Chapter 4.   Design of Systems
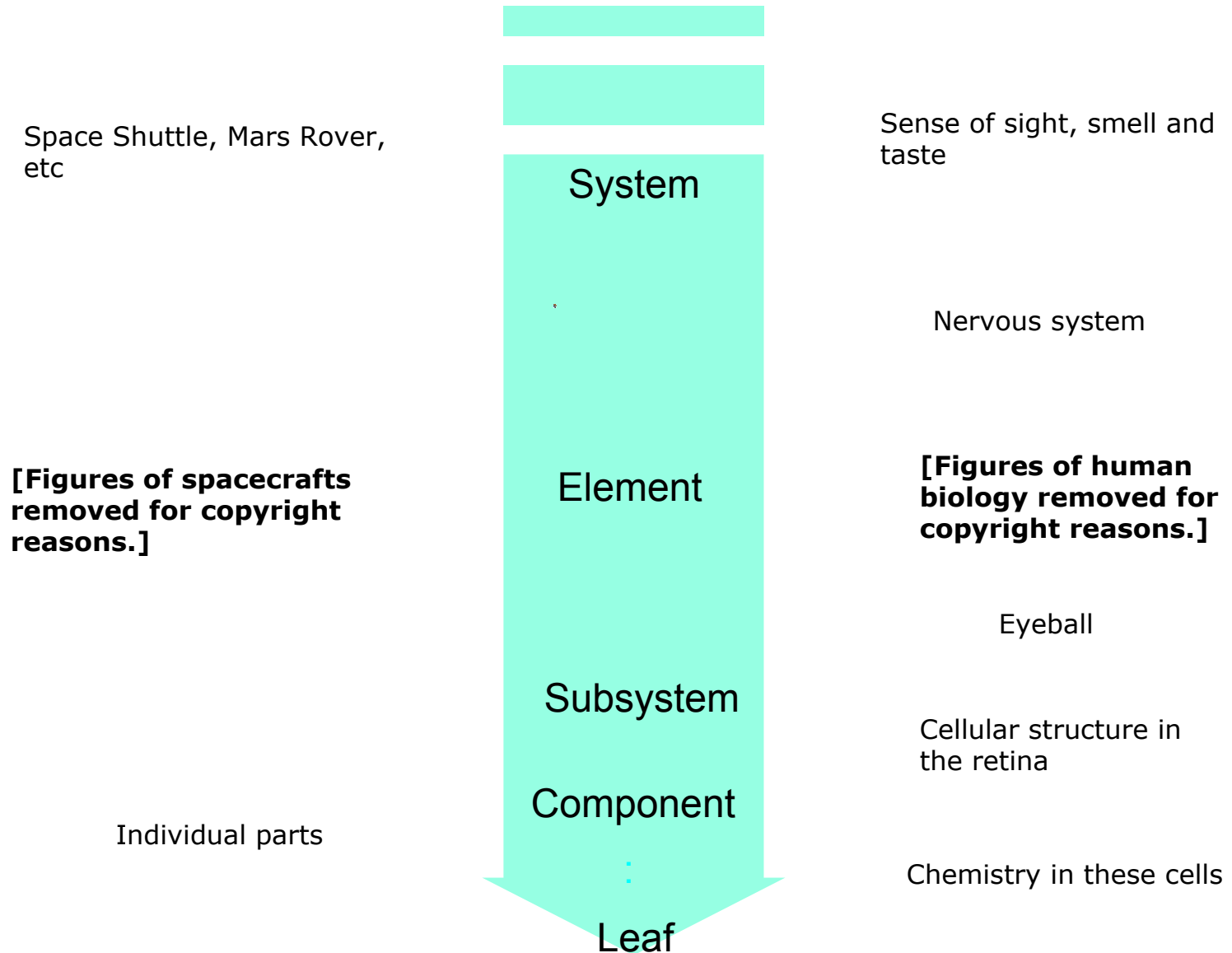
Introduction

What is a system?

CMP machine
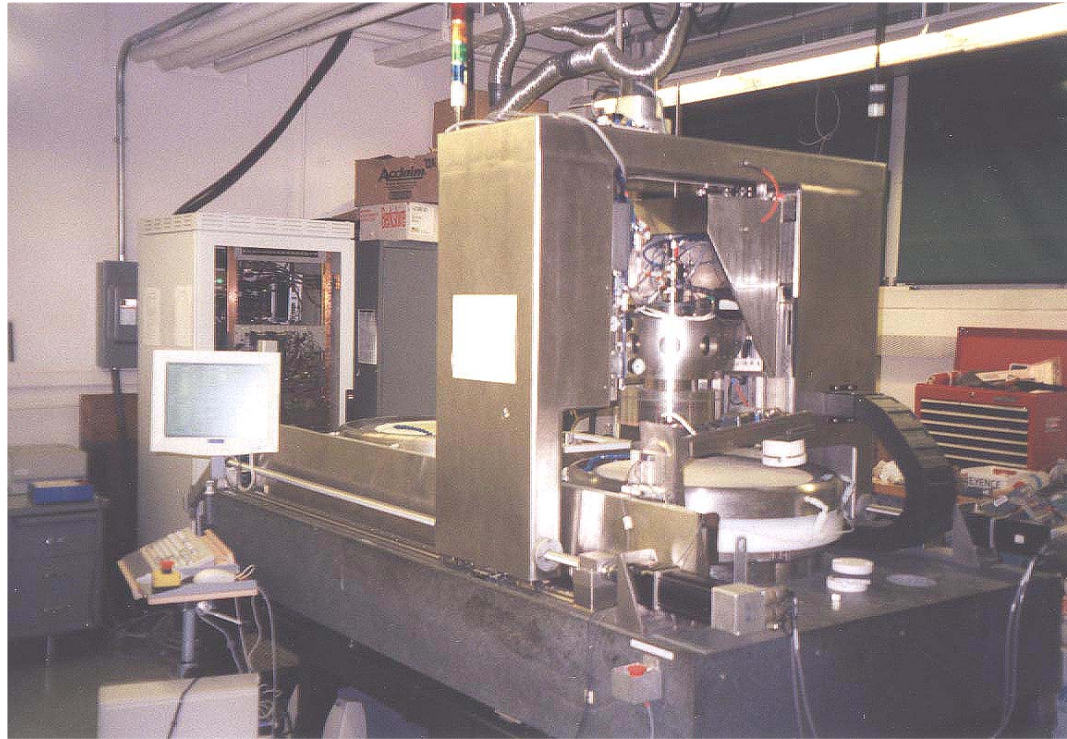
Engineering Systems & Bio System

Space Shuttle, Mars Rover, etc

Sense of sight, smell and taste

System

Nervous system

[Figures of spacecrafts removed for copyright reasons.]

Element

[Figures of human biology removed for copyright reasons.]

Eyeball

Subsystem

Cellular structure in the retina

Component

Individual parts

Chemistry in these cells

Leaf

Definition of a System:

An assemblage of

sub-systems,

hardware and software components, and

people

designed to perform a set of tasks so as to

satisfy the specified FRs

and constraints.

Examples of Systems

Software

Machines

Manufacturing systems

Materials

Products

Government

Issues Related to System Design

1. How should a complex system be designed?

2. How should the complex relationships between various components of a system be coordinated and managed?

Issues Related to System Design

3. How can the stability and controllability of a system be guaranteed?

4. What is the role of human operators in a system?

Classification of Systems

Are all systems alike?

How should we classify systems?

Classification of Systems

Should the system be classified based on the *physical* size of the system?

or

Should it be based on the number and nature of the *functions*

that the system must perform?

Classification of Systems

Why does the classification of systems

based on *functions*

rather than physical size

make more sense?

Classification of Systems

- Large systems from Small systems

- Static systems from Dynamic systems

- Fixed systems from Flexible systems

- Passive systems from Active systems

- Open systems from Closed systems

Axiomatic Design Theory
for Fixed Systems

Is the design of systems different from the design of other things?

Axiomatic Design Theory
for Fixed Systems

i. The First Step in Designing a System: Define FRs of the System

ii. Mapping between the Domains: a Step in Creating System Architecture

iii. The Independence of System Functions

iv. Information Content for Systems: the Best Design

v. Decomposition

vi. System architecture

Measurement of Information content

How do we measure the information content of a system that has many decomposed layers in its hierarchy?

Information Content of Systems

How do we measure the information content of a system that has many decomposed layers in its hierarchy?

$$I_{system} = \Sigma \ I_{\text{highest level FRi}} \ = \ - \ \Sigma \ \log \ (A_c)_{\text{highest level FRi}}$$

where $(A_c)_{\text{highest level Fri}}$ is the area of the common range associated with each one of the highest level FRi.

Information Content of a Large System

$$I_{system} = \Sigma\ I_{highest\ level\ FRi} = -\ \Sigma\ \log\ (A_c)_{highest\ level\ FRi}$$

where $(A_c)_{highest\ level\ FRi}$ is the area of the common range associated with each one of the highest level FRi.

How do we determine I_{system} when $\Sigma\, I_{highest\ level\ FRi}$ is not known?

$$I_{system} = \Sigma \log(p_{leaf}) = -\Sigma \log (A_c)_{leaf}$$

where $(A_c)_{leaf}$ is the area of the common range associated with each leaf.

$$\Sigma \log (A_c)_{leaf} = \Sigma \log (A_c)_{highest\ level\ FRi}$$

In the case of a coupled design, it is expected that in most cases,

$$\Sigma \log (A_c)_{leaf} < \Sigma \log (A_c)_{highest\ level\ FRi}$$

since any change in any other FR in the same set of FRs at a given level will affect the A_c.

Information associated with physical integration (i.e., assembly)

$$I_{system} = \Sigma \log(p_{leaf}) + I_a = - \Sigma \log (A_c)_{leaf} + I_a$$

where I_a is the information associated with assembly of modules.

$$\Sigma \log (A_c)_{leaf} + I_a < \Sigma \log (A_c)_{highest\ level\ FRi}$$

Definition of Module

$$\left\{ \begin{array}{c} FR1 \\ FR2 \end{array} \right\} = \left[\begin{array}{cc} a & 0 \\ b & c \end{array} \right] \left\{ \begin{array}{c} DP1 \\ DP2 \end{array} \right\}$$

Definition of Module --- M_i

$$FR_1 = a\ DP_1 = M_1 * DP_1$$
$$FR_2 = b\ DP_1 + c\ DP_2 = M_2 * DP_2$$

where $M_2 = b\ (DP_1 / DP_2) + c.$

Definition of Modules

$$\Delta FR1 = a\ \Delta DP1 = M1 * \Delta DP1$$

$$\Delta FR2 = b\ \Delta DP1 + c\ \Delta DP2 = M2 * \Delta DP2$$

Decomposition of {FRs}, {DPs}, and {PVs}

How does the decomposition process affect the outcome of the design process?

Definition S1   (Equivalent Designs)

Two designs are defined to be
"equivalent"

if they satisfy the same set of the highest level FRs within the bounds established by the same set of constraints,

even though the mapping and decomposition process might have yielded designs that have substantially different lower level FRs and all DPs for each of these designs.

Definition S2: (Identical Designs)

Designs that fulfill the same set of the highest level FRs and satisfy the Independence Axiom with zero information content are defined to be *"identical"* if their lower level FRs and all DPs are also the same.

Theorem S1 (Decomposition and System Performance)

The decomposition process does not affect the overall performance of the design if the highest level FRs and Cs are satisfied and if the information content is zero, irrespective of the specific decomposition process.

Theorem S2: (Cost of Equivalent Systems)

Two "equivalent" designs can have a substantially different cost structure, although they perform the same set of functions and they may even have the same information content.

Design and Operation of Large Systems

What is a Large System?

> The telephone system for Boston,
>
> The government bureaucracy,
>
> An assembly plant for automobiles,
>
> A software system that controls nuclear power plants, and
>
> Boeing 747 airplanes

Design and Operation of Large Systems

Is it the physical size, the number of components, or the number of functions that make it large?

What is a Large Flexible System?

Definition of a Large Flexible System

A system is a large flexible system if the total number of FRs that the system must satisfy during its lifetime is large and if at different times, the system is required to satisfy different subsets of FRs.

Axiomatic Design of a Large Flexible System

How do we design a large flexible system?

- Define FRs and Constraints
- Knowledge base -- DPs for FRs
- Develop design concepts -- A set of DPs for the
      design task
- Physical integration
- Develop alternative designs
- Choose the best based on information measure

The *knowledge base* can be structured as follows:

FR1 \$ (DP1^a, DP1^b,......................, DP1^m)

FR2 \$ (DP2^a, DP2^b,......................, DP2^q)

FR3 \$ (DP3^a, DP3^b,......................, DP3^w)

....

....

FRn \$ (DPn^a, DPn^b,......................, DPn^s)

Synthesis of a Large Flexible System

SAAB,
Developer of Defense Airplanes of Sweden

SAAB's Research: Letter from Exec. VP, Professor Billy Fredriksson

We have an interesting ongoing research on product development utilizing your Axiomatic Design Theory. Building on your theory Gunnar Holmberg (PhD-student from Saab) is using AD on high systems architecture level to design systems for life cycle flexibility. This is in order to efficiently add new unknown functionality through the life of the systems.

It would be interesting and valuable to us to discuss this with you. I am planning to go to US in April and plan to be at MIT 23 April. Would you be available at MIT?

Synthesis of a Large Flexible System

Suppose the subsets of FRs change as a function of time as follows:

$$@ \; t = 0, \quad \text{the subsets are}$$
$$\{FRs\}_0 = \{FR1, FR5, FR7, FRn\}$$

$$@ \; t = T_1, \; \{FRs\}_1 = \{FR3, FR5, FR8, FRm\}$$

$$@ \; t = T_2, \; \{FRs\}_2 = \{FR3, FR9, FR10, FRn\}$$

How shall we choose DPs?

System Synthesis through Physical Integration of DPs

How do we combine the lower-level DPs to synthesize the higher-level DPs?
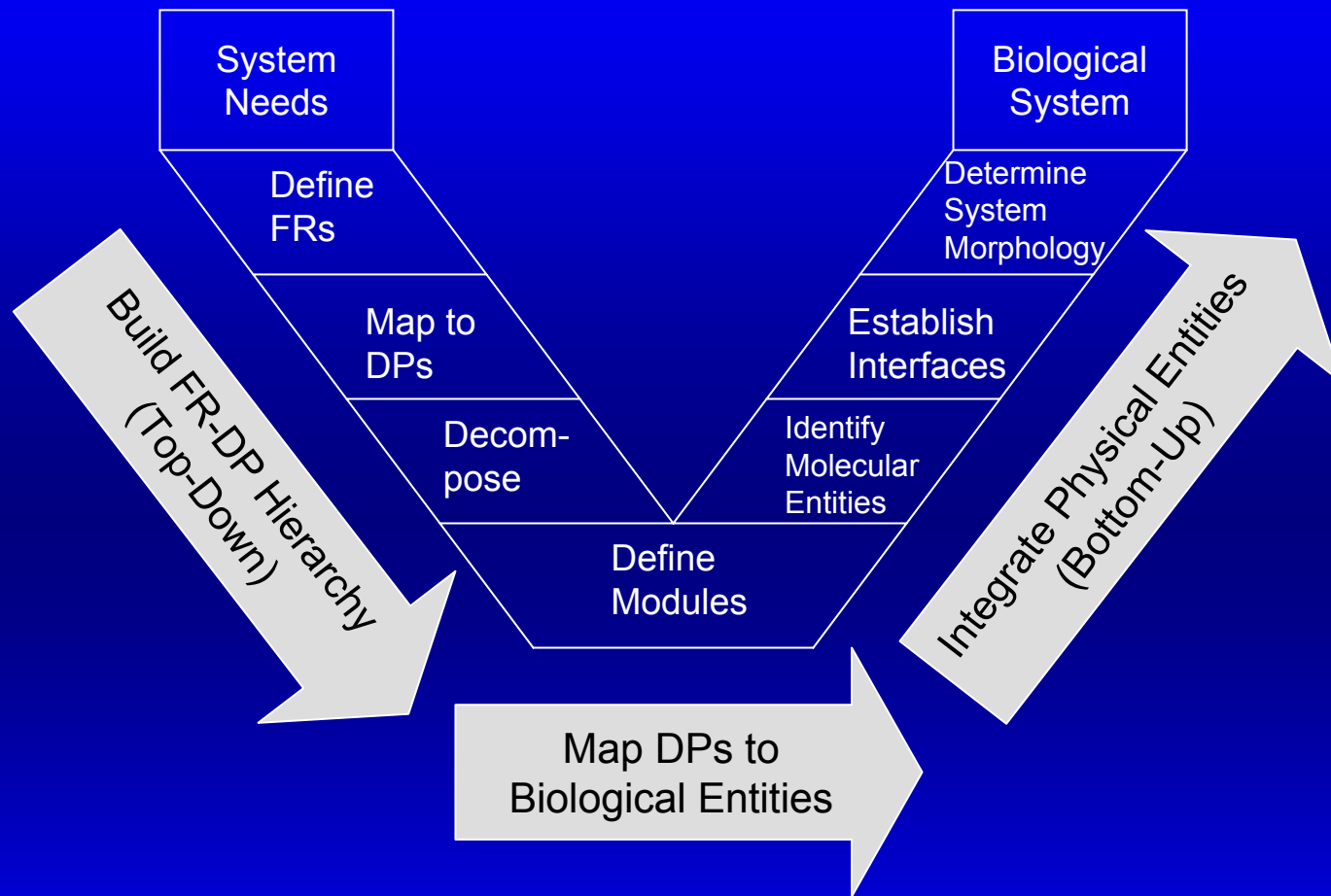
V-Model

Figure1. V-model overview of system analysis using the Design Matrix. The V-Model describes how the Design Matrix of AD is used to study hierarchical nature of biological systems

System Design & Development

TECHNICAL CREDIBILITY

• **NASA Requirements (Level 1 & 2)**

Customer Needs

Define FRs

Map to DPs

Decompose

Define Modules

• **Identifies Lowest Level Requirements & Interactions**

Build FR -DP hierarchy (Top -down)

CREDIBLE COST ASSESSMENT

• **Technical Scope Defined is Scope Estimated**

Detailed system

Satisfy system morphology

Construct local assemblies

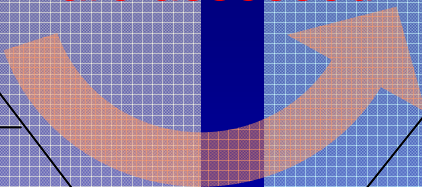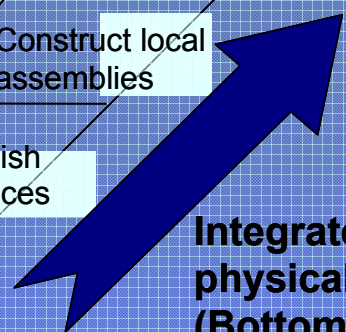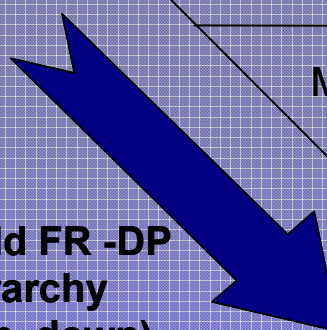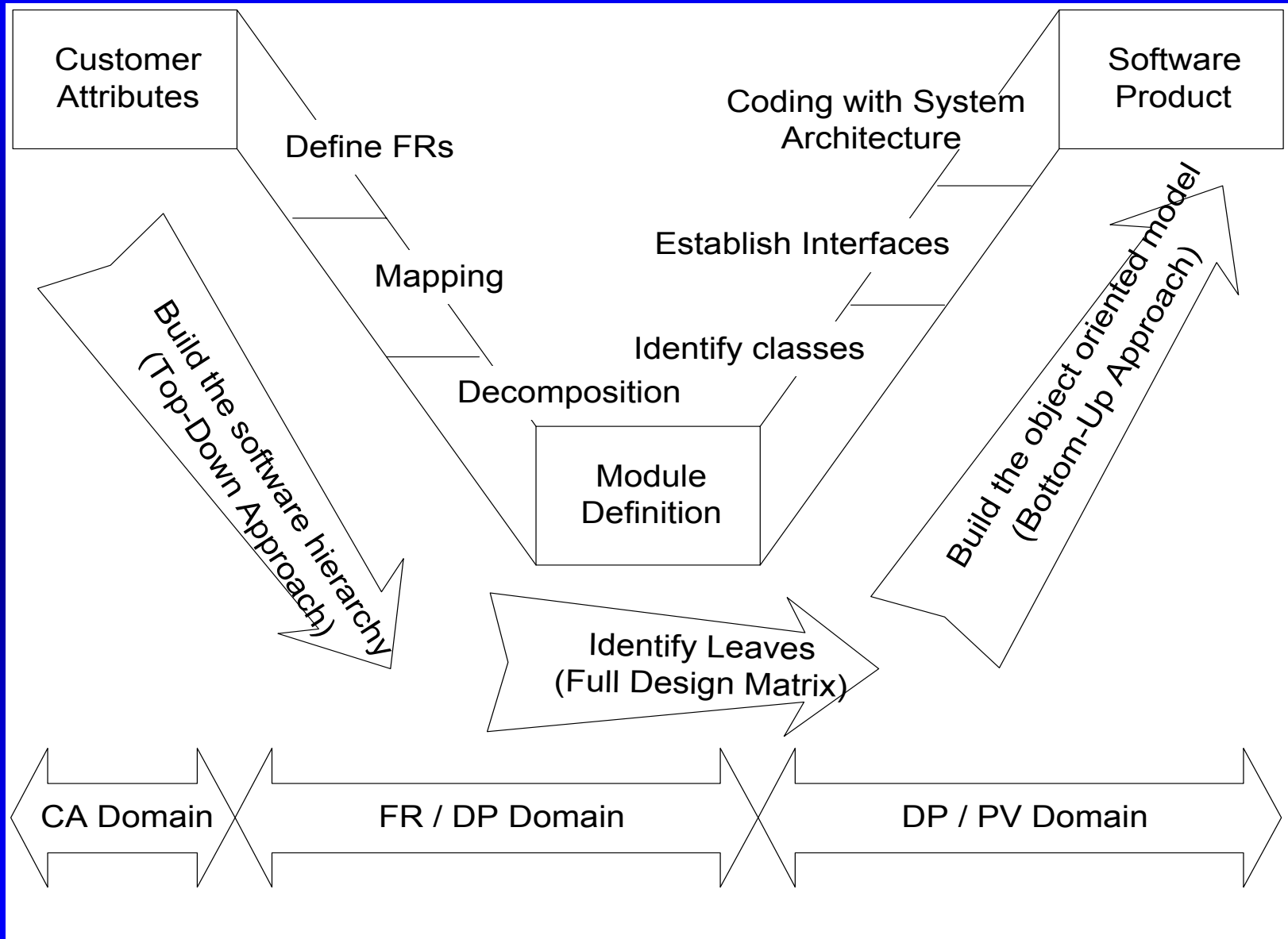Establish interfaces

Identify physical components

• **Estimate the Systems Physical Solutions**

System changes are assessed

Integrate physical entities (Bottom -up)

Mapping DPs into physical entities

The V-Model

Theorem S3   (Importance of High Level Decisions)

The quality of design depends on the selection of FRs and the mapping from domain to domain. Wrong selection of FRs made at the highest levels of design domains cannot be rectified through the lower level design decisions.

Theorem S4   (The Best Design)

**The best design for a large flexible system
that satisfies *n* FRs
can be chosen among the proposed designs
that satisfy the Independence Axiom
if the complete set of the subsets of {FRs}
that the large flexible system must satisfy
over its life
is known *a priori*.**

Theorem S5 (The Need for a Better Design)

When the complete set of the subsets of {FRs}

that a given large flexible system must satisfy

over its life

is not known *a priori,*

there is no guarantee that

a specific design will always have the minimum information content for all possible subsets and thus, there is no guarantee that the same design

is the best at *all times*.

Theorem S6 (Improving the probability of success)

The probability of choosing the best design for a large flexible system increases as the known subsets of {FRs} that the system must satisfy approach the complete set that the system is likely to encounter during its life.

Theorem S7 (Infinite Adaptability versus Completeness)

The large flexible system with an infinite adaptability (or flexibility) may not represent the best design when the large system is used in a situation where the complete set of the subsets of {FRs} that the system must satisfy is known *a priori.*

Theorem S8 (Complexity of a Large Flexible System)

A large system is not necessarily complex if it has a high probability of satisfying the {FRs} specified for the system.

Theorem S9   (Quality of Design)

The quality of design of a large flexible system is determined by the quality of the database, the proper selection of FRs, and the mapping process.

Representation of the System Architecture of Fixed Systems

Is there a need to represent the system architecture?

How do we present it in a concise manner?

Three different but equivalent ways of representing a system:

(1) FR/DP/PV hierarchies with corresponding design matrices,

(2) Module function diagram, and

(3) Flow diagram.

Hierarchies in Design Domains through Decomposition of {FRs}, {DPs}, and {PVs}: Representation of the System Architecture

What is a design hierarchy and how does that represent the system architecture?

Suppose that we have completed a system design such that the FR and the DP hierarchies are:

$$\begin{Bmatrix} FR1 \\ FR2 \end{Bmatrix} = \begin{bmatrix} A11 & 0 \\ 0 & A22 \end{bmatrix} \begin{Bmatrix} DP1 \\ DP2 \end{Bmatrix}$$

$$\begin{Bmatrix} FR11 \\ FR12 \end{Bmatrix} = \begin{bmatrix} X & O \\ X & X \end{bmatrix} \begin{Bmatrix} DP11 \\ DP12 \end{Bmatrix}$$

$$\begin{Bmatrix} FR21 \\ FR22 \\ FR23 \end{Bmatrix} = \begin{bmatrix} X & 0 & 0 \\ X & X & 0 \\ 0 & 0 & X \end{bmatrix} \begin{Bmatrix} DP21 \\ DP22 \\ DP23 \end{Bmatrix}$$

$$\begin{Bmatrix} FR121 \\ FR122 \\ FR123 \end{Bmatrix} = \begin{bmatrix} X & 0 & 0 \\ X & X & 0 \\ X & 0 & X \end{bmatrix} \begin{Bmatrix} DP121 \\ DP122 \\ DP123 \end{Bmatrix}$$

$$\begin{Bmatrix} FR1231 \\ FR1232 \end{Bmatrix} = \begin{bmatrix} a & 0 \\ b & c \end{bmatrix} \begin{Bmatrix} DP1231 \\ DP1232 \end{Bmatrix}$$

Figure removed for copyright reasons.
See Figure 4.1 in Suh, *Axiomatic Design* (2001).

Modules

FR1231 = a DP1231 = M1231 * DP1231

FR1232 = b DP1231 + c DP1232 = M1232 * DP1232

where M1232 = b (DP1231 / DP1232) + c.

Design Matrix and Module-Junction Diagrams – Another Means of System Representation

Since there can be many modules distributed throughout a system, how can we represent the inter-relationship among modules in a system design?
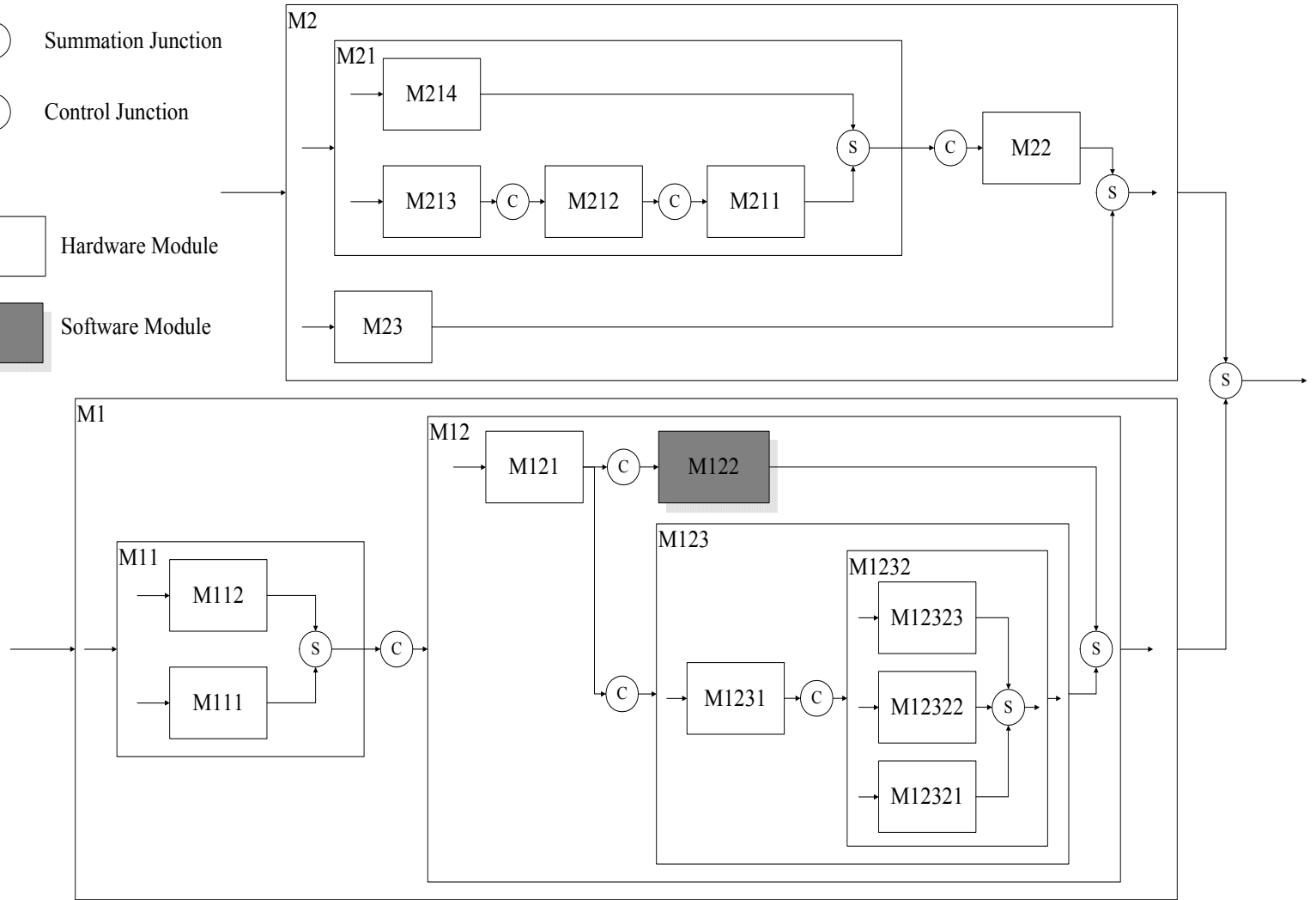
Figures removed for copyright reasons.
See Figures 4.2-4.6 in Suh, *Axiomatic Design* (2001).

S   Summation Junction

C   Control Junction

Hardware Module

Software Module

M2

M21

M214

M213 → C → M212 → C → M211 → S → C → M22 → S

M23

M1

M12

M121 → C → M122

M11

M112

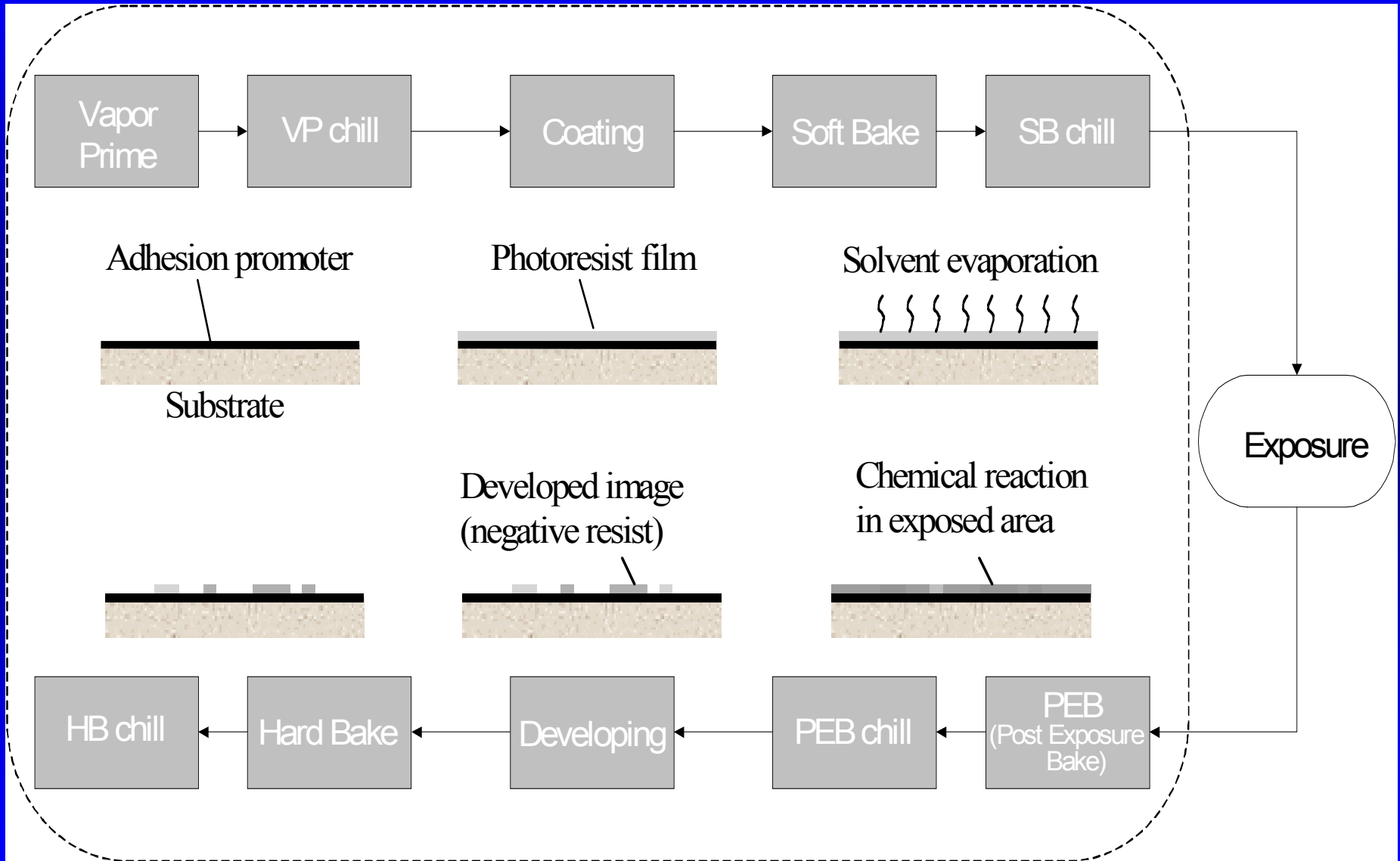M111 → S → C

M123

M1232

M12323

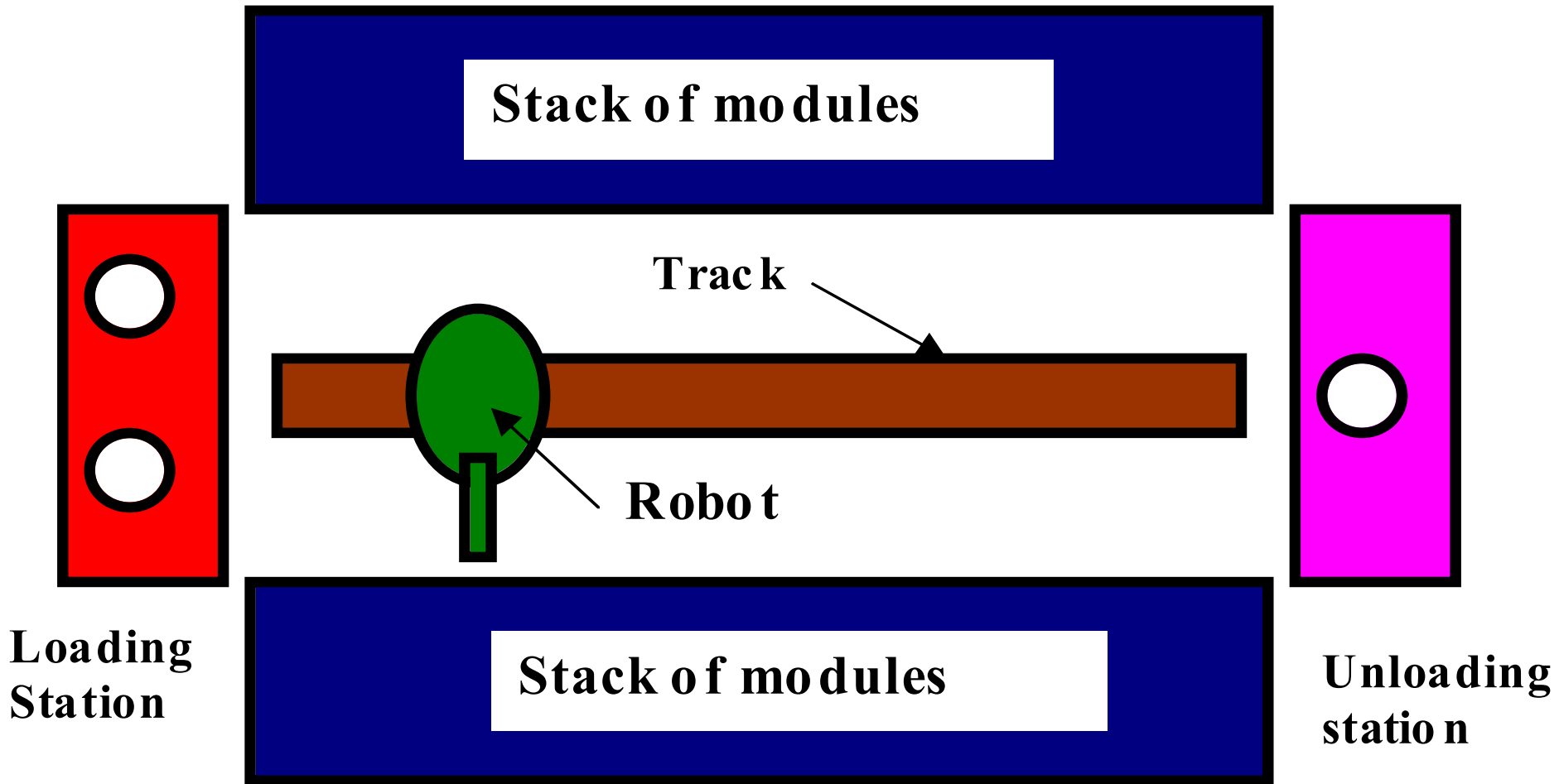M1231 → C → C → M12322 → S

M12321

System Control Command (SCC)

How do we operate a system?

Example 4.1 Design of Wafer Processing Equipment

Example 4.1 Design of Wafer Processing Equipment

Stack of modules

Track

Robot

Stack of modules

Loading Station

Unloading station

Example 4.1 Design of Wafer Processing Equipment

Constraints are:

- Cost
- Footprint
- Reliability
- Safety
- Serviceability
- Manufacturability
- Contamination
- Minimization of wafer temperature variation

The Highest Level FRs, DPs, and the Design Matrix

FR1 = coat wafers with desired resist film at desired throughput rate	X	O	O	X	DP1 = coating process modules
FR2 = develop exposed film at desired rate	O	X	O	X	DP2 = developing process modules
FR3 = transport wafer from input pt. to modules to output pt.	X	X	X	X	DP3 = transport system
FR4 = control the system functions	O	O	O	X	DP4 = system architecture

The design equation has a triangular matrix:

FR4 = control the system functions	X	O	O	O	DP4 = system architecture
FR1 = coat wafers with desired resist film at desired throughput rate	X	X	O	O	DP1 = coating process modules
FR2 = develop exposed film at desired rate	X	O	X	O	DP2 = developing process modules
FR3 = transport wafer from input pt. to modules to output pt.	X	X	X	X	DP3 = transport system

Description of the Decisions Made

FR1 : **Coating thickness = 0.5 – 0.8μm**

Coating uniformity within wafer = 15 A   (3σ)

wafer-to-wafer = 10

cassette-to-cassette(24Hr.) = 13A

FR2 : **Critical Dimension (C.D.)**

within wafer = 0.010μm

wafer-to-wafer = 0.005μm

cassette-to-cassette(24Hr) = 0.010μm

FR3 : **various flow capability**

Transfer time ≤ 10 sec - Overhead time should not be a throughput limiter

FR4 : **process recipe/plan generating high-level command (e.g. On/Off, Accel/Decel)**

The Second Level FRs, DPs, and the Design Matrix

Parent FR : FR1 = coat wafers with desired resist film at desired throughput rate				
Parent DP: DP1 = coating process modules				
Constraints : Decomposition of DP1 must not affect FR2, FR4				
FR11 = prepare wafer for coating	X	O	O	DP11 = thermal process module1
FR12 = coat the wafer with resist	O	X	O	DP12 = (N1) spin coater
FR13 = complete coating process	O	O	X	DP13 = thermal process module2

Description of the Decision Made at the Second Level

FR11 = wafer surface adhesion, surface temperature uniformity before coating, etc. (This will be taken into account at the decomposition of FR11/DP11)

FR12 = various photoresist capability to produce uniform and repeatable film (same spec. as with FR1)
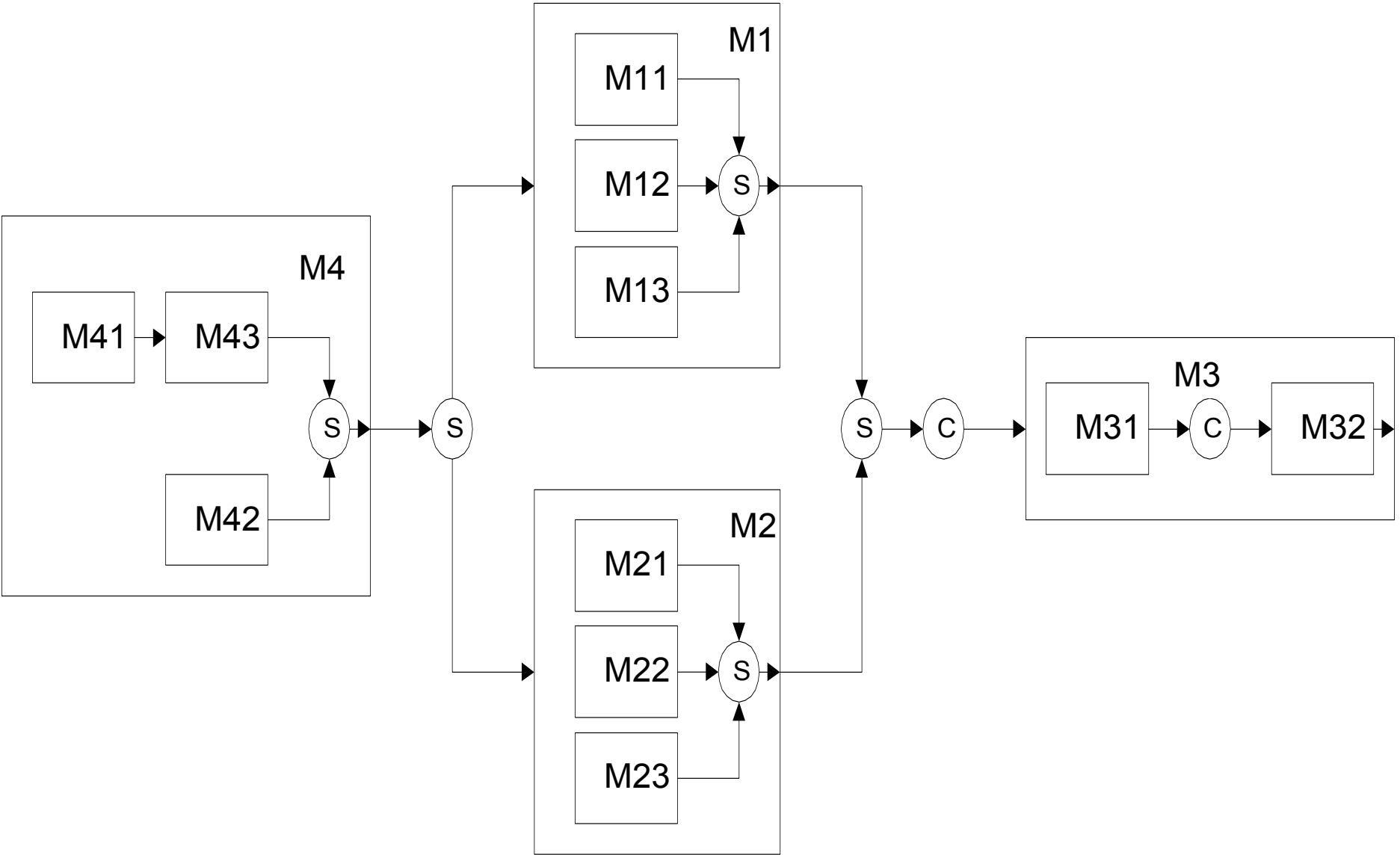
FR13 : final resist film thickness control. (This will be taken into account when FR13/DP13 are decomposed.)

Master Matrix Chart

	DP11	DP12	DP13	DP21	DP22	DP23	DP31	DP32	DP41	DP42	DP43
FR11	X	O	O	O^1	O	O^1	O	O	X	O	X
FR12	O	X	O	O	O^2	O	O	O	X	O	X
FR13	O	O	X	O^1	O	O^1	O	O	X	O	X
FR21	O^1	O	O^1	X	O	O	O^6	O^6	X	O	X
FR22	O	O^2	O	O	X	O	O	O	X	O	X
FR23	O^1	O	O^1	O	O	X	O	O	X	O	X
FR31	X	X	X	X	X	X	X	O	X	O	O
FR32	X^3	O^4	X^3	X^3	O^4	X^3	X	X	O	O	X
FR41	O	O	O	O	O	O	O	O	X	O	O
FR42	O^5	O^5	O^5	O^5	O^5	O^5	O	O	O	X	O
FR43	O^*	O^*	O^*	O^*	O^*	O^*	O	O	X	O	X

Notes :

1. Thermal effect must be considered among thermal process modules - we can use some kinds of thermal shields or we can do that with appropriate layout.

2. Spin module must not affect each other in the sense of vibration, particle generation, etc.

3. The evidence of this X is the utilization of IBTA robot.

4. Standard method for wafer hand-off is required.

5. Flexible or standard way of supervising (sensing)

6. There must be no delay from stepper to thermal process module 3.

Flow chart at the second level

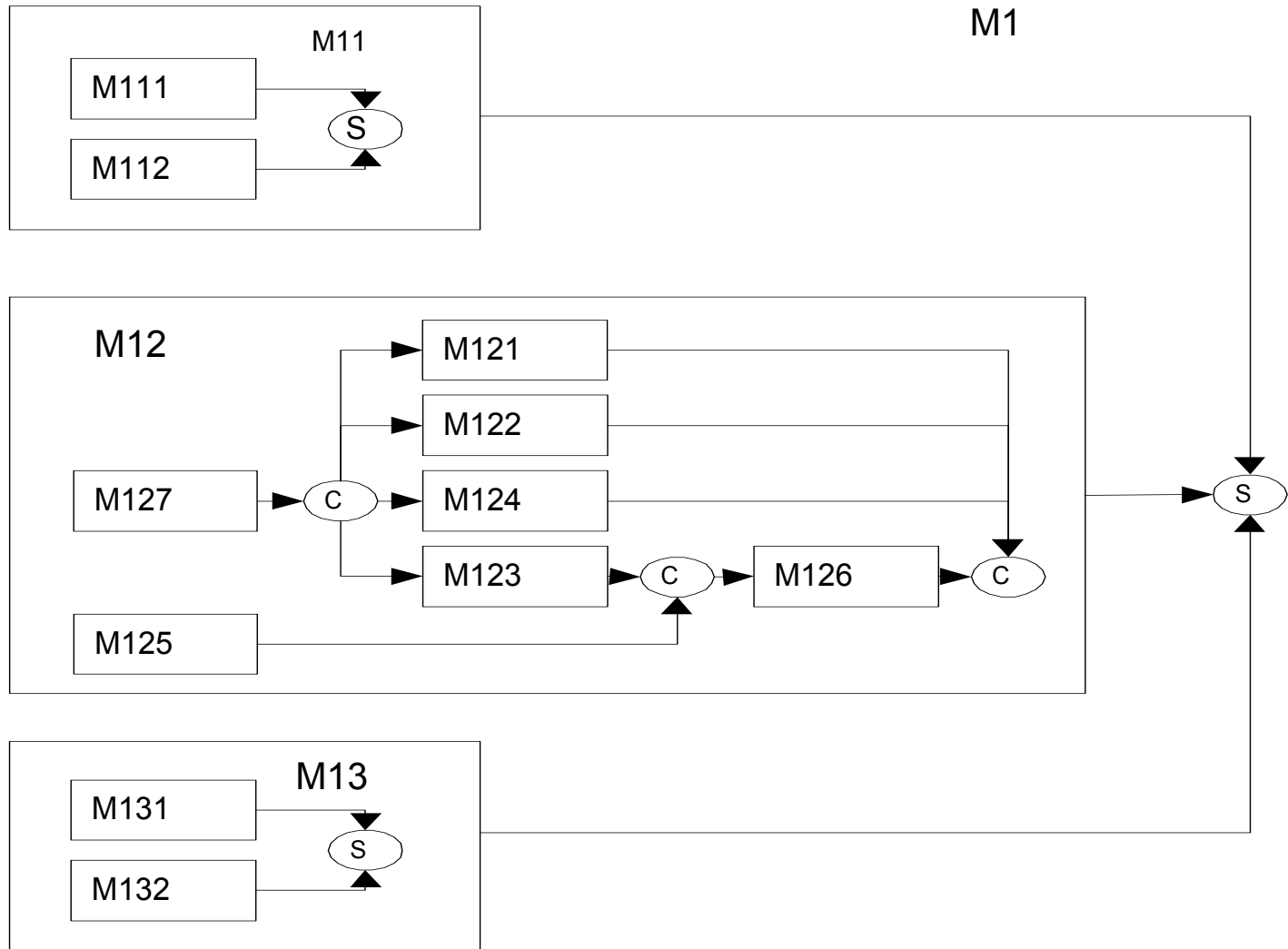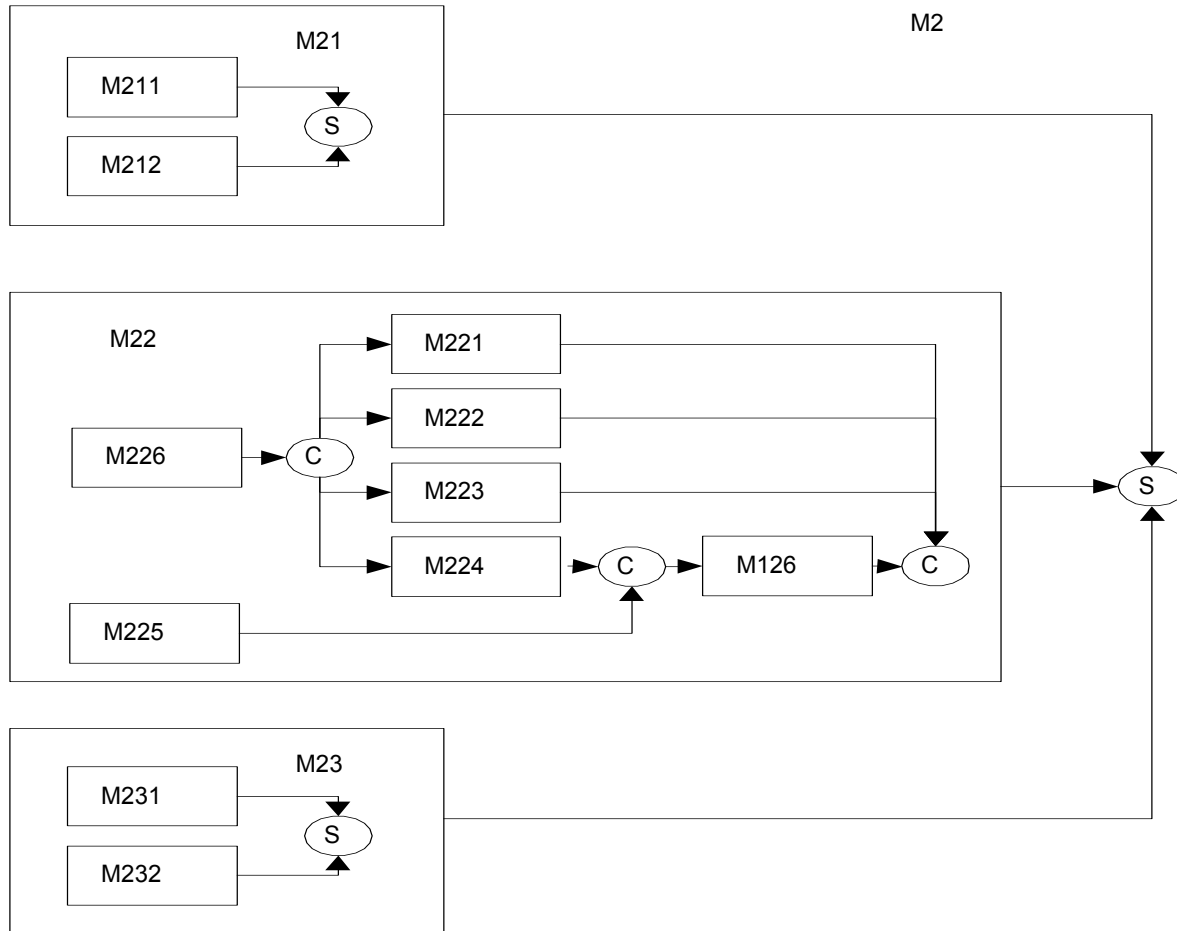**Figure b Flow Chart [FR1x/FR1xx]**

M2

M21

M211

M212

S

M22

M226

C

M221

M222

M223

M224

M225

C

M126

C

S

M23

M231

M232

S

Flow Chart [FR2x/FR2xx]

Mathematical Modeling, Simulation, and Optimization of Systems

Application of the Flow Diagram of the System Architecture

What is the system architecture good for?

Application of the Flow Diagram of the System Architecture

1. Diagnosis of system failure –

2. Engineering change orders –

3. Job assignment and management of a system development team –

4. Distributed systems –

5. System design through assembly of modules –

6. System consisting of hardware and software –