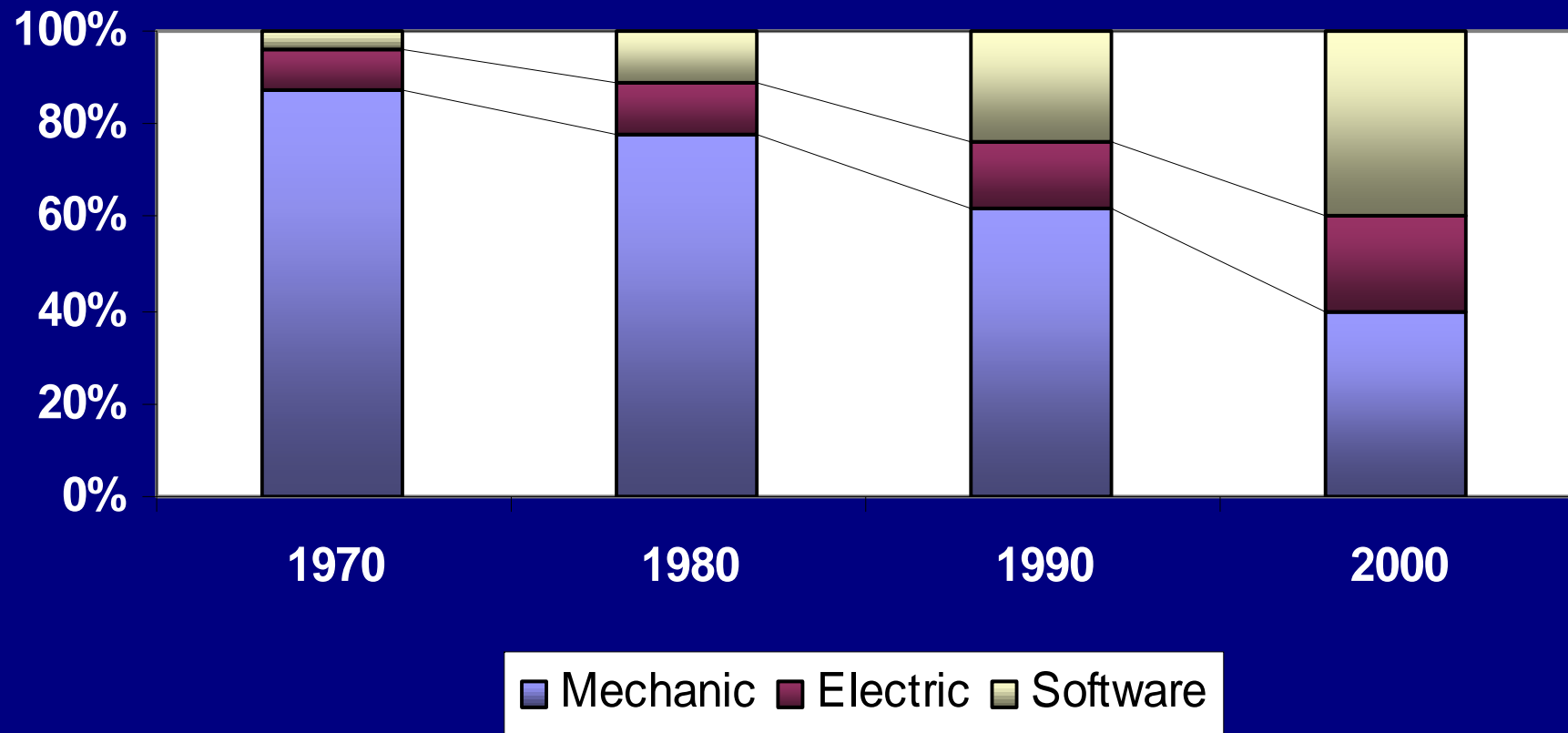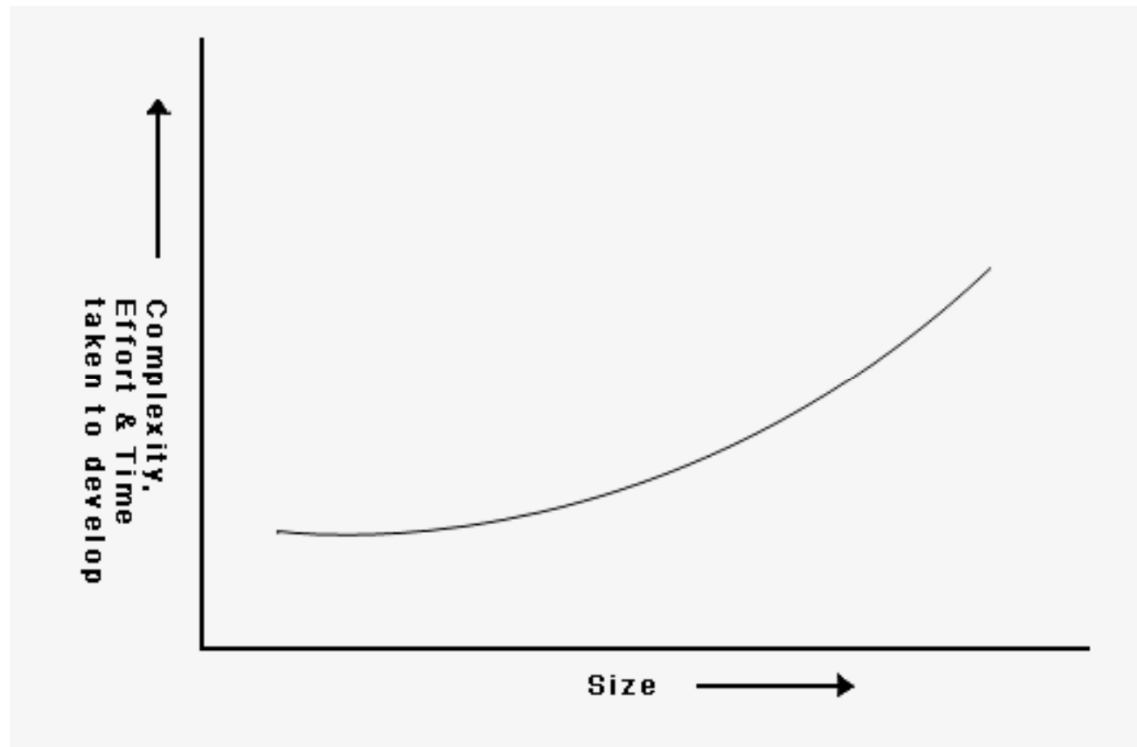# PLCopen
for efficiency in automation

Eelco van der Wal

Managing Director PLCopen

Percentage of Software development costs in production systems (source: McKinsey)

Managing Complexity

100 – 10,000 – 1mio – 100mio Lines of Code

Exponentially increasing complexity

Modern Software Development Process

With rigorous implementation

Software Program vs. Software Product

- **Made by an individual for own use**
- **Limited functionality**
- **User interface less important**
- **Little documentation**
- **Individual development style**

- **Made by a group for usage by others**
- **Larger functionality**
- **User interface very important**
- **Well documented**
- **Accepted SW engineering methods**

Exploratory Style vs.
Software Engineering Method

- **Based on error correction**

- **Finding errors during final product testing**

- **Coding is the goal, creating quickly a working system, and modifying till satisfactory**

- **Focused to error prevention**

- **Find errors as early as possible**

- **A structured approach, clear specifications, clear phases**

- **Periodic reviews during all stages of the project**

(Costly) Engineering Habits

- **Reuse work from similar former projects**

- **Include them into the new project**

- **And start adapting them to the new project requirements**

Copy - Paste & Modify

Dangers of copy paste & modify

- **The "not-invented-here syndrome" – only own artifacts (developed in the past)**

- **Non-predictable quality**

- **Prone to errors and reuse potential is wasted**

- **Unsystematically**

- **Difficult to maintain and manage**

- **Very costly over the life cycle**

Modern Software Development Process

With rigorous implementation

Why Structured Software Development ?

- **Software = key to system quality: errors cost money**

- **Increased requirements: 100 rungs of ladder now 10,000 rungs or even 100,000**

- **Not a one-man job - but a team with different know how and background**

- **Commissioning, Installation, Maintenance, and Improvements are essential phases**

- **Kaizen, on-going imrpovements**

"... the never ending story of software "

enhancements…..

**….. new requirements ...**

…. new functionalities ….

…. new wishes ...

Modern Software Development Process

- **Multiple clearly separated phases - project definition**

- **Top-down approach**

- **Multiple people involved**

- **With different backgrounds**

- **Multiple disciplines involved**

- **Based on Functional Requirements**
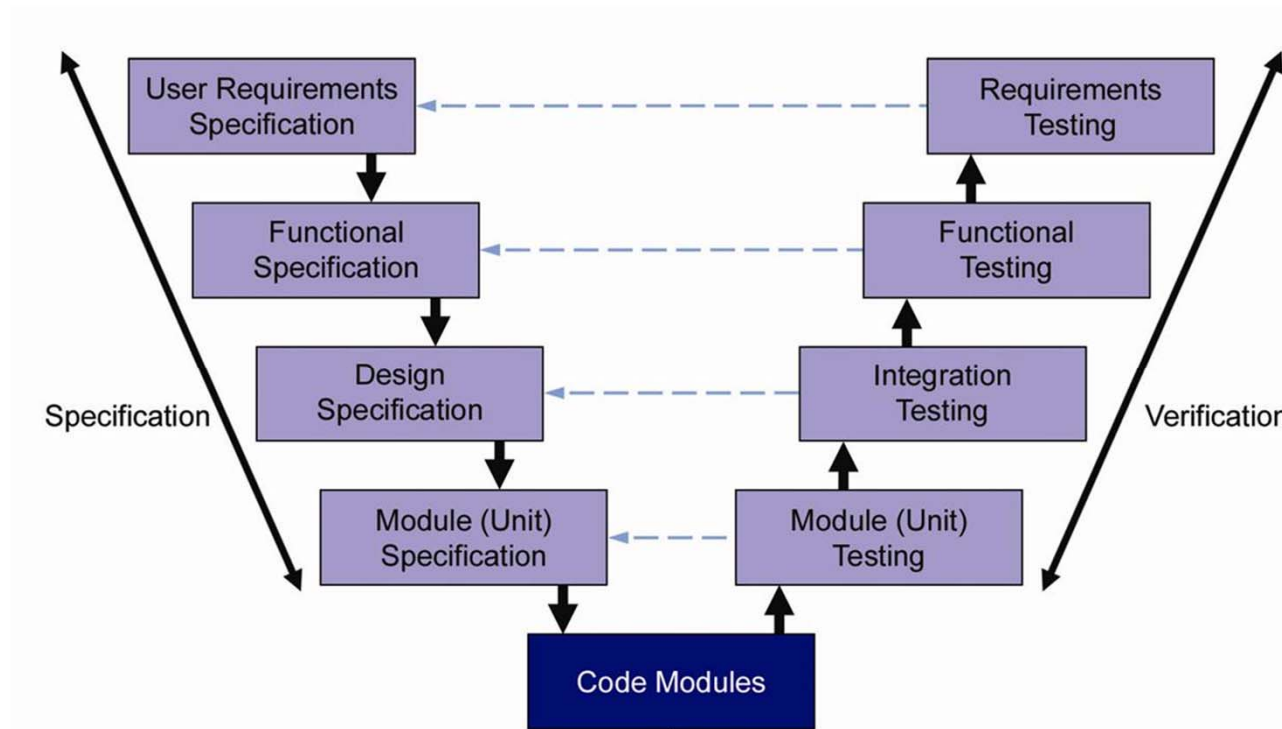
Example of Software Development Process

Different Phases

Analysis

Design
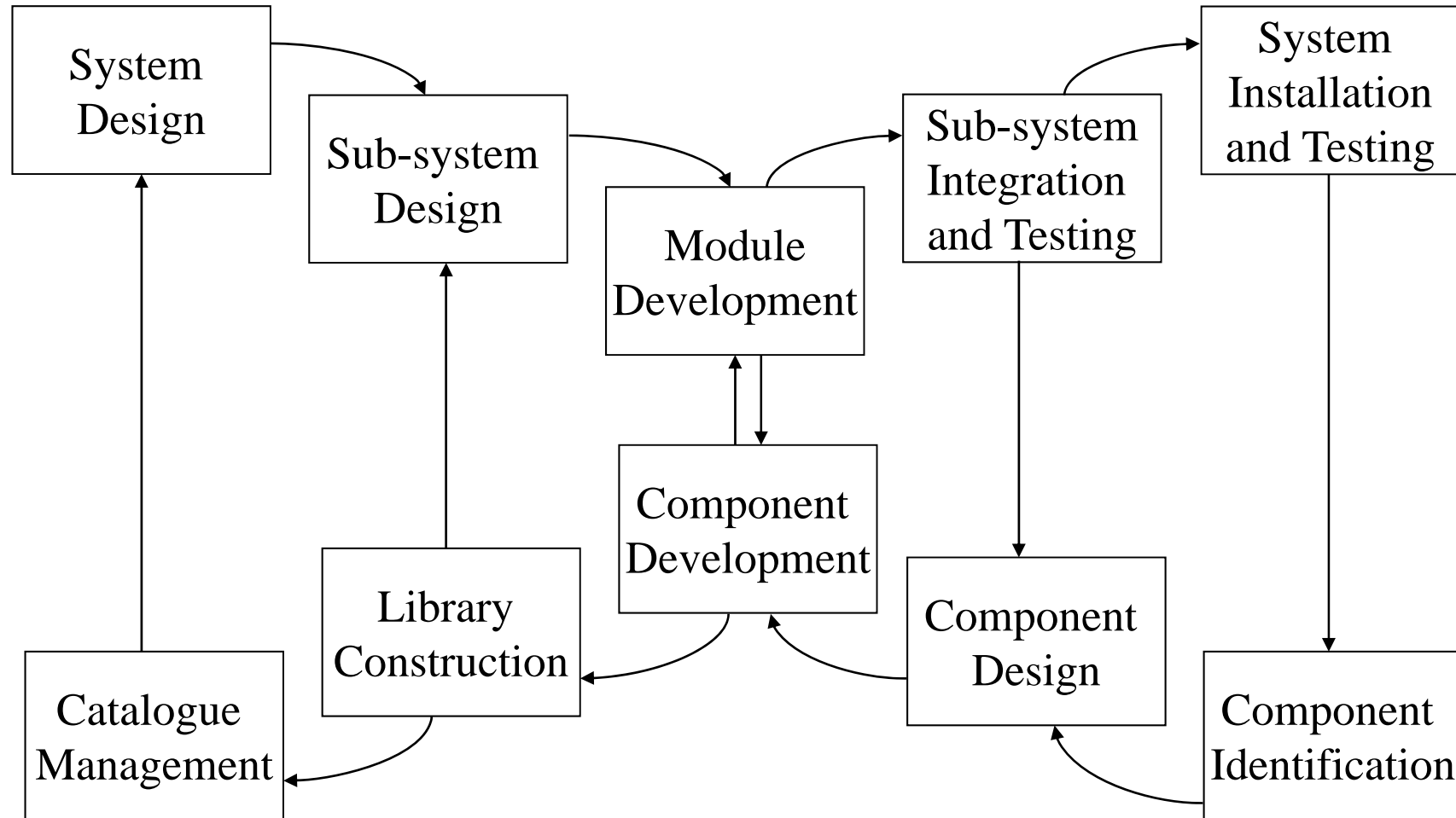
Development

Installation

Maintenance..
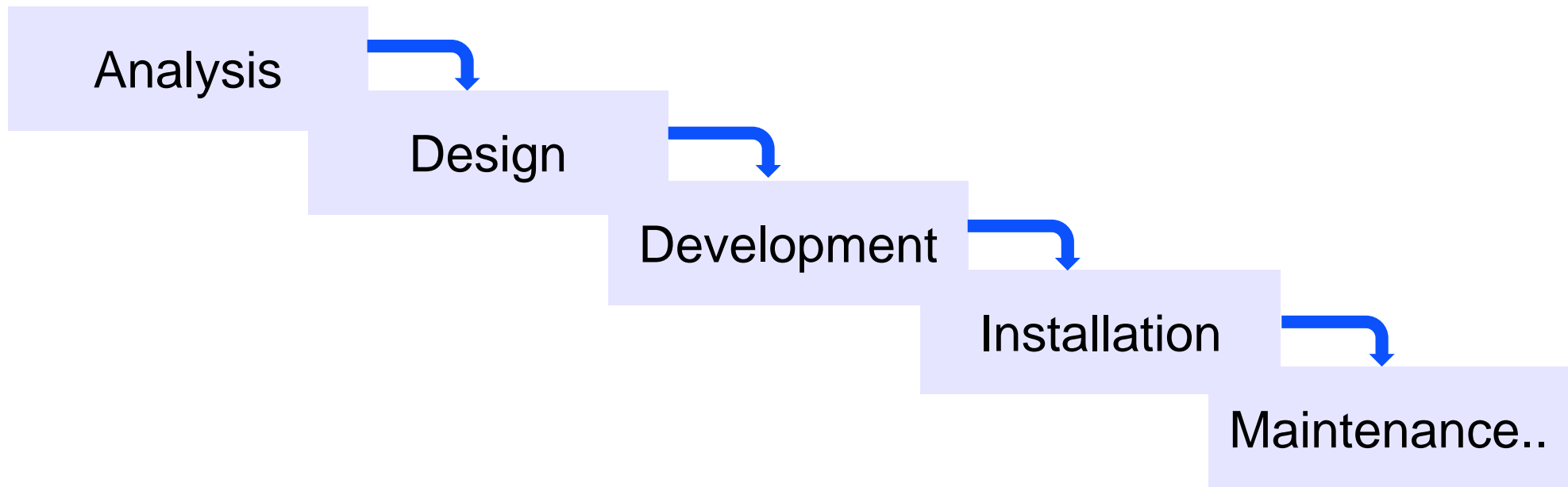
"Waterfall – model"

Example of Software Development Process

V-model

Link between

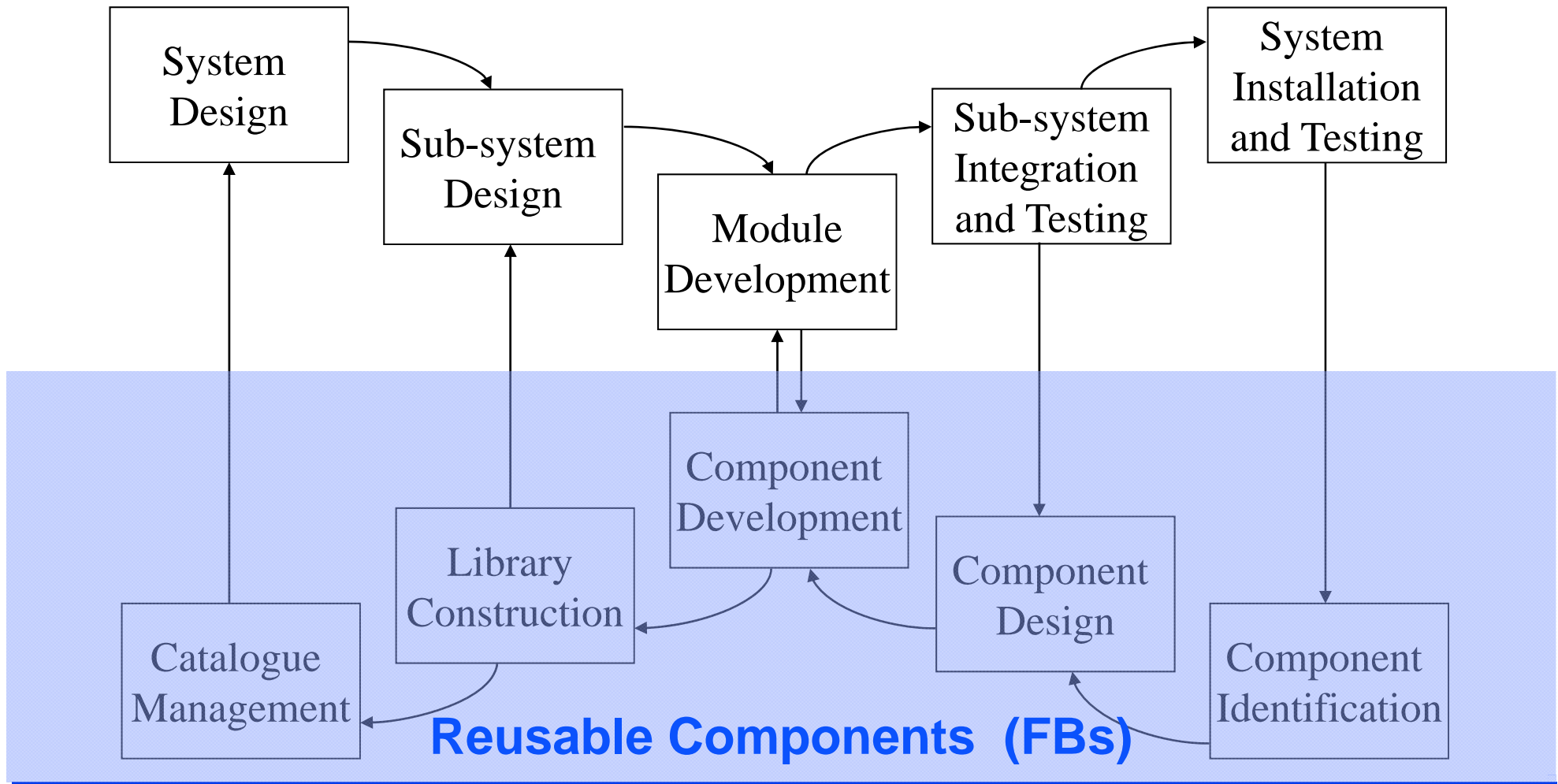Specification
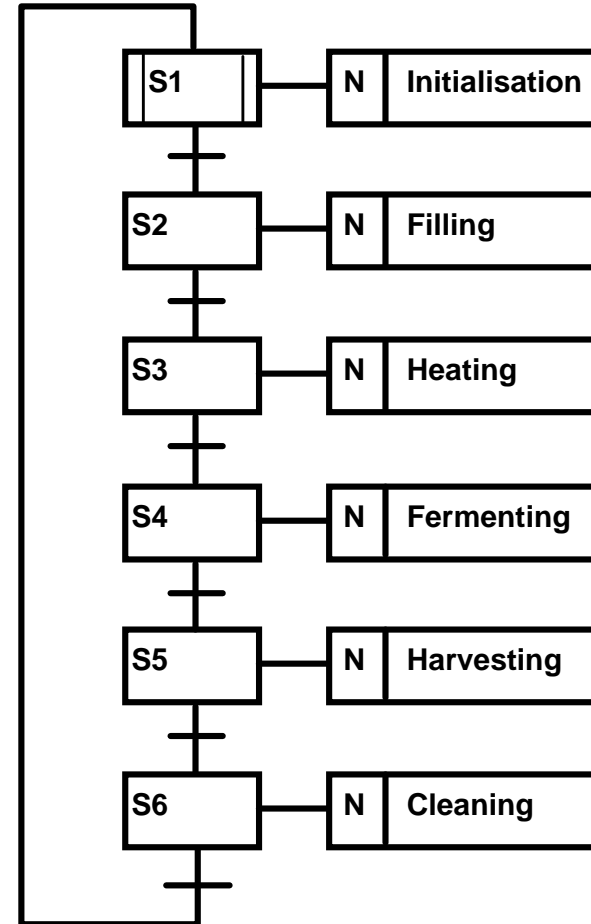
and testing

The X-Model for SW Development

System Design → Sub-system Design → Module Development → Sub-system Integration and Testing → System Installation and Testing

Catalogue Management, Library Construction, Component Development, Component Design, Component Identification

The X-Model for SW Development

Application Software

System Design

Sub-system Design

Module Development

Sub-system Integration and Testing

System Installation and Testing

Component Development

Library Construction

Component Design

Catalogue Management

Component Identification

The X-Model for SW Development

Decomposition with Sequential Function Chart, SFC

S1	N	Initialisation
S2	N	Filling
S3	N	Heating
S4	N	Fermenting
S5	N	Harvesting
S6	N	Cleaning

A hierarchy of encapsulation
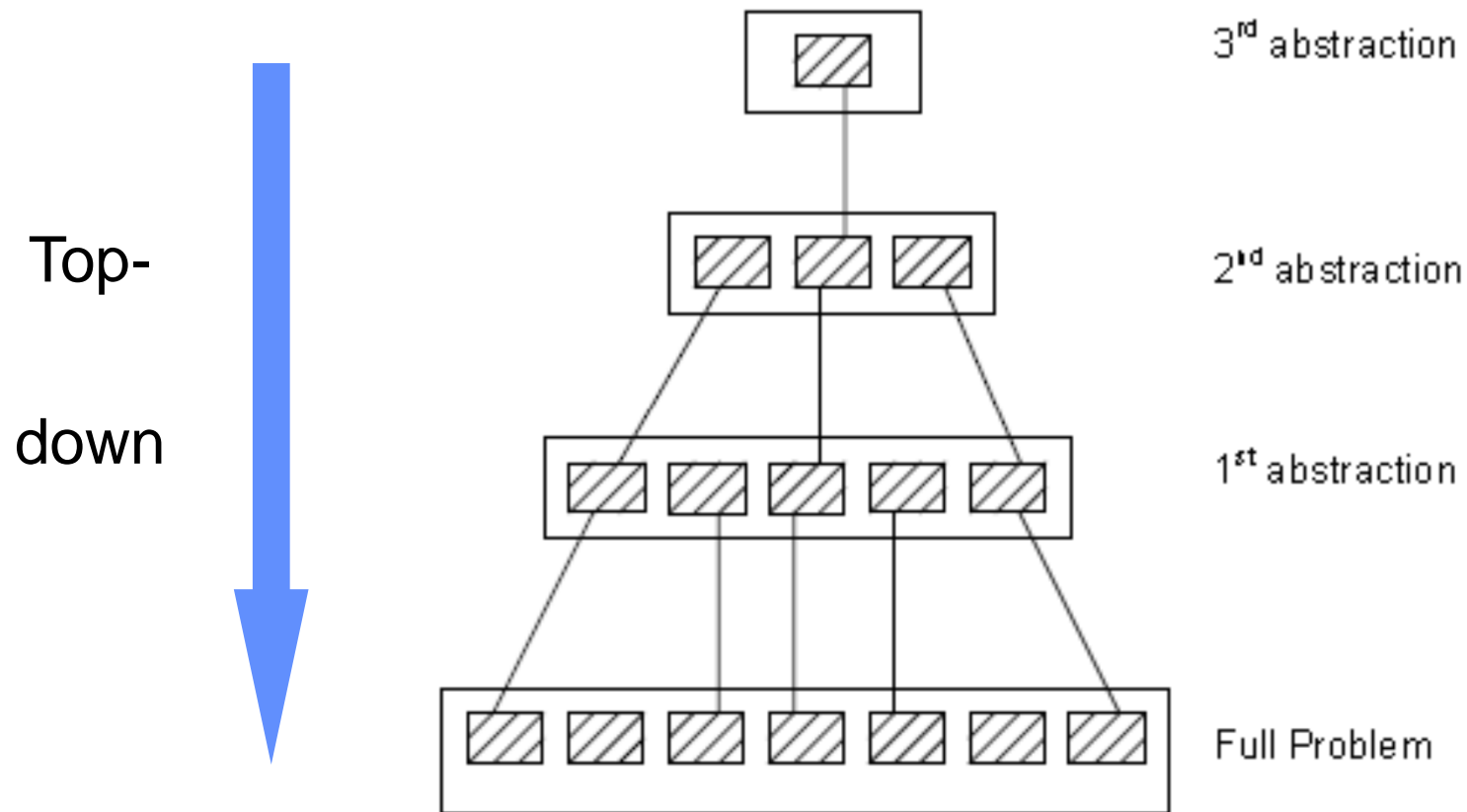
Decomposition and Reuse

Automation application

PROGRAM

Type	Local

GLOBAL

FUNCTION

Type	Local

INPUT

INPUT OUTPUT IN_OUT EXTERNAL

FUNCTION_BLOCK

Type	Local
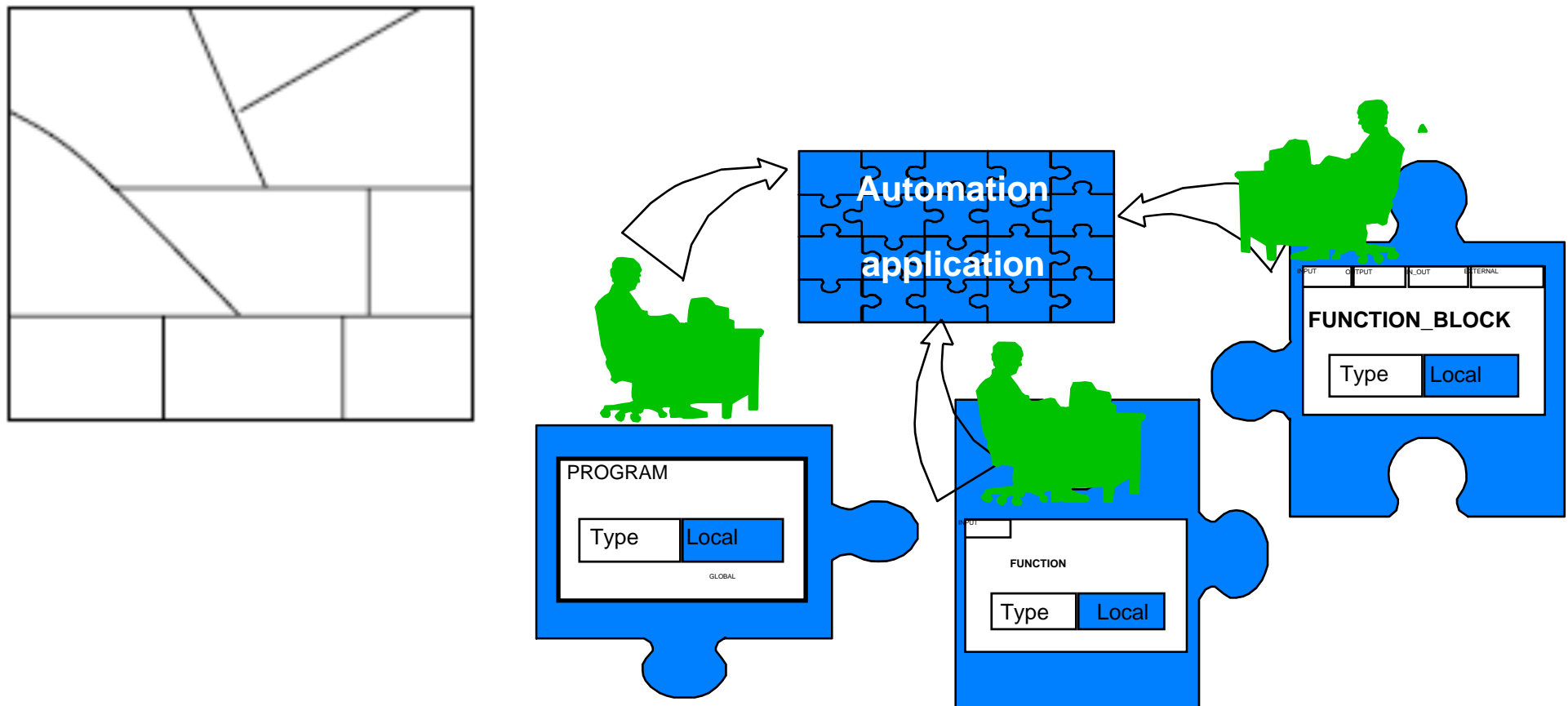
Bottom-up after top-down

First decompose – then fill it in

Top-down

Bottom up
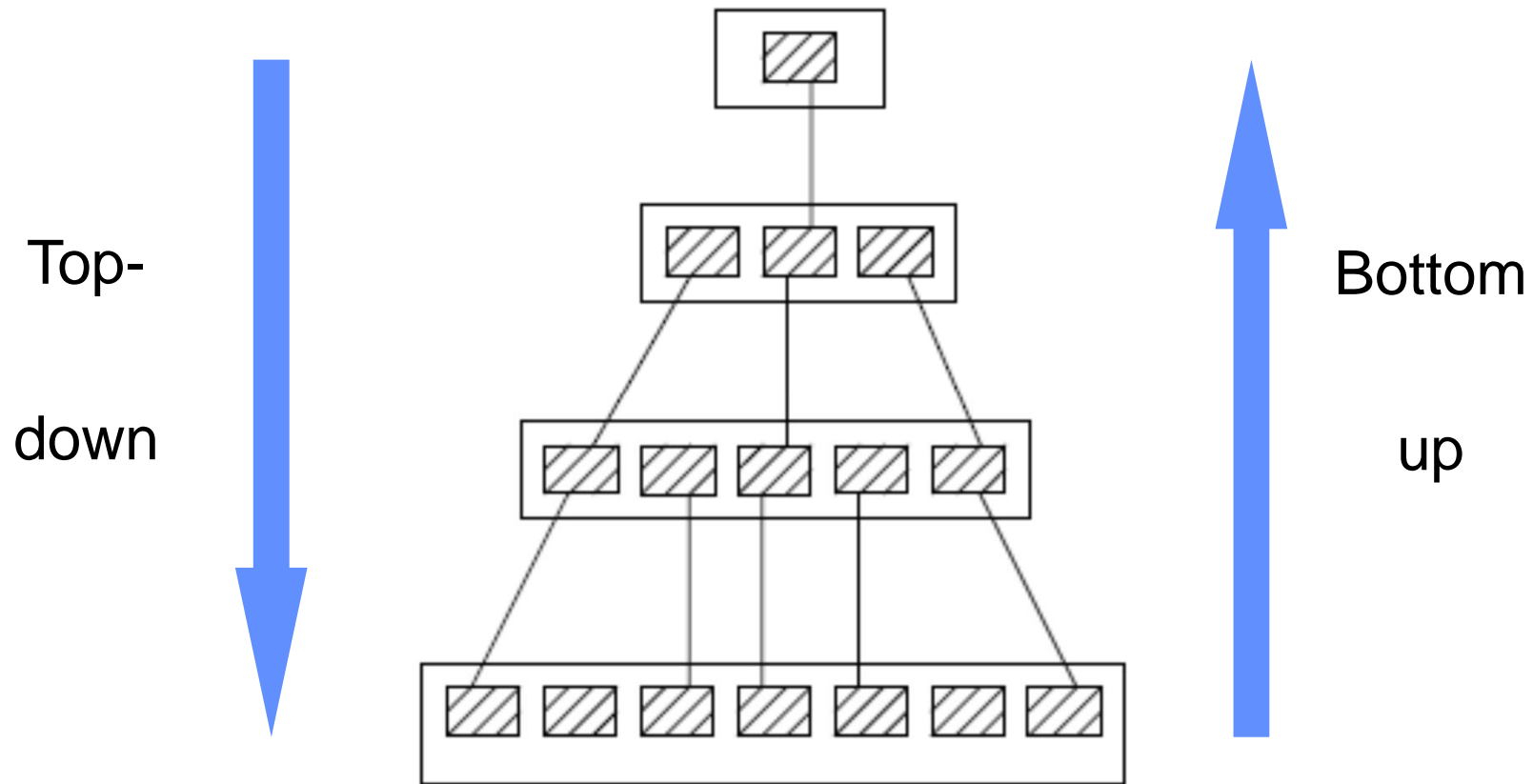
Abstraction via Function Blocks

	MC_MoveAbsolute	
AXIS_REF	Axis — — — — — — — — — — Axis	AXIS_REF
BOOL	Execute	Done — BOOL
BOOL	ContinuousUpdate	Busy — BOOL
REAL	Position	Active — BOOL
REAL	Velocity	CommandAborted — BOOL
REAL	Acceleration	Error — BOOL
REAL	Deceleration	ErrorID — WORD
REAL	Jerk	
MC_DIRECTION	Direction	
MC_BUFFER_MODE	BufferMode	
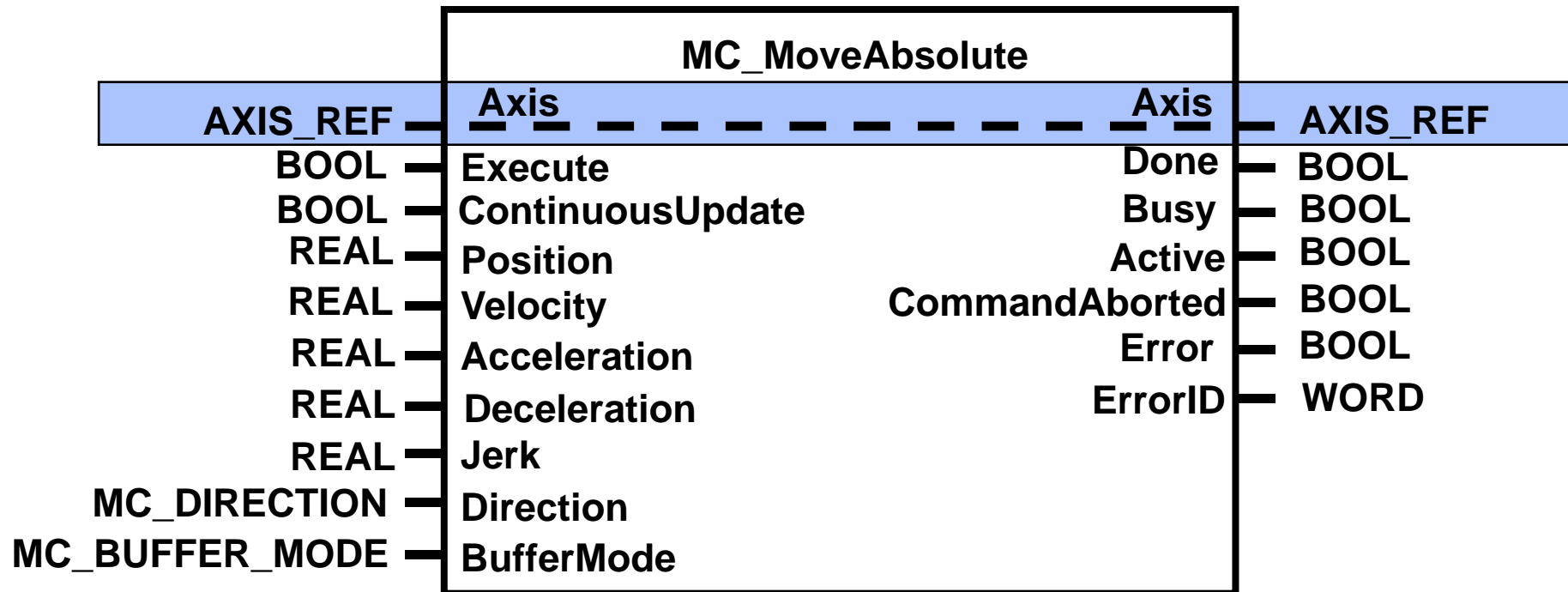
Abstraction via Axis_Ref

Axis1

FB

Time or event driven

Task Manager

Axis_Ref Structure

Conver-sion

I/F

Drive

Axis_Ref with 2 FBs

Axis1

FB1 FB2

Task Manager

Time or event driven

Axis_Ref Structure

Conver-sion

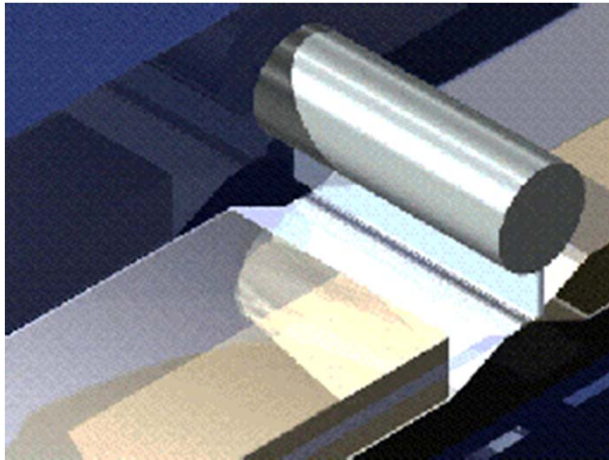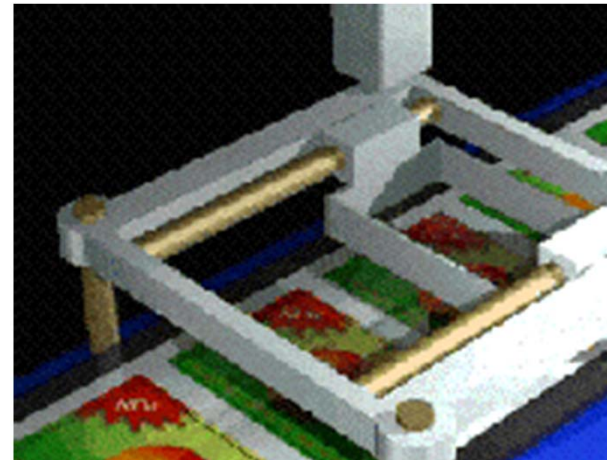I/F Drive
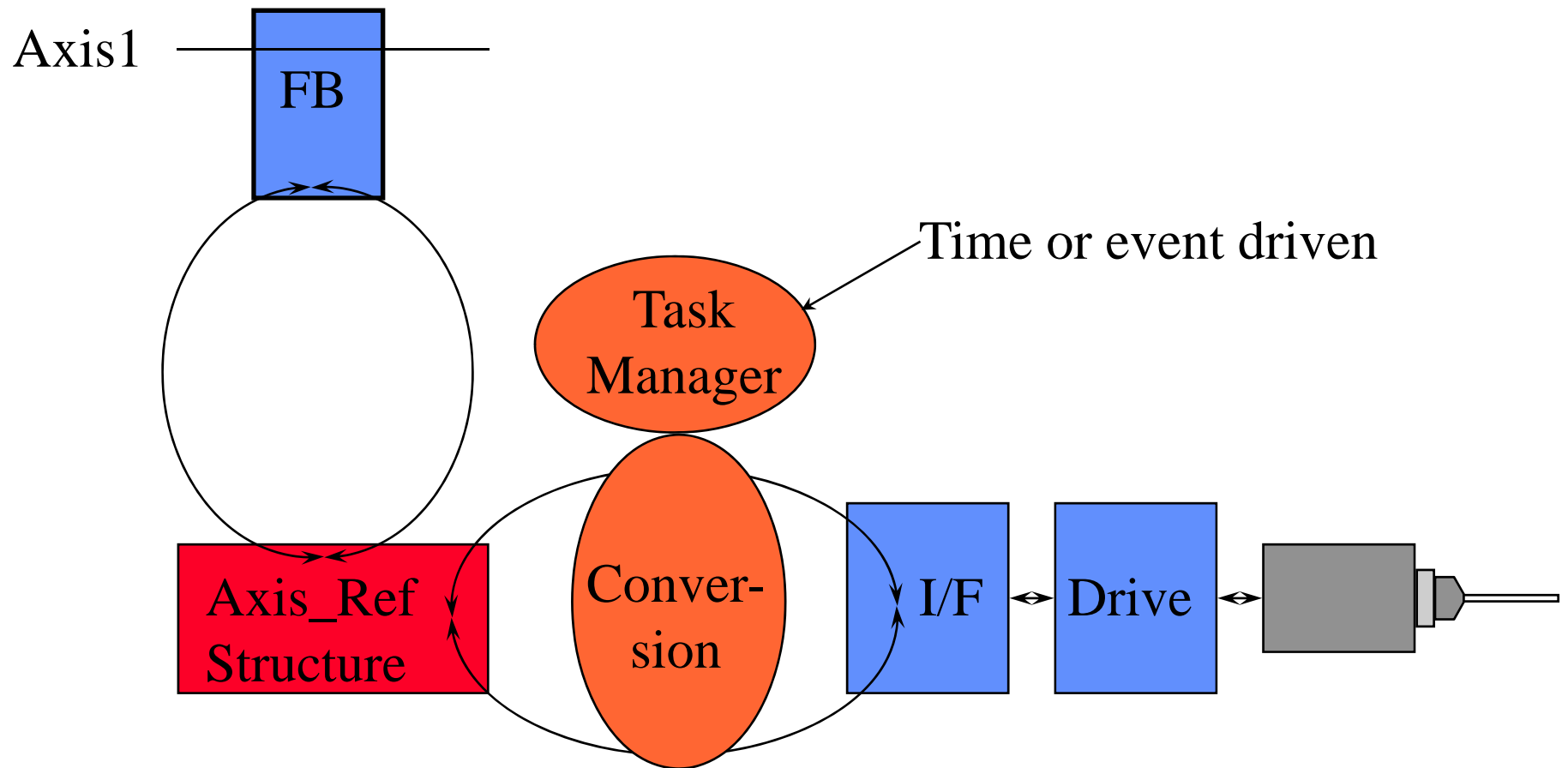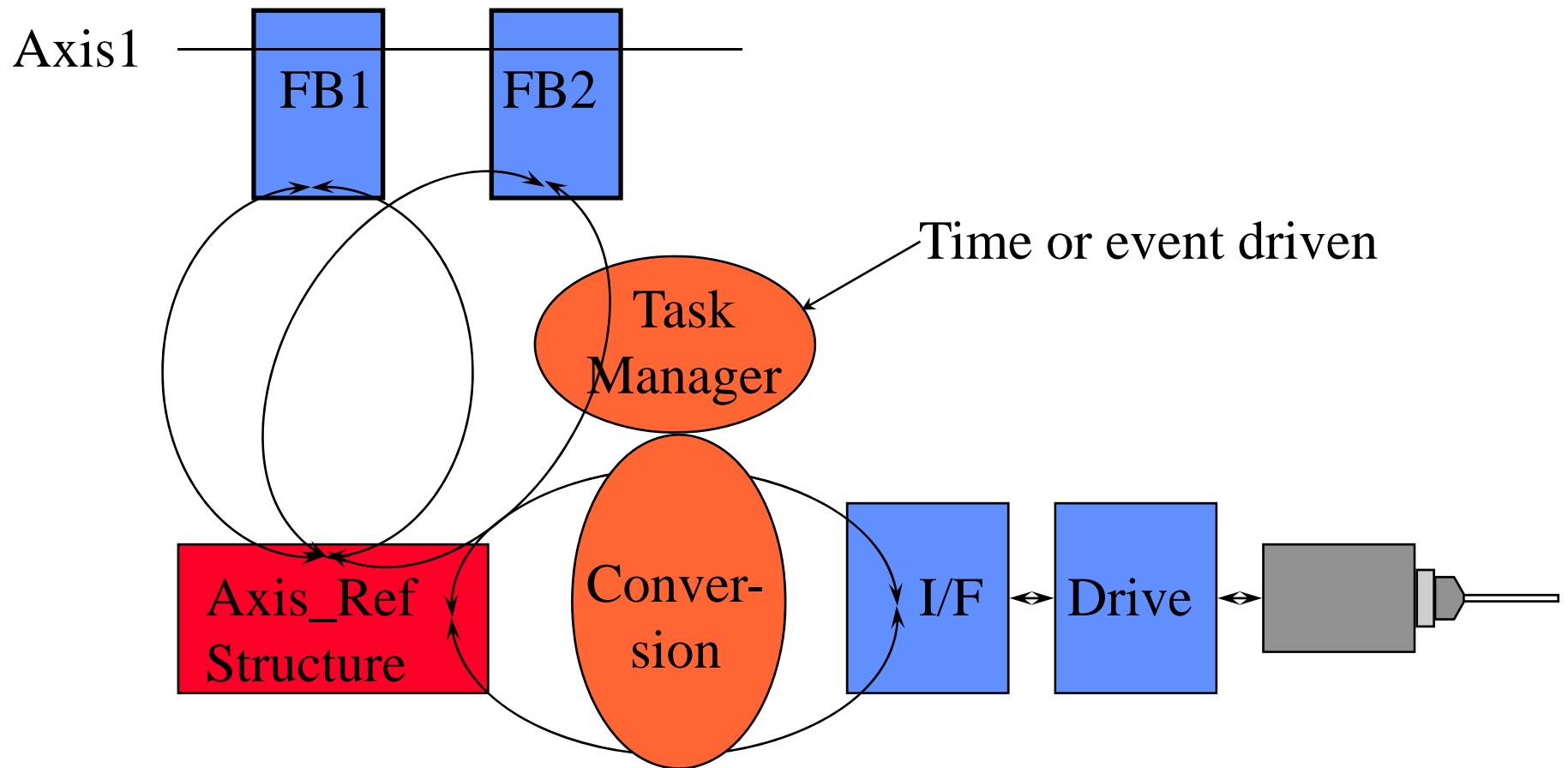
Mechatronic solutions
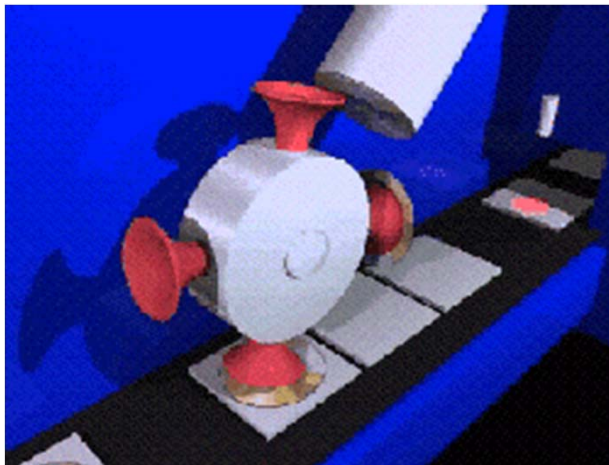
Mechanical solution.
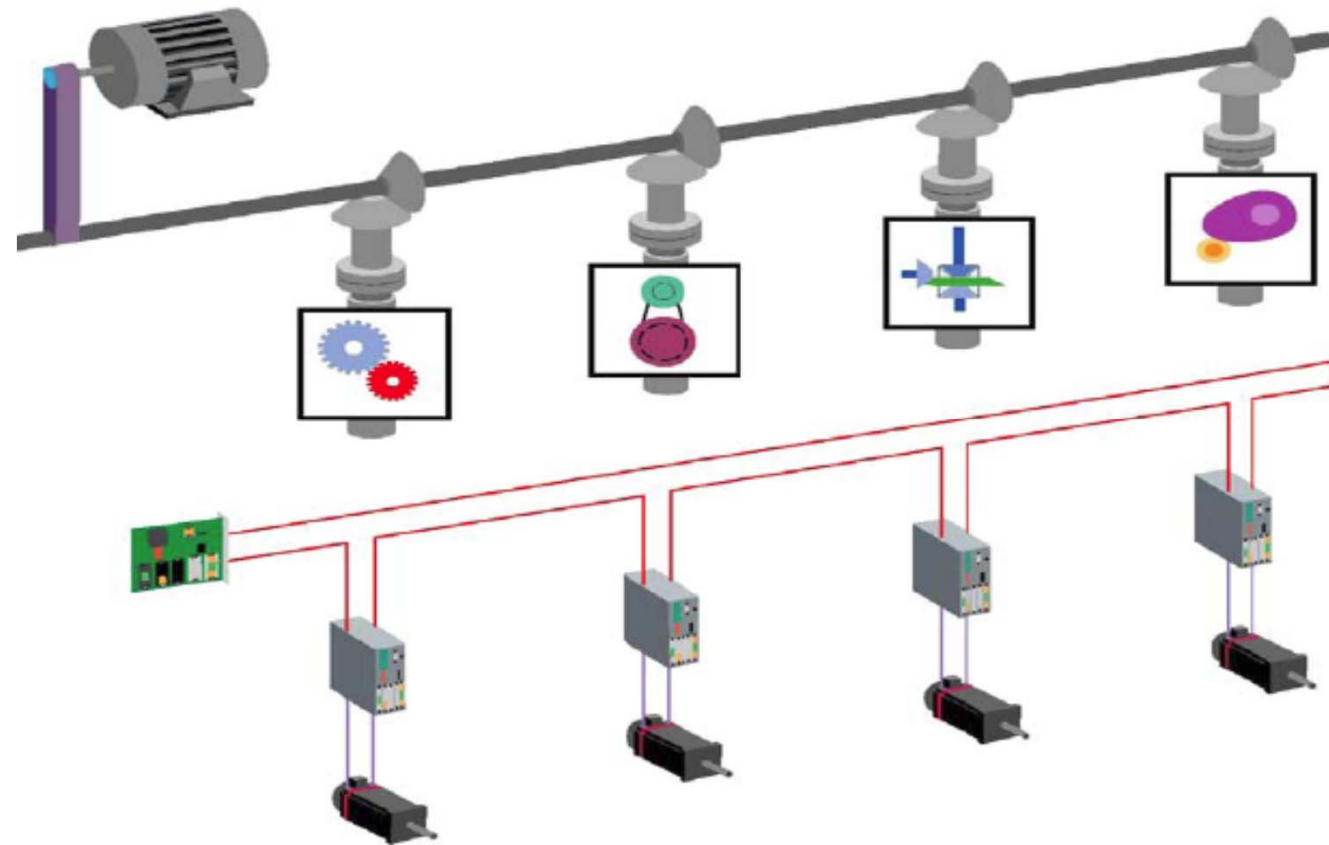
Control solution

Encapsulation

Cut-to-length
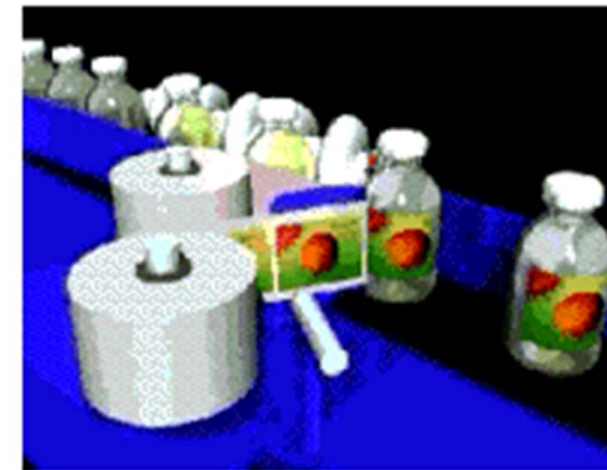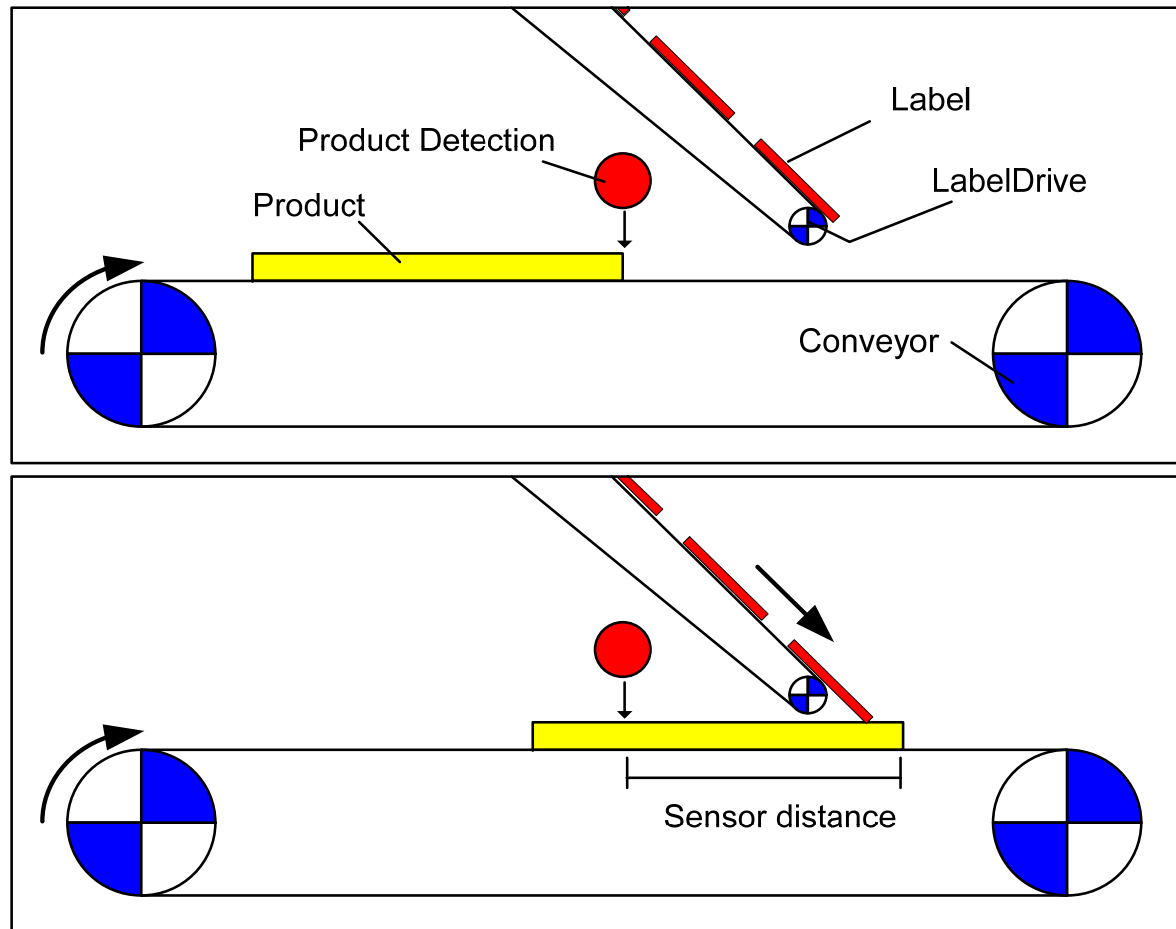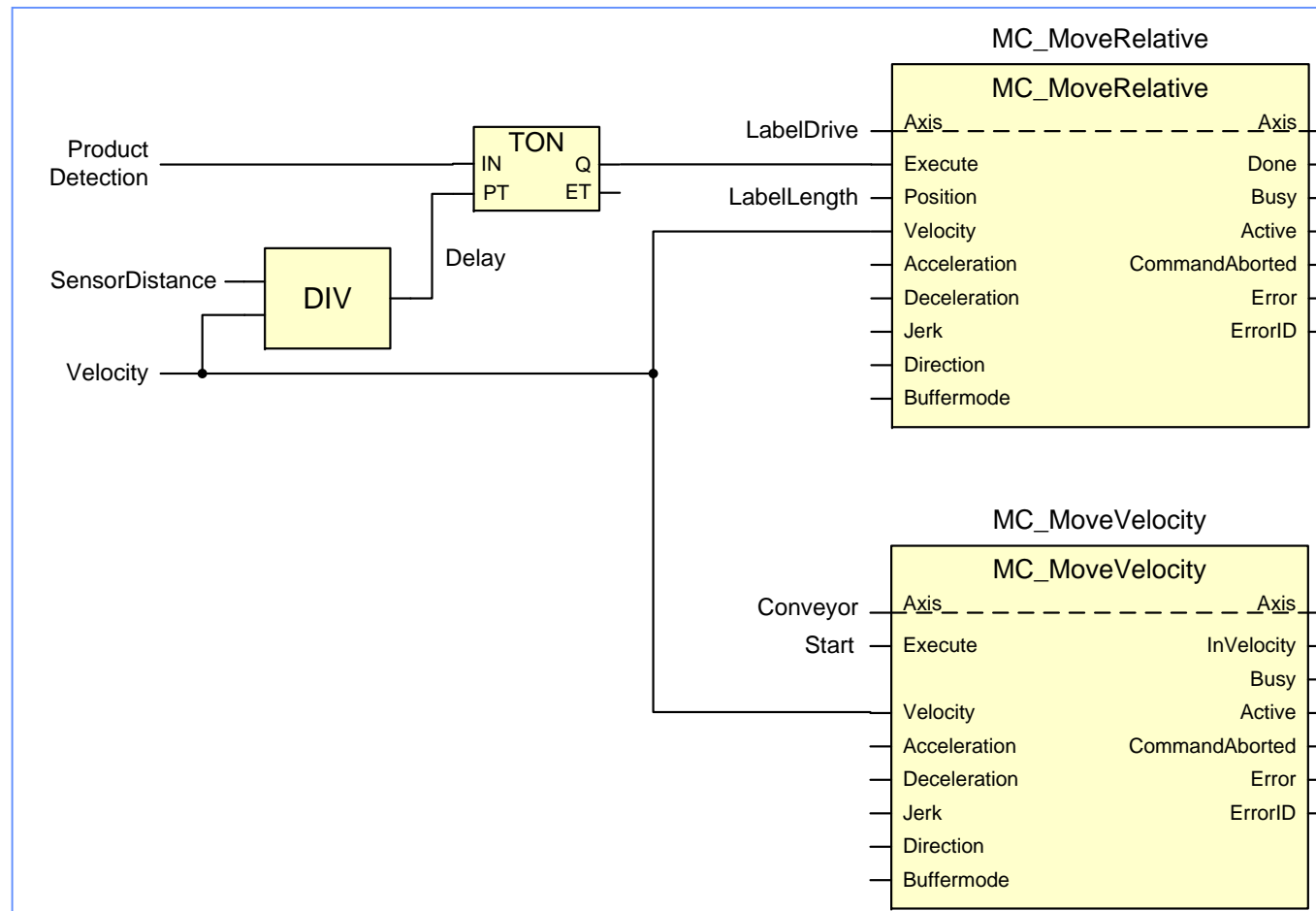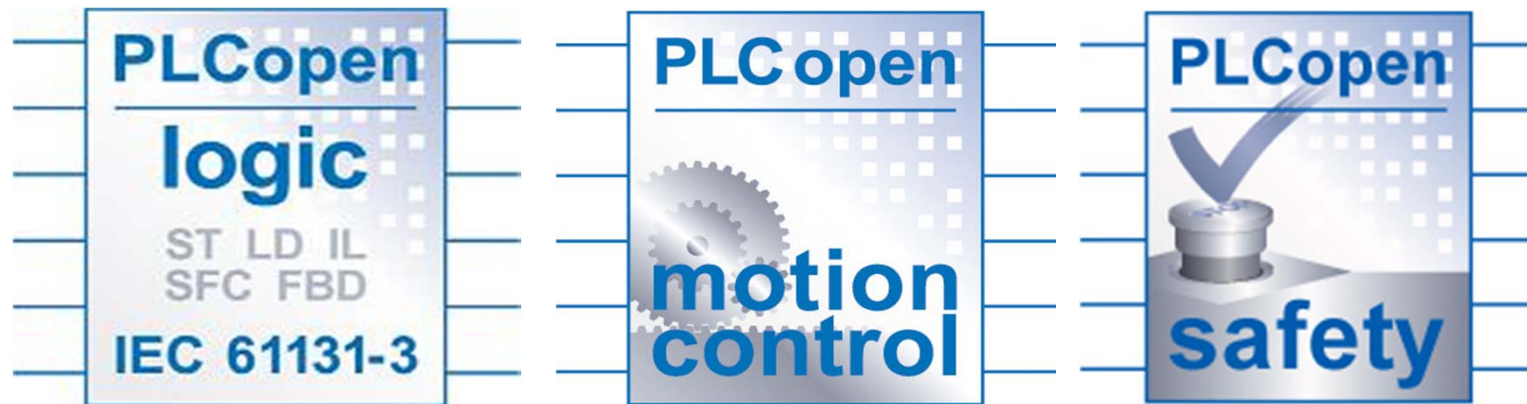
Cutting

Positioning

Labelling

Labelling

Label

Product Detection

Product

LabelDrive

Conveyor

Sensor distance

The labelling program

PLCopen
for efficiency in automation

PLCopen

Combining Logic, Motion and Safety

Providing Structuring, Decomposition, Reuse and less training

Overview PLCopen

PLCopen
for efficiency in automation

More Information...
and to download the specifications (f.o.c)

www.PLCopen.org

www.PLCopen-Japan.jp

Free-of-Charge electronic Newsletter 'PLCopening' (in English)

email: evdwal@plcopen.org

Thanks !

End of this presentation