the contributing scrap or rework rates as non-mutually exclusive independent events.

4.2.4 Cost Estimator

The cost estimator provides the capability to calculate the recurring manufacturing cost for each operation of the process sequence. The following calculations will be performed:

- Labor standards for each operation will be calculated for setup and run time categories. The value for each of these categories will be calculated through the evaluation of an equation. The equation may include design parameters. Each category will have an associated labor grade or bid code for each operation.
- Estimated ideal cost for each operation will be calculated from labor standard values multiplied by the wage rate of the labor category performing the operation, and the production efficiency value for that operation.
- Rework operations will be calculated based on the rework rate determined by the yield and rework analyzer multiplied by labor standards for the rework condition. The labor grade wage rates and production efficiencies would then be applied.
- For each operation, the estimated actual cost will be calculated by multiplying the estimated ideal cost by the number of units processed, including both good and scrapped units. The number of units processed by each operation will be calculated from the value of the required good units at the subsequent operation divided by the yield at the operation under evaluation.
- The total estimated ideal cost and total estimated actual cost for each process sequence will be calculated by summing the individual operation cost of each. The estimated actual cost for a good unit will be calculated by dividing the total estimated actual cost for the process by the number of good units produced.

4.2.5 Manufacturing Advisor

The manufacturing advisor provides the capability to view the results produced by each process participating in an analysis. The advisor will include the following capabilities:

- Provide viewing capabilities for single process analysis results including process sequence, yield and rework, cost, and guidelines.

- Provide a mechanism for comparing and displaying the results from two runs of an analysis on a single process sequence.
- Provide the capability to summarize design features causing manufacturing guideline violations across multiple processes. Report recommendations on these guideline violations.
- Provide a summary report, identifying cost drivers, for each process contributing to a multi-process analysis for a given design database.

5. Conclusions

Analysis of specific manufacturing areas, with respect to the high level design cycle, was performed. Since MO is to be demonstrated using a printed wiring board, the selected areas included printed wiring assembly and fabrication. Based on the study, the internal capabilities and external interface requirements for the MO system were defined.

The external interface requirements include communications for the two tiered virtual tiger team and a CAD database interface. The two tiered approach consists of a cross functional product team linked to teams within each of the functions, in this case a manufacturing process team. This approach will support communication among the members of each team, and between the product and process team. The CAD interface, which MO will support, will be between the Raytheon Automated Placement and Interconnect Design System (RAPIDS) and the ROSE database. RAPIDS will provide a graphical CAD environment for displaying and manipulating the PWB product design. The interface will be bi-directional to support manipulation of the data within ROSE and subsequent re-use by RAPIDS.

An information model, representing PWB design and manufacturing data, was developed with the EXPRESS information modeling language used by the PDES/STEP standards. The existing RAPIDS data dictionary was used as the basis for this PWB EXPRESS model.

Using Object-Oriented Analysis (OOA) techniques, the internal requirements were modeled and organized into five major subject areas: process analyzer, guidelines analyzer, yield & rework analyzer, cost estimator, and manufacturing advisor. The process analyzer will select the process sequence required to manufacture the product design. The guideline analyzer will evaluate a design against a set of design for manufacturing guidelines. The yield and rework analyzer will calculate yield and rework rates for a selected process sequence, associated with a product design, on an operation by operation basis. The cost estimator will calculate the recurring manufacturing cost for each operation of the process sequence. The manufacturing advisor will analyze the data generated by the individual analyses, and guide the negotiation/trade-off process by identifying major cost drivers and guideline violations. It also recommends design alternatives based on the influence of the design parameters on the cost analysis.

Raytheon will continue development of MO during the next quarter based on the "Functional Requirements and Measure of Performance For the Manufacturing Optimization (MO) System" document developed during the reporting period. Raytheon will continue development of MO during the next quarter based on the functional requirements. An initial prototype will be developed during the next quarter with a target demonstration date in September 1992. Raytheon is also in the process of developing the Design Specification which will be delivered during the fourth quarter of 1992. Raytheon attended the DARPA DICE Phase IV in Progress Review on June 18 - 19, 1992. Appendix III contains the presentation slides from that review.

6. References

1. BR-20558-1, 14 June 1991, DARPA Initiative In Concurrent Engineering (DICE) Manufacturing Optimization - Volume I - Technical.
2. CDRL No. 0002-AC-1, 19 March 1992, Operational Concept Document For The Manufacturing Optimization (MO) System, Contract No. MDA972-92-C-0020.
3. CDRL No. 0002-AC-2, 19 March 1992, Description of Concurrent Engineering Technology For The Manufacturing Optimization (MO) System, Contract No. MDA972-92-C-0020.
4. CDRL No. 0002-AC-3, 31 May 1992, Functional Requirements and Measure of Performance For The Manufacturing Optimization (MO) System, Contract No. MDA972-92-C-0020.

7. Notes

7.1 Acronyms

CAEO	Computer Aided Engineering Operations
CDRL	Contract Data Requirements List
DARPA	Defense Advanced Research Projects Agency
DFMA	Design for Manufacturing and Assembly
DICE	DARPA Initiative In Concurrent Engineering
MO	Manufacturing Optimization
MSD	Missile Systems Division
PWA	Printed Wiring Assembly
PWB	Printed Wiring Board
PWF	Printed Wiring Fabrication
RAPIDS	Raytheon Automated Placement and Interconnect Design System
ROSE	Rensselaer Object System For Engineering
RSD	RAPIDS Structured Database
RLD	RAPIDS Library Database

Appendix I - Raytheon PWB EXPRESS Model

10. PWB EXPRESS Schemas

10.1 rpd_design.exp

```
-- This is top level schema for the Raytheon PWB EXPRESS model.  
-- The model is primarily derived from Raytheon's Automated Placement  
-- and Interconnect Design System (RAPIDS) data dictionary. RAPIDS is  
-- a concurrent engineering design station for Printed Wiring Boards.  
-- Its database was designed to capture data from many diverse CAE, CAD, CAM, CAT  
-- systems as well as analysis systems for thermal, reliability,  
-- critical signal analysis, and manufacturability.  
-- Emphasis was placed on making the model extremely modular and  
-- flexible.
```

```
-- AUTHOR: T. Laliberty
```

```
LAST MODIFICATION : 22 June 1992
```

```
INCLUDE 'rpdtypes.exp';  
INCLUDE 'rpd_header.exp';  
INCLUDE 'alias.exp';  
INCLUDE 'annotation.exp';  
INCLUDE 'cari.exp';  
INCLUDE 'class.exp';  
INCLUDE 'comment.exp';  
INCLUDE 'dr_block.exp';  
INCLUDE 'gate.exp';  
INCLUDE 'net.exp';  
INCLUDE 'metal_area.exp';  
INCLUDE 'part.exp';  
INCLUDE 'pin.exp';  
INCLUDE 'route.exp';  
INCLUDE 'via.exp';  
INCLUDE 'xref.exp';  
INCLUDE 'shape.exp';  
INCLUDE 'stackup.exp';  
INCLUDE 'model.exp';
```

```
SCHEMA rpd_design;
```

```
REFERENCE FROM rpdtypes_schema;  
REFERENCE FROM rpd_header_schema;  
REFERENCE FROM alias_schema;  
REFERENCE FROM annotation_schema;  
REFERENCE FROM cari_schema;  
REFERENCE FROM class_schema;  
REFERENCE FROM comment_schema;  
REFERENCE FROM dr_block_schema;  
REFERENCE FROM gate_schema;  
REFERENCE FROM net_schema;  
REFERENCE FROM metal_area_schema;  
REFERENCE FROM part_schema;  
REFERENCE FROM pin_schema;  
REFERENCE FROM route_schema;  
REFERENCE FROM via_schema;  
REFERENCE FROM xref_schema;  
REFERENCE FROM model_schema;  
REFERENCE FROM shape_schema;  
REFERENCE FROM stackup_schema;
```

```

ENTITY rpd_design_rec;
  alias_header : header_rec;
  aliases : LIST [0:?] of alias_rec;           -- list of aliases
  annotation_header : header_rec;
  annotations : LIST [0:?] of annotation_rec; -- list of annotations
  cari_header : header_rec;
  cari_rules : LIST [0:?] of cari_rule_rec;   -- list of cari rules
  class_header : header_rec;
  classes : LIST [0:?] of class_rec;         -- list of classes
  comment_header : header_rec;
  comments : LIST [0:?] of comment_rec;      -- list of design comments
  dr_block_header : header_rec;
  dr_blocks : LIST [0:?] of dr_block_rec;     -- list of design rule blocks
  gate_header : header_rec;
  gates : LIST [0:?] of gate_rec;           -- list of gates
  net_header : header_rec;
  nets : LIST [0:?] of net_rec;             -- list of nets
  part_header : header_rec;
  parts : LIST [0:?] of part_rec;          -- list of parts
  pins_header : header_rec;
  pins : LIST [0:?] of pin_rec;           -- list of pins
  route_header : header_rec;
  routes : LIST [0:?] of route_rec;        -- list of routes
  vias_header : header_rec;
  vias : LIST [0:?] of via_rec;           -- list of vias
  xref_header : header_rec;
  xrefs : LIST [0:?] of xref_rec;         -- list of xrefs
  shapes_header : header_rec;
  shapes : LIST [0:?] of pad_shape_rec;     -- list of pad shapes
  stackups_header : header_rec;
  stackups : LIST [0:?] of stackup_rec;     -- list of pad stackups
  models : LIST [0:?] of model_rec;        -- list of part mechanical models
END_ENTITY;

END_SCHEMA;

```

10.2 rpdtypes.exp

```

-- This schema defines types and entities that are used throughout the
-- entire PWB model.

-- AUTHOR : T. Laliberty                      LAST MODIFICATION : 22 June 1992

SCHEMA rpdtypes_schema;

TYPE token = STRING; END_TYPE;

TYPE name_type = STRING; END_TYPE;

TYPE layer_type = STRING; END_TYPE;

TYPE keyword = STRING; END_TYPE;

TYPE dimension = INTEGER; END_TYPE;

TYPE shape_type = STRING; END_TYPE;

TYPE loading_type = REAL; END_TYPE;

TYPE blocking_type = STRING; END_TYPE;

-- BINARY data type is not currently supported by the EXPRESS compiler

```

```
-- Assuming 8 bit characters (256 layers, 1 bit per layer)
   TYPE bitmask = ARRAY [0:31] of STRING(1); END_TYPE;
```

```
ENTITY time_rec;
  high : INTEGER;
  low  : INTEGER;
END_ENTITY;
```

```
ENTITY r_range_rec;
  minimum : REAL;
  maximum : REAL;
END_ENTITY;
```

```
ENTITY i_range_rec;
  minimum : INTEGER;
  maximum : INTEGER;
END_ENTITY;
```

```
ENTITY r_span_rec;
  minimum : REAL;
  maximum : REAL;
  span    : REAL;
END_ENTITY;
```

```
ENTITY i_span_rec;
  minimum : INTEGER;
  maximum : INTEGER;
  span    : INTEGER;
END_ENTITY;
```

```
ENTITY pin_name_rec;
  device : name_type;
  gate   : name_type;
  pin    : name_type;
END_ENTITY;
```

```
ENTITY vertex_rec;
  x : dimension;
  y : dimension;
  radius : dimension;
END_ENTITY;
```

```
ENTITY point_rec;
  x : dimension;
  y : dimension;
END_ENTITY;
```

```
ENTITY loading_rec;
  rated : REAL;
  derated : REAL;
  actual : REAL;
END_ENTITY;
```

```
ENTITY attribute_rec;
  key : keyword;
  value : STRING;
END_ENTITY;
```

```
END_SCHEMA;
```

10.3 rpd_header.exp

```
-- This schema defines entities for the unit and scale of other entity
```

```
-- instances and the creation, access, and modification time entites.
-- AUTHOR : T. Laliberty                LAST MODIFICATION : 22 June 1992
-- INCLUDE 'rpdtypes.exp';

SCHEMA rpd_header_schema;

REFERENCE FROM rpdtypes_schema;

ENTITY version_rec;
  name : NAME_TYPE;
  revision : NAME_TYPE;
END_ENTITY;

ENTITY header_rec;
  file_name : NAME_TYPE;
  version : NAME_TYPE;
  creation : TIME_REC;
  access : TIME_REC;
  modification : TIME_REC;
  unit : NAME_TYPE;
  scale : REAL;
  tool : NAME_TYPE;
  tool_ver : INTEGER;
  tool_rev : INTEGER;
  assembly : version_rec;
  drawing : version_rec;
  codeid : NAME_TYPE;                -- Wire Wrap code id
  comment : STRING;
  attribute : LIST OF ATTRIBUTE_REC;
END_ENTITY;

END_SCHEMA;
```

10.4 alias.exp

```
-- This is the EXPRESS schema for storing data aliases required by
-- limitations of some CAX system (i.e. NET names in one system are
-- restricted to a particular length that has been violated by a system
-- that is upstream in the design process)

-- AUTHOR : T. Laliberty                LAST MODIFICATION : 22 June 1992
-- INCLUDE 'rpdtypes.exp';

SCHEMA alias_schema;

REFERENCE FROM rpdtypes_schema;

ENTITY alias_list_rec;
  rapids_name : NAME_TYPE;
  alias_name : NAME_TYPE;
  object_name : NAME_TYPE;
END_ENTITY;

ENTITY alias_rec;
  object : NAME_TYPE;                -- type of object
  property : NAME_TYPE;              -- object property
  system : NAME_TYPE;                -- system requiring an alias
  alias_list : LIST [0:?] of alias_list_rec;  -- list of aliases
  comment : NAME_TYPE;
END_ENTITY;
```

END_SCHEMA;

10.5 annotation.exp

```
-- This is the EXPRESS model for annotaton data. Currently,  
-- annotation is limited to text.  
  
-- AUTHOR T. Laliberty          LAST MODIFICATION : 22 June 1992  
  
-- INCLUDE 'rpdtypes.exp';  
  
SCHEMA annotation_schema;  
  
REFERENCE FROM rpdtypes_schema;  
  
ENTITY annotation_rec;  
  text : STRING;          -- label  
  text_height : DIMENSION; -- text size  
  text_width : DIMENSION; -- text size  
  line_width : DIMENSION; -- width of text line  
  layer : NAME_TYPE;     -- text layer  
  location : POINT_REC;  -- text location  
  rotation : INTEGER;    -- text rotation  
  justification : NAME_TYPE; -- text justification  
END_ENTITY;  
  
END_SCHEMA;
```

10.6 cari.exp

```
-- This Express model is inplace for Raytheon legacy data for  
-- its proprietary Computer Aided Routing of Interconnect (CARI)  
-- system. As a generic model this should be eleminated.  
  
-- AUTHOR : T. Laliberty          LAST MODIFICATION : 22 June 1992  
  
-- INCLUDE 'rpdtypes.exp';  
  
SCHEMA cari_schema;  
  
REFERENCE FROM rpdtypes_schema;  
  
ENTITY cari_rule_rec;  
  cari_id : NAME_TYPE;    -- keyword for CARI record  
  record : NAME_TYPE;    -- CARI record card image  
  comment : NAME_TYPE;   -- pointer to comment string  
END_ENTITY;  
  
END_SCHEMA;
```

10.7 class.exp

```
-- This EXPRESS model defines data entities for classifyong signal  
-- nets into groups for particular design rules.  
  
-- AUTHOR : T. Laliberty          LAST MODIFICATION : 22 June 1992  
  
-- INCLUDE 'rpdtypes.exp';
```

```

SCHEMA class_schema;

REFERENCE FROM rpdtypes_schema;

ENTITY class_rec;
  group_name : NAME_TYPE;           -- class identifier
  design_rules : NAME_TYPE;         -- design rules block
  signal_list : LIST [0:?] of NAME_TYPE; -- signals in the class
  attribute : LIST [0:?] of ATTRIBUTE_REC; -- user defined attribute
  comments : LIST [0:?] of STRING;   -- text description
END_ENTITY;

END_SCHEMA;

```

10.8 comment.exp

```

-- This schema defines a single entity for a comment
-- a list of comments is kept with each PWB design.

-- AUTHOR : T. Laliberty           LAST MODIFIATION : 22 June 1992

-- INCLUDE 'rpdtypes.exp';

SCHEMA comment_schema;

REFERENCE FROM rpdtypes_schema;

ENTITY comment_rec;
  comment : NAME_TYPE;
END_ENTITY;

END_SCHEMA;

```

10.9 dr_block.exp

```

-- This EXPRESS schema defines entities for design rules.
-- Design rules are stored in named blocks. Each block except for the
-- GLOBAL block has a Parent name which it inherits from.

-- AUTHOR : T. Laliberty           LAST MODIFICATION : 22 June 1992

-- INCLUDE 'rpdtypes.exp';

SCHEMA dr_block_schema;

REFERENCE FROM rpdtypes_schema;

ENTITY substrate_block_rec;
  name : NAME_TYPE;                 -- substra = name
  technology : NAME_TYPE;           -- technology code
  mode : INTEGER;                   -- code for mode
  layers : INTEGER;                 -- number of layers
  pad_stack_file : NAME_TYPE;       -- RLD file containing pad stackups
  layer_model : LIST [0:?] of LAYER_TYPE; -- layer model names
  separation : LIST [0:?] of INTEGER; -- spacing between layers
  prepreg_mat : NAME_TYPE;          -- prepreg material
  substrate_mat : NAME_TYPE;        -- substrate material
  solder_mat : NAME_TYPE;           -- solder_mask material
  attribute : LIST [0:?] of ATTRIBUTE_REC; -- user defined attributes
END_ENTITY;

```

```

ENTITY via_spec_rec;
  via_shape : STRING;           -- default via shape
  via_length : DIMENSION;      -- default via length
  via_height : DIMENSION;      -- default via height
END_ENTITY;

ENTITY via_step_rec;
  via_spacing : DIMENSION;     -- minimum via separation
  via_depth : INTEGER;        -- maximum via depth
  first_layer : INTEGER;      -- first stepping layer
  pattern : NAME_TYPE;        -- stepping pattern
  direction : REAL;           -- direction for first step
END_ENTITY;

ENTITY min_space_rec;
  line_to_line : INTEGER;     -- line-to-line spacing
  line_to_pad : INTEGER;     -- line-to-pad spacing
  pad_to_pad : INTEGER;      -- pad-to-pad spacing
  line_to_profile : INTEGER;  -- line-to-profile spacing
  pad_to_profile : INTEGER;   -- pad-to-profile spacing
END_ENTITY;

ENTITY design_block_rec;
  boundary : LIST [0:?] of vertex_rec; -- design rules boundary
  layer_t : LAYER_TYPE;       -- design rules layer
  layer_polarity : NAME_TYPE; -- layer polarity codes
  x_grid : LIST [0:?] of REAL; -- board routing x grid size
  y_grid : LIST [0:?] of REAL; -- board routing y grid size
  grid_offset : POINT_REC;    -- routing grid offset
  x_via_grid : LIST [0:?] of REAL; -- board via x grid size
  y_via_grid : LIST [0:?] of REAL; -- board via y grid size
  via_grid_offset : POINT_REC; -- via grid offset
  spacing : min_space_rec;    -- feature spacing rules
  via_spec : via_spec_rec;    -- pointer to default via
  via_stepping : via_step_rec; -- via stepping data
  acid_trap : INTEGER;       -- acid trap angle
  attribute : LIST [0:?] of ATTRIBUTE_REC; -- user defined attributes
END_ENTITY;

ENTITY miter_rec;
  angle : DIMENSION;          -- mitering angle
  length : I_RANGE_REC;      -- length of miter
END_ENTITY;

ENTITY termination_rec;
  term_type : TOKEN;         -- type of termination (INPUT |
OUTPUT| DUAL)
  value : REAL;              -- resistor value in ohms
  unterm : DIMENSION;       -- max unterminated length
END_ENTITY;

ENTITY necking_rec;
  line_width : DIMENSION;    -- minimum necked width
  length : I_RANGE_REC;     -- length of neck
  spacing : DIMENSION;      -- unnecked spacing between 2 necks
END_ENTITY;

ENTITY parallelism_rec;
  parallel_type : NAME_TYPE; -- total or individual
  plane : NAME_TYPE;        -- coplanar or biplanar
  separation : DIMENSION;   -- separation threshold between traces
  limit : DIMENSION;       -- parallel traces length threshold
END_ENTITY;

```

```

ENTITY shield_rec;
  shield_type : NAME_TYPE;           -- shielding type: microstrip,
  stripline,                          -- grounded, guarded, shielded
                                       -- signal shield connected
                                       -- cover width for shield
                                       -- stripline width
                                       -- isolation dist
                                       -- via post space distance
                                       -- stackup for vias for posts
  signal : NAME_TYPE;
  cover_width : DIMENSION;
  strip_width : DIMENSION;
  isolation : DIMENSION;
  post_spacing : DIMENSION;
  post_stackup : NAME_TYPE;
END_ENTITY;

ENTITY signal_block_rec;
  layers : bitmask;                  -- eligible routing layers
  layer_t : LIST [0:?] of LAYER_TYPE; -- list of layer types
  signal_type : NAME_TYPE;          -- signal type: power, ground, ecl,
  etc.
  line_width : DIMENSION;           -- default wire line width
  line_shape : NAME_TYPE;           -- line aperture_shape
  max_length : DIMENSION;           -- max signal conductor length
  min_length : DIMENSION;           -- min signal conductor length
  stub : DIMENSION;                 -- max stub length
  net_order : NAME_TYPE;            -- stringing algorithm: MST, DAISY,
STAR, WIREWRAP
  route_bias : REAL;                -- routing priority
  clearance : DIMENSION;             -- net isolation distance
  place_bias : REAL;                -- placement priority
  via_type : NAME_TYPE;              -- pad stack for via
  transmission : DIMENSION;          -- max transmission length
  span : DIMENSION;                 -- driver span
  via_count : INTEGER;               -- maximum # of vias
  tolerance : DIMENSION;            -- matched length tolerance
  miter : miter_rec;                 -- corner mitering rules
  termination : termination_rec;     -- terminatin rules
  necking : necking_rec;             -- necking rules
  parallelism : LIST [0:?] of parallelism_rec; -- parallelism rules
  delay_rule : r_span_rec;           -- propagation delay rules
  shield_data : shield_rec;          -- shielding rules
  attribute : LIST [0:?] of ATTRIBUTE_REC; -- user defined attributes
END_ENTITY;

ENTITY layer_block_rec;
  layer_t : LAYER_TYPE;              -- design rules layer
  cu_weight : REAL;                  -- copper weight
  thickness : REAL;                  -- thickness of metal
  impedance : INTEGER;               -- layer impedance
  purpose : NAME_TYPE;               -- user define purpose
  attribute : LIST [0:?] of ATTRIBUTE_REC; -- user defined attributes
END_ENTITY;

ENTITY device_block_rec;
  x_grid : LIST [0:?] of REAL;       -- placement grid size
  y_grid : LIST [0:?] of REAL;       -- placement grid size
  grid_offset : POINT_REC;           -- placement grid offset
  layer_name : LAYER_TYPE;           -- component placement layer
  via_flag : BOOLEAN;                -- via inhibit flag
  location_set : NAME_TYPE;          -- placement location set
  auto_insert : NAME_TYPE;           -- auto insertion code
  technology : NAME_TYPE;            -- device technology
  device_bias : REAL;                -- device affinity
  thermal_bias : REAL;               -- thermal affinity
  space_rule : LIST [0:?] OF NAME_TYPE; -- placement spacing rule
  decoupling : DIMENSION;            -- decoupling distance
  overlap : LIST [0:?] OF NAME_TYPE; -- placement overlap rule

```

```

wire_bond : I_RANGE_REC;           -- wire bonding device rules
aspect : R_RANGE_REC;             -- aspect ratio for resist
heat_sink : NAME_TYPE;           -- heat sink id
attribute : LIST [0:?] of ATTRIBUTE_REC; -- user defined attributes
END_ENTITY;

ENTITY metal_area_block_rec;
pin_clearance : DIMENSION;        -- metal to pin clearance
via_clearance : DIMENSION;        -- metal to via clearance
wire_clearance : DIMENSION;       -- metal to wire clearance
conn_number : INTEGER;            -- connections to each pin
conn_width : DIMENSION;           -- width of pin connections
cutout_flag : BOOLEAN;            -- flag to generate cutouts
suppress_flag : BOOLEAN;          -- unused pad suppression
show_connect : BOOLEAN;           -- show pad connections
default_drill : DIMENSION;        -- default drill size
attribute : LIST [0:?] of ATTRIBUTE_REC; -- user defined attributes
END_ENTITY;

ENTITY dr_block_rec;
block_name : NAME_TYPE;           -- name of design rule block
parent_name : NAME_TYPE;         -- name of parent design rule block
substrate_block : substrate_block_rec; -- substrate rules
design_block : design_block_rec;  -- design rules
signal_block : signal_block_rec;  -- signal rules
layer_block : layer_block_rec;    -- level rules
device_block : device_block_rec;  -- signal rules
metal_area_block : metal_area_block_rec; -- metal area rules
END_ENTITY;

END_SCHEMA;

```

10.10 gate.exp

```

-- This schema defines entities for device gates.
-- AUTHOR : T. Laliberty           LAST MODIFICATION : 22 June 1992
-- INCLUDE 'rpdtypes.exp';

SCHEMA gate_schema;

REFERENCE FROM rpdtypes_schema;

ENTITY gate_package_rec;
component : NAME_TYPE;           -- symbolic component name
gate_no : NAME_TYPE;             -- element number
END_ENTITY;

ENTITY sheet_rec;
num : NAME_TYPE;                 -- sheet number
x_location : REAL;               -- location on sheet
y_location : REAL;               -- location on sheet
END_ENTITY;

ENTITY gate_net_rec;
logic_pin : NAME_TYPE;           -- logical pin name
signal : NAME_TYPE;              -- default net name
END_ENTITY;

ENTITY gate_rec;
instance : NAME_TYPE;            -- gate name (handle)
package : gate_package_rec;      -- package reference

```

```

old_package : gate_package_rec;          -- original package ref
gate_swap_code : NAME_TYPE;             -- swap group name
swap_inhibit : INTEGER;                 -- gate/pin swapability
gate_count : INTEGER;                   -- identical gate/device
sheet : sheet_rec;                       -- schematic location
comment : NAME_TYPE;                     -- pointer to comment string
signal_map : LIST [0:?] of gate_net_rec; -- list of pins and nets
old_signal_map : LIST [0:?] of gate_net_rec; -- list of pins and nets
attribute : LIST [0:?] of attribute_rec; -- user defined attribute
END_ENTITY;

```

END_SCHEMA;

10.11 net.exp

-- This schema defines entites for net signals.

-- AUTHOR : T. Laliberty

LAST MODIFICATION : 22 June 1992

```

-- INCLUDE 'rpdtypes.exp';
-- INCLUDE 'pin.exp';
-- INCLUDE 'via.exp';
-- INCLUDE 'route.exp';
-- INCLUDE 'dr_block.exp';

```

SCHEMA net_schema;

```

REFERENCE FROM rpdtypes_schema;
REFERENCE FROM pin_schema;
REFERENCE FROM via_schema;
REFERENCE FROM route_schema;
REFERENCE FROM metal_area_schema;
REFERENCE FROM dr_block_schema;

```

```

ENTITY ww_pin_data_rec;
  method : NAME_TYPE;          -- installation method
  code : NAME_TYPE;            -- wire type code
  sequence : INTEGER;          -- wrap sequence
  group : NAME_TYPE;           -- wire group
  length : DIMENSION;          -- xs wire length
  findno : NAME_TYPE;          --
  inst_path : STRING;          -- installation path
END_ENTITY;

```

```

ENTITY ww_data_rec;
  run_number : INTEGER;        -- wire wrap run number
  func : NAME_TYPE;            -- net function
END_ENTITY;

```

```

ENTITY ww_pin_pair_rec;
  method : NAME_TYPE;          -- installation method
  code : NAME_TYPE;            -- wire type code
  sequence : INTEGER;          -- wrap sequence
  group : NAME_TYPE;           -- wire group
  length : INTEGER;            -- xs wire length
  findno : NAME_TYPE;          --
  inst_path : NAME_TYPE;       -- installation path
END_ENTITY;

```

```

ENTITY pin_pair_rec;
  t_pin_name : pin_name_rec;   -- to pin name
  f_pin_name : pin_name_rec;   -- from pin name
  t_pin : pin_rec;              -- to pin object

```

```

f_pin : pin_rec;           -- from pin object
pp_index : INTEGER;       -- index to route object
pp : route_rec;          -- pointer to route object
ww_pins : ww_pin_pair_rec; -- wire wrap pin pair data
END_ENTITY;

ENTITY net_rec;
  name : NAME_TYPE;           -- name of net
  design_rules : NAME_TYPE;  -- design rules block
  signal_type : NAME_TYPE;   -- signal type
  pin_pairs : LIST [0:?] OF pin_pair_rec; -- list of pin pairs
  ww_data : ww_data_rec;     -- wire wrap data
  layer : BITMASK;           -- eligible routing layers
  layer_t : LIST [0:?] OF NAME_TYPE; -- list of layer types
  line_width : DIMENSION;    -- line width for routing
  line_shape : NAME_TYPE;    -- line aperture_shape
  max_length : DIMENSION;    -- minimum total wire length
  min_length : DIMENSION;    -- maximum total wire length
  stub : DIMENSION;         -- maximum stub length
  net_order : NAME_TYPE;     -- stringing algorithm
  clearance : DIMENSION;    -- net isolation distance
  route_bias : REAL;        -- routing priority
  place_bias : REAL;        -- placement priority
  via_type : NAME_TYPE;     -- absolute pin(via) type
  transmission : DIMENSION; -- transmission length
  span : DIMENSION;         -- driver span
  via_count : INTEGER;      -- maximum # of vias
  miter : miter_rec;        -- corner mitering rules
  termination : termination_rec; -- terminatin rules
  necking : necking_rec;    -- necking rules
  parallelism : LIST [0:?] OF parallelism_rec; -- parallelism rules
  shield : shield_rec;      -- shielding rules
  pin_names : LIST [0:?] OF pin_name_rec; -- pin names in the net
  pins : LIST [0:?] OF pin_rec; -- pin records in the net
  routes : LIST [0:?] OF route_rec; -- list of net routes
  vias : LIST [0:?] OF via_rec; -- list of net vias
  metal_areas : LIST [0:?] OF metal_area_rec; -- list of net metal areas
  delay_rule : r_span_rec;  -- propagation delay rules
  comment : NAME_TYPE;     -- comment string
  attribute : LIST [0:?] OF ATTRIBUTE_REC; -- user defined attribute
END_ENTITY;

END_SCHEMA;

```

10.12 metal_area.exp

```

-- This schema defines entities for metal areas (areas of a PWB
-- flooded with conductor material).

-- AUTHOR : T. Laliberty           LAST MODIFICATION : 22 June 1992

-- INCLUDE 'rpdtypes.exp';
-- INCLUDE 'dr_block.exp';

SCHEMA metal_area_schema;

REFERENCE FROM rpdtypes_schema;
REFERENCE FROM dr_block_schema;

ENTITY cutout_rec;
  cutout_type : NAME_TYPE;           -- type of cutout
  points : LIST [0:?] OF POINT_REC; -- cutout description
END_ENTITY;

```

```

ENTITY metal_area_rec;
  signal : NAME_TYPE;
  metal_area_type : NAME_TYPE; -- type of metal area
  style : NAME_TYPE; -- style of metal area
  design_rules : dr_block_rec; -- name of design rule block
  aperture : DIMENSION; -- apperature for photoplot
  spacing : DIMENSION; -- line spacing in photoplot
  layer : INTEGER; -- layer for metal area
  cutout_shape : NAME_TYPE; -- shape for pin cutouts
  origin : POINT_REC; -- boundary origin
  boundary : LIST [0:?] of POINT_REC; -- boundary description
  user_cutouts : LIST [0:?] of cutout_rec; -- defined cutouts
  auto_cutouts : LIST [0:?] of cutout_rec; -- generated cutouts
  comment : NAME_TYPE; -- comment string
  attribute : LIST [0:?] of ATTRIBUTE_REC; -- user defined attribute
END_ENTITY;

END_SCHEMA;

```

10.13 part.exp

```

-- This schema defines the electrical characteristics of the PWB
-- components.

-- AUTHOR : T. Laliberty                LAST MODIFICATION : 22 June 1992

-- INCLUDE 'rpdtypes.exp';

SCHEMA part_schema;

REFERENCE FROM rpdtypes_schema;

ENTITY pin_map_rec;
  logic_pin : NAME_TYPE; -- logical pin name
  component_pin : NAME_TYPE; -- component pin name
  pin_swap_code : NAME_TYPE; -- pin swap group
END_ENTITY;

ENTITY element_rec;
  elem_no : NAME_TYPE; -- element number
  elem_swap : NAME_TYPE; -- element Swap Code
  pin_map : LIST [0:?] OF pin_map_rec; -- element to device pin map
END_ENTITY;

ENTITY geo_data_rec;
  rev : NAME_TYPE; -- pin data rev
  modn : NAME_TYPE; -- pin data mod
  clear_z : DIMENSION; -- component CLEARZ
  height : DIMENSION; -- component HEIGHT
  length : DIMENSION; -- component LENGTH
  width : DIMENSION; -- clib component WIDTH
  hsx : DIMENSION; -- clib HSX pin spacing
  hsy : DIMENSION; -- clib HSY pin spacing
  mass : REAL; -- component MASS
  pin_offset : point_rec; -- pin offset
END_ENTITY;

ENTITY op_data_rec;
  rev : NAME_TYPE; -- pin data rev
  modn : NAME_TYPE; -- pin data mod
  power_dissip : REAL; -- power dissipation

```

```

max_power_dissip : REAL;           -- max power dissipation
peak_power : REAL;                 -- peak power
min_power : REAL;                  -- min power
END_ENTITY;

```

```

ENTITY therm_data_rec;
  rev : NAME_TYPE;                 -- pin data rev
  modn : NAME_TYPE;                -- pin data mod
  emit : REAL;
  rsbtm : REAL;
  rsjb : REAL;
  rsjc : REAL;
  rstop : REAL;
  spht : REAL;
  jtm : REAL;
  thermal_type_code : INTEGER;
  thermal_type : NAME_TYPE;
END_ENTITY;

```

```

ENTITY pin_time_rec;
  min : REAL;
  typical : REAL;
  max : REAL;
END_ENTITY;

```

```

ENTITY input_current_rec;
  iil : REAL;                       -- low current
  iih : REAL;                       -- high current
END_ENTITY;

```

```

ENTITY input_voltage_rec;
  vil : REAL;                       -- low voltage
  vih : REAL;                       -- high voltage
END_ENTITY;

```

```

ENTITY output_current_rec;
  iol : REAL;
  ioh : REAL;
  iozl : REAL;
  iozh : REAL;
END_ENTITY;

```

```

ENTITY output_voltage_rec;
  vol : REAL;                       -- low voltage
  voh : REAL;                       -- high voltage
  vol_min : REAL;                   -- min voltage
  voh_max : REAL;                   -- max voltage
END_ENTITY;

```

```

ENTITY bi_pin_rec;
  input_current : input_current_rec;
  input_voltage : input_voltage_rec;
  output_current : output_current_rec;
  output_voltage : output_voltage_rec;
END_ENTITY;

```

```

ENTITY in_pin_rec;
  input_current : input_current_rec;
  input_voltage : input_voltage_rec;
END_ENTITY;

```

```

ENTITY ou_pin_rec;
  ou_config_code : INTEGER;
  ou_config : NAME_TYPE;

```

```

output_current : output_current_rec;
output_voltage : output_voltage_rec;
END_ENTITY;

ENTITY pin_data_rec;
  rev : NAME_TYPE;           -- pin data rev
  modn : NAME_TYPE;         -- pin data mod
  pin_number : NAME_TYPE;   -- component pin number
  pin_name : NAME_TYPE;    -- component pin name
  pin_swap_code : NAME_TYPE; -- pin swap group name
  pin_offset : POINT_REC;  -- center of the pin relative to the
  origin of the device
  capacitance : REAL;
  fall_time : pin_time_rec; -- rise time
  rise_time : pin_time_rec; -- fall time
  pin_type : NAME_TYPE;    -- B, I, O
  bi_pin : bi_pin_rec;    -- bi_directional pin data
  in_pin : in_pin_rec;    -- input pin data
  ou_pin : ou_pin_rec;    -- output pin data
END_ENTITY;

ENTITY prop_delay_rec;
  rev : NAME_TYPE;           -- pin data rev
  modn : NAME_TYPE;         -- pin data mod
  pin_name_start : NAME_TYPE;
  pin_name_end : NAME_TYPE;
  pin_num_start : NAME_TYPE;
  pin_num_end : NAME_TYPE;
  phl : REAL;
  plh : REAL;
  unateness : NAME_TYPE;
END_ENTITY;

ENTITY part_rec;
  part : NAME_TYPE;         -- part name
  technology : NAME_TYPE;  -- device technology
  spice_model : NAME_TYPE; -- spice model for the device
  heat_flag : BOOLEAN;    -- heat sensitivity flag
  stat_flag : BOOLEAN;    -- static sensitivity flag
  polar_flag : BOOLEAN;   -- polar component flag
  part_type : NAME_TYPE;  -- component type
  part_class : NAME_TYPE; -- component class
  description : STRING;   -- component description
  mil_spec : NAME_TYPE;   -- component mil_spec name
  findno : NAME_TYPE;    -- component find number
  tolerance : NAME_TYPE;  -- component tolerance
  value : NAME_TYPE;     -- component value
  mech_name : NAME_TYPE;  -- mechanical name
  manufacturer : NAME_TYPE; -- part manufacturer
  elements : LIST [0:?] OF element_rec; -- list of elements in part
  geo_data : geo_data_rec; -- geometry data
  op_data : op_data_rec;
  therm_data : therm_data_rec; -- thermal data
  pin_data : LIST [0:?] OF pin_data_rec; -- pin data
  delay_data : LIST [0:?] OF prop_delay_rec; -- delay data
  comment : NAME_TYPE;    -- comment string
  attribute : LIST [0:?] OF ATTRIBUTE_REC; -- user defined attributes
END_ENTITY;

END_SCHEMA;

```

10.14 pin.exp

```
-- This schema defines entities for component pins instatiated on
-- the PWB.

-- AUTHOR : T. Laliberty                LAST MODIFICATION : 22 June 1992

-- INCLUDE 'rpdtypes.exp';

SCHEMA pin_schema;

REFERENCE FROM rpdtypes_schema;

TYPE function_type = STRING(1) FIXED; END_TYPE;
    -- I for input or source
    -- O output or sink
    -- B bidirectional
    -- T pin on a terminating resistor

ENTITY load_data_rec;
    power : LOADING_TYPE;                -- power loading data
    voltage : LOADING_TYPE;              -- voltage loading data
    current : LOADING_TYPE;              -- current loading data
    temperature : LOADING_TYPE;          -- temperature loading data
END_ENTITY;

ENTITY pin_rec;
    pin : NAME_TYPE;                    -- pin name
    signal : NAME_TYPE;                 -- signal name
    offset : POINT_REC;                 -- pin offset from origin
    location : POINT_REC;               -- pin location on board
    rotation : REAL;                   -- pin rotation in degrees
    range : BITMASK;                   -- pin depth
    suppression : BITMASK;              -- pad suppression mask
    func : FUNCTION_TYPE;               -- pin function code
    stepping : REAL;                    -- first stepping direction
    pin_type : NAME_TYPE;               -- absolute pin type
    swap_inhibit : INTEGER;             -- gate/pin swapability
    load_data : load_data_rec;          -- pin loading data
    comment : NAME_TYPE;                -- comment string
    attribute : LIST [0:?] of ATTRIBUTE_REC; -- user defined attributes
END_ENTITY;

END_SCHEMA;
```

10.15 route.exp

```
-- This schema defines entities for conductor routes of net signals.

-- AUTHOR : T. Laliberty                LAST MODIFICATION : 22 June 1992

-- INCLUDE 'rpdtypes.exp';
-- INCLUDE 'net.exp';
-- INCLUDE 'pin.exp';

SCHEMA route_schema;

REFERENCE FROM rpdtypes_schema;
REFERENCE FROM net_schema;
REFERENCE FROM pin_schema;

ENTITY segment_rec;
    x : DIMENSION;                      -- x coord of point on the path
    y : DIMENSION;                      -- y coord of point on the path
```

```

radius : INTEGER;                -- for circular segment
segment_width : DIMENSION;      -- the width of the segment
END_ENTITY;

```

```

ENTITY ww_route_data_rec;
revision : NAME_TYPE;           -- wire revision
sequence : INTEGER;            -- wire wrap sequence
bends : LIST [0:?] of POINT_REC; -- wire wrap bend points
END_ENTITY;

```

```

ENTITY route_rec;
signal : NAME_TYPE;            -- associated signal name
route_type : NAME_TYPE;       -- type of connection
status : NAME_TYPE;           -- path status
target_name : pin_name_rec;   -- assigned target pin name
object_name : pin_name_rec;   -- assigned object pin name
target_pin : pin_rec;         -- assigned target pin
object_pin : pin_rec;         -- assigned object pin
target_loc : POINT_REC;       -- coordinates of the target
object_loc : POINT_REC;       -- coordinates of the object
protect : BOOLEAN;            -- path protection flag
target_layer : INTEGER;       -- assigned starting layer
object_layer : INTEGER;       -- assigned ending layer
path : LIST [0:?] OF segment_rec; -- list of path segments
shield_id : INTEGER;          -- code for linking shielding
pin_pair_index : INTEGER;     -- link to pin-pair data
pin_pair : pin_pair_rec;     -- link to pin-pair data
ww_data : ww_route_data_rec;  -- wire wrapping data
comment : NAME_TYPE;
END_ENTITY;

```

```
END_SCHEMA;
```

10.16 via.exp

```
-- This schema defines entities for signal net vias.
```

```
-- AUTHOR : T. Laliberty                LAST MODIFICATION : 22 June 1992
```

```
-- INCLUDE 'rpdtypes.exp';
-- INCLUDE 'dr_block.exp';
```

```
SCHEMA via_schema;
```

```
REFERENCE FROM rpdtypes_schema;
REFERENCE FROM dr_block_schema;
REFERENCE FROM net_schema;
```

```

ENTITY via_rec;
signal : NAME_TYPE;            -- name of signal net
location : POINT_REC;         -- board coordinates
rotation : REAL;              -- via rotation in degrees
range : BITMASK;              -- pin depth
suppression : BITMASK;       -- pad suppression mask
via_type : NAME_TYPE;         -- absolute via type
via_use : NAME_TYPE;         -- special via use
shield_id : INTEGER;          -- code for linking shielding
shield : shield_rec;
comment : NAME_TYPE;          -- comment string
attribute : LIST [0:?] of ATTRIBUTE_REC; -- user defined attributes
END_ENTITY;

```

```
END_SCHEMA;
```

10.17 xref.exp

```
-- This schema defines entites for the device cross-references.

-- AUTHOR : T. Laliberty                LAST MODIFICATION : 22 June 1992

-- INCLUDE 'rpdtypes.exp';

SCHEMA xref_schema;

REFERENCE FROM rpdtypes_schema;
REFERENCE FROM pin_schema;

ENTITY xref_rec;
  symbolic : NAME_TYPE;                -- symbolic name
  old_symbolic : NAME_TYPE;            -- old symbolic name
  model : NAME_TYPE;                   -- mechanical model name
  location : POINT_REC;                -- board location
  mirror : INTEGER;                   -- mirror flag
  rotation : REAL;                     -- rotation flag
  symbolic_flag : BOOLEAN;             -- symbolic pin names used flag
  external : BOOLEAN;                 -- connector flag
  usa_device : NAME_TYPE;.            -- USA device names
  physical : NAME_TYPE;               -- CLIB device name
  raytheon : NAME_TYPE;               -- raytheon part number
  design_rules : NAME_TYPE;          -- design rules block
  layer : NAME_TYPE;                  -- component placement layer
  via_flag : BOOLEAN;                -- inhibit via under device
  location_set : NAME_TYPE;           -- placement location set
  auto_insert : NAME_TYPE;            -- auto insertion code
  swap_inhibit : INTEGER;             -- gate/pin swapability code
  fix : BOOLEAN;                      -- fixed placement flag
  device_bias : REAL;                 -- device affinity
  thermal_bias : REAL;                -- thermal affinity
  coupling : LIST [0:?] of NAME_TYPE; -- placement coupled devices
  decoupling : INTEGER;               -- decoupling distance
  space_rule : LIST [0:?] of NAME_TYPE; -- placement spacing rule
  overlap : LIST [0:?] of NAME_TYPE;  -- placement overlap rule
  heat_sink : NAME_TYPE;              -- heat sink name
  load_data : load_data_rec;          -- loading data
  comment : NAME_TYPE;                -- comment : ring
  attribute : LIST [0:?] of attribute_rec; -- user defined attributes
END_ENTITY;

END_SCHEMA;
```

10.18 shape.exp

```
-- This schema defines entities for pin and via pad shapes.

-- AUTHOR : T. Laliberty                LAST MODIFICATION : 22 June 1992

-- INCLUDE 'rpdtypes.exp';

SCHEMA shape_schema;

REFERENCE FROM rpdtypes_schema;

ENTITY shape_rec;
  shape : NAME_TYPE;                  -- shape type
```

```

width : DIMENSION;           -- aperature width
outline : LIST [0:?] of VERTEX_REC; -- shape description
END_ENTITY;

ENTITY pad_shape_rec;
  name : NAME_TYPE;           -- shape name
  pads : LIST [0:?] of shape_rec; -- pad shapes
END_ENTITY;

END_SCHEMA;

```

10.19 stackup.exp

```

-- This schema defines entities for pin and via pad stackups.
-- Various pad shapes for each layer are combined. The layer
-- assignments are then combined to form the padstack.

-- AUTHOR : T. Laliberty           LAST MODIFICATION : 22 June 1992

--INCLUDE 'rpdtypes.exp';
--INCLUDE 'shape.exp';

SCHEMA stackup_schema;

REFERENCE FROM rpdtypes_schema;
REFERENCE FROM shape_schema;

ENTITY pad_rec;
  pad_name : NAME_TYPE;           -- shape name
  pad_shape : PAD_SHAPE_REC;      -- pad shapes
  func : NAME_TYPE;              -- pad function
END_ENTITY;

ENTITY pad_stack_rec;
  model : NAME_TYPE;             -- layer model
  offset : POINT_REC;           -- pad offset
  pad_list : LIST [0:?] of pad_rec; -- pad_names
END_ENTITY;

ENTITY stackup_rec;
  stack_name : NAME_TYPE;        -- name of stackup
  pad_stack : LIST [0:?] of pad_stack_rec; -- pad stackups
  drill : INTEGER;              -- default drill size
  comments : LIST [0:?] of STRING; -- list of comments
END_ENTITY;

END_SCHEMA;

```

10.20 model.exp

```

-- This schema defines entities for the mechanical model of PWB
-- components.

-- AUTHOR : T. Laliberty           LAST MODIFICATION : 22 June 1992

-- INCLUDE 'rpdtypes.exp';
-- INCLUDE 'rpd_header.exp';
-- INCLUDE 'stackup.exp';

SCHEMA model_schema;

```

REFERENCE FROM rpdtypes_schema;
REFERENCE FROM rpd_header_schema;
REFERENCE FROM stackup_schema;

ENTITY rev_data_rec;
 issue_date : NAME_TYPE; -- date of issue
 revision : NAME_TYPE; -- revision number
 eco : NAME_TYPE; -- latest eco number
 eco_date : NAME_TYPE; -- date of latest eco
END_ENTITY;

ENTITY dev_origin_rec;
 origin_type : NAME_TYPE; -- origin types
 center : POINT_REC; -- device center
 offset : POINT_REC; -- placement offset
 mirror : INTEGER; -- reflection code
END_ENTITY;

ENTITY label_rec;
 text : STRING; -- label text
 height : DIMENSION; -- text size
 width : DIMENSION; -- text size
 location : POINT_REC; -- text location
 rotation : INTEGER; -- text rotation
 line_width : DIMENSION; -- width of text line
 justify : NAME_TYPE; -- text justification
END_ENTITY;

ENTITY boundary_rec;
 boundary_type : NAME_TYPE; -- type of boundary
 shape : NAME_TYPE; -- boundary outline shape
 outline : LIST [0:?] of VERTEX_REC; -- boundary outline vertices
 layers : LIST [0:?] of NAME_TYPE; -- boundary layers
END_ENTITY;

ENTITY obstruction_rec;
 obstruction_type : NAME_TYPE; -- type of obstruction
 shape : SHAPE_TYPE; -- outline shape
 outline : LIST [0:?] of VERTEX_REC; -- pad outline
 layers : LIST [0:?] of LAYER_TYPE; -- pad layers
 blocking : LIST [0:?] of BLOCKING_TYPE; -- blocking codes
END_ENTITY;

ENTITY device_rec;
 symbolic : NAME_TYPE; -- symbolic name
 physical : NAME_TYPE; -- physical name
 model : NAME_TYPE; -- mechanical model name
 location : POINT_REC; -- location on board
 rotation : REAL; -- rotation in degrees
 mirror : INTEGER; -- mirror flag
END_ENTITY;

ENTITY dev_pin_rec;
 physical : STRING; -- physical pin name (must be string of
 integers)
 symbolic : NAME_TYPE; -- symbolic pin name
 location : POINT_REC; -- pin location
 drill : DIMENSION; -- default drill size
 stackup_name : NAME_TYPE; -- pad stackup name
 stackup : STACKUP_REC; -- pad stackup record
 rotation : REAL; -- stackup rotation
 offset : POINT_REC; -- stackup offset
 stepping : INTEGER; -- first stepping direction
END_ENTITY;

```

ENTITY thermal_rec;
  thermal_type : NAME_TYPE;           -- type of thermal relief
  width : DIMENSION;                 -- line width
  spacing : DIMENSION;               -- line spacing
  stackup_name : NAME_TYPE;          -- stackup name
  stackup : STACKUP_REC;             -- stackup record
END_ENTITY;

ENTITY package_rec;
  package_type : NAME_TYPE;          -- package type
  category : NAME_TYPE;              -- package category
  orientation : NAME_TYPE;           -- package orientation
  distance : DIMENSION;              -- pin row separation
  depth : DIMENSION;                 -- package depth
  height : DIMENSION;                -- package height
  width : DIMENSION;                 -- package width
  lead : DIMENSION;                  -- package lead diameter
  fix : BOOLEAN;                     -- fixed device flag
  body_diameter : DIMENSION;         -- package body diameter
  span : DIMENSION;                  -- package pin span
  insert : NAME_TYPE;                -- package insertion code
  mechanical : BOOLEAN;              -- mechanical device flag
  auto_ww_offset : POINT_REC;        -- automatic wirewrap offset
  auto_ww_trp : INTEGER;             -- automatic wirewrap initial trp
  semi_ww_offset : POINT_REC;        -- semiautomatic wirewrap offset
  semi_ww_trp : INTEGER;             -- semiautomatic wirewrap initial trp
END_ENTITY;

ENTITY model_rec;
  header : header_rec;               -- pointer to header record
  mm_name : NAME_TYPE;                -- mechanical model name
  rev_data : rev_data_rec;           -- revision data
  origin : dev_origin_rec;           -- origin data
  package : package_rec;             -- packafing data
  labels : LIST [0:?] of label_rec;  -- list of labels
  boundaries : LIST [0:?] of boundary_rec; -- list of boundaries
  obstructions : LIST [0:?] of obstruction_rec; -- list of obstructions
  devices : LIST [0:?] of device_rec; -- list of devices
  pins : LIST [0:?] of dev_pin_rec;  -- list of pins
  thermals : LIST [0:?] of thermal_rec; -- list of thermal reliefs
  comments : LIST [0:?] of STRING;   -- list of comments
  attribute : LIST [0:?] of attribute_rec; -- list of user defined attributes
END_ENTITY;

END_SCHEMA;

```

Appendix II - MO OOA Model

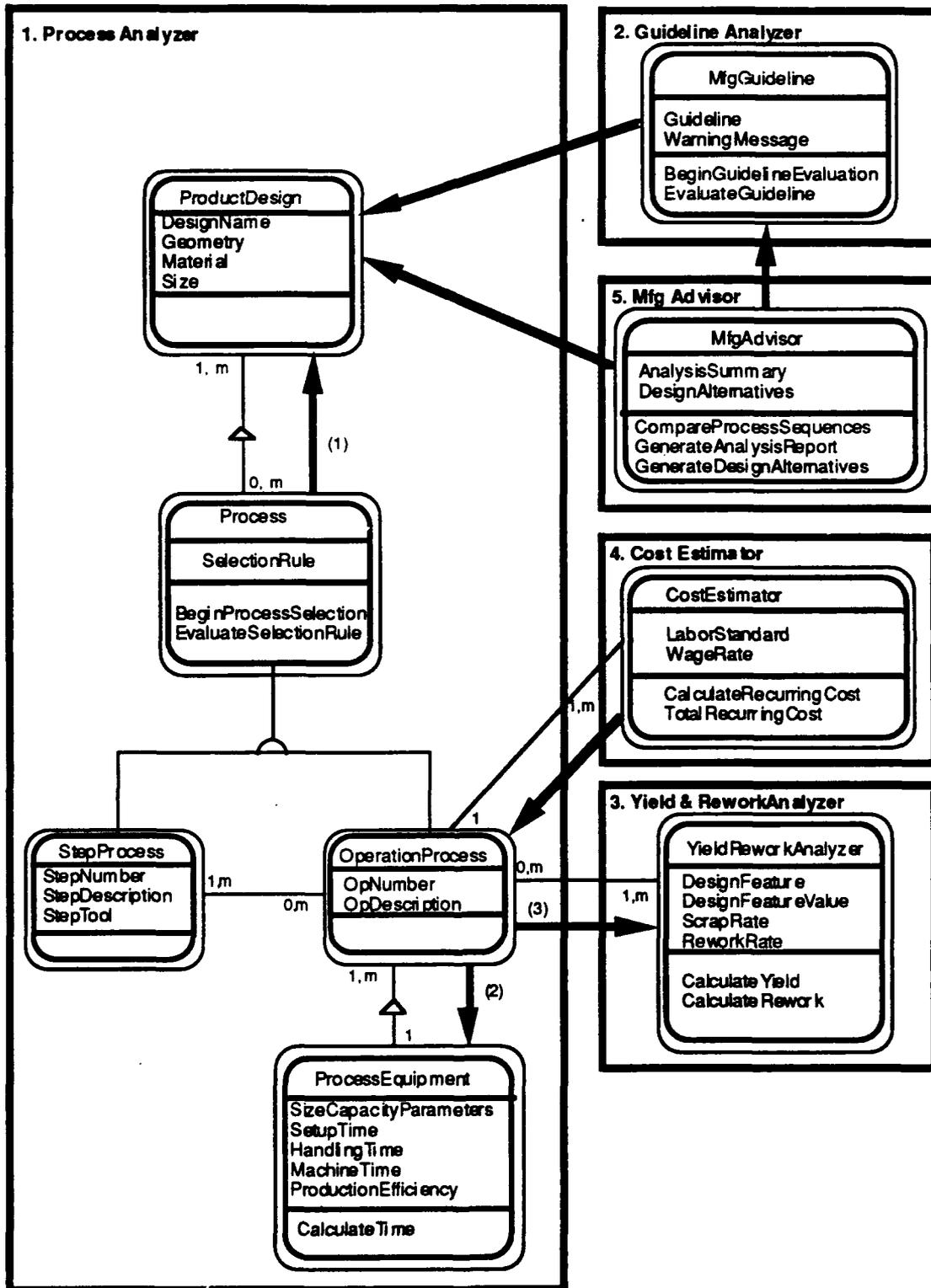


Figure II-1. MO OOA Diagram

Appendix III - Second Quarterly Review Presentation Slides

Raytheon



**DARPA Initiative in Concurrent Engineering (DICE)
Phase 4 In Progress Review**

Manufacturing Optimization (MO)

**Robert Bryant
Raytheon Co., MSD
June 19, 1992**

Raytheon

Manufacturing Optimization (MO)



Agenda

- **Summary of Program Activities**
- **MO Concept Overview**
- **Typical PCB Manufacturing Flow**
- **Manufacturability Issues**
- **High Level PCB Design Flow**
- **Design Cycle Impacts on MFG**
- **Functional Capabilities**
- **Plans**



Summary of Program Activities

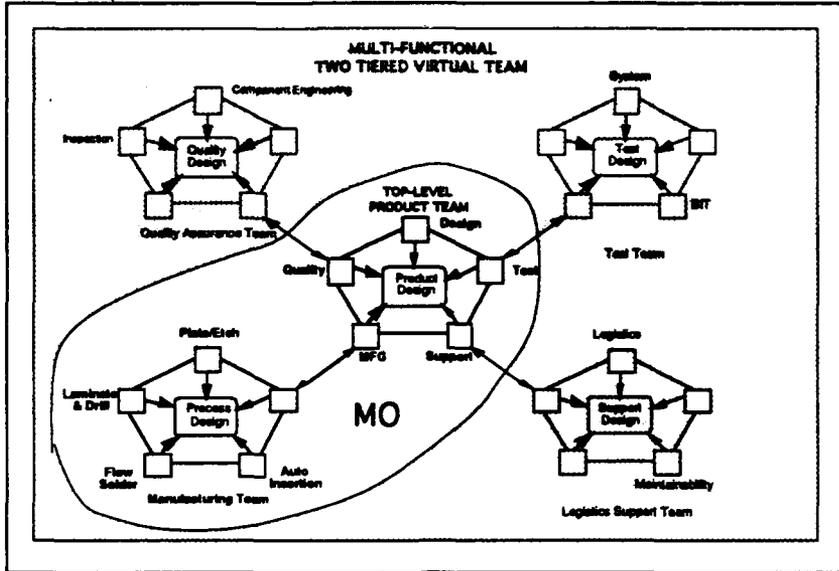
- **Developed Operational Concept**
- **Evaluated DICE Software**
- **Developed Functional Requirements**
- **Developed PCB EXPRESS models based on RAPIDS Database Structure and Process Data Structure**



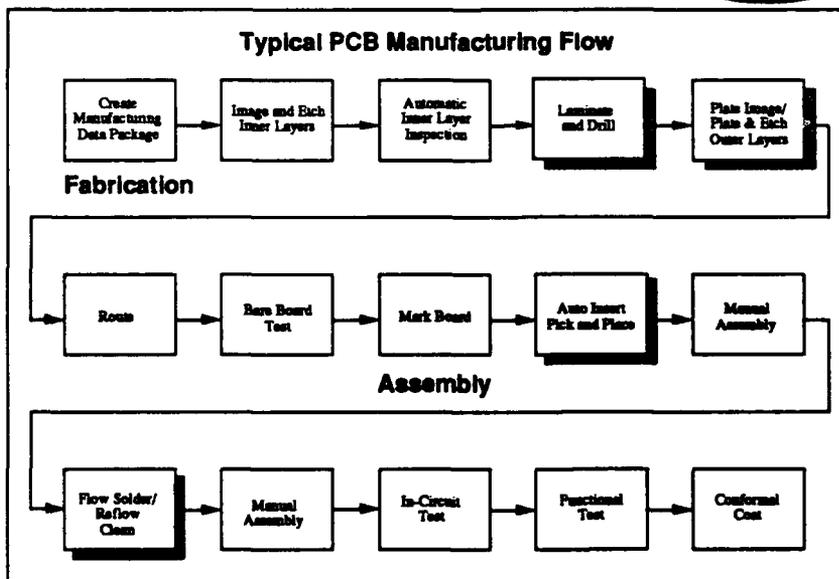
MO Concept

- **Facilitate a two tier team approach to product/process development.**
 - **Product Design is analyzed by multiple manufacturing engineers**
 - **Product/Process changes are traded concurrently in the product and process domains**
- **Provide a tool set that supports DFMA**
 - **Process Selection Algorithm**
 - **Cost/Yield Estimates**

Raytheon Manufacturing Optimization (MO)



Raytheon Manufacturing Optimization (MO)





Design Attributes That Influence Manufacturing Processes

LAMINATE AND DRILL PROCESS:

Laminate multiple circuit layers under heat and pressure and perform automatic precision drill.

CRITICAL DESIGN ATTRIBUTES:

Laminate

- Blind/Buried Vias
- Number of Layers
- Copper Balance
- Layer Stackup
- Board/Laminate Thickness
- Impedance Control Requirement
- Laminate/Prepreg Material
- Board Dimensions

Drill

- Number of Layers
- Pad Size/Accuracy
- Hole Size/Aspect Ratio
- Board Materials
- Board Thickness
- Unused Pad Removal
- Minimum Annular Ring
- Number of Holes, Sizes



Design Attributes That Influence Manufacturing Processes

IMAGE, PLATE AND ETCH OUTER LAYERS PROCESSES:

Photographic and chemical plating/etching operation.

CRITICAL DESIGN ATTRIBUTES:

- PTH Diameter/Aspect Ratio
- Available Registration Aids
- Feature Sizes, Spacing, Tolerance
- Material Selection
- Layer Stackup
 - Presence of Interconnect on Outer Layers
 - Position of Ground Planes
 - Metal Balance/Density
 - Outer Laminate Copper Thickness
- Length of Parallel Interconnect Lines



Design Attributes That Influence Manufacturing Processes

AUTO INSERTION/PICK AND PLACE PROCESS:

Automatic insertion of through hole components and attachment of surface mount devices.

CRITICAL DESIGN ATTRIBUTES:

- Insertability of each component type
- Number of components by insertion type
- Component orientation by component type
- Component-component spacing
- Component-obstruction spacing
- Board thickness vs. component lead length
- Lead diameter vs. hole diameter
- Static sensitivity of components
- Sequencer compatibility of components
- Component bonding/attachment method



Design Attributes That Influence Manufacturing Processes

FLOW SOLDER PROCESS:

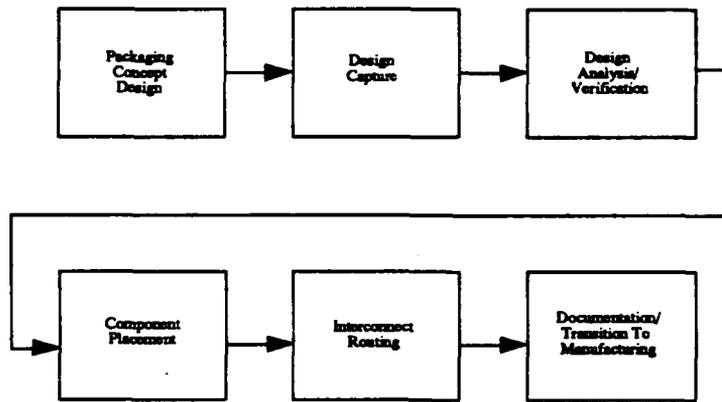
Through-hole components soldered using a "wave" of molten solder and surface mount reflowed using vapor phase, IR, convection, or combination.

CRITICAL DESIGN ATTRIBUTES:

- Board thickness/Lead protrusion
- Thermal sensitivity of components
- Metal Balance
- Component Orientation
- Board Geometry
- Presence of interconnect or ground plane on solder side
- Mixture of surface mount & through hole components
- Pad geometry, feature spacing, and orientation on solder side
- PTH lead diameter
- Aspect ratio of through-hole



High Level PCB Design Flow



Packaging Concept Design

Function(s) Performed:

- Establish Design Concept
- Concept Review

Attributes Determined:

- Number of board types
- Board geometry (area, length, width, aspect ratio)
- Number of interconnect, power, and ground layers
- Layer stackup, layer-layer spacing, and tolerances
- PWB materials
- Design Rules: trace width, space, and via size
- Component family and attachment method
- Thermal management
- Test Strategy
- Cost, performance, and reliability budgets



Design Capture

Function(s) Performed:

- Define Design Details
- Capture Engineering Schematic
- Detail Design Review
 - Functional
 - Manufacturability Assessment
 - Thermal Analysis
 - Testability
 - Reliability
- Define/Review Engineering Parts List

Attributes Determined:

- Component Selection
- Schematic Capture
- Thermal, static, and noise sensitivity
- Component mounting/attachment method
- Circuit complexity and packaging density
- Operating frequency and clock rate



Design Analysis/Verification

Function(s) Performed:

- Perform Simulation
- Select Components
- Signal Analysis
- Load Analysis
- Write Detail Design Memo
- Write Preliminary Test Requirement

Attributes Determined:

- Timing and Fault Grading
- Critical Signals
- Preliminary Test Requirement Specification



Component Placement & Analysis

Function(s) Performed:

- Process Schematic Database
- Define Component Distribution
- Define Power Dissipations
- Establish Placement Concept
- Perform Engineering Placement
- Review Placement

Attributes Determined:

- Component Location and Orientation
- Refine Design Rules



Interconnect Routing

Function(s) Performed:

- Route Critical Signals
- Automatic Routing
- Back Annotation
- Electronic Route Check
- Drafting Check
- Engineering Route Review

Attributes Determined:

- Routing
- Design Rules: trace width, space, and via size
- Number of Layers
- Layer Stackup



Documentation/Transition To Production

Function(s) Performed:

- Documentation Generation
- Post Processor Database Output
- Non-Recurring Engineering Prep. Costs
- Production Readiness Review
- Release to CM

Attributes Determined:

- Schematic Plot
- Parts List
- Master Artwork
- Drill Drawing
- Assembly Drawing
- Drill Tapes
- Inspection Tapes
- Bare-Board Test Tapes



MO Functional Requirements

External Interface Requirements

- Two Tiered Virtual Tiger Team Communications
- CAD Database Interface

MO Capability Requirements

- Process Analyzer
- Guideline Analyzer
- Yield & Rework Analyzer
- Cost Estimator
- Manufacturing Advisor



External Interface Requirements

Two Tiered Virtual Tiger Team Requirements

- **Product - to - Process Team Communication**
 - Notification of design task completed or other pertinent status information.
 - Notification and issuance of database available for analysis.
 - Notification of alternative designs or trade-offs decisions under consideration.
- **Process - to Product Team Communication**
 - Notification and issuance of analysis results.
 - Notification and issuance of modified database with recommended changes.
 - Notification of changes to the process, guidelines, cost or yield models.



External Interface Requirements

CAD Database Interface Requirements

- PWB product design data will be stored in ROSE
- RAPIDS will provide graphical CAD environment for displaying and manipulating PWB product design.
- MO will support an interface from RAPIDS to ROSE.
- RAPIDS to ROSE interface will be bi-directional to support manipulation of product data in ROSE and subsequent re-use in RAPIDS.



Capability Requirements

Process Analyzer

- Select the process sequence required to manufacture the product design based on product design or process parameters.
- Represents manufacturing process by three levels of abstraction: process, operation, and operational step.
- Process selection rules will be represented as "if - then" structures.

Sample Fabrication Process Data

Selection Rule: IF number of layers > 2 THEN

Opno	Op Description	LGrade	Setup	Run	Efficiency
10	"mark part no."	10	0.00000	0.12345	3.12345
20	"pierce tooling holes"	7	0.00000	0.12345	2.12345
30	"oxide treatment"	7	0.12345	0.12345	1.12345
40	"bake panels"	10	0.12345	0.23456	2.12345



Capability Requirements

Guideline Analyzer

- Evaluate a design against a set of design for manufacturing guidelines.
- Manufacturing guidelines will delineate quantitative and/or qualitative manufacturability issues.
- Guideline rules will be represented as "if - then" structures.

Sample Guideline Data

Guideline: IF power/ground layers are not symmetrically positioned in layer stackup THEN

Recommendation: In order to meet the bow and twist specification of less than 0.015 in/in, it is important to have a balanced construction. This means a board stackup should have nearly symmetrical positioning of power and ground planes and interconnect layers with respect to the center-line of the board cross section.



Capability Requirements

Yield & Rework Analyzer

- Calculates yield and rework rates for a selected process sequence associated with a product design.
- Yield and rework rates will be calculated on an operation level within the process sequence.
- Yield and Rework rate for each design feature associated with an operation will be calculated using a look-up table or through evaluation of an equation.

Sample Fabrication Yield Look-Up Table Data

Opno	Design Feature	Value	Scrap Rate
10	"aspect ratio"	5.0	0.05000
20	"aspect ratio"	4.0	0.02000



Capability Requirements

Cost Estimator

- Calculates recurring manufacturing cost for each operation of the process sequence. The following calculations will be performed:
 - Labor standard equations for setup and run time categories.
 - Estimated ideal cost for each operation based on labor standard values and wage rates.
 - Rework operation based on rework rate.
 - Estimated actual cost for each operation.
 - Total estimated ideal and actual cost for each process sequence.



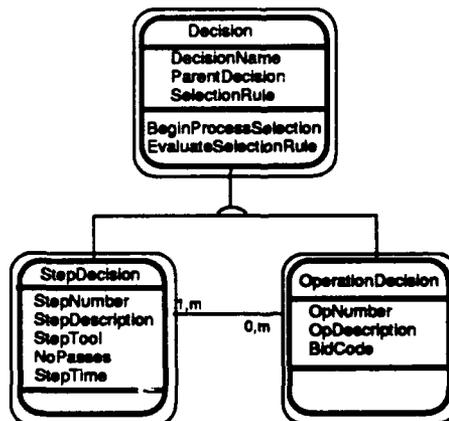
Capability Requirements

Manufacturing Advisor

- Provides viewing of single process analysis results including process sequence, yield & rework, cost, and guidelines.
- Provides a mechanism for comparing and displaying the results from two runs of an analysis on a single process sequence.
- Provides the capability to summarize design features causing manufacturing guideline violations across multiple processes.
- Reports recommendations associated with guideline violations.
- Provides a summary report, identifying cost drivers, for each process contributing to a multi-process analysis for a given product design.



Process Model OOA Diagram Segment



**Simple Process Model EXPRESS Schema**

```
SCHEMA process_model;
  ENTITY Decision;
    name: STRING;
    parent: STRING;
    rules: LIST [0:?] OF STRING;
  END_ENTITY;
  ENTITY Operation
    SUBTYPE OF (Decision);
    op_desc: STRING;
    bid_code: INTEGER;
  END_ENTITY;
  ENTITY Step
    SUBTYPE OF (Decision);
    step_no: INTEGER;
    step_desc: STRING;
    cutting_tool: STRING;
    no_passes: INTEGER;
    step_time: REAL;
  END_ENTITY;
  ENTITY Process;
    sequence: LIST [0:?] OF Decision;
  END_ENTITY;
END_SCHEMA;
```

**Sample ROSE Program - Reads in Process Plan Database**

```
/* Include headers for ROSE Library and program classes */
#include <stdio.h>
#include "rose.h"
#include "process_model.h"

declare(List,Process);
implement(List,Process);

main()
{
  int x = 0;
  int x1 = 0;
  ListOfProcess dList;
  ListOfDecision *seq;

  /** Read in the Process Plan Database File **/
  ROSE.useDesign("selected_processes");
  ROSE.findObjects (&dList);

  int n = dList.size();
  printf("N = %d\n", n);
  for (int i=0; i< n; i++) {
    seq = dList[i]->sequence();
    for (int j=0; j< seq->size(); j++)
      printf("Process%d = %s\n", i+1, (*seq)[j]->name());
  }
  ROSE.display();
}
```



Plans For Next Period

- **Start MO Design**
- **Build (1st Pass) Prototype**
- **PCB/CM Evaluation**
- **RM Evaluation**
- **Interface to IPO on EXPRESS Models**

Distribution List

DPRO-Raytheon
C/O Raytheon Company
Spencer Lab., Wayside Ave.
(one copy of each report)

Defense Advanced Research Projects Agency
ATTN: Defense Sciences Office; Dr. H. Lee Buchanan
Virginia Square Plaza
3701 N. Fairfax Drive
Arlington, VA. 22203-1714
(one copy of each report)

Defense Advanced Research Projects Agency
ATTN: Electronic Systems Technology Office; Capt. Nicholas J. Naclerio, USAF
Virginia Square Plaza
3701 N. Fairfax Drive
Arlington, VA. 22203-1714
(one copy of each report)

Defense Advanced Research Projects Agency
ATTN: Contracts Management Office; Mr. Donald C. Sharkus
Virginia Square Plaza
3701 N. Fairfax Drive
Arlington, VA. 22203-1714
(one copy of each report)

Defense Technical Information Center
Building 5, Cameron Station
ATTN: Selections
Alexandria, VA 22304
(two copies of each report)