



DESIGN AND OPERATION OF

Robotic Production Systems

From Concept to Commissioning

Bogdan Mocan · Mircea Fulea · Mircea Murar

Technical University of Cluj-Napoca

U.T.PRESS
Cluj-Napoca, 2026
ISBN 978-606-737-828-3

DESIGN AND OPERATION OF

Robotic Production Systems

From Concept to Commissioning

Bogdan MOCAN

Prof., PhD (eng.), Habil.

Lead Author and Coordinator

with co-authors

Mircea FULEA

Assoc. Prof., PhD (eng.), Habil.

Mircea MURAR

Senior Lecturer, PhD (eng.)

Department of Engineering Design and Robotics
Technical University of Cluj-Napoca



U.T.PRESS

Cluj - Napoca, 2026

ISBN 978-606-737-850-4



Editura U.T.PRESS
Str. Observatorului nr. 34
400775 Cluj-Napoca
Tel.: 0264-401.999
e-mail: utpress@biblio.utcluj.ro
www.utcluj.ro/editura

Recenzia: Prof.dr.ing. Neamțu Călin
Prof.dr.ing. Dragomir Mihai

Pregătire format electronic on-line: Gabriela Groza

Copyright © 2026 Editura U.T.PRESS
Reproducerea integrală sau parțială a textului sau ilustrațiilor din această carte este posibilă numai cu acordul prealabil scris al editurii U.T.PRESS.

ISBN 978-606-737-850-4

*To my students —
past, present, and future —
who will design the factories of tomorrow.*

Preface

As coordinator of this textbook, I have shaped its structure, pedagogical approach, and overall editorial direction, drawing on more than a decade of teaching the CESPR course at the Technical University of Cluj-Napoca. The work has benefited significantly from the contributions of my co-authors, Assoc. Prof. Mircea Fulea and Senior Lecturer Mircea Murar, whose expertise has enriched specific chapters and made this a genuinely collaborative volume.

This book grew out of a simple observation: the most common cause of underperformance in robotic production systems is not the robot. It is the system around the robot — the gripper that drops parts, the conveyor that jams, the safety fence placed too close for maintenance, the PLC handshake that hangs without a timeout, the layout that forces operators to walk sixteen metres to clear a jam. The robot itself, in almost every case I have encountered in twenty years of academic and industrial practice, works exactly as the manufacturer specified. The system into which it is integrated does not.

This observation is the thesis of the book: **robots generate value through their integration into well-designed production systems**. The robot is one component; the system is the deliverable. Every chapter, every worked example, every industrial practice note in these pages returns, directly or indirectly, to this central idea.

The book is written for fourth-year engineering students in the Robotics specialisation at the Technical University of Cluj-Napoca, but its audience extends to any engineer — mechanical, electrical, control, or industrial — who designs, integrates, operates, or maintains robotic production systems. The prerequisite is a foundational understanding of industrial robotics: what a robot is, how it moves, what an end-effector does. Everything beyond that is developed from first principles within these pages.

Structure

The fourteen chapters are organised in four Parts that mirror the lifecycle of a robotic production system. **Part I (Chapters 1–3)** establishes the vocabulary: the role of robots in modern manufacturing, the components and flows of a production system, and the strategic importance of layout design. **Part II (Chapters 4–7)** is the engineering core: a structured six-step design methodology, robot and EOAT selection with multi-criteria methods, control architecture and PLC integration, and simulation-based validation. **Part III (Chapters 8–10)** addresses the operational phase: safety engineering per ISO 12100 and ISO/TS 15066, performance monitoring through OEE and SCADA, and maintenance strategies from reactive through predictive. **Part IV (Chapters 11–14)** looks forward: collaborative robots and AI, adaptive and mobile systems, industrial case studies that ground the theory in practice, and the commissioning process that closes the loop between virtual design and physical reality.

Five appendices provide reference material: a mathematical toolkit, a standards compendium, a glossary and acronym reference, data tables and formulas, and project templates and checklists.

Pedagogical Approach

Each chapter follows a consistent structure: **Learning Objectives** that state what the student will be able to do after studying the chapter; theoretical foundations presented through clear prose and numbered equations; **Worked Examples** with step-by-step numerical solutions that can be reproduced with a calculator; **Industrial Practice Notes** drawn from real installations that show how textbook principles play out on the factory floor; **Common Pitfalls** that warn against the mistakes most frequently made by both students and practising engineers; **Key Takeaways** that distil the chapter to its essential messages; **Review Questions** that test both understanding and application; and **References** that point to the primary literature for further study.

The worked examples are deliberately practical. They use real robot models, real gripper specifications, real cost figures, and real performance data. Where simplification is necessary for clarity, the simplification is stated explicitly so that the student knows what

a production-grade calculation would add. I have tried, throughout, to write the book I wished I had when I was learning this discipline — one that respects the reader's intelligence while being honest about the complexity of real engineering.

Acknowledgements

This book would not exist without the students of the CESPR course at the Technical University of Cluj-Napoca, whose questions, projects, and feedback over the years have shaped every chapter. I am grateful to my colleagues in the Department of Engineering Design and Robotics for their support and intellectual generosity, and to the industrial partners — particularly in the automotive and food-processing sectors of the Cluj region — who opened their factories to our students and shared the realities of robotic production that no textbook can fully capture.

As coordinator of this volume, I am particularly grateful to my co-authors — Assoc. Prof. Mircea Fulea and Senior Lecturer Mircea Murar — for their expertise, dedication, and collegiality in contributing to specific chapters. Their work has made this textbook a genuinely collaborative effort, and my gratitude for their partnership runs deeper than any formal acknowledgement can convey.

Finally, I thank my family for their patience during the many evenings and weekends that this manuscript consumed. The debt is real and acknowledged.

Bogdan Mocan

Cluj-Napoca, July 2026

Contents

Preface	iv
Acknowledgements	vii
List of Abbreviations	ix
About the Lead Author.....	xii
PART I — Concepts and Fundamentals	1
Chapter 1 — The Role of Robots in Modern Manufacturing Systems.....	1
Chapter 2 — Components, Flows, and System Architecture	24
Chapter 3 — Process Layout Design for Robotic Production Systems.....	49
PART II — Design and Integration	71
Chapter 4 — Methodology for Designing Robotic Cells.....	71
Chapter 5 — Robot Selection and End-of-Arm Tooling (EoAT)	93
Chapter 6 — Communication, Control Architecture, and PLC Integration.....	118
Chapter 7 — Simulation and Digital Validation of Robotic Systems	141
PART III — Operation, Safety, and Performance.....	161
Chapter 8 — Safety Engineering in Robotic Production Systems	161
Chapter 9 — Operation, Monitoring, and Performance Metrics.....	193
Chapter 10 — Maintenance, Reliability, and Lifecycle Engineering	231
PART IV — Advanced Topics and Industry Trends	260
Chapter 11 — Collaborative Robots and AI-Enhanced Production	260
Chapter 12 — Adaptive, Reconfigurable, and Mobile Robotic Systems.....	286
Chapter 13 — Industrial Case Studies and Best Practices	314
Chapter 14 — Commissioning and Course Synthesis	341
APPENDICES.....	364
Appendix A — Mathematical Foundations	365
Appendix B — Standards Compendium.....	376
Appendix C — Glossary and Acronyms	384
Appendix D — Reference Data Tables.....	394
Appendix E — Project Templates and Checklists	402

List of Abbreviations

Abbreviation	Meaning
AGV	Automated Guided Vehicle
AHP	Analytic Hierarchy Process
AI	Artificial Intelligence
AMR	Autonomous Mobile Robot
AOI	Automated Optical Inspection
ATC	Automatic Tool Changer
BIW	Body-in-White
CAD	Computer-Aided Design
CBM	Condition-Based Maintenance
CIM	Computer-Integrated Manufacturing
CIP	Cleaning-in-Place
CNN	Convolutional Neural Network
CRAFT	Computerized Relative Allocation of Facilities Technique
CT	Cycle Time
DMS	Dedicated Manufacturing System
DOF	Degrees of Freedom
EOAT	End-of-Arm Tooling
ERP	Enterprise Resource Planning
FAT	Factory Acceptance Test
FBD	Function Block Diagram
FMS	Flexible Manufacturing System
GA	Genetic Algorithm
GAMP	Good Automated Manufacturing Practice
GDS	Global Discovery Server
HIL	Hardware-in-the-Loop
HMI	Human-Machine Interface
HRI	Human-Robot Interaction
IEC	International Electrotechnical Commission

Abbreviation	Meaning
IFR	International Federation of Robotics
IIoT	Industrial Internet of Things
IO-Link	Point-to-point industrial communication interface
IR	Industrial Robot
IRR	Internal Rate of Return
ISA	International Society of Automation
ISO	International Organization for Standardization
KPI	Key Performance Indicator
LD	Ladder Diagram
LIDAR	Light Detection and Ranging
LLM	Large Language Model
MES	Manufacturing Execution System
MHC	Material-handling cost coefficient
MIL	Model-in-the-Loop
MQTT	MQTT publish/subscribe messaging protocol
MRP	Media Redundancy Protocol (PROFINET)
MTBF	Mean Time Between Failures
MTTR	Mean Time to Repair
NPV	Net Present Value
OEE	Overall Equipment Effectiveness
OLP	Offline Programming
OPC UA	Open Platform Communications Unified Architecture
PFL	Power and Force Limiting
PL	Performance Level (ISO 13849-1)
PLC	Programmable Logic Controller
PPM	Parts per Million
QoS	Quality of Service
RaaS	Robotics as a Service
RL	Reinforcement Learning
RMS	Reconfigurable Manufacturing System
ROI	Return on Investment
RTM	Requirements Traceability Matrix
SAT	Site Acceptance Test
SCADA	Supervisory Control and Data Acquisition

Abbreviation	Meaning
SCARA	Selective Compliance Articulated Robot Arm
SFC	Sequential Function Chart
SIL	Safety Integrity Level
SiL	Software-in-the-Loop
SLAM	Simultaneous Localisation and Mapping
SLP	Systematic Layout Planning
SMED	Single-Minute Exchange of Die
SMS	Safety-Rated Monitored Stop
SSM	Speed and Separation Monitoring
ST	Structured Text
TCO	Total Cost of Ownership
TCP	Tool Center Point
TPM	Total Productive Maintenance
TSN	Time-Sensitive Networking
TT	Takt Time
URLLC	Ultra-Reliable Low-Latency Communication
VA	Value-Added
VC	Virtual Commissioning
WIP	Work in Progress
WMS	Warehouse Management System

About the Lead Author

Bogdan Mocan is Professor of Industrial Robotics and Automation at the Technical University of Cluj-Napoca (UTCN), Romania, where he holds the *habilitation* in Mechanical Engineering. He leads the Interaction Design Research Group within the Department of Engineering Design and Robotics.

His research spans industrial robotic automation, robot-assisted surgery, and human-robot interaction, with a particular focus on the design and optimisation of robotic production systems for welding, handling, packaging, and palletising applications. He has published over 80 papers in international journals and conferences, and holds multiple patents in the field of robotic system design.

Prof. Mocan is the author of *Sisteme Robotizate de Sudare cu Arc Electric* (UTPress, ISBN 978-973-662-881-8) and co-author of *Multidisciplinary Design of Industrial Robotic Automation Solutions — Practical Guide for Students* (UTPress, ISBN 978-606-737-246-5). He has supervised over 100 bachelor's and master's theses in robotics and has been instrumental in developing the CESPR course that forms the basis of this textbook, for which he serves as coordinator and lead author.

He has served as visiting researcher at universities in Germany, Austria, and the United Kingdom, and maintains active collaborations with industrial partners in automotive, food processing, and medical robotics. He is a member of the European Robotics Association (euRobotics) and the Romanian Society for Robotics.

Contact: bogdan.mocan@campus.utcluj.ro

Web: <https://sites.google.com/view/clujrobotics>

IXD Group: <https://ixd.utcluj.ro>

CHAPTER

1

The Role of Robots in Modern Manufacturing Systems

From isolated machines to integrated, intelligent production

This opening chapter establishes the conceptual framework of the CESPR course and of the textbook that supports it by positioning industrial robots within manufacturing systems, rather than treating them as standalone automated devices. The discussion begins with a focused historical overview of industrial robotics, highlighting the transition from rigid, isolated automation to integrated, data-driven, and flexible production systems. Special emphasis is placed on the distinction between robotic cells and robotic production lines, as well as on the evolving role of humans in increasingly automated environments.

Industry 4.0 and Industry 5.0 paradigms are introduced to explain how connectivity, digitalisation, sustainability, resilience, and human-centric design are reshaping manufacturing. The chapter concludes by introducing the central idea that underpins both the course and this textbook: **robots generate value only through their integration into well-designed production systems.**

LEARNING OBJECTIVES

1. Trace the four-era evolution of industrial robotics used in this course, from early manipulators to connected and adaptive systems.
2. Distinguish robotic cells from production lines by architecture, flexibility, throughput, scalability, and investment.
3. Explain the four Industry 4.0 principles and apply a transparent readiness rubric.
4. Explain the Industry 5.0 pillars—human-centricity, sustainability, and resilience—and their design implications.
5. Interpret IFR installation and density data, including the limits of national averages and short-term growth rates.
6. Apply the six-dimensional Value Generation Framework: material flow, EOAT, sensors, safety, controls, and lifecycle support.
7. Evaluate automation value beyond labour savings, including quality, throughput, flexibility, data, and ergonomics.
8. Identify technical and organisational risks such as poor process design, over-specification, and under-utilisation.

1.1 Introduction

The word ‘robot’ entered the public vocabulary through Karel Čapek’s 1920 play R.U.R. (Rossum’s Universal Robots), where it described artificial workers. A century later, ISO 8373:2021 defines an industrial robot as an automatically controlled, reprogrammable, multipurpose manipulator, programmable in three or more axes, intended for automation applications in an industrial environment. The journey from theatrical metaphor to a standardised engineering concept passed through decades of technical development, each adding capabilities, constraints, and integration practices that shape today’s robotic production systems.

This chapter does not attempt a comprehensive history of robotics — several excellent monographs serve that purpose. Instead, it traces the engineering trajectory that is most relevant to the design and operation of production systems: the progressive integration of the robot into its surrounding infrastructure, from a caged manipulator executing a fixed program to a networked, sensor-rich, adaptive element of a digitally connected factory.

1.2 Historical Evolution of Industrial Robotics

For the purposes of this course and textbook, the evolution of industrial robotics is organised into four overlapping eras. This is a pedagogical taxonomy rather than a universally standardised periodisation; each era is characterised by a dominant technology enabler and a representative mode of deployment.

1.2.1 Era 1 — Pioneering (1961–1975)

George Devol’s patent for a ‘Programmed Article Transfer’ (1954) and the subsequent founding of Unimation with Joseph Engelberger led to the installation of the first Unimate robot at General Motors’ Ternstedt plant in 1961. The robot performed die-casting extraction—a dangerous and repetitive task that justified automation despite the high cost and limited flexibility of early hydraulic manipulators. These systems were largely standalone, with limited sensing—primarily internal position feedback and hardwired interlocks—and little or no communication with surrounding equipment.

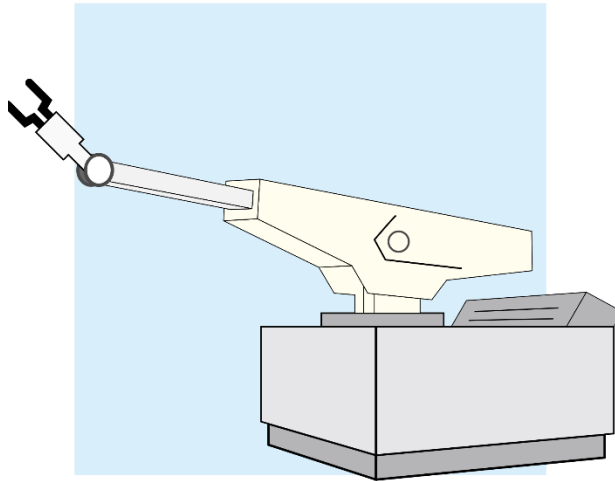


Figure 1.1. Schematic representation of a polar (spherical) robot configuration, inspired by the Unimate—the first industrial robot deployed in commercial production at General Motors in 1961. The configuration uses rotational and prismatic motion to position the end-effector within an approximately spherical workspace. Source: EBattleP, Wikimedia Commons, licensed under CC BY-SA 4.0.

1.2.2 Era 2 — Industrialisation (1975–1995)

The introduction of microprocessor-based controllers in the mid-1970s transformed industrial robots into increasingly precise and programmable production tools. Electric servo drives progressively replaced hydraulic actuation in many applications; representative repeatability improved from the millimetre scale of early systems to approximately ± 0.1 mm in mature industrial installations. Programming moved from drum sequencers and hardwired logic to teach pendants with point-to-point storage. The SCARA architecture, developed under Hiroshi Makino's leadership and commercialised in the early 1980s, introduced a robot type optimised for high-speed horizontal assembly. Arc-welding and paint-spraying applications, driven strongly by automotive demand and occupational-safety requirements, became major deployment areas.

1.2.3 Era 3 — Intelligent Integration (1995–2015)

Vision systems, force/torque sensors, and industrial fieldbus communication transformed the robot from a largely open-loop task executor into a responsive element of a coordinated production system. Offline programming (OLP) decoupled part of the programming and validation effort from the production floor. The first edition of ISO 10218-1, published in 2006, formalised safety requirements for industrial robots and supported more systematic risk reduction. The commercial introduction of the UR5 in

2008 accelerated the adoption of collaborative applications in which robots and humans may share parts of a workspace, subject to application-specific risk assessment and appropriate safeguarding.

1.2.4 Era 4 — Connected and Adaptive (2015–present)

Industry 4.0 concepts—cyber-physical systems, IIoT, edge and cloud computing, and digital twins—repositioned the robot as one node in a connected production environment. AI-based vision, reinforcement learning, and natural-language-assisted programming are extending robot capabilities beyond fixed, pre-programmed paths, while autonomous mobile robots are reducing dependence on fixed material-flow layouts. According to World Robotics 2025, 542,000 industrial robots were installed worldwide in 2024 and the global operational stock reached 4.664 million units.

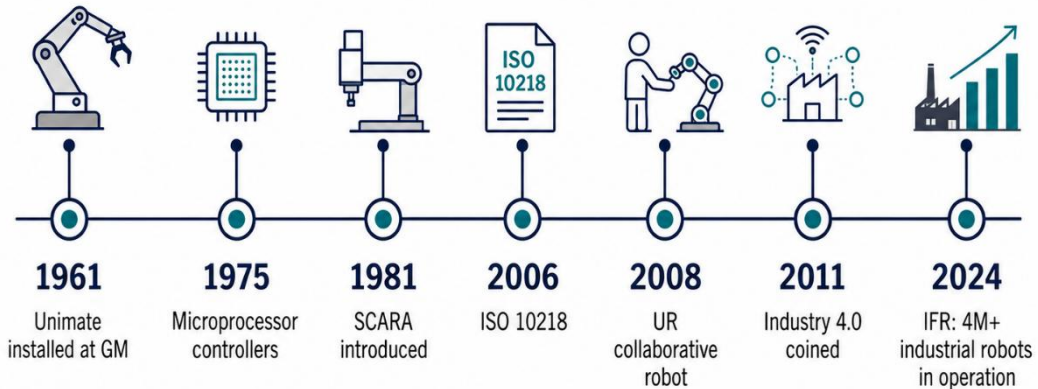


Figure 1.2. Selected milestones in industrial robotics: Unimate (1961), microprocessor controllers (mid-1970s), SCARA commercialisation (early 1980s), ISO 10218-1 (2006), UR5 introduction (2008), Industry 4.0 (2011), and the 2024 global stock and installation figures reported by IFR.

Table 1.1. Four eras of industrial robotics evolution.

Era	Period	Key Enabler	Robot Capability	Deployment Mode
Pioneering	1961–1975	Hydraulic actuation	Fixed-sequence manipulation; limited internal sensing	Standalone, safeguarded machines
Industrialisation	1975–1995	Microprocessor controllers and	Teach-pendant programming;	Application-specific robotic cells

Era	Period	Key Enabler	Robot Capability	Deployment Mode
		electric servo drives	improving repeatability	
Intelligent Integration	1995–2015	Vision, force sensing, fieldbus, OLP	Sensor-guided behaviour and coordinated operation	Integrated cells and production lines
Connected & Adaptive	2015–present	IIoT, AI, edge/cloud, digital twins	Data-driven adaptation, mobility, advanced analytics	Networked and reconfigurable production systems

WORKED EXAMPLE — Classifying a Robot Installation by Era

The following configurations are illustrative. Classify each installation using the four-era taxonomy and identify an upgrade that would move it toward the next capability level.

Installation A: A 1998 KUKA KR150 arc-welding cell is teach-pendant programmed, has no vision, and exchanges only hardwired I/O with relay logic. Classification: Era 2 (Industrialisation). Upgrade toward Era 3: add seam-tracking vision or sensing and replace relay logic with a PLC and industrial fieldbus.

Installation B: A 2018 ABB IRB 6700 machine-tending line uses four robots, machine vision, PROFINET IRT, and an OPC UA connection to MES. Classification: late Era 3 / transitional Era 4. Upgrade: add a maintained digital twin, condition-based analytics, and closed-loop use of production data for optimisation.

Installation C: A 2023 FANUC CRX-25iA cell combines AI-assisted bin picking, fleet supervision, and data exchange across similar cells. Classification: Era 4 (Connected and Adaptive). The engineering challenge is to verify that learning and adaptation deliver safe, repeatable production benefits.

1.3 Robotic Cells vs. Robotic Production Lines

The fundamental organisational unit of robotic production is the **robotic cell** — a bounded workspace containing one or more robots, their end-effectors, fixtures, sensors, and safety perimeter, designed to perform a coherent set of operations on a workpiece or product. A **robotic production line** connects multiple cells (or stations, some of which may be manual or semi-automated) through material-handling systems, sharing a common control architecture and producing a finished or semi-finished product.

Table 1.2. Comparison of robotic cells and robotic production lines. Ranges are indicative and depend on process complexity, geography, regulatory requirements, production volume, and the degree of integration.

Characteristic	Robotic Cell	Robotic Production Line
Scope	Single operation or small operation group	Complete production sequence
Robots	1–3 typically	10–200+
Investment	€80k–€500k	€2M–€50M+
Flexibility	High (reprogrammable)	Lower (interdependent stations)
Throughput	Moderate	Very high
Commissioning	2–6 weeks	3–12 months
Typical application	Welding cell, machine tending	BIW assembly, packaging line

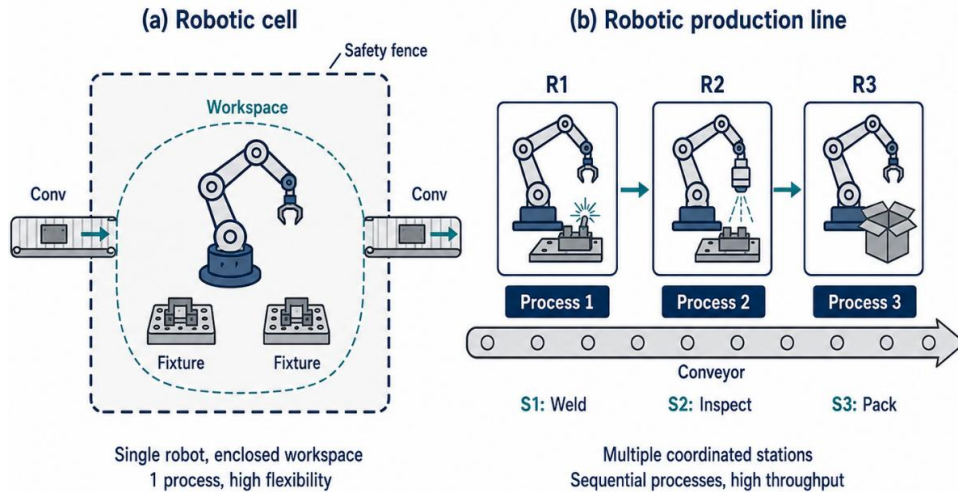


Figure 1.3. Conceptual comparison between (a) a robotic cell, with local safeguarding and one or more related operations, and (b) a robotic production line, with multiple coordinated stations connected by shared material and information flows.

WORKED EXAMPLE — Choosing Between a Cell and a Line

A manufacturer produces 50,000 units/year of a consumer-electronics product requiring four assembly operations. Each operation takes 20–30 s. Demand is expected to grow by 15% annually for five years. Should the company invest initially in a robotic cell or in a connected production line?

Current demand analysis: $50,000 \text{ units/year} \div 250 \text{ working days} \div 16 \text{ h/day} = 12.5 \text{ units/h}$, corresponding to a takt time of 288 s. The four operations require approximately 100 s of processing; allowing 20 s for handling and transfer gives a nominal cell cycle of 120 s. At 85% planned availability, sustainable capacity is $3,600/120 \times 0.85 = 25.5 \text{ units/h}$. Current capacity loading is therefore about 49%.

Year 5 demand: $50,000 \times 1.15^5 = 100,568 \text{ units/year}$, or approximately 25.1 units/h. The corresponding takt time is 143 s. One cell would then operate at approximately 99% of its sustainable capacity, leaving inadequate margin for changeovers, variability, maintenance, and recovery from faults.

Decision: Begin with one modular cell, but reserve layout, utilities, controls, and material-flow capacity for a second parallel station when loading reaches 80–85%, expected in years 3–4. A dedicated line is premature at the initial volume. Key

principle: size the architecture from demand trajectory and sustainable—not nominal—capacity.

1.4 Industry 4.0 and Industry 5.0 Paradigms

The Industry 4.0 concept was introduced in Germany in 2011 and developed strategically in the Acatech recommendations of 2013. For application-level analysis in this course, four frequently cited design principles—interconnection, information transparency, technical assistance, and decentralised decisions—are used following the synthesis proposed by Hermann, Pentek, and Otto (2016):

INDUSTRIAL PRACTICE NOTE — The Four Principles of Industry 4.0

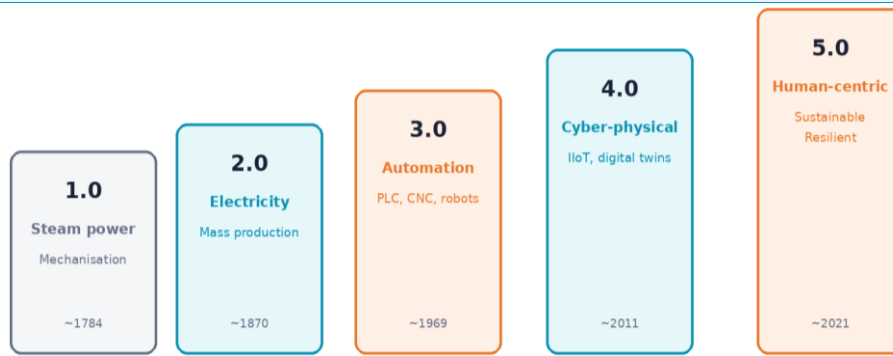
(1) INTERCONNECTION: machines, sensors, and humans are connected via the Industrial Internet of Things. Every robot, PLC, sensor, and HMI becomes a node in a network that exchanges data in real time. (2) INFORMATION TRANSPARENCY: systems create virtual replicas (digital twins) of the physical factory. The digital twin enables simulation, prediction, and optimisation before physical action. (3) TECHNICAL ASSISTANCE: systems support humans in decision-making and in performing difficult or unsafe tasks. Cobots, AR-guided assembly, and AI-based quality inspection are manifestations. (4) DECENTRALISED DECISIONS: systems make as many decisions as possible autonomously. Edge computing on robot controllers, self-optimising welding parameters, and AMR fleet managers that reroute around obstacles are examples.

Industry 5.0, articulated by the European Commission in 2021, complements rather than replaces Industry 4.0. It adds three value-oriented dimensions that broaden the design objectives of industrial systems beyond digital performance alone:

INDUSTRIAL PRACTICE NOTE — The Three Pillars of Industry 5.0

(1) HUMAN-CENTRICITY: technology should adapt to the worker, not vice versa. Collaborative robotics, ergonomic workstation design, and AI-augmented decision

support all serve this principle. The measure of success is not robot utilisation but total system performance including human well-being. (2) SUSTAINABILITY: production systems must minimise resource consumption, enable circular economy principles, and support green manufacturing. Energy per unit, embodied carbon, and end-of-life recovery are emerging KPIs alongside OEE and cost. (3) RESILIENCE: production must be robust against disruptions (pandemics, supply chain shocks, geopolitical events) through distributed, flexible, and reconfigurable architectures. RMS (Chapter 12) is the architectural response to resilience demands.



Industry 5.0 complements the digital foundation of Industry 4.0 with human-centricity, sustainability, and resilience.

Figure 1.4. Simplified progression from mechanisation and mass production to automation and cyber-physical production. Industry 5.0 is presented as an emerging design and policy paradigm that complements Industry 4.0 through human-centricity, sustainability, and resilience.

WORKED EXAMPLE — Industry 4.0 Readiness Assessment for a Robotic Cell

Assess the Industry 4.0 readiness of a welding cell with a KUKA KR20 robot, a Fronius TPS 500i power source, hardwired PLC interfaces, no MES connection, and manual quality inspection. Use the following illustrative screening scale for each principle: 1 = absent/manual; 2 = isolated or basic digital function; 3 = connected and data-collecting; 4 = integrated with analytics and decision support; 5 = adaptive or decentralised operation.

Interconnection: 2/5. The robot and welding source exchange basic digital I/O. There is no fieldbus, OPC UA interface, or higher-level data connection. Upgrade path:

migrate process communication to an appropriate industrial network and introduce a governed OPC UA gateway.

Information transparency: 1/5. There is no maintained virtual model and no systematic process-data history beyond PLC alarms. Upgrade path: develop the cell in a KUKA-compatible offline simulation environment, such as KUKA.Sim or iiQWorks.Sim, and implement time-stamped weld-parameter logging.

Technical assistance: 2/5. The robot removes the operator from the welding task, but quality inspection and fault interpretation remain manual. Upgrade path: add automated weld-bead inspection and an HMI that presents actionable diagnostic information.

Decentralised decisions: 1/5. All decisions are pre-programmed or made by the operator. Upgrade path: introduce validated adaptive adjustment of selected welding parameters using real-time process feedback, within defined safety and quality limits.

Overall screening result: 6/20 (30%). This is a legacy installation with limited Industry 4.0 functionality. A phased programme—connectivity and data acquisition first, followed by simulation, automated inspection, and constrained adaptive control—could raise the illustrative score to approximately 14–16/20. A budget range of €25–40k may be plausible for selected hardware and software, but engineering effort, cybersecurity, validation, and production downtime must be estimated separately.

1.5 Robot Density and Global Deployment Trends

The International Federation of Robotics (IFR) publishes annual data on robot installations, operational stock, and robot density—the number of operational industrial robots per 10,000 manufacturing employees. World Robotics 2025 reports 542,000 new installations and a global operational stock of 4.664 million robots in 2024. In April 2026, IFR published updated 2024 density values based on revised labour-market data. These metrics provide a useful high-level view of adoption, but they require careful interpretation.

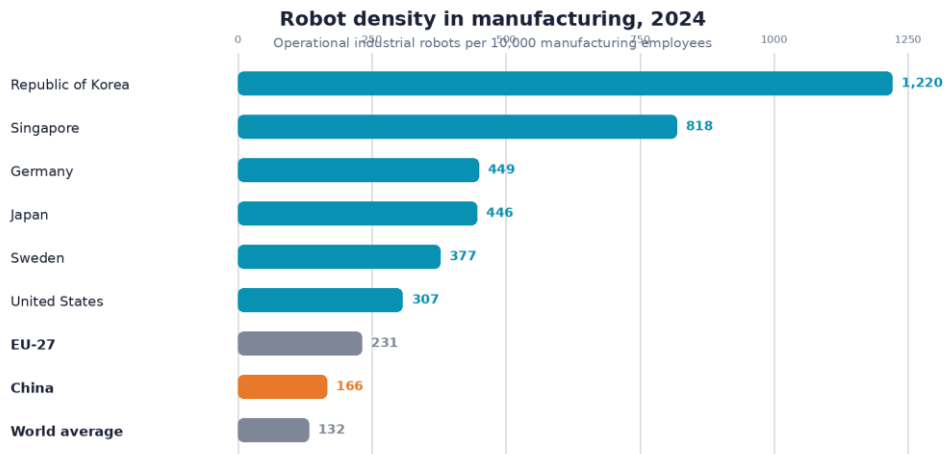
Table 1.3. Selected robot-density values for 2024, published by IFR in April 2026. Density = operational industrial robots per 10,000 manufacturing employees.

Economy / region	Density (2024)	Manufacturing context	Interpretation
Republic of Korea	1,220	Electronics and automotive	Highest reported national density
Singapore	818	Electronics and semiconductors	High density with a small manufacturing workforce
Germany	449	Automotive, machinery, diversified manufacturing	Highest density in Europe
Japan	446	Automotive, electronics, robot manufacturing	Major producer and user of industrial robots
Sweden	377	Advanced manufacturing	Among the five highest national densities
United States	307	Large and heterogeneous manufacturing base	8th worldwide
EU-27	231	Regional benchmark	Above the world average
China	166	Very large and geographically diverse manufacturing base	Revised labour-force denominator; about 2 million robots in operation
World average	132	Global benchmark	Reference value for international comparison

INDUSTRIAL PRACTICE NOTE — Reading IFR Density Metrics

Robot-density rankings can mask structurally different production systems. A high value may result from concentrated electronics or automotive investment, a small manufacturing workforce, or broad automation across many plants. When benchmarking, combine national density with sector-specific adoption, operational stock, application mix, plant scale, labour productivity, and production structure.

Robot density measures adoption intensity; it does not directly measure competitiveness, profitability, OEE, or technological sophistication.



Source: International Federation of Robotics, public release of 8 April 2026 (2024 data).

Figure 1.5. Selected 2024 robot-density values reported by IFR in April 2026. Regional and global averages are included as reference points; national values should not be interpreted without considering workforce structure and sector mix.

WORKED EXAMPLE — Analysing Robot Adoption by Sector

World Robotics 2025 reports the following 2024 installation data: electrical/electronics, 128,899 units (+2% year-on-year; 2019–2024 CAGR +8%); automotive, 126,088 units (–7% year-on-year; CAGR +5%); and metal and machinery, 88,777 units (+16% year-on-year; CAGR +12%). A student argues that automotive is in structural decline because its 2024 installation count fell by 7%. Evaluate this claim.

One-year change versus long-term trend: The –7% figure describes the change from 2023 to 2024, whereas the five-year CAGR remains positive at +5%. A single-year contraction can reflect investment cycles, project timing, macroeconomic conditions, or transitions in vehicle technology; it does not by itself establish long-term decline.

Scale and maturity: Automotive remained the second-largest customer industry in 2024, only about 2,800 units behind electrical/electronics. Its mature installed base

and cyclic investment pattern differ from the faster expansion of metal and machinery, which grew from a smaller base and recorded a +12% five-year CAGR.

Correct interpretation: Electrical/electronics reclaimed the leading position in 2024, automotive remained a very large but cyclical market, and metal and machinery showed the strongest five-year growth among the three sectors. Sector attractiveness should therefore be assessed using both absolute volume and multi-year growth, not one isolated percentage.

1.6 The Value Generation Framework

The central thesis of this course and textbook can be stated simply: a robot is a general-purpose manipulator that becomes a productive asset only when embedded in a well-designed system. The system—not the robot alone—generates value. This thesis has six corollaries, each corresponding to an integration dimension:

- 1. Material flow.** Parts must arrive at the right place, at the right time, in the right orientation. Without reliable feeding and transport, the fastest robot in the world stands idle.
- 2. End-of-arm tooling.** The robot must grasp, manipulate, process, and release the workpiece reliably. EOAT-related faults are a frequent source of unplanned stops because small errors in gripping, sensing, utilities, or part presentation propagate directly into the cycle.
- 3. Sensor integration.** The system can detect errors, verify quality, and adapt to variability. Sensors close the loop between the physical process and the control system.
- 4. Safety.** Workers and equipment are protected through engineered safety measures. Safety is not a constraint on productivity; it is a prerequisite for it.
- 5. Control architecture.** All components communicate reliably and synchronise their actions. The control architecture is the nervous system of the robotic production system.
- 6. Lifecycle support.** The system is maintained, monitored, documented, and continuously improved over its operational life. Without structured maintenance

and configuration control, availability, process capability, and diagnostic quality deteriorate progressively.

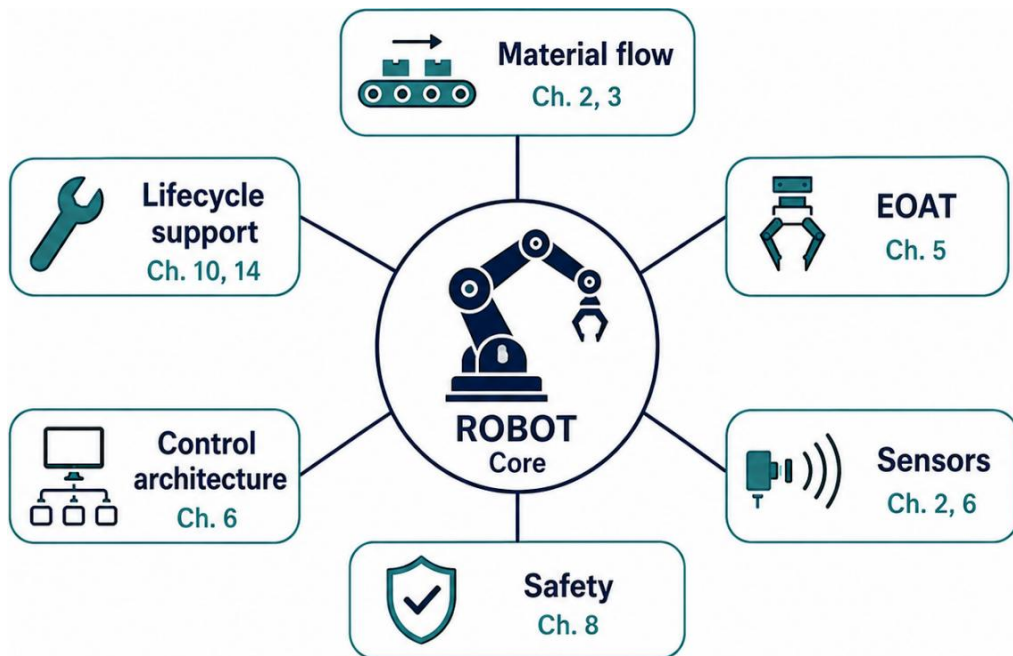


Figure 1.6. The robot core surrounded by six system-integration dimensions: material flow, EOAT, sensors, safety, control architecture, and lifecycle support. The chapter references indicate where each dimension is developed in the course and textbook.

WORKED EXAMPLE — Diagnosing a Cell's Underperformance Using the Value Framework

A palletising cell with a FANUC M-710iC robot achieves only 65% of its design throughput. The robot itself operates correctly. Use the six-dimension framework to systematically diagnose the problem.

Material flow: The infeed conveyor delivers boxes in random orientation. The robot must re-orient 40% of boxes before palletising, adding 3 s per affected cycle. **Root cause identified.** Solution: add a mechanical orientor upstream (€8,000). Expected improvement: +25% throughput.

EOAT: The vacuum gripper is correctly sized and performs reliably. No issue is identified in this dimension.

Sensors: Box-presence sensor triggers 0.4 s late due to mounting position. **Contributing factor.** Solution: reposition sensor (€200, 1 hour). Expected improvement: +5%.

Safety / Control / Lifecycle: No issues identified in these dimensions.

Diagnosis: Two of the six dimensions are degraded. The estimated multiplicative effect is $0.75 \times 0.95 = 0.7125$, which is of the same order as the observed 65% throughput. The remaining gap should not be dismissed as random variation; it requires measurement of micro-stoppages, acceleration limits, queueing, and recovery time. Total remediation cost: €8,200. Dividing the observed 65% by 0.7125 gives an estimated post-remediation throughput of about 91% of design. **Key lesson:** the robot was not the primary fault source; the surrounding system constrained its performance.

1.7 Economic and Strategic Motivations for Robotic Automation

The traditional justification for robotic automation is labour cost reduction: a robot replaces n workers, the investment pays back in m years. This framing, while valid for high-labour-content operations, captures only a fraction of the value that well-designed robotic systems generate. Six categories of benefit should be evaluated in any automation business case:

Table 1.4. Six categories of automation benefit. The numerical ranges are illustrative screening values, not universal benchmarks; each business case requires application-specific data and sensitivity analysis.

Benefit Category	Mechanism	Typical Magnitude	Measurability
Direct labour savings	Fewer operators per shift	20–60% of labour cost	High (headcount $\times$ cost)
Quality improvement	Consistent process, fewer defects	30–90% defect reduction	High (scrap + rework cost)
Throughput increase	Faster cycle, higher availability	20–50% more output	High (units $\times$ margin)
Flexibility	Quick changeover, multiple variants	Enables product-mix changes	Medium (opportunity value)
Data generation	Process data for CI and AI	Foundation for Industry 4.0	Low (strategic option value)
Ergonomic improvement	Removes hazardous/repetitive tasks	€5–20k/year injury avoidance	Medium (insurance + absence)

A complete business case evaluates all six categories while avoiding double counting. Labour-only calculations may understate the value of quality, sustainable capacity, flexibility, data infrastructure, and ergonomic risk reduction. Conversely, projected benefits should be included only when they are measurable, attributable to the project, and supported by realistic demand and availability assumptions. The most sophisticated organisations also evaluate the option value of installed data infrastructure—the ability to deploy AI-based inspection, condition monitoring, or digital-twin analytics once reliable data acquisition is already in place.

INDUSTRIAL PRACTICE NOTE — Beyond Labour Replacement: The Full Business Case

Illustrative teaching case, with rounded values: a Romanian automotive supplier evaluates a robotic machine-tending cell requiring an investment of approximately €456k. Direct labour savings are estimated at €109k/year, giving a labour-only simple payback of $456/109 \approx 4.2$ years. The validated full business case also includes €85k/year from scrap and rework reduction, €120k/year in contribution margin from confirmed additional capacity, and €12k/year in quantified ergonomic and absence-related benefits. Total annual benefit becomes approximately €326k, giving a simple payback of $456/326 \approx 1.4$ years. The lesson is not that every project achieves this result, but that all material and non-overlapping benefit categories should be evaluated transparently before a decision is made.

1.8 Book and Course Roadmap: From Concept to Commissioning

This textbook is organised into four Parts that correspond to the four teaching modules of the CESPR course. The chapter sequence therefore functions both as the architecture of the book and as the progression of the semester:

- Part I / Module I — Concepts and Fundamentals (Chapters 1–3; Weeks 1–3): robots in manufacturing, system components and flows, and layout design; establishes the vocabulary and system-level perspective.
- Part II / Module II — Design and Integration (Chapters 4–7; Weeks 4–7): cell-design methodology, robot and EOAT selection, controls, simulation, and validation; the engineering core of the course.
- Part III / Module III — Operation, Safety, and Performance (Chapters 8–10; Weeks 8–10): safety, monitoring and KPIs, maintenance, reliability, and lifecycle management.
- Part IV / Module IV — Advanced Topics and Industry Trends (Chapters 11–14; Weeks 11–14): collaborative and AI-enabled robots, adaptive/mobile systems, industrial case studies, and commissioning.

Each chapter follows a consistent structure: Learning Objectives, theoretical foundations, worked examples with numerical calculations, industrial practice notes, common pitfalls, key takeaways, review questions, and references. Most worked examples are designed to be solvable with a calculator and the formulas in the text; where specialist software is required, this is stated explicitly.

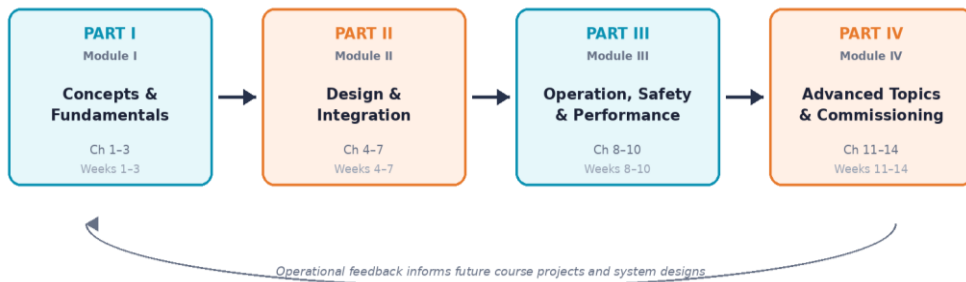


Figure 1.7. Book-and-course roadmap: the four Parts correspond to the four CESP modules; operational feedback informs later projects and future system designs.

Common Pitfalls

Pitfall 1: Treating the robot as the system.

Procurement teams that optimise robot selection while neglecting EOAT, feeding, sensing, safety, controls, utilities, and commissioning consistently produce underperforming cells. In many integrated applications, the robot represents only a fraction of the total project cost; the surrounding engineering largely determines performance.

Pitfall 2: Automating a bad process.

Robotic automation amplifies whatever process it is applied to — including its defects. A manual process with 5% scrap rate, when automated without process improvement, becomes a robotic process with 5% scrap at higher speed. Fix the process first.

Pitfall 3: Over-specifying the robot.

Selecting a 200 kg payload robot for a 15 kg application ‘because we might need it later’ can waste capital, floor space, and energy. Selection must consider not only payload but also centre of mass, inertia, reach, process loads, accuracy, duty cycle, mounting orientation, and dynamic performance. Apply a documented engineering margin rather than a generic percentage.

Pitfall 4: Ignoring the human element.

Every robotic cell interacts with operators who load, supervise, maintain, and troubleshoot. Systems designed without operator input produce frustration, workarounds, and chronic underperformance.

Pitfall 5: Confusing robot density with competitiveness.

Robot count alone does not reveal sustainable output, quality, flexibility, labour productivity, or utilisation. A large installed base can coexist with poor OEE and chronic losses. Density measures adoption intensity, not operational effectiveness.

Pitfall 6: Assuming Industry 4.0 requires a greenfield factory.

Many Industry 4.0 implementations are retrofits on existing equipment. An industrial gateway or edge device may be a modest hardware investment, but compatibility, data semantics, cybersecurity, validation, engineering effort, and lifecycle support must be assessed before claiming successful integration.

Pitfall 7: Treating Industry 5.0 as marketing.

European sustainability-reporting and supply-chain requirements continue to evolve, while major customers increasingly request auditable data on energy, emissions, materials, and working conditions. Engineers who dismiss Industry 5.0 as marketing may find that their systems cannot produce the evidence required by customers, regulators, or internal governance.

Pitfall 8: Benchmarking against national averages instead of sector data.

A national robot-density value masks large differences between sectors, plant sizes, regions, and production strategies. Automotive, electronics, food, and general manufacturing can display radically different adoption patterns within the same country. Benchmark against comparable processes and sectors, not only against a national headline figure.

Pitfall 9: Expecting immediate ROI from connectivity.

Connecting a cell to the cloud generates data. Data generates value only when someone analyses it, makes decisions based on it, and acts. Budget the analytics and decision-making effort alongside the connectivity infrastructure.

Pitfall 10: Skipping the value-generation framework during design.

The six-dimension check—material flow, EOAT, sensors, safety, control, and lifecycle support—should be applied at every design review. Early verification of all six dimensions substantially reduces avoidable commissioning surprises.

KEY TAKEAWAYS

- The four eras used in this course are a pedagogical taxonomy tracing the progression from standalone manipulators to connected and adaptive production systems.
- Robotic cells and production lines serve different contexts; architecture must reflect demand, variety, sustainable capacity, flexibility, and investment horizon.
- Industry 4.0 provides connectivity and data capabilities, while Industry 5.0 broadens the objective toward human-centricity, sustainability, and resilience.
- Robot-density and sector statistics indicate adoption, not competitiveness; they must be interpreted with workforce, sector, operational-stock, and plant-performance data.
- The central thesis is system-level: value is created by integrating material flow, EOAT, sensors, safety, controls, and lifecycle support around the robot.
- Industry 4.0 readiness can be improved incrementally, but scores are meaningful only when the assessment rubric is explicit and consistently applied.
- Economic justification extends beyond labour savings, but benefits must be measurable, attributable, non-overlapping, and based on realistic assumptions.
- The engineer's task is to design, validate, commission, operate, and improve the production system—not merely to deploy a robot.

Review Questions

Answer the following questions to consolidate your understanding of the chapter material. A thorough response should draw on concepts, formulas, and examples presented in the text.

1. Trace the evolution of industrial robotics through the four-era taxonomy used in this course. For each era, identify the dominant technology enabler, a representative application, and one limitation of the classification.
2. A manufacturer produces 200 units/day of a product requiring six operations. Takt time is 144 s. Compare a single multi-operation robotic cell with a six-station line, including handling time, planned availability, changeovers, and capacity margin in your reasoning.
3. Explain the four principles of Industry 4.0 and give one concrete example of each as applied to a robotic welding cell.

4. IFR reports a 2024 robot density of 449 for Germany and 231 for the EU-27. Does this mean German factories are approximately 1.9 times more automated or competitive than the EU average? Critically evaluate the comparison and identify the additional data required.
5. Apply the Value Generation Framework to a robotic cell you have seen (in person, in video, or in a case study). Which of the six integration dimensions is strongest? Which is weakest?
6. A company's CEO states: 'We need to reach Industry 4.0 maturity within 2 years.' Design a phased upgrade plan for an existing Era 2 welding cell, specifying investments and expected I4.0 readiness scores at each phase.
7. Compare the economic justification for robotic automation in automotive (high volume, mature market) versus food & beverage (growing market, hygiene constraints). Which factors differ most between the two sectors?
8. The chapter claims that 'automating a bad process amplifies its defects.' Give two specific examples from manufacturing where this has occurred, and describe how the process should have been improved before automation.
9. Industry 5.0 adds sustainability and resilience as design objectives. How would these objectives change the specifications for a new robotic packaging line compared to a purely Industry 4.0-driven design?

References

- [1] International Federation of Robotics (2025) World Robotics 2025 — Industrial Robots. VDMA Services GmbH, Frankfurt.
- [2] Siciliano, B., Sciavicco, L., Villani, L., and Oriolo, G. (2009) Robotics: Modelling, Planning and Control. Springer, London. ISBN 978-1-84628-642-1.
- [3] Groover, M.P. (2020) Automation, Production Systems, and Computer-Integrated Manufacturing, 5th ed. Pearson. ISBN 978-0-13-460546-3.
- [4] Kagermann, H., Wahlster, W., and Helbig, J. (2013) Recommendations for Implementing the Strategic Initiative INDUSTRIE 4.0. Acatech, Munich.
- [5] Hermann, M., Pentek, T., and Otto, B. (2016) Design Principles for Industrie 4.0 Scenarios. Proceedings of the 49th Hawaii International Conference on System Sciences, pp. 3928–3937. doi:10.1109/HICSS.2016.488.
- [6] European Commission (2021) Industry 5.0: Towards a Sustainable, Human-Centric and Resilient European Industry. Publications Office of the European Union.
- [7] Nof, S.Y. (ed.) (1999) Handbook of Industrial Robotics, 2nd ed. John Wiley & Sons. doi:10.1002/9780470172506.
- [8] ISO 8373:2021, Robotics — Vocabulary. International Organization for Standardization, Geneva.

- [9] ISO 10218-1:2025, Robotics — Safety Requirements — Part 1: Industrial Robots. International Organization for Standardization, Geneva.
- [10] ISO 10218-2:2025, Robotics — Safety Requirements — Part 2: Industrial Robot Applications and Robot Cells. International Organization for Standardization, Geneva.
- [11] Xu, X., Lu, Y., Vogel-Heuser, B., and Wang, L. (2021) Industry 4.0 and Industry 5.0— Inception, Conception and Perception. *Journal of Manufacturing Systems*, 61, 530–535. doi:10.1016/j.jmsy.2021.10.006.
- [12] Mocan, B. et al. (2018) *Multidisciplinary Design of Industrial Robotic Automation Solutions—Practical Guide for Students*. UTPress. ISBN 978-606-737-246-5.
- [13] Mocan, B. (2025) *Design and Operation of Robotic Production Systems—Course Notes*. Technical University of Cluj-Napoca.
- [14] Čapek, K. (1920) *R.U.R. (Rossum’s Universal Robots)*. Prague.
- [15] Engelberger, J.F. (1980) *Robotics in Practice: Management and Applications of Industrial Robots*. Kogan Page, London.
- [16] Universal Robots (2018) *Universal Robots Celebrates 10-Year Anniversary of Selling the World’s First Commercially Viable Collaborative Robot*. Odense, Denmark.

CHAPTER

2

Components, Flows, and System Architecture

A system-level view of robotic production

This chapter examines the internal structure of a robotic production system as a coordinated ensemble of mechanical, electrical, control, safety, and information components. Industrial robots, end-of-arm tooling, sensors, conveyors, AGVs, buffers, and auxiliary devices are treated as elements of an interconnected whole rather than as independent purchases. Particular attention is given to three flows that cross subsystem boundaries: material, information, and energy.

The chapter emphasises that system losses frequently originate in poorly designed interfaces and flows rather than in the nominal performance of the robot. Synchronisation between stations, local versus system-level optimisation, bottleneck behaviour, and subsystem interdependencies are developed as foundations for the design methods used throughout the CESPR course.

LEARNING OBJECTIVES

1. Identify the four principal subsystems of a robotic production system (robotic, perirobotic, transport/transfer, associated) and explain their interdependencies.
2. Analyse the three fundamental flows through a production system: material, information, and energy.
3. Apply takt time and bottleneck analysis to identify system synchronisation problems and quantify throughput gaps.
4. Apply Little's Law ($WIP = Throughput \times Lead\ Time$) to diagnose WIP accumulation patterns and predict lead times.
5. Specify end-of-arm tooling type (vacuum, mechanical, magnetic, specialised) based on workpiece characteristics and task requirements.
6. Describe typical transport and transfer technologies (belt, roller, chain, pallet, AGV, AMR) and select the appropriate one for a given application.
7. Use Kingman's approximation and utilisation scenarios to estimate queueing sensitivity and support first-cut buffer decisions.
8. Calculate balance efficiency for a multi-station line and evaluate whether rebalancing is justified.
9. Map the four-level information flow hierarchy (field, control, supervisory, MES/ERP) to appropriate communication protocols.
10. Estimate energy consumption profiles for robotic cells and identify idle-mode waste as an improvement target.

2.1 Introduction: The System-Level View

A robotic production system can be analysed through four interacting subsystems that must be designed concurrently (Figure 2.1): the robotic subsystem (one or more industrial robots), the perirobotic subsystem (EOAT, fixtures, feeders, sensors, and

process peripherals), the transport and transfer subsystem (conveyors, AGVs, AMRs, indexers, and buffers), and the associated systems (safety, electrical power, compressed air, environmental controls, and information infrastructure). Each subsystem is rooted in a different engineering discipline, but system quality is determined primarily by their interfaces.

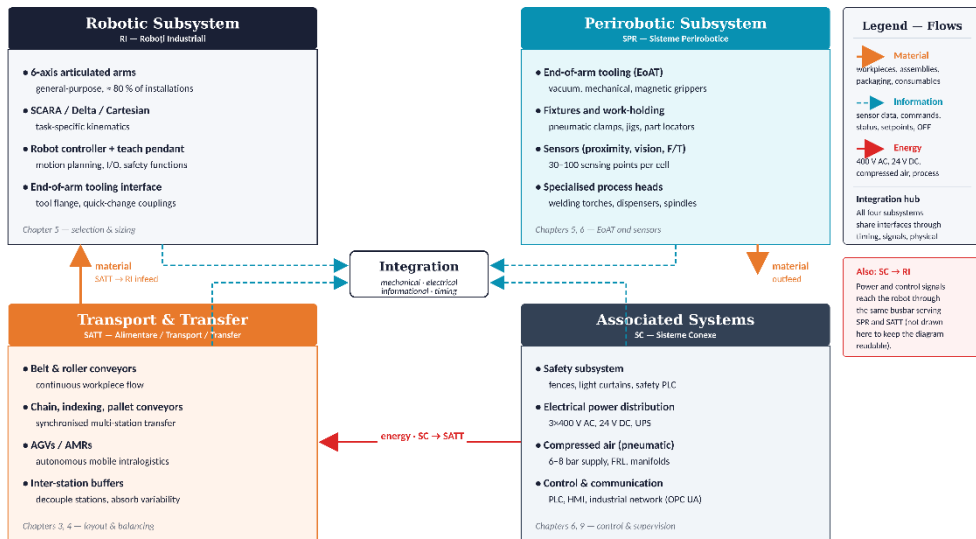


Figure 2.1. Four interacting subsystems and the material, information, and energy flows that connect them.

The decomposition commonly used in Romanian technical literature — RI + SPR + SATT + SC — maps to Roboți Industriali, Sisteme Perirotice, Sisteme de Alimentare/Transport/Transfer, and Sisteme Conexă. It is a useful design-review model: each component has a primary subsystem assignment, while cross-cutting elements such as smart EOAT, safety PLCs, and industrial networks must also be assessed at every interface they influence.

2.2 Components of a Robotic Production System

2.2.1 Industrial Robots (IR)

The industrial robot is the most visible component of the cell, but it is not automatically the dominant determinant of system performance. Robot selection, addressed in detail in Chapter 5, requires matching payload and inertia, reach envelope, repeatability, speed, degrees of freedom, mounting configuration, environmental rating, and controller

functions to the application. The following configurations are representative; payload and performance ranges are indicative and model-dependent:

- 6-axis articulated arms: widely used general-purpose manipulators for welding, material handling, painting, assembly, and machine tending.
- SCARA robots: high-speed, predominantly planar assembly and handling, typically with four controlled axes and strong use in electronics and small-part production.
- Delta/parallel robots: very high-speed pick-and-place for low-mass products, particularly in food packaging, pharmaceuticals, and consumer goods.
- Cartesian/gantry robots: rectangular work envelopes and structurally direct linear motion for machine tending, palletising, dispensing, and large-part handling.
- Dual-arm robots: coordinated bimanual manipulation for complex assembly, laboratory automation, and applications requiring two-handed dexterity; adoption remains application-specific.

Table 2.1. Indicative comparison of five industrial robot configurations; actual values depend on the selected model and operating conditions.

Robot Type	DOF	Typical Payload	Speed	Typical repeatability	Main Application
6-axis articulated	6	3–1,000+ kg	Medium–High	±0.02–0.1 mm	Welding, handling, painting
SCARA	4	1–50 kg	Very high	±0.01–0.05 mm	Electronics assembly
Delta/parallel	3–6	0.5–15 kg	Extreme	±0.05–0.1 mm	Pick-and-place, packaging
Cartesian/gantry	3	5–1,000 kg	Medium	±0.02–0.05 mm	Machine tending, large parts
Dual-arm	7–15	2–10 kg/arm	Low–Medium	±0.02–0.1 mm	Complex assembly

2.2.2 End-of-Arm Tooling (EOAT)

End-of-arm tooling is the engineered interface between the robot and the workpiece or process. EOAT selection can dominate reliability because it determines how forces, tolerances, contamination, utilities, and part variability are managed at the point of

contact. The price ranges below are indicative orders of magnitude for the tool itself and exclude engineering, validation, utilities, controls, and integration:

- Vacuum grippers: suction cups powered by Venturi generators or pumps; best for flat, non-porous surfaces (boxes, glass, sheet metal, plastic panels). Indicative hardware cost: €200–2,000.
- Mechanical grippers: 2-finger, 3-finger, or multi-finger with pneumatic or electric actuation; best for irregular shapes and high-grip-force applications. Indicative hardware cost: €500–5,000.
- Magnetic grippers: permanent or electromagnetic; restricted to ferrous materials but excellent for steel-plate handling. Indicative hardware cost: €300–2,000.
- Specialised tooling: welding torches, paint spray guns, dispensing nozzles, machining spindles, and custom task-specific end-effectors. Indicative hardware cost: €1,000–50,000+.

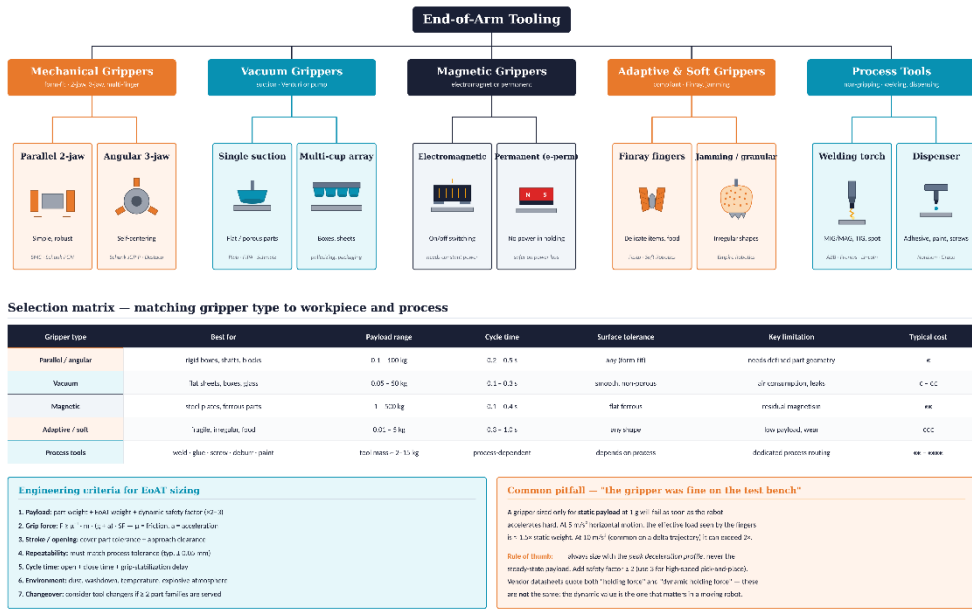


Figure 2.2. EOAT taxonomy, selection matrix, and engineering sizing guide.

Figure 2.2 combines a taxonomy of five EOAT families with a first-pass selection matrix and engineering sizing checks. The taxonomy begins with the task and workpiece rather than with a preferred vendor or actuation technology. Mechanical, vacuum, magnetic, adaptive/soft, and process tools differ in contact mechanism, tolerance to

variability, cycle-time capability, environmental compatibility, and failure mode. The matrix is intentionally qualitative; final selection must use manufacturer data, application trials, and the robot's payload–reach–inertia limits.

The selection matrix summarises first-pass trade-offs between the five families. Final selection must be validated against payload, cycle time, surface tolerance, environmental conditions, utility requirements, and failure consequences.

The engineering checks shown in Figure 2.2 include the combined EOAT–workpiece mass, centre of gravity, inertia, dynamic grip force, stroke and opening, repeatability, release confirmation, environmental conditions, utilities, wear, maintenance access, and changeover provisions. For friction gripping, the required normal force must be derived from the load case, acceleration, friction coefficient, jaw geometry, and a documented safety factor.

Engineering reminder: verify EOAT performance against the robot's most demanding acceleration and deceleration profiles, not only against static payload. The safety factor should be justified from uncertainty, consequence of loss of grip, redundancy, and applicable company or sector requirements rather than selected as a universal constant.

2.2.3 Sensors and Vision Systems

Sensors enable the system to observe state, verify transitions, detect faults, and adapt to controlled variability. Depending on complexity, a modern robotic cell may contain anything from a small set of discrete sensors to dozens of sensing and diagnostic points distributed across the four subsystems (Figure 2.3). The following cost and quantity ranges are indicative and vary substantially with performance, certification, optics, environmental rating, and integration scope:

- Proximity and presence sensors (inductive, capacitive, optical): detect whether a part is present at a fixture, gripper, or conveyor location. Indicative hardware cost: €20–200 each; quantity is application-dependent.
- Vision systems (2D, 3D, thermal, hyperspectral): identify, localise, inspect, and measure workpieces. Indicative hardware cost: €2,000–30,000 per camera, excluding lighting, optics, software, and integration.

- Force/torque sensors: enable force-controlled assembly, polishing, and delicate manipulation. Indicative hardware cost: €3,000–15,000; commonly mounted at the robot wrist or integrated into the robot structure.
- **Encoders and resolvers:** provide precise joint-position feedback for closed-loop motion control. Integrated in the robot; not separately purchased.
- Safety sensors (light curtains, laser scanners, safety mats): detect access to protected zones and initiate safety functions. Device positioning is governed by the application risk assessment and standards including ISO 13855:2024.

Table 2.2. Indicative characteristics of sensor technologies commonly used in robotic production cells.

Sensor Type	Detection Range	Response Time	Typical Cost	Application
Inductive proximity	1–40 mm	< 1 ms	€20–80	Metal part presence
Photoelectric	0.1–30 m	< 1 ms	€50–200	Part detection, counting
2D vision camera	FOV dependent	10–100 ms	€2,000–10,000	Part identification, inspection
3D structured light	0.3–2 m	50–500 ms	€5,000–25,000	Bin picking, measurement
Force/torque (6-axis)	Contact	< 1 ms	€3,000–15,000	Assembly, polishing
Safety laser scanner	0.5–60 m	< 80 ms	€2,000–8,000	Collaborative zone monitoring

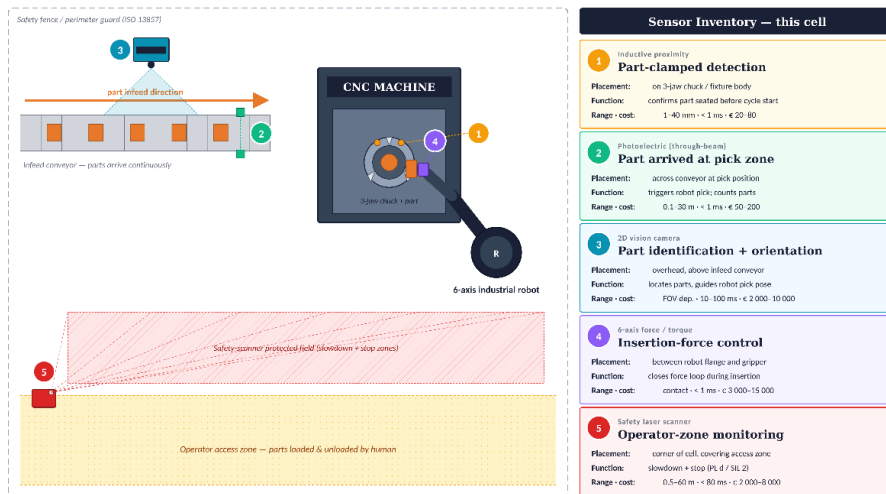


Figure 2.3. Sensor placement in a representative machine-tending cell, showing process, condition, and safety sensing functions.

2.2.4 Transport and Transfer Systems (SATT)

Transport and transfer systems (Table 2.3) move workpieces between stations. Selection depends on part mass and geometry, required throughput, accumulation strategy, positioning accuracy, cleanliness, maintainability, route flexibility, and interaction with robot pick and place windows:

Table 2.3. Indicative comparison of transport and transfer technologies.

Technology	Payload	Speed	Precision	Flexibility	Typical Use
Belt conveyor	0.1–100 kg	0.1–2 m/s	Low (± 5 mm)	Low (fixed path)	General transport
Roller conveyor	10–500 kg	0.1–1 m/s	Low	Low	Heavy parts, accumulation
Pallet transfer	1–50 kg	0.5–1 m/s	High (± 0.1 mm)	Medium	Precision indexing
AGV	100–10,000 kg	0.5–1.5 m/s	Medium (± 15 mm)	Medium (fixed path)	Heavy pallet transport
AMR	50–1,500 kg	1–2 m/s	Medium (± 25 mm)	High (free roaming)	Flexible intralogistics

INDUSTRIAL PRACTICE NOTE — Conveyor Selection Rules of Thumb

Three practical checks prevent many conveyor-related commissioning problems. (1) **SPEED AND CAPACITY:** derive conveyor capacity from the required arrival rate, spacing, acceleration, and accumulation strategy; a margin may be appropriate, but it should not create uncontrolled accumulation. (2) **ACCUMULATION:** size the zone from expected micro-stops and variability rather than from a universal number of part lengths. (3) **CONTACT SURFACE:** match belt or roller material to friction, contamination, wash-down, temperature, and product-damage requirements. Conveyor cost and complexity should be budgeted explicitly within the system specification.

2.2.5 Associated Systems

Associated systems provide the infrastructure without which the other three subsystems cannot function. They are often less visible than the robot and EOAT, yet safety, electrical distribution, industrial IT, pneumatics, and environmental control can represent a substantial share of the engineering effort, investment, validation scope, and commissioning risk:

- **Safety system:** safety PLC, emergency stops, interlocked gates, light curtains, and laser scanners. Safety-related control design should follow the project risk assessment and relevant standards, including ISO 13849-1:2023 or IEC 62061:2021 with its applicable amendments.
- **Electrical infrastructure:** main disconnect, distribution panel, servo drives, UPS for critical controllers. Typical cell: 3-phase 400V/50Hz, 16–63A.
- **Pneumatic system:** centralised compressor, FRL unit, distribution manifold, tubing. Typical cell: 6 bar, 200–500 NI/min.
- **IT infrastructure:** industrial Ethernet switches, secure remote access, OPC UA gateways, SCADA/MES interfaces, time synchronisation, network segmentation, and cybersecurity controls aligned with the IEC 62443 series.
- **Environmental controls:** lighting (200–1000 lx for vision), temperature ($\pm 2^{\circ}\text{C}$ for precision), dust extraction for grinding/polishing.

A common design error is to budget associated systems as a flat percentage of robot cost. This fails for complex cells where safety, IT, and environmental systems dominate. Budget each category explicitly based on the cell's requirements specification.

2.3 Material Flow, Information Flow and Energy Flow Analysis

2.3.1 Material Flow

Material flow (Table 2.4) describes how physical workpieces move through the system. It is analysed to identify unnecessary transport, re-handling, queues, starvation, blocking, long lead times, and layout-induced losses. Useful metrics include throughput, work-in-progress (WIP), lead time, value-added ratio, handling frequency, and travel distance. Targets are application-dependent and should be interpreted against customer demand, process physics, product mix, and a validated baseline.

Table 2.4. *Material-flow metrics, formulas, and diagnostic interpretation.*

Flow Metric	Formula	Reference / target	Diagnostic Value
Throughput (TH)	Parts completed / Time	At least customer demand	Below target = capacity gap
Work-in-Progress (WIP)	Parts in system at any time	Consistent with TH and LT	High WIP = hidden queues
Lead Time (LT)	Entry to exit time per part	Minimise waiting and transport	Long LT = excessive waiting
VA Ratio	VA time / LT	Improve against validated baseline	Low ratio = flow inefficiency
Touch Frequency	Handling events per part	Minimise non-value handling	High = re-handling waste
Travel Distance	Total path length per part	Minimise within layout constraints	Long = poor layout

WORKED EXAMPLE — Packaging Cell Material Flow Analysis

A robotic packaging cell receives 3,000 products/hour from an upstream line. Each product must be oriented, inspected, placed in a box, and the full box shrink-wrapped. The cell has one delta robot (cycle time 0.8 s/pick), one vision inspection station (0.6 s/product), and a wrapping station (4 s/box of 12 products).

Pick rate: $3,000/3,600 = 0.83$ pps required. Delta robot at 0.8 s/pick = 1.25 pps — sufficient (margin 50%).

Inspection: 0.6 s/product = 1.67 pps — sufficient.

Wrapping: $60/4 = 15$ boxes/min = 180 products/min = 10,800/h — sufficient.

Bottleneck: All stations have excess capacity. Takt time = $3,600/3,000 = 1.2$ s/product. The cell is demand-limited, not capacity-limited.

Buffer concept: Provide an accumulator between the delta robot and inspection and a small box buffer before the wrapper. Final capacities should be validated from arrival and service variability, recovery time, reject handling, and the permitted footprint; the values 20 products and 2 boxes are illustrative assumptions, not universal rules.

2.3.2 Information Flow

Information flow comprises sensor values, commands, state transitions, alarms, recipes, diagnostics, production records, and business transactions. Data volume can range from modest discrete I/O traffic to high-rate vision, waveform, and traceability streams. A layered architecture is useful, but protocol choice must reflect determinism, safety requirements, bandwidth, interoperability, cybersecurity, and lifecycle support rather than only nominal latency (Table 2.5):

Table 2.5. Four-level information-flow hierarchy with indicative latency scales and protocol roles.

Level	Latency	Data Type	Protocol Examples	Chapter
Device / field	0.4–30 ms typical	Sensor and actuator process data	IO-Link, AS-i, distributed I/O	Ch 6
Real-time control	<0.1–10 ms	Motion, PLC–robot handshakes, safety	EtherCAT, PROFINET RT/IRT, PROFIsafe	Ch 6

Level	Latency	Data Type	Protocol Examples	Chapter
Supervisory	0.1–10 s	HMI, alarms, cell status, recipes	OPC UA client/server, vendor APIs	Ch 9
MES / ERP	seconds–hours	Orders, genealogy, quality, OEE	OPC UA, MQTT, REST, XML/JSON	Ch 9

2.3.3 Energy Flow

Energy flow includes electrical power for servo drives and control electronics, compressed air for pneumatic EOAT and fixtures, and process-specific utilities such as welding power, laser energy, cooling water, extraction, or vacuum. Consumption varies from a few kilowatts for compact handling cells to much higher levels when the process equipment dominates. Compressed air and energised idle states are frequent improvement targets, but their significance must be established by measurement over a representative production cycle.

Figure 2.4 illustrates the integrated architecture of an industrial robotic cell and distinguishes the physical movement of material from information exchange and energy supply. The flows intersect at multiple interfaces; they do not merely converge at the robot.

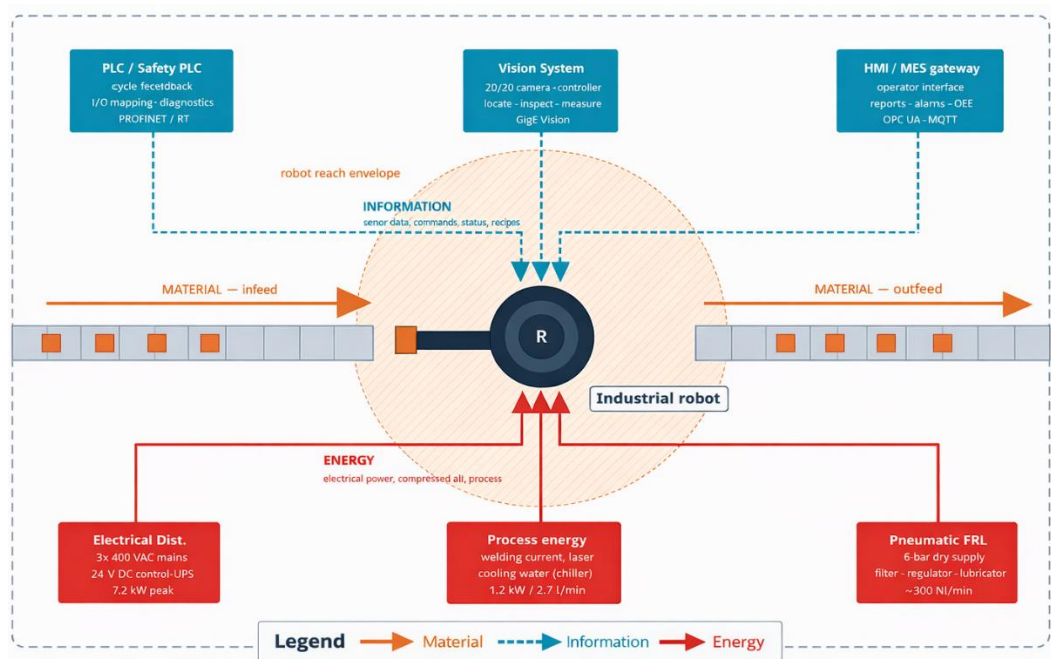


Figure 2.4. Material, information, and energy flows in an integrated robotic cell.

INDUSTRIAL PRACTICE NOTE — Quick Energy Audit for a Robotic Cell

A rapid screening audit can identify the largest measurement priorities, but a defensible energy baseline requires representative production data. (1) Measure peak and cycle-average electrical power at the main supply. (2) Measure energised idle power and verify available servo-sleep or equipment standby modes. (3) Measure compressed-air flow at the cell manifold and compare it with expected consumption; leaks can produce substantial annual cost depending on pressure and operating hours. (4) Log multiple production cycles and investigate unexplained variation. Quantify the opportunity before specifying a percentage reduction target.

WORKED EXAMPLE — Energy Cost Comparison: Pneumatic vs. Electric Gripper

A palletising cell operates 16 h/day, 250 days/year. For a deliberately conservative continuous-duty comparison, assume a pneumatic vacuum ejector consumes 0.5 Nm³/min throughout the operating period at €0.03/Nm³, while an electric vacuum pump draws 0.4 kW continuously at €0.15/kWh.

Pneumatic: $0.5 \text{ Nm}^3/\text{min} \times 60 \text{ min} \times 16 \text{ h} \times 250 \text{ days} \times €0.03/\text{Nm}^3 = €3,600/\text{year}$.

Electric: $0.4 \text{ kW} \times 16 \text{ h} \times 250 \text{ days} \times €0.15/\text{kWh} = €240/\text{year}$.

Result: Under these continuous-duty assumptions, the pneumatic option costs €3,600/year and the electric option €240/year, giving a difference of €3,360/year and a simple payback of approximately 4.3 months for a €1,200 price premium. Real projects must include duty cycle, vacuum holding valves, ejector efficiency, standby power, leakage, maintenance, noise, and process risk before generalising the result.

2.4 Synchronisation and Bottleneck Identification

In a robotic production system, synchronisation aligns the cycle capability of interconnected resources with the demand-driven takt time. Analysis should distinguish takt time, station cycle time, changeover and downtime losses, precedence constraints, parallel resources, and the behaviour of buffers under variability. The bottleneck is the

resource or constraint that limits achievable system throughput under the analysed operating conditions.

Takt time (TT) is the required average output interval needed to satisfy customer demand during the available production time:

$$\text{TT} = \text{Available production time per shift} / \text{Customer demand per shift}$$

Example: 450 min of net available time and a demand of 900 units give $\text{TT} = 30 \text{ s/unit}$. Station cycle times should be analysed against TT with explicit allowances for losses and variability. The longest deterministic station cycle is a first bottleneck indicator, but actual throughput can also be limited by failures, changeovers, starvation, blocking, or shared resources. Balance efficiency, $\sum \text{CT}_i / (n \times \max \text{CT}_i)$, is a useful screening metric; values such as 0.85 are often used as heuristics, not universal acceptance criteria.

INDUSTRIAL PRACTICE NOTE — Buffer Sizing with Kingman's Formula

Kingman's approximation $Wq \approx [p/(1-p)] \times [(Ca^2+Cs^2)/2] \times ts$ gives a first-cut estimate of mean queueing time for a single-server station, where p is utilisation, Ca and Cs are the coefficients of variation of arrivals and service, and ts is mean service time. For $p = 0.85$, $Ca = Cs = 0.3$, and $ts = 30$ s, $Wq \approx 15.3$ s. If utilisation rises to 0.95 with the other terms unchanged, Wq increases to approximately 51.3 s — about 3.35 times higher. Convert this estimate into buffer capacity only after accounting for the required service level, finite-buffer behaviour, blocking, starvation, failures, and recovery policy.

WORKED EXAMPLE — Little's Law Applied to a 5-Station Line

A conveyor-linked assembly line has 5 stations. Average cycle times: 28, 34, 42, 38, 31 s. Each inter-station buffer holds 6 parts; observed WIP is 34 parts. Compute throughput, lead time, and identify where WIP accumulates.

Bottleneck: Station 3 at 42 s. System throughput = $3,600/42 = 85.7$ parts/h.

Little's Law: $WIP = TH \times LT$. Rearranging: $LT = WIP/TH = 34/(85.7/3600) = 1,428$ s $\approx$ 23.8 min.

Interpretation: A part spends 23.8 min in the line but only 2.9 min in nominal station processing, giving a processing-time ratio of approximately 12%. Because Station 3 is the slowest station, WIP would be expected to accumulate primarily upstream of it; the actual buffer distribution must be confirmed from observations or event logs rather than inferred from total WIP alone.

INDUSTRIAL PRACTICE NOTE — Buffer Sizing with Kingman's Formula

Kingman's approximation $Wq \approx [p/(1-p)] \times [(Ca^2 + Cs^2)/2] \times ts$ gives a first-cut estimate of mean queueing time for a single-server station, where p is utilisation, Ca and Cs are the coefficients of variation of arrivals and service, and ts is mean service time. For $p = 0.85$, $Ca = Cs = 0.3$, and $ts = 30$ s, $Wq \approx 15.3$ s. If utilisation rises to 0.95 with the other terms unchanged, Wq increases to approximately 51.3 s — about 3.35 times higher. Convert this estimate into buffer capacity only after accounting for the required service level, finite-buffer behaviour, blocking, starvation, failures, and recovery policy.

WORKED EXAMPLE — Five-Station Bottleneck Analysis

A production line has five stations: $S1=28s$, $S2=32s$, $S3=45s$, $S4=30s$, $S5=27s$. Takt time $TT=35s$. Analyse.

Bottleneck: $S3$ at 45s exceeds TT . Line cannot meet takt. Throughput = $3,600/45 = 80$ parts/h. Required = $3,600/35 = 103$. Gap: 23 parts/h (22%).

Balance efficiency: $(28+32+45+30+27)/(5 \times 45) = 162/225 = 0.72$. This indicates substantial imbalance and justifies evaluating reallocation or parallelisation, but the decision threshold remains application-dependent.

Options: (1) Decompose $S3$ and move work to stations with available capacity. (2) Duplicate all or part of $S3$ in parallel. (3) Reduce $S3$ cycle time through trajectory optimisation, overlapping operations, fixture or EOAT redesign, and removal of waiting states. (4) Add operating time if the shortfall is temporary. Compare alternatives using throughput, quality, risk, and lifecycle cost.

INDUSTRIAL PRACTICE NOTE — Buffer Sizing with Kingman's Formula

Kingman's approximation $Wq \approx [p/(1-p)] \times [(Ca^2+Cs^2)/2] \times ts$ gives a first-cut estimate of mean queueing time for a single-server station, where p is utilisation, Ca and Cs are the coefficients of variation of arrivals and service, and ts is mean service time. For $p = 0.85$, $Ca = Cs = 0.3$, and $ts = 30$ s, $Wq \approx 15.3$ s. If utilisation rises to 0.95 with the other terms unchanged, Wq increases to approximately 51.3 s — about 3.35 times higher. Convert this estimate into buffer capacity only after accounting for the required service level, finite-buffer behaviour, blocking, starvation, failures, and recovery policy.

WORKED EXAMPLE — Takt Time Response to Demand Change

A line operates at $TT = 40$ s/unit (2 shifts $\times$ 450 min, demand 1,350 units/day). The customer increases demand by 25% to 1,688 units/day. Can the line respond without capital investment?

New TT: $(2 \times 450 \times 60) / 1,688 = 32.0$ s/unit. Current bottleneck CT = 38 s. Gap: $38 - 32 = 6$ s.

Option A — Add a shift: 3 shifts $\times$ 450 min. New $TT = (3 \times 450 \times 60) / 1,688 = 48.0$ s. Bottleneck at 38 s is well within takt. Cost: additional shift labour ($\approx 30\%$ increase in operating cost).

Option B — Reduce bottleneck CT: The station must reach $CT \leq 32$ s, requiring a 6 s reduction (approximately 16%). Potential measures include robot-path optimisation, concurrent EOAT actions, pre-positioning, elimination of control delays, fixture redesign, or splitting a sub-operation. SMED is relevant only when the lost time is associated with setup or changeover rather than the repetitive station cycle.

Decision: Option B is preferable when demand is sustained and the improvement is technically robust; an additional shift is more reversible when the demand increase is temporary. The comparison must include labour availability, maintenance windows, quality risk, and capital recovery. Key principle: takt time is demand-driven and must be recalculated whenever available time or demand changes.

Figure 2.5 presents the synchronisation analysis of a five-station serial line relative to $TT = 35$ s. Station S3, with $CT = 45$ s, is the deterministic bottleneck in the simplified example and constrains nominal output to 80 parts/h versus a required 103 parts/h.

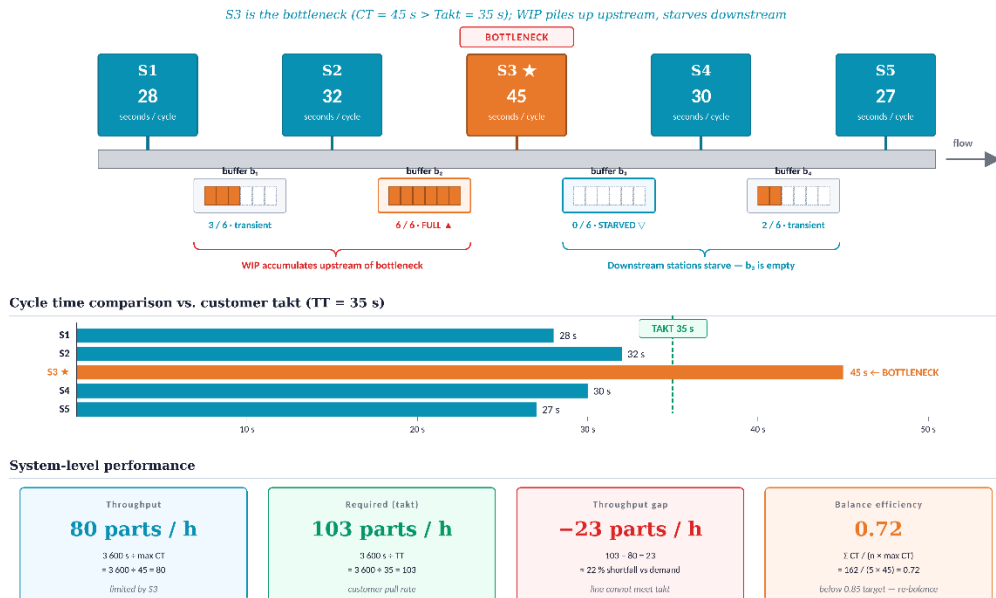


Figure 2.5. Five-station line showing takt time, station cycle times, representative buffer states, and system-level performance.

In the deterministic illustration, the upstream buffer is shown full and the downstream buffer empty to communicate the expected direction of queue formation and starvation around the bottleneck. In an actual line, buffer states fluctuate and should be verified through time-stamped production data. The line's nominal throughput gap is 23 parts/h and its balance efficiency is 0.72.

The example demonstrates why takt compliance must be assessed at system level: several stations may individually operate below takt while a single constraint prevents the line from meeting demand.

2.5 Subsystem Interdependencies

The four subsystems interact through shared loads, interfaces, timing constraints, utilities, data, safety functions, and failure propagation. Understanding these

interdependencies (Figure 2.6) is essential because a design change in one subsystem can alter the requirements or margins of the others (Table 2.6).

Table 2.6. Examples of subsystem interdependencies and change propagation.

Change in...	Propagates to...	Example
Robot speed (↑)	Sensor response (↑), Energy (↑), Safety zones (↑)	Faster robot needs faster vision trigger and larger safety distance
EOAT mass (↑)	Robot payload margin (↓), Cycle time (↑)	Heavier gripper forces slower acceleration or robot upgrade
Conveyor speed (↑)	Buffer sizing (↓), Vision exposure (↓)	Faster belt needs smaller accumulation but shorter camera exposure
Safeguarding concept changed	Robot speed/modes, sensor coverage, safety functions	Fence removal requires a new risk assessment and may require scanners, monitored zones, or alternative safeguards
MES connection added	Data volume (↑), Cybersecurity scope (↑)	OPC UA endpoint needs firewall rules and certificate management

couples with → ↓ affects	RI — Robotic arm · controller · drives	SPR — Perirobotic EoAT · fixtures · sensors	SATT — Transport conveyors · AGV · buffers	SC — Associated safety · power · PLC · IT	Legend
RI Robotic	self	STRONG M I EoAT mass → payload Gripper geometry defines robot reach usage; sensor triggers at robot cycle rate.	STRONG T M Cycle sync · pick window Conveyor indexing must arrive within sdt; faster robot → tighter spec.	STRONG E I Power + safety distance Faster robot → larger ISO 13855 distance. Safety PLC arbitrates motion stops.	Couplings M Mechanical mass, force, geom. E Electrical power, 24 V, air I Informational signals, safety T Timing sync, sequencing Strength STRONG Joint design review is mandatory medium standard interface specification suffices self diagonal cell (same subsystem)
SPR Perirobotic	STRONG M I Payload → gripper choice Robot capacity limits EoAT mass; F/T data returns to the robot controller.	self	T Part handoff window EoAT grasp opens the exact moment a conveyor slot is available; sensor gates it.	E I Sensor power + network 24 V DC and pneumatic supply reach sensors; vision data flows on fieldbus.	
SATT Transport	STRONG T M Part presentation timing Conveyor speed must match robot cycle; alignment sets the robot approach pose.	T Handoff gating Conveyor indexer and EoAT clamp coordinate via a shared part-present token.	self	E Drive power VFDs for conveyor motors run from main distribution; AGVs share charging.	Key rule Every change in RI propagates through all three STRONG columns/rows → it must be re-verified in SPR, SATT and SC before sign-off.
SC Associated	STRONG E I Power, signals, safety PLC Full power train and every safety signal that stops the robot originate in SC.	E I Sensor infrastructure Network cabling, 24 V buses and pneumatic supply reach every sensor and fixture.	E Drive power + network Same electrical infrastr. delivers VFD power and AGV fleet communication.	self	How to read Row = subsystem that CHANGES. Column = subsystem that is AFFECTED. Matrix is symmetric.

Figure 2.6. Interaction matrix for the robotic, perirobotic, transport/transfer, and associated subsystems; orange cells indicate strong coupling.

WORKED EXAMPLE — Tracing a Design Change Through All Four Subsystems

An engineer proposes increasing the robot's maximum speed from 1.5 m/s to 2.0 m/s TCP to reduce cycle time by 2 s. Trace the impact through all four subsystems.

Robotic subsystem: Higher TCP speed requires verifying servo capacity and checking payload derating at speed. If derating exceeds the available margin, the speed increase is not feasible without EOAT mass reduction.

Perirobotic subsystem: Vision system trigger must fire at the new speed. At 2.0 m/s, a part crossing a 50-mm trigger window is visible for only 25 ms. If camera exposure exceeds this, images blur. Force/torque sensor sampling rate must also be verified.

Transport subsystem: If cycle time drops by 2 s, the infeed conveyor must deliver parts 2 s faster. Check accumulation zone adequacy. At higher robot speed, the pick-zone dwell time decreases; conveyor indexing precision becomes more critical.

Associated systems: ISO 13855:2024 uses positioning formulas such as $S = K \times T + C$, where K represents an assumed human approach speed and T is the total response time of the safeguarding and machine stopping chain. Increasing TCP speed does not directly redefine K , but it can increase robot stopping time or stopping distance and therefore increase T or the required safeguarded separation. The stopping performance must be measured or obtained from validated manufacturer data for the actual load and motion profile.

Conclusion: A nominal 2 s cycle-time improvement requires coordinated verification across all four subsystems. The review should document affected requirements, calculations, test evidence, residual risk, and investment. Engineering effort is application-dependent; assigning a universal number of hours would be misleading.

INDUSTRIAL PRACTICE NOTE — Managing Subsystem Interdependencies in Design Reviews

At each design review, require an INTERDEPENDENCY CHECK: for every proposed change, trace its impact through all four subsystems using the interaction matrix. Common failure patterns include: (1) increasing robot speed without revalidating stopping performance and safeguard positioning; (2) redesigning EOAT without updating payload, centre-of-gravity, inertia, and reach checks; and (3) adding a vision or data service without separating real-time control traffic from non-deterministic image or telemetry transfer. A concise change-impact checklist records whether each subsystem is affected, the mechanism, the verification method, and the responsible engineer.

Common Pitfalls

Pitfall 1: Selecting the robot first, the system second.

Engineers who begin by choosing a robot model and then design the system around it consistently produce suboptimal cells. Begin with the task analysis, derive system requirements, then select the robot.

Pitfall 2: Underestimating EOAT design effort.

EOAT design can consume a substantial share of application engineering because it combines mechanics, sensing, utilities, controls, wear, part tolerances, and failure recovery. Allocating only a token effort commonly produces unreliable gripping, release, or process performance.

Pitfall 3: Ignoring buffer sizing.

A line with no decoupling can propagate micro-stops directly to downstream resources. Kingman's approximation estimates queueing sensitivity; finite-buffer sizing still requires variability data, blocking/starvation logic, recovery policy, and simulation or empirical validation.

Pitfall 4: Confusing cycle time with takt time.

Cycle time describes resource performance; takt time expresses the output pace required by demand and available time. A cycle time well below takt may represent excess

capacity, resilience, or over-investment depending on utilisation, product mix, downtime, and strategic reserve. A bottleneck cycle above takt indicates a capacity shortfall under the analysed assumptions.

Pitfall 5: Neglecting energy flow in the design phase.

Energy significance depends strongly on the process. Pneumatic leaks, energised idle states, extraction, cooling, and oversized utilities can remain invisible in nominal cycle-time simulation but become evident in measured lifecycle cost and sustainability data.

Pitfall 6: Designing information flow as an afterthought.

Adding OPC UA, MQTT, historian, or MES connectivity after commissioning is often more difficult because interfaces, security zones, data models, and validation were not reserved in the original architecture. Specify data ownership, semantics, timing, and cybersecurity requirements during system design.

Pitfall 7: Over-sizing conveyors for safety margin.

Oversizing a conveyor for a much heavier hypothetical product can waste capital, energy, and floor space. Size the system from validated load cases and a documented growth allowance, including accumulation, impact, starts/stops, and maintenance conditions.

Pitfall 8: Treating the four subsystems as independent purchases.

Multi-vendor procurement is normal, but interface ownership must be explicit. Without a responsible system integrator and controlled interface specifications, local purchase savings can be offset by commissioning defects, incompatible assumptions, and ambiguous acceptance responsibility.

Pitfall 9: Using VA ratio as the only flow metric.

VA ratio (value-added time / total lead time) measures flow efficiency but not flow stability. A cell with 40% VA ratio and stable output is better than one with 55% VA ratio and erratic stoppages.

Pitfall 10: Ignoring subsystem interdependencies during changes.

A 'simple' robot speed increase that is not checked against safety distances, sensor response times, and conveyor speeds will generate integration defects at commissioning.

KEY TAKEAWAYS

- A robotic production system comprises four interacting subsystems: robotic, perirobotic, transport/transfer, and associated systems. Design them together, not sequentially.
- EOAT selection can dominate reliability. Verify EOAT + workpiece mass, centre of gravity, inertia, utilities, and dynamic grip requirements against the robot and process limits.
- Three flows must be analysed: material flow (parts), information flow (data/commands), and energy flow (power/pneumatics). Inefficiencies arise from poor flow design more often than from robot limitations.
- Takt time establishes the synchronisation heartbeat; the bottleneck station determines actual system throughput. Re-derive TT whenever demand changes.
- Little's Law ($WIP = TH \times LT$) links system-level WIP, throughput, and lead time. Bottleneck theory indicates where queues are likely to form, but actual buffer states must be measured.
- Kingman's approximation quantifies how queueing sensitivity increases with utilisation and variability; finite-buffer decisions require service-level targets and validation.
- Balance efficiency is a screening metric rather than a universal pass/fail threshold. Rebalancing is justified only when it improves the governing system objective.
- The information hierarchy separates device, real-time control, supervisory, and MES/ERP functions. Protocols must be selected for determinism, bandwidth, interoperability, safety, and cybersecurity.
- Energy audits can reveal practical improvement opportunities in compressed air, idle states, process utilities, and equipment sequencing; quantify savings from representative measurements.
- Subsystem interdependencies mean that every design change must be traced through all four subsystems using an interaction matrix or change-impact checklist.

Review Questions

Answer the following questions to consolidate your understanding of the chapter material. A thorough response should draw on concepts, formulas, and examples presented in the text.

1. Name the four principal subsystems of a robotic production system and give a one-sentence description of each. Which subsystem is most often underestimated in project budgets?
2. A palletising cell handles 25 kg boxes with a vacuum EOAT weighing 4 kg. The robot is rated for 35 kg payload at the required reach. What is the static payload margin? What additional information about centre of gravity, inertia, acceleration, vacuum reserve, and failure consequence is required before declaring the design acceptable?
3. Define takt time and cycle time. A shift is 450 min with customer demand of 600 units. What is the takt time? If the bottleneck station has a cycle time of 50 s, can the demand be met?
4. Apply Little's Law to a 4-station line with CTs of 22, 28, 35, 25 s and observed WIP of 20 parts. Compute throughput, lead time, and VA ratio. Where does WIP accumulate?
5. Describe the hierarchy of communication protocols in a modern robotic cell. What type of information flows at each level, and what are the latency requirements?
6. Compare the annual energy cost of a pneumatic vacuum system ($0.8 \text{ Nm}^3/\text{min}$ at $\text{€}0.03/\text{Nm}^3$) and an electric vacuum pump (0.6 kW at $\text{€}0.15/\text{kWh}$) for a cell operating 16 h/day, 250 days/year. Calculate both continuous-duty and 35% duty-cycle scenarios and discuss which assumptions control the payback conclusion.
7. A production line has balance efficiency of 0.68. The manager asks whether adding a sixth station would improve throughput. Under what conditions would adding a station help, and when would it be counterproductive?
8. Identify three examples of subsystem interdependencies where a change in the transport subsystem would propagate to the robotic subsystem. For each, describe the mechanism and the verification needed.
9. A cell currently operates at 85% of design throughput. Using the four-subsystem framework, outline a systematic diagnostic procedure to identify which subsystem is responsible for the 15% shortfall.

References

- [1] Groover, M.P. (2019) Automation, Production Systems, and Computer-Integrated Manufacturing, 5th ed. Pearson. ISBN 978-0-13-460546-3.
- [2] Siciliano, B., Sciavicco, L., Villani, L., and Oriolo, G. (2009) Robotics: Modelling, Planning and Control. Springer. ISBN 978-1-84628-641-4.
- [3] Nof, S.Y. (ed.) (1999) Handbook of Industrial Robotics, 2nd ed. John Wiley & Sons. ISBN 978-0-471-17783-8.
- [4] Little, J.D.C. (1961) A proof for the queuing formula: $L = \lambda W$. Operations Research, 9(3), pp. 383–387.
- [5] Kingman, J.F.C. (1961) The single server queue in heavy traffic. Mathematical Proceedings of the Cambridge Philosophical Society, 57(4), pp. 902–904.
- [6] Hopp, W.J. and Spearman, M.L. (2011) Factory Physics, 3rd ed. Waveland Press. ISBN 978-1-57766-739-1.
- [7] ISO 8373:2021, Robotics — Vocabulary. International Organization for Standardization, Geneva.
- [8] ISO 13855:2024, Safety of machinery — Positioning of safeguards with respect to the approach of the human body. ISO, Geneva.
- [9] ISO 13849-1:2023, Safety of machinery — Safety-related parts of control systems — Part 1: General principles for design. ISO, Geneva.
- [10] IEC 62061:2021, Safety of machinery — Functional safety of safety-related control systems, including Amendment 1:2024. International Electrotechnical Commission, Geneva.
- [11] IEC 62443 series, Security for industrial automation and control systems. International Electrotechnical Commission, Geneva.
- [12] Mocan, B. et al. (2018) Multidisciplinary Design of Industrial Robotic Automation Solutions — Practical Guide for Students. UTPress. ISBN 978-606-737-246-5.
- [13] Mocan, B. (2026) Design and Operation of Robotic Production Systems — CESPR Course Support. Technical University of Cluj-Napoca.
- [14] EtherCAT Technology Group (2026) EtherCAT Technology Overview: short cycle times and distributed-clock synchronisation.
- [15] OPC Foundation (2026) OPC UA: interoperability and secure industrial information exchange.

CHAPTER

3

Process Layout Design for Robotic Production Systems

From floor plan to strategic engineering decision

Layout design translates a system concept into a physical implementation. The choices made at this stage have lifetime consequences: a poor layout can materially reduce throughput, complicate maintenance, create safety hazards, and limit the system's ability to adapt to product changes. Conversely, a good layout becomes largely invisible — the system simply works.

A common engineering mistake is to treat layout design as a final drawing step, performed after all other design decisions are frozen. In reality, layout design must iterate with component selection, flow analysis, and safety engineering. The best practice is to develop candidate layouts early, evaluate them with simulation, and refine until the trade-offs are explicit and acceptable.

LEARNING OBJECTIVES

1. Identify and compare four fundamental layout typologies (linear, U-shaped, cellular, hybrid) and their suitability for different production contexts.
2. Quantify the impact of layout decisions on cycle time, throughput, flexibility, and operational cost using the material handling cost coefficient (MHC).
3. Apply Systematic Layout Planning (SLP) methodology including activity relationship charts and space relationship diagrams.
4. Apply CRAFT and genetic algorithm approaches for computational layout optimisation and compare their strengths.
5. Allocate space for safety zones (ISO 10218-2), maintenance corridors, operator workstations, and logistics pathways.
6. Calculate cell footprint requirements from robot reach envelopes, conveyor dimensions, and safety clearances.
7. Evaluate layout alternatives using weighted multi-criteria scoring with sensitivity analysis.
8. Design layout expansion strategies that accommodate future capacity growth without rebuilding the cell.
9. Integrate layout design with the material flow analysis from Chapter 2 and the safety engineering of Chapter 8.

10. Recognise the iterative nature of layout design and its coupling with component selection, control architecture, and simulation validation.

3.1 Introduction

Layout design is the activity of determining where each component of the robotic production system will be physically located in the factory. It is one of the earliest design decisions, yet its consequences persist throughout the system's multi-year operational life. The layout determines how far parts must travel between stations, how easily operators can access the cell for intervention, where safeguards and access points must be placed, and whether the cell can be expanded or reconfigured when production requirements change.

Three constraints drive every layout decision: performance (cycle time, throughput, utilisation), safety (risk reduction, access and escape routes, and definition of maximum, restricted, and safeguarded spaces in accordance with ISO 10218-2:2025 and applicable regulations), and flexibility (ability to adapt to future product or volume changes). The engineer's task is to find the layout that best balances these constraints within the available floor space and budget.

3.2 Layout Typologies

3.2.1 Linear Layout

In a linear layout, workstations are arranged in a straight line with material flowing from one end to the other. This is the classic production-line layout, optimised for high-volume, low-variety production. Linear layouts are typical in automotive BIW welding lines, bottling lines, and electronics assembly. Advantages include a clear flow direction, straightforward control, and high throughput potential. Reconfiguration can be difficult; without adequate buffers, bypasses, or redundancy, a station failure can propagate through the line, and operator access may be distributed over a long distance.

3.2.2 U-Shaped Layout

The U-shaped layout curves the production flow back on itself so that input and output are at the same end. This allows one operator to monitor multiple stations, enables material replenishment at a single logistics point, and shortens internal communication

distances. U-shaped layouts are widely used in lean manufacturing cells. In robotics, they work well for machine-tending cells where one robot loads and unloads several CNC machines arranged in a U.

3.2.3 Cellular Layout

A cellular layout groups dissimilar machines into cells, each designed to process a family of similar parts. Cellular layouts can provide high flexibility and are particularly suited to medium-volume, medium-variety production. They can reduce setup time when parts in a family share similar setups, improve multi-skill operation, and shorten lead times by reducing inter-cell transport.

3.2.4 Hybrid Layouts

Real-world production systems rarely use a single pure layout type. Hybrid layouts combine elements of linear, U-shaped, and cellular typologies — for example, a main linear backbone with U-shaped sub-cells for complex operations. Hybrid design requires careful analysis of material flow, operator movement, and reconfigurability, but can provide an effective overall compromise.

The choice between typologies is fundamentally a volume–variety decision, although the boundaries overlap and depend on process technology, automation level, product geometry, and changeover strategy. Table 3.1 provides indicative planning ranges rather than fixed selection thresholds.

Table 3.1. Indicative volume–variety positioning of layout typologies.

Layout	Annual Volume	Product Variants	Typical Industries
Linear	> 100,000 units	1–3	Automotive BIW, bottling, cigarettes
U-shaped	10,000–100,000	3–10	CNC machine tending, electronic assembly
Cellular	1,000–50,000	10–100	Job shops, medical devices, aerospace
Hybrid	5,000–200,000	5–50	Appliance assembly, food packaging

Figure 3.1 presents an illustrative strategic mapping of manufacturing system configurations as a function of product variety and annual production volume, both

represented on logarithmic scales. Four broad regions — linear, U-shaped, cellular, and hybrid — show typical rather than exclusive operating domains. Linear systems occupy the high-volume, low-variety region, while cellular systems tend toward lower volumes and higher variety. U-shaped layouts cover many medium-volume applications, and hybrid systems bridge overlapping domains when a single typology cannot satisfy all constraints.

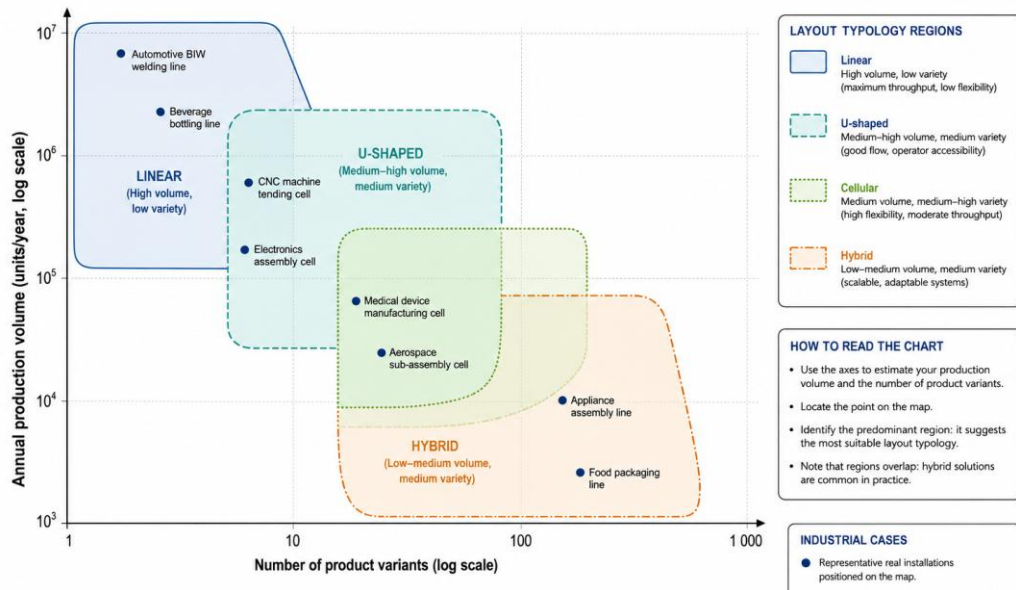


Figure 3.1. Volume–variety map and indicative layout-typology regions. Representative industrial cases are positioned for illustration; the regions overlap and do not constitute strict decision boundaries. Source: adapted from Muther (1994) and Groover (2019).

The superimposed points are illustrative industrial cases rather than a statistically validated dataset. They show how bottling and SMT applications often align with linear configurations, whereas aerospace and medical-device applications may favour cellular or hybrid arrangements because of higher variety and lower volumes. The figure highlights the trade-off between throughput efficiency and flexibility and provides an initial decision-support map that must be complemented by process, safety, and economic analysis.

INDUSTRIAL PRACTICE NOTE — Illustrative Layout Mistakes and Their Cost

Three illustrative cases based on common industrial situations: (1) A palletising line was designed with only 0.6 m between the robot fence and the building wall. Maintenance of the J1 motor required fence removal and crane access, creating an illustrative cost of €3,200 per event. (2) A CNC-tending cell used a linear arrangement for three machines and one robot. The operator station was 8 m from the infeed; repeated jam recovery created avoidable walking time. (3) An assembly line allowed a logistics route to cross an AGV path without an engineered traffic-control strategy, creating repeated near-miss potential. The numerical values are illustrative; the engineering lesson is to validate maintenance access, operator travel, and traffic interfaces before the layout is frozen.

Table 3.2. Indicative comparison of layout typologies for robotic production systems.

Criterion	Linear	U-Shaped	Cellular	Hybrid
Volume suitability	High	Medium–High	Medium	Medium–High
Variety suitability	Low	Medium	High	High
Flexibility	Low	Medium	High	High
Operator productivity	Low	High	High	High
Fault tolerance	Low–Medium	Medium	Medium–High	Medium–High
Initial investment	High	Medium	Medium	Medium–High
Reconfiguration ease	Low	Medium	High	Medium

Figure 3.2 compares four fundamental production layouts: Linear, U-Shaped, Cellular, and Hybrid. Each configuration offers distinct trade-offs between throughput, space efficiency, and flexibility. While linear and U-shaped lines focus on high-volume flow and compact footprints, cellular layouts excel in handling diverse product families. The hybrid model represents a sophisticated evolution, combining a main sequential "backbone" with specialized sub-cells to balance efficiency and complexity. Selecting the optimal layout requires a careful analysis of the production volume and the variety of the product mix.

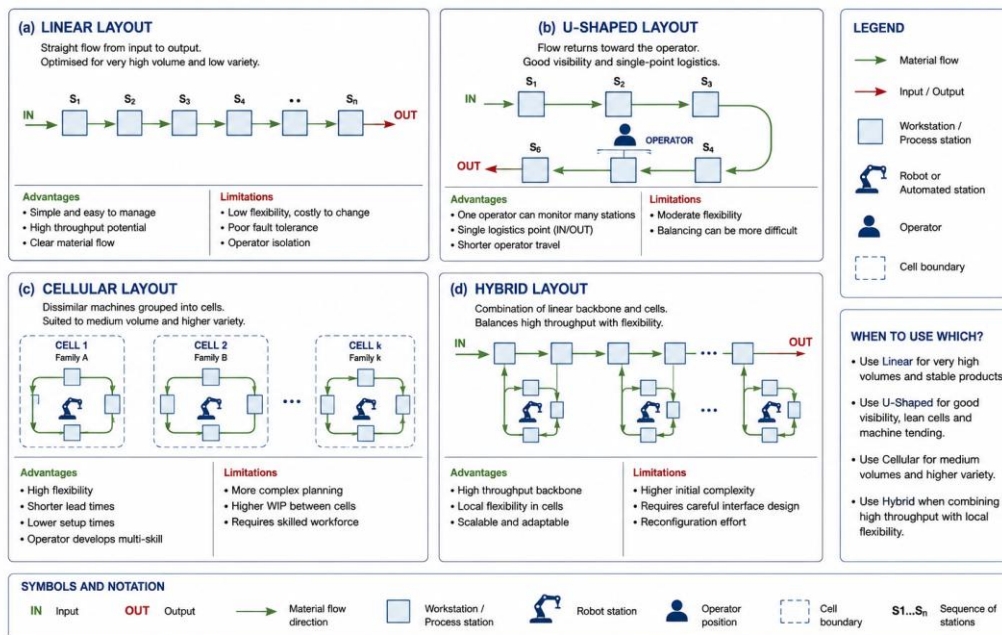


Figure 3.2. Fundamental layout typologies for robotic production systems: (a) linear, (b) U-shaped, (c) cellular, and (d) hybrid.

WORKED EXAMPLE — Selecting a Layout Typology for a New Production Line

A manufacturer of electric motor housings needs a new robotic production system. Annual volume: 80,000 units. Product variants: 4 (different sizes, same process family). Processes: machining (2 CNC), deburring (1 robot), washing (1 station), inspection (1 vision), assembly (1 robot). Available floor area: 12 × 18 m. Select the layout typology.

Volume assessment: $80,000 \text{ units/year} \div 250 \text{ days} \div 16 \text{ h} = 20 \text{ units/h}$. $TT = 3,600/20 = 180 \text{ s}$. This is medium volume — not high enough to justify a dedicated linear line per variant.

Variety assessment: 4 variants require changeover capability. Linear layout would need full-line changeover; cellular handles variant switching at the cell level.

Process flow: 6 process steps in sequence suggest a flow-based layout. But the 2 CNC machines share a robot for loading — a natural U-shape opportunity.

Decision: Hybrid layout — U-shaped machining sub-cell (2 CNC + 1 robot) feeding a linear flow (deburring → washing → inspection → assembly). This combines the operator-efficiency advantage of U-shape for machine tending with the clear flow direction of linear for sequential processing. Total footprint estimate: 10 × 15 m = 150 m² (fits within 216 m² available, leaving 30% for expansion).

INDUSTRIAL PRACTICE NOTE — A Three-Iteration Layout Process

A practical course project benefits from iterative layout development rather than a single frozen drawing. Iteration 1 (Concept, typically Weeks 1–2): block-level layout using 2D shapes to establish flow direction, cell grouping, and approximate footprint. Iteration 2 (Preliminary Design, typically Weeks 4–6): detailed 2D/3D layout using actual equipment dimensions, reach envelopes, preliminary safeguarded zones, and maintenance access. Iteration 3 (Final Design, typically Weeks 8–10): simulation-validated layout with verified cycle times, collision-free robot paths, and operator-task allocation. The week ranges are indicative for the CESPR course and should be adapted to project complexity. Each iteration should include a cross-functional review involving production, safety, maintenance, and logistics. Skipping intermediate validation increases the risk of layout-related commissioning rework.

3.3 Impact of Layout on System Performance

Layout decisions have measurable impact on multiple performance dimensions. Cycle time is affected through material-handling time between stations; throughput depends on bottleneck spacing and buffer placement; flexibility is constrained by the ability to add, remove, or relocate stations; ergonomic quality is shaped by operator walking distances; and safety is enforced by spatial relationships between human zones and robot workspaces.

3.3.1 The Material Handling Cost Coefficient

A useful metric for evaluating layout efficiency is the **material handling cost coefficient**:

$$\text{MHC} = \sum (f_{ij} \times d_{ij} \times c_{ij})$$

where f_{ij} is flow volume between stations i and j , d_{ij} is physical distance, and c_{ij} is handling cost per unit distance. Minimising MHC is the objective function of many layout optimisation algorithms. Compared with an arbitrary or ad-hoc arrangement, a structured optimisation can reduce MHC, but the achievable improvement is project-specific and depends on the initial layout, constraints, and flow matrix.

WORKED EXAMPLE — Calculating Material Handling Cost for Two Layout Alternatives

A 4-station cell has the following daily flow volumes (parts/day): $S1 \rightarrow S2 = 500$, $S2 \rightarrow S3 = 500$, $S3 \rightarrow S4 = 480$, $S1 \rightarrow S4 = 20$ (reject path). Handling cost: €0.01/part/metre. Compare Layout A (linear) vs. Layout B (U-shaped).

Layout A (linear): Distances: $S1-S2 = 4\text{m}$, $S2-S3 = 4\text{m}$, $S3-S4 = 4\text{m}$, $S1-S4 = 12\text{m}$.
 $MHC\_A = (500 \times 4 + 500 \times 4 + 480 \times 4 + 20 \times 12) \times 0.01 = (2000 + 2000 + 1920 + 240) \times 0.01 = €61.60/\text{day}$.

Layout B (U-shaped): $S1$ and $S4$ are adjacent. Distances: $S1-S2 = 4\text{m}$, $S2-S3 = 4\text{m}$, $S3-S4 = 4\text{m}$, $S1-S4 = 2\text{m}$. $MHC\_B = (500 \times 4 + 500 \times 4 + 480 \times 4 + 20 \times 2) \times 0.01 = (2000 + 2000 + 1920 + 40) \times 0.01 = €59.60/\text{day}$.

Savings: €500/year. The U-shaped alternative shortens the reject path from 12 m to 2 m while the main flow is unchanged. **Insight:** MHC can reveal non-obvious secondary-flow improvements.

3.4 Layout Optimisation Methods

3.4.1 Systematic Layout Planning (SLP)

Systematic Layout Planning, developed by Richard Muther and subsequently refined with Lee Hales, is presented in this course through an adapted six-step sequence: (1) gather input data on product, quantity, routing, supporting services, and time; (2) construct the activity relationship chart; (3) develop the activity relationship diagram; (4) determine space requirements; (5) construct the space relationship diagram; and (6) evaluate alternatives and select the preferred layout. Full SLP practice can contain additional working forms and iteration loops.

The activity relationship chart assigns a letter code to each pair of activities: A = absolutely necessary, E = especially important, I = important, O = ordinary, U =

unimportant, X = undesirable. The chart is filled by a cross-functional team, and its content drives all subsequent layout decisions.

3.4.2 CRAFT Algorithm

CRAFT (Computerized Relative Allocation of Facilities Technique) is an iterative improvement algorithm that starts with an initial layout and progressively swaps pairs of activities to reduce total MHC. CRAFT remains a useful baseline, but it converges to a local optimum and its result depends on the initial layout and the swaps permitted. Metaheuristic methods may find better solutions in some cases, but no fixed percentage improvement over CRAFT should be assumed.

3.4.3 Genetic Algorithms for Layout Optimisation

For complex layout problems with many constraints, genetic algorithms (GA) offer a flexible optimisation approach. Each candidate layout is encoded as a chromosome, and a population evolves through selection, crossover, and mutation. GA can incorporate multiple objectives such as cost, safety, ergonomics, and expandability, together with constraints such as fixed locations and safeguarded zones. Simulation platforms such as FlexSim and Visual Components can evaluate candidate layouts, but optimisation usually requires dedicated modules, scripting, or coupling to external algorithms. Python libraries such as DEAP and PyGAD can be used to implement the search logic.

An illustrative GA study for a six-station cell might use a population of 50–100 candidates and several hundred generations, but population size, stopping criterion, constraint handling, and computation time must be justified for the specific encoding and objective function. The result should be reported against the same constraints and cost model used for CRAFT; a fixed improvement percentage or universal run time cannot be assumed.

3.4.4 Activity Relationship Chart: A Practical Example

The activity relationship chart (ARC) is the cornerstone of SLP. For a robotic packaging cell with 6 activities — (A) Infeed conveyor, (B) Vision inspection, (C) Delta robot pick-place, (D) Carton forming, (E) Palletising, (F) Pallet wrapper — the cross-functional team assigns proximity ratings (table 3.3):

Table 3.3. Activity relationship chart for a packaging cell. Ratings: A = absolutely necessary, E = especially important, I = important, O = ordinary, U = unimportant, X = undesirable.

Pair	Rating	Reason
A–C (Infeed–Robot)	A (Absolutely necessary)	Robot picks directly from infeed
B–C (Vision–Robot)	A	Vision triggers robot pick
C–D (Robot–Carton)	E (Especially important)	Robot places into cartons
D–E (Carton–Palletise)	I (Important)	Filled cartons move to palletiser
E–F (Palletise–Wrapper)	A	Full pallet wraps immediately
A–F (Infeed–Wrapper)	X (Undesirable)	Incoming parts must not contact wrapped pallets
B–E (Vision–Palletise)	U (Unimportant)	No direct interaction

The A-ratings (infeed–robot, vision–robot, palletise–wrapper) constrain these pairs to be adjacent. The X-rating (infeed–wrapper) forces separation. These constraints immediately eliminate many layout candidates and guide the SLP process toward feasible solutions (figure 3.3).

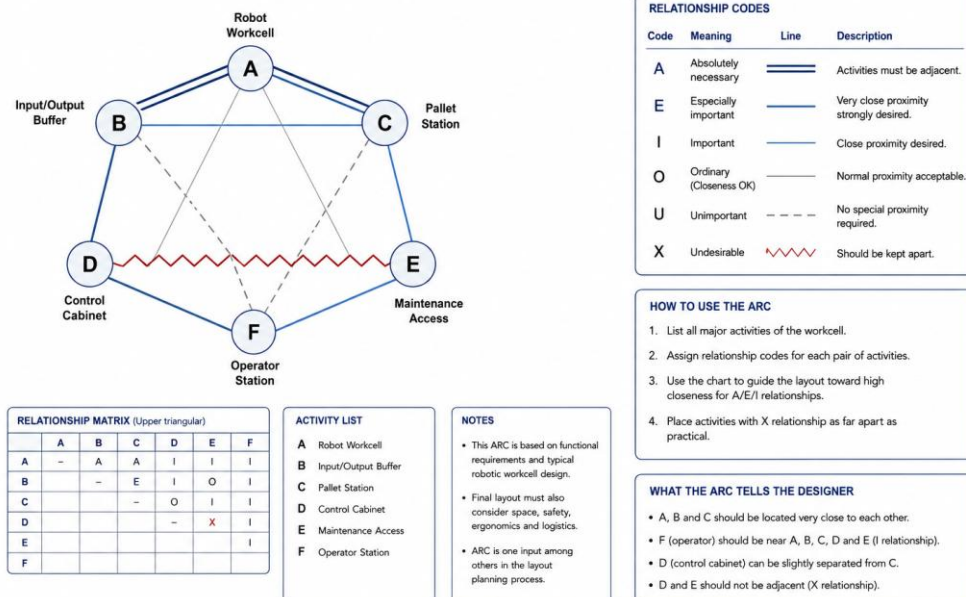


Figure 3.3. Activity Relationship Chart (ARC) for a robotic workcell, showing proximity codes, the upper-triangular relationship matrix, and the resulting layout implications. Source: adapted from Muther (1994).

Table 3.4. Comparison of layout optimisation methods.

Method	Input Required	Computation	Optimality	Best For
SLP (manual)	Flow data, team judgment	Hours–days (manual)	Feasible, judgement-based	Small cells, early design
CRAFT	From-to matrix, cost matrix	Usually fast for small/medium cases	Local optimum	Baseline comparison
Genetic Algorithm	Same + constraints	Problem- and parameter-dependent	Near-global candidate; no guarantee	Complex, multi-objective
Simulated Annealing	Same as GA	Problem- and parameter-dependent	Near-global candidate; no guarantee	Large search spaces

WORKED EXAMPLE — SLP Multi-Criteria Scoring for a Packaging Cell

Three layout candidates (Linear L, U-shaped U, Cellular C) are scored against 5 weighted criteria. Weights derived from stakeholder AHP.

Criteria and weights: Throughput 0.30, Flexibility 0.20, Ergonomics 0.20, Footprint 0.15, Maintenance access 0.15.

Scores (L, U, C): Throughput (5,4,3); Flexibility (2,4,5); Ergonomics (3,5,4); Footprint (3,5,2); Maintenance (4,3,5).

Results: L = 3.55, U = 4.20, C = 3.75. U-shaped wins for the stated weights. If the four non-throughput weights are reduced proportionally as the throughput weight increases, the Linear and U-shaped alternatives become equal at a throughput weight of approximately 0.58. The conclusion is therefore robust within a broad, but explicitly defined, sensitivity range.

3.5 Space Allocation for Safety and Logistics

A layout is not complete until safety zones and logistics pathways have been allocated. Six categories of space must be explicitly designed (Figure 3.4):

- Robot and safeguarding zones: define the robot maximum space, restricted space, safeguarded space, hazard zones, and access points through risk assessment and in accordance with ISO 10218-2:2025. The protective perimeter is not determined by robot reach alone.
- Operator workstations: allocate ergonomic working space, clear lines of sight, and logical material presentation. Preliminary distances of about 1.0–1.5 m may be used in concept layouts, but final dimensions depend on the task, anthropometry, accessibility needs, and applicable regulations.
- Maintenance corridors: provide access based on the largest maintenance task, the tools required, component-removal paths, and emergency egress. Values such as 0.8 m or 1.2 m are project assumptions, not universal ISO minima.
- Logistics pathways: size pedestrian and vehicle routes from traffic volume, vehicle envelope and turning radius, visibility, segregation strategy, and national requirements. Concept values such as 1.2–1.4 m pedestrian routes or a 3.0 m one-way forklift lane must be validated for the actual application.
- Emergency exits: maintain unobstructed escape routes and provide the number and width of exits required by national fire and building regulations.
- Utilities access: position compressed-air supplies, electrical disconnects, and network interfaces so that isolation, inspection, and maintenance can be performed safely.

INDUSTRIAL PRACTICE NOTE — Designing Safe Logistics Corridors

Do not apply a universal walkway width without context. Establish corridor dimensions from pedestrian flow, carried loads, maintenance tools, accessibility requirements, vehicle envelopes, visibility, and national rules. (1) DIRECTIONAL FLOW can reduce conflicts but does not automatically halve the required width. (2) T-INTERSECTIONS with obstructed visibility require engineered controls such as mirrors, warning devices, speed limits, or right-of-way rules. (3) FORKLIFT AND AGV ROUTES should be physically or functionally segregated where the risk assessment

requires; painted lines alone may be insufficient. Record the assumptions and validate them with the safety and logistics teams.

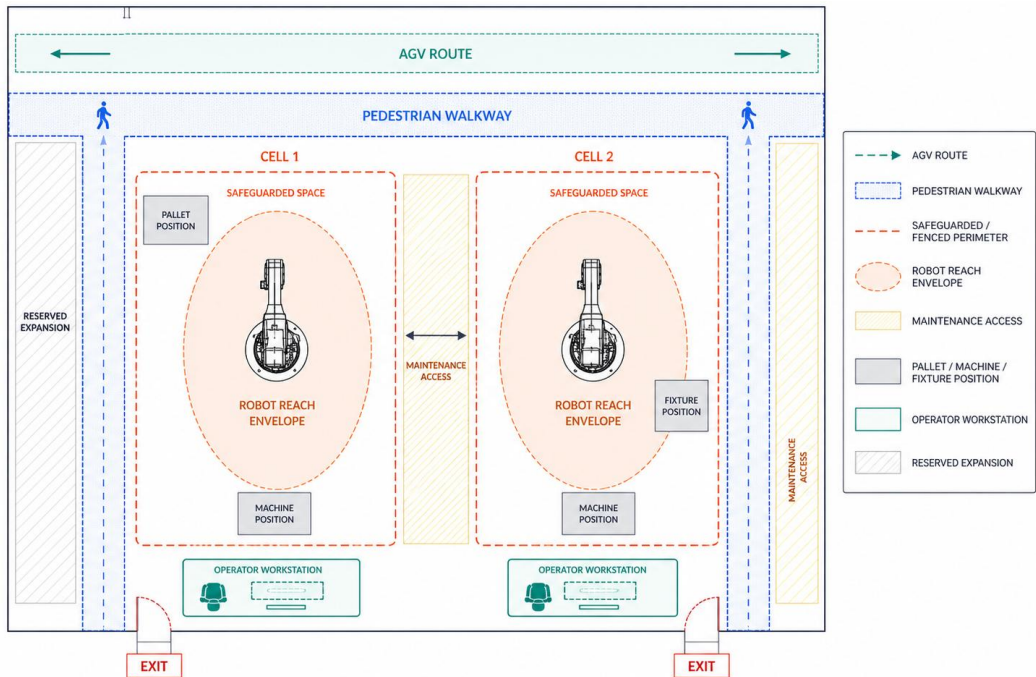


Figure 3.4. Engineering top view of two robotic cells, showing safeguarded perimeters, robot reach envelopes, maintenance access, operator workstations, pedestrian circulation, the AGV route, emergency exits, and reserved expansion space.

WORKED EXAMPLE — Robotic Palletising Cell Space Allocation

A 6-axis robot (2.0 m reach, pedestal-mounted) stacks 40×40×30 cm boxes onto 1.2×1.0 m pallets. Conveyor from east, pallet changer on west, operator on south.

Robot workspace: circle diameter 4.0 m = 12.57 m².

Preliminary safeguarded envelope: for concept estimation only, assume 0.3 m around the 4.0 m-diameter reach circle, giving a 4.6 m × 4.6 m square = 21.2 m². The final fence position must be determined from the restricted and maximum spaces, stopping performance, access prevention, and maintenance requirements.

Conveyor approach: $2\text{ m} \times 0.8\text{ m} = 1.6\text{ m}^2$. **Pallet changer:** assume a $2\text{ m} \times 2\text{ m}$ device plus a 1 m-wide access strip along one 2 m side: $4 + 2 = 6\text{ m}^2$.

Operator + maintenance + walkway: $4 + 4.6 + 6.4 = 15\text{ m}^2$.

Total: $21.2 + 1.6 + 6 + 15 = 43.8\text{ m}^2$ for the defined core areas. Adding $10\text{--}20\text{ m}^2$ for staging and vehicle manoeuvring gives an indicative total of approximately $54\text{--}64\text{ m}^2$. **Interpretation:** this example corresponds to about 4.3–5.1 times the robot reach-circle area. Such ratios are project-specific and must not replace component-by-component space allocation.

3.6 Designing for Future Expansion

A layout designed only for current requirements can become a constraint as demand grows or products change (Figure 3.5). Three expansion strategies should therefore be considered during initial layout design:

1. Reserved footprint. Allocate an application-specific area adjacent to the cell for future stations, conveyors, or robots. A reserve of 20–30% may be used as a preliminary scenario when growth is plausible, but the final allowance should follow the capacity roadmap and site constraints. Mark the area on the layout drawing as ‘reserved for expansion’ and protect it from unplanned use.
2. Modular infrastructure. Provide justified electrical, pneumatic, network, and control-system headroom for the credible expanded configuration. The additional cost of spare capacity at initial installation is often lower than replacing infrastructure during an expansion, but the values must be estimated for the actual project.
3. Expansion-compatible layout topology. U-shaped and cellular layouts often permit additions at an open end or as new cells, whereas linear layouts usually require extension in one direction. The preferred topology still depends on the building, material-flow sequence, utilities, safeguarding concept, and shutdown strategy.

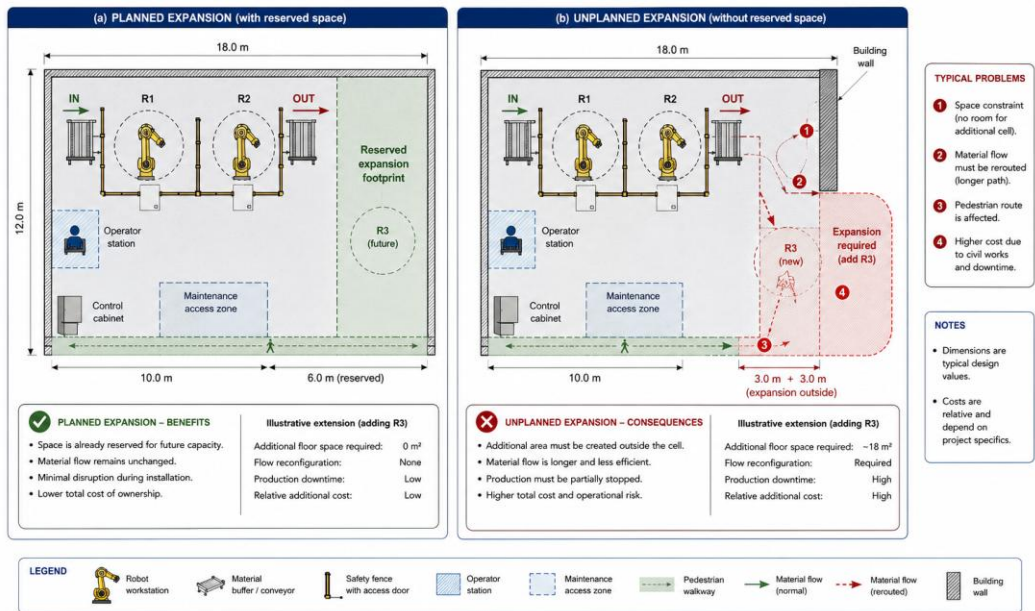


Figure 3.5. Planned versus unplanned expansion: comparison of a reserved-space scenario with a constrained extension, illustrating the effects on material flow, pedestrian access, downtime, and relative project cost.

WORKED EXAMPLE — Cost of Expansion: Planned vs. Unplanned

Consider an illustrative scenario in which a palletising cell was installed without expansion planning and demand later increased by 40%, requiring a second robot. Compare the following planned and unplanned expansion assumptions.

Illustrative unplanned expansion: No reserved footprint → relocate adjacent storage (€8,000). Electrical panel at capacity → replace panel (€6,500). No spare I/O on PLC → add expansion rack (€3,200). Safety fence must be reconfigured → new fence sections and re-validation (€12,000). Production stop during installation: 3 weeks × €4,000/day = €60,000. Total unplanned: €89,700.

Illustrative planned expansion: Reserved footprint included in the original layout: €0 marginal relocation cost. Additional panel capacity at original installation: +€800. Spare I/O planned: +€400. Fence designed with a removable expansion section: +€2,000. Installation during a planned shutdown: 1 week × €4,000/day = €20,000. Total planned: €23,200.

Result for this scenario: €89,700 – €23,200 = €66,500 avoided cost (74%). The €3,200 expansion-readiness allowance corresponds to an avoided cost of about 20.8 times the initial allowance. **Lesson:** evaluate credible expansion scenarios during concept design, while recognising that actual costs and savings are project-specific.

INDUSTRIAL PRACTICE NOTE — Layout Design Review Checklist

Before freezing a layout for manufacturing, verify: (1) REACH AND DEXTERITY — check all task, approach, and retreat points against the robot workspace and payload data, TCP orientation, singularities, cable dress, and tool mass. A 90% reach value may be used as an early screening assumption, but it is not a universal acceptance limit. (2) COLLISION AND CLEARANCE — simulate the robot with EOAT, workpieces, cable dress, and moving peripherals; define a project-specific allowance for modelling uncertainty, structural and cable deflection, part tolerances, calibration error, and dynamic behaviour rather than applying a universal 15 mm margin. (3) SAFEGUARD POSITIONING — verify access prevention and safeguard distances using ISO 10218-2:2025, ISO 13855:2024 where applicable, measured stopping time, and the risk assessment; re-verify after changes to speed, payload, tooling, or control. (4) MAINTENANCE ACCESS — confirm that each service task can be completed with the necessary tools and component-removal path, ideally through a maintenance-team walkthrough. (5) LOGISTICS FLOW — separate or control material-delivery, vehicle, and pedestrian paths according to the traffic risk assessment. (6) EXPANSION READINESS — identify reserved space and verify infrastructure headroom for credible future scenarios. (7) UTILITIES ROUTING — prevent cables, hoses, and pneumatic lines from creating floor-level trip or vehicle hazards. Items 1–3 require simulation and engineering verification; Items 4–7 require a 3D review and, where possible, a physical walkthrough with operations personnel.

Figure 3.6 provides a practical application of floor-space planning for advanced automation. This axonometric view of a palletising cell translates conceptual layout

design into detailed zonification. By defining dedicated areas for maintenance access, operator workstations, logistics pathways, and reserved expansion, the layout supports operational safety, material-flow efficiency, and long-term adaptability. This is a useful next step after selecting the overall production-system topology.



Figure 3.6. Axonometric verification view of the robotic workcell and logistics infrastructure. The dimensions shown are illustrative and must be confirmed through detailed engineering and application-specific risk assessment.

Common Pitfalls

Pitfall 1: Treating layout as a final drawing step.

Layout design must iterate with component selection and simulation. Freezing the layout after all other decisions are made produces a constrained, suboptimal result.

Pitfall 2: Optimising for current demand only.

A layout designed only for today's demand can become a constraint when products or volumes change. Define a project-specific expansion allowance; 20–30% is an illustrative planning range, not a universal requirement.

Pitfall 3: Ignoring maintenance access.

A cell with only 0.4 m clearance behind equipment may be impossible to maintain without a shutdown and partial dismantling. Required access width must be derived from the maintenance task, tools, component-removal path, ergonomics, and emergency egress.

Pitfall 4: Using a successful layout as a template without analysis.

A layout that works in Plant A may fail in Plant B if floor dimensions, utilities locations, or operator workflows differ. Always derive from constraints.

Pitfall 5: Placing safety fences as close as possible to the robot.

Tight fences can reduce footprint but may eliminate maintenance access, future expansion, and safe intervention. There is no universal 0.3 m clearance rule; the final perimeter must follow the restricted-space design, stopping performance, access-prevention measures, and risk assessment.

Pitfall 6: Designing logistics paths as afterthoughts.

Material delivery routes that cross operator walkways or require forklift entry into pedestrian zones create chronic safety hazards and are expensive to fix post-installation.

Pitfall 7: Choosing layout typology based on preference rather than analysis.

A U-shaped layout can be highly effective in lean and machine-tending applications, but may be unsuitable for a very high-speed, high-volume, single-product process where a linear flow offers simpler transport and control.

Pitfall 8: Ignoring the third dimension.

2D layout drawings miss overhead clearances (crane travel, cable trays, lighting), below-floor utilities (drainage, cable trenches), and mezzanine structures. Always verify in 3D.

Pitfall 9: Skipping the activity relationship chart in SLP.

Teams that jump directly to space allocation without establishing proximity requirements produce layouts that look efficient but have poor flow logic.

Pitfall 10: Not involving operators in layout review.

Engineers optimise for material flow; operators know where they need to stand, what they need to see, and how they need to move. Both perspectives are essential.

KEY TAKEAWAYS

- Layout design is a strategic engineering decision made early in the design process, and its consequences persist throughout the system's multi-year operational life.
- Four fundamental typologies — linear, U-shaped, cellular, and hybrid — serve overlapping volume–variety domains; selection must follow the actual process and constraints.
- Layout directly affects cycle time, flexibility, safety, maintainability, and operational cost; the magnitude of the effect is project-specific.
- The material handling cost coefficient (MHC) provides a quantitative basis for comparing layout alternatives.
- SLP provides structured manual layout optimisation; CRAFT and genetic algorithms provide computational alternatives for complex problems.
- Safety zones, maintenance corridors, and logistics pathways must be explicitly designed and dimensioned, not treated as leftover space.
- Cell footprint must be calculated from robot spaces, safeguards, equipment, maintenance, operator, and logistics requirements; simple reach-area multipliers are only preliminary screening tools.
- A concept → preliminary → final iteration sequence is a practical workflow, but the number and maturity of iterations should match project complexity.
- Expansion-readiness can avoid substantial future cost; the benefit must be quantified through project-specific scenarios rather than universal percentages.
- Layout design must iterate with component selection, control architecture, and simulation — it is never a standalone activity.

Review Questions

Answer the following questions to consolidate your understanding of the chapter material. A thorough response should draw on concepts, formulas, and examples presented in the text.

1. Describe the four fundamental layout typologies. For each, give one industrial application where it is typically preferred and explain why.

2. Why is layout design considered a strategic decision rather than a final drawing step? Give two examples of how layout impacts long-term system performance.
3. A company produces 12 product variants at 5,000 units/year each. Which layout typology is most appropriate? Justify with quantitative analysis.
4. Calculate MHC for a 4-station cell with the following daily flows: $A \rightarrow B=800$, $B \rightarrow C=780$, $C \rightarrow D=760$, $A \rightarrow D=40$ (rejects). Compare linear (4m spacing) vs. U-shaped (A and D adjacent at 2m).
5. Explain the SLP methodology. What is the purpose of the activity relationship chart, and what happens if it is skipped?
6. Develop a preliminary floor-space estimate for a robotic cell containing a 2.5 m reach robot, a 3×1.5 m conveyor, a 2×1 m operator station, a maintenance-access zone, and a pedestrian route. State all assumptions, distinguish robot reach from safeguarded space, and explain how the final dimensions would be validated.
7. A cell was designed without expansion planning. Three years later, a second robot must be added. Estimate the cost penalty of unplanned vs. planned expansion and identify the three largest cost drivers.
8. What is the material handling cost coefficient and how can it reveal non-obvious layout improvements?
9. List six space allocations that must be included in a robotic-system layout beyond the robot workspace. For each, identify the design basis, the project-specific information required, and the applicable standard, regulation, or company rule.

References

- [1] Muther, R. and Hales, L. (2015) Systematic Layout Planning, 4th ed. Management & Industrial Research Publications. ISBN 978-0-933684-06-5.
- [2] Groover, M.P. (2018) Automation, Production Systems, and Computer-Integrated Manufacturing, 5th ed. Pearson. ISBN 978-0-13-460546-3.
- [3] Tompkins, J.A. et al. (2010) Facilities Planning, 4th ed. Wiley. ISBN 978-0-470-44404-7.
- [4] ISO 10218-2:2025, Robotics — Safety requirements — Part 2: Industrial robot applications and robot cells. ISO, Geneva.
- [5] ISO 13855:2024, Safety of machinery — Positioning of safeguards with respect to the approach of the human body. ISO, Geneva.
- [6] Siciliano, B., Sciavicco, L., Villani, L., and Oriolo, G. (2009) Robotics: Modelling, Planning and Control. Springer London. ISBN 978-1-84628-641-4.
- [7] Mocan, B. et al. (2018) Multidisciplinary Design of Industrial Robotic Automation Solutions. UTPress. ISBN 978-606-737-246-5.

- [8]** Mocan, B. (2026) Design and Operation of Robotic Production Systems — CESPR Course Notes. Technical University of Cluj-Napoca.
- [9]** Armour, G.C. and Buffa, E.S. (1963) A heuristic algorithm and simulation approach to relative location of facilities. *Management Science*, 9(2), pp. 294–309.
- [10]** Gen, M. and Cheng, R. (2000) *Genetic Algorithms and Engineering Optimization*. Wiley. ISBN 978-0-471-31531-5.
- [11]** ISO 12100:2010, Safety of machinery — General principles for design — Risk assessment and risk reduction. ISO, Geneva.
- [12]** Nof, S.Y. (ed.) (1999) *Handbook of Industrial Robotics*, 2nd ed. John Wiley & Sons. ISBN 978-0-471-17783-8.
- [13]** Hopp, W.J. and Spearman, M.L. (2011) *Factory Physics*, 3rd ed. Waveland Press. ISBN 978-1-57766-739-1.
- [14]** International Federation of Robotics (2025) *World Robotics 2025 — Industrial Robots*. VDMA Services GmbH, Frankfurt.
- [15]** Xu, X., Lu, Y., Vogel-Heuser, B., and Wang, L. (2021) Industry 4.0 and Industry 5.0 — Inception, conception and perception. *Journal of Manufacturing Systems*, 61, pp. 530–535.

CHAPTER

4

Methodology for Designing Robotic Cells

From requirements to validated design in six structured steps

This chapter introduces a structured, engineering-oriented methodology for designing robotic cells. The process begins with the analysis of production requirements and continues with task decomposition, operation sequencing, and the definition of the working cycle. Basic concepts related to cycle time calculation, task balancing, and productivity estimation are presented to illustrate how design decisions directly influence system performance.

The iterative nature of robotic cell design is emphasised, highlighting the need to continuously validate assumptions and refine solutions. Students are encouraged to view design as a **decision-making process under technical and operational constraints**, rather than as a purely creative exercise.

LEARNING OBJECTIVES

1. Apply the six-step methodology to design a robotic cell from requirements through validated design.
2. Conduct a requirements analysis that captures functional, performance, safety, and economic specifications.
3. Decompose a production task into basic robotic operations (approach, grasp, move, place, release) and estimate motion times.
4. Analyse operation sequencing, identify the critical path, and distinguish between sequential and parallel operations.
5. Calculate cycle time components (t_p , t_h , t_m , t_w) and compute value-added fraction as a design quality indicator.
6. Balance tasks across stations using the Ranked Positional Weight method and distinguish balance efficiency from capacity margin.
7. Apply three levels of design validation: analytical check, simulation verification, and formal design review.
8. Construct a Requirements Traceability Matrix (RTM) linking specifications to design elements and verification tests.
9. Estimate throughput, productivity, and capacity margin from cycle-time analysis and takt-time comparison.

10. Manage cycle-time estimation uncertainty across three design iterations using the indicative CESPR planning bands (concept $\pm 30\%$, detailed $\pm 15\%$, post-simulation $\pm 5\%$).

4.1 Introduction

The design of a robotic cell is not an unstructured act of invention; it is a controlled engineering synthesis process. Given a set of production requirements — what must be produced, at what rate, to what quality, and within what budget — the engineer's task is to develop a cell configuration that meets those requirements reliably, safely, and economically. The six-step methodology presented in this chapter provides a repeatable framework for that synthesis.

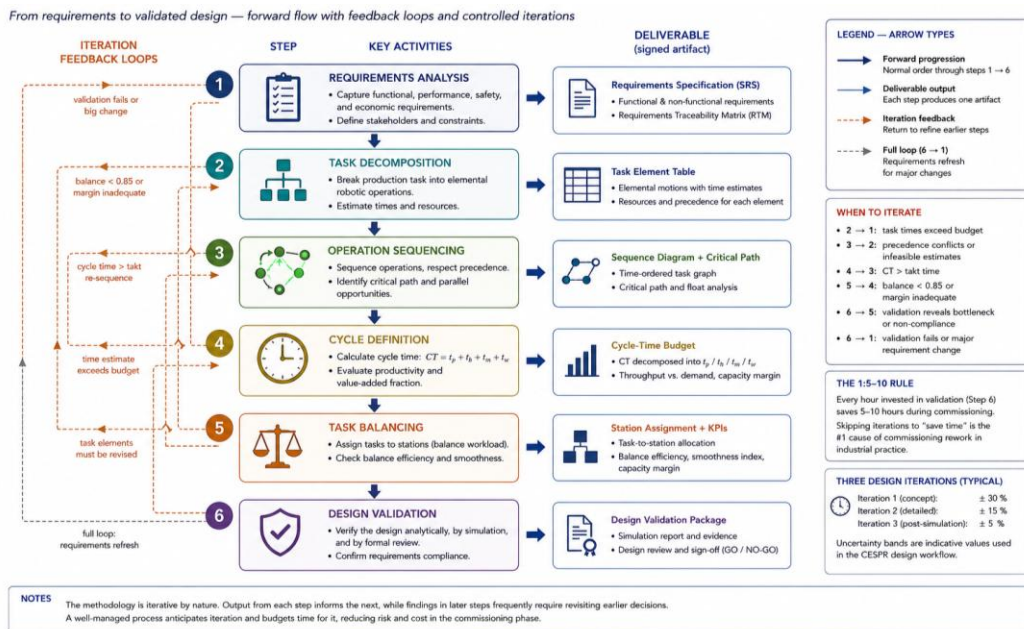


Figure 4.1. Six-step robotic-cell design methodology: Requirements Analysis → Task Decomposition → Operation Sequencing → Cycle Definition → Task Balancing → Design Validation. Selected feedback loops emphasise that later findings can require earlier design decisions to be revisited.

The methodology is iterative (Figure 4.1): each step produces outputs that inform subsequent steps, and findings from later stages may require earlier decisions to be reconsidered. A well-managed design process plans for controlled iteration and

configuration updates; a poorly managed process treats preliminary decisions as irreversible and transfers unresolved problems to commissioning.

4.2 The Six-Step Design Methodology

The following methodology establishes a structured framework for the systematic design and optimisation of robotic production systems. Each stage aligns engineering activities with technical, operational, safety, and economic objectives. Table 4.1 summarises the principal activities, deliverables, and indicative planning durations; actual schedules depend on application complexity, supplier response, validation depth, and the maturity of the input requirements.

Table 4.1. *The six-step design methodology with deliverables and typical durations.*

Step	Activity	Key Deliverable	Indicative Duration
1	Requirements Analysis	Requirements Specification	1–2 weeks
2	Task Decomposition	Task list with time estimates	1 week
3	Operation Sequencing	Sequence diagram + critical path	3–5 days
4	Cycle Definition	Cycle time breakdown (t_p , t_h , t_m , t_w)	3–5 days
5	Task Balancing	Station assignment + balance efficiency	2–3 days
6	Design Validation	Validated design package	1–2 weeks

As a planning reference, a conventional single-cell design may require approximately 6–8 weeks from kickoff to a validated design package, while multi-cell lines commonly require longer programmes. These are indicative ranges rather than contractual norms. Schedule risk is strongly influenced by requirements maturity, interface complexity, supplier data availability, and the speed of stakeholder decisions.

INDUSTRIAL PRACTICE NOTE — Requirements Traceability Matrix (RTM)

An RTM links each baselined requirement to its source, rationale, allocated design element, verification method, test-case identifier, acceptance owner, status, and change reference. It answers two essential questions: ‘did the delivered system satisfy every agreed requirement?’ and ‘what design and verification evidence is affected when a requirement changes?’. Keep the RTM under configuration control in an appropriate requirements-management tool or controlled spreadsheet, and review it at every project milestone.

4.3 Step 1: Requirements Analysis

Requirements analysis is the highest-leverage activity in the design process. Getting requirements wrong — or incomplete — cascades through every subsequent step. The requirements specification should cover five categories:

1. **Functional requirements:** What operations must the cell perform? What products? What variants? What quality specifications?
2. **Performance requirements:** Throughput (units/hour), cycle time, takt time, availability target, OEE target.
3. **Safety requirements:** Applicable standards (including ISO 12100 and ISO 10218-2:2025), intended operating modes, identified safety functions, and the required performance level (PLr) for each relevant safety function.
4. **Physical constraints:** Available floor space, ceiling height, utility access, environmental conditions (temperature, dust, humidity).
5. **Economic constraints:** Budget ceiling, target payback period, operating cost targets.

A common failure is to specify performance requirements without specifying the conditions under which they must be met. ‘Throughput: 100 parts/hour’ is incomplete; ‘throughput: 100 parts/hour at steady state with product variant A, assuming 90% availability’ is testable.

Requirements gathering typically involves three sources: (1) the customer specifies what the system must produce and the business constraints; (2) the operations and maintenance teams specify how the system will be operated, serviced, recovered, and changed over; and (3) applicable standards and legal requirements define mandatory

safety and compliance constraints (Table 4.2). A cross-functional workshop is often an effective starting point, but the specification normally requires follow-up interviews, document review, and formal clarification of unresolved assumptions.

Table 4.2. Requirements categories with sources, examples, and verification methods.

Category	Source	Example Requirement	Verification Method
Functional	Customer	Handle 3 product variants	Run all variants during FAT
Performance	Customer + Operations	CT ≤ 30 s at 90% availability	8-hour endurance test
Safety	Standards + Safety engineer	PLr d for the specified safety function; architecture justified by risk assessment	Safety-function validation report
Physical	Facilities + Operations	Footprint ≤ 5×6 m	Layout drawing verification
Economic	Customer + Finance	Payback < 3 years	Business case review at M+12
Maintainability	Maintenance team	Required service access and removal paths defined	3D review and physical walkthrough

To ensure project success, the engineering team must translate vague operational goals into concrete, measurable metrics that leave no room for ambiguity during the validation phase. Table 4.3 contrasts high-quality, actionable specifications with common but inadequate descriptors, highlighting the precision required for professional industrial documentation.

Table 4.3. Good vs. poor requirement specifications.

Requirement Category	Good Specification	Poor Specification
Throughput	100 parts/h at 90% availability, variant A	100 parts/h
Quality	Defect rate < 500 PPM, vision-verified	High quality

Requirement Category	Good Specification	Poor Specification
Safety	PLR d for the defined safety function, with architecture justified by risk assessment	Safe operation
Budget	€250k total, €8k/year maintenance	Cost-effective
Flexibility	3 variants; validated changeover time < 15 min under stated conditions	Flexible for future

WORKED EXAMPLE — Writing a Requirements Specification for a Machine-Tending Cell

A manufacturer needs a robotic cell to tend two CNC lathes producing automotive shaft components. Daily demand: 400 conforming parts across two 450-minute shifts. Three part variants (lengths 150, 200, and 250 mm; all Ø40 mm steel). Write the principal requirements.

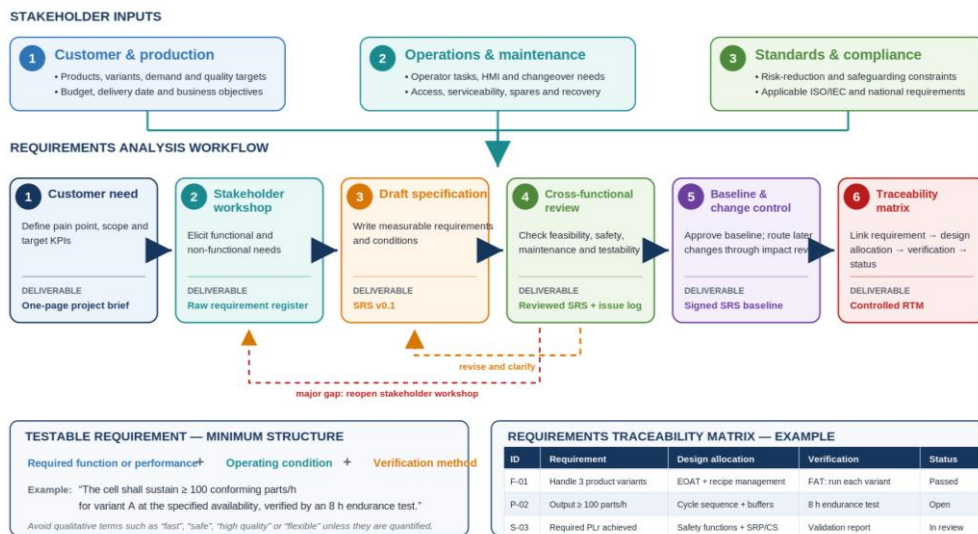
Functional: Load raw blanks from the infeed conveyor into either CNC chuck, unload finished parts to the outfeed conveyor, handle all three variants through a controlled recipe change, and verify part presence and chuck-clamping status before machining starts.

Performance: Customer-demand interval $TT = (2 \times 450 \times 60)/400 = 135$ s per conforming part, corresponding to 26.7 parts/h. The cell shall demonstrate a sustained output of at least 26.7 conforming parts/h over an eight-hour endurance test using the specified product mix. At 90% cell availability, the nominal design output must be at least 29.6 parts/h, equivalent to an average nominal interdeparture time of no more than 121.5 s. If an OEE of 80% is imposed as a contractual system-level requirement, the corresponding nominal interdeparture target is no more than 108 s. The longest CNC machining time is 90 s and the robot unload/load/verification service time shall be no more than 40 s per machine. Because the two lathes operate in parallel, feasibility must be confirmed by a resource-constrained timing analysis rather than by simply adding machining and robot-service times.

Safety: Use a safeguarded cell because the CNC chucks and robot create crushing and entanglement hazards. Access gates shall be interlocked and, where the stopping-time assessment requires it, provided with guard locking. Safety functions shall achieve the PLr determined by risk assessment; a Category 3 architecture may be used where justified. Reset controls shall be located outside the hazardous space, and lockout/tagout provisions shall cover the robot, CNC machines, and associated energy sources.

Physical: Available floor area: 5 × 6 m. Ceiling height: 4 m. Three-phase 400 V/32 A supply and 6 bar compressed air are available. Ambient operating range: 15–35 °C. Service access and component-removal paths must be validated in the 3D layout.

Economic: Budget ceiling: €180,000. Target simple payback: less than three years, subject to a business case that includes labour, quality, availability, tooling, maintenance, and ramp-up assumptions. The current manual-tending cost is estimated at €72,000/year for one operator per shift across two shifts.



Requirements are baselined, not permanently frozen: any later change requires formal impact assessment and RTM update.

Figure 4.2. Requirements-analysis workflow: Customer need → Stakeholder interviews → Draft specification → Cross-functional review → Baselined requirements → RTM. After baselining, changes remain possible only through formal change control and impact assessment.

4.4 Step 2: Task Decomposition

Task decomposition breaks the production task into elemental robot motions and operations. Each element has an estimated time, a required resource (robot, sensor,

gripper), and a precedence relationship to other elements. The decomposition serves three purposes: it reveals the structure of the cycle, it enables time estimation, and it identifies opportunities for parallelism.

Standard robot motion elements include: **approach** (move to pre-grasp position), **grasp** (close gripper/activate vacuum), **retract** (lift from fixture), **transfer** (move to target position), **place** (lower into fixture/conveyor), and **release** (open gripper/deactivate vacuum). Each element's time depends on distance, speed profile (trapezoidal or S-curve), payload, and accuracy requirement at the endpoint.

WORKED EXAMPLE — Task Decomposition for a Packaging Cell

Task: pick a product from a conveyor, verify via vision, place in a box, and wrap. Decompose into elements and estimate cycle time.

T1 — Pick from conveyor: approach (0.4 s), close gripper (0.1 s), lift (0.3 s). Total 0.8 s.

T2 — Move to inspection: 0.5 s (motion).

T3 — Vision inspection: 0.5 s (image capture + inference), release 0.1 s. Total 0.6 s.

T4 — Pick after inspection: 0.3 s.

T5 — Move to box and place: 0.6 s motion + 0.1 s release = 0.7 s.

T6 — Wrap (parallel): 4 s per batch of 12 products = 0.33 s/product average capacity. It is off the cycle-limiting path only if an intermediate buffer decouples the wrapper and the wrapper remains capable of processing the accumulated batches without blocking the upstream robot.

Steady-state cycle time: $T1+T2+T3+T4+T5 = 0.8+0.5+0.6+0.3+0.7 = 2.9$ s.

Theoretical robot-limited throughput = $3,600/2.9 \approx 1,241$ products/h. Against a demand of 900 products/h, theoretical capacity headroom is approximately 38%. First-piece lead time remains longer because it includes batching and wrapping.

4.5 Step 3: Operation Sequencing

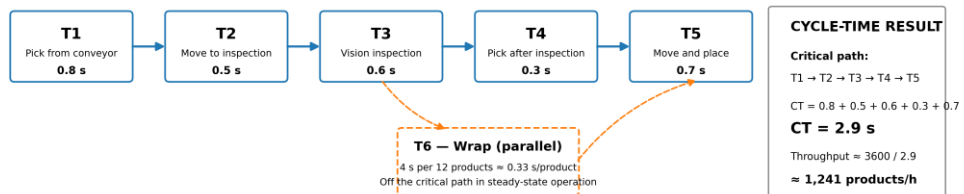
Operation sequencing arranges the decomposed tasks into a time-ordered sequence, respecting precedence constraints. The **critical path** is the longest chain of dependent

operations; it determines the minimum possible cycle time. Operations not on the critical path have **float** — they can be delayed without affecting the cycle.

Identifying the critical path is essential because: (1) reducing a task on the active critical path can reduce cycle time; (2) non-critical operations may have scheduling float that can support load balancing; and (3) adding resources to an off-path task will not improve throughput unless that task becomes cycle-limiting or the change improves reliability, quality, or recoverability.

PRECEDENCE GRAPH AND CRITICAL PATH

Tasks T1–T5 form the cycle-limiting sequential chain; T6 is executed in parallel.



TIMING DIAGRAM

Sequential tasks determine the cycle period; T6 overlaps the critical-path window.

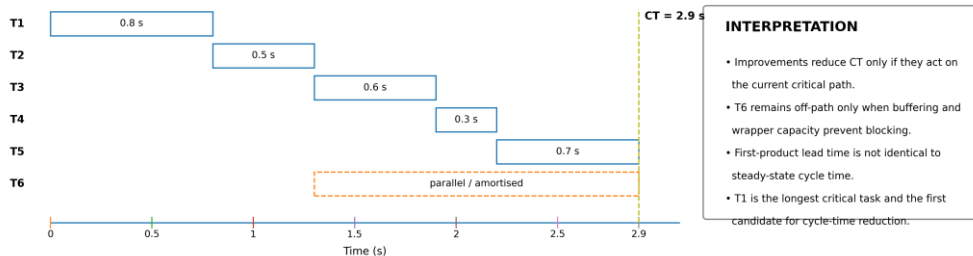


Figure 4.3. Operation-sequence diagram for the packaging cell. Tasks T1–T5 form the steady-state critical path with $CT = 2.9$ s. Task T6 is shown as a buffered parallel branch; it remains off-path only while its batch-processing capacity and buffer prevent blocking.

4.6 Step 4: Cycle Definition and Cycle Time Calculation

Cycle time (CT) is the interval between successive repetitions of the same reference event in a stable production cycle, such as two consecutive conforming-part completions. For a purely sequential cycle-limiting path, it may be decomposed into four components:

$CT = t_p + t_h + t_m + t_w$ (for non-overlapping components on the cycle-limiting path)

- **t_p (process time):** value-added time — the robot performs the actual task (welding, placing, inspecting).

- **t_h (handling time):** loading/unloading, part orientation, gripper operations.
- **t_m (machine auxiliary time):** non-value-adding equipment operations such as torch cleaning, tool changing, or fixture indexing.
- **t_w (waiting time):** idle time due to synchronisation, buffer starvation, or upstream delays.

When operations overlap, cell cycle time is determined from the resource-constrained timing diagram or cyclic schedule rather than by summing all activity durations. The value-added fraction ($VA = t_p/CT$) can support comparison between alternatives, but its interpretation is application-dependent. Values such as 40–60% may be used as indicative screening ranges in selected applications, not as universal acceptance criteria.

INDUSTRIAL PRACTICE NOTE — Cycle-Time Estimation Across Design Iterations

The CESPR workflow uses three indicative uncertainty bands for planning and communication: concept design (approximately $\pm 30\%$, based on preliminary motion and process assumptions), detailed design (approximately $\pm 15\%$, using supplier data and a developed 3D model), and simulation-supported design (approximately $\pm 5\%$ where model fidelity, controller settings, EOAT behaviour, and process data justify that confidence). These are not guaranteed accuracy levels. State both the estimate and its basis; reporting '68 s $\pm$ 20 s at concept stage' is more defensible than presenting an unsupported single value.

WORKED EXAMPLE — Welding Cell Cycle-Time Calculation

Arc-welding cell: one 6-axis robot, fixed fixture, and a 400 mm linear weld at 1.0 m/min.

t_p (weld): 400 mm / 1,000 mm/min = 24 s.

t_h (load/unload): 20 s (manual).

t_m (torch cleaning): a 12 s cleaning operation occurs every third cycle, giving an average contribution of $12/3 = 4$ s/cycle.

Average CT: 24 + 20 + 4 = 48 s. $VA = 24/48 = 50\%$. Theoretical throughput = $3,600/48 = 75$ parts/h.

Improvement: Automated loading reduces t_h to 5 s, giving an average CT of 33 s and a theoretical throughput of approximately 109 parts/h, subject to fixture, process, safety, and availability constraints.

Effective cycle time management requires a granular understanding of how production time is allocated across value-added and non-value-added activities. **Table 4.4** decomposes the total cycle time into its primary constituents, identifying typical distribution ranges and providing targeted engineering strategies to minimize these intervals for maximum throughput.

Table 4.4. Cycle time components with typical ranges and reduction strategies.

Component	Symbol	Indicative Range	Reduction Strategy
Process (value-added)	t_p	20–60% of CT	Faster process parameters, better toolpath
Handling	t_h	15–40% of CT	Automated loading, dual-gripper, quick-change
Machine auxiliary	t_m	5–15% of CT	Overlap where feasible; optimise service intervals and tool-change strategy
Waiting	t_w	0–25% of CT	Better synchronisation, buffer optimisation

WORKED EXAMPLE — Cycle-Time Improvement Scenarios for a Palletising Cell

A palletising cell has CT = 12 s ($t_p = 4$ s, $t_h = 5$ s, $t_m = 1$ s, $t_w = 2$ s). TT = 10 s. The cell cannot meet takt. Evaluate three improvement options.

Option A — Motion optimisation: Reduce t_p from 4 s to 3 s through validated trajectory, acceleration, and blending changes. New CT = 11 s, still above TT. Hardware cost may be negligible, but dynamic limits, payload, safeguarding, and process stability must be revalidated.

Option B — Dual gripper: Pick two boxes per robot cycle. Under the illustrative assumption that t_h increases to 6 s, t_p remains 4 s for the pair, t_m remains 1 s, and waiting is eliminated, effective CT per box = $(4+6+1+0)/2 = 5.5$ s. This option is feasible only if robot payload/inertia, box geometry, pallet pattern, and EOAT reliability support two-box handling. Illustrative gripper cost: €3,500.

Option C — Eliminate waiting: Buffer optimisation removes $t_w = 2$ s. New CT = 10 s = TT. The cell meets nominal demand but has zero capacity margin, so availability and variability must be assessed. Illustrative cost: €1,500.

Recommendation: Option B provides the largest nominal margin if the application supports dual handling. Options A and C combined give CT = $3+5+1+0 = 9$ s, not 8 s, and may offer a lower-risk alternative depending on engineering effort and process constraints. **Key principle:** evaluate the complete cycle, resource constraints, and feasibility assumptions rather than optimising one time component in isolation.

4.7 Step 5: Task Balancing

For multi-station lines, task balancing distributes work across stations to reduce avoidable idle time while satisfying precedence and cycle-time constraints. Balance efficiency, $BE = \sum CT_i / (n \times CT_{max})$, measures the uniformity of station loading. A value such as 0.85 is often used as an engineering screening threshold, but the acceptable value is application- and cost-dependent. Balance efficiency must not be confused with capacity margin: a perfectly balanced line may still retain margin below takt time, while an uneven line may nevertheless meet demand.

Three additional metrics quantify productivity:

- Throughput rate (TR): $TR = 1/CT_{bottleneck}$ in consistent time units; for CT in seconds, nominal throughput in parts/h is $3,600/CT_{bottleneck}$. Sustained output must also account for availability, quality losses, and blocking/starvation.
- Capacity margin: $CM = (TT - CT_{bottleneck})/TT \times 100\%$. Positive nominal margin indicates that the bottleneck cycle is below takt; values such as 5–15% are common planning allowances, but the required margin depends on variability, buffers, availability, and commercial risk.

- RMS station-load deviation (SI_RMS): $SI\_RMS = \sqrt{[\sum(CT\_max - CT\_i)^2/n]}$. Lower values indicate a smoother load distribution; SI_RMS = 0 represents equal station loads. No universal time-based acceptance threshold applies.

For a multi-station robotic line, the distribution of work should be checked against precedence, station capacity, takt time, and resource constraints. Table 4.5 summarises useful screening metrics. Their target values must be selected for the specific application rather than applied as universal pass/fail limits.

Table 4.5. Productivity metrics for task balancing.

Metric	Formula	Screening Interpretation	Typical Design Response
Balance efficiency	$\sum CT\_i / (n \times CT\_max)$	Higher is better; ≥ 0.85 often used for preliminary screening	Reassign tasks; review station count or parallel resources
Capacity margin	$(TT - CT\_max) / TT$	Positive; planning allowance often 5–15%	Reduce CT_max, add capacity, or revisit operating strategy
RMS station-load deviation	$\sqrt{[\sum(CT\_max - CT\_i)^2/n]}$	Minimise and compare alternatives; no universal threshold	Redistribute feasible tasks between stations
VA fraction	t_p / CT	Application-dependent	Reduce handling, auxiliary, or waiting time where justified

WORKED EXAMPLE — Line Balancing Using the Ranked Positional Weight Method

Eight tasks: A=12 s, B=8 s, C=15 s, D=10 s, E=14 s, F=6 s, G=11 s, H=9 s. Precedence: A→B, A→C, B→D, C→D, D→E, D→F, E→G, F→G, G→H. TT = 20 s.

Minimum stations: $[W/TT] = [85/20] = 5$.

Ranked positional weights: A=85, C=65, B=58, D=50, E=34, F=26, G=20, H=9. At each station, assign the highest-ranked eligible task that fits the remaining station time while respecting all precedence constraints.

Feasible assignment: S1: A(12)+B(8)=20 s; S2: C(15)=15 s; S3: D(10)+F(6)=16 s; S4: E(14)=14 s; S5: G(11)+H(9)=20 s. This assignment respects C→D and all other precedence relationships.

Evaluation: $BE = 85 / (5 \times 20) = 0.85$. $CT_{max} = 20 \text{ s} = TT$, so nominal capacity margin is 0%. $SI_{RMS} = \sqrt{[(0^2 + 5^2 + 4^2 + 6^2 + 0^2) / 5]} \approx 3.92 \text{ s}$. The solution is feasible and meets the screening efficiency value, but it is tight against takt and should be evaluated for variability, availability, and alternative assignments. The Ranked Positional Weight method is a heuristic and does not guarantee a global optimum.

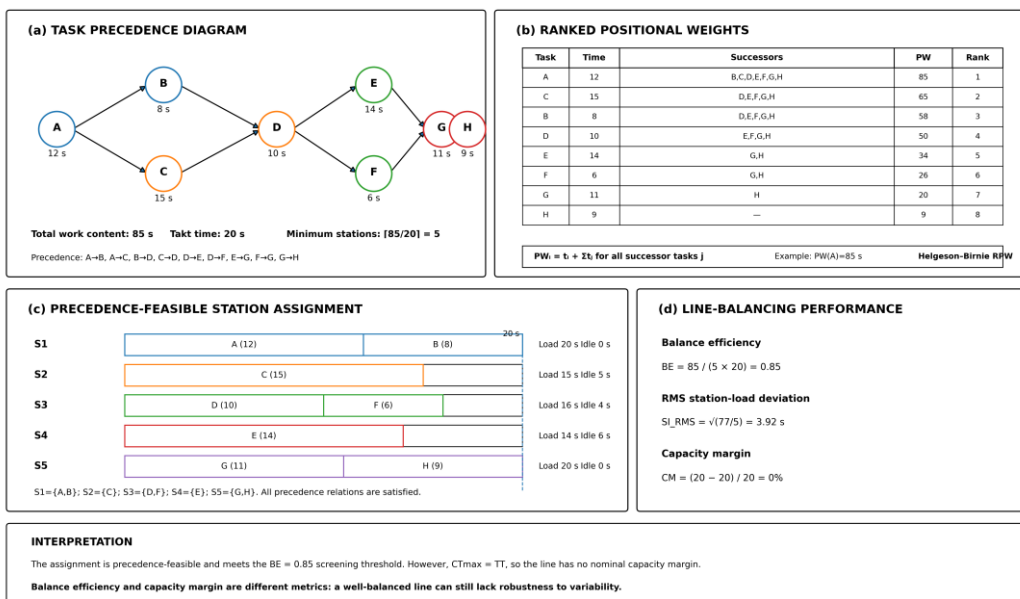


Figure 4.4. Ranked Positional Weight line-balancing example. The precedence-feasible assignment is $S1=\{A,B\}$, $S2=\{C\}$, $S3=\{D,F\}$, $S4=\{E\}$, and $S5=\{G,H\}$. The resulting balance efficiency is 0.85, the RMS station-load deviation is 3.92 s, and the nominal capacity margin is 0%.

4.8 Step 6: Design Validation

Design validation verifies that the proposed cell will meet its requirements before physical implementation. Three levels of validation are standard practice:

1. **Analytical validation:** hand calculations of cycle time, throughput, and capacity margin. Verifies that the numbers are internally consistent. Takes hours.
2. **Simulation-supported validation:** OLP and process-simulation tools can assess reachability, modelled interference, sequencing, and cycle time. Residual uncertainty depends on model fidelity, controller settings, payload and inertia data, EOAT behaviour, PLC/communication delays, process variability, and calibration quality. Takes days for a conventional cell.
3. **Formal design review:** cross-functional review of the design package by engineering, safety, maintenance, production, and customer representatives. It addresses organisational, operational, maintainability, and acceptance issues that simulation alone cannot resolve. A conventional single-cell review often takes several hours.

Validation is an essential risk-reduction activity. Analytical checks may require hours, simulation studies may require days, and commissioning rework can require substantially more time and cost. The frequently cited '1:5–10' relationship is best treated as a cost-of-change heuristic: defects are generally cheaper to correct before fabrication and integration, but the actual ratio is project-specific.

INDUSTRIAL PRACTICE NOTE — Structuring an Effective Design Review

A formal design review for a robotic cell should cover seven topics in sequence: (1) REQUIREMENTS RECAP — has the spec changed since kickoff? (2) LAYOUT WALKTHROUGH — 3D model review with maintenance and production. (3) CYCLE-TIME AUDIT — simulation vs. analytical CT, with variance explained. (4) SAFETY REVIEW — preliminary risk assessment findings, safety-device placement. (5) CONTROL ARCHITECTURE — I/O list, handshake diagram, safety-PLC logic. (6) OPEN ISSUES — logged with owner, priority, and resolution date. (7) GO/NO-GO DECISION — proceed to build, revise, or return to concept. Duration: 2–4 hours for a single cell. Mandatory attendees: lead engineer, safety engineer, maintenance supervisor, production manager, customer representative. Minutes distributed within 24 hours; open issues tracked weekly until closure.

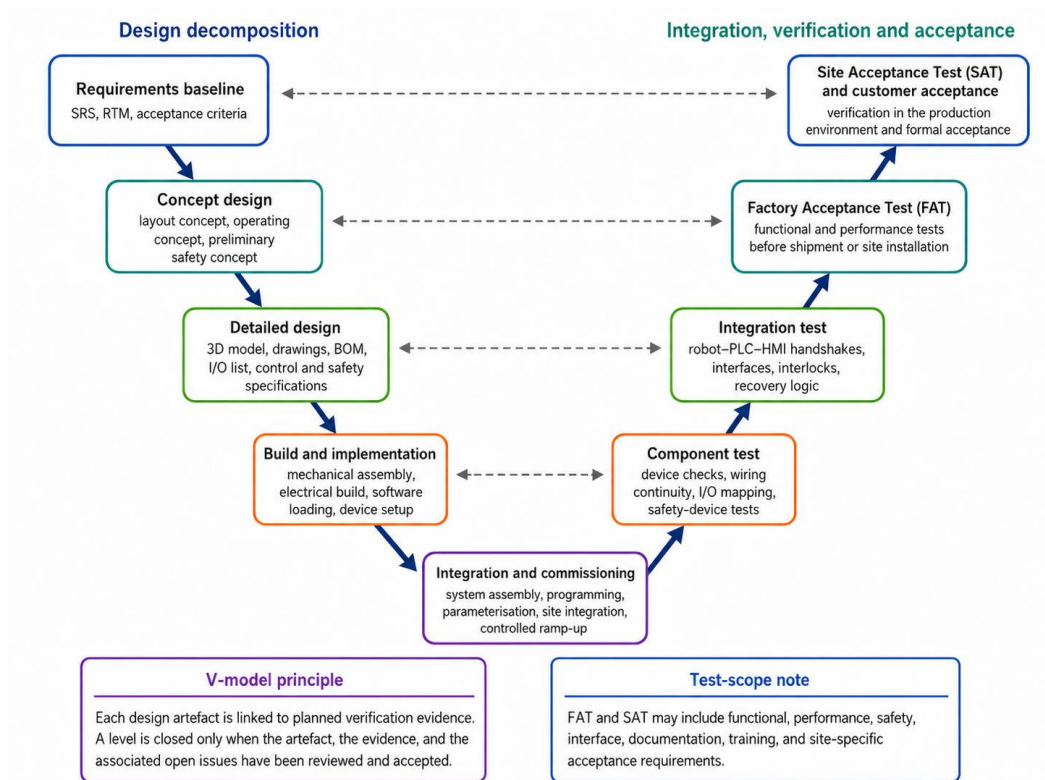


Figure 4.5. Adapted V-model for robotic-cell engineering. The left branch develops requirements and design detail toward implementation; the right branch rises through component testing, integration testing, FAT, SAT, and formal acceptance. Horizontal links indicate traceability between design artefacts and planned verification evidence.

4.9 The Iterative Nature of Cell Design

The methodology is presented linearly for clarity, but in practice it is deeply iterative. Three common loops:

- 1. Requirements–Feasibility:** Step 1 requirements may prove infeasible at Step 4 (cycle time too short). Return to Step 1 to negotiate or to Step 3 to redesign the sequence.
- 2. Cycle–Balance:** Step 4 cycle time may produce poor balance at Step 5. Return to Step 2 to re-decompose or Step 4 to reassign operations.
- 3. Validation–Design:** Step 6 simulation reveals collisions or CT discrepancies. Return to Step 3 or 5 to resolve.

Plan for multiple controlled passes through the six steps. In the CESPR workflow, Pass 1 develops the concept using an indicative $\pm 30\%$ uncertainty band; Pass 2 refines the

design using supplier data and models (approximately $\pm 15\%$); and Pass 3 produces a simulation-supported, baselined design (approximately $\pm 5\%$ where justified by model fidelity). These bands are planning aids, not guaranteed accuracies. A design is released for manufacturing only after its remaining assumptions, open issues, and verification evidence are explicitly accepted.

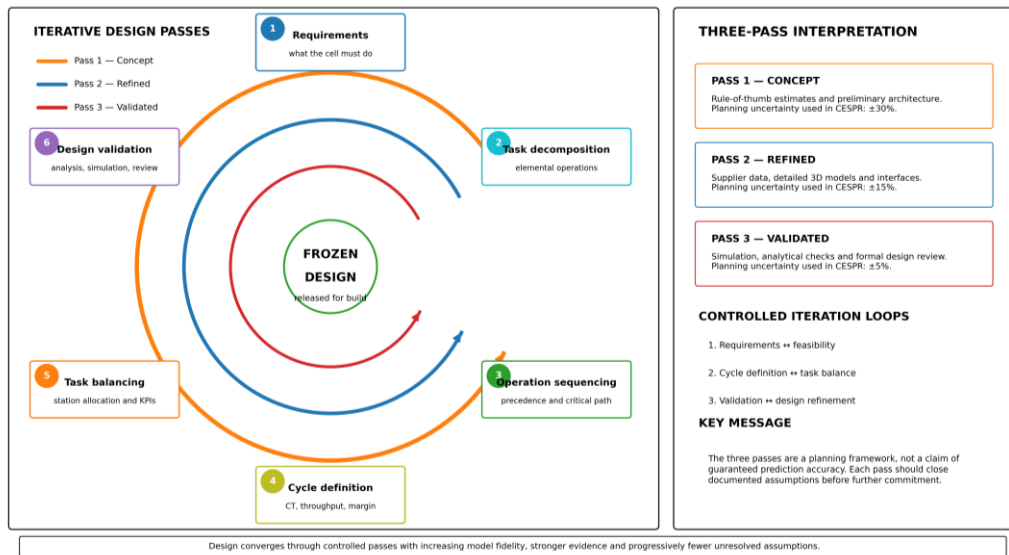


Figure 4.6. Three-pass spiral used in the CESPR workflow. Each pass revisits the six design steps with increasing model fidelity and stronger evidence. The indicated $\pm 30\%$, $\pm 15\%$, and $\pm 5\%$ bands are illustrative planning ranges rather than guaranteed prediction errors.

Common Pitfalls

Pitfall 1: Starting with robot selection instead of requirements.

The six-step methodology begins with requirements, not catalogues. Teams that select a robot first and then design around it consistently produce suboptimal cells.

Pitfall 2: Incomplete requirements specification.

Incomplete or ambiguous requirements are a major source of design change and acceptance disputes. Every requirement must be testable: ‘high quality’ is not testable, whereas ‘defect rate < 500 PPM under the defined product mix and inspection method’ is.

Pitfall 3: Estimating cycle time from a single iteration.

Concept-stage cycle-time estimates can carry substantial uncertainty. In the CESPR workflow, $\pm 30\%$ is used as an indicative planning band; presenting a preliminary estimate as an exact commitment misleads stakeholders.

Pitfall 4: Ignoring handling time in cycle-time analysis.

t_h (loading, unloading, orientation) often exceeds t_p (process time). A cell that optimises welding speed while neglecting loading time misses the dominant cycle-time component.

Pitfall 5: Confusing balance efficiency with capacity margin.

Balance efficiency describes how evenly work is distributed; it does not measure the margin between the bottleneck cycle time and takt time. A line may be perfectly balanced and still have positive capacity margin. Variability protection must be evaluated separately through capacity margin, buffers, availability, and blocking/starvation behaviour.

Pitfall 6: Treating validation as a formality.

A design review that rubber-stamps the design without genuine critique is worse than no review — it creates false confidence. Effective reviews require preparation, diverse perspectives, and the authority to reject.

Pitfall 7: Skipping the RTM.

Without a Requirements Traceability Matrix, there is no systematic mechanism for linking every baselined requirement to design allocation, verification evidence, and acceptance status. Omissions and uncontrolled changes are therefore harder to detect.

Pitfall 8: Optimising one CT component while ignoring others.

Reducing t_p by 2 s through a faster weld speed that increases t_m by 3 s (more frequent torch cleaning) is a net loss. Always evaluate the full CT equation.

Pitfall 9: Assuming the critical path is obvious.

In complex cells with parallel operations, the critical path may shift depending on product variant or operating conditions. Always compute it; never assume.

Pitfall 10: Not budgeting for design iteration.

The methodology is iterative by design. Projects planned as a single irreversible pass tend to underestimate the effort required for clarification, feasibility feedback, simulation findings, and controlled change.

KEY TAKEAWAYS

- The six-step methodology (Requirements → Decomposition → Sequencing → Cycle → Balancing → Validation) provides a structured framework. Feedback loops between steps are essential.
- Requirements analysis is the highest-leverage activity; requirements must be measurable, baselined, traceable, and change-controlled.
- Task decomposition reveals opportunities for cycle-time reduction; the critical path identifies where optimisation effort should be focused.
- For sequential operations, cycle time can be decomposed into t_p , t_h , t_m , and t_w ; with overlapping operations, CT must be derived from the resource-constrained cyclic schedule.
- Task balancing distributes workload across stations. Balance efficiency measures load uniformity, while capacity margin separately measures the distance between the bottleneck cycle and takt time.
- Analytical checks, simulation-supported validation, and formal design review reduce downstream risk; cost-of-change ratios are project-specific heuristics rather than universal constants.
- The RTM links every baselined requirement to its source, design allocation, verification method, test evidence, acceptance status, and change history.
- The CESPR workflow uses three illustrative uncertainty bands ($\pm 30\% \rightarrow \pm 15\% \rightarrow \pm 5\%$) to communicate increasing design maturity; the justified uncertainty depends on the available evidence.
- Improvement scenarios should evaluate all four CT components simultaneously; reducing one while increasing another may yield net loss.

- The methodology is iterative. Plan for multiple controlled passes and release the design only after assumptions, open issues, and verification evidence are formally accepted.

Review Questions

Answer the following questions to consolidate your understanding of the chapter material. A thorough response should draw on concepts, formulas, and examples presented in the text.

1. Describe the six-step design methodology and explain why it is iterative. What happens if Step 1 (requirements) is performed poorly?
2. A manufacturer needs a cell to produce 250 parts/shift (450 min). Write the performance requirements section of the specification, including takt time, availability target, and OEE target.
3. Decompose the following task into elemental motions and estimate the cycle time: pick a metal bracket from a conveyor, place it in a CNC chuck, close the chuck, retract, wait for machining (60 s), open chuck, extract part, place on outfeed.
4. Define the four components of cycle time. A cell has $CT = 55$ s with $t_p = 22$ s, $t_h = 18$ s, $t_m = 8$ s, $t_w = 7$ s. Compute the VA fraction. Which component would you target first for improvement?
5. An assembly line has six tasks with durations 15, 22, 18, 25, 12, and 20 s, with defined precedence relations and $TT = 30$ s. Apply the Ranked Positional Weight method, verify precedence feasibility, and compute minimum station count, balance efficiency, capacity margin, and RMS station-load deviation.
6. Explain the cost-of-change principle often summarised by the '1:5–10' heuristic. Why should the ratio be treated as project-specific? Give an example where insufficient early validation could create expensive commissioning rework.
7. A customer specification states 'the cell must be flexible.' Rewrite this as a testable requirement with specific acceptance criteria.
8. Explain the difference between the critical path and non-critical operations. In a cell with two parallel robots, can the critical path change depending on the product variant? Give an example.
9. Construct a simplified RTM for a three-requirement cell: (R1) sustained throughput ≥ 80 conforming parts/h under stated conditions, (R2) validated changeover < 10

min, and (R3) PLr d for a defined safety function. For each, specify the source, design allocation, verification method, test-case ID, and acceptance status.

References

- [1] Groover, M.P. (2018) Automation, Production Systems, and Computer-Integrated Manufacturing, 5th ed. Pearson. ISBN 978-0-13-460546-3.
- [2] Siciliano, B., Sciavicco, L., Villani, L., and Oriolo, G. (2009) Robotics: Modelling, Planning and Control. Springer. ISBN 978-1-84628-641-4.
- [3] Nof, S.Y. (ed.) (1999) Handbook of Industrial Robotics, 2nd ed. John Wiley & Sons. ISBN 978-0-471-17783-8.
- [4] Mocan, B. et al. (2018) Multidisciplinary Design of Industrial Robotic Automation Solutions. UTPress. ISBN 978-606-737-246-5.
- [5] Mocan, B. (2025) Design and Operation of Robotic Production Systems — Course Notes. Technical University of Cluj-Napoca.
- [6] Helgeson, W.B. and Birnie, D.P. (1961) Assembly Line Balancing Using the Ranked Positional Weight Technique. *Journal of Industrial Engineering*, 12(6), pp. 394–398.
- [7] Moodie, C.L. and Young, H.H. (1965) A Heuristic Method of Assembly Line Balancing for Assumptions of Constant or Variable Work Element Times. *Journal of Industrial Engineering*, 16(1), pp. 23–29.
- [8] Hopp, W.J. and Spearman, M.L. (2011) Factory Physics, 3rd ed. Waveland Press. ISBN 978-1-57766-739-1.
- [9] ISO 12100:2010, Safety of machinery — General principles for design — Risk assessment and risk reduction. ISO, Geneva.
- [10] ISO 10218-2:2025, Robotics — Safety requirements — Part 2: Industrial robot applications and robot cells. ISO, Geneva.
- [11] ISO 13849-1:2023, Safety of machinery — Safety-related parts of control systems — Part 1: General principles for design. ISO, Geneva.
- [12] Boothroyd, G., Dewhurst, P., and Knight, W. (2011) Product Design for Manufacture and Assembly, 3rd ed. CRC Press.
- [13] Scholl, A. (1999) Balancing and Sequencing of Assembly Lines, 2nd ed. Physica-Verlag, Heidelberg.
- [14] Shell, R.L. and Hall, E.L. (eds.) (2000) Handbook of Industrial Automation. Marcel Dekker.
- [15] Xu, X., Lu, Y., Vogel-Heuser, B., and Wang, L. (2021) Industry 4.0 and Industry 5.0— Inception, conception and perception. *Journal of Manufacturing Systems*, 61, pp. 530–535. <https://doi.org/10.1016/j.jmsy.2021.10.006>.

CHAPTER

5

Robot Selection and End-of-Arm Tooling (EoAT)

Translating process requirements into technical specifications

In this chapter, students learn how to translate cell-level process requirements into defensible specifications for robot and equipment selection. Payload, reachability, pose accuracy, pose repeatability, speed and dynamic load behaviour are treated as coupled design variables rather than isolated catalogue values. End-effectors, fixtures, feeding equipment and tool-changing architectures are presented as integral parts of the robotic system.

The chapter introduces AHP-based multi-criteria decision-making and sensitivity analysis. It also establishes a disciplined verification sequence: apply non-negotiable technical and safety gates first, rank only the feasible candidates, and validate the selected robot–EoAT combination using manufacturer load data, simulation and representative production trials. The governing principle remains that the robot must fit the cell—not the other way around.

LEARNING OBJECTIVES

1. Derive robot-selection requirements—load, reachability, accuracy, repeatability, degrees of freedom and cycle performance—from cell-level task analysis.
2. Distinguish pose accuracy from pose repeatability and identify the complete tolerance stack for a given application.
3. Apply AHP to derive defensible criteria weights and combine them with evidence-based alternative assessment and sensitivity analysis.
4. Verify robot load feasibility using total mass, centre of gravity, moments of inertia, manufacturer load data and the required trajectory.
5. Classify vacuum, mechanical, magnetic, compliant and specialised end-effectors and select an appropriate gripping principle.
6. Size a vacuum gripping system using pressure differential, effective cup area, load direction, safety factor, leakage and evacuation-time requirements.
7. Calculate mechanical gripping force for friction and form-closure concepts, accounting for load direction, friction, finger length and the manufacturer's force definition.
8. Maintain an EoAT mass-properties ledger covering mass, centre of gravity, inertia and moving services.
9. Quantify automatic tool-changer overhead and compare it with multi-tool and dedicated-tool alternatives.
10. Recognise and avoid the recurrent technical, safety and procurement pitfalls identified in this chapter.

5.1 Introduction

Robot and EoAT selection decisions are made early, propagate through the complete cell design and are expensive to reverse. A robot purchase fixes the controller ecosystem, mechanical interfaces, service strategy and a substantial part of the safety architecture. An end-effector usually embeds product-specific geometry, sensing and process knowledge, making late redesign equally disruptive.

A robust selection process follows the sequence requirements → hard technical gates → candidate set → multi-criteria evaluation → sensitivity analysis → validation → documented decision. Requirements originate in the task analysis developed in Chapter 4. Manufacturer catalogues are used only after the application envelope has been defined, and the final decision record should preserve assumptions, data sources, calculations and rejected alternatives.

5.2 Payload, Reach, and Accuracy Requirements

5.2.1 Payload

A robot's rated payload is the maximum permitted total load at the mechanical interface under conditions specified by the manufacturer. **The moving load includes the workpiece, end-effector, adapter, sensors and all services carried by the wrist.** A single catalogue payload value is therefore insufficient: feasibility must be checked using the manufacturer's load diagrams and/or load-validation software with the complete mass properties and the required operating conditions.

Geometric reach and admissible load must not be conflated. Reach is a kinematic quantity measured from the robot base reference to the TCP for the defined tool, whereas manufacturer load diagrams are normally referenced to the wrist flange and describe permitted combinations of total mass, centre-of-gravity position and moments of inertia. A generic payload-versus-horizontal-reach curve—or a universal percentage derating with reach or speed—must not be assumed. Figure 5.1 separates these two verification problems.

Robot joint loads nevertheless depend on robot configuration, gravitational moments, payload inertia, acceleration, velocity-dependent effects and any arm-mounted equipment. For a multi-component tool, calculate the combined centre of gravity from $\mathbf{r}_G = \sum \mathbf{m}_i \mathbf{r}_i / \sum \mathbf{m}_i$ and obtain the inertia tensor from CAD or by applying the parallel-axis theorem. The scalar approximation $J_{\text{flange}} = J_G + m d^2$ is valid only about a specified axis. Enter the validated mass properties in the controller or manufacturer verification tool and repeat the check for the critical robot configurations, orientations and trajectories.

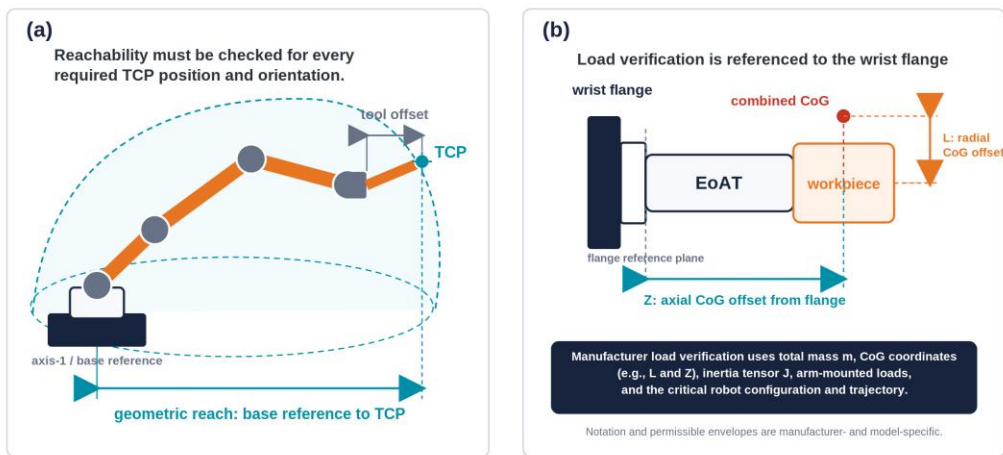


Figure 5.1. Distinction between (a) geometric reach, measured from the robot base reference to the TCP and including the tool offset, and (b) wrist-load characterisation, referenced to the flange by total mass, centre-of-gravity coordinates and inertia. Reachability and load feasibility are separate checks, but both must be validated for the critical robot configurations and trajectories using manufacturer data or software.

5.2.2 Reach

Reach is normally specified as the maximum distance from the robot base reference—typically the axis-1 centreline or base coordinate system—to the TCP. The TCP is the endpoint of this measurement; the distance from the wrist flange to the TCP is the tool offset. Reachability is the ability to attain every required TCP pose, including the specified orientation, without violating joint limits, encountering singularities or colliding with the cell, cable dress, fixtures or guarding. Because the workspace is configuration-dependent and may contain internal exclusions, verify all critical pick, process, inspection, maintenance and recovery poses in a 3D model rather than selecting a robot from nominal maximum reach alone.

5.2.3 Accuracy and Repeatability

ISO 9283 distinguishes pose accuracy—the difference between a commanded pose and the mean achieved pose—from pose repeatability—the dispersion of achieved poses when the same pose is commanded repeatedly. **Most robot datasheets emphasise repeatability, not absolute pose accuracy.** Application accuracy also depends on base and tool calibration, fixture tolerances, thermal effects, compliance, sensor calibration and process forces. Calibration can improve absolute positioning, but vision-guided or force-guided applications may instead rely on accurate relative calibration and closed-loop correction. The relevant requirement is therefore the complete task tolerance stack, not one isolated datasheet number.

Table 5.1 converts process requirements into verifiable robot specifications. Performance tests should follow the applicable ISO 9283 definitions and procedures using calibrated measurement equipment appropriate to the required uncertainty—for example, a laser tracker, photogrammetric system or coordinate-measuring system. A ballbar is primarily a machine-tool diagnostic instrument and should not be presented as the general method for industrial-robot path verification.

Table 5.1. Robot specification parameters: derivation, frequent errors and verification evidence.

Parameter	How to derive the requirement	Frequent error	Verification evidence
Payload / load data	Total moving mass + combined CoG + inertia + arm loads	Checking mass alone or using a generic dynamic factor	Manufacturer load diagram/software; controller load identification; trajectory test
Reachability	All TCP poses and orientations + clearance and recovery access	Using only maximum radial reach	3D simulation with joint limits, singularities, cable dress and collisions
Pose repeatability	Allowable robot contribution within the complete process tolerance stack	Treating repeatability as absolute accuracy	Datasheet/ISO 9283 evidence + application-specific repeatability trial

Parameter	How to derive the requirement	Frequent error	Verification evidence
Pose accuracy	Absolute frame-to-frame positioning requirement after calibration	Assuming calibration of the robot alone closes the full tolerance stack	Calibrated measurement and complete cell calibration
Speed / cycle	Motion, acceleration, settling, sensing and process-time allocation	Using maximum axis or TCP speed as sustained cycle performance	Simulation with realistic trajectory; timed commissioning trial
DOF / configuration	Task orientations, approach paths, obstacles and process constraints	Adding axes without demonstrating need or neglecting external axes	Reachability and singularity analysis; lifecycle-cost comparison

INDUSTRIAL PRACTICE NOTE — Payload Margins Based on Uncertainty

There is no universal ‘60% rule’ that replaces load verification. At concept stage, reserve a documented contingency for uncertain EoAT mass, centre-of-gravity movement and future product variants. As the design matures, replace contingency with measured or CAD-derived mass properties and demonstrate compliance using the manufacturer’s load tools at the required cycle. A scalar ‘dynamic mass’ obtained by adding ma/g to the physical mass is not a valid substitute for robot joint-load analysis.

WORKED EXAMPLE — Load Feasibility Using CoG and Inertia

A machine-tending application handles an 8.0 kg part with a 3.5 kg EoAT. CAD gives the combined load properties at the flange as $m = 11.5$ kg, radial CoG offset $L = 0.16$ m, axial offset $Z = 0.22$ m and maximum relevant inertia $J = 0.43$ kg·m². Peak TCP acceleration on the proposed transfer trajectory is 6 m/s².

Manufacturer check: for the candidate robot and intended wrist range, the 11.5 kg load diagram permits $L \leq 0.20$ m and $Z \leq 0.25$ m, but the permitted inertia for the critical axis is 0.35 kg·m².

Result: mass and CoG pass; inertia fails. The configuration is not yet feasible, even though 11.5 kg is below the nominal 20 kg payload.

Corrective options: shorten or lighten the distal tooling, move components towards the flange, reduce the demanding trajectory and revalidate, or select a robot with a larger admissible inertia envelope. The final choice must be confirmed in the manufacturer's load tool and cycle simulation.

5.3 Multi-Criteria Robot Selection with AHP

The Analytic Hierarchy Process (AHP), developed by Thomas Saaty, structures multi-criteria decisions by decomposing the goal into criteria and alternatives, deriving priorities from reciprocal pairwise comparisons and checking the internal consistency of the judgements. A full AHP compares alternatives pairwise under every criterion. For a compact engineering exercise, this chapter uses AHP to derive criterion weights and then applies anchored 1–10 alternative ratings; this hybrid must be identified explicitly rather than presented as a pure AHP calculation.

5.3.1 The AHP Decision Hierarchy

Typical robot-selection criteria include performance, total cost of ownership, integration fit and risk. Performance covers verified load capability, reachability, cycle performance and process precision. Cost includes acquisition, engineering, peripherals, energy, maintenance and downtime. Integration addresses controller architecture, offline programming, communication interfaces, in-house competence and reuse of existing standards. Risk covers technical maturity, delivery, cybersecurity, service support, spare-parts availability and supplier continuity. Non-negotiable safety and technical constraints

should be treated as hard gates, not traded away through weighting.

5.3.2 Pairwise Comparisons and Consistency

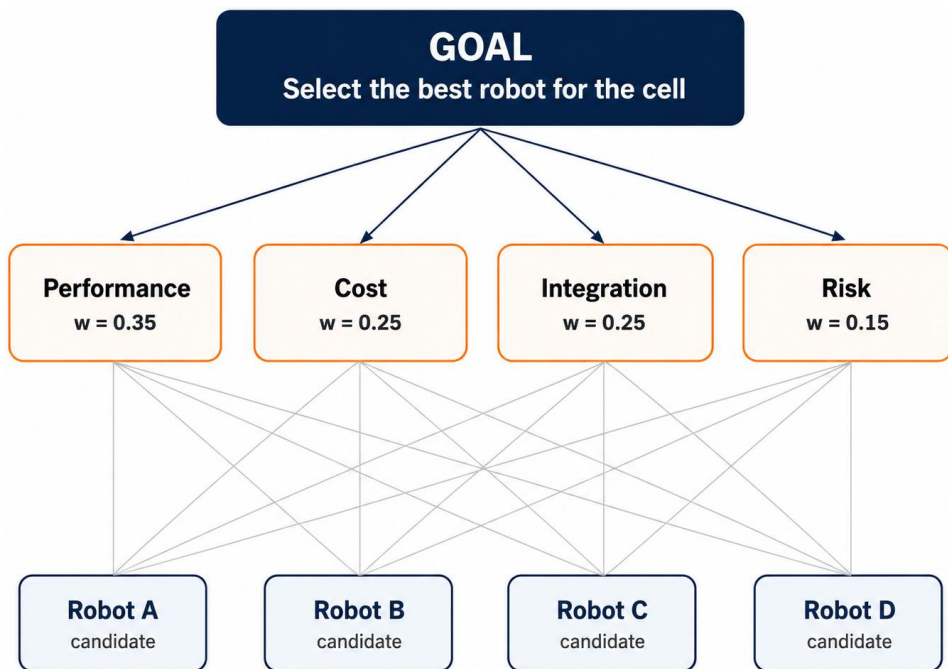
For each pair of criteria, the team assigns relative importance using Saaty's 1–9 scale (Table 5.2) and reciprocal values for the inverse judgement. The conventional acceptance guideline is $CR \leq 0.10$, although the threshold is not a substitute for engineering review. If CR is excessive, inspect the comparison triads, revisit ambiguous definitions and correct the judgements with the decision team; do not alter numbers merely to force a lower ratio.

Table 5.2. Saaty's 1–9 scale for pairwise comparisons in AHP.

Scale Value	Meaning	Example
1	Equal importance	Performance and Cost are equally important
3	Moderate importance	Performance is moderately more important than Risk
5	Strong importance	Cost is strongly more important than Risk
7	Very strong importance	Safety is very strongly more important than Cost
9	Extreme importance	Safety is extremely more important than aesthetics
2, 4, 6, 8	Intermediate values	Used for fine-tuning between scale points

Weights may be obtained from the principal eigenvector or approximated by the row geometric-mean method, followed by normalisation. The consistency index is $CI = (\lambda_{\max} - n)/(n - 1)$, and $CR = CI/RI$, where RI depends on matrix order; for $n = 4$, $RI = 0.90$. The value $\lambda_{\max}$ can be estimated from the component-wise ratios $(Aw)_i/w_i$. Sensitivity analysis is essential because apparently precise rankings can depend strongly on uncertain weights or ratings.

Figure 5.2 shows the decision hierarchy used in the worked example. Hard technical gates are applied before ranking. Criteria weights express the agreed priorities, while each alternative must be assessed from documented evidence using consistent scoring anchors.



Pairwise comparisons determine criteria weights; alternative scores are aggregated into a final ranking.

Figure 5.2. AHP-based decision hierarchy for robot selection: goal, weighted criteria and feasible candidate robots. Non-negotiable technical, safety, environmental and budget gates are applied before the ranking stage.

WORKED EXAMPLE — AHP-Weighted Comparison of Four Candidates

A machine-tending cell requires a verified 15 kg load case, 1.2 m reachability, pose repeatability no worse than ± 0.10 mm and an installed budget ceiling of €60,000. Four candidates A–D pass the hard gates. Commercial values are anonymised because quotations and product variants are project-specific.

Pairwise matrix (Performance, Cost, Integration, Risk): $[[1, 2, 1, 3], [1/2, 1, 1, 2], [1, 1, 1, 2], [1/3, 1/2, 1/2, 1]]$. Geometric-mean weights are 0.365, 0.233, 0.277 and 0.125; CR = 0.017.

Anchored ratings (1–10): Performance A/B/C/D = 8/9/7/8; Cost = 9/6/7/8; Integration = 9/7/6/6; Risk = 8/8/7/6.

Weighted results: A = 8.51, B = 7.62, D = 7.20 and C = 6.72. Candidate A is preferred in the base case. This result is traceable, but it remains conditional on the rating evidence and sensitivity analysis.

Table 5.3 tests the ranking under four explicit priority scenarios. Candidate A remains first in three scenarios, while Candidate B becomes preferable when performance receives a dominant weight of 0.65. The sensitivity result is more informative than an unsupported ‘tipping point’: it shows precisely which strategic assumption can reverse the decision.

Table 5.3. Sensitivity analysis for the AHP-weighted rating model. Weights sum to 1.00 in every scenario.

Scenario	Performance	Cost	Integration	Risk	Winner	Score
Base case	0.365	0.233	0.277	0.125	A	8.51
Budget-constrained	0.25	0.40	0.20	0.15	A	8.60
Performance-critical	0.65	0.10	0.10	0.15	B	8.35
New site / lower ecosystem value	0.40	0.25	0.10	0.25	A	8.35

5.4 End-of-Arm Tooling (EoAT)

5.4.1 EoAT Taxonomy

End-of-arm tooling should be classified by physical interaction principle and process function, not merely by commercial product category. Vacuum systems generate normal force through pressure differential; mechanical grippers use friction or form closure; magnetic systems act on ferromagnetic materials; compliant devices adapt to fragile or variable geometry; and process tools deliver operations such as welding, fastening or dispensing. Automatic tool changers are tooling interfaces rather than gripping principles, but they materially affect the EoAT architecture.

Table 5.4. EoAT families, working principles, suitable applications and primary design checks.

EoAT family	Working principle	Suitable applications	Primary design checks	Typical limitation
Vacuum cup / area gripper	Pressure differential over effective sealed area	Sealable sheets, cartons, glass, panels and packaged products	Pull-off and shear load, moment, leakage, evacuation time, vacuum sensing	Porosity, contamination, edge leakage and loss of vacuum
Parallel / centric mechanical	Normal jaw force with friction and/or geometry	Prismatic and cylindrical parts with defined grasp regions	Jaw force definition, finger length, stroke, friction, contact stress	Surface marking, tolerance sensitivity, limited part-family range
Form-closure custom fingers	Geometric capture rather than friction alone	Features, flanges, holes, rims and parts with repeatable geometry	Insertion clearance, retention direction, tolerance stack, release	Product-specific tooling and jam risk
Magnetic	Permanent, electro-permanent or electromagnetic attraction	Ferromagnetic sheet, castings, blanks and machined parts	Air gap, pull-off/shear force, residual magnetism,	Material restriction and reduced force across coatings or gaps

EoAT family	Working principle	Suitable applications	Primary design checks	Typical limitation
			power-loss state	
Soft / compliant	Elastic or underactuated contact conforms to the part	Fragile, irregular, deformable and variable products	Repeatability, pressure distribution, durability, hygiene	Lower stiffness, wear and more difficult precision modelling
Specialised	Needle, Bernoulli, adhesive, electrostatic or other principle	Textiles, porous foods, wafers and application-specific surfaces	Damage mechanism, contamination, energy and release behaviour	Narrow application range and specialised validation
Process tool	Transfers process energy or material	Welding, screwdriving, deburring, dispensing, inspection	Reaction forces, utilities, calibration, process quality and safety	May dominate mass, inertia and service routing
Tool changer / multi-tool interface	Connects, locks and transfers utilities between tools	High-mix or multi-operation cells	Lock verification, payload/inertia , utility coupling, docking tolerance	Cycle overhead, maintenance and added failure modes

Selection begins with the part and the worst credible process load case, then considers surface condition, grasp geometry, allowable contact pressure, required precision, environment, sensing and loss-of-energy behaviour. Figure 5.3 provides a structured screening logic. It is deliberately non-exclusive: hybrid grippers and redundant retention may be necessary when one physical principle cannot satisfy all requirements.

Selection may require a hybrid solution; validate the chosen concept on representative production parts.

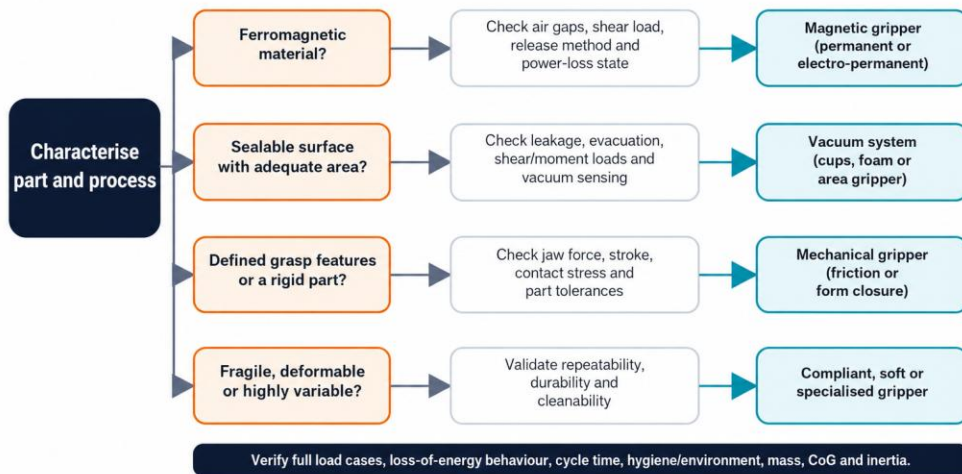


Figure 5.3. Structured EoAT selection logic based on material, surface sealability, grasp features and part compliance. Every branch is followed by verification of load cases, loss-of-energy behaviour, cycle time, environment and complete mass properties.

5.4.2 EoAT Selection Criteria

EoAT selection is a system-engineering activity. The gripper must retain every approved part variant under normal motion, emergency stop and credible fault conditions while preserving the required placement quality and process safety. The relevant evidence combines calculations, manufacturer data, risk assessment and trials on representative production parts.

For a friction grip, the available tangential holding force is approximately $\Sigma(\mu_i N_i)$ and must exceed the factored external load in the relevant direction. For a vacuum grip, the theoretical normal force is ΔpA , but effective force is reduced by cup geometry, leakage, load direction and moments. Formulae are screening tools: finger deflection, contact stress, vacuum evacuation, sensor thresholds and energy-loss behaviour must also be verified. Table 5.5 lists the minimum review set.

Table 5.5. *EoAT selection criteria and the evidence required for design acceptance.*

Criterion	Key question	Required evidence
Part and surface compatibility	Will every approved variant be grasped without unacceptable damage?	Part-family analysis and trials on representative worst-case parts
Load-case capacity	Are pull-off, shear and moment capacities sufficient in normal and stop conditions?	Free-body model, manufacturer data and measured margin
Loss-of-energy behaviour	What happens after loss of vacuum, air or electrical power?	Risk assessment, check valve/reservoir or secondary retention, monitored test
Placement quality	Does the grasp locate the part within the process tolerance stack?	Repeatability, compliance and datum-transfer measurement
Cycle performance	Are close/open, evacuation, sensing and stabilisation times within the budget?	Timed test using production settings
Environment / hygiene	Are materials and geometry suitable for temperature, contamination and cleaning?	Applicable regulatory/standard review; cleanability and material evidence
Reliability / maintainability	Can wear items be inspected and replaced within the maintenance strategy?	Duty profile, preventive-maintenance plan and spares strategy
Mass properties	Do total mass, CoG and inertia remain within the robot's validated load envelope?	CAD/measurement ledger and manufacturer load validation

INDUSTRIAL PRACTICE NOTE — EoAT Verification Protocol

Separate engineering screening from statistical reliability demonstration. A 100-cycle trial on each variant can expose gross problems, but zero failures in 100 picks does not establish a 99.5% success probability. With zero failures, at least 598 independent cycles are required to demonstrate reliability of at least 99.5% with one-sided 95% confidence. Define the target, confidence level, independence assumptions and acceptance rule before testing. Include the heaviest, lightest, most slippery, most porous and most dimensionally extreme approved parts; measure grip signals, placement dispersion, cycle time, stop behaviour and wear.

INDUSTRIAL PRACTICE NOTE — EoAT Mass-Properties Ledger

Maintain a controlled ledger from concept to commissioning: gripper body, fingers, adapters, fasteners, sensors, valves, connectors, cable brackets, moving hoses/cables and the workpiece. Record mass, CoG coordinates and inertia—not mass alone—and link each value to CAD or measurement evidence. Set a design contingency appropriate to the maturity of the concept, then reduce it as components are frozen. Review the ledger at every design change and re-run the manufacturer load validation whenever geometry or services move.

WORKED EXAMPLE — Vacuum Gripper Sizing for a Sealed Carton

Task: handle a 4.0 kg sealed carton by its horizontal upper surface using two cups. Horizontal acceleration $a_h = 5.0 \text{ m/s}^2$, no commanded vertical acceleration, pressure differential $\Delta p = 60 \text{ kPa}$, normal-force factor $\eta = 0.75$ and safety factor $SF = 2.0$.

Factored normal pull-off load: $F_n = mgSF = 4 \times 9.81 \times 2 = 78.5 \text{ N}$.

Required total effective area: $A_{\text{req}} = F_n / (\Delta p \eta) = 78.5 / (60,000 \times 0.75) = 0.00174 \text{ m}^2 = 17.4 \text{ cm}^2$. With two equal cups, each requires at least 8.72 cm^2 , equivalent to a theoretical diameter of 33.3 mm.

Factored tangential shear load: $F_t = ma_h SF = 4 \times 5 \times 2 = 40.0 \text{ N}$. This demand must be checked against the manufacturer's measured shear capacity for the selected cups and surface; it cannot be inferred from $\Delta p A$ alone.

Selection and verification: two 40 mm cups provide 25.1 cm^2 total nominal area and 113 N estimated effective normal force, exceeding the 78.5 N factored pull-off demand. Verify the 40.0 N shear demand, overturning moments, permeability, evacuation time, vacuum-switch threshold and loss-of-vacuum risk controls. Airflow and compressed-air cost cannot be derived from holding force alone; they depend on leakage, volume, ejector/pump characteristics and on-time.

WORKED EXAMPLE — Mechanical Gripper Force for a Cylindrical Part

A two-jaw parallel gripper holds a 2.5 kg steel cylinder during horizontal acceleration of 4.0 m/s^2 . The tested friction coefficient between the jaw inserts and part is $\mu = 0.60$; $SF = 2.0$. Gravity and horizontal acceleration are orthogonal.

Factored tangential load: $F_t = mv(g^2 + a_h^2)SF = 2.5\sqrt{9.81^2 + 4^2} \times 2 = 52.9 \text{ N}$.

Required normal force per jaw: $N_{\text{jaw}} \geq F_t / (2\mu) = 52.9 / (2 \times 0.60) = 44.1 \text{ N}$ per jaw. If a catalogue states total closing force rather than force per jaw, the required total is at least 88.2 N.

Verification: select using the force available at the actual operating pressure and finger length, then verify finger deflection, contact stress, opening tolerance, external moments and emergency-stop load. Friction values must be established on representative surface conditions.

5.5 Automatic Tool Changers (ATC)

Automatic tool changers (ATCs) allow one robot to use several end-effectors or process tools while preserving a standard mechanical and utility interface. The architecture comprises a robot-side master, tool-side plates, locking and verification functions, utility modules and one or more docking stations. Selection must cover payload, moment and inertia limits, locking force, repeatability, environmental protection, utility capacity, fail-safe state and the consequences of an incorrect or incomplete lock.

Tool-change overhead is not only the catalogue lock/unlock time. It includes approach, alignment, uncoupling, travel to the next tool, coupling, utility stabilisation, lock verification and recovery allowances. The per-part contribution is $t_{ATC} = n_{swap}(t_{unmate} + t_{travel} + t_{mate} + t_{verify})$. Compare this value with the final cycle time and takt requirement. A multi-tool EoAT eliminates docking but increases mass, inertia, cable complexity and inactive-tool collision risk; a dedicated second robot changes capital cost, floor space and line balancing.

INDUSTRIAL PRACTICE NOTE — ATC Duty-Cycle Calculation

Calculate the actual tool-change sequence rather than applying a fixed seconds-per-swap assumption. The overhead fraction of the final cycle is $t_{ATC}/(t_{base} + t_{ATC})$. Include docking search, controller handshakes, utility pressure build-up and fault recovery in the model. Compare ATC, multi-tool and dedicated-tool architectures using the same takt, payload, collision, maintenance and risk assumptions. Wear and inspection intervals are model- and application-specific and must come from the selected manufacturer's documentation.

WORKED EXAMPLE — ATC versus Dual-Tool EoAT

A cell alternates between vacuum handling and screwdriving. The base operation time is 30 s. An ATC sequence requires two 4 s changes per part. A dual-tool concept adds 1.5 kg and 1.5 s of extra motion per cycle. Illustrative commercial allowances are €4,500 for the ATC plus €800/year maintenance and €2,800 for the dual-tool adapter and routing.

ATC architecture: cycle = $30 + 8 = 38$ s; throughput = $3,600/38 = 94.7$ parts/h. Tool-changing occupies $8/38 = 21.1\%$ of the final cycle.

Dual-tool architecture: cycle = $30 + 1.5 = 31.5$ s; throughput = 114.3 parts/h, 20.7% above the ATC case.

Decision: the dual-tool concept is preferred only if the additional mass, CoG, inertia, utility routing and inactive-tool collision envelope pass validation. ATC may still be superior where tools are heavy, numerous, contaminated, rarely used or incompatible on one flange. The conclusion is application-specific, not a universal threshold based on tool count.

Figure 5.4 compares the two architectures. The ATC configuration stores separate tools and moves the robot through a verified docking sequence. The dual-tool configuration carries both tools continuously; it removes docking delay but changes the robot load and creates an additional swept volume that must be included in collision and safety analysis.

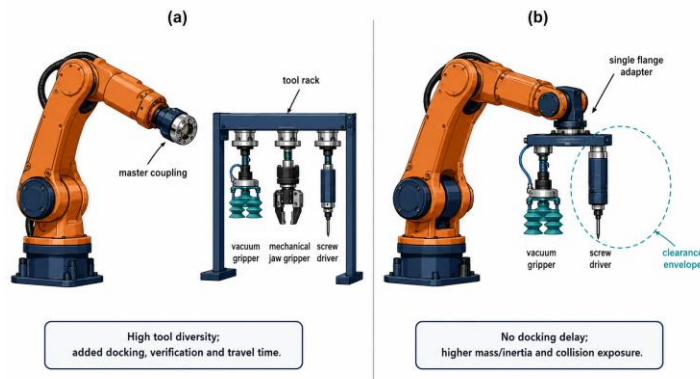


Figure 5.4. Comparison of multi-function tooling architectures: (a) automatic tool changer with a tool rack; (b) dual-tool EoAT with both tools mounted permanently. The inactive-tool clearance envelope is a critical design constraint in the dual-tool case.

Neither architecture is inherently more flexible or more productive in every application. The correct comparison includes final cycle time, complete mass properties, docking and utility reliability, collision exposure, maintenance access, contamination control, changeover strategy and lifecycle cost.

5.6 Matching Robot Type to Application

Kinematic configuration determines orientation capability, workspace topology, stiffness, speed, payload distribution and integration complexity. Six-axis articulated robots are suited to tasks requiring general orientation control and obstacle avoidance. SCARA robots are effective for fast planar assembly and handling with vertical insertion. Delta/parallel robots are advantageous for lightweight high-speed sorting, while Cartesian and gantry systems provide scalable rectangular coverage. Dedicated electronic-component placement machines may outperform all general-purpose robot types; robot selection must remain application-specific rather than based on a simplified speed slogan.

Table 5.6. Robot configuration guide by application and principal constraint.

Application	Typical configuration	Why it fits	Principal constraint / alternative
Arc welding	6-axis articulated	Continuous orientation control and access around the joint	Process cable routing, singularities and fume/safety integration
Top-down assembly / PCB handling	SCARA	High planar stiffness, fast horizontal motion and vertical insertion	Limited out-of-plane orientation; dedicated placement equipment may be superior
High-speed lightweight sorting	Delta/parallel or high-speed SCARA	Low moving inertia and short repetitive paths	Restricted payload, orientation and workspace; verify actual picks/min on the path
Large rectangular coverage	Cartesian / gantry	Scalable axes and predictable	Structure, floor/overhead

Application	Typical configuration	Why it fits	Principal constraint / alternative
		rectangular workspace	integration and lower orientation dexterity
Complex machine tending	6-axis, possibly on a linear axis	Dexterity around doors, fixtures and multiple machines	External-axis coordination, reachability and guarding
Deformable or bimanual assembly	Dual-arm or two coordinated robots	Simultaneous holding and manipulation	Coordination complexity, collision avoidance and sensing
Collaborative application	Robot and cell engineered for the selected collaborative method	Can enable shared-space operation under defined conditions	A 'cobot' is not automatically safe; ISO 10218-2 risk assessment and validated safety functions are required

A mismatch may appear as insufficient capability or as unnecessary complexity. More importantly, robot type does not determine safety by itself. A collaborative application is a property of the complete task and cell, including end-effector, workpiece, speed, separation monitoring, contact hazards and validated safety functions. ISO 10218-1:2025 covers industrial-robot requirements, while ISO 10218-2:2025 covers integration of robot applications and cells; ISO/TS 15066:2016 remains relevant to collaborative operation. Physical guarding may still be required for hazards created by the process or EoAT.

Figure 5.5 compares schematic workspace topologies. These envelopes are intentionally simplified and must not be used for final reach verification.

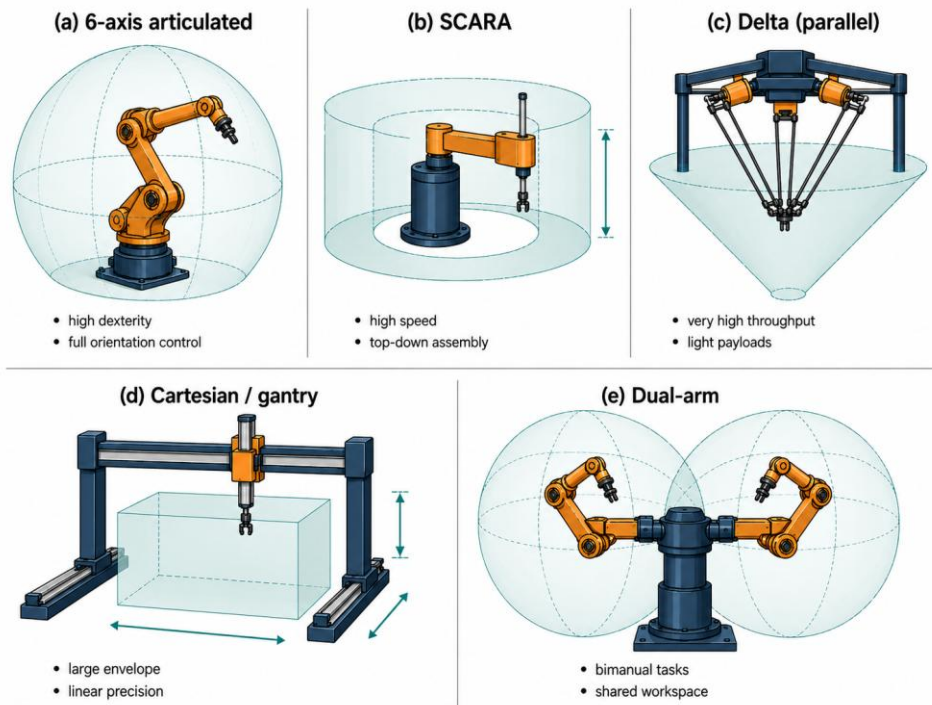


Figure 5.5. Schematic workspace topologies for five robot configurations: (a) 6-axis articulated, (b) SCARA, (c) delta/parallel, (d) Cartesian/gantry and (e) dual-arm/coordinated pair. Exact reachable volumes depend on joint limits, mounting, tool orientation and collision constraints.

The diagrams communicate topology rather than exact geometry. An articulated robot has a configuration-dependent volume with internal exclusions and singularities; a SCARA workspace is typically annular rather than a full cylinder; a delta workspace depends strongly on linkage and orientation constraints; and the useful dual-arm region is the collision-free intersection of two workspaces. Final selection therefore requires the manufacturer workspace model and the complete digital cell.

Common Pitfalls

Pitfall 1: Over-sizing the robot.

Selecting a much larger robot ‘for future flexibility’ can increase capital cost, floor loading, footprint, energy use and collision consequences while reducing attainable acceleration. Define realistic product-growth scenarios and compare modular alternatives rather than buying unused capacity.

Pitfall 2: Ignoring manufacturer load data.

Checking only nominal payload or workpiece mass ignores EoAT mass, CoG, inertia, arm loads and trajectory. Validate the complete moving load with the specific manufacturer's data and software.

Pitfall 3: Confusing accuracy, repeatability and process capability.

Repeatability alone does not guarantee accurate insertion or vision performance. Analyse calibration, sensing, compliance, fixture error and process forces within the complete tolerance stack.

Pitfall 4: Choosing EoAT before understanding workpiece variability.

A gripping principle that succeeds on clean prototypes may fail on oily, dusty, porous, warped or dimensionally extreme production parts. Test the approved part family and surface conditions.

Pitfall 5: Tracking EoAT mass but not CoG and inertia.

Design changes add fingers, sensors, adapters and services and may shift the CoG even when mass growth seems small. Maintain a mass-properties ledger and revalidate after every change.

Pitfall 6: Single-criterion selection.

Choosing the least expensive robot—or the customary brand—conceals trade-offs. Use hard gates, documented criteria, consistency checking and sensitivity analysis so assumptions remain visible.

Pitfall 7: Ignoring the integration ecosystem.

A technically strong robot can create unacceptable operational risk if local service, cybersecurity support, spares, training or compatible engineering tools are weak. Evaluate the ecosystem as lifecycle evidence, not brand preference.

Pitfall 8: Specifying more DOF than needed.

Extra axes may help dexterity, but they also add cost, programming complexity, singularities and maintenance. Select the simplest configuration that satisfies every required pose and future scenario.

Pitfall 9: Not testing EoAT on worst-case workpieces.

The nominal part is rarely the limiting case. Include extremes of mass, friction, porosity, geometry, temperature, contamination and tolerance—and consider combinations of adverse conditions.

Pitfall 10: Treating ATC as a universal solution.

An ATC adds docking hardware, controls, utilities, maintenance and cycle time. Compare the calculated sequence with multi-tool and dedicated-tool architectures using the same assumptions.

KEY TAKEAWAYS

- Derive robot requirements from the complete task and cell, then apply hard technical and safety gates before commercial ranking.
- Robot load feasibility is defined by total moving mass, centre of gravity, inertia, arm loads and the required trajectory—not by nominal payload or a universal reach-derating percentage.
- Pose accuracy and pose repeatability are different ISO 9283 performance characteristics; application capability depends on the complete tolerance stack.
- AHP can provide traceable criterion priorities, but a hybrid AHP-weighted rating model must be labelled correctly and supported by scoring anchors and sensitivity analysis.
- EoAT selection begins with the worst credible part, surface and load case and must include energy-loss behaviour, sensing, environment and maintainability.
- Vacuum force depends on pressure differential and effective area; shear, moments, leakage and evacuation time often govern the real design.
- Mechanical gripping calculations must use the relevant load vector, tested friction, actual finger length and the manufacturer's definition of gripping force.
- Maintain an EoAT mass-properties ledger covering mass, CoG and inertia. Revalidate the robot load whenever the tool or moving services change.
- Calculate the complete ATC sequence. Compare ATC, multi-tool and dedicated-tool alternatives using the same takt, load, collision, maintenance and risk assumptions.
- Robot configuration must match task topology. A collaborative robot does not make an application safe without a complete risk assessment and validated safety functions.

Review Questions

Answer the following questions to consolidate your understanding of the chapter material. A thorough response should draw on concepts, formulas, and examples presented in the text.

1. A robot handles a 15 kg workpiece with a 4 kg EoAT. Explain why multiplying 19 kg by a generic dynamic factor is insufficient for robot selection. List the additional load data and validation steps required.
2. Distinguish pose accuracy, pose repeatability and process capability. For vision-guided bin picking and for pin insertion with 0.02 mm clearance, identify the dominant contributors to the complete tolerance stack.
3. Construct the AHP pairwise matrix for Performance, Cost and Risk when Performance is moderately more important than Cost (value 3) and Cost is between equally and moderately more important than Risk (value 2). Compute the geometric-mean weights and CR.
4. A four-criterion matrix has CR = 0.14. What does this indicate, and how should the team identify and revise inconsistent comparison triads without manipulating the result mechanically?
5. Two vacuum cups lift a 6 kg sealed plate from its horizontal upper surface. The pressure differential is 60 kPa, normal-force factor $\eta = 0.75$, horizontal acceleration is 4 m/s^2 and SF = 2.0. Determine the minimum effective normal area and theoretical diameter per cup from the pull-off load, calculate the factored tangential shear demand, and list the checks not captured by these calculations.
6. A two-jaw gripper holds a 1.8 kg aluminium cylinder during horizontal acceleration of 5 m/s^2 . The tested friction coefficient is $\mu = 0.40$ and SF = 2.0. Compute the required normal force per jaw using the resultant of gravity and horizontal acceleration.
7. A cell has a 42 s base cycle and two tool changes. Each complete change sequence contributes 3 s. Compute the final cycle, throughput and ATC overhead as a percentage of the final cycle. Which additional factors determine whether a dual-tool concept is preferable?
8. A robot's manufacturer load tool permits 32 kg for a specified CoG and inertia. Workpiece and EoAT mass total 25 kg, but the calculated inertia exceeds the permitted value by 18%. Is the configuration feasible? Explain why mass margin alone is not decisive.

9. Compare a six-axis articulated robot, SCARA and dedicated placement machine for a PCB-related application requiring 120 placements/min. Define the information needed before selecting a configuration and avoid assuming that robot type alone determines performance.

References

- [1] Groover, M.P. (2018) Automation, Production Systems, and Computer-Integrated Manufacturing, 5th ed. Pearson. ISBN 978-0-13-460546-3.
- [2] Siciliano, B., Sciavicco, L., Villani, L. and Oriolo, G. (2009) Robotics: Modelling, Planning and Control. Springer London. ISBN 978-1-84628-641-4.
- [3] Saaty, T.L. (1980) The Analytic Hierarchy Process. McGraw-Hill, New York.
- [4] Nof, S.Y. (ed.) (1999) Handbook of Industrial Robotics, 2nd ed. Wiley. ISBN 978-0-471-17783-8.
- [5] Mocan, B. et al. (2018) Multidisciplinary Design of Industrial Robotic Automation Solutions. UTPress. ISBN 978-606-737-246-5.
- [6] Mocan, B. (2025) Design and Operation of Robotic Production Systems—Course Notes. Technical University of Cluj-Napoca.
- [7] ISO 9283:1998, Manipulating industrial robots—Performance criteria and related test methods. ISO, Geneva.
- [8] ISO 10218-1:2025, Robotics—Safety requirements—Part 1: Industrial robots. ISO, Geneva.
- [9] ISO 10218-2:2025, Robotics—Safety requirements—Part 2: Industrial robot applications and robot cells. ISO, Geneva.
- [10] ISO/TS 15066:2016, Robots and robotic devices—Collaborative robots. ISO, Geneva.
- [11] ISO 9409-1:2004, Manipulating industrial robots—Mechanical interfaces—Part 1: Plates. ISO, Geneva.
- [12] ISO 11593:2022, Robots for industrial environments—Automatic end-effector exchange systems—Vocabulary. ISO, Geneva.
- [13] ISO/TR 20218-1:2018, Robotics—Safety design for industrial robot systems—Part 1: End-effectors. ISO, Geneva.
- [14] ISO 14159:2002, Safety of machinery—Hygiene requirements for the design of machinery. ISO, Geneva.
- [15] Monkman, G.J., Hesse, S., Steinmann, R. and Schunk, H. (2007) Robot Grippers. Wiley-VCH. ISBN 978-3-527-40619-7.
- [16] Schmalz GmbH (2026) Vacuum Knowledge: Theoretical Holding Force of a Suction Cup and System Design Guidance. Glatten.
- [17] SMC Corporation (2026) Vacuum Equipment Model Selection—Technical Data. Tokyo.
- [18] ATI Industrial Automation (2026) Robotic Tool Changer Product Documentation and Maintenance Guidance. Apex, North Carolina.

CHAPTER

6

Communication, Control Architecture, and PLC Integration

The nervous system of the robotic production system

This chapter examines the control and communication infrastructure that enables coordinated operation within robotic production systems. It distinguishes the responsibilities of robot controllers, programmable logic controllers (PLCs), safety-related control systems, sensors, human-machine interfaces (HMIs), supervisory systems, and manufacturing software. Classical device interfaces and modern Industrial Ethernet technologies are discussed with attention to timing, determinism, diagnostics, interoperability, lifecycle support, and cybersecurity.

Control architecture is the nervous system of a robotic production system: it determines whether the components operate as a coordinated, diagnosable, and safe whole rather than as a collection of independent devices.

LEARNING OBJECTIVES

1. Describe the ISA-95 functional hierarchy and distinguish it from a Purdue-style architectural representation.
2. Compare PROFINET, EtherCAT, and EtherNet/IP using timing mechanisms, ecosystem compatibility, safety profiles, diagnostics, and lifecycle constraints.
3. Explain OPC UA as a semantic integration technology and describe the role of the OPC UA Robotics Companion Specification.
4. Explain the current IEC 61131-3 programming suite (ST, LD, FBD, and SFC) and recognise Instruction List as a legacy language.
5. Design standard and safety-related I/O mapping between the robot, PLC, and peripheral equipment in a robotic cell.
6. Implement request-acknowledge-complete handshakes with timeouts, reset conditions, and explicit fault-state transitions.
7. Apply IEC 62443 principles through risk assessment, zones and conduits, target security levels, least privilege, and secure remote access.
8. Apply ISA-101 HMI lifecycle principles and IEC 62682/EEMUA alarm-management principles to operator-interface design.
9. Plan integration testing that verifies I/O, handshakes, negative cases, safety functions, timing, and recovery before commissioning.

10. Select OPC UA and MQTT appropriately for edge and cloud data exchange without using either as a substitute for deterministic or safety-related control.

6.1 The Hierarchical Control Architecture

Industrial automation is commonly represented through five functional levels. ISA-95 defines the interface between enterprise and manufacturing-control functions, whereas the Purdue model is a related architectural representation used to organise systems and network boundaries. The levels are therefore responsibility boundaries rather than universal prescriptions for latency, devices, or protocols (table 6.1).

Table 6.1. Five functional levels used in ISA-95/Purdue-style control architecture.

Level	Functional responsibility	Indicative timescale	Typical systems	Common interfaces
0	Physical process	µs-ms	Sensors, valves, drives, tooling	Discrete/analogue I/O, IO-Link, drive interfaces
1	Basic control	ms	PLCs, robot controllers, safety controllers	PROFINET, EtherCAT, EtherNet/IP, safety profiles
2	Area and supervisory control	s-min	HMI, SCADA, cell supervisor, local historian	OPC UA, Modbus TCP, vendor APIs
3	Manufacturing operations management	min-h	MES/MOM, quality, scheduling, maintenance	OPC UA, REST, MQTT, databases
4	Business planning and logistics	h-days	ERP, supply chain, finance	Enterprise APIs, message brokers, database connectors

The preferred design principle is controlled information flow through documented interfaces. Most real-time control remains within Levels 0-1, while supervisory and business functions operate at higher levels (figure 6.1). Cross-level communication can be valid—for example, condition-monitoring data sent from a drive to an analytics

platform—but it should pass through engineered gateways, security controls, and an explicit ownership model rather than through ad hoc connections.



The levels define responsibilities and interfaces. Cross-level data paths are acceptable only through engineered, controlled gateways.

Figure 6.1. Five-level functional hierarchy: Level 0 represents the physical process and Level 4 business planning and logistics. The side arrows indicate increasing aggregation upwards and increasing control directness and determinism downwards.

6.2 Industrial Ethernet Technologies for Robotic Systems

Level 0 and Level 1 equipment may use hardwired I/O, IO-Link, drive-specific interfaces, or Industrial Ethernet. PROFINET, EtherCAT, and EtherNet/IP are widely supported options, but no protocol is universally superior. Selection must be based on the complete system requirement and the devices that must interoperate (table 6.2).

Table 6.2. Engineering comparison of PROFINET, EtherCAT, and EtherNet/IP.

Technology	Timing and synchronisation	Safety profile	Ecosystem fit	Engineering cautions
PROFINET RT / IRT	IRT supports sub-millisecond cycles and, on suitable hardware, update times down to 31.25 μs. Scheduled traffic and clock synchronisation provide sub-microsecond jitter capability.	PROFIsafe	Broad multi-vendor; common in Siemens-based plants.	Verify IRT-capable devices, topology, update time, and bandwidth.
EtherCAT	Cycle times of 100 μs or below and synchronisation jitter below 1 μs are achievable in suitable systems using on-the-	FSoE	High-performance motion;	Verify master and device profiles, topology, task

Technology	Timing and synchronisation	Safety profile	Ecosystem fit	Engineering cautions
	fly processing and distributed clocks.		common with Beckhoff.	timing, and recovery.
EtherNet/IP	General I/O commonly uses millisecond requested packet intervals; CIP Sync and CIP Motion support time-synchronised coordinated motion.	CIP Safety	Broad CIP; common in Rockwell plants.	Engineer RPI, QoS, multicast, switch capacity, clock sync, and recovery.

6.2.1 OPC UA: The Semantic Integration Layer

OPC Unified Architecture (IEC 62541) provides platform-independent, secure, and semantically structured communication between devices, control systems, and enterprise software. Its information models can represent objects, relationships, units, status, and data quality rather than only raw registers. Domain-specific Companion Specifications standardise these models. OPC 40010-1, Robotics Part 1, provides a common information model for vertical integration of industrial robots, including identification, status, operational information, diagnostics, and related asset data. OPC UA is not, by itself, a replacement for a deterministic motion bus.

6.2.2 MQTT for Machine-to-Cloud Telemetry

MQTT is a lightweight, brokered publish-subscribe protocol suited to telemetry, event distribution, and intermittently connected edge-to-cloud applications. Quality of Service levels 0, 1, and 2 govern message-delivery behaviour, but they do not create deterministic real-time control or safety integrity at application level. A robust architecture may use an Industrial Ethernet technology for cyclic cell control, OPC UA for contextualised machine integration, and MQTT for selected telemetry to an edge or cloud platform.

INDUSTRIAL PRACTICE NOTE — Engineering PROFINET IRT for Coordinated Motion

For coordinated motion, the relevant quantities are update time, jitter, clock accuracy, topology, network load, and recovery behaviour—not the nominal link speed alone. PROFINET IRT can provide sub-millisecond cycles and sub-

microsecond jitter when the complete IRT path is engineered accordingly. Verify the required conformance class and real-time class; confirm that every controller, device, and infrastructure component in the IRT domain supports the selected update time; calculate bandwidth for cyclic and acyclic traffic; and validate the topology under representative load. Media Redundancy Protocol (MRP) improves availability after a link failure, but it does not inherently improve motion determinism. Record the final network parameters in the integration specification.

Figure 6.2 presents a requirement-driven selection workflow. Timing and synchronisation are assessed together with ecosystem compatibility, safety functions, diagnostics, topology, resilience, engineering competence, and lifecycle support. Technologies that fail a mandatory requirement are removed from the shortlist. The remaining architecture must then be validated using representative controllers, devices, switches, traffic loads, fault conditions, and recovery scenarios. A safety profile such as PROFI-safe, FSoE, or CIP Safety is an enabling mechanism; achievement of a required SIL or PL depends on the design and validation of the complete safety function.

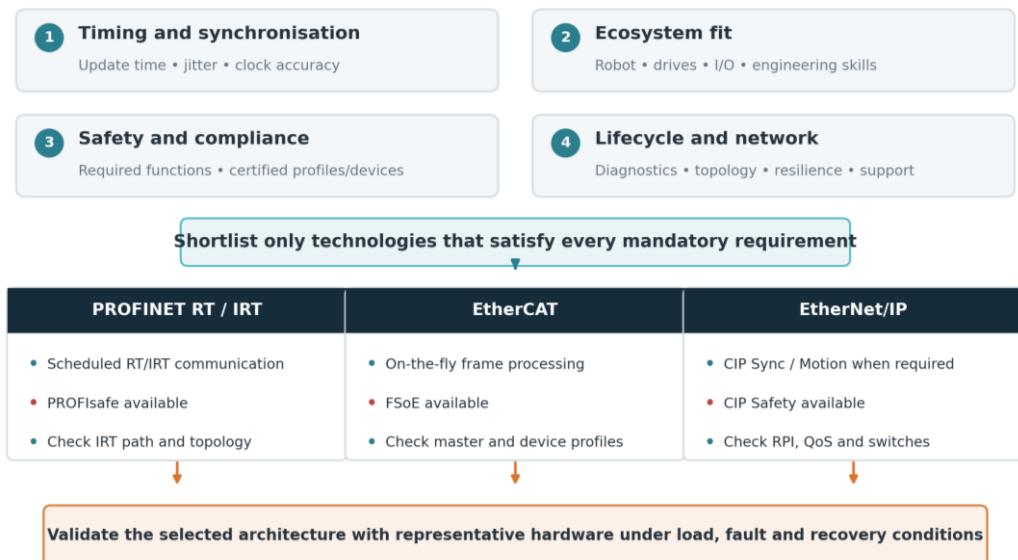


Figure 6.2. Protocol-selection workflow based on timing, ecosystem fit, safety and compliance, lifecycle, network design, and hardware validation.

6.3 Programmable Logic Controllers (PLC)

PLCs implement deterministic logic, sequencing, interlocking, diagnostics, and coordination at Level 1. IEC 61131-3:2025 specifies Structured Text (ST), Ladder Diagram (LD), and Function Block Diagram (FBD), together with Sequential Function Chart (SFC) elements for structuring sequential behaviour. Instruction List (IL), present in older editions, is a legacy language and is not part of the current edition (table 6.3).

Table 6.3. *Programming languages and sequencing elements in IEC 61131-3:2025.*

Element	Classification in IEC 61131-3:2025	Main strength	Typical use	Status
Structured Text (ST)	Textual language	Algorithms, data structures, reusable functions	Calculations, communications, complex logic	Current
Ladder Diagram (LD)	Graphical language	Relay-style binary logic and transparent online diagnostics	Interlocks, permissives, maintenance troubleshooting	Current
Function Block Diagram (FBD)	Graphical language	Function and signal flow	Control loops and reusable function blocks	Current
Sequential Function Chart (SFC)	Sequencing and structuring elements	Explicit steps, transitions, and actions	Machine sequences and handshakes	Current structuring formalism
Instruction List (IL)	Textual language in earlier editions	Compact low-level representation	Maintenance of legacy projects only	Removed from the 2025 edition

A maintainable cell program normally separates concerns: SFC or an equivalent state-machine structure defines the overall sequence; ST implements calculations, data handling, communication services, and reusable logic; LD or FBD may be used where their graphical form improves diagnosis and maintenance. The choice is not based on popularity percentages but on clarity, testability, team competence, and compliance with the site software standard.

The PLC ecosystem influences engineering tools, network integration, available safety functions, diagnostics, service competence, spare parts, and long-term support. Figure 6.3 compares three representative ecosystems without implying a universal ranking:

- Siemens (TIA Portal / SIMATIC S7-1500): integrated engineering for PLC, HMI, drives, PROFINET, and PROFIsafe; frequently selected where the site standard is Siemens-based.
- Rockwell Automation (Studio 5000 / ControlLogix): EtherNet/IP, CIP Safety, and coordinated-motion integration within the CIP ecosystem; frequently selected at Rockwell-based sites.
- Beckhoff (TwinCAT 3 / CX and industrial PCs): PC-based control, EtherCAT, distributed clocks, and modular software integration; frequently selected for high-speed and precision-motion machines.

Changing ecosystem after detailed design can cause substantial re-engineering. Selection should therefore consider the site standard, device compatibility, required motion and safety functions, engineering and maintenance competence, local support, cybersecurity capabilities, lifecycle availability, licensing, and total cost of ownership. No fixed weighting is valid for every project.

CRITERION	SIEMENS	ROCKWELL AUTOMATION	BECKHOFF
PLATFORM	SIMATIC S7-1500 TIA Portal	ControlLogix Studio 5000	CX / IPC TwinCAT 3
NATIVE NETWORK	PROFINET	EtherNet/IP	EtherCAT
SAFETY	PROFIsafe	CIP Safety	TwinSAFE / FSoE
ENGINEERING EMPHASIS	Integrated automation and diagnostics	CIP device profiles and controller-drive integration	PC-based control and distributed clocks
TYPICAL FIT	General manufacturing, automotive, infrastructure	Discrete/process plants, especially North American sites	High-speed machines, precision motion, modular equipment

Selection follows site standards, application requirements, support capability and lifecycle cost — not a universal ranking.

Figure 6.3. Representative comparison of Siemens, Rockwell Automation, and Beckhoff PLC ecosystems. The appropriate choice depends on site standards and application requirements rather than a universal performance ranking.

6.4 Robot-PLC Integration

The robot-PLC interface is a critical integration boundary. It requires a documented signal-naming convention; explicit ownership of sequence coordination; a request-acknowledge-complete handshake with timeouts and reset conditions; deterministic handling of duplicate, late, and contradictory signals; and a safety-related control path that is independent of the standard production handshake. The PLC often coordinates the cell sequence, but the terms requester/responder or sequence coordinator/device are clearer than master/slave.

WORKED EXAMPLE — I/O Mapping for a Welding Cell

The cell integrates a KUKA KR 20 robot, a Fronius TPS 500i welding source, a two-position pneumatic positioner, a light curtain, an operator start device, and a status beacon.

PLC → robot: PLC_StartReq, PLC_PartReady, PLC_ProgramNumber, and PLC_CompleteAck.

Robot → PLC: Rbt_StartAck, Rbt_Busy, Rbt_CycleComplete, Rbt_Fault, Rbt_WeldOn, and Rbt_InHome.

PLC → welding source: WeldEnable, Current_Setpoint, Voltage_Setpoint, Gas_PreFlow_Time, and recipe selection over the validated device interface.

Welding source → PLC: Welder_Ready, Welder_Fault, Actual_Current, Actual_Voltage, ArcPresent, and ArcTime.

Safety-related signals: dual-channel light-curtain OSSD, emergency-stop channels, guard interlock, and safe-stop or safe-torque-off outputs. The required PLr or SIL is determined by risk assessment and must be validated for the complete safety function.

Sequence: (1) verify permissives; (2) assert StartReq; (3) require StartAck/Busy within the validated timeout; (4) remove StartReq; (5) monitor welding and faults; (6) require CycleComplete; (7) assert CompleteAck; (8) require the robot to clear CycleComplete; and (9) clear CompleteAck. Every wait state has a defined timeout and fault transition.

WORKED EXAMPLE — Estimating I/O Count for PLC Sizing

A palletising cell has one robot (6 status inputs and 5 command outputs), an infeed conveyor (3 sensors and 1 motor), a pallet conveyor (2 sensors and 1 motor), a dual-channel light curtain, a dual-channel emergency stop, a dual-channel guard interlock, one start pushbutton, and a three-colour beacon. The HMI and networked devices are not counted as discrete I/O unless hardwired signals are also required.

Standard digital inputs: robot status 6 + conveyor sensors 3 + 2 + start pushbutton 1 = 12 DI.

Standard digital outputs: robot commands 5 + conveyor motors 2 + beacon 3 = 10 DO.

Safety channels: light curtain 2 F-DI + emergency stop 2 F-DI + guard interlock 2 F-DI = 6 F-DI; redundant final switching or drive-safe commands = 2 F-DO.

Total physical channels: 12 DI + 10 DO + 6 F-DI + 2 F-DO = 30 channels. Standard and failsafe channels must be sized separately; they cannot be combined as interchangeable “points”.

Selection: choose an F-CPU based on program size, scan time, communication load, diagnostics, and safety requirements, together with suitable standard and failsafe I/O modules. One illustrative capacity is 16 DI, 16 DO, 8 F-DI, and 4 F-DO, leaving documented headroom. Expansion allowance is project-specific and should include channels, addresses, panel space, power, network capacity, and spare terminals.

Figure 6.4 maps the standard command and status interfaces around the cell PLC and shows the safety-related control system as an independent layer. Signal direction, ownership, data type, update behaviour, timeout, default state, and terminal or network address should be recorded in the I/O and interface specification. The diagram also separates production acknowledgements from safety functions: a standard “ready” bit cannot substitute for a validated guard, emergency-stop, or safe-drive function.

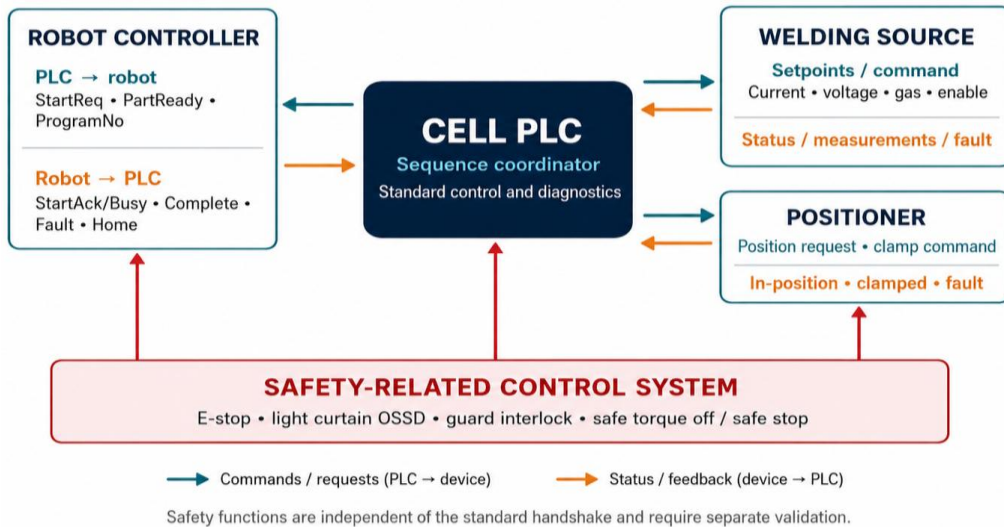


Figure 6.4. I/O and handshake map for the welding cell. Teal arrows represent commands or requests, orange arrows status or acknowledgements, and red connections the independent safety-related control path.

6.5 Cybersecurity — IEC 62443

Every connected robotic cell is part of the industrial cyber-physical attack surface. The ISA/IEC 62443 series is organised into general, policies and procedures, system, and component document groups. At system level, IEC 62443-3-2 requires definition of the system under consideration, risk assessment, partitioning into zones and conduits, and assignment of target security levels (SL-T). Security measures must therefore follow the assessed risk rather than a fixed network template. Core design principles include:

1. **Zones and conduits:** group assets with comparable security requirements into zones and permit only necessary communication through controlled conduits. A screened industrial DMZ is a common OT/IT pattern, but its exact design follows the risk assessment and operational data flows.
2. **Identity and least privilege:** authenticate users and services, remove or disable unnecessary accounts, apply role-based access, and restrict engineering access. Where device capabilities are limited, apply documented compensating controls at the network and operational layers.
3. **Secure communication and remote access:** use OPC UA Sign&Encrypt where supported, TLS for MQTT, and controlled remote access through a jump host or gateway with VPN, multi-factor authentication, approval, logging, and time

limitation. Insecure commissioning modes must not be carried into production without an explicit, risk-accepted exception.

4. Lifecycle controls: maintain an asset inventory, approved configurations, tested backup and restore procedures, vulnerability and patch processes, security monitoring, incident response, and change control. Availability and safety constraints must be considered before applying firmware or security updates.

INDUSTRIAL PRACTICE NOTE — OPC UA Security in Practice

Commissioning teams sometimes use SecurityPolicy None to simplify initial connectivity. Before production release, each required endpoint should be configured with an approved security policy and message mode, trusted certificates, controlled application identities, and a documented certificate-renewal process. Restrict endpoints through firewall allow lists and disable unused discovery or anonymous-access features. A Global Discovery Server can support certificate lifecycle management, but it must itself be placed and protected according to the zone-and-conduit design. Legacy devices that cannot support the required controls should be isolated and protected through compensating measures rather than treated as compliant by default.

WORKED EXAMPLE — Designing Network Segmentation for Robotic Cells

A factory has four robotic cells, a central SCADA and historian, engineering workstations, an MES/ERP interface, and selected cloud services. The architecture must follow IEC 62443 risk-assessment and zone-and-conduit principles.

Cell zones: place each cell, or a justified group of cells, in a zone containing its PLC, robot controller, drives, and I/O. Permit only required industrial protocols and management traffic through allow-listed conduits. Prevent direct internet access.

Supervisory zone: place SCADA, the historian, and controlled engineering services in a separate zone. Route cell-to-supervisory traffic through an industrial firewall or security gateway. Use authenticated and encrypted application protocols where supported.

Industrial DMZ: terminate OT/IT data exchange, update services, file transfer, and remote-access sessions in a DMZ. Do not allow a remote user or cloud service to connect directly to a cell controller. A one-way gateway is selected only when the consequence analysis justifies one-directional flow.

Remote access: use a managed jump host, VPN, multi-factor authentication, named accounts, approval, session logging or recording where required, and automatic expiry. Emergency access must also be documented and reviewed.

Security level: assign SL-T separately to each zone and conduit from the risk assessment. Labels such as “SL 2” or “SL 3” cannot be inferred solely from whether a zone contains a cell, SCADA, or MES.

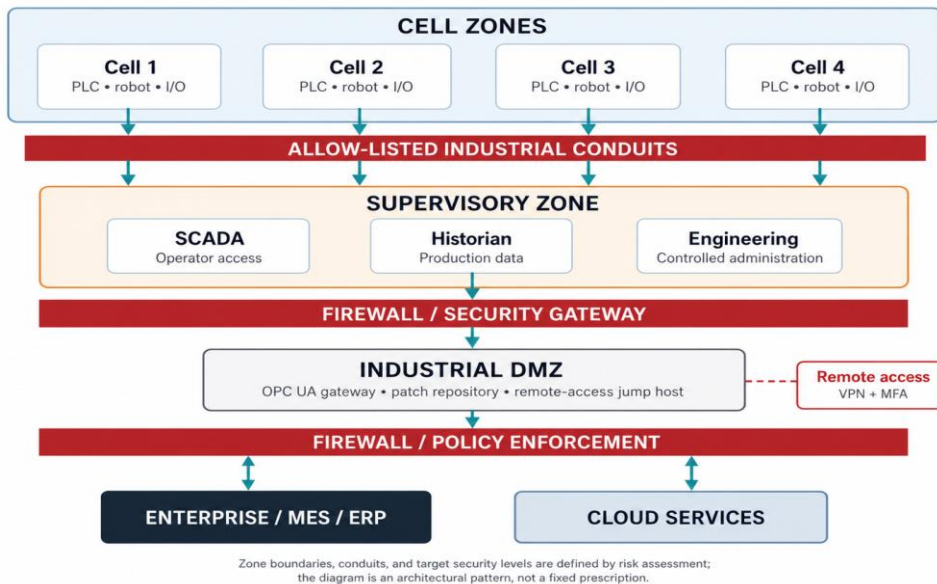


Figure 6.5. IEC 62443 zone-and-conduit pattern for four robotic cells, a supervisory zone, an industrial DMZ, enterprise systems, cloud services, and controlled remote access. Target security levels are assigned by risk assessment.

6.6 HMI and SCADA Design

The HMI is the operator’s window into the cell and part of a complete lifecycle that includes philosophy, design, implementation, verification, operation, maintenance, and continuous improvement. ISA-101 does not prescribe a single colour palette or screen

template; it requires a consistent, task-oriented interface that supports situation awareness. Alarm-system lifecycle requirements are addressed by IEC 62682 and ISA-18.2, with EEMUA 191 providing widely used engineering guidance. Key principles are:

- **Visual hierarchy:** use a neutral, low-salience base so that abnormal or actionable information is conspicuous. Colour must have consistent meaning and should not be the only coding mechanism.
- **Actionable alarms:** an alarm indicates an abnormal condition that requires a timely operator response. Status messages and events that require no action belong in an event log rather than the alarm list.
- **Task-based navigation:** provide a cell overview, area or station views, and equipment diagnostics with consistent navigation, context, and breadcrumbs. The hierarchy should match operator tasks rather than a fixed number of screens.
- **Trends and context:** show engineering units, limits, operating state, quality, and selected real-time or historical trends that help the operator diagnose the current situation. Time range and parameters should be chosen for the task.

Table 6.4 contrasts high-performance HMI practice with patterns that obscure process state or encourage alarm fatigue. The objective is not visual minimalism for its own sake, but faster recognition, diagnosis, and response. Design decisions should be documented in an HMI philosophy and verified with representative operators and abnormal scenarios.

Table 6.4. *High-performance HMI design: recommended and weak practices.*

HMI element	Recommended practice	Weak practice
Visual base	Neutral, consistent background; colour reserved for defined meanings	Decorative gradients, saturated backgrounds, inconsistent colours
Alarm presentation	Priority, consequence, response, time, state, and acknowledgement status	Flashing or colour alone, no operator response guidance
Numeric display	Readable font, engineering units, suitable precision and limits	Raw counts, excessive precision, missing units or limits
Navigation	Task-based hierarchy, consistent controls, context and breadcrumbs	Unstructured screen collection and hidden navigation

HMI element	Recommended practice	Weak practice
Trends	Selected variables, relevant time range, limits and event context	Too many traces, unsuitable scale, no reference limits

Figure 6.6 illustrates a high-performance HMI concept for a welding cell. The overview keeps normal equipment states visually quiet, while the KPI, trend, and alarm areas provide operational context. Alarm rows include priority, time, description, and a concise response. The precise colour assignments, alarm priorities, and navigation structure must be defined by the project HMI and alarm philosophies rather than copied without validation.

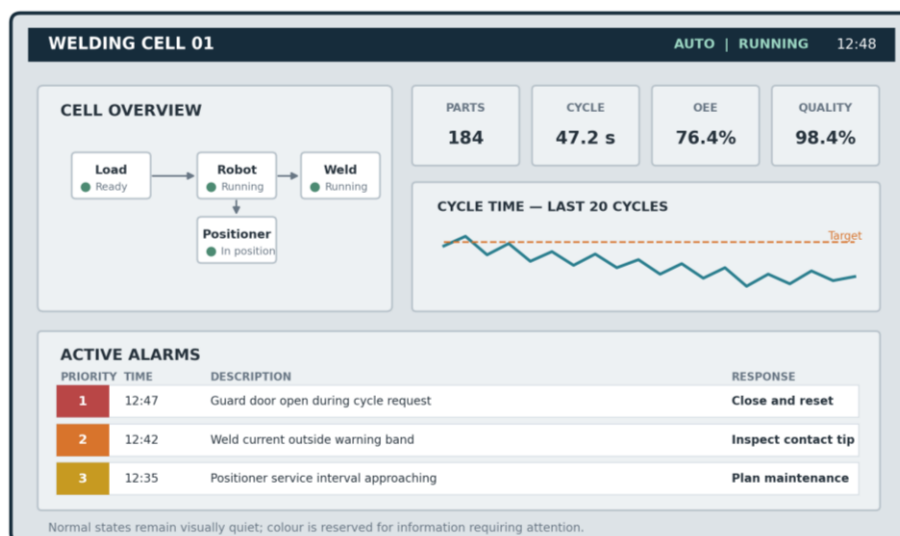


Figure 6.6. High-performance HMI concept for a welding cell, with cell overview, key performance indicators, a cycle-time trend, and prioritised alarms with operator responses.

WORKED EXAMPLE — HMI Alarm Rationalisation for a Packaging Line

Illustrative dataset: a packaging line produces 180 alarm annunciations during an eight-hour shift. Operators report alarm fatigue and frequently acknowledge alarms without reading them.

Step 1 — Analyse: export a representative alarm log and classify the annunciations. In this example, 45% (81) are informational, 30% (54) are nuisance or duplicate

conditions, 15% (27) are actionable process alarms, and 10% (18) are high-consequence alarms.

Step 2 — Rationalise: move informational messages to the event log. Remove duplicates and address nuisance-alarm causes through appropriate setpoints, deadbands, delays, state-based suppression, or maintenance. Do not suppress a genuine alarm merely to reduce the count.

Step 3 — Define each alarm: document the abnormal condition, consequence, required response, priority, setpoint, deadband, delay, shelving rules, ownership, and verification test. Priority is based on consequence and response time, not on how often an alarm occurs.

Result: the illustrative count falls from 180 to 45 annunciations per shift, or 5.6 per hour on average. This arithmetic does not by itself prove acceptable performance; the result must be assessed against the site alarm philosophy, peak ten-minute rates, standing alarms, alarm floods, operator workload, and IEC 62682/EEMUA guidance.

6.7 Integration Testing

Before factory acceptance testing (FAT), the integrated control architecture should be verified against the interface specification, safety requirements, and acceptance criteria. Testing proceeds from individual points to complete sequences and includes positive, negative, boundary, and recovery cases:

1. Point-to-point I/O test: verify every physical and mapped network signal from source to destination. Confirm address, direction, polarity, scaling, engineering units, default state, terminal or device identifier, diagnostics, and HMI indication. Record the result in an I/O test sheet.
2. Handshake and fault-injection test: execute each request-acknowledge-complete sequence step by step. Test late, missing, duplicate, contradictory, and out-of-order signals; verify that every timeout, reset, and fault transition produces the specified state and diagnostic message.
3. Safety-function validation: validate every safety function against its specification using analysis and testing in accordance with the applicable machinery-safety

standards. Measure stopping time and distance under the defined worst-case conditions, verify reset and restart behaviour, and retain objective evidence. Production speed alone is not the acceptance criterion; the validated worst case is.

4. Sequence, performance, and recovery test: run representative production cycles at controlled commissioning settings and then at the validated operating conditions. Monitor cycle-time distribution, network and controller load, missed handshakes, alarm behaviour, product quality, controlled stops, restart, power-cycle recovery, and loss-and-restoration of communications. The number and duration of cycles follow the risk and acceptance plan.

INDUSTRIAL PRACTICE NOTE — Integration Testing: Shortcuts That Create Latent Faults

Three shortcuts repeatedly create avoidable commissioning risk. (1) Skipping point-to-point verification assumes that drawings, wiring, addresses, polarity, and software mapping all agree; the test is the evidence that they do. (2) Testing only the nominal handshake proves neither timeout behaviour nor recovery from missing, duplicate, late, or contradictory signals. (3) Validating a safety function only in a convenient low-energy condition does not demonstrate the specified worst-case stopping performance. Each test should have a precondition, stimulus, expected response, acceptance criterion, measured result, evidence reference, and disposition of deviations.

WORKED EXAMPLE — Verifying Robot-PLC Handshake Timing

A welding-cell specification requires the robot to assert StartAck/Busy within 500 ms after the PLC asserts StartReq. A measured response is 450 ms. Is the margin adequate?

Timing budget: the nominal margin is $500 - 450 = 50$ ms. If measured worst-case contributions are 5 ms from network variation, 2 ms from PLC scan alignment, and 20 ms from robot-controller load, the budgeted response becomes 477 ms, leaving only 23 ms.

Assessment: the calculation demonstrates limited residual margin but does not justify a failure probability. Adequacy must be established from repeated tests under representative maximum load, startup state, communication faults, and controller scheduling conditions.

Options: (1) reduce the source latency by preloading or restructuring the robot program; (2) increase the timeout to a value supported by the process hazard and recovery requirements; and (3) improve diagnostics to distinguish acknowledgement timeout from later process faults. Increasing the timeout delays fault detection but does not lengthen a normal successful cycle.

Decision: first remove avoidable latency, then set and validate a timeout with documented margin. Do not remove the timeout: that converts a diagnosable fault into an indefinite wait state.

Figure 6.7 shows a four-phase request/acknowledge/complete exchange. The PLC raises StartReq, the robot acknowledges and becomes busy, welding is enabled during the process, and the robot raises CycleComplete. The PLC then raises CompleteAck; both parties clear their completion signals in a defined order. A 500 ms boundary applies only to the initial acknowledgement, not to the whole cycle. This return-to-zero handshake prevents a stale high signal from being mistaken for a new event.

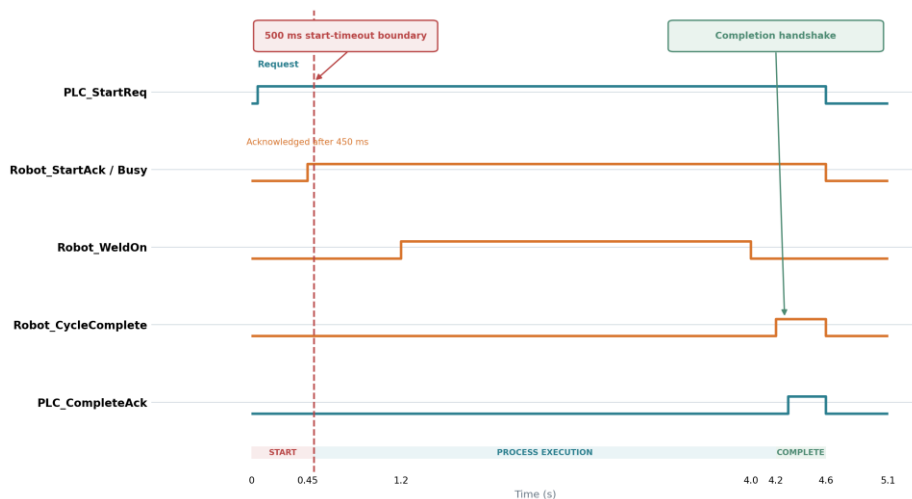


Figure 6.7. Timing diagram for a robot-PLC request/acknowledge/complete handshake, including a 500 ms start-acknowledgement timeout and a completion reset sequence.

Common Pitfalls

Pitfall 1: Ambiguous sequence ownership.

If the PLC and robot can both initiate the same state transition without a defined protocol, race conditions and deadlocks can occur. Identify the sequence coordinator, requester, responder, permitted transitions, and reset authority in the interface specification.

Pitfall 2: Leaving insecure OPC UA endpoints enabled.

SecurityPolicy None, anonymous access, broad trust lists, and unmanaged certificates may be useful during isolated laboratory setup but should not remain enabled in production unless a documented risk assessment accepts the exposure and compensating controls are implemented.

Pitfall 3: Omitting handshake timeouts.

Every wait state needs a timeout, a diagnostic cause, a safe or controlled fault transition, and a defined recovery path. The timeout value must be derived and validated rather than selected arbitrarily.

Pitfall 4: Designing the HMI as a decorative dashboard.

Saturated backgrounds, animation, and inconsistent colour coding compete with abnormal information. Use a task-oriented visual hierarchy and verify that operators can detect, diagnose, and respond to representative abnormal conditions.

Pitfall 5: Skipping point-to-point I/O verification.

A correct drawing does not prove correct wiring, addressing, polarity, scaling, software mapping, or HMI indication. Point-to-point testing provides traceable evidence for the implemented system.

Pitfall 6: Validating safety only under convenient conditions.

Safety validation must cover the specified worst-case conditions, measured stopping performance, reset and restart behaviour, fault exclusions, and diagnostic coverage—not merely a reduced-speed demonstration.

Pitfall 7: Uncontrolled cross-level communication.

Direct data paths may be technically possible, but they should not bypass architecture ownership, cybersecurity controls, semantic mediation, rate limiting, or change management. Use engineered gateways and documented conduits.

Pitfall 8: Selecting network infrastructure by office-IT criteria alone.

Industrial switches and media should be selected from the environmental, availability, timing, diagnostics, redundancy, conformance, and cybersecurity requirements. Managed features are needed where VLANs, QoS, synchronisation, monitoring, or access control are part of the design.

Pitfall 9: Failing to maintain the I/O and interface specification.

For each signal or data item, record its name, owner, direction, address, data type, units, normal and fault state, update behaviour, timeout, device, cable or network reference, and functional description. Keep the document aligned with the as-built software.

Pitfall 10: Treating cybersecurity as a commissioning add-on.

Define the system under consideration, zones, conduits, target security levels, remote-access concept, account model, logging, backup, patching, and incident response during requirements and design. Late retrofit usually constrains both security and availability.

KEY TAKEAWAYS

- ISA-95 functional levels and Purdue-style architecture are related but not identical; levels define responsibilities, while cross-level data paths require controlled interfaces.
- PROFINET, EtherCAT, and EtherNet/IP must be compared as complete engineered systems, not by a single nominal cycle-time or installed-base ranking.
- OPC UA provides secure semantic integration; MQTT provides brokered publish-subscribe telemetry. Neither replaces a deterministic motion network or a validated safety function.
- IEC 61131-3:2025 specifies ST, LD, and FBD, with SFC sequencing elements. IL is retained only in legacy projects.

- Robot-PLC integration requires explicit sequence ownership, request/acknowledge/complete handshakes, return-to-zero behaviour, timeouts, and documented I/O mapping.
- Standard production signals and safety-related signals have different integrity requirements and must be designed and validated separately.
- IEC 62443 uses risk assessment, zones, conduits, target security levels, least privilege, secure remote access, and lifecycle controls.
- ISA-101 addresses the HMI lifecycle and situation awareness; IEC 62682/ISA-18.2 and EEMUA guidance address alarm management and rationalisation.
- Integration testing progresses from point-to-point verification to fault-injected handshakes, safety validation, complete sequences, performance, and recovery.
- PLC and network capacity should include documented project-specific headroom across I/O channels, addresses, panel space, processing, bandwidth, and lifecycle needs.

Review Questions

Answer the following questions to consolidate your understanding. Where numerical values are required, state assumptions and distinguish nominal values from validated worst-case performance.

1. Draw the five-level functional hierarchy and explain how ISA-95 differs from a Purdue-style architecture. Which functions normally require the strictest timing, and why?
2. Compare PROFINET IRT, EtherCAT, and EtherNet/IP for a coordinated welding cell. Define the timing, ecosystem, safety, topology, diagnostics, and validation criteria before recommending a technology.
3. Explain the distinct roles of an Industrial Ethernet control network, OPC UA, and MQTT. Design an architecture that supports deterministic cell control, semantic machine integration, and cloud analytics.
4. Recalculate the palletising-cell I/O example when the guard interlock and emergency stop are dual-channel. Separate standard DI/DO from F-DI/F-DO and propose suitable capacity headroom.
5. Design an SFC or state-machine cycle for a machine-tending cell: operator loads part → guard closes → robot loads CNC → CNC machines → robot unloads → guard

opens. Include request, acknowledgement, completion, reset, timeout, and fault-recovery signals.

6. A start acknowledgement is measured at 480 ms against a 500 ms timeout. Construct a timing budget, identify the evidence needed to validate the margin, and explain how changing the timeout affects normal cycle time and fault-detection time.
7. Redesign a packaging-line HMI with saturated backgrounds and 200 alarm annunciations per shift. Separate HMI visual-hierarchy actions from alarm-lifecycle actions and define how the redesign will be verified with operators.
8. Define the system under consideration, zones, conduits, target security levels, and remote-access controls for a factory with six robotic cells, central SCADA, MES/ERP integration, and selected cloud services.
9. Prepare an integration-test matrix for one robot-PLC handshake and one guard-interlocking safety function. Include nominal, boundary, negative, fault-injection, reset, power-cycle, and communication-recovery cases.

References

- [1] Groover, M.P. (2020) Automation, Production Systems, and Computer-Integrated Manufacturing, 5th ed. Pearson.
- [2] ANSI/ISA-95.00.01-2025, Enterprise-Control System Integration — Part 1: Models and Terminology. ISA, Research Triangle Park, NC.
- [3] IEC 61131-3:2025, Programmable controllers — Part 3: Programming languages. IEC, Geneva.
- [4] IEC 62541-1:2020, OPC Unified Architecture — Part 1: Overview and concepts. IEC, Geneva.
- [5] OPC Foundation and VDMA (2025) OPC 40010-1, Robotics Part 1: Vertical Integration, Version 1.02. OPC Foundation.
- [6] OASIS (2019) MQTT Version 5.0. OASIS Standard.
- [7] IEC 62443-3-2:2020, Security for industrial automation and control systems — Part 3-2: Security risk assessment for system design. IEC, Geneva.
- [8] IEC 62443-3-3:2013, Industrial communication networks — Network and system security — Part 3-3: System security requirements and security levels. IEC, Geneva.
- [9] ANSI/ISA-101.01-2015, Human Machine Interfaces for Process Automation Systems. ISA, Research Triangle Park, NC.
- [10] IEC 62682:2022, Management of alarm systems for the process industries. IEC, Geneva.
- [11] EEMUA 191 (2024) Alarm Systems: A Guide to Design, Management and Procurement, 4th ed. EEMUA, London.
- [12] PROFIBUS & PROFINET International (2026) PROFINET Technology — The Easy Way. PI.
- [13] EtherCAT Technology Group (ETG) (2025) EtherCAT Technology and Application Information. ETG, Nuremberg.

- [14]** ODVA (2020) The Common Industrial Protocol (CIP) and the Family of CIP Networks. ODVA, Ann Arbor, MI.
- [15]** ISO 13849-1:2023, Safety of machinery — Safety-related parts of control systems — Part 1: General principles for design. ISO, Geneva.
- [16]** ISO 13849-2:2012, Safety of machinery — Safety-related parts of control systems — Part 2: Validation. ISO, Geneva.
- [17]** Mocan, B. (2025) Design and Operation of Robotic Production Systems — Course Notes. Technical University of Cluj-Napoca.

CHAPTER

7

Simulation and Digital Validation of Robotic Systems

Catching design errors in the engineering office, not on the factory floor

The focus of this chapter is on validating design decisions before physical implementation. Simulation is presented as a powerful engineering tool for evaluating cycle times, detecting bottlenecks, analysing resource utilisation, and identifying potential collisions. Rather than emphasising software operation, the chapter stresses simulation as a **risk-reduction and decision-support mechanism**.

Students learn how iterative simulation-based validation contributes to cost reduction, shorter commissioning times, and improved system robustness. The chapter concludes with a candid discussion of simulation limitations — because **the map is not the territory**, and engineers who confuse the two create overconfident designs that fail at commissioning.

LEARNING OBJECTIVES

1. Articulate the economic case for simulation-based validation (typical 30–50% commissioning-issue shift-left).
2. Select appropriate simulation techniques (reachability, collision, motion, cycle-time, virtual commissioning) for each design phase.
3. Distinguish online programming, offline programming (OLP), and digital-twin approaches and select the right one for a given context.
4. Evaluate commercial simulation platforms (RoboDK, Visual Components, Tecnomatix, RobotStudio, ROBOGUIDE) by capability, cost, and ecosystem fit.
5. Classify digital-twin maturity across four levels (Model, Shadow, Twin, Intelligent Twin) and identify the investment required for each.
6. Investigate and explain simulation-vs-reality discrepancies using systematic root-cause analysis.
7. Configure collision-check thresholds and proximity warnings that account for real-world cable dress, thermal deformation, and wear.
8. Build a simulation model following the structured eight-stage workflow: define scope → import CAD → define kinematics → program paths and logic → simulate and validate → optimise → prepare for virtual commissioning → export and hand over.

9. Integrate simulation activities into the project lifecycle at concept, detailed design, and pre-commissioning stages.
10. Recognise the six common simulation pitfalls and implement mitigation strategies for each.

7.1 The Case for Simulation

Simulation-based validation shifts 30–50% of commissioning issues from the factory floor to the engineering office, where they cost 3–10× less to resolve. The economic case is straightforward: every hour of simulation time avoids 5–10 hours of on-site rework. For a typical 6-robot cell, simulation investment of €15–25K returns €60–150K in avoided commissioning delays.

Beyond cost avoidance, simulation provides three strategic benefits (table 7.1): (1) **design exploration** — evaluating multiple layout and sequence alternatives without building any of them; (2) **stakeholder alignment** — visualising the cell for non-technical decision-makers; (3) **training** — operators can learn the cell before it exists physically.

Table 7.1. Simulation benefits with typical magnitudes and timing.

Simulation Benefit	Typical Magnitude	When Realised
Commissioning time reduction	20–40%	During SAT/ramp-up
Design error detection	30–50% of issues caught early	During detailed design
Cycle-time optimisation	5–15% improvement	Before FAT
Operator training head start	2–4 weeks earlier readiness	Before installation
Layout alternative evaluation	3–5 candidates tested in days	During concept design

Figure 7.1 illustrates the **cost-of-change curve**, a fundamental principle in project management that quantifies the financial impact of error detection timing. The diagram maps the project lifecycle—from initial concept through to full-scale production—against the cost to rectify mistakes, plotted on a **logarithmic scale**. By identifying errors during the "design" phase via advanced simulation, engineers can address issues when the cost of intervention is **10–100× lower** than during Site Acceptance Testing (SAT) or live production. This visual representation underscores the massive ROI of "front-loading"

the engineering process, proving that early-stage virtual validation is not merely a technical convenience but a critical strategy for mitigating exponential financial risk.

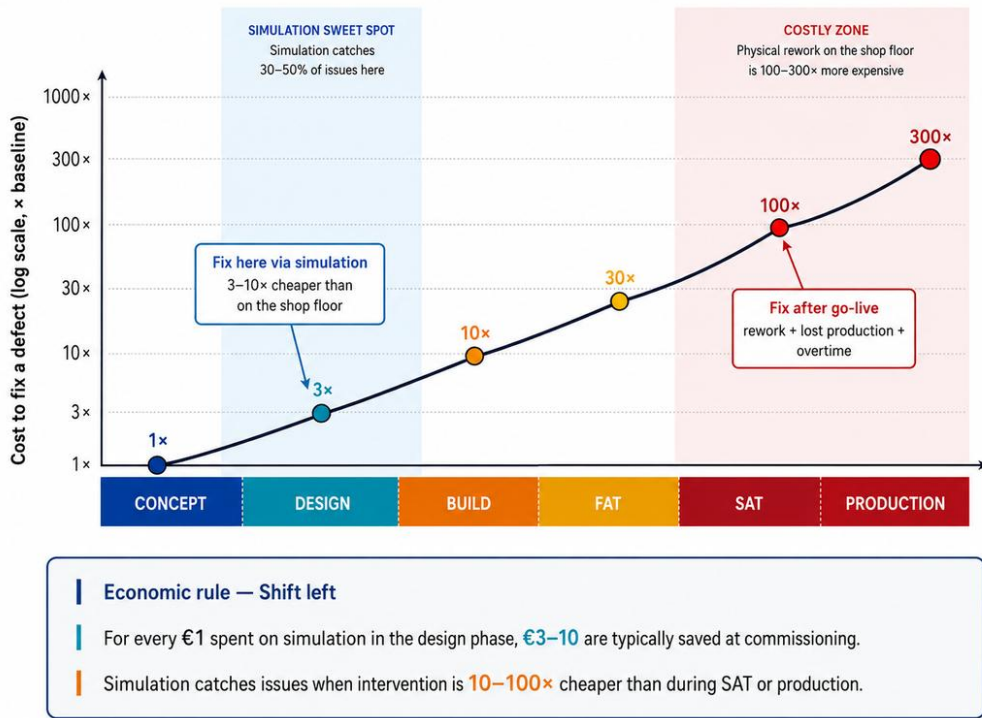


Figure 7.1. Cost-of-change curve: X-axis shows project phases (concept, design, build, FAT, SAT, production). Y-axis shows cost to fix an error (log scale). Simulation catches errors in the ‘design’ phase where fix cost is 10–100× lower than at SAT/production.

7.2 Simulation Techniques for Robotic Systems

Five simulation techniques serve different purposes at different design stages (table 7.2):

Table 7.2. Simulation techniques ranked by purpose, phase, accuracy, and effort.

Technique	Purpose	Design Phase	Accuracy	Effort
Reachability analysis	Can the robot reach all required points?	Concept	±5 mm	Hours
Collision detection	Are paths free of interference?	Detailed design	±1 mm (CAD)	Hours–days
Cycle-time estimation	Does the cycle meet takt time?	Detailed design	±5–10%	Days

Technique	Purpose	Design Phase	Accuracy	Effort
Process simulation	Full sequence with logic, sensors, I/O	Pre-FAT	$\pm 3\text{--}5\%$	Days–weeks
Virtual commissioning	Real controllers + virtual plant	Pre-FAT	$\pm 1\text{--}3\%$	Weeks

7.2.1 Reachability Analysis

The first simulation check: can the robot’s TCP reach every required position with an acceptable approach angle? Reachability failures at concept stage are easy to fix (relocate robot base, choose longer-reach model). At commissioning, they are catastrophic (the robot physically cannot do its job). Rule: verify reachability at ALL positions including worst-case workpiece orientations and edge-of-pallet locations, not just the nominal pick/place points.

7.2.2 Collision Detection

Collision detection verifies that the robot’s full trajectory — including the EoAT, workpiece, cables, and any mounted accessories — is free of interference with the cell’s static and dynamic elements. Three levels: (1) robot-to-fixture, (2) robot-to-robot (in multi-robot cells), (3) cable/hose sweep envelope (the least-modelled and most-frequently-violated).

INDUSTRIAL PRACTICE NOTE — Collision-Check Thresholds That Work

Industrial practice: set WARN at 20 mm, HARD at 10 mm. At commissioning, expect real trajectories within 5–15 mm of obstacles — simulation is optimistic about cable dress, wear, and thermal deformation. A path at 8 mm simulated clearance will almost certainly scrape something in production. Conservative design adds 10–15 mm margin BEYOND the warning, especially near cables and maintenance-accessed fixtures. Re-verify after EoAT changes — a 2 kg sensor addition shifts the reach envelope subtly.

7.2.3 Cycle-Time Estimation

Cycle-time simulation uses the robot's kinematic model and motion-planning algorithm to estimate how long each trajectory segment takes. Three sources of discrepancy between simulated and real cycle times are systematic: (1) **settling time** at precise-stop points (simulation often assumes zero); (2) **acceleration derating** (operators reduce to 80–90% to reduce fixture wear); (3) **I/O wait states** (sensor response, valve actuation) are often idealised in simulation.

WORKED EXAMPLE — Investigating Simulation-vs-Reality Cycle-Time Discrepancy

A spot-welding cell simulated at 58.2 s in Tecnomatix. Commissioning measured 61.8 s over 500 cycles (+6.2%). Investigate.

Source 1 — Wait states: Simulation assumed 0.3 s fixture-open; actual cylinder 0.7 s (under-specified regulator). $0.4 \text{ s} \times 7 \text{ fixtures} = 2.8 \text{ s}$.

Source 2 — TCP settling: Simulation used zero settling at precise-stop; actual 0.08 s for $\pm 0.2 \text{ mm}$. $14 \text{ stops} \times 0.08 \text{ s} = 1.12 \text{ s}$.

Source 3 — Acceleration margin: Simulation at 100%; production at 90% (operator-adjusted). $\approx 0.3 \text{ s}$.

Sum: $2.8 + 1.12 + 0.3 = 4.22 \text{ s}$. Matches 3.6 s residual within 0.6 s uncertainty (stochastic).

Lesson: 5–10% sim-reality gap is normal. Systematic investigation explains it. Unexplained >10% gaps warn of deeper modelling issues. After commissioning, **calibrate the simulation to match reality** — it then becomes a reliable baseline for future what-if analysis.

7.3 Offline Programming vs. Online Programming

The choice between **Offline Programming (OLP)** and **Online Programming** represents a fundamental decision in robotic systems engineering, balancing the immediate feedback of the physical world against the efficiency of a virtual environment. While **Online Programming**—typically performed via a teach pendant—allows an operator to interact directly with the robot and its environment to account for real-world nuances, it requires the robotic cell to be taken out of production, leading to significant downtime.

Conversely, **OLP** leverages digital twins and CAD data to generate complex paths without interrupting the production flow, significantly reducing commissioning times.

This comparison is not merely a matter of convenience but a strategic trade-off involving cycle time optimization, safety, and cost (table 7.3). Online methods remain indispensable for fine-tuning and small-batch adjustments where high precision relative to a physical part is required. However, for large-scale operations or complex geometries like those found in aerospace or automotive welding, OLP provides a safer, more scalable solution by allowing for collision detection and reachability analysis long before the first physical component is even manufactured.

Table 7.3. Online vs. offline programming comparison.

Criterion	Online (Teach Pendant)	Offline (OLP)
Where	On the physical robot	On a PC, in simulation
Robot availability	Robot must be available (production stops)	Robot continues production
Accuracy	High (taught on real hardware)	Depends on model calibration
Speed of programming	Slow (hours per program)	Fast (minutes, if model is calibrated)
Complex paths	Difficult (point-by-point)	Easy (CAD-based path generation)
Cost	Low (no software licence)	Medium (€5–25K/year licence)
Best for	Simple cells, one-off tasks	Complex cells, frequent reprogramming

For robotic production systems, OLP is the standard for any cell with >10 programmed positions or >2 product variants. Online teaching remains practical for simple single-program cells where the robot is available during programming (e.g., new installation commissioning).

INDUSTRIAL PRACTICE NOTE — OLP Accuracy: Calibration Is Non-Negotiable

OLP programs generated from CAD models assume perfect robot kinematics and perfect cell geometry. Reality differs: robot base position error ± 2 mm, fixture

position error ± 1 mm, robot kinematic errors ± 0.5 mm, thermal expansion ± 0.2 mm. Combined, the OLP path may miss the target by 3–5 mm — catastrophic for precision tasks. Solution: (1) **ROBOT CALIBRATION** (€3–8K) reduces kinematic errors to ± 0.1 mm. (2) **FRAME CALIBRATION** (touching reference points on fixtures) aligns the simulation frame to the physical frame. (3) **PRODUCTION TEST** (10–20 cycles at reduced speed) validates before full-speed operation. Budget 4–8 hours for calibration per cell; never skip it.

Figure 7.2 delineates the structured **OLP (Offline Programming) workflow**, a multi-stage process that bridges the gap between digital design and physical execution.

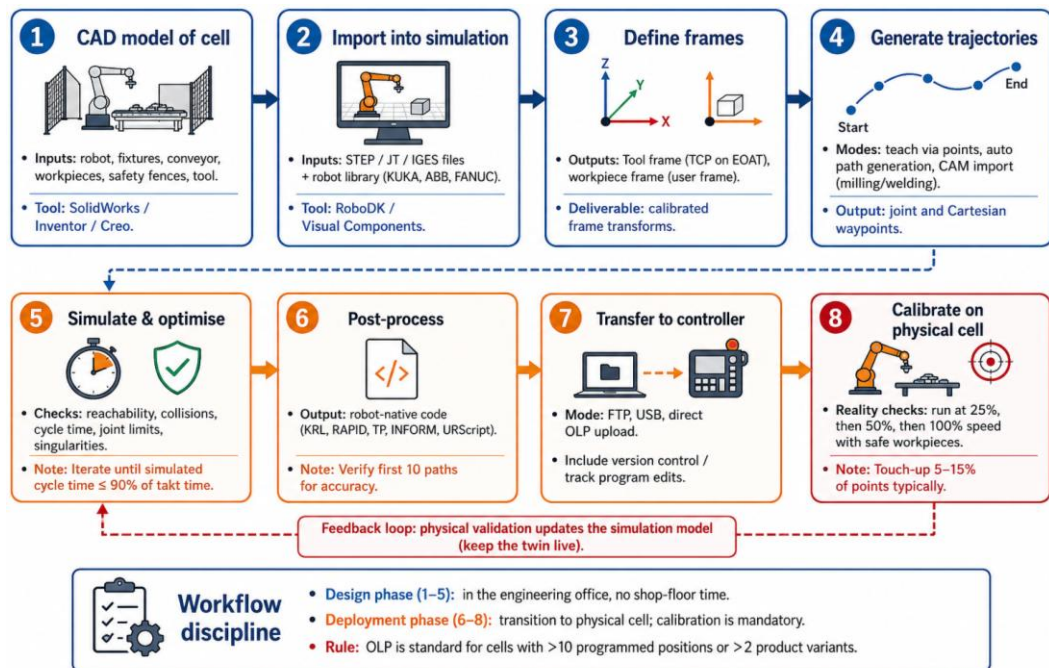


Figure 7.2. OLP workflow: CAD model of cell → Import into simulation (RoboDK/Visual Components) → Define tool frame and workpiece frames → Generate trajectories → Simulate and optimise → Post-process to robot-native code → Transfer to robot controller → Calibrate on physical cell.

The cycle begins with the creation of a high-fidelity **CAD model** of the cell, which is imported into specialized simulation software such as RoboDK or Visual Components. By defining precise **tool and workpiece frames**, engineers can generate complex trajectories

and perform iterative optimization in a virtual space, ensuring collision-free movement and cycle-time efficiency. Once validated, the simulation is converted via a **post-processor** into robot-native code (e.g., RAPID, KRL). The final, critical phase involves transferring this code to the physical controller and performing a **cell calibration**, which synchronizes the virtual coordinates with the real-world environment to compensate for minor mechanical discrepancies.

WORKED EXAMPLE — Deciding Between OLP and Online Programming

A machine-tending cell with 2 CNC lathes needs programs for 8 part variants. Each variant requires ~30 programmed points. The robot produces during 2 shifts/day. Compare OLP vs. online teaching.

Online teaching: 8 variants $\times$ 30 points $\times$ 3 min/point = 720 min = 12 hours of robot downtime. At 2 shifts/day, this requires 1.5 shifts of production loss (€5,400 lost output at €300/hour margin). Plus the programmer's time: 12 h $\times$ €50/h = €600. Total: €6,000.

OLP: RoboDK licence €5,000/year. Programming time: 8 variants $\times$ 30 points $\times$ 1 min/point = 240 min = 4 hours (€200 labour). Calibration: 4 hours (€200 + €0 production loss — done during planned maintenance). Transfer + validation: 2 hours (€100). Year 1 total: €5,500.

Break-even: OLP costs €500 less in year 1 AND avoids 1.5 shifts of production loss. In year 2+, OLP costs only €5,000 (licence) + €500 (labour for new variants) = €5,500 vs. €6,000+ for online. OLP dominates for any cell with >4 variants.

7.4 Commercial Simulation Platforms

Platform selection follows the engineering team's ecosystem (table 7.4): if the site uses ABB robots exclusively, RobotStudio's VirtualController provides the highest fidelity at the lowest cost. For mixed-brand sites or vendor-agnostic design, RoboDK (budget) or Visual Components (mid-range) are the standard choices. Tecnomatix is justified for enterprise-scale projects with virtual commissioning requirements.

Table 7.4. Commercial simulation platforms comparison.

Platform	Type	Robot Brands	Licence Cost	Strengths
RoboDK	Vendor-agnostic OLP	All major (500+ models)	€3–5K/year	Low cost, easy learning curve, Python API
Visual Components	Vendor-agnostic DES	All major	€8–15K/year	Layout + process simulation, 3D library
Tecnomatix (Siemens)	Enterprise DES + VC	All major	€15–40K/year	Full virtual commissioning, MES integration
RobotStudio (ABB)	Vendor-specific OLP	ABB only	€2–5K/year	True controller emulation (VirtualController)
ROBOGUIDE (FANUC)	Vendor-specific OLP	FANUC only	€3–6K/year	Accurate FANUC kinematics, HandlingTool
Gazebo/ROS2	Open-source research	Custom + some industrial	Free	Research, AI training, custom robots

Figure 7.3 presents a **platform-selection framework** organised around controller fidelity, robot-brand coverage, process and material-flow scope, virtual-commissioning requirements, cost, and available expertise. Vendor-specific platforms are preferred when accurate emulation of a particular controller is critical; multi-brand OLP platforms suit mixed fleets and path generation; process and digital-manufacturing platforms are justified when cell-, line-, or factory-level validation and PLC integration dominate. Research or highly customised applications may require open-source or bespoke modelling. The selected platform should then be piloted on a representative scenario and validated against real-system data before full deployment.

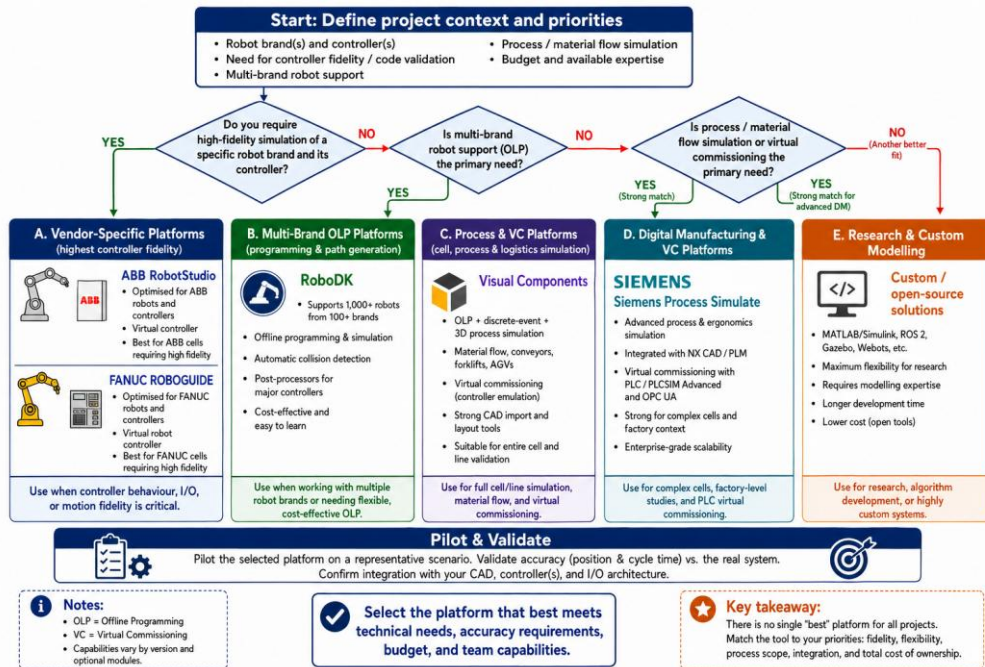


Figure 7.3. Platform-selection framework linking controller fidelity, multi-brand OLP, process simulation, virtual commissioning, digital manufacturing, and custom modelling to project requirements, followed by pilot validation.

7.5 The Digital Twin

The concept of the digital twin extends simulation from a design-phase tool to an operational-phase infrastructure. Four levels of maturity are commonly distinguished (table 7.5):

Table 7.5. Four levels of digital-twin maturity.

Level	Name	Data Flow	Update	Typical Use
1	Digital Model	None (manual)	On demand	Training, offline what-if
2	Digital Shadow	Physical → Virtual	Seconds–minutes	Monitoring, dashboards, post-hoc
3	Digital Twin	Bidirectional	Real-time	Adaptive control, optimisation
4	Intelligent Twin	Bidirectional + AI	Real-time + predictive	Self-optimising production

Most industrial practice in 2025 operates at Level 2 (digital shadow) with aspirations toward Level 3. Level 4 (intelligent twin with AI-driven self-optimisation) exists in pilot installations but is not yet production-standard (figure 7.4). The investment required increases roughly 3× per level: Level 1 costs €5–15K (simulation software); Level 2 adds €15–40K (sensors, gateway, cloud); Level 3 adds €40–120K (real-time infrastructure, control feedback); Level 4 adds €100K+ (AI development, validation).

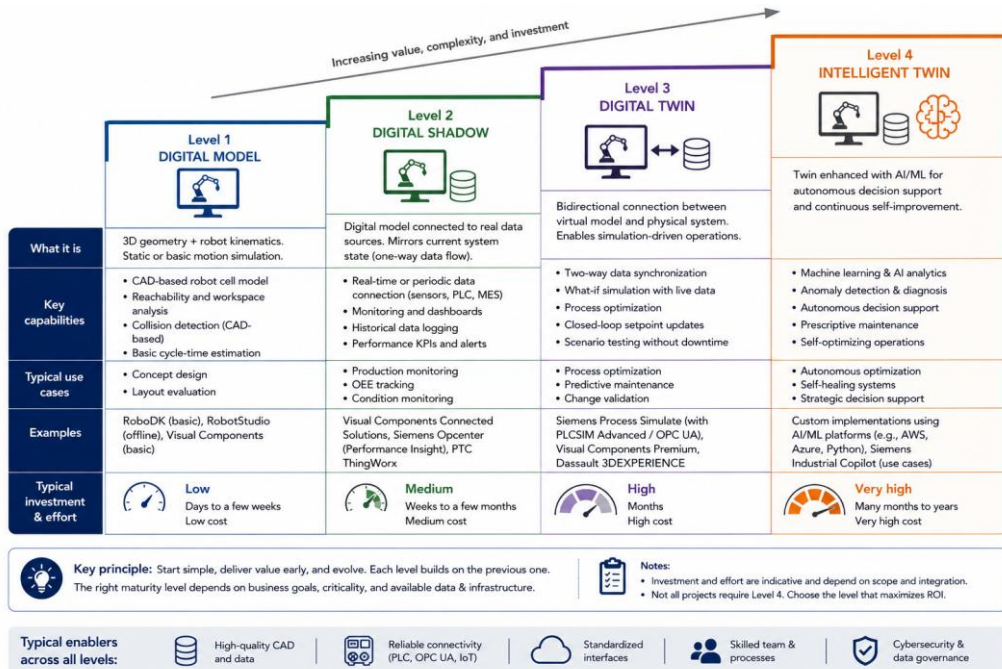


Figure 7.4. Digital twin maturity staircase: four levels from Digital Model (static) to Intelligent Twin (AI-driven). Each step shows the required investment, data infrastructure, and typical industrial examples.

WORKED EXAMPLE — Assessing Digital-Twin Maturity of a Welding Cell

A welding cell has a Tecnomatix Process Simulate model created during design. After commissioning, the model was not updated. Weld parameters are logged to a historian but not connected to the model. Assess the digital-twin level and propose an upgrade path.

Current level: 1 (Digital Model). The simulation model exists but has no live data connection. It was useful during design but is now stale — any changes made during commissioning (path adjustments, timing tweaks) are not reflected.

Upgrade to Level 2 (Digital Shadow): Connect the historian to the simulation model via OPC UA. Update the model with as-built robot paths. Display live OEE, cycle time, and weld-quality metrics on the simulation dashboard. Cost: €12K (OPC UA gateway + model update + dashboard). Timeline: 3 weeks.

Upgrade to Level 3 (Digital Twin): Add bidirectional control: the twin proposes optimised weld parameters (speed, current) based on real-time material-thickness variation; the PLC executes them. Requires: real-time sensor for material thickness (€8K), parameter-optimisation algorithm (€25K development), safety validation of automatic parameter changes (€15K). Total: €48K + 4 months. ROI depends on defect-rate reduction.

7.6 Simulation Limitations: The Map Is Not the Territory

Six systematic sources of sim-reality discrepancy must be understood and budgeted:

- 1. Kinematic model errors:** joint backlash, gear compliance, thermal expansion change the actual TCP position by 0.1–0.5 mm vs. the ideal kinematic model.
- 2. Dynamic model simplifications:** simulation often uses trapezoidal velocity profiles when the real controller uses S-curve; settling time at fine positions is frequently zero in simulation but 50–200 ms in reality.
- 3. Cable and hose routing:** the most common unmodelled element. Cable dress changes the effective reach envelope by 5–15 mm and creates collision risks that pure-kinematic simulation misses.
- 4. Sensor response times:** simulation often treats sensors as instantaneous; real sensors have 1–50 ms response depending on type.
- 5. Operator behaviour:** simulation assumes deterministic operator timing; real operators vary by ± 20 –40% on manual tasks. Buffer sizing and cycle-time margins must account for this.

- 6. Environmental factors:** temperature affects robot repeatability (± 0.01 mm/ $^{\circ}\text{C}$ typical); lighting affects vision systems; vibration from adjacent equipment affects precision tasks.

WORKED EXAMPLE — Budgeting the Simulation-Reality Gap

A cell simulates at 42.0 s cycle time. The takt time is 45.0 s. Is there sufficient margin?

Expected gap: 5–10% is normal. At 42.0 s, expect real CT of 44.1–46.2 s.

Risk assessment: The upper bound (46.2 s) exceeds TT (45.0 s). There is a ~40% probability that the cell will NOT meet takt without optimisation after commissioning.

Decision rule: Simulated CT should be $\leq 90\%$ of TT to provide a comfortable margin. Here: $42.0/45.0 = 93\%$ — above the 90% threshold. The cell is at risk. **Action:** optimise the simulation to reduce CT to 40.5 s (90% of TT) before proceeding to build. Alternatively, accept the risk and budget 2 weeks of cycle-time optimisation during ramp-up.

7.7 Building a Simulation Model: Workflow

A structured workflow prevents the most common modelling errors:

- 1. Define scope and objectives:** define the use case, boundaries, success criteria, KPIs, accuracy requirements, and assumptions.
- 2. Import CAD and build the environment:** import and simplify cell geometry; verify scale, layout, and coordinate alignment.
- 3. Define kinematics and components:** select robot and controller models; define joints, payload, TCP, tools, and peripherals.
- 4. Program paths and logic:** create trajectories, I/O signals, sensor events, and the cell sequence.
- 5. Simulate and validate:** check reachability, collisions, clearances, approach angles, cycle time, and KPIs.

6. Optimise and iterate: refine paths, sequence, layout, tools, speeds, and resources until targets are met.

7. Prepare for virtual commissioning: connect the controller; test programs, I/O, cell logic, safety functions, and interlocks.

8. Export and hand over: generate native code and documentation; export engineering outputs and archive the controlled model version.

Figure 7.5 details the **simulation model-building workflow**, an eight-stage engineering cycle from scope definition and CAD preparation through virtual commissioning to export and controlled handover. Each stage produces reusable engineering evidence, while operational data closes the loop for model update and re-validation.

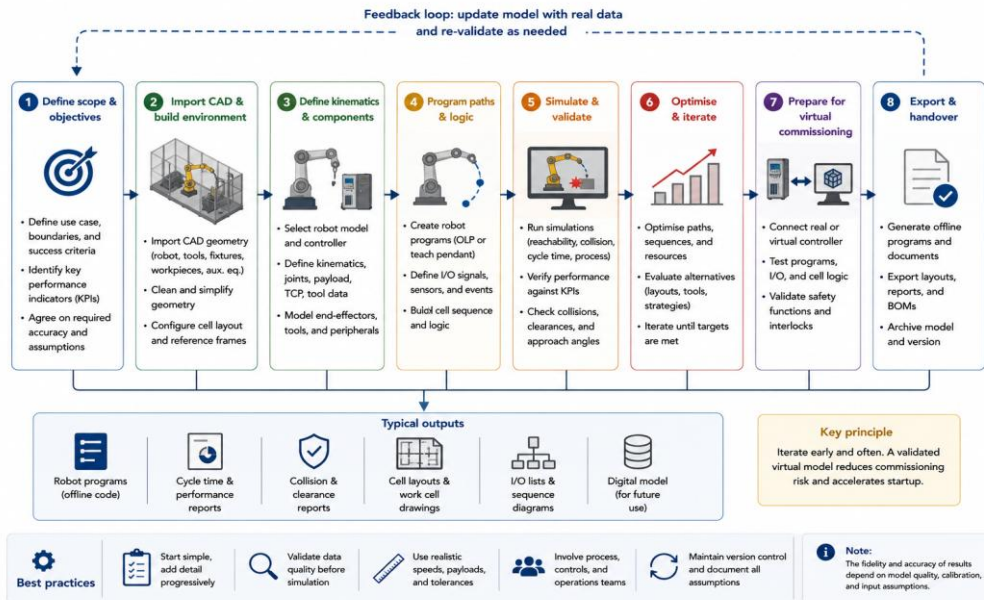


Figure 7.5. Eight-stage simulation model-building workflow from scope definition to export and handover, including typical outputs and a closed feedback loop for model update and re-validation.

A critical feature of this workflow is the **closed feedback loop** from handover and operational evidence back to scope definition and model development. Results from simulation, virtual commissioning, and reduced-speed physical validation are used to update assumptions, coordinate frames, kinematic parameters, I/O logic, cycle-time data, and controlled model versions. This maintains alignment between the virtual

representation and the as-built, as-operated system and preserves the model as a reliable asset for future modifications.

WORKED EXAMPLE — Selecting a Simulation Platform for a Mixed-Brand Cell

A system integrator designs a cell with 2 ABB IRB 6700 robots and 1 FANUC CRX-25iA cobot. The cell includes 3 conveyors, a vision station, and safety fences. The customer requires virtual commissioning with PLC-in-the-loop. Select the platform.

Option A — RobotStudio: Excellent for ABB robots (VirtualController), but cannot simulate the FANUC CRX. Eliminated.

Option B — RoboDK: Supports both ABB and FANUC. Good OLP. But virtual commissioning (PLC-in-the-loop) is limited — no native PLCSIM integration. Cost: €5K/year.

Option C — Visual Components: Supports both brands. Has PLC connectivity via OPC UA. Virtual commissioning possible with external PLCSIM Advanced. Cost: €12K/year.

Option D — Tecnomatix Process Simulate: Full multi-brand support + native PLCSIM Advanced integration + virtual commissioning. Cost: €25K/year.

Decision: Option C (Visual Components) provides the best balance of capability and cost. Tecnomatix is over-specified (and 2× the price) for a single-cell project. RoboDK lacks the virtual commissioning capability the customer requires. **Key principle:** platform selection follows project requirements, not brand loyalty.

INDUSTRIAL PRACTICE NOTE — Simulation Investment Rules of Thumb

Three rules for budgeting simulation effort: (1) SIMULATION COST should be 3–5% of total cell project cost. Below 3%, the model will be too crude to catch real issues. Above 5%, diminishing returns set in. (2) MODELLING TIME scales as N^2 with number of interacting devices: 1 robot = 2 days, 2 robots = 5 days, 4 robots = 12 days, 8 robots = 30 days. This is because interaction verification (collision between

robots, handshake timing) grows quadratically. (3) CALIBRATION TIME = 0.5 days per robot + 0.5 days per major fixture. Budget this separately from modelling; it happens AFTER physical installation and BEFORE production programming.

Common Pitfalls

Pitfall 1: Trusting simulation cycle time as absolute.

5–10% sim-reality gap is normal. Present simulated CT with uncertainty: ‘42.0 s $\pm$ 4 s’, not ‘42.0 s’.

Pitfall 2: Skipping cable and hose modelling.

Cable dress is the #1 unmodelled element and the #1 source of unexpected collisions at commissioning. Model at least the major cable runs.

Pitfall 3: Using a stale simulation model.

A model created during design and never updated during commissioning is worse than no model — it creates false confidence. Update after every significant physical change.

Pitfall 4: Over-investing in visual fidelity.

Photorealistic rendering impresses stakeholders but does not improve engineering accuracy. Invest in kinematic fidelity (correct joint limits, payload derating) rather than visual polish.

Pitfall 5: Simulating at ideal conditions only.

Simulation at 100% speed, perfect parts, zero variance produces optimistic results. Always run worst-case scenarios: maximum payload, slowest part variant, degraded sensor response.

Pitfall 6: Confusing digital model with digital twin.

A static simulation labelled ‘digital twin’ that has no live data connection is marketing, not engineering. Specify the twin level explicitly.

Pitfall 7: Ignoring post-processor accuracy.

The post-processor that converts simulation paths to robot-native code introduces its own errors (joint-angle rounding, interpolation differences). Always verify the first 10 paths on the physical robot at reduced speed.

Pitfall 8: Not calibrating the robot before OLP.

An uncalibrated robot has 3–5 mm absolute error. OLP programs will miss targets by this amount. Robot calibration (€3–8K) is mandatory for OLP-dependent cells.

Pitfall 9: Treating simulation as a substitute for physical testing.

Simulation validates the design concept; physical testing validates the implementation. Both are necessary; neither is sufficient alone.

Pitfall 10: Budgeting simulation as overhead rather than investment.

Simulation that catches 30–50% of commissioning issues returns 3–10× its cost. Budget it as a line item with expected ROI, not as an optional overhead to be cut when the project runs over.

KEY TAKEAWAYS

- Simulation shifts 30–50% of commissioning issues left, where they cost 3–10× less to fix. Every hour of simulation saves 5–10 hours of on-site rework.
- Five techniques serve different phases: reachability (concept), collision (design), cycle-time (design), process simulation (pre-FAT), virtual commissioning (pre-FAT).
- OLP dominates for cells with >4 variants or >10 programmed points. Online teaching remains practical for simple single-program cells.
- Platform selection follows project requirements: RoboDK (budget, vendor-agnostic), Visual Components (mid-range, process simulation), Tecnomatix (enterprise, virtual commissioning).
- Digital-twin maturity spans four levels (Model → Shadow → Twin → Intelligent Twin). Most practice is at Level 2; investment increases ~3× per level.
- The 5–10% sim-reality gap is normal and systematic. Calibrate the model after commissioning to create a reliable baseline for future analysis.
- Simulated CT should be ≤90% of TT to provide margin for the sim-reality gap. At >93%, the cell is at risk of missing takt.

- Cable and hose routing is the #1 unmodelled element. Model at least the major cable runs; add 10–15 mm margin beyond the simulation warning threshold.
- Simulation cost should be 3–5% of total project cost. Modelling effort scales as N^2 with interacting devices. Calibration is budgeted separately.
- Simulation validates the design; physical testing validates the implementation. Both are necessary; neither is sufficient.

Review Questions

Answer the following questions to consolidate your understanding of the chapter material. A thorough response should draw on concepts, formulas, and examples presented in the text.

1. Articulate the economic case for simulation in a 4-robot cell project costing €400K. What simulation investment is appropriate, and what return should be expected?
2. A cell simulates at 38 s cycle time with a takt time of 42 s. Is the margin sufficient? What is the probability of meeting takt at commissioning? What action would you recommend?
3. Compare RoboDK, Visual Components, and Tecnomatix for a project involving 3 KUKA robots and PLC-in-the-loop virtual commissioning. Which platform would you select and why?
4. Explain the four levels of digital-twin maturity. A welding cell has a Tecnomatix model with no live data connection. What level is it? What would upgrading to Level 2 require?
5. List three systematic sources of simulation-reality discrepancy and, for each, quantify the typical magnitude and describe a mitigation strategy.
6. When is online teaching preferable to OLP? Give a specific scenario where online teaching is the correct choice despite OLP being available.
7. A collision check shows 12 mm clearance between the robot's EoAT and a fixture during a trajectory segment. Is this acceptable? What factors should be considered?
8. Describe the eight-stage simulation model-building workflow. Which stages are most vulnerable to modelling errors, and why?
9. A project manager proposes cutting the simulation budget from 4% to 1% of project cost to offset a schedule overrun. Formulate the engineering argument against this decision.

References

- [1] Groover, M.P. (2020) Automation, Production Systems, and Computer-Integrated Manufacturing, 5th ed. Pearson. ISBN 978-0-13-460546-3.
- [2] Siciliano, B. et al. (2019) Robotics: Modelling, Planning and Control, 2nd ed. Springer. ISBN 978-1-84628-641-4.
- [3] RoboDK Inc. (2024) RoboDK Documentation — Offline Programming and Simulation. Montreal.
- [4] Visual Components Oy (2024) Visual Components 4.x User Guide. Espoo, Finland.
- [5] Siemens AG (2023) Tecnomatix Process Simulate — Virtual Commissioning Guide. Nuremberg.
- [6] ABB Robotics (2023) RobotStudio — VirtualController Technical Reference. Västerås.
- [7] FANUC Corporation (2023) ROBOGUIDE HandlingTool User Manual. Oshino, Japan.
- [8] Grieves, M. and Vickers, J. (2017) Digital twin: Mitigating unpredictable emergent behavior. In: Transdisciplinary Perspectives on Complex Systems, Springer, pp. 85–113.
- [9] Hoffmann, P. et al. (2010) Virtual commissioning of manufacturing systems. Proc. 24th European Conf. on Modelling and Simulation, pp. 175–181.
- [10] Mocan, B. et al. (2018) Multidisciplinary Design of Industrial Robotic Automation Solutions. UTPress. ISBN 978-606-737-246-5.
- [11] Mocan, B. (2025) Design and Operation of Robotic Production Systems — Course Notes. UTCN.
- [12] Reinhart, G. and Wunsch, G. (2007) Economic application of virtual commissioning. Production Engineering, 1(4), pp. 371–379.
- [13] ISO 9283:1998, Manipulating industrial robots — Performance criteria and related test methods. ISO, Geneva.
- [14] Dassault Systèmes (2022) DELMIA Robotics Documentation. Vélizy.
- [15] Xu, X. and Lu, Y. (2021) Industry 4.0 and Smart Manufacturing. Elsevier.

CHAPTER

8

Safety Engineering in Robotic Production Systems

*From compliance to design principle — safety as
integration, not retrofit*

This chapter opens Module III, which addresses safe operation and lifecycle management in robotic production systems. Safety engineering, however, does not begin at commissioning: it starts with the definition of intended use and machine limits, continues through design, integration and validation, and remains active during operation, maintenance, modification and decommissioning. A robotic production system is complete only when its risks have been systematically reduced and its safety functions have been verified for all relevant tasks, operating modes and categories of personnel.

LEARNING OBJECTIVES

1. Explain safety engineering as a lifecycle design discipline for robotic production systems
2. Apply the ISO 12100:2010 risk assessment and three-step risk-reduction method
3. Identify hazards across tasks, operating modes and lifecycle phases
4. Select inherently safe design measures, safeguards and information for use in the correct priority order
5. Specify safety functions, safe states, response times and reset/restart behaviour
6. Determine PLr using the ISO 13849-1:2023 risk graph and distinguish required from achieved PL
7. Explain the respective scopes of ISO 10218-1:2025 and ISO 10218-2:2025
8. Differentiate the four collaborative application methods and their validation needs
9. Calculate safeguard positioning parameters and evaluate measured PFL force/pressure data
10. Plan and document design verification, FAT, SAT and lifecycle revalidation

8.1 Why Safety Engineering Matters

Safety in robotic production systems cannot be treated as an afterthought or a final inspection activity. It is a core engineering discipline that must be integrated from conceptual design onwards. Inadequate safety engineering can lead to harm, legal non-compliance, production interruption, loss of equipment availability and long-term organisational consequences.

Industrial robots can move rapidly, generate substantial forces and manipulate heavy, sharp or hot workpieces. Personnel may be exposed during loading, inspection, programming, troubleshooting, maintenance or collaborative tasks. The safety engineer must therefore enable the required production function while reducing risk across the complete machinery lifecycle.

8.1.1 Safety-by-Design Philosophy

Safety-by-Design embeds safety requirements in the engineering process rather than adding protective devices after the main design has been frozen. A practical lifecycle can be organised into four connected phases:

- **Phase 1 — Concept and machine limits:** Define intended use, reasonably foreseeable misuse, lifecycle tasks and space, time and environmental limits. Perform preliminary hazard identification before fixing the layout or robot concept.
- **Phase 2 — Risk assessment and risk reduction:** Identify hazards and hazardous situations, estimate and evaluate risk, and apply the ISO 12100 three-step method in the prescribed priority order.
- **Phase 3 — Engineering and verification:** Implement inherently safe design measures, safeguards, safety-related parts of control systems (SRP/CS), procedures and information for use. Verify the design against the safety requirements specification.
- **Phase 4 — Validation and lifecycle control:** Validate the integrated safety functions, release the system with controlled documentation, and re-assess safety after incidents, significant changes or safety-relevant modifications.

Key principle: risk reduction is normally more effective when hazards are eliminated or reduced during concept development. Late changes frequently constrain the available technical options and can disrupt layout, controls and commissioning.

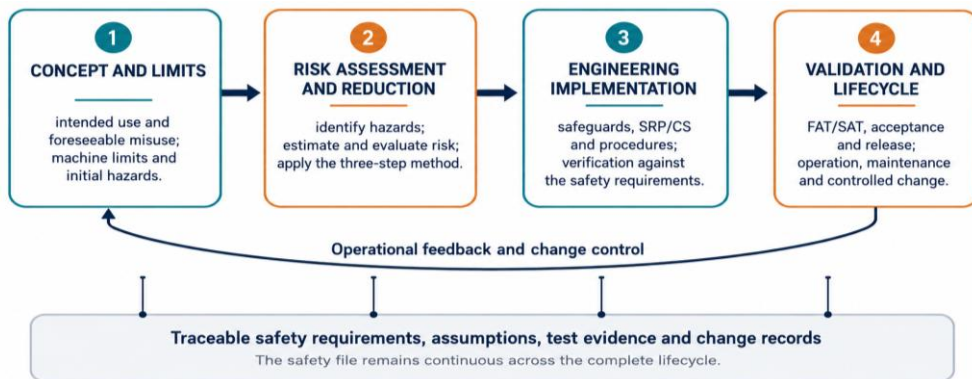


Figure 8.1. Safety-by-Design lifecycle linking machine limits, risk assessment and reduction, engineering verification, validation and lifecycle change control.

8.1.2 Safety's Impact on System Design

Safety requirements shape virtually every aspect of a robotic production system:

- **Layout design:** Safeguarded space, restricted space, access routes and maintenance positions must be derived from the robot and process envelopes.

Safeguard positioning and reach prevention are determined using the applicable Type-B and Type-C standards rather than a generic clearance allowance.

- **Robot and application selection:** The risk assessment determines whether separation, safeguarded coexistence or a collaborative application is appropriate. 'Collaborative' describes an application and operating condition, not an inherently safe robot category.
- **Control architecture:** The safety functions, required PLr or SIL, response times, reset behaviour, mode selection and interfaces between the robot, safety controller and final elements must be specified before detailed control design.
- **EoAT and workpiece design:** Geometry, sharp edges, accessible pinch points, clamping force, retained energy and behaviour after loss of power are safety-critical characteristics of the complete application.
- **Operating and maintenance concepts:** Manual reduced-speed operation, automatic operation, high-speed manual operation where permitted, energy isolation, access control and recovery from faults must be addressed as defined tasks, not left to informal practice.
- **Productivity and availability:** A well-designed safety concept can reduce nuisance stops, simplify recovery and make required interventions safer and more predictable. Productivity claims must nevertheless be verified against the actual process and safeguarding concept.

8.1.3 The Cost of Safety Failures

The economic consequences of an accident are not limited to the immediately visible costs. A serious event can generate medical and repair costs, production interruption, investigation and corrective-action effort, retraining, contractual consequences and loss of workforce confidence. The magnitude and distribution of these costs are case-specific and should not be represented by fixed percentages unless the data source and scope are explicitly defined.

- **Direct and readily visible consequences:** medical response, damaged equipment, immediate downtime, legal or regulatory actions and emergency repair.
- **Indirect operational consequences:** investigation, production recovery, rework, temporary process changes, replacement labour, additional supervision and management time.

- **Longer-term organisational consequences:** reputation, workforce confidence, customer assurance, insurance conditions and the credibility of the safety management system.

For engineering decisions, the important conclusion is not a universal multiplier but the need to include availability, recovery and organisational effects in the business case for prevention. Verified company data should be distinguished from scenario estimates.

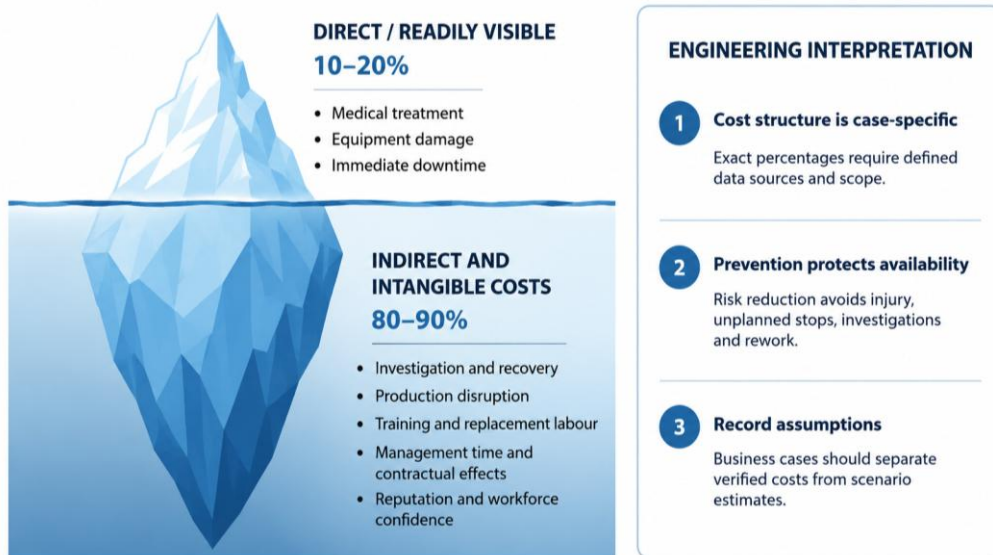


Figure 8.2. Conceptual structure of accident-related costs: readily visible direct consequences and less visible indirect and organisational effects. Relative proportions are case-specific.

8.2 Regulatory Framework and Key Standards

The safety of robotic production systems is governed by a hierarchy of machinery-safety standards. The 2025 editions of ISO 10218-1 and ISO 10218-2 are the current Type-C references for industrial robots, industrial robot applications and robot cells. Type-A and Type-B standards remain essential for risk assessment, functional safety, safeguard positioning and protective-device design.

8.2.1 The Three-Level Standard Hierarchy

Table 8.1. Hierarchy of safety standards by ISO/IEC types

Type	Scope	Representative current standards
Type A — Basic	Basic concepts, terminology and design principles for machinery	ISO 12100:2010 — risk assessment and risk reduction
Type B1 — Safety aspects	Safety aspects applicable across machine categories	ISO 13849-1:2023; IEC 62061:2021; ISO 13855:2024; ISO 13857:2019
Type B2 — Safeguards and devices	Design and selection of guards, interlocks and protective equipment	ISO 14120:2015; ISO 14119:2024; ISO 13850:2015; IEC 60204-1:2016+A1:2021
Type C — Machine-specific	Requirements for a defined machinery category	ISO 10218-1:2025 and ISO 10218-2:2025; ISO/TS 15066:2016 and ISO/PAS 5672:2023 provide supporting collaborative-application guidance

Important: a Type-C standard can specify requirements that take precedence over Type-A or Type-B provisions for the machinery and hazards within its scope. This does not remove the obligation to perform a documented risk assessment and to address hazards or lifecycle tasks not fully covered by the Type-C standard.

8.2.2 EU Machinery Regulation 2023/1230

Regulation (EU) 2023/1230 on machinery applies on a mandatory basis from 20 January 2027 and replaces Directive 2006/42/EC. For systems placed on the EU market before that date, the Machinery Directive remains the primary machinery legislation. The Regulation introduces or clarifies several matters relevant to modern robotic systems:

- **Conformity assessment:** CE marking remains mandatory. The manufacturer of the completed machinery must select the applicable conformity-assessment procedure; machinery categories listed as presenting a higher risk can require a notified-body route or another procedure prescribed by the Regulation.
- **AI and self-evolving behaviour:** safety-related systems using self-evolving behaviour are explicitly addressed, and certain categories are included in the higher-risk machinery list. The safety case must cover the intended operating envelope and foreseeable inputs.
- **Digital instructions and declarations:** instructions for use and the EU declaration of conformity may be supplied digitally when the Regulation’s accessibility,

identification, retention and paper-copy conditions are met. A QR code is one possible implementation, not a universal legal requirement.

- **Cyber-safety:** protection against corruption of safety-relevant software, data and control systems becomes an explicit design concern for connected machinery.
- **Substantial modification:** a physical or digital modification that creates a new hazard or increases an existing risk and requires new protective measures can make the modifier responsible for a new conformity assessment.

8.2.3 CE Marking Process for Robotic Systems

For a system integrator that becomes the manufacturer of the completed robotic production system, the conformity process normally includes the following activities:

- 1. Identify the applicable Union legislation and the relevant harmonised or state-of-the-art standards for the date on which the system is placed on the market or put into service.
- 2. Define the machinery, its intended use, interfaces and limits, then perform and document the ISO 12100 risk assessment.
- 3. Apply risk reduction in the ISO 12100 priority order and specify each safety function, including its required performance and response time.
- 4. Verify the design and compile the technical documentation: risk assessment, drawings, calculations, software and parameter records, component evidence, validation reports and instructions.
- 5. Complete the applicable conformity-assessment procedure and draw up the EU Declaration of Conformity.
- 6. Affix the CE marking and retain the required technical documentation.

The legal role depends on the commercial and technical arrangement. An integrator that places the completed assembly on the market or puts it into service under its own responsibility generally becomes the manufacturer of that assembly. CE-marked components do not remove the need for system-level risk assessment, integration verification and conformity assessment.

8.3 Risk Assessment Methodology (ISO 12100)

ISO 12100:2010 is the foundational Type-A standard for machinery safety. It specifies terminology and an iterative methodology for defining machine limits, identifying hazards, estimating and evaluating risk, and reducing risk throughout the relevant lifecycle phases. The standard does not prescribe one universal numerical scoring matrix.

8.3.1 The Five-Step Risk Assessment Process

Table 8.2. Five-step risk assessment process per ISO 12100

Step	Activity	Key engineering questions
1	Determine machine limits	What is the intended use and foreseeable misuse? Which tasks, persons, modes, spaces, time limits and environments are included?
2	Identify hazards and hazardous situations	Which energy sources, motions, process conditions and human tasks can cause harm in each lifecycle phase?
3	Estimate risk	How severe can the harm be, and how likely is it considering exposure, occurrence and avoidance?
4	Evaluate risk	Is further risk reduction required after applying the relevant legal and standards criteria?
5	Reduce risk and iterate	Apply inherently safe design, then safeguards/complementary measures, then information for use; verify and re-estimate residual risk.

Critical: after implementing a risk-reduction measure, the residual risk must be re-estimated and re-evaluated. Documentation should maintain traceability from the hazardous situation to the selected measure, verification evidence, residual risk and information for use.

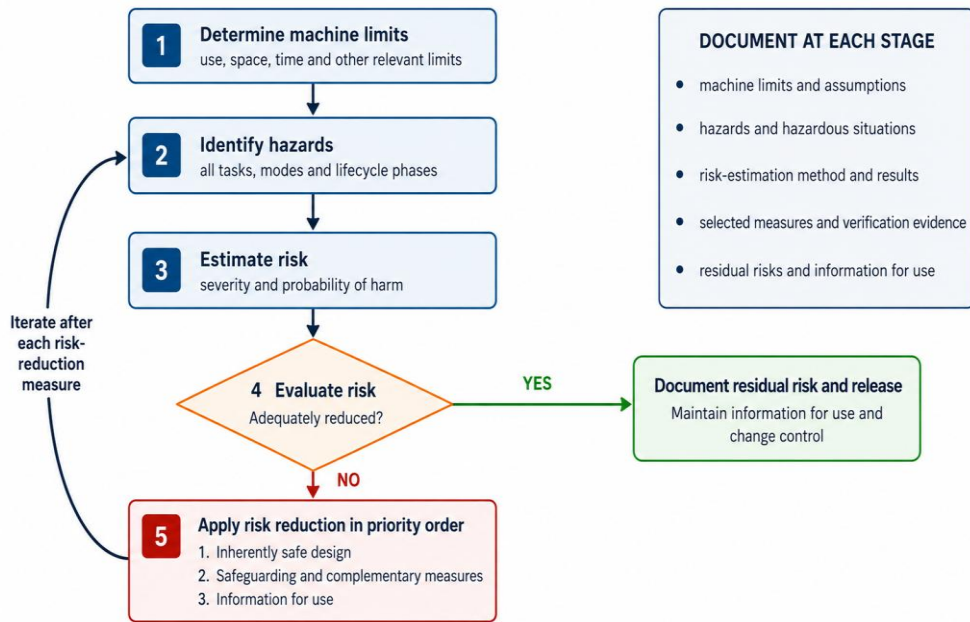


Figure 8.3. ISO 12100 risk assessment and risk-reduction process, including the iterative loop and documentation required at each stage.

8.3.2 Determining Machine Limits (Step 1)

Before any hazards can be identified, the engineer must clearly define the boundaries of the assessment:

Table 8.3. Categories of machine limits per ISO 12100

Limit category	Description	Examples in robotic systems
Use limits	Intended use and reasonably foreseeable misuse	Automatic production, manual reduced-speed operation, high-speed manual verification where permitted, maintenance, cleaning, changeover and recovery
Space limits	Physical and functional boundaries	Maximum and restricted spaces, safeguarded space, collaborative workspace, process and maintenance access
Time limits	Lifecycle and temporal assumptions	Mission time, operating hours, proof-test or inspection intervals, wear parts, software support and decommissioning

Limit category	Description	Examples in robotic systems
Other limits	Environment, materials and interfaces	Temperature, contamination, lighting, EMC, vibration, explosive atmosphere, utilities, network interfaces and handled materials

Incomplete machine limits are a common source of missed hazards. Maintenance, cleaning, teaching, fault recovery and decommissioning must be included where relevant. Reasonably foreseeable misuse—such as entering a safeguarded space to retrieve a dropped part without completing the required energy-isolation procedure—must also be considered.

8.3.3 Hazard Identification (Step 2)

Hazard identification is decisive because an unidentified hazardous situation cannot be evaluated or reduced. The analysis should cover the robot, EoAT, workpiece, process equipment, control system, energy sources, environment and human tasks.

Table 8.4. *Hazard categories in robotic production systems*

Category	Examples	Typical sources in robotic cells
Mechanical	Crushing, shearing, impact, entanglement, puncture, ejection	Robot and positioner motion, EoAT, tooling, conveyors, falling or released workpieces
Electrical	Electric shock, arc, short circuit, unexpected energisation	Controllers, drives, cabinets, damaged cables, incorrect bonding, stored electrical energy
Thermal / radiation	Burns, hot surfaces, arc or laser radiation	Welding, curing, heated tooling, process parts, lasers and UV sources
Noise and vibration	Hearing damage, hand-arm or whole-body vibration	Spindles, impact tools, pneumatics, conveyors and process equipment
Ergonomic / organisational	Manual handling, awkward posture, fatigue, cognitive overload	Loading, recovery, HMI and alarm design, maintenance access and production pressure
Control / HRI	Unexpected start, incorrect mode, collision, trapping, unsafe restart	Software, sensors, communication, mode transitions, human presence near moving equipment
Substances and emissions	Fumes, lubricants, adhesives, cleaning chemicals, dust	Welding, machining, process media, maintenance and cleaning

Category	Examples	Typical sources in robotic cells
Cybersecurity affecting safety	Unauthorised change, corrupted safety data, loss of integrity	Remote access, networked controllers, software updates and safety configuration

Important: hazard identification must cover all relevant lifecycle phases and operating modes. A multidisciplinary team should include design, controls, robot programming, functional safety, maintenance and operations competence. Operator and maintenance experience is particularly valuable for identifying non-routine interventions and foreseeable workarounds.

8.3.4 Risk Estimation (Step 3)

For each hazardous situation, risk is estimated from the severity of possible harm and the probability of that harm occurring. The probability element normally considers exposure, occurrence of the hazardous event and the possibility of avoiding or limiting harm.

Risk = f(severity of harm, probability of occurrence of harm)

ISO 12100 does not mandate a specific S1–S3 scale or a specific matrix. An organisation may use a documented qualitative or quantitative method provided that the criteria are defined, consistently applied and suitable for the machinery. A project-specific method should address:

- **Exposure:** the frequency and duration of a person's presence in the hazardous situation.
- **Occurrence of the hazardous event:** the likelihood of unexpected motion, component failure, process upset or human action that creates the event.
- **Avoidance or limitation of harm:** the person's ability to perceive the event, escape or reduce its consequences, considering speed, visibility, available space and competence.

Table 8.5 shows an illustrative qualitative matrix for prioritisation. It is not prescribed by ISO 12100 and must not be confused with the ISO 13849-1 risk graph used to determine PLr.

Table 8.5. Example qualitative risk matrix for project prioritisation (non-normative)

Severity / likelihood	Very low	Low	Medium	High
Catastrophic or fatal	Medium	High	Very high	Very high
Serious, irreversible	Low	Medium	High	Very high
Slight, reversible	Low	Low	Medium	High

A risk matrix supports consistent prioritisation, but risk acceptance cannot be based on a colour alone. The engineer must apply the three-step method, applicable Type-B and Type-C requirements, the state of the art and the project's documented evaluation criteria.

8.3.5 Risk Evaluation and Adequate Risk Reduction

Risk evaluation determines whether further risk reduction is required. ISO 12100 asks whether the risk has been adequately reduced after the three-step method and the relevant standards have been applied; it does not prescribe ALARP as a universal acceptance rule.

Where an organisation or jurisdiction uses an ALARP or 'reasonably practicable' concept, it can supplement the machinery-safety process but cannot replace applicable legal requirements, Type-C requirements or available protective measures that represent the state of the art. The rationale for every accepted residual risk must remain traceable.

8.4 Risk Reduction Strategies

When risk is not adequately reduced, ISO 12100 requires measures to be applied in the following priority order: inherently safe design measures; safeguarding and complementary protective measures; and information for use.

8.4.1 Step 1 — Inherently Safe Design Measures

The most effective form of risk reduction is to eliminate the hazard through design choices:

- **Reduce energy by design:** lower moving mass, process force or maximum speed where the production function permits. Because kinetic energy is proportional to velocity squared, halving velocity reduces kinetic energy by 75%; safety-related speed limitation used as a control function must still be specified and validated.
- **Remove hazardous geometry:** avoid sharp edges, protrusions, trapping gaps and accessible pinch points on the robot, EoAT, fixtures and workpiece interfaces.
- **Limit force and torque at source:** select actuators, transmissions, pneumatic pressure and mechanical compliance so that the required task can be achieved with the lowest practicable stored and transmitted energy. Controller limits alone do not establish PFL compliance.
- **Prevent gravity-driven motion:** orient mechanisms and provide retention so that tooling or workpieces do not fall after loss of power, pressure or control.
- **Minimise stored energy:** reduce pneumatic accumulations, spring preload, suspended loads and other stored-energy sources; provide controlled dissipation and isolation where elimination is impossible.
- **Use suitable extra-low-voltage circuits:** PELV or SELV can reduce shock risk in interaction zones when correctly designed, but it does not eliminate risks from high current, stored energy, incorrect earthing or higher-voltage circuits elsewhere in the cell.
- **Separate hazardous tasks by design:** arrange the layout and sequence so that human intervention does not coincide with hazardous motion whenever this is compatible with the intended function.

8.4.2 Step 2 — Safeguarding and Complementary Protective Measures

Risks that cannot be sufficiently reduced by design must be addressed using guards, protective devices, safety-related control functions and complementary measures selected for the hazardous situation and required performance.

Table 8.6. *Safeguard types and applicable standards*

Protective measure	Principle	Principal reference	Representative use
Fixed or movable guard	Physical prevention of access	ISO 14120; ISO 13857	Perimeter or local enclosure of hazardous motion
Interlocked guard / guard locking	Stop or prevent hazardous functions when access is possible	ISO 14119:2024	Loading, maintenance and changeover access
Light curtain or other AOPD	Electro-sensitive detection of body parts	IEC 61496-1/-2; ISO 13855:2024	Frequent access without a physical barrier
Safety laser scanner	Configurable protective fields	IEC 61496-3; ISO 13855:2024	Area protection, approach monitoring and SSM inputs
Pressure-sensitive mat or floor	Detect presence by applied pressure	ISO 13856-1	Defined standing or approach areas
Emergency-stop function	Complementary manual protective function; stop category selected by risk assessment	ISO 13850; IEC 60204-1	Emergency intervention; not a substitute for safeguarding
Two-hand control	Requires concurrent actuation and separation from hazard	ISO 13851	Tasks where both hands must remain at the control position
Three-position enabling device	Enables motion only in the centre position	ISO 10218-1:2025	Manual operation inside safeguarded space

8.4.3 Step 3 — Information for Use

Residual risks that remain after design and safeguarding must be communicated through the instruction handbook, task-specific procedures, markings, training and personal protective equipment where applicable. Information must identify the hazard, the residual risk, the relevant tasks and the required protective behaviour.

Important: information for use is the third and least effective step. It must not be used to justify omitting a feasible inherently safe design measure or safeguard.

WORKED EXAMPLE 8.1 — SAFETY DISTANCE CALCULATION

Problem: A vertically mounted light curtain with 30 mm resolution protects a palletising cell. The light-curtain response time is 15 ms and the measured worst-case stopping time of the robot and safety logic is 300 ms. Calculate the illustrative minimum distance for perpendicular approach using the ISO 13855 AOPD method for $d \leq 40$ mm.

Step 1 — Total response time: $T = 0.015 + 0.300 = 0.315$ s.

Step 2 — Supplementary distance: $C = 8(d - 14) = 8(30 - 14) = 128$ mm.

Step 3 — Initial calculation with $K = 2000$ mm/s: $S = 2000 \times 0.315 + 128 = 758$ mm.

Step 4 — Because $S > 500$ mm, recalculate with $K = 1600$ mm/s: $S = 1600 \times 0.315 + 128 = 632$ mm. This remains greater than the 500 mm lower bound.

Result: specify at least 632 mm before tolerances and installation allowance; a practical drawing could use 650 mm only if all other ISO 13855 conditions are satisfied. Verification: confirm measured worst-case stopping time, detection capability, reach-over/under/around possibilities, reflective or environmental effects, and the final location of the nearest hazardous point. The simplified equation does not replace the complete ISO 13855:2024 assessment.

INDUSTRIAL PRACTICE NOTE — Functional-Safety Retrofit of an Established Welding Line

Context: an established multi-robot welding line had accumulated process changes, temporary recovery methods and maintenance interventions that were not fully represented in the original risk assessment.

Engineering response: the retrofit began with renewed machine limits and task observation, followed by a safety requirements specification for each access point and hazardous motion. Interlocks, final switching elements, reset logic, stopping-time data and safe-space functions were reviewed as complete safety chains rather than as isolated components.

Validation: every modified safety function was tested individually, including discrepancy faults, feedback monitoring, restart prevention and worst-case stopping time. Deviations were corrected under configuration control and retested.

Lesson: legacy hardware is not automatically non-compliant, but modifications, changed tasks and foreseeable workarounds can invalidate old assumptions. Retrofit decisions must be based on the current application risk assessment, not on the age of a component alone.

Didactic scenario based on recurrent industrial retrofit practice; no company-specific performance claim is intended.

8.5 Safety Devices and Safety PLCs

Modern robotic production systems commonly use safety relays, configurable safety controllers, safety PLCs and safety-capable robot or drive subsystems. A component may be certified for a maximum capability, but PL or SIL is achieved by the complete safety function—from sensor through logic and communications to the final element and machine response.

8.5.1 Safety-Rated Control Functions

Table 8.7. Representative drive-based safety functions and their application

Function	Full name	Description	Representative use
STO	Safe Torque Off	Prevents torque-producing energy from being applied to the motor; does not guarantee position holding	Emergency or protective stopping sequence when removal of torque is the required safe state
SS1	Safe Stop 1	Initiates controlled deceleration and then activates STO	Controlled stop where uncontrolled coasting would increase risk
SOS	Safe Operating Stop	Maintains a stopped position while the drive remains energised under safety-related monitoring	Temporary human presence with faster controlled restart
SLS	Safely Limited Speed	Prevents speed from exceeding a specified limit	Manual reduced-speed operation and reduced-mode zones
SLP	Safely Limited Position	Prevents motion beyond a specified position or range	Virtual boundaries and restricted robot space
SBC	Safe Brake Control	Provides a safety-related output for an external holding brake	Vertical axes and prevention of gravity-driven motion

The required architecture depends on PLr or SIL, the selected technology and the Type-C requirements. Verification must consider component reliability, architecture, diagnostic coverage, common-cause failure, systematic capability, software, communication and total response time. A ‘SIL 3’ or ‘PL e’ label on a controller does not by itself establish the performance of the integrated safety function.

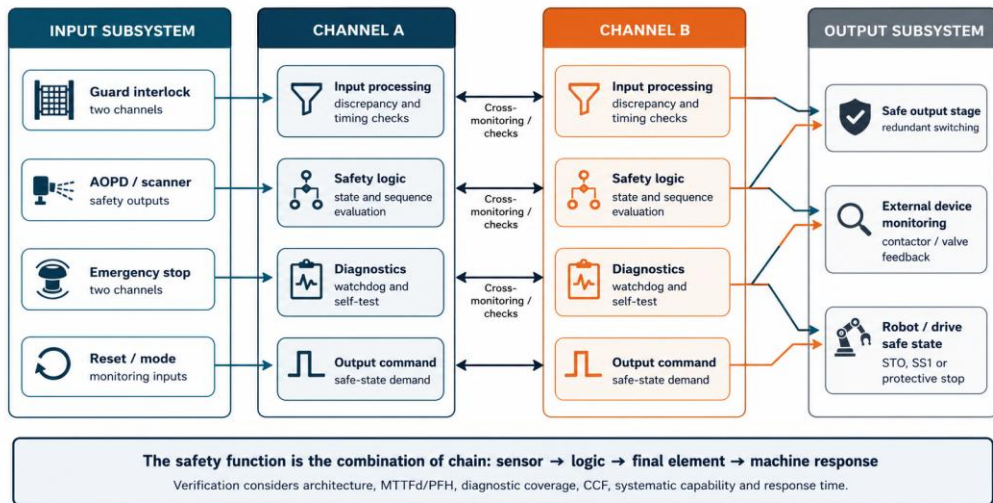


Figure 8.4. Example dual-channel safety-related control architecture showing input, logic, cross-monitoring, output comparison, final elements and external-device feedback.

8.5.2 Safety Distance Calculation (ISO 13855:2024)

Safeguards such as light curtains, laser scanners and pressure-sensitive devices must be positioned so that the protective function is completed before a person can reach the hazard. ISO 13855:2024 provides the applicable methodology for the approach of the human body.

$$S = K \times T + C$$

In the generic expression, S is the minimum distance, K is the approach-speed parameter, T is the total response time of the protective function and C is an additional distance defined by the safeguard type, detection capability and possible reach. The appropriate formula and parameter values depend on the approach geometry and protective device. T must include sensor, safety logic, communication, drive/robot stopping and relevant mechanical delays, using measured or validated worst-case values.

8.5.3 Performance Level Determination (ISO 13849-1)

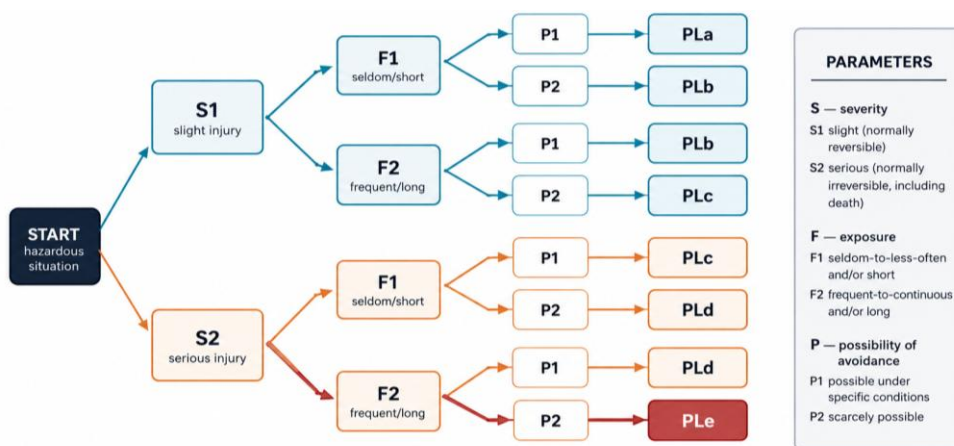
ISO 13849-1:2023 defines Performance Level (PL) for safety-related parts of control systems. The required Performance Level, PLr, can be determined using the standard's risk graph, which uses three parameters:

- S — Severity of injury: S1 (slight, reversible) or S2 (serious, irreversible, including death)
- F — Frequency and/or duration of exposure: F1 (seldom to less often, short duration) or F2 (frequent to continuous, long duration)
- P — Possibility of avoiding the hazard: P1 (possible under specific conditions) or P2 (scarcely possible)

The risk graph leads to PLr a through PLr e. The result specifies the required risk-reduction capability of the safety function; the achieved PL is then verified separately using the complete SRP/CS design.

Table 8.8. ISO 13849-1 risk-graph outcomes and illustrative interpretations

Risk-graph path	PLr	Illustrative interpretation only
S1, F1, P1	PL a	Lowest risk-reduction requirement in the graph
S1, F1, P2 or S1, F2, P1	PL b	Higher exposure or reduced possibility of avoidance
S1, F2, P2 or S2, F1, P1	PL c	Serious harm with limited exposure, or slight harm with adverse exposure/avoidance
S2, F1, P2 or S2, F2, P1	PL d	Serious harm combined with adverse exposure or avoidance
S2, F2, P2	PL e	Highest risk-reduction requirement in the graph



Highlighted example: S2 + F2 + P2 gives PLr = PLe. The graph determines the required level; achieved PL is verified separately.

Figure 8.5. ISO 13849-1 risk graph for PLr determination. The highlighted path S2–F2–P2 leads to PLr e.

WORKED EXAMPLE 8.2 — PLr DETERMINATION USING THE RISK GRAPH

Problem: An interlocked access gate initiates a robot protective stop. Access is frequent, the potential injury from moving equipment is serious, and avoidance in the confined space is scarcely possible.

Risk-graph parameters:

S = S2 — serious, normally irreversible injury, including the possibility of death.

F = F2 — frequent-to-continuous exposure and/or long exposure duration.

P = P2 — avoidance or limitation of harm is scarcely possible.

Result: S2 + F2 + P2 gives PLr e for the complete gate-open-to-safe-state function.

Design implication: PL e is normally associated with a Category 4 architecture in the simplified ISO 13849-1 relationship. The achieved PL must be calculated and verified for the complete input–logic–output chain, including MTTFd/PFH, diagnostic coverage, CCF, software and fault behaviour. A safety PLC certified for PL e is only one subsystem.

Sensitivity: if exposure or avoidance is assigned differently, the result can fall to PL d or PL c. Parameter selection must be justified from the actual task and not chosen to obtain a preferred architecture.

8.6 ISO 10218:2025 — Industrial Robot Safety

8.6.1 ISO 10218-1:2025 — Industrial Robots

ISO 10218-1:2025 addresses the industrial robot as partly completed machinery and defines requirements for inherent design, protective measures, safety functions, interfaces and information needed by the integrator. Representative topics include:

- Safety-related stopping, speed, position and space-limiting functions where provided and claimed.
- Emergency-stop and protective-stop interfaces, with defined stop behaviour and restart conditions.
- Reduced-speed safety function, limiting speed to no more than 250 mm/s where that function is used.
- Three-position enabling devices and single-point-of-control provisions for manual operation inside the safeguarded space.

- Mode selection, prevention of unexpected start-up and interfaces for external safeguarding.
- Mechanical and control-system measures that prevent unintended motion or uncontrolled release of energy.
- Information for integration, including stopping performance, limits, payload conditions, safety-function capability and required maintenance.
- Cybersecurity-related measures where unauthorised access or corrupted data could affect a safety function.

The robot manufacturer is responsible for conformity of the robot product and for supplying the integration information. The system integrator remains responsible for the risks created by the application and the completed cell.

8.6.2 ISO 10218-2:2025 — Industrial Robot Applications and Robot Cells

ISO 10218-2:2025 covers integration, commissioning, operation, maintenance and decommissioning of industrial robot applications and robot cells. The integrator's responsibilities include:

- Layout and installation — Robot placement, safeguard positioning, safety zone definition, floor markings, access control, and maintenance access routes.
- Safeguard selection and integration — Choosing appropriate safeguards and integrating them with the safety control system at the required Performance Level.
- Control system integration — Interconnection of robot safety functions with the cell-level safety PLC, including I/O mapping, response times, and diagnostic coverage.
- Application-level risk assessment covering all tasks, modes, collaborative conditions, fault recovery, maintenance and reasonably foreseeable misuse.
- Verification and validation of the complete safety functions, technical documentation, instructions, competence requirements and lifecycle controls.

For a robotic system designer or integrator, ISO 10218-2 is the principal Type-C reference because it addresses the completed application rather than the robot alone.

8.6.3 Operating Modes

Table 8.9. Representative operating modes and access conditions in ISO 10218:2025

Mode	Speed / control condition	Core safety requirements	Access condition
Manual reduced speed	Reduced-speed safety function; no greater than 250 mm/s	Three-position enabling device, single point of control, deliberate motion command	Authorised person can be inside safeguarded space for defined tasks
High-speed manual verification	Speed above reduced-speed limit only under the conditions permitted by ISO 10218:2025 and the application risk assessment	Hold-to-run or equivalent controls, enabling function and additional protective measures	Under the 2025 concept, performed from outside the safeguarded space unless a more specific compliant arrangement applies
Automatic	Process speed and automatic sequence	All required safeguards and safety functions active; controlled mode selection and restart	No person in safeguarded space unless the validated collaborative application explicitly permits it

INDUSTRIAL PRACTICE NOTE — PFL Validation of a Shared Assembly Task

Context: a robot presents a moulded component while an operator installs fasteners in the same workspace. The robot, vacuum gripper and workpiece can all contact the operator.

Validation method: the team defines credible quasi-static and transient contact points, body regions, payloads, speeds and approach directions. Force and pressure are measured with a suitable pressure-force measurement device using the method in ISO/PAS 5672:2023 and the biomechanical criteria selected from the controlled project standard.

Finding: broad padded surfaces can satisfy the criteria while an exposed workpiece corner fails the pressure criterion because the contact area is small. The corrective action is geometrical—remove or guard the corner, increase contact area or prevent the contact—not merely reduce a controller torque value.

Lesson: PFL validation belongs to the complete application and must include the workpiece, fixtures and all reachable contact geometries under representative worst-case conditions.

Didactic scenario; numerical acceptance limits must be taken from the controlled standard and recorded in the validation plan.

8.7 Collaborative Applications — ISO 10218:2025, ISO/TS 15066 and ISO/PAS 5672

The 2025 editions of ISO 10218 integrate collaborative-application requirements into the Type-C framework. ISO/TS 15066:2016 remains a published technical specification providing biomechanical guidance, while ISO/PAS 5672:2023 specifies test methods for measuring and analysing force and pressure in human–robot contacts. ISO/TS 15066 is under revision, so controlled project documentation should identify the edition used.

8.7.1 Four Collaborative Application Methods

Method 1 — Safety-Rated Monitored Stop (SRMS): hazardous robot motion is stopped before a person enters the collaborative workspace. Drive power can remain available in a safety-rated stopped condition, but motion cannot resume until the specified restart conditions are satisfied.

Method 2 — Hand Guiding: after the robot has reached the required stopped state, the operator commands motion through a hand-guiding device. Speed, enabling conditions and the accessible workspace remain safety-related constraints.

Method 3 — Speed and Separation Monitoring (SSM): robot speed is controlled as a function of separation from the person. The protective separation distance must account for human approach, robot motion, sensing uncertainty, system response and stopping behaviour.

Method 4 — Power and Force Limiting (PFL): contact can be permitted only when the complete application—including robot, EoAT, workpiece and environment—keeps force, pressure and contact conditions within the validated criteria for the relevant body region and contact type.

8.7.2 PFL Body Model and Force/Pressure Thresholds

ISO/TS 15066 distinguishes quasi-static and transient contact and provides biomechanical data by body region. These values are application inputs, not a substitute for risk assessment or measurement of the actual contact scenario.

- Quasi-static contact (clamping): the body region can be trapped between the moving system and another object, so force and pressure can be sustained and escape can be limited.

- Transient contact (impact): the person is not clamped and can move away, so the contact duration is shorter. The test method and applicable limit still depend on body region, geometry and measurement conditions.

-

Table 8.10. Selected ISO/TS 15066 biomechanical values for teaching examples

Body region	Max force — quasi-static (N)	Max force — transient (N)	Max pressure — quasi-static (N/cm ²)	Max pressure — transient (N/cm ²)
Chest	140	140	25	50
Upper arm / elbow	150	190	30	50
Forearm / wrist	160	220	35	50
Hand / fingers	140	280	100	300
Thigh / knee	220	250	50	60

Critical design implication: the assessment covers the entire moving system and every credible contact point. Rounded geometry, increased contact area and mechanical compliance can reduce pressure, but compliance must be demonstrated by measurement under representative worst-case conditions using an appropriate force/pressure measurement device.

8.7.3 Summary of Collaborative Application Methods

Table 8.11. Comparison of four collaborative application methods

Method	Contact during hazardous motion?	Motion condition	Primary safety technology	Representative application
SRMS	No	Robot reaches and maintains a safety-rated stopped condition	Presence detection and monitored stop	Loading, inspection or intervention with robot stationary
Hand guiding	Intentional guided interaction	Limited, operator-commanded motion	Hand-guiding device, enabling function and speed limitation	Teaching, positioning and assisted manipulation

Method	Contact during hazardous motion?	Motion condition	Primary safety technology	Representative application
SSM	No contact intended	Speed and stopping behaviour depend on separation	Safety-rated person detection and separation calculation	Shared workspace with dynamic productivity
PFL	Contact can be permitted	Motion limited by validated contact criteria	Robot/drive limits plus force-pressure measurement and safe geometry	Assembly, handover, packaging and inspection

Key insight: methods can be combined or changed between process phases, but every mode and transition must be included in the risk assessment and safety requirements specification. A ‘cobot’ is not inherently safe; safety is a property of the validated application.

WORKED EXAMPLE 8.3 — PFL FORCE/PRESSURE EVALUATION USING MEASURED DATA

Problem: During a quasi-static forearm contact test, the pressure-force measurement device records a peak force of 120 N. The effective contact area is 5 cm² on the padded face and 2 cm² at an exposed edge. Project criteria imported from the controlled biomechanical table are $F_{\text{limit}} = 160 \text{ N}$ and $p_{\text{limit}} = 35 \text{ N/cm}^2$.

Force check: $120 \text{ N} < 160 \text{ N}$ — the measured force criterion is satisfied for both geometries.

Padded face pressure: $p = F/A = 120/5 = 24 \text{ N/cm}^2$.

$24 \text{ N/cm}^2 < 35 \text{ N/cm}^2$ — the padded face satisfies the pressure criterion.

Exposed edge pressure: $p = 120/2 = 60 \text{ N/cm}^2$.

$60 \text{ N/cm}^2 > 35 \text{ N/cm}^2$ — the edge fails even though the force criterion passes.

Engineering decision: remove or guard the edge, increase the contact area or prevent the quasi-static contact, then repeat the measurement at the worst credible payload, speed, direction and contact location.

Important: a calculation from robot mass and an assumed impact duration cannot establish compliance. Acceptance requires representative force/pressure measurement and documented uncertainty according to the selected validation method.

8.8 Safety Validation Methodology

Safety validation demonstrates that each safety function and the complete safety-related control system meet the specified requirements under representative and fault conditions. It is based on a validation plan, objective acceptance criteria, calibrated or traceable measurement where required, configuration control and recorded evidence.

8.8.1 Validation Planning

The validation plan should identify every safety function, its trigger and safe state, PLr or SIL target, stopping-time or distance criterion, reset and restart behaviour, relevant fault tests, test equipment, responsibilities and required records. Validation independence should be proportionate to the risk and project organisation.

8.8.2 FAT and SAT Safety Tests

Validation is commonly divided between FAT and SAT, but design verification precedes both and safety-relevant modifications can require partial or complete revalidation.

Factory Acceptance Testing (FAT) is conducted at the integrator's facility before shipment, to the extent that the final configuration and interfaces are available. Typical safety tests include:

- Emergency-stop function — actuate every device and verify the specified stop behaviour, stopping time, reset and prevention of unexpected restart.
- Guard interlocking and guard locking — verify all access points, discrepancy detection, escape release where applicable, and restart conditions.
- Safety-related motion functions — verify configured limits such as reduced speed, safely limited position, safe operating stop or drive-based stop functions with suitable measurement methods.
- Protective-device integration — verify detection capability, protective fields, muting or blanking where used, response time and calculated minimum distance.
- Mode and control selection — verify permitted motion, enabling-device behaviour, single-point-of-control and restrictions for each operating mode.

Site Acceptance Testing (SAT) repeats all tests affected by transport, reassembly, site utilities, network integration, environmental conditions or site-specific interfaces. SAT confirms that the installed system matches the approved configuration and that no new hazardous situation has been introduced.

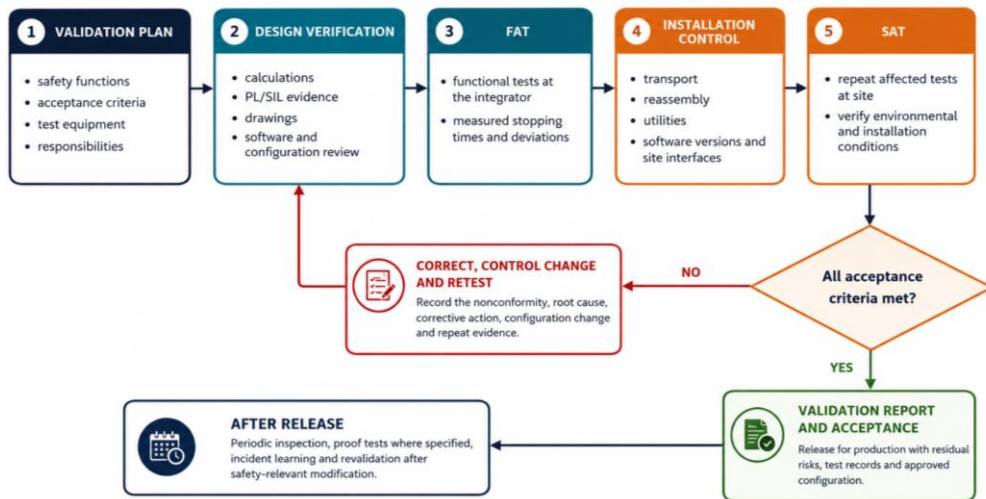


Figure 8.6. Safety validation flow from planning and design verification through FAT, installation control, SAT, correction and retest, acceptance and lifecycle revalidation.

WORKED EXAMPLE 8.4 — SAFETY VALIDATION CHECKLIST (EXTRACT)

The acceptance criterion for every test is taken from the safety requirements specification; generic times or distances are not substituted for project values.

Test 1 — Emergency stop: actuate every device. Verify the specified stop category, measured stopping time, reset and prevention of automatic restart. Result: ____

Test 2 — Gate interlock and locking: open each access point in every relevant mode. Verify stop, fault detection, locking/escape functions and restart conditions. Result: ____

Test 3 — Light curtain: test the detection capability at representative positions, verify protective-field coverage and confirm the installed distance is not less than the calculated project minimum. Result: ____

Test 4 — Reduced speed / SLS: command the maximum permitted motion and verify the safety-related speed limit with appropriate measuring equipment. Result: ____

Test 5 — Safely limited position / space: approach each configured boundary from representative directions and verify the safe response before the allowed boundary is exceeded. Result: ____

Test 6 — Loss of power or pressure: verify the defined safe state, retained loads, brake or valve behaviour, and prevention of unexpected motion during restoration. Result: ____

Record test conditions, software and parameter versions, instruments, results, deviations, corrective actions, retest evidence, tester and reviewer.

INDUSTRIAL PRACTICE NOTE — From Certified Safety Functions to a Validated Cell

Context: modern robot controllers provide configurable safety functions such as speed, position, space, momentum, force or stop monitoring. Product certification defines the capability and constraints of the component.

Integrator responsibility: select the functions required by the application risk assessment, define their limits and interfaces, protect configuration access, and preserve software and parameter versions in the technical documentation.

Validation: verify that the configured limits produce the required safe state for every relevant robot configuration, payload, tool and process condition. Include external devices, communications, stopping performance and final elements; do not validate the robot controller in isolation.

Lifecycle control: safety-relevant parameter changes require authorisation, traceability, impact assessment and revalidation of the affected functions.

A certified function is a building block. The integrator validates the configured assembly and its application-level risk reduction.

Common Pitfalls in Safety Engineering

Pitfall 1: Incomplete hazard identification.

Failing to consider all lifecycle phases (maintenance, changeover, cleaning, decommissioning) or all operating modes (T1, T2, automatic, collaborative). The most dangerous hazards are often those that occur during non-routine activities.

Pitfall 2: Safety as a retrofit.

Designing the main system first and attempting to add safety afterwards restricts layout, control and process options and often creates avoidable re-engineering.

Pitfall 3: Ignoring the EoAT and workpiece.

Assessing only the robot, when the gripper, the part being manipulated, and process materials (adhesives, sharp edges, hot surfaces) often present the greatest hazards.

Pitfall 4: Assuming a collaborative robot is automatically safe.

The robot is only one element of the application. EoAT, workpiece, fixtures, surrounding structures, process energy and foreseeable human contact must all be assessed.

Pitfall 5: Incorrect safety distance calculations.

Using wrong approach speed (K), underestimating total stopping time (T), or neglecting supplementary distance (C). Always measure actual stopping time during commissioning.

Pitfall 6: Confusing PLr determination with PL verification.

PLr (required) is determined from the risk graph. PL (achieved) is verified through architectural analysis of the safety system. Both must be documented and $PL \geq PLr$.

Pitfall 7: Inadequate documentation.

Risk assessment, safety requirements, verification, validation and residual-risk information form part of the technical documentation required for conformity assessment.

Pitfall 8: Bypassing or defeating safety devices.

Designing systems that incentivise operators to bypass safeguards (e.g., restart procedures too complex, muting too aggressive, access too restricted for routine tasks).

Pitfall 9: Neglecting foreseeable misuse.

ISO 12100 requires intended use and reasonably foreseeable misuse to be considered. The design should reduce incentives and opportunities to defeat safeguards and should make the safe method practical for routine work.

Pitfall 10: No periodic safety review.

Commissioning validation is not the end of the lifecycle. Inspection, proof testing where specified, preventive maintenance and revalidation after safety-relevant modification must follow documented intervals derived from component instructions, the risk assessment, site policy and applicable standards. ISO 13849-1 and IEC 62061 do not prescribe one universal annual interval.

KEY TAKEAWAYS

Safety is a lifecycle design discipline. Machine limits, tasks, safeguards, control functions and validation must be considered before the production concept is frozen.

ISO 12100 defines an iterative process: determine limits, identify hazards, estimate and evaluate risk, reduce risk in the prescribed priority order, verify and reassess residual risk.

ISO 12100 does not prescribe one universal matrix or ALARP threshold; the chosen estimation method and acceptance criteria must be documented and applied consistently.

PLr is determined from the ISO 13849-1 risk graph; achieved PL is verified for the complete safety function and must meet or exceed PLr.

ISO 10218-1:2025 addresses the industrial robot; ISO 10218-2:2025 addresses the industrial robot application and robot cell.

Collaborative safety is an application property. SRMS, hand guiding, SSM and PFL can be combined, but every method and transition must be risk-assessed and validated.

PFL acceptance requires representative force and pressure measurement of the robot, EoAT and workpiece contact scenarios; controller limits alone are insufficient.

Safeguard positioning uses ISO 13855:2024 and measured worst-case response time. Reach-over, reach-under and bypass possibilities must also be checked.

FAT and SAT are parts of a wider verification and validation process. Deviations require controlled correction and retest before release.

Regulation (EU) 2023/1230 applies from 20 January 2027; the manufacturer of the completed system remains responsible for system-level conformity assessment.

Review Questions

1. Describe the three steps of the risk reduction method defined in ISO 12100. Why must they be applied in the specified order? Give an example of each step for a robotic palletising cell.
2. Explain the difference between ISO 10218-1 and ISO 10218-2 in terms of scope and responsibility. Which standard is most relevant to a robotic system integrator, and why?
3. A vertically mounted light curtain has 30 mm resolution, a sensor response time of 20 ms, safety-logic processing time of 20 ms and a measured robot stopping time of 280 ms. Using the appropriate ISO 13855 approach for an AOPD with $d \leq 40$ mm, calculate $C = 8(d - 14)$, perform the initial calculation with $K = 2000$ mm/s, and determine whether recalculation with $K = 1600$ mm/s is required. What additional reach-over and installation checks remain?
4. Using the ISO 13849-1 risk graph, determine the PLr for a safety laser scanner that monitors the approach zone of a robotic cell where operators frequently load parts (every 2 minutes) and the potential injury is serious (crushing). The operator can see the robot but cannot avoid it in the confined loading zone.
5. Describe the four collaborative application methods: safety-rated monitored stop, hand guiding, speed and separation monitoring, and power and force limiting. For each, state the central safety condition and one representative sensing or control technology.

6. A PFL validation test on the operator's forearm records a peak quasi-static force of 120 N. The measured contact area is 5 cm² with the padded tool surface and 2 cm² at an exposed edge. Calculate the two pressures and compare force and pressure with project limits of 160 N and 35 N/cm² imported from the controlled biomechanical table. Which geometry passes, and what engineering action is required?
7. Explain the difference between quasi-static contact and transient contact in PFL mode. Why does quasi-static contact have lower force thresholds? Give a practical example of each in a robotic assembly scenario.
8. Describe the safety validation process for a robotic production system, including the distinction between FAT and SAT. List five specific safety tests that should be performed during FAT.
9. A robotic packaging cell handles consumer products at high speed. Operators access the cell once per shift for quality sampling. Design the safeguarding concept: select appropriate safeguards, calculate safety distances, determine the PLr for the primary safety function, and outline the validation tests.

References

- [1] ISO 12100:2010, Safety of machinery — General principles for design — Risk assessment and risk reduction.
- [2] ISO 10218-1:2025, Robotics — Safety requirements — Part 1: Industrial robots.
- [3] ISO 10218-2:2025, Robotics — Safety requirements — Part 2: Industrial robot applications and robot cells.
- [4] ISO/TS 15066:2016, Robots and robotic devices — Collaborative robots.
- [5] ISO/PAS 5672:2023, Robotics — Collaborative applications — Test methods for measuring forces and pressures in human–robot contacts.
- [6] ISO 13849-1:2023, Safety of machinery — Safety-related parts of control systems — Part 1: General principles for design.
- [7] IEC 62061:2021, Safety of machinery — Functional safety of safety-related control systems.
- [8] ISO 13855:2024, Safety of machinery — Positioning of safeguards with respect to the approach of the human body.
- [9] ISO 13857:2019, Safety of machinery — Safety distances to prevent hazard zones being reached by upper and lower limbs.
- [10] ISO 14119:2024, Safety of machinery — Interlocking devices associated with guards — Principles for design and selection.
- [11] ISO 14120:2015, Safety of machinery — Guards — General requirements for the design and construction of fixed and movable guards.
- [12] ISO 13850:2015, Safety of machinery — Emergency stop function — Principles for design.
- [13] IEC 60204-1:2016+A1:2021, Safety of machinery — Electrical equipment of machines — Part 1: General requirements.

[14] IEC 61800-5-2:2016, Adjustable speed electrical power drive systems — Part 5-2: Safety requirements — Functional.

[15] Regulation (EU) 2023/1230 of the European Parliament and of the Council of 14 June 2023 on machinery.

9

Operation, Monitoring, and Performance Metrics

From measurement to management — KPIs and digital supervision driving operational excellence

With the robotic production system designed, integrated, and safety-validated, attention shifts to sustained operation. This chapter examines how operational data are converted into reliable decisions through clearly defined metrics, traceable data contexts, and disciplined review routines. A system can appear productive while losses accumulate invisibly through short stops, speed drift, quality escapes, energy waste, or poorly classified downtime; effective supervision makes those losses measurable and actionable.

LEARNING OBJECTIVES

1. Define and calculate the key performance indicators (KPIs) used in robotic production systems
2. Compute Overall Equipment Effectiveness (OEE) and interpret its three constituent factors
3. Explain the role of SCADA systems in real-time monitoring of robotic production
4. Describe how Manufacturing Execution Systems (MES) support production scheduling and traceability
5. Analyse cycle time, takt time, and throughput relationships for production line balancing
6. Perform downtime analysis using Pareto classification and root cause identification
7. Apply data-driven decision-making principles to operational optimisation
8. Evaluate the impact of Industry 4.0 technologies on production monitoring and control
9. Calculate First Pass Yield and rolled throughput yield for multi-station production lines
10. Design an effective HMI layout following high-performance graphics principles

9.1 The Operational Phase in Context

The operational phase is the period in which the robotic production system creates value by producing conforming output, meeting delivery commitments, and realising the return on the investment made during design and commissioning. Because this phase may extend over many years and normally dominates lifecycle cost, operational performance must be managed through three interconnected capabilities:

- **Monitoring** — Continuously collecting data on system status, process parameters, and production outputs in real time. Every robot cycle, every sensor reading, every fault code is a data point.
- **Measurement** — Quantifying system performance through standardised Key Performance Indicators (KPIs) that enable objective comparison across shifts, products, and time periods.

- **Management** — Using monitored data and measured KPIs to make informed decisions about scheduling, maintenance, process adjustments, and capacity planning.

9.1.1 The Monitoring-Measurement-Management Loop

These capabilities form a closed improvement loop. Validated shop-floor data are transformed into contextualised indicators; the indicators are reviewed against targets and operating constraints; actions are assigned, implemented, and verified; and the resulting process state generates the next set of observations. The loop is effective only when definitions, data quality, ownership, and closure evidence are controlled.

Key insight: The value of monitoring lies in the decisions it enables. Large data volumes do not create improvement by themselves; a small, well-defined set of indicators linked to named actions and review routines is often more valuable than an extensive but unused data infrastructure.

Figure 9.1 shows the Monitoring–Measurement–Management loop and the two controls that make it operationally credible: a data-quality gate before KPI calculation and an action-control step that records ownership, deadlines, evidence, and closure.

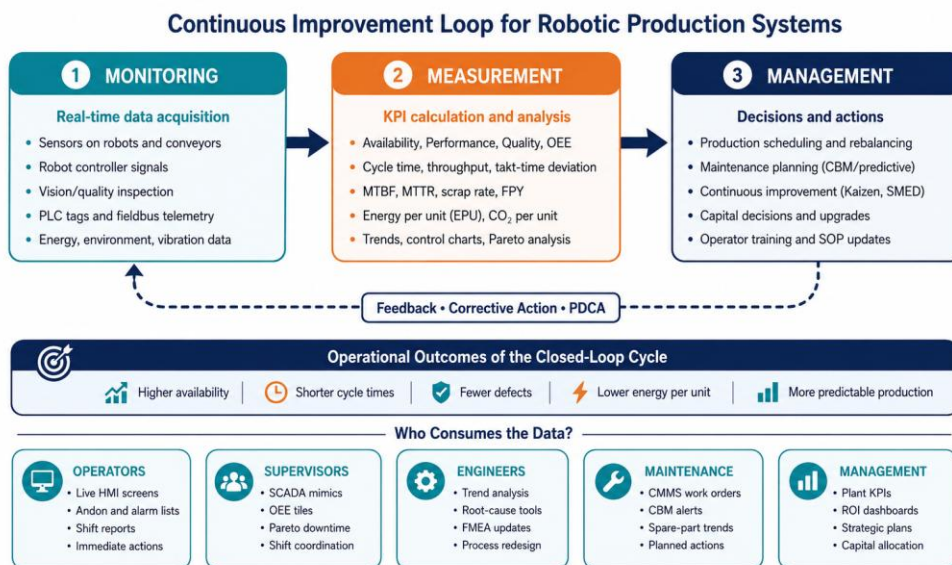


Figure 9.1. Monitoring → Measurement → Management: a continuous improvement loop for robotic production systems.

9.1.2 The ISA-95 Automation Pyramid

ISA-95, published internationally as IEC 62264, defines functional levels and interfaces between enterprise and manufacturing-control activities. It is a reference model for responsibilities and information exchange, not a mandatory network topology and not a rule that every software product must occupy only one level (table 9.1).

Table 9.1. Architectural framework for monitoring and control systems

Level	Name	Systems	Time Horizon	Data Examples
Level 0	Physical Process	Sensors, actuators, robots	Milliseconds	Joint positions, forces, temperatures, and I/O states
Level 1	Basic Control	PLCs, robot controllers	Milliseconds–seconds	Program execution, machine sequencing, and motion commands
Level 2	Supervisory Control	SCADA, HMI	Seconds–minutes	Process synoptics, alarms, trends, and operator actions
Level 3	Manufacturing Operations	MES	Minutes–hours	Work orders, quality records, OEE, production scheduling
Level 4	Business Planning	ERP	Hours–days	Customer orders, procurement, financial reporting, capacity planning

Information generally moves upward as events and production results are contextualised and aggregated, while orders, recipes, and constraints move downward towards execution. The hierarchy does not imply that every raw signal should be transmitted to higher levels; each interface should define semantics, ownership, update rate, retention, and cybersecurity controls.

Figure 9.2 presents the five ISA-95 functional levels together with typical systems, decision horizons, and examples of information exchanged between them.

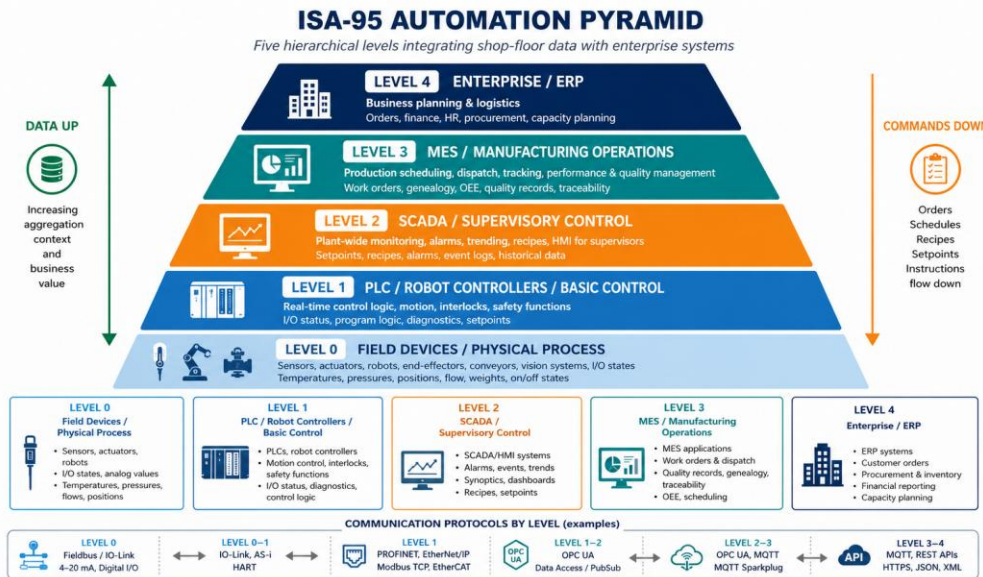


Figure 9.2. ISA-95 automation pyramid showing the five hierarchical levels and the data and command flows between them.

9.2 Key Performance Indicators (KPIs) for Robotic Production

Key Performance Indicators provide a consistent vocabulary for discussing system performance. Each KPI should have a documented formula, unit, time base, production scope, inclusion and exclusion rules, data source, and responsible owner; otherwise, apparently identical values may not be comparable across shifts, products, or sites.

9.2.1 Time-Based Indicators

Cycle Time (CT) is the elapsed time between equivalent reference points in consecutive production cycles, for example from the start of one unit to the start of the next. For a single robotic cell it may be decomposed as:

$$CT = \text{Motion time} + \text{Process time} + \text{Load/Unload time} + \text{Wait time}$$

For a stable single-cycle resource, the theoretical rate is the reciprocal of cycle time after units are made consistent. If CT is expressed in seconds per unit, the theoretical throughput is $3600/CT$ units per hour. In a line, practical capacity is governed by the active bottleneck and may also depend on buffers, parallel resources, and product mix.

Takt Time (TT) is the pace at which products must be produced to meet customer demand:

$$TT = \text{Available Production Time} / \text{Customer Demand}$$

For example, if available production time is 450 minutes per shift and customer demand is 900 units per shift, $TT = 30$ seconds per unit. The system's cycle time must satisfy $CT \leq TT$ to meet demand. If $CT > TT$, the system is a bottleneck.

Lead Time is the total time from order receipt to product delivery. While cycle time measures the production process itself, lead time includes all upstream and downstream activities: order processing, material procurement, queue time, production, inspection, packaging, and shipping.

9.2.2 Output-Based Indicators

Throughput is the number of conforming units produced per unit of elapsed time. The denominator must be stated explicitly: calendar time, scheduled time, planned production time, or run time. For production reporting, a common definition is:

$$\text{Actual Throughput} = \text{Good Units Produced} / \text{Stated Time Basis}$$

Operational availability is the proportion of the defined planned production time during which the equipment is able to perform the required production function:

$$\text{Availability} = \text{Run Time} / \text{Planned Production Time} \times 100\%$$

9.2.3 Quality-Based Indicators

First Pass Yield (FPY) is the proportion of units that pass a defined operation without rework, repair, or retest:

$$\text{FPY} = \text{First-Pass Good Units} / \text{Units Entering the Operation} \times 100\%$$

For a multi-station line, the product of the station first-pass yields is normally called Rolled Throughput Yield (RTY), not line FPY:

$$\text{RTY} = Y_1 \times Y_2 \times Y_3 \times \dots \times Y_n$$

Small quality losses therefore compound across a line. Three stations at 98% first-pass yield give $\text{RTY} = 0.98^3 = 94.1\%$; five such stations give $\text{RTY} = 0.98^5 = 90.4\%$.

9.2.4 Economic Indicators

Cost per Unit (CPU) captures the total production cost divided by good units produced. It includes direct costs (materials, energy, labour) and allocated indirect costs (depreciation, maintenance, overhead):

$$\text{CPU} = (\text{Direct Costs} + \text{Allocated Indirect Costs}) / \text{Good Units Produced}$$

Energy per Unit (EPU) normalises energy consumption to conforming production output and should be reported with the product, recipe, shift, and metering boundary:

$$\text{EPU} = \text{Total Energy Consumed (kWh)} / \text{Good Units Produced}$$

Monitoring EPU reveals avoidable consumption during idle periods, inefficient trajectories, compressed-air leakage, and process drift. Standby share is highly application-dependent, so it should be measured from the local baseline rather than assumed from a generic percentage.

9.3 Overall Equipment Effectiveness (OEE)

Overall Equipment Effectiveness (OEE) is a widely used method for separating losses into availability, speed, and quality components. It is meaningful only when the planned-production boundary, ideal cycle time, good-count rule, and loss categories are defined consistently:

$$\text{OEE} = \text{Availability} \times \text{Performance} \times \text{Quality}$$

9.3.1 The Three OEE Factors

Table 9.2 defines the three OEE factors. The decomposition is more informative than the final product because it shows whether the dominant opportunity lies in uptime, operating rate, or quality yield.

Table 9.2. *The three OEE Factors—Availability, Performance, and Quality*

Factor	Formula	What It Captures	Losses Included
Availability (A)	Run Time / Planned Production Time	Is the machine running?	Equipment failures, changeovers and adjustments within the defined planned-production window
Performance (P)	(Ideal CT × Total Units) / Run Time	Is it running at full speed?	Reduced speed, minor stops, idling, and cycle-rate losses
Quality (Q)	Good Units / Total Units	Are outputs acceptable?	Rejects, rework, startup defects, and process nonconformities

9.3.2 The Six Big Losses

The Six Big Losses from Total Productive Maintenance provide a practical classification for connecting OEE losses to corrective actions. Their magnitude must be measured from the local event history; universal percentage ranges are not technically defensible.

OEE is linked to the Six Big Losses from Total Productive Maintenance (TPM):

Table 9.3. Mapping OEE Factors to the Six Big Losses in Robotic Production

OEE Factor	Loss Type	Examples in Robotic Production	Primary Engineering Response
Availability	1. Equipment failure	Robot faults, servo failures, controller errors, pneumatic leaks	Fault analysis, reliability improvement, condition monitoring
Availability	2. Setup & adjustments	Product changeovers, tool changes, calibration, program uploads	SMED, pre-staging, standard work, recipe and tooling control
Performance	3. Idling & minor stops	Part jams, sensor misreads, conveyor hesitations, comm. timeouts	Automatic event capture, sensor and flow diagnostics
Performance	4. Reduced speed	Below-design CT due to wear, conservative parameters, suboptimal paths	Path and process optimisation; wear and parameter verification
Quality	5. Process defects	Misaligned parts, weld defects, incorrect assembly, dimensional errors	Process control, inspection, and permanent cause removal
Quality	6. Startup rejects	Parts during warmup, calibration runs, first-article inspection	Warm-up stabilisation, first-article control, and recipe verification

9.3.3 OEE Benchmarks and Interpretation

OEE should be interpreted as a contextual improvement metric rather than a universal ranking. The frequently cited 85% value is a heuristic obtained from approximately 90% Availability, 95% Performance, and 99.9% Quality; it is not a requirement in ISO, IEC, or ISA standards.

- Use the plant baseline first: compare the same asset, product family, schedule, and counting rules over time. Targets should reflect process capability, business demand, maintenance strategy, and the economic cost of each loss category.
- A low OEE does not by itself prove that a process is failing. It may reflect an inappropriate ideal cycle time, an overly broad planned-production window, frequent legitimate product changes, or genuinely poor equipment performance. The factor definitions and event classifications must be audited before conclusions are drawn.

- Cross-site or cross-machine comparisons are valid only when the compared systems use equivalent scopes and formulas. Dissimilar products, batch sizes, operating policies, or quality rules can produce misleading league tables.

An OEE of 85% does not mean that 15% of units are defective. The gap represents the combined effect of time losses, rate losses, and quality losses within the selected production boundary.

Engineering takeaway: Always display Availability, Performance, and Quality separately. The lowest factor is not automatically the only priority, but the decomposition directs investigation towards reliability and changeover, cycle-rate losses, or quality causes.

WORKED EXAMPLE 9.1 — OEE CALCULATION FOR A ROBOTIC PALLETISING CELL

Problem: A robotic palletising cell operates on a single 8-hour shift (480 min). Data recorded:

- Planned downtime (scheduled breaks): 30 min
- Unplanned downtime (robot fault + conveyor jam): 45 min
- Changeover time: 15 min
- Ideal cycle time: 12 seconds per box
- Total boxes processed: 1,800
- Rejected boxes (damaged, misaligned): 36

Calculate OEE and identify the primary loss category.

Solution:

Planned Production Time = $480 - 30 = 450$ min

Run Time = $450 - 45 - 15 = 390$ min

Availability = $390 / 450 = 86.7\%$

Ideal output at run time = $390 \times 60 / 12 = 1,950$ boxes

Performance = $1,800 / 1,950 = 92.3\%$

Quality = $(1,800 - 36) / 1,800 = 98.0\%$

$$\text{OEE} = 0.867 \times 0.923 \times 0.980 = 78.4\%$$

Analysis: Primary loss is Availability (86.7%), driven by 45 min unplanned downtime + 15 min changeover. Addressing the root cause of robot faults and improving changeover procedures (SMED) would yield the greatest OEE improvement. Quality at 98% also suggests 2% box damage — gripper design and conveyor interface investigation recommended.

WORKED EXAMPLE 9.2 — OEE IMPROVEMENT SCENARIO ANALYSIS

Problem: The plant manager asks: “If we reduce changeover time from 15 min to 5 min and eliminate the conveyor jam (20 min of the 45 min unplanned downtime), what will our new OEE be?”

Solution:

New unplanned downtime = $45 - 20 = 25$ min

New changeover = 5 min

New Run Time = $450 - 25 - 5 = 420$ min

New Availability = $420 / 450 = 93.3\%$

New ideal output = $420 \times 60 / 12 = 2,100$ boxes

Assuming Performance and Quality remain unchanged: $P = 92.3\%$, $Q = 98.0\%$

New actual production = $2,100 \times 0.923 = 1,938$ boxes (estimated)

New OEE = $0.933 \times 0.923 \times 0.980 = 84.4\%$

Impact: OEE increases from 78.4% to 84.4% (+6.0 percentage points). Estimated total output is 1,938 boxes and estimated good output is $1,938 \times 0.98 \approx 1,899$ boxes, compared with 1,764 good boxes in the baseline: approximately 135 additional good boxes per shift. At a margin of €2/box, the annual benefit is about €67,500 for 250 shifts. With €20,000 total implementation cost, the simple payback is approximately 3.6 months.

INDUSTRIAL PRACTICE NOTE — OEE Benchmarking on a High-Mix SMT Electronics Line

Context. A contract electronics manufacturer running a surface-mount technology (SMT) line for high-mix, low-volume production (12–18 product changeovers per week, batch sizes 200–2 000 units) deployed an MES-integrated OEE measurement framework after observing inconsistent throughput between shifts. Initial baseline measurements over two weeks showed a plant-stated OEE of 65% but a recalculated, decomposed OEE of 51% (Availability 78%, Performance 72%, Quality

WORKED EXAMPLE 9.2 — OEE IMPROVEMENT SCENARIO ANALYSIS

91%) — the gap explained by previously unlogged micro-stoppages and changeover time treated as ‘non-production’.

Approach and outcomes. After segregating each OEE factor into a separate live tile and adding a dedicated changeover timer, the operations team identified that changeover dominated Availability losses (40% of all unscheduled time). A SMED-style rationalisation programme — pre-staging feeders, parallelising stencil cleaning, single-button program switching — reduced average changeover from 38 min to 14 min over four months. Decomposed OEE rose to A 88%, P 79%, Q 93%, giving a true OEE of approximately 65% — equal to the previously self-reported figure but now traceable, repeatable, and comparable shift-to-shift.

Lesson. Two lessons stand out. First, a single-number OEE is meaningful only against a known decomposition; without A, P, Q visibility, improvement programmes target the wrong root cause. Second, the largest gains in high-mix environments typically come from changeover (an Availability lever), not from cycle-time tuning (a Performance lever) — a counter-intuitive finding for engineers trained to chase faster placement rates. Decomposed measurement is what made the right choice obvious.

Illustrative example: details are anonymised and figures are typical of industry-reported ranges; this note is not a citation of a specific public case.

9.4 SCADA Systems for Real-Time Monitoring

Supervisory Control and Data Acquisition (SCADA) provides real-time supervision, alarming, trending, and operator interaction for production equipment. It connects shop-floor control systems to operational information services while preserving the separation between supervisory functions and time-critical machine control.

9.4.1 SCADA Architecture

A practical SCADA architecture can be described through four interacting layers:

- Field level — Sensors, actuators, robot controllers, PLCs. Data is generated here (cycle counts, fault codes, temperatures, positions, energy readings).
- Communication layer — Industrial Ethernet, fieldbus, and interoperability services transport data between devices and supervisory applications. Required update rate, determinism, data quality, redundancy, and cybersecurity must be defined for each interface.

- Supervisory level — SCADA server and HMI that aggregate, process, visualise, and store production data. Operators interact through HMI screens.
- Operations-integration layer — Controlled exchange with MES, historians, reporting services, and selected enterprise applications through engineered interfaces and security boundaries.

9.4.2 Industrial Communication Protocols

Protocol selection should be based on the required communication pattern, timing, information model, installed ecosystem, lifecycle support, and security controls (table 9.4). Link speed alone is not a sufficient selection criterion.

Table 9.4. Industrial Communication Protocols

Protocol	Type	Speed	Key Feature	Typical Use
PROFINET	Industrial Ethernet	100 Mbit/s typical; higher-rate variants	RT/IRT profiles; switched Ethernet	Distributed I/O, motion, Siemens and multi-vendor systems
EtherNet/IP	Industrial Ethernet	Implementation-dependent Ethernet rate	CIP object model and broad ecosystem	Rockwell-centric and multi-vendor automation
EtherCAT	Industrial Ethernet	100 Mbit/s	On-the-fly frame processing and distributed clocks	High-performance motion and synchronised I/O
OPC UA	Interoperability framework	Transport-dependent	Information models, client/server, PubSub, security	SCADA–MES, gateways, structured OT data exchange
MQTT	Application messaging	Transport-dependent	Lightweight publish/subscribe and brokered topics	Edge-to-cloud events and Unified Namespace patterns
PROFIBUS	Fieldbus (legacy)	Up to 12 Mbit/s	Mature installed base	Legacy-system support and migration
IO-Link	Point-to-point device link	Up to 230.4 kbit/s	Smart-sensor data and parameterisation	Sensor/actuator identification and diagnostics

9.4.3 SCADA Functions in Robotic Production

Table 9.5 summarises the principal SCADA functions in robotic production. The system should turn device events and process variables into actionable displays, prioritised alarms, trends, and reports while maintaining a traceable time base and data-quality status.

Table 9.5. SCADA Functions within a robotic production environment

Function	Description	Example
Real-time visualisation	Graphical system status, robot positions, process parameters	Synoptic display with colour-coded station indicators
Alarm management	Detection, recording, prioritisation of abnormal conditions	Robot fault E-4021 triggers alarm, logs timestamp, notifies maintenance
Data acquisition	Continuous collection and storage of process variables	Cycle time, joint temperatures, gripper force logged every cycle
Trend analysis	Historical data visualisation for pattern identification	Motor current trending reveals gradual bearing wear
Report generation	Automated production reports (shift, daily, weekly, OEE)	End-of-shift OEE report by station with loss breakdown
Remote access	Monitoring from remote locations via secure connection	Authorised engineer performs read-only diagnostics through a managed VPN and jump host

9.4.4 HMI Design Principles (High-Performance Graphics)

The effectiveness of SCADA depends strongly on HMI quality. ISA-101 promotes a lifecycle approach based on an HMI philosophy, consistent design rules, implementation, operation, and continuous maintenance. High-performance graphics favour clear hierarchy and colour-by-exception rather than decorative realism:

- Situation awareness — The overview should make normal, abnormal, and uncertain states distinguishable at a glance. Colour meanings must be defined in the site HMI philosophy and applied consistently.
- Neutral visual hierarchy — Use restrained backgrounds and reserve saturated colour for information that requires attention. Shape, position, text, and trend should also convey state so that colour is not the only cue.

- Alarm rationalisation — Apply an alarm-management lifecycle such as ISA-18.2/IEC 62682. Each alarm should be necessary, prioritised, actionable, and linked to a defined operator response and consequence of inaction.
- Navigation hierarchy — Use a consistent path from plant or line overview to area, station, and device detail. The number of navigation steps should be justified by the task rather than imposed as a universal fixed limit.
- Consistent templates — Equivalent stations should use the same object library, status conventions, terminology, and interaction patterns.
- Trend and analogue context — Combine values with limits, rates of change, bar indicators, or sparklines when the pattern matters more than the instantaneous number.
- Eliminate non-informative decoration — Photorealistic equipment, unnecessary animation, and three-dimensional piping can increase cognitive load without improving diagnosis.

Figure 9.3 contrasts a visually busy conventional display with a high-performance display that keeps normal conditions quiet, exposes trends, and reserves strong colour for abnormalities requiring attention.

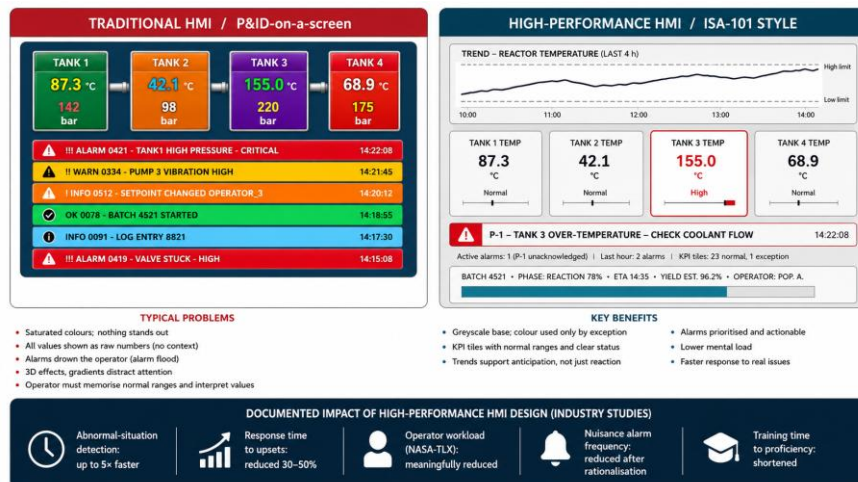


Figure 9.3. Traditional vs. high-performance HMI design philosophies, per ISA-101 and ASM Consortium guidance.

INDUSTRIAL PRACTICE NOTE — SCADA-Driven Shift Visibility in Automotive Body-in-White Assembly

Context. An automotive Tier-1 supplier operating a multi-cell body-in-white welding line — six robotic stations producing approximately 320 vehicle bodies per shift — implemented a unified SCADA layer (OPC UA based) feeding a cloud-hosted MES. Before the upgrade, individual cell HMIs each displayed independent KPIs and shift supervisors compiled daily reports manually from station logs. Latency between an off-spec event and a documented response was typically two to six hours, often exceeding a full shift.

Approach and outcomes. Within the first quarter after deployment, the integrated SCADA dashboard delivered: (i) cell-level OEE updated every 60 s, with automatic Pareto ranking of downtime causes by station; (ii) a cross-cell synchronisation view that highlighted bottleneck migration in real time; (iii) shift-handover reports auto-generated and emailed within 5 min of shift end. Reported outcomes after six months: average shift OEE rose from approximately 71% to 78% (typical industry band for body-in-white lines), unplanned downtime dropped by roughly 30%, and the cycle from anomaly detection to corrective action was compressed from hours to under 15 minutes.

Lesson. The principal lesson is that SCADA deployment value is unlocked by data integration across cells and by closing the loop with operators and supervisors — not by adding more sensors per se. Information available at every level of the ISA-95 pyramid only translates to action when timeliness and presentation match the decision horizon of each role. A second lesson: Pareto-driven dashboards reveal that the same three to five root causes recur across stations, justifying systemic interventions rather than station-by-station fixes.

Illustrative example: details are anonymised and figures are typical of industry-reported ranges; this note is not a citation of a specific public case.

9.5 Manufacturing Execution Systems (MES)

While SCADA provides second-to-minute shop-floor visibility and operator supervision, Manufacturing Operations Management and MES functions operate mainly at ISA-95 Level 3, coordinating work, materials, quality, genealogy, and performance across production resources.

9.5.1 Core MES Functions (ISA-95)

Table 9.6 summarises common MES functions for robotic production. The exact product boundary varies by implementation: some maintenance, quality, or document functions may reside in specialised applications integrated with MES rather than in one monolithic system.

Table 9.6. Core MES (Manufacturing Execution System) Functions

MES Function	Description	Robotic Production Relevance
Production scheduling	Sequencing work orders based on priorities, resources, constraints	Determines product variant sequence, triggering program and tooling changes
Production tracking	Monitoring work order progress through production in real time	Tracks each unit from raw material to finished goods, including which robot processed it
Quality management	Recording inspection results, enforcing quality rules	Links quality deviations to specific stations, programs, and time windows
Performance analysis	Calculating KPIs from production data	Generates OEE reports by shift, product, station, and time period
Maintenance management	Scheduling and tracking maintenance activities	Triggers work orders based on operating hours or condition monitoring data
Material tracking	Tracking materials and consumables through the process	Batch traceability for regulated industries (pharma, food, automotive)
Labour management	Tracking operator assignments, certifications, training	Supports role, competence, and assignment checks for authorised operating tasks
Document management	Providing correct work instructions and drawings	Delivers product-specific setup sheets, quality criteria, and safety procedures

9.5.2 MES–SCADA–ERP Data Integration

Data exchange through the ISA-95 interfaces is bidirectional. Production events, counts, genealogy, and equipment states are contextualised for Level 3 applications; schedules, work instructions, recipes, and authorised parameters are dispatched towards execution. Control commands should remain within the validated control architecture rather than being issued directly by business systems.

OPC UA is widely used for structured OT data exchange, while REST APIs, message queues, and event-streaming platforms are common at enterprise interfaces. Unified Namespace architectures can reduce point-to-point coupling, but they still require

governed topic structures, authoritative data ownership, schema control, security, and lifecycle management.

9.6 Cycle Time Analysis and Production Scheduling

9.6.1 Cycle Time Decomposition

Table 9.7 decomposes cycle time into motion, process, handling, waiting, and tool-change components. The decomposition distinguishes mechanical limits from sequence and coordination losses so that optimisation is directed at the dominant contributor rather than at the most visible motion.

Table 9.7. Structured Cycle Time Decomposition

Component	Description	Optimisation Approach
Robot motion time	TCP movement between positions	Path optimisation, higher acceleration, shorter paths, via-point reduction
Process time	Value-adding operation (welding, assembly, inspection)	Process parameter optimisation, parallel processing, tool upgrades
Load/unload time	Part presentation and removal	Fixture redesign, automated feeding, dual-station (pendulum) loading
Wait/idle time	Waiting for external events	Buffer sizing, line balancing, signal timing optimisation, parallel operations
Tool change time	Automatic tool exchange	Faster changers, reduced tool variety, multi-function EoAT

Bottleneck rule: In a serial line without sufficient parallel capacity, the resource with the greatest effective cycle time constrains steady-state throughput. The active bottleneck may shift with product mix, failures, buffers, changeovers, or parallel routing, so it should be confirmed from operational data.

9.6.2 Line Balancing

Line balancing distributes work elements across stations to minimise idle time and maximise throughput:

$$\text{Line Efficiency} = \frac{\text{Sum}(\text{Station CTs})}{(\text{N\_stations} \times \text{CT\_bottleneck})} \times 100\%$$

A perfectly balanced deterministic line would have 100%-line efficiency. Real systems normally retain some imbalance because tasks are indivisible, process times vary, and technical or safety constraints limit redistribution; therefore, the acceptable target is application specific.

WORKED EXAMPLE 9.3 — TAKT TIME, BOTTLENECK, AND LINE BALANCING

Problem: A robotic assembly line produces electronic modules. Customer demand: 720 units per 8-hour shift (480 min available, 30 min breaks). The line has 5 stations:

- Station 1 (pick & place components): CT = 33 s
- Station 2 (soldering): CT = 42 s
- Station 3 (inspection): CT = 28 s
- Station 4 (assembly): CT = 38 s
- Station 5 (packaging): CT = 31 s

Evaluate: Can the line meet demand? Identify the bottleneck. Calculate line efficiency. Propose improvements.

Solution:

Available time = $480 - 30 = 450 \text{ min} = 27,000 \text{ s}$

Takt time = $27,000 / 720 = 37.5 \text{ s/unit}$

Bottleneck = Station 2 (soldering) at 42 s $\rightarrow \text{CT} > \text{TT}$

Result: The line CANNOT meet demand. Station 2 exceeds takt time by 4.5 s.

Maximum throughput = $27,000 / 42 = 642 \text{ units/shift}$ (vs. 720 required)

Shortfall = $720 - 642 = 78 \text{ units/shift}$

Line efficiency = $(33+42+28+38+31) / (5 \times 42) \times 100\% = 172/210 = 81.9\%$

Improvement options:

(a) Optimise Station 2 robot path + soldering parameters to reduce CT from 42 s to $\leq 37.5 \text{ s}$

(b) Redistribute work: move some soldering preparation to Station 1 (currently 33 s, has 4.5 s capacity)

(c) Add parallel Station 2b for dual-track soldering (effective CT = $42/2 = 21 \text{ s}$)

(d) If option (a) reduces Station 2 to 37 s: new efficiency = $(33+37+28+38+31)/(5 \times 38) = 167/190 = 87.9\%$ and demand is met

9.7 Downtime Analysis and Root Cause Identification

9.7.1 Downtime Classification

Table 9.8 classifies downtime by source and specifies the evidence that should be recorded for each event. The relative contribution of each category must be established from the local event history rather than assumed from generic industry percentages.

Table 9.8. *Downtime Classification*

Category	Examples in Robotic Production	Evidence to Record
Equipment failure	Robot faults, servo drive failures, PLC errors, sensor failures, pneumatic leaks	Fault code, affected asset, start/end time, repair and verification
Changeover & setup	Product changeovers, tool changes, program uploads, calibration	Product transition, tasks, start/end time, first-good-unit confirmation
Material-related	Part jams, material shortages, wrong material, conveyor blockages	Material state, lot, blockage location, replenishment or supplier cause
Quality-related	Stops for quality investigation, rework loops, inspection holds	Deviation code, affected units, disposition, inspection and release time
Organisational	Operator unavailability, shift handover, missing work orders, awaiting engineering	Waiting reason, responsible function, communication and approval delay
External	Utility failures (power, air, water), upstream/downstream stoppages	Utility or upstream/downstream source, duration, dependency and recovery

9.7.2 Pareto Analysis

Pareto analysis ranks causes by a selected impact measure, commonly total downtime minutes, and overlays the cumulative share. The familiar 80/20 statement is a heuristic, not a rule: the useful boundary is the point at which the few highest-ranked causes capture most of the recoverable loss.

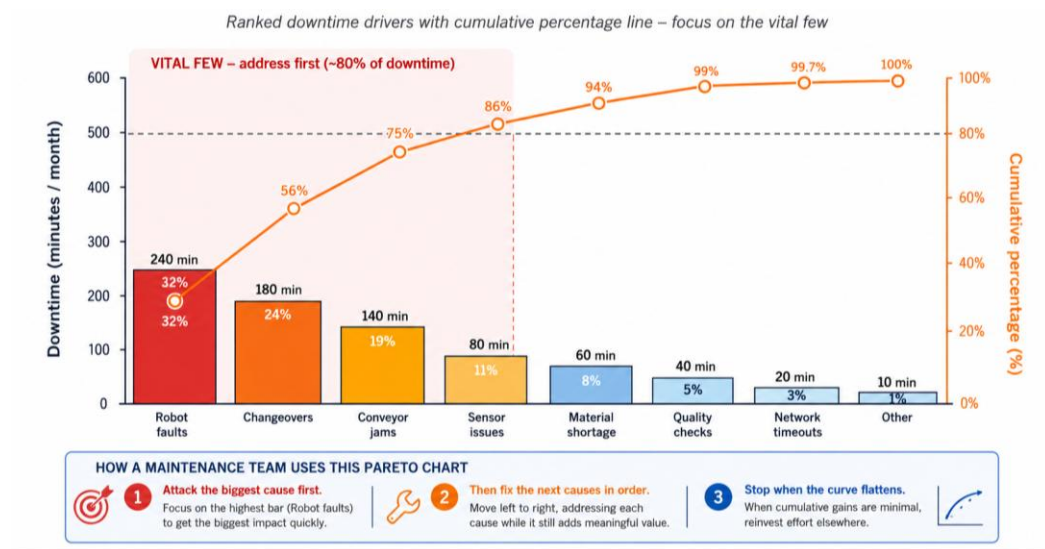


Figure 9.4. Pareto analysis of downtime causes showing the vital few that account for most operational losses.

Strategic Insight

The main value of figure 9.4 is to prevent maintenance by anecdote. The chart shows that four causes account for 80.4% of the recorded downtime in the worked example, providing a defensible starting point for root-cause analysis. After corrective actions, the Pareto chart must be regenerated because the ranking will change.

9.7.3 Root Cause Analysis (RCA) Techniques

Table 9.9 compares complementary root-cause-analysis methods. The 5 Whys is suited to a focused causal chain; Ishikawa supports multidisciplinary cause exploration; Fault Tree Analysis structures deductive logic for critical events; and FMEA examines potential failure modes proactively. The method should match the complexity and consequence of the problem.

Table 9.9. Root Cause Analysis (RCA) Techniques

Technique	Approach	Best For	Example
5 Whys	Progressive questioning from symptom to root cause	Single-event analysis, quick investigations	Robot stops → servo error → overload → gripper blocked → part misaligned → feeder guide worn
Ishikawa (Fishbone)	Categorise causes: Man, Machine, Method, Material,	Brainstorming sessions with	Quality defect mapped across all 6 categories with team input

Technique	Approach	Best For	Example
	Measurement, Environment	multidisciplinary teams	
Fault Tree Analysis	Top-down deductive logic with AND/OR gates	Critical failures, safety-related events	System-level failure decomposition with probability calculations
Failure Mode & Effects Analysis (FMEA)	Systematic evaluation of failure modes, effects, and detection capability	Proactive risk assessment, design review	$RPN = \text{Severity} \times \text{Occurrence} \times \text{Detection}$ for each failure mode

WORKED EXAMPLE 9.4 — DOWNTIME PARETO ANALYSIS AND 5 WHYS

Problem: A robotic packaging cell recorded the following downtime events over one month (20 shifts):

- Conveyor jams: 14 events, total 280 min
- Robot servo faults: 6 events, total 180 min
- Product changeovers: 20 events, total 160 min
- Gripper air leak: 4 events, total 120 min
- Sensor misreads: 22 events, total 88 min
- Material shortage: 3 events, total 60 min
- Other: 8 events, total 32 min

Total downtime: 920 min across 20 shifts. Perform Pareto analysis and apply 5 Whys to the top cause.

Pareto ranking:

1. Conveyor jams: 280 min (30.4%) — cumulative: 30.4%
2. Robot servo faults: 180 min (19.6%) — cumulative: 50.0%
3. Changeovers: 160 min (17.4%) — cumulative: 67.4%
4. Gripper air leak: 120 min (13.0%) — cumulative: 80.4%

Top 4 causes account for 80.4% of total downtime.

5 Whys for #1 (conveyor jams):

Why 1: Why does the conveyor jam? → Boxes pile up at the infeed

Why 2: Why do boxes pile up? → Robot pick cycle is occasionally slower than infeed rate

WORKED EXAMPLE 9.4 — DOWNTIME PARETO ANALYSIS AND 5 WHYS

Why 3: Why is the pick cycle slower? → Vision system takes extra time on certain box orientations

Why 4: Why does vision take longer? → Low-contrast labels on dark boxes cause multiple retries

Why 5: Why are labels low-contrast? → Supplier changed label material 3 months ago

Root cause: Supplier label material change. Corrective action: specify label contrast requirements in supplier quality agreement + adjust vision system lighting/algorithm for new labels.

INDUSTRIAL PRACTICE NOTE — Pareto-Driven Downtime Reduction in Pharmaceutical Blister Packaging

Context. A pharmaceutical packaging line — pre-formed blister sealing followed by carton insertion at approximately 240 packs per minute — suffered chronic short stops averaging 2.3 events per hour, each lasting 30–90 s. The plant manager requested a structured downtime analysis after a 12-week period in which line OEE remained stuck around 62% despite operator training and a doubled spares budget. Each station already produced detailed event logs but no aggregation or prioritisation existed.

Approach and outcomes. A Pareto analysis of one month of event data ranked downtime causes by total minutes lost: foil-feed misalignment 31%, label sensor false rejects 22%, carton magazine empty 17%, gripper slip 9%, plus 15 additional causes accounting for the remaining 21% combined. The team applied 5-Whys to the top three causes only. Foil-feed misalignment traced to a worn tensioner roller on a 6-month replacement cycle (extended to weekly inspection); the sensor issue resolved with a software hysteresis update; carton magazine refills were repositioned upstream into the routine operator round. Within eight weeks, the top three causes contributed less than 25% of the (now smaller) total downtime, and OEE rose from 62% to 74%.

Lesson. The lesson is the inverse of intuition: ranking is more valuable than completeness. The plant had detailed event logs for years but had never produced a Pareto chart — and so engineering effort had been spread thinly across all 18 causes rather than concentrated on the vital few. Once the top three were addressed, the next iteration's Pareto naturally surfaced different drivers, illustrating that downtime reduction is iterative, with the ranking itself shifting as fixes land.

Illustrative example: details are anonymised and figures are typical of industry-reported ranges; this note is not a citation of a specific public case.

9.8 Industry 4.0 and Advanced Monitoring Technologies

Industry 4.0 expands production monitoring from local visualisation towards interoperable data services, edge analytics, fleet-level learning, and governed feedback. Industry 5.0 adds explicit human-centric, resilient, and sustainability objectives. The following sections examine IIoT and edge computing, digital models and twins, cloud and AI analytics, and sustainability monitoring.

9.8.1 IIoT and Edge Computing

The Industrial Internet of Things (IIoT) connects robots, drives, sensors, tools, meters, and information systems through an interoperable data architecture. The engineering objective is not to connect every signal to the cloud, but to make selected information available with defined semantics, quality, timing, ownership, and security for decisions that cross equipment or organisational boundaries.

A practical deployment has three broad layers. Devices generate data at native rates; edge gateways and industrial computers aggregate, timestamp, filter, and calculate features near the source; enterprise or cloud services provide longer-term storage, cross-asset analytics, model training, and integration with MES, ERP, and maintenance systems.

Edge processing is selected when latency, bandwidth, resilience, data sovereignty, or local autonomy make remote processing unsuitable. Time-critical control and safety functions remain in validated controllers; edge analytics may detect anomalies, calculate KPIs, or prepare recommendations, but must not be presented as a substitute for certified safety logic.

High-rate waveform data can dominate network capacity. Edge feature extraction may calculate RMS, crest factor, spectral peaks, envelopes, and event windows, while retaining raw data locally for selected diagnostics. Feature reduction is valuable only if the chosen features preserve the information required for the diagnostic task; this must be verified rather than assumed.

Table 9.10. *Edge-versus-cloud placement of monitoring functions in robotic production systems*

Monitoring function	Typical location	Driver for the choice	Response Horizon
Safety interlocks, emergency stop logic	Device (safety PLC)	Certified functional-safety architecture; bounded total response	Application-specific; verified in the safety calculation
Vibration anomaly detection	Edge gateway	High sampling rate; bandwidth reduction via features	100–500 ms

Monitoring function	Typical location	Driver for the choice	Response Horizon
Cycle-by-cycle OEE calculation	Edge gateway	Local dashboard; autonomy if network is down	1–5 s
Fleet benchmarking across sites	Cloud	Cross-plant aggregation; central data lake	Minutes–hours
ML model training for RUL	Cloud	Compute-heavy; needs data from multiple assets	Batch (daily/weekly)
ML model inference for RUL	Edge (deployed)	Trained in cloud, deployed to edge for low-latency scoring	1–10 s
Energy and sustainability reporting	Cloud / enterprise	Reporting horizon; integration with enterprise data and approved factors	Shift to reporting period

Protocol selection is another architectural dimension. OPC UA (IEC 62541) provides information modelling, client/server services, PubSub, and security mechanisms. Deterministic field-level communication is addressed through OPC UA FX profiles and suitable networks, including TSN, rather than by assuming that every OPC UA connection is deterministic. MQTT complements OPC UA for lightweight publish–subscribe messaging across edge and cloud boundaries.

Cybersecurity must be designed with the monitoring architecture. IEC 62443 organises assets into zones and conduits according to risk and required security levels. Cell, operations, DMZ, and enterprise zones should exchange only authorised flows through controlled gateways; one-way gateways or data diodes are used only where the required direction and consequence analysis justify them.

WORKED EXAMPLE 9.5 — BANDWIDTH SAVINGS FROM EDGE FEATURE EXTRACTION

Problem: A robotic welding cell has 12 sensors: 3 tri-axial accelerometers sampled at 25.6 kHz (bearing analysis), 4 thermocouples at 10 Hz, 3 current transformers at 10 kHz, and 2 acoustic emission probes at 100 kHz. The plant has 20 such cells and a shared 100 Mbit/s uplink. Compare the bandwidth required for raw streaming versus edge feature extraction with a once-per-cycle summary (15 s cycle, 40 features per sensor, 4 bytes per feature).

Raw bandwidth per cell:

Accelerometers: $3 \times 3 \times 25\,600 \times 4 = 922 \text{ kB/s} = 7.4 \text{ Mbit/s}$

Thermocouples: $4 \times 10 \times 4 = 0.16 \text{ kB/s} \approx \text{negligible}$

Current transformers: $3 \times 10\,000 \times 4 = 120 \text{ kB/s} = 0.96 \text{ Mbit/s}$

Acoustic emission: $2 \times 100\,000 \times 4 = 800 \text{ kB/s} = 6.4 \text{ Mbit/s}$

Raw total per cell $\approx 14.7 \text{ Mbit/s}$. Across 20 cells: 294 Mbit/s — nearly $3\times$ the uplink capacity.

Edge-extracted bandwidth per cell:

$12 \times 40 \times 4 = 1\,920 \text{ B}$ per 15 s cycle = $128 \text{ B/s} \approx 1 \text{ kbit/s}$ per cell.

Across 20 cells: $\approx 20 \text{ kbit/s}$, about 0.02% of the uplink.

Result: edge feature extraction reduces upload bandwidth by a factor of approximately 14 000. Raw data remains available locally for forensic analysis; only compact features flow to the cloud — enabling fleet analytics that would be physically impossible with raw streaming.

Industrial practice — Start edge processing at commissioning, not as a retrofit

Retrofitting monitoring to an operating cell can require additional sensors, cabinet space, network segmentation, downtime, and renewed cybersecurity review. During new-cell design, reserve space and power for an industrial gateway, separate safety-related and standard monitoring paths, define the required device interfaces, provide a governed monitoring VLAN or conduit, and document data ownership from commissioning. The design-stage cost is usually easier to control than a retrofit performed under production constraints.

9.8.2 Digital Shadow and Digital Twin

The terms digital model, digital shadow, and digital twin are used inconsistently across industry. In this chapter, a digital model is an offline representation without automated data exchange; a digital shadow has automated one-way data flow from the physical asset to the virtual representation; and a digital twin adds a governed feedback path from the model towards the physical system.

The distinction matters because observation and authority require different assurance. A shadow can support dashboards, playback, diagnostics, and what-if analysis without changing the plant. A twin that proposes or applies setpoints requires validated model accuracy, bounded authority, change control, cybersecurity, and—where safety can be affected—an engineering safety case. Closed-loop autonomy should therefore be introduced only for clearly delimited functions.

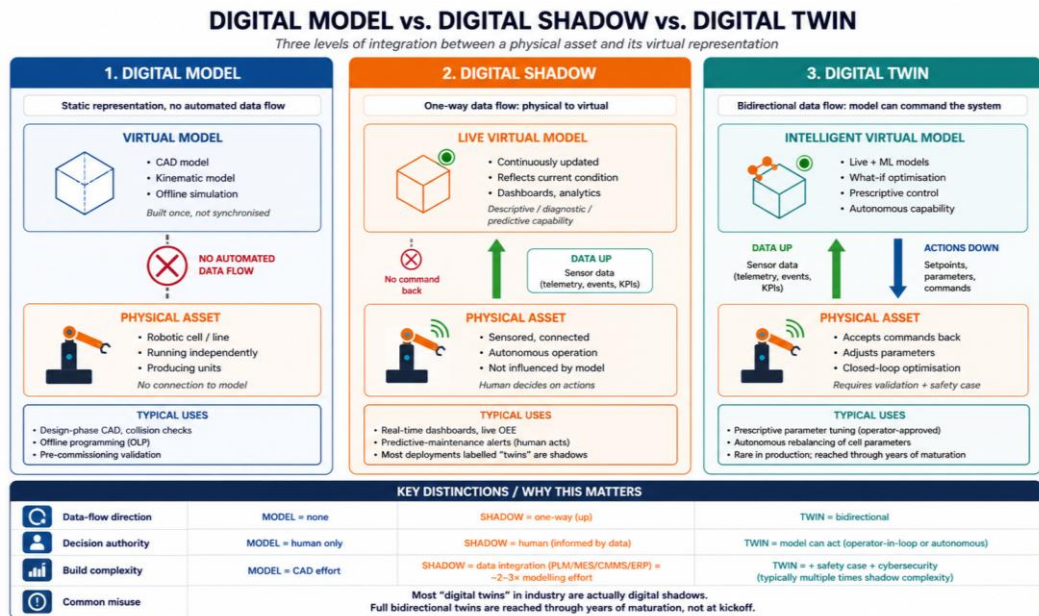


Figure 9.5. Practical distinction between a digital model, a one-way digital shadow, and a digital twin with a validated feedback path.

Twin granularity should follow the decision to be supported. Component representations are suited to condition monitoring; asset-level models support robot or machine lifecycle analysis; cell-level models support sequence, layout, and cycle-time studies; line or plant models support flow, scheduling, and energy balancing. Greater scope increases integration effort and normally reduces the level of physical detail that can be maintained reliably.

Table 9.11. Operational analytics and feedback maturity levels

Level	Capability	Data flow	Typical deliverable
1 — Descriptive	What is happening?	Static model or one-way data, depending on implementation	Live dashboard, cycle-time trace, current OEE

Level	Capability	Data flow	Typical deliverable
2 — Diagnostic	Why did it happen?	One-way, with historical playback and correlation	Downtime root-cause reports, Pareto, drift analysis
3 — Predictive	What is likely to happen?	One-way; model forecasts degradation or failure	RUL alerts, quality-drift warnings, throughput forecast
4 — Prescriptive	What should we do?	Model proposes an action; authorised person or workflow approves it	Recommended setpoints, maintenance schedule, scheduling hints
5 — Autonomous	Act without human in loop	Closed-loop feedback within defined authority and safeguards	Self-tuning control, adaptive trajectories, auto-rebalancing

Table 9.11 separates descriptive, diagnostic, predictive, prescriptive, and autonomous capabilities. These levels are not identical to the model–shadow–twin terminology: a shadow may support several one-way analytics levels, while prescriptive recommendations may still require human approval before any feedback is applied.

A useful production model must be connected to the information needed for its decisions: geometry and configuration from PLM, orders and output from MES, maintenance history from CMMS, and cost or planning context from ERP. Integration and data-governance effort can exceed the modelling effort, so interfaces, identifiers, versioning, and ownership should be planned before visual fidelity is increased.

Industrial practice — Start with a shadow and a single decision

The most common failure mode of a digital-twin initiative is starting with the target state (autonomous optimisation of the whole line) and never reaching a usable product. The successful pattern is to identify a single decision where a descriptive shadow would already create visible value — for example, the daily scheduling meeting that currently relies on verbal handover from the night shift — and build exactly that shadow. Once it is trusted, extend it with predictive features. Only after the predictive layer is trusted should the team consider prescriptive or autonomous closure of the loop. The pace depends on data quality, integration, validation, and organisational readiness; skipping assurance steps usually delays adoption rather than accelerating it.

9.8.3 Cloud Platforms and AI-Driven Analytics

Industrial cloud and enterprise platforms aggregate data from multiple assets or sites for fleet analytics, benchmarking, and machine-learning workflows. Current options include vendor-native services such as Siemens Insights Hub, ABB Ability, and FANUC FIELD; general-purpose services from Microsoft Azure, AWS, and Google Cloud; self-managed open-source stacks; and specialist time-series analytics tools. Product names and service boundaries change rapidly, so architectures should be described by capability and interface rather than by a single vendor label.

Table 9.12. Representative cloud platform families for robotic production

Platform family	Strength	Limitation	Typical fit
Vendor-native (e.g., Siemens Insights Hub, ABB Ability, FANUC FIELD)	Deep integration with native equipment; controller-level data	Harder to use with mixed-vendor fleets	Plants standardised on one vendor
General-purpose cloud and industrial data services (Azure, AWS, Google Cloud)	Broad ecosystem; strong ML/AI tools; enterprise integration	Requires more integration work at the OT boundary	Mixed-vendor plants; enterprises with cloud-first IT
Open-source (Kubernetes + OPC UA)	Full data ownership; no vendor lock-in	Requires in-house engineering capability	Regulated industries, sensitive IP, large engineering teams
Analytics-focused (Seeq, TrendMiner, Uptake)	Purpose-built for time-series analytics and root-cause search	Not a full data platform; needs upstream data source	Overlay on an existing historian or IIoT stack

Machine-learning methods extend monitoring beyond fixed thresholds. Anomaly-detection models learn normal operating patterns; classification models identify known fault classes; regression and survival models estimate continuous quantities such as remaining useful life; temporal models such as LSTM or transformer networks capture sequence effects; and reinforcement learning addresses bounded decision policies where a safe training and deployment framework exists.

A production ML lifecycle comprises governed data acquisition, feature engineering, training and validation, deployment, and ongoing performance monitoring. Models should be versioned with their data, features, thresholds, and intended operating domain. Drift detection, retraining criteria, rollback, and human review distinguish a production system from a one-off demonstration.

Three recurring constraints are label scarcity, concept drift, and explainability. Rare failures produce small and imbalanced labelled datasets; process and product changes invalidate historical relationships; and an unexplained probability score may not support an engineering action. Domain rules, causal checks, uncertainty reporting, and interpretable feature contributions should therefore accompany model outputs.

WORKED EXAMPLE 9.6 — JUSTIFYING A FLEET-LEVEL PREDICTIVE MODEL

Problem: A company operates 40 identical robotic assembly cells across four plants. Each cell experiences on average 2 unplanned bearing failures per year (cost €8 000 per failure including downtime). An ML vendor proposes a fleet-level RUL model trained on 18 months of data, priced at €45 000 one-off plus €12 000/year subscription. Reference cases claim 60% reduction in unplanned bearing failures. Evaluate the business case over 5 years.

Baseline losses:

$40 \text{ cells} \times 2 \text{ failures/year} \times €8\,000 = €640\,000/\text{year}$

Over 5 years: €3 200 000

With model (60% reduction):

Avoided losses/year: $0.6 \times €640\,000 = €384\,000$

Over 5 years: €1 920 000

Total model cost over 5 years:

$€45\,000 + 5 \times €12\,000 = €105\,000$

Plus internal integration effort (est.): €80 000

Total: €185 000

Net benefit over 5 years: €1 920 000 – €185 000 = €1 735 000. Simple payback: €185 000 / €384 000 = ~5.8 months. If the model's effectiveness is only half the claimed rate (30% reduction), payback rises to ~11.6 months — still attractive. Under the stated assumptions the fleet-level case is attractive; the result remains sensitive to true failure cost, achieved detection performance, false alarms, integration effort, and model-maintenance cost.

9.8.4 Energy Monitoring and Sustainability (Industry 5.0)

Industry 5.0 frames industrial development around human-centricity, resilience, and sustainability. Production monitoring consequently includes energy per unit, carbon-equivalent emissions, waste, and resource-efficiency indicators alongside OEE and quality. EU sustainability-reporting requirements were substantially revised in 2026, and legal applicability depends on company scope and national implementation; regardless of reporting status, traceable metering supports internal cost control, customer requirements, and verified environmental claims.

Energy monitoring begins with a defined electrical and utility boundary. Plant meters support billing and high-level reporting; feeder, machine, and sub-component meters support operational diagnosis. Metering equipment and installation should follow the applicable electrical-measurement standards, including relevant parts of IEC 61557, while ISO 14955 provides useful energy-efficiency principles for machine tools that may inform—but do not directly govern—robotic cells.

Table 9.13. *Energy and sustainability KPIs for robotic production under Industry 5.0*

KPI	Definition	Target-Setting Basis	Primary influence
Energy per unit (kWh/unit)	Total cell electrical energy ÷ good units produced	Product- and recipe-specific baseline; verified improvement plan	Trajectories, standby, utilisation
Carbon per unit (kgCO₂e/unit)	Energy per unit × emission factor of electricity mix	Approved electricity emission factor and decarbonisation plan	Grid mix, efficiency, material
Specific energy consumption (SEC)	Energy per kg of product output	Process and product baseline with comparable operating conditions	Process choice, cycle design
Standby share (%)	Energy consumed while not producing ÷ total	Local non-production baseline and operational constraints	Sleep modes, stop-on-idle, scheduling

KPI	Definition	Target-Setting Basis	Primary influence
Material recovery rate	Mass of reused/recycled/remanufactured material ÷ input	Material-specific circularity objective and approved disposition routes	Design-for-disassembly, supplier network
Compressed-air leak rate (%)	Air consumed during idle ÷ total air consumed	Idle-flow baseline, leak survey, and site compressed-air strategy	Seal condition, fitting quality, maintenance

Greenhouse-gas reporting commonly distinguishes Scope 1 direct emissions, Scope 2 purchased energy, and Scope 3 value-chain emissions. Production monitoring directly supports activity data for the applicable scopes, but emission factors, allocation rules, market- versus location-based methods, and reporting boundaries must be version-controlled and approved.

Standby energy should be measured rather than assumed. Robot controllers, servo drives, conveyors, extraction systems, compressed-air equipment, and cell auxiliaries can remain energised during non-productive periods. Controlled sleep or motor-off states, stop-on-idle logic, leak isolation, and scheduling can reduce this load, provided restart time, equipment life, process stability, and safety requirements are evaluated.

Energy-aware trajectory planning trades cycle time, peak joint acceleration, payload, gravity effects, and regenerative opportunities. Savings are strongly application-dependent and should be demonstrated against a measured baseline. Any trajectory or parameter change must remain within process-quality, robot, tooling, and safety constraints.

Waste-stream monitoring should attribute scrap and rework to product, station, time, and cause. The resulting data support Pareto-driven process improvement and circular-economy reporting. Recovery-rate targets must be defined for the material and process rather than copied from a generic best-practice percentage.



Figure 9.6. Energy monitoring dashboard for a robotic cell with Industry 5.0 sustainability KPIs.

Industrial practice — Meter first, optimise second

The most expensive mistake in an energy-reduction programme is launching optimisation projects without measurement. Without a per-cell energy baseline, the engineering team cannot prioritise, cannot quantify savings, and cannot distinguish real improvement from seasonal variation or product-mix changes. A typical sequence is: (1) install sub-metering on all robots, compressors, and major auxiliary loads; (2) collect four to eight weeks of baseline data across representative production; (3) publish a ranked list of the top 10 consumers; (4) attack the top three with targeted interventions; (5) re-measure and update the baseline. Without Step 1 and Step 2, the subsequent steps are guesswork. The business case should include meters, installation, data integration, calibration, maintenance, and the value of the decisions the measurements enable.

KEY TAKEAWAYS — Industry 4.0 and Advanced Monitoring Technologies

- IIoT extends monitoring by unifying previously siloed devices; edge computing is the architectural element that keeps latency low, bandwidth costs manageable, and operation resilient to network outages.
- OPC UA (IEC 62541) and MQTT are the two protocols that dominate modern IIoT stacks. Security follows IEC 62443, organised into zones separated by firewalls and, in the most critical cases, data diodes.
- Digital model, digital shadow, and digital twin denote distinct levels of integration. Most deployments labelled as twins are in fact shadows; full bidirectional twins require validation, safety-case evidence, and cybersecurity controls that most organisations reach only over several years.
- A twin's usefulness is limited more often by data integration (PLM, MES, CMMS, ERP) than by modelling fidelity. Plan the integration work before the modelling work.
- Cloud platforms enable fleet analytics and ML at scales impossible in single-cell deployments. The four common families — vendor-native, general-purpose, open-source, and analytics-focused — each fit different organisational contexts.
- AI in production monitoring succeeds when label scarcity, concept drift, and explainability are addressed explicitly. MLOps maturity is what distinguishes a proof of concept from a production system.
- Industry 5.0 adds sustainability KPIs — energy per unit, CO₂e per unit, standby share, material recovery — that sit alongside OEE. Reporting obligations depend on legal scope, but traceable sustainability metrics are increasingly required by customers, management systems, and regulation.
- The highest-leverage energy interventions are standby management, energy-optimised trajectories, and waste-stream monitoring. Metering the baseline is always the first step; optimising without data wastes engineering effort.

Common Pitfalls in Operation and Monitoring

Pitfall 1: Measuring without acting.

Collecting vast amounts of data but never using it for decisions. Data collection without purpose is a cost, not an investment. Make every metric tied to a defined decision threshold or recurring review meeting.

Pitfall 2: Focusing on OEE as a single number.

OEE is only useful when decomposed into A, P, Q. A single percentage hides the nature of losses and prevents targeted action. Always present OEE decomposed: each factor — Availability, Performance, Quality — must be visible and analysed separately.

Pitfall 3: Ignoring micro-stoppages.

Micro-stoppages are often under-recorded because they do not trigger a formal downtime event. Configure event logic from the process definition—for example, a statistically meaningful cycle-time deviation or repeated blocked/waiting state—rather than applying an arbitrary universal duration or percentage threshold.

Pitfall 4: Poor alarm management.

Alarm floods, nuisance alarms, and poorly defined priorities create alarm fatigue. Apply the ISA-18.2/IEC 62682 lifecycle: rationalise every alarm, define the required operator response and consequence of inaction, monitor alarm-system performance, and manage changes. Alarm-rate targets should be selected from the operating context and documented philosophy.

Pitfall 5: Excluding changeover time from analysis.

Treating changeover as normal and therefore invisible prevents targeted improvement. Record changeover start and finish, product transition, tasks, delays, and first-good-unit confirmation; then apply SMED principles and verify the achieved reduction against the baseline.

Pitfall 6: No baseline measurement.

Improvement cannot be demonstrated without a stable baseline. The required observation period depends on product mix, failure frequency, and seasonality; it should cover representative operating conditions and retain the original definition and raw evidence for later comparison.

Pitfall 7: Conflating cycle time and takt time.

CT is a system property; TT is a demand property. They must be tracked independently. Cycle time captures the system's capability; takt time captures the customer's demand. They must reconcile, but they are computed and reported on different inputs.

Pitfall 8: Overcomplicating HMI screens.

Adding every available signal creates information overload. Effective HMIs apply an HMI philosophy, a consistent hierarchy, restrained colour, and role-specific displays. The

overview should contain only the information required to recognise abnormal conditions and select the next diagnostic view.

Pitfall 9: Ignoring energy consumption.

Energy may be a significant operating cost and is increasingly requested in customer, sustainability, and management reporting. Make metering boundaries, emission factors, and product allocation explicit, and display energy KPIs alongside production context so that apparent improvements are not caused merely by product-mix changes.

Pitfall 10: Not closing the feedback loop.

Monitoring and measurement are only valuable if they lead to management decisions. Regular review meetings (daily, weekly) where KPI trends are discussed and actions assigned are essential. Schedule weekly tier meetings where the top three Pareto items each receive a named owner and a deadline, then re-check at the next cycle whether the metric moved.

KEY TAKEAWAYS

The operational phase is the longest and most economically significant phase — effective monitoring is essential for sustained performance.

OEE = Availability × Performance × Quality. Decomposition enables targeted improvement; the widely cited 85% value is a heuristic, not a universal standard.

Takt time is set by demand; cycle time is a system property. CT must be ≤ TT. The bottleneck station determines maximum line throughput.

SCADA provides real-time visibility (Level 2); MES manages production execution (Level 3); ERP handles business planning (Level 4).

High-performance HMI design uses grey backgrounds, colour-by-exception, alarm rationalisation, and consistent templates.

Pareto analysis identifies the vital few downtime causes; RCA techniques (5 Whys, Ishikawa, FTA, FMEA) determine root causes.

Station FPY measures first-pass conformance at one operation; rolled throughput yield compounds the first-pass yields across a line.

Industry 4.0 technologies (IIoT, edge computing, digital twin, cloud analytics) transform monitoring from retrospective to predictive.

Energy per Unit (EPU) is an increasingly critical KPI driven by sustainability requirements and Industry 5.0.

The value of monitoring is in the decisions it enables, not in the data collected. Close the feedback loop.

Review Questions

1. Define OEE and explain its three constituent factors. A system has $A=88\%$, $P=94\%$, $Q=99\%$. Calculate OEE. Which factor offers the greatest improvement potential?
2. A robotic welding cell operates for 480 min/shift with 20 min planned breaks. Unplanned downtime totals 50 min, changeover takes 10 min. Ideal CT is 45 s, actual production is 480 units, 12 are rejected. Calculate OEE and identify the primary loss.
3. Explain the difference between cycle time and takt time. A line has $CT = 40$ s and $TT = 35$ s. What are the consequences and possible solutions?
4. Describe the four layers of a SCADA system. What role does each play in monitoring a robotic production line?
5. List six core MES functions per ISA-95. How does MES differ from SCADA in scope and time horizon?
6. A 5-station robotic assembly line has CTs of 28, 32, 30, 35, and 29 seconds. $TT = 33$ s. Identify the bottleneck, determine if demand can be met, calculate line efficiency, and propose a rebalancing strategy.
7. Explain the Pareto principle as applied to downtime analysis. A cell has 5 downtime causes with durations 200, 150, 100, 40, and 10 min. Which causes should be prioritised?
8. Describe four HMI design principles per the high-performance graphics methodology. Why is alarm rationalisation important?
9. Compare digital shadow and digital twin. How does each contribute to production monitoring? Give an example of a prescriptive digital twin application in robotic production.

References

- [1] Groover, M.P., Automation, Production Systems, and Computer-Integrated Manufacturing, 5th ed., Pearson, 2020, Chapters 25–26.

- [2] Nakajima, S., Introduction to TPM: Total Productive Maintenance, Productivity Press, 1988.
- [3] Hansen, R.C., Overall Equipment Effectiveness: A Powerful Production/Maintenance Tool for Increased Profits, Industrial Press, 2001.
- [4] ISA-95 / IEC 62264 — Enterprise-Control System Integration, current published parts.
- [5] ANSI/ISA-101.01-2015 — Human Machine Interfaces for Process Automation Systems.
- [6] Hollifield, B. and Habibi, E., The High Performance HMI Handbook, PAS / Plant Automation Services, 2010.
- [7] Siemens, SIMATIC WinCC / WinCC Unified — SCADA and HMI documentation, accessed 2026.
- [8] Muchiri, P. and Pintelon, L., 'Performance Measurement Using OEE', Int. J. Production Research, 46(13), 2008, pp. 3517–3535.
- [9] Xu, X. and Lu, Y., Industry 4.0 and Smart Manufacturing, Elsevier, 2021.
- [10] ABB Ability — Connected Services for Robotics, ABB Documentation, 2024.
- [11] FANUC FIELD System — IoT Platform for Manufacturing, FANUC Technical Documentation, 2023.
- [12] European Commission, Industry 5.0 — Towards a Sustainable, Human-Centric and Resilient European Industry, 2021.
- [14] IEC 62541 series — OPC Unified Architecture.
- [15] IEC 62443 series — Security for Industrial Automation and Control Systems.
- [16] ANSI/ISA-18.2 / IEC 62682 — Management of Alarm Systems for the Process Industries.
- [17] Directive (EU) 2026/470 amending corporate sustainability reporting and due-diligence requirements.
- [18] GHG Protocol, Corporate Accounting and Reporting Standard and Scope 2 Guidance.
- [13] Mocan, B., Design and Operation of Robotic Production Systems — Course Notes, 2025–2026.

CHAPTER

10

Maintenance, Reliability, and Lifecycle Engineering

From reactive repairs to predictive intelligence

This chapter closes Module III by addressing the long-term perspective of robotic production systems. While Chapter 8 ensured the system is safe and Chapter 9 showed how to measure its performance, this chapter asks: how do we keep the system performing reliably over its entire useful life? The answer lies at the intersection of reliability engineering, maintenance strategy, and lifecycle thinking — disciplines that determine whether a robotic investment delivers sustained value or progressively deteriorates into an expensive source of unplanned downtime.

LEARNING OBJECTIVES

1. Define and calculate key reliability metrics: MTBF, MTTR, and availability
2. Calculate system-level reliability for series configurations and explain the assumptions required for active redundancy
3. Compare maintenance strategies from reactive to predictive and explain their trade-offs
4. Describe condition monitoring techniques applicable to robotic production systems
5. Apply Reliability-Centred Maintenance (RCM) principles for strategy selection
6. Explain Total Cost of Ownership (TCO) and its components for robotic systems
7. Identify the lifecycle phases and their engineering implications
8. Evaluate how digital twins support predictive maintenance and lifecycle management
9. Apply the bathtub curve model to interpret failure rate patterns
10. Develop a spare parts criticality analysis and structured maintenance plan

10.1 Reliability Engineering Fundamentals

Reliability is the probability that a system will perform its intended function without failure for a specified period of time under stated operating conditions. For robotic production systems, reliability directly translates into availability — the proportion of planned production time during which the system is actually capable of producing.

10.1.1 Mean Time Between Failures (MTBF)

MTBF is the average time elapsed between consecutive failures of a repairable system during its useful-life period:

$$\text{MTBF} = \text{Total Operating Time} / \text{Number of Failures}$$

For example, if a robotic welding cell operates for 4,000 hours during a year and experiences 8 failures, its MTBF is 500 hours. MTBF is a statistical average and does not predict when the next failure will occur — it characterises the overall reliability of the system population.

10.1.2 System Reliability: Series and Redundant Configurations

For a series system with statistically independent components and approximately constant failure rates, component failure rates may be added. This approximation is useful during the useful-life period, but it is not a general rule for dependent failures, non-exponential lifetimes, common-cause failures, or repairable redundant architectures.

Series configuration — all components must operate for the cell to produce:

$$1/\text{MTBF}_{\text{system}} = 1/\text{MTBF}_1 + 1/\text{MTBF}_2 + \dots + 1/\text{MTBF}_n$$

Active parallel redundancy — the system continues while at least one of n identical, independent elements remains functional:

$$\text{MTTF}_{\text{parallel}} = \text{MTBF}_{\text{component}} \times (1 + 1/2 + 1/3 + \dots + 1/n) \quad [\text{non-repairable mission, exponential lifetimes}]$$

For two identical elements, the mission mean time to failure is 1.5 times the individual mean under this simplified model. Repairable redundancy requires a state-space or Markov availability model that includes repair rates, switching behaviour, diagnostics and common-cause failure. Dual-channel functional-safety architectures are designed for fault tolerance and diagnostic coverage; they should not be described by this simple production-redundancy equation.

WORKED EXAMPLE 10.1 — SYSTEM MTBF FOR A ROBOTIC CELL

Problem: A robotic palletising cell consists of the following components in series (any failure stops the cell):

- Robot: MTBF = 40,000 h
- Robot controller: MTBF = 60,000 h
- Conveyor system: MTBF = 15,000 h
- Gripper assembly: MTBF = 8,000 h
- Safety PLC: MTBF = 200,000 h
- Sensors (5 units, each MTBF = 50,000 h): $\text{MTBF}_{\text{sensors}} = 50,000/5 = 10,000 \text{ h}$

Calculate the system MTBF. If the cell operates 4,000 h/year, how many failures per year are expected?

Solution:

$$\begin{aligned} 1/\text{MTBF}_{\text{sys}} &= 1/40,000 + 1/60,000 + 1/15,000 + 1/8,000 + 1/200,000 + 1/10,000 \\ &= 0.0000250 + 0.0000167 + 0.0000667 + 0.000125 + 0.0000050 + 0.000100 \\ &= 0.0003383 \end{aligned}$$

$$\text{MTBF}_{\text{system}} = 1/0.0003383 = 2,956 \text{ hours}$$

$$\text{Expected failures per year} = 4,000 / 2,956 = 1.35 \text{ failures/year}$$

Analysis: The gripper assembly (MTBF = 8,000 h) is the dominant contributor to system unreliability, followed by sensors (10,000 h effective) and the conveyor (15,000 h). Improvement priority: (1) upgrade gripper design or implement CBM for gripper; (2) use higher-reliability sensors or add redundancy; (3) implement preventive maintenance for conveyor drive components.

10.1.3 Mean Time to Repair (MTTR)

Mean Time to Repair (MTTR) is the average active corrective-maintenance time needed to diagnose, repair, test and return a failed item to a serviceable state. Organisations should define the start and end events explicitly so that the metric is comparable across assets and reporting periods:

$$\text{MTTR} = \text{Total Active Corrective-Maintenance Time} / \text{Number of Repairs}$$

Logistics delay, waiting for personnel, administrative delay and production restart may be reported separately as Mean Down Time or Mean Time to Restoration. Combining them silently into MTTR makes benchmarking misleading. Both active repair time and total restoration time matter: a technically simple repair can still cause a long outage when access, spares or approvals are poor.

Table 10.1 separates engineering and support factors that influence active repair and total restoration time.

Table 10.1. Factors influencing Mean Time to Repair (MTTR) in robotic systems.

Factor	Impact on MTTR	Improvement Strategy
Diagnostic capability	Rich fault information and event history shorten diagnosis	Use robot controllers with diagnostic logging; train technicians on fault code interpretation
Spare parts availability	Waiting for parts can add days to repair time	Critical spares analysis and on-site inventory for high-impact components
Maintenance skill level	Competence affects diagnosis, repair quality and repeatability	Manufacturer certification programmes; hands-on training on actual equipment
System accessibility	Poor access, identification or modularity extends active repair time	Design for maintainability: access panels, cable labelling, modular components
Remote diagnostics	Secure expert support can shorten diagnosis and restoration	Secure remote access infrastructure; service contracts with remote support SLA
Documentation quality	Controlled, current procedures reduce error and rework	Maintain up-to-date maintenance manuals, electrical schematics, and fault trees

10.1.4 Availability

For a repairable item with constant failure and repair rates, inherent availability combines corrective-maintenance reliability and maintainability:

$$\text{Inherent Availability (Ai)} = \text{MTBF} / (\text{MTBF} + \text{MTTR}) \times 100\%$$

Inherent availability excludes preventive maintenance, logistics and administrative delay. Operational availability should be calculated directly from observed time: $A_o = \text{Uptime} / (\text{Uptime} + \text{Total Downtime})$. For Worked Example 10.1, $\text{MTTR} = 4 \text{ h}$ gives $A_i = 2,956 / (2,956$

+ 4) = 99.86%. High inherent availability does not guarantee high production availability when changeovers, upstream starvation, maintenance or organisational delays are significant.

WORKED EXAMPLE 10.2 — MTBF, MTTR, AND AVAILABILITY FROM FIELD DATA

Problem: A robotic packaging cell operates 16 h/day, 250 days/year (4,000 h/year). Past year failure data:

- Failure 1: Robot servo fault — 4 h downtime
- Failure 2: Conveyor motor failure — 8 h downtime
- Failure 3: Gripper pneumatic leak — 2 h downtime
- Failure 4: Safety scanner calibration loss — 3 h downtime
- Failure 5: PLC communication timeout — 1 h downtime

Calculate MTBF, MTTR, and inherent availability. Identify the priority for improvement.

Solution:

Total operating time = 4,000 – (4+8+2+3+1) = 3,982 h

Number of failures = 5

MTBF = 3,982 / 5 = 796.4 hours

Total repair time = 18 h; **MTTR = 18 / 5 = 3.6 hours**

$A_i = 796.4 / (796.4 + 3.6) \times 100\% = 99.55\%$

Observed operational availability for this data set = 3,982 / 4,000 × 100% = 99.55% because the only recorded downtime is corrective repair.

Priority: The conveyor motor event is the largest restoration loss. Stocking a compatible motor and prepared change procedure primarily reduces restoration

time; vibration trending may increase warning time and reduce the probability of an unplanned failure. Both effects should be verified using subsequent field data.

10.2 The Bathtub Curve and Failure Patterns

The bathtub curve is a conceptual model of how hazard rate may evolve through early life, mature service and ageing. It is not a universal calendar for every component: electronics, software, seals, cables and bearings can exhibit different patterns. The model is useful only when supported by component-specific field or test data.

Table 10.2. *Conceptual phases of the bathtub curve and corresponding engineering responses.*

Phase	Indicative lifecycle region	Failure-rate tendency	Typical causes	Engineering response
Infant mortality	Early-life / commissioning	Decreasing	Manufacturing, installation, configuration and commissioning defects	Verification, controlled burn-in where justified, warranty follow-up, early-life monitoring
Useful life	Mature service	Approximately constant (exponential model)	Random failures, external events, operational variation and residual defects	Condition monitoring, corrective maintenance, diagnostics and spares management
Wear-out	Ageing / end-of-life	Increasing	Wear, fatigue, corrosion, insulation and seal degradation, obsolescence	Condition- or age-based replacement, retrofit planning and decommissioning decisions

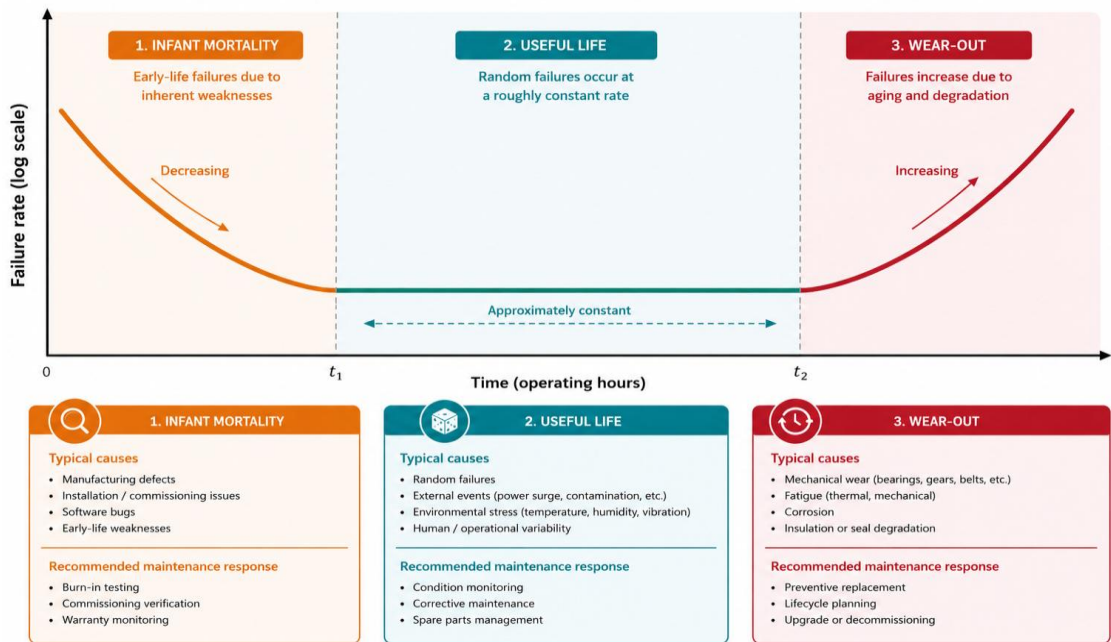


Figure 10.1. Conceptual bathtub curve showing decreasing early-life hazard, an approximately constant useful-life region, and increasing wear-out hazard.

10.2.1 Failure Patterns in Robotic Systems

To ensure the long-term reliability of a robotic work cell, maintenance strategies must be differentiated based on the unique failure characteristics of each subsystem. Unlike simpler machinery, a robot is a complex integration of mechanical, electronic, and pneumatic components, each of which degrades according to different physical principles. While some parts follow a predictable lifecycle, others fail without warning or suffer from "invisible" degradation such as software memory leaks or sensor drift. Understanding these **Failure Patterns** is a prerequisite for transitioning from reactive "firefighting" to a proactive, reliability-centred maintenance (RCM) framework.

Table 10.3 links component failure behaviour to monitoring and replacement strategy. The purpose is to avoid applying age-based maintenance where failures are predominantly random, while using condition or usage information where degradation is observable.

Monitoring technique must match the physical failure mechanism: vibration and lubricant analysis suit rotating elements; thermography and electrical signatures suit

drives and connections; calibration and signal-quality checks suit sensors. The selected indicator must be repeatable, trendable and linked to a maintenance action.

Table 10.3. Failure patterns by component type in robotic systems.

Component Type	Failure Pattern	Monitoring Approach	Replacement Strategy
Mechanical (bearings, gears, belts, seals)	Classic bathtub with clear wear-out	Vibration analysis, temperature, visual inspection	Preventive/condition-based replacement
Electronic (servo drives, power supplies, PCBs)	Predominantly random; some infant mortality	Current monitoring, thermal imaging, error log analysis	Corrective replacement + critical spares
Software / firmware	No physical wear; triggered by specific conditions	Error logs, watchdog counters, memory monitoring	Firmware updates, periodic restarts
Pneumatic (cylinders, valves, hoses)	Wear-out driven by seal degradation	Pressure/flow monitoring, cycle counting	Preventive replacement at cycle count threshold
Sensors (encoders, proximity, vision, F/T)	Random failure + gradual drift	Periodic calibration verification, signal quality monitoring	Calibration-based; replace on drift detection

10.2.2 The Weibull Distribution

The Weibull distribution is the most versatile tool for modelling failure data in reliability engineering. It can represent all three phases of the bathtub curve through a single shape parameter β :

- $\beta < 1$: Decreasing failure rate (infant mortality) — failures decrease over time as weak units are eliminated
- $\beta = 1$: Constant failure rate (useful life) — the Weibull reduces to the exponential distribution, failures are random and memoryless
- $\beta > 1$: Increasing failure rate (wear-out) — failures increase over time as components degrade
- $\beta \approx 3\text{--}4$: Pronounced wear-out behaviour — often associated with fatigue-like mechanisms and a comparatively concentrated characteristic life

In practice, Weibull parameters must be fitted with confidence bounds and checked for censoring, mixed populations and changing duty. When $\beta < 1$, indiscriminate age replacement can reintroduce early-life failures. When $\beta > 1$, planned replacement may be justified if the economic and safety consequences support it; β alone does not define the interval.

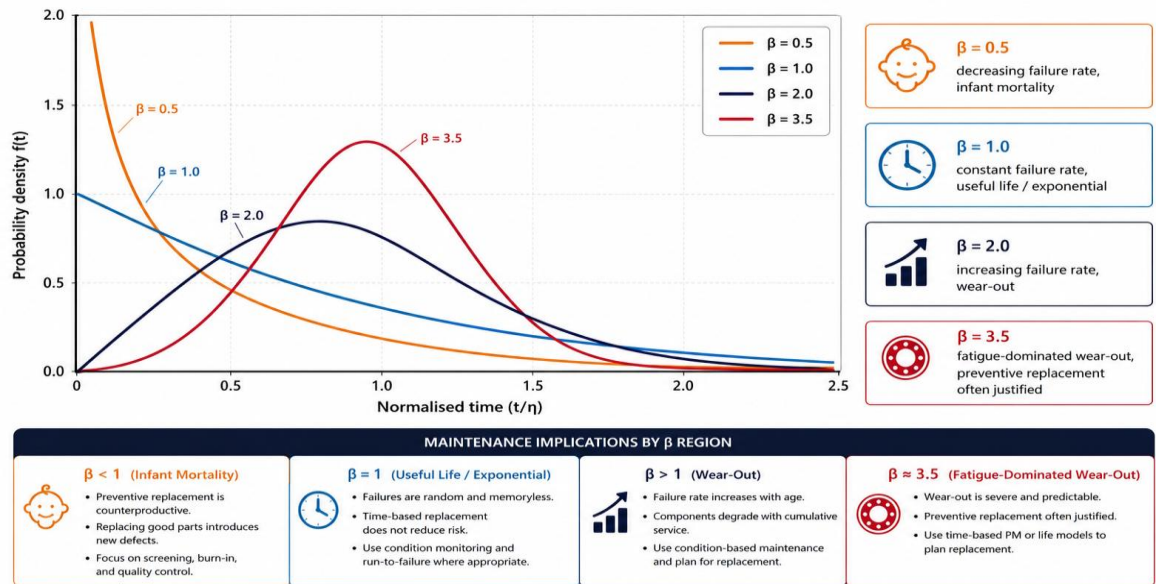


Figure 10.2. Weibull probability density functions for representative shape parameters: $\beta = 0.5$ (decreasing hazard), $\beta = 1.0$ (exponential), $\beta = 2.0$ (increasing hazard), and $\beta = 3.5$ (pronounced wear-out).

10.3 Maintenance Strategies

10.3.1 Reactive Maintenance (Run-to-Failure)

Run-to-failure is a deliberate policy in which corrective action begins after functional failure. It can be appropriate for inexpensive, non-safety-critical items with short replacement time, readily available spares and negligible secondary consequences. It is not an absence of planning: the failure mode, response, spare and restoration procedure should still be defined.

Strategic Profile

- **Advantages:**
 - Low planned-maintenance overhead: no condition-monitoring system or periodic replacement task is required for the selected failure mode.

- High utilisation of component life: replacement follows functional failure rather than a conservative calendar interval.
- Simple decision logic: the strategy needs a controlled corrective response, suitable spares and competent diagnosis rather than advanced prognostics.
- **Disadvantages:**
 - Unplanned restoration: failure timing is uncertain and can interrupt production at the least favourable moment.
 - Secondary damage: an uncontrolled primary failure can damage adjacent components and extend restoration time.
 - Safety and compliance limits: run-to-failure is unacceptable where loss of function can create an intolerable risk or non-compliance.

Pragmatic Application

The decision should follow a consequence-of-failure analysis. Run-to-failure is acceptable only when safety, environmental, quality and production consequences remain tolerable and the restoration path is controlled.

Typical Use Cases:

- **Indicator Components:** Light bulbs, LEDs, and HMI backlights.
- **Non-Critical Sensors:** Redundant sensors where the process can continue safely without immediate data.
- **High-Volume Consumables:** Drill bits or sandpaper where wear is highly variable but the part is inexpensive and quick to swap.
- **General Facilities:** Office HVAC filters or restroom fixtures within the factory site.

10.3.2 Preventive Maintenance (Time-Based)

Time- or usage-based maintenance performs specified tasks at planned intervals. Table 10.4 gives illustrative activities; the governing interval is the current OEM instruction, duty profile, legal requirement and risk-based maintenance plan.

Table 10.4. *Illustrative time- and usage-based maintenance activities.*

Component	PM Activity	Interval basis / example
Robot joints	Grease replenishment (specified type and quantity per manufacturer)	OEM operating-hour schedule; duty and environment dependent
Robot batteries	Backup battery replacement (encoder position memory)	OEM date or alarm-based interval; model dependent
Timing belts / cables	Inspection, tension check, replacement	Inspection and replacement per OEM condition / service schedule
Pneumatic filters	Replace elements, drain moisture traps	Pressure-drop indicator, contamination and documented inspection interval
Conveyor belts	Tension adjustment, surface inspection, alignment	Inspection frequency based on duty, loading and observed wear
Safety devices	Functional verification of E-stops, light curtains, scanners	Proof-test / inspection interval defined by risk assessment, safety-function design, OEM instructions and applicable regulation
Electrical connections	Retorque terminals, inspect cables, check grounding	Thermography / inspection interval based on load, environment and maintenance plan
Vision system	Lens cleaning, illumination check, calibration verification	Cleaning and calibration based on contamination, process capability and drift data

10.3.3 Condition-Based Maintenance (CBM)

Condition-based maintenance uses measured condition and operating context to trigger inspection or intervention. Alert thresholds must be asset-specific, repeatable and linked to an action; Table 10.5 therefore presents decision criteria rather than universal limits.

Table 10.5. *Condition-based maintenance techniques.*

Technique	What It Detects	Application	Decision criterion example
Vibration analysis	Bearing wear, gear damage, imbalance, misalignment	Robot joints, conveyor motors, gearboxes	Trend, alarm zones and machine class defined using baseline data and the applicable ISO 20816 part; no universal 4.5 mm/s limit
Thermal imaging (IR)	Overheating connections, motor hot spots, friction	Electrical cabinets, servo motors, pneumatics	Temperature rise relative to comparable load, ambient and a validated baseline
Current signature	Motor winding degradation, rotor defects, load imbalance	Robot servos, conveyor drives, pump motors	Deviation in current, harmonics or torque signature relative to a known healthy operating envelope

Technique	What It Detects	Application	Decision criterion example
Oil/grease analysis	Contamination, metal particles, viscosity change	Robot joint lubricant, gearbox oil	Particle count, wear metals, moisture and viscosity assessed against lubricant, gearbox and laboratory guidance
Acoustic emission	Micro-cracking, compressed air leaks, valve chattering	Structural parts, pneumatic systems	Leak or impact energy above a validated baseline and confirmed by location / repeatability
Encoder drift	Position accuracy degradation, encoder bearing wear	Robot encoders, conveyor position tracking	Position or signal-quality drift beyond process capability, calibration or OEM tolerance

10.3.4 Vibration Analysis in Detail

Vibration analysis is the most widely used CBM technique for rotating machinery in robotic systems. Different fault types produce characteristic frequency signatures (table 10.6):

Table 10.6. *Vibration analysis — fault types and characteristic frequencies.*

Fault Type	Characteristic Frequency	Vibration Pattern	Severity Indication
Imbalance	1× RPM	Sinusoidal at shaft speed	Proportional to vibration amplitude
Misalignment	Often 1× and/or 2× RPM	Axial and radial components; phase relationship is important	Trend, phase and direction distinguish misalignment from imbalance or looseness
Bearing inner race defect	BPFI (Ball Pass Freq Inner)	Modulated by shaft speed	Increasing amplitude with crack growth
Bearing outer race defect	BPFO (Ball Pass Freq Outer)	Steady amplitude, fixed position	Amplitude proportional to defect size
Gear mesh fault	GMF (n × RPM)	Sidebands around gear mesh frequency	Increasing sideband amplitude
Looseness	Multiple harmonics of RPM	Subharmonics ($\frac{1}{2}\times$, $\frac{1}{3}\times$ RPM)	Truncated waveform, broad spectrum

Figure 10.3 illustrates a diagnostic comparison rather than a universal alarm template. A healthy reference is compared with a simulated outer-race defect showing BPFO harmonics, sidebands and an elevated noise floor. Interpretation requires the bearing geometry, rotational speed, sensor location, load and a repeatable baseline.

Technical interpretation

- Healthy reference: a low noise floor and a dominant rotational component provide the baseline for the stated operating condition.
- Defect example: periodic impacts at BPFO and its harmonics, with modulation sidebands and rising broadband energy, support an outer-race diagnosis when confirmed by bearing geometry and operating speed.
 - BPFO harmonics indicate repeated rolling-element impacts at the outer-race defect frequency.
 - A rising noise floor can indicate increasing impact energy, looseness or lubrication deterioration and should be trended.
 - Sidebands reveal modulation by shaft speed or load variation and strengthen the diagnostic interpretation.

Maintenance decision

The spectrum supports condition-based and predictive decisions only when the fault signature is confirmed, severity is trended and the intervention threshold is linked to consequence and lead time. A planned inspection or replacement should be followed by a repeat measurement to verify that the signature has returned to the expected baseline.

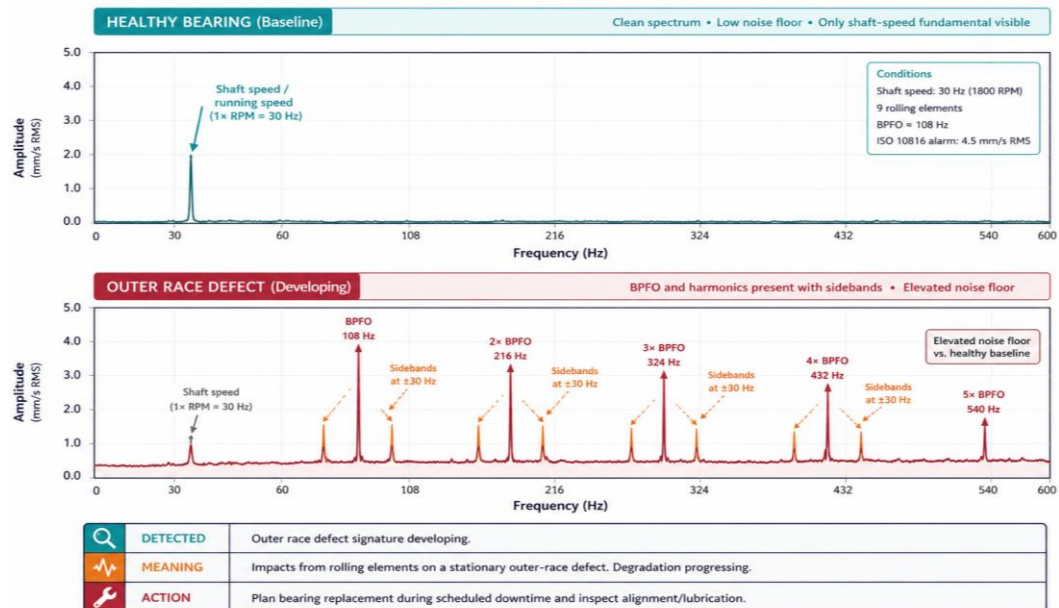


Figure 10.3. Vibration spectrum comparison: healthy bearing (top, clean spectrum) vs. bearing with outer race defect (bottom, showing BPFO harmonics and increased noise floor).

10.3.5 Predictive Maintenance (PdM)

Predictive maintenance extends condition monitoring by estimating future health, failure probability or remaining useful life (RUL). Predictions should include uncertainty and must be converted into an explicit maintenance decision rule:

- Data acquisition — Continuous sensor data collection (vibration, temperature, current, pressure, cycle counts) through IIoT sensors and edge devices.
- Feature extraction — Transforming raw signals into features (RMS level, frequency peaks, temperature rate-of-change, statistical moments) that correlate with degradation.
- Model training — Using historical data with labelled failure events to train models mapping feature patterns to Remaining Useful Life (RUL) estimates.
- Decision support — Presenting RUL predictions through dashboards, triggering work orders when RUL drops below threshold.

Commercial robot-service platforms illustrate different maturity levels: ABB Connected Services and Condition-Based Maintenance provide monitoring, diagnostics and trend-based support, while FANUC Zero Down Time uses connected monitoring and predictive analytics. Product coverage and claimed benefits are vendor- and contract-specific; the cell-level maintenance case remains the user's responsibility.

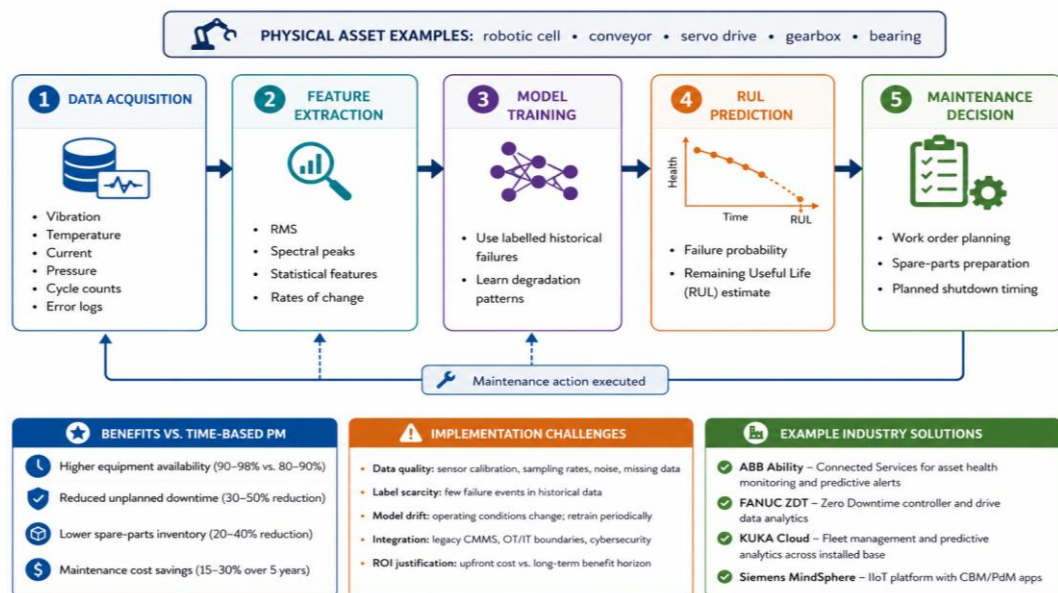


Figure 10.4. Predictive-maintenance workflow from data acquisition and feature engineering through model development, health/RUL estimation, and a governed maintenance decision.

10.3.6 Reliability-Centred Maintenance (RCM)

RCM is a structured methodology for determining the optimal maintenance strategy for each component based on its failure consequences and failure patterns. The RCM decision logic asks seven questions:

- 1. What are the functions and performance standards of the asset in its operating context?
- 2. In what ways can it fail to fulfil its functions (functional failures)?
- 3. What causes each functional failure (failure modes)?
- 4. What happens when each failure occurs (failure effects)?
- 5. In what way does each failure matter (failure consequences)?
- 6. What can be done to predict or prevent each failure (proactive tasks)?
- 7. What should be done if a suitable proactive task cannot be found (default actions)?

Table 10.7 applies RCM consequence logic. Hidden and safety-related failures require failure-finding, risk reduction or redesign, whereas operational and non-operational failures are managed according to production consequence and economic justification.

Table 10.7. Maintenance Decision Framework Based on Failure Consequences

Consequence Category	Description	Preferred Strategy
Hidden failure	Failure is not evident to operators under normal conditions	Scheduled failure-finding task at a justified interval (for example, proof testing of a dormant protective function)
Safety / environmental	Failure could cause injury or environmental damage	Proactive task that reduces risk to tolerable level; redesign if no task is effective
Operational	Failure affects production output, quality, or customer service	Proactive task justified if cost < consequence; otherwise run-to-failure
Non-operational	Failure causes only direct repair cost, no production impact	Run-to-failure unless repair cost is very high

Key RCM insight: not every failure mode benefits from scheduled replacement. The selected task must be technically effective, applicable at the chosen interval and justified by the consequence of failure. Where no proactive task is effective, the appropriate default may be redesign, failure-finding or controlled run-to-failure.

10.3.7 Maintenance Strategy Comparison

Table 10.8 compares the decision logic and information requirements of four maintenance strategies. It is not a ranking: the economically correct strategy depends on failure behaviour, detectability, consequence and restoration capability.

Table 10.8. Maintenance Strategy Comparison

Criterion	Reactive	Preventive	Condition-Based	Predictive
Trigger	Failure occurs	Fixed schedule	Degradation detected	Failure predicted
Downtime type	Unplanned	Planned	Planned	Planned (optimised)
Use of component life	Potentially high, but failure is unplanned	May replace early if interval is conservative	Intervention follows measured degradation	Intervention follows forecast with stated uncertainty
Sensor investment	None	None	Moderate	High

Criterion	Reactive	Preventive	Condition-Based	Predictive
Data/analytics need	None	Low	Moderate	High (ML/AI)
Best for	Non-critical, cheap parts	Known wear-out patterns	Critical with measurable degradation	High-value, high-consequence
Cost profile	Low planned cost; potentially high consequence cost	Predictable planned cost; risk of over-maintenance	Monitoring cost balanced against avoided failures	Highest analytics maturity; justified only by sufficient value

Strategic Insight

Moving from reactive to predictive maintenance increases data, competence and governance requirements. It does not automatically reduce lifecycle cost. Predictive maintenance is justified when degradation is observable, predictions are accurate enough for the decision horizon, and avoided consequences exceed sensing, integration and model-lifecycle costs.

10.4 Total Cost of Ownership (TCO)

The true cost of an industrial robot extends far beyond the initial invoice. **Total Cost of Ownership (TCO)** is a comprehensive financial metric that accounts for every dollar spent from the moment of purchase to the eventual removal of the asset. The mathematical model for this is expressed as:

$$\text{TCO} = C_{\text{acquisition}} + C_{\text{installation}} + C_{\text{operation}} + C_{\text{maintenance}} + C_{\text{downtime}} + C_{\text{decommissioning}}$$

Understanding this equation is vital for procurement; a "budget" robot with high energy consumption or frequent spare part requirements can quickly become more expensive than a premium, high-reliability alternative.

Table 10.9. TCO components and illustrative planning ranges for an integrated robotic system

Cost Category	Components	Illustrative share
Acquisition	Robot, controller, EoAT, safety devices, PLC, sensors, conveyors, fencing, software	20–35%
Installation & commissioning	Integration, programming, electrical work, safety validation, FAT/SAT, training	5–15%

Cost Category	Components	Illustrative share
Operation	Energy, consumables (grease, air, tooling), operator labour	15–25%
Maintenance	Spare parts, labour, service contracts, calibration, software updates	15–25%
Downtime cost	Lost production, quality losses, expediting, penalty clauses	10–20%
Decommissioning	Removal, disposal, site remediation, asset write-off	2–5%

Strategic Insight

The ranges in Table 10.9 are illustrative planning assumptions, not industry benchmarks; the total may also include financing, floor space, consumables, cybersecurity, training and residual value. Sensitivity analysis should therefore vary downtime cost, service life, utilisation, energy price and major-repair assumptions rather than relying on a single TCO estimate.

WORKED EXAMPLE 10.3 — TCO COMPARISON: ROBOT A VS. ROBOT B

Problem: Two robots for a packaging cell, 10-year lifecycle, 4,000 h/year, downtime cost €500/h:

Robot A: €85,000 purchase. Maintenance €4,500/yr. Energy €2,800/yr. MTBF = 12,000 h. MTTR = 6 h.

Robot B: €110,000 purchase. Maintenance €2,800/yr. Energy €2,200/yr. MTBF = 25,000 h. MTTR = 3 h.

Installation = 15% of purchase price for both.

Robot A TCO:

Acquisition: €85,000 + Installation: €12,750 + Maintenance: €45,000 + Energy: €28,000

Failures: $40,000/12,000 = 3.33 \rightarrow$ Downtime cost: $3.33 \times 6 \times €500 = €9,990$

TCO_A = €180,740

Robot B TCO:

Acquisition: €110,000 + Installation: €16,500 + Maintenance: €28,000 + Energy: €22,000

Failures: $40,000/25,000 = 1.6 \rightarrow$ Downtime cost: $1.6 \times 3 \times \text{€}500 = \text{€}2,400$

TCO_B = €178,900

Result: Robot B saves approximately €1,840 over 10 years at €500/h downtime, despite the higher purchase price. At €5,000/h downtime, TCO_A $\approx$ €270,750 and TCO_B $\approx$ €200,500, so Robot B saves approximately €70,250. The result is highly sensitive to failure independence, duty, repair-time definition and downtime valuation.

10.5 Spare Parts Optimisation

A structured spare-parts strategy balances inventory cost against restoration delay. ABC value classification should be combined with VED criticality so that both capital exposure and production consequence influence the stocking policy.

10.5.1 ABC Value Classification

ABC analysis classifies items by annual consumption value. The familiar 80/15/5 proportions are rules of thumb rather than fixed thresholds; each organisation should derive cut-offs from its own inventory data.

Table 10.10. ABC Inventory Classification for Spare Parts Management

Class	Consumption Value	Management Approach
A items (high-value few)	~20% of items, ~80% of total value	Tight control, precise demand forecasting, economic order quantity, regular review
B items (moderate)	~30% of items, ~15% of total value	Moderate control, periodic review, safety stock based on lead time
C items (trivial many)	~50% of items, ~5% of total value	Simplified control, bulk ordering, generous safety stock (low carrying cost)

Strategic Insight

ABC controls capital and review effort, but it does not measure production criticality. A low-value component can still stop a line; value classification must therefore be combined with a consequence-based criticality assessment.

10.5.2 Criticality Assessment (VED)

The VED criticality code complements ABC value: Vital items stop safe or compliant production, Essential items significantly reduce capability but may have a temporary

workaround, and Desirable items cause limited inconvenience. The code is based on consequence, workaround, lead time, interchangeability and failure demand.

Strategic Insight

Combining ABC and VED creates a spare-parts policy matrix. A low-value but Vital fuse, connector or sensor may justify on-site stock, whereas an expensive but readily interchangeable part may be centrally stocked. This avoids both capital-heavy overstocking and production loss caused by missing low-cost parts.

Table 10.11. Spare-parts criticality assessment using VED classes

Criticality class	Definition	Stocking policy	Example
Vital (V)	Failure stops safe or compliant production; no practical workaround; restoration lead time exceeds tolerance	On-site stock or formally guaranteed rapid supply; validate interchangeability and preservation requirements	Robot servo motor, safety controller, critical drive or proprietary sensor
Essential (E)	Failure reduces capacity or quality; a temporary workaround or reduced-rate mode exists	Stock based on lead time, demand and tolerated reduced capability	Gripper fingers, common sensors, pneumatic valves
Desirable (D)	Failure causes limited inconvenience; item is readily available or substitutable	Order on demand or hold low-cost bulk stock	Indicator lights, cable ties, standard fasteners

WORKED EXAMPLE 10.4 — SPARE PARTS CRITICALITY ANALYSIS

Problem: Classify the following spare parts for a robotic welding cell and recommend stocking policy:

- Robot servo motor (€3,200, lead time 6 weeks, failure stops cell)
- Welding torch nozzle (€45, lead time 3 days, consumable, replaced monthly)
- Proximity sensor (€85, lead time 1 week, failure stops station)
- Safety light curtain (€1,800, lead time 4 weeks, failure stops cell)
- Conveyor belt (€350, lead time 2 weeks, degradation visible)

Solution:

Servo motor: A-V (high value, critical) → Stock 1 unit on-site. Cost of stocking (€3,200) << cost of 6 weeks downtime.

Torch nozzle: C-E (low value, important consumable) → Stock 6-month supply (6 units). Bulk order quarterly.

Proximity sensor: C-V (low value, critical) → Stock 2 units on-site. Lead time (1 week) is too long for production stop.

Light curtain: A-V (high value, critical) → Stock 1 unit on-site. Safety-critical — cell cannot operate without it.

Conveyor belt: B-E (medium value, important) → Stock 1 unit. CBM monitors belt condition; replacement is plannable.

10.6 Lifecycle Phases of a Robotic Production System

The long-term success of an automated facility depends not only on the initial technical deployment but on a structured, data-driven approach to maintaining asset integrity and managing the complexities of the system's entire operational lifespan.

Table 10.12 structures the lifecycle from concept to decommissioning. Durations are illustrative and vary with cell complexity, procurement lead time, regulatory context and upgrade strategy; the purpose of the table is to assign engineering responsibilities and evidence at each phase.

Table 10.12. *Lifecycle Phases of a Robotic Production System*

Phase	Duration	Key Activities	Engineering Focus
1. Conceptual design	2–6 months	Requirements, feasibility, concept generation, preliminary layout	Process analysis, robot pre-selection, preliminary hazard ID
2. Detailed design	3–9 months	Detailed engineering, equipment selection, simulation, control design	3D modelling, risk assessment, EoAT design, control specification
3. Procurement	2–6 months	Equipment ordering, custom fabrication, panel building, software	Vendor management, quality inspection, sub-assembly testing
4. Integration & commissioning	1–4 months	Assembly, wiring, programming, FAT, installation, SAT, training	System debugging, performance verification, safety validation, documentation

Phase	Duration	Key Activities	Engineering Focus
5. Production operation	10–20 years	Production, monitoring, maintenance, improvement, changeovers	OEE tracking, downtime analysis, process optimisation, energy management
6. Upgrade & retrofit	Variable	Technology upgrade, capacity expansion, obsolescence management	New robot integration, control migration, safety re-assessment
7. Decommissioning	1–3 months	Shutdown, removal, disposal, site remediation, knowledge transfer	Environmental compliance, data archiving, reuse assessment

Strategic Insight

The operational phase normally dominates calendar life and lifecycle expenditure, while many maintainability, obsolescence and data-quality constraints are fixed during detailed design. Lifecycle engineering therefore requires explicit feedback from failures, repair time, spares consumption and modifications into future specifications.

10.6.1 Design for Maintainability Checklist

Decisions during design (Phases 1–2) have the greatest impact on lifecycle costs:

- Access and egress: provide task-specific working space, escape routes and component-removal envelopes verified by ergonomic and safety assessment
- Modular assemblies: use quick-disconnect fittings, plug-in modules, and standardised interfaces for rapid replacement
- Cable management: systematic labelling (both ends), cable trays with 20% spare capacity, colour-coded by function
- Handling provisions: provide certified lifting points or mechanical-assistance access wherever manual handling is not acceptable under the site risk assessment
- Diagnostic access: test points, status LEDs, and diagnostic ports accessible without removing covers
- Standardisation: minimise variety of fasteners, connectors, and component types across the system
- Expansion allowance: document justified spare I/O, addressing, network bandwidth, cabinet space, power and cooling capacity rather than applying a universal percentage

- Documentation integration: QR codes on major components linking to maintenance procedures and spare parts lists
-

10.6.2 Commissioning: FAT and SAT

Commissioning is the critical transition from design to operation. A structured process includes:

- FAT (Factory Acceptance Test) — Verification at the integrator’s facility. Tests all functions against specification. Customer witnesses.
- SAT (Site Acceptance Test) — Verification after installation at customer site with actual utilities, materials, and conditions.
- Safety validation — Functional testing of all safety devices and functions, documented per ISO 10218-2.
- Performance qualification — Demonstration that cycle time, throughput, quality, and availability meet agreed targets.
- Training — Structured programmes for operators, maintenance technicians, and supervisors.
- Documentation handover — Complete technical package: schematics, programs, risk assessment, maintenance schedules, spare parts lists.

10.7 Digital Twins for Maintenance and Lifecycle Management

A maintenance digital twin combines an identified asset, governed data connections and one or more validated models used for a defined decision. Many practical deployments are digital shadows with one-way data flow; bidirectional parameter changes require additional safety, cybersecurity and change-control evidence.

Table 10.13 summarises maintenance applications that can be supported by a governed digital model or digital shadow. The required model fidelity and data flow should be selected from the decision to be supported, not from a generic ambition to reproduce the complete asset.

Table 10.13. *Digital Twins for Maintenance and Lifecycle Management*

Application	Description	Benefit
Predictive analytics	Analyses sensor patterns against degradation models for RUL prediction	Optimal maintenance timing — not too early, not too late

Application	Description	Benefit
Virtual diagnostics	Simulates potential root causes when faults occur	Diagnostic time from hours to minutes; remote troubleshooting
What-if simulation	Tests parameter changes on digital model before physical system	Risk-free experimentation; validates procedures without disruption
Spare parts optimisation	Predicts degradation across fleet for JIT procurement	Reduced inventory cost while maintaining availability
Training simulation	Technicians practice fault diagnosis on digital twin	Faster training; reduced errors on physical system
Lifecycle planning	Long-term simulation of aging under projected scenarios	Informed CAPEX planning; optimised asset utilisation

Strategic Insight

Table 10.13 shows that value comes from supporting a maintenance decision, not from visual fidelity alone. Virtual diagnostics and what-if analysis are useful only when the model is calibrated, configuration is traceable and the user understands uncertainty and model limitations.

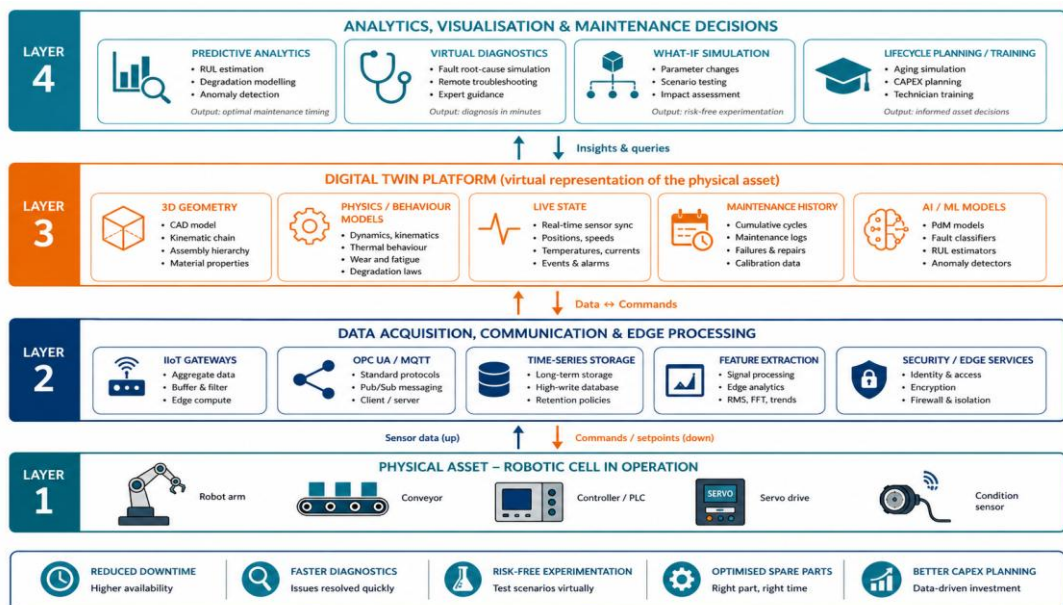


Figure 10.5. Digital twin architecture: Physical system → Data acquisition → Digital twin platform → Analytics & visualisation → Maintenance decisions.

Implementation challenges include sensor calibration, time alignment, operating-context data, model validation, configuration management, legacy integration, cybersecurity, organisational ownership and cost justification. Start with one high-value decision and expand only after its data and model have been trusted (Figure 10.5).

Common Pitfalls in Maintenance and Lifecycle Engineering

Pitfall 1: Treating maintenance as a cost to minimise.

Cutting budgets increases unplanned downtime, which is always more expensive than planned maintenance. Maintenance is an investment in availability.

Pitfall 2: Applying the same strategy to all components.

PM for everything wastes resources on run-to-failure candidates and under-protects components needing CBM/PdM.

Pitfall 3: Ignoring manufacturer schedules.

Robot manufacturers specify greasing intervals, battery replacement, and calibration for good reason. Deviating without analysis voids warranties and reduces reliability.

Pitfall 4: Not stocking critical spare parts.

A low-cost sensor can create a large production loss when no substitute exists and replenishment exceeds the tolerated restoration time. Combined ABC–VED analysis is therefore essential.

Pitfall 5: Neglecting software maintenance.

Firmware updates include bug fixes, performance improvements, and security patches. Software is not static after commissioning.

Pitfall 6: Focusing solely on MTBF, ignoring MTTR.

Reducing repair time (better diagnostics, training, spare parts) is often faster and cheaper than improving component reliability.

Pitfall 7: No Weibull analysis of failure data.

Without understanding β , you cannot determine if preventive replacement is beneficial ($\beta > 1$) or counterproductive ($\beta < 1$).

Pitfall 8: Excessive or unjustified preventive maintenance.

Calendar tasks without an effective failure-mode link can waste resources and introduce maintenance-induced faults. Every task needs a technical basis, interval rationale and feedback from results.

Pitfall 9: No decommissioning plan.

Rushed end-of-life decisions cause environmental non-compliance, loss of institutional knowledge, and missed reuse opportunities.

Pitfall 10: Disconnecting design from operational feedback.

Maintenance data should feed back into new system design. Without this loop, the same design mistakes repeat.

KEY TAKEAWAYS

Reliability is quantified through MTBF, MTTR, and Availability = $MTBF / (MTBF + MTTR)$. System MTBF in series is always lower than the weakest component.

The bathtub curve has three phases. The Weibull β parameter determines whether preventive replacement is beneficial ($\beta > 1$) or counterproductive ($\beta < 1$).

Four maintenance strategies exist: reactive, preventive, condition-based, predictive. A practical programme uses a mix, assigned per component via RCM logic.

Reliability-Centred Maintenance selects technically effective and consequence-justified tasks. The goal is the right task for the right failure mode, not maximum maintenance.

Vibration analysis is the primary CBM technique for rotating machinery. Different faults produce characteristic frequency signatures.

TCO captures acquisition, integration, operation, maintenance, downtime, upgrade and end-of-life assumptions. Sensitivity analysis is more reliable than universal percentage shares.

Spare parts combine ABC value with VED criticality. Low-value Vital items may require on-site stock, while high-value items may use pooled or guaranteed supply.

The lifecycle spans 7 phases. Design decisions have the greatest impact on lifecycle costs but receive the least attention in many projects.

Digital twins extend from design into operation: predictive maintenance, virtual diagnostics, what-if analysis, lifecycle planning.

Commissioning (FAT/SAT) is a critical quality gate. Rushing it is the most common cause of early-life reliability problems.

Review Questions

1. A robotic cell has components in series: Robot (MTBF 35,000 h), PLC (MTBF 150,000 h), Conveyor (MTBF 12,000 h), Gripper (MTBF 6,000 h), Sensors (4 units, each 40,000 h). Calculate system MTBF and expected failures per year (4,000 h/yr). Identify improvement priorities.
2. Explain the three phases of the bathtub curve. How does the Weibull β parameter relate to each phase? Why is preventive replacement counterproductive when $\beta < 1$?
3. Compare reactive, preventive, condition-based, and predictive maintenance. For a robotic welding cell, assign a strategy to: robot servo motor, gripper fingers, safety E-stop button, conveyor belt, indicator light. Justify each choice.
4. Explain TCO. Robot X costs €70,000 (MTBF 8,000 h, MTTR 5 h, maintenance €5,000/yr, energy €3,000/yr) vs. Robot Y at €95,000 (MTBF 20,000 h, MTTR 2 h, maintenance €3,000/yr, energy €2,500/yr). 4,000 h/yr, 10 years, €400/h downtime, 12% installation. Calculate TCO and recommend.
5. Describe the seven RCM questions. How do failure consequence categories drive maintenance strategy selection?
6. List five condition monitoring techniques for robotic systems. For vibration analysis, name three fault types and their characteristic frequency signatures.
7. Explain the combined ABC–VED spare-parts classification. Classify: robot servo motor (€3,200, 6-week lead), proximity sensor (€85, 1-week lead, production-critical), cable tie (€0.50, available locally).
8. Describe the seven lifecycle phases. Why do design decisions have the greatest impact on lifecycle costs? List four Design for Maintainability principles.
9. Explain three digital twin applications in maintenance. A conveyor motor shows increasing vibration at BPFO frequency. Describe the CBM response: what is detected, what does it mean, and what action should be taken?

References

- [1] Groover, M.P., Automation, Production Systems, and Computer-Integrated Manufacturing, 5th ed., Pearson, 2020, Chapter 27.
- [2] Moubray, J., Reliability-Centred Maintenance, 2nd ed., Industrial Press, 1997.

- [3] Jardine, A.K.S. and Tsang, A.H.C., *Maintenance, Replacement, and Reliability*, 2nd ed., CRC Press, 2013.
- [4] Abernethy, R.B., *The New Weibull Handbook*, 5th ed., self-published, 2006.
- [5] ISO 20816-1:2016 — Mechanical vibration — Measurement and evaluation of machine vibration — Part 1: General guidelines.
- [6] EN 13306:2017 — Maintenance — Maintenance terminology.
- [7] IEC 60300-3-11:2009 — Dependability management — Part 3-11: Application guide — Reliability centred maintenance.
- [8] IEC 60300-3-10:2025 — Dependability management — Part 3-10: Application guide — Maintainability and maintenance.
- [9] ABB Robotics, *Connected Services and Condition-Based Maintenance*, official service documentation, accessed 2026.
- [10] FANUC, *Zero Down Time (ZDT) — connected monitoring and predictive-maintenance* documentation, accessed 2026.
- [11] Tao, F. et al., 'Digital Twin in Industry: State-of-the-Art', *IEEE Trans. Industrial Informatics*, 15(4), 2019, pp. 2405–2415.
- [12] ISO 13849-1:2023 — Safety of machinery — Safety-related parts of control systems — Part 1: General principles for design.
- [13] European Commission, *Industry 5.0 — Towards a sustainable, human-centric and resilient European industry*, 2021.
- [14] Mocan, B., *Design and Operation of Robotic Production Systems — Course Notes*, 2025–2026.

CHAPTER

11

Collaborative Robots and AI-Enhanced Production

From caged power to shared workspaces and learning systems

For most of the history of industrial robotics, high-speed manipulators operated inside safeguarded spaces. Operators loaded fixtures through interlocked access points, initiated automatic cycles, and remained outside the hazardous zone while the robot moved. This architecture remains appropriate whenever productivity, payload, tooling hazards, or stopping performance make close human proximity unnecessary.

A commercially important shift occurred in December 2008, when the first UR5 was installed at Linatex in Denmark for CNC machine tending. The installation demonstrated that a lightweight robot with integrated safety functions and accessible programming could be deployed beside personnel after an application-specific risk assessment. The decisive change was not the removal of a fence; it was the engineering of a complete collaborative application in which robot, end-effector, workpiece, layout, control system, and human task were assessed together [18].

In parallel, advances in perception and learning expanded what robotic systems could infer from production data. Deep-learning models improved defect recognition and object localisation; reinforcement learning enabled policies for difficult manipulation problems; and language-based assistants began to support programme generation and task planning. These capabilities increase flexibility, but they do not replace deterministic safety functions. This chapter therefore treats collaboration and artificial intelligence as complementary engineering layers that must be integrated, verified, and governed independently.

LEARNING OBJECTIVES

1. Distinguish collaborative robots from conventional industrial robots in terms of mechanical design, safety-related functions, control philosophy, and application constraints.
2. Apply the four collaborative operation methods described in ISO 10218-2:2025 and ISO/TS 15066:2016, and select an appropriate method or combination for a given task.
3. Design an ergonomic human-robot workstation using ISO/TS 15066 biomechanical guidance, ISO/PAS 5672 contact-measurement methods, and recognised manual-handling principles.
4. Develop a human-robot task-allocation strategy that exploits complementary strengths while controlling handover, workload, and exception handling.

5. Identify the principal AI techniques used in industrial robotics — classical vision, deep learning, reinforcement learning, and language-based assistance — and match them to suitable application classes.
6. Select a 2D or 3D vision architecture and a classical or learned processing method for an inspection or guidance application, with technical and economic justification.
7. Evaluate the readiness and economic viability of an AI-enhanced robotic solution and justify a build, buy, pilot, or defer decision.
8. Explain the verification, cybersecurity, human-oversight, regulatory, and liability challenges introduced when AI is integrated into a safety-relevant robotic application.

11.1 Cobots: Design for Coexistence

A collaborative robot is not an inherently safe machine and it is not exempt from the risk assessment required for industrial robot applications. The term describes a robot designed to facilitate collaborative operation, but safety is established only at application level under ISO 10218-1:2025, ISO 10218-2:2025, and the complementary guidance of ISO/TS 15066:2016 [3–5]. Four recurring design characteristics support this objective:

- **Reduced moving mass and contact-tolerant geometry.** Many cobots use lighter links, rounded covers, and limited exposed pinch features. This can reduce collision energy, although product ranges overlap and some high-payload cobots are heavier than small conventional industrial robots.
- **Contact detection and external-force estimation.** Depending on the manufacturer, contact may be detected by joint torque sensors, motor-current estimation, a wrist force-torque sensor, or a combination. Joint torque sensing at every axis is common in some families but is not universal.
- **Controlled mechanical behaviour.** Transmission ratios, brakes, friction compensation, and gravity compensation are selected to support manual guidance, controlled stopping, and force-sensitive motion. Back-drivability varies substantially and should not be assumed from the label ‘cobot’.
- **Safety-related monitoring functions.** Certified functions can limit joint position, speed, power, momentum, stopping time, and stopping distance, or monitor tool orientation and safe zones. Performance levels are manufacturer- and function-

specific; the integrator must validate the complete safety chain against ISO 13849-1 or IEC 62061 as applicable [8].

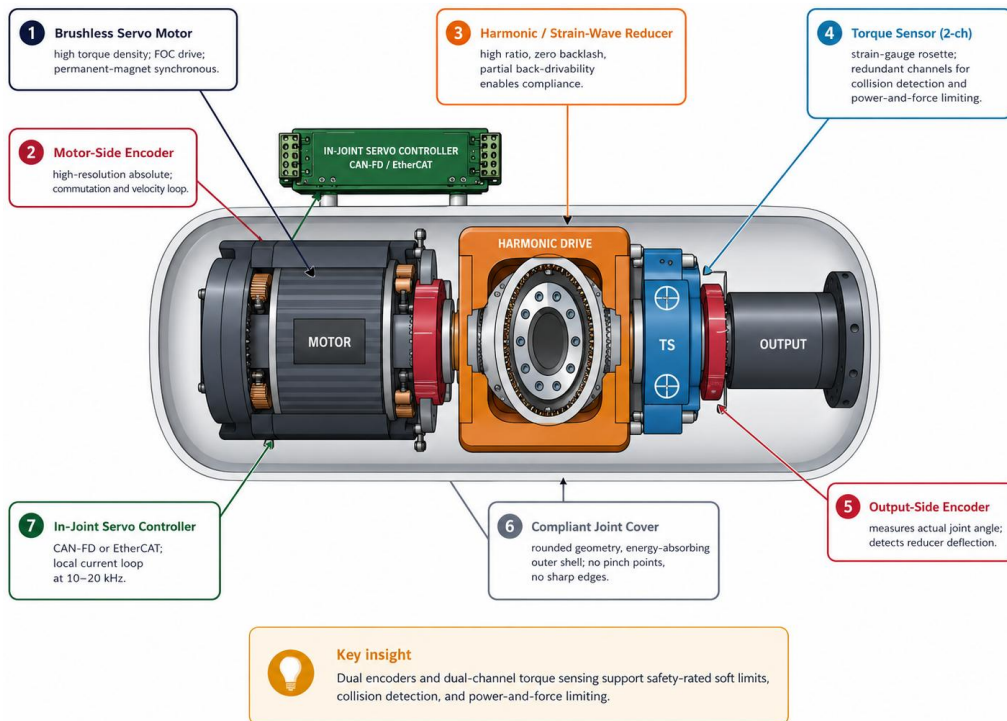


Figure 11.1. Conceptual safety-related architecture of a collaborative robot joint.

11.1.1 The Cobot Market Landscape

The International Federation of Robotics reported almost 55,000 collaborative robots installed worldwide in 2022, representing 9.9% of industrial robot installations. In 2023, their share increased to 10.5% of 541,000 installations — approximately 57,000 units — while total industrial robot installations reached 542,000 in 2024 [1,2]. Cobots therefore remain a minority of new installations, but they are prominent in high-mix assembly, machine tending, inspection, laboratory automation, and applications where rapid changeover is valuable.

Table 11.1 provides a representative portfolio snapshot for July 2026. The market now spans compact arms for electronics assembly, high-reach systems for machine tending, and heavy-payload variants for palletising. Competitive differentiation increasingly concerns ecosystem maturity, force-control capability, integrated vision, service support, and compatibility with industrial automation platforms rather than payload alone.

Higher nominal payload does not imply that the robot may move that payload at full speed beside a person. Collaborative operation is an application property: the tool, workpiece, trajectory, accessible contact regions, trapping hazards, stopping performance, and sensing coverage must all be validated. Payload ranges in the table are manufacturer nominal maxima and should be confirmed against the current technical data for the selected variant [17].

Table 11.1. Representative collaborative robot portfolios (July 2026 snapshot)

Supplier	Representative families	Nominal payload range	Selected engineering characteristics
Universal Robots	e-Series; UR Series	3–35 kg	Broad application ecosystem; safety-related limits and extensive third-party integration
FANUC	CRX; CR	3–50 kg	CRX simplified teaching and industrial service support; portfolio extends to heavy payloads
ABB	YuMi; GoFa; SWIFTI	0.5–14 kg	Torque-sensed GoFa variants; RobotStudio integration; high-speed SSM-oriented SWIFTI applications
KUKA	LBR iisy; LBR iiwa	3–15 kg	Integrated joint torque sensing; force and impedance control for assembly
Doosan Robotics	A; M; H; P series	5–30 kg	Six-axis torque sensing on selected series; long-reach and palletising variants
Techman Robot	TM AI Cobot S series	5–30 kg	Integrated vision and AI-oriented software; heavy-payload S-series variants
OMRON	TM; TM S series	5–30 kg nominal	Integrated vision and close coupling with OMRON control, safety, and mobile-robot platforms

INDUSTRIAL PRACTICE NOTE

IN PRACTICE: PLAN FOR A PRODUCTION CELL, NOT A DEMONSTRATION

A basic pick-and-place demonstration can be taught in hours, but a production-ready collaborative workstation normally requires several weeks of engineering. An illustrative programme is: risk assessment and concept design; detailed layout and tooling; installation and controls integration; programming and process development; validation of safety functions, stopping performance, and PFL contact limits where applicable; operator training; and a monitored production ramp-up. Activities can overlap, but safety validation and process capability cannot be compressed into a vendor demonstration. The schedule must include change control for every later modification to tooling, payload, speed, or layout.

Note: this planning sequence is illustrative. Actual duration depends on application complexity, component lead times, validation scope, and local conformity-assessment requirements.

11.2 Applying ISO 10218 and ISO/TS 15066 in Practice

Chapter 8 introduced the four established collaborative operation methods. ISO 10218-2:2025 now provides the principal application-level framework, while ISO/TS 15066:2016 remains current and supplies additional guidance for collaborative applications, particularly power and force limiting. The methods can be combined within one cycle, provided that every transition is implemented by safety-related control functions and validated [4,5].

11.2.1 Mode Selection Logic

The following decision sequence is a practical starting point; it does not replace the formal risk assessment:

- 1. If personnel do not need to be present during robot motion, use conventional safeguarding or a safety-rated monitored stop (SRMS) for intermittent access.
- 2. If an operator must physically guide the robot, use hand guiding only after an SRMS, with an enabling device and monitored speed as required by the application.
- 3. If human and robot must move simultaneously without intentional contact, use speed and separation monitoring (SSM). The protective separation distance

must include sensor response, control reaction, robot stopping distance, human approach speed, and measurement uncertainty.

- 4. Use power and force limiting (PFL) only where foreseeable transient and quasi-static contacts can remain below the body-region-specific force and pressure limits, and where sharp, hot, trapping, or cutting hazards have been eliminated or separately safeguarded.

WORKED EXAMPLE 11.1 — MODE SELECTION FOR A SCREWDRIVING CELL

An electronics assembler uses a cobot to drive M2 screws while an operator loads the fixture, removes the completed housing, and performs a visual check. Takt time is 35 s. The screwdriver body is rounded, but the rotating bit and the fixture create puncture, entanglement, and trapping hazards.

Analysis. The operator and robot need simultaneous access during transfer, so SRMS throughout the cycle would reduce productivity. PFL alone is insufficient during active screwdriving because the bit and fixture hazards are not represented by blunt-impact limits.

Decision. Use PFL or SSM for transfer and approach, then place the robot in an SRMS or protect the driving micro-zone with an interlocked guard while the bit rotates. A reduced-speed final approach is useful for process control, but it counts as a safety measure only when implemented and monitored by a safety-rated function.

Verification. Validate the stop and restart sequence, separation distance, tool guard, fixture pinch points, and every foreseeable contact scenario. Where PFL is used, measure both force and pressure with an ISO/PAS 5672-compliant device at the worst-case robot configurations and body regions. A result that is acceptable for the palm cannot be generalised to the fingers, knuckles, or wrist [5,6].

11.2.2 Designing the Collaborative Workstation

A collaborative workstation must integrate risk reduction, process capability, and ergonomics. Practical design principles include:

- **Anthropometric fit.** Select bench height, reach distances, and presentation angles from the user population and task demands. An adjustable working height

of approximately 750–950 mm is a useful starting range, not a universal requirement.

- **Visibility and predictability.** Maintain a clear line of sight to the tool and shared workspace, avoid unexpected approaches from behind, and use smooth trajectories with understandable motion cues.
- **Material presentation.** Place high-frequency components in the operator's primary reach zone and separate incoming, work-in-progress, and completed parts to reduce cross-reach and handover ambiguity.
- **Unambiguous status indication.** Use colour, text, and where necessary sound in a manner consistent with IEC 60073 and IEC 60204-1. Do not use colour as the sole safety indication or assign application-specific meanings that contradict established conventions.
- **Egress and anti-trapping.** Maintain an unobstructed escape path and prevent the robot, workpiece, or fixed equipment from trapping the operator. Sensor occlusion and reset locations must also be evaluated.

11.2.3 Human-Robot Task Allocation

Effective collaboration is based on complementary task allocation rather than simple substitution. Robots are advantageous when a task is repetitive, measurable, force-controlled, or hazardous; humans retain advantages in dexterous handling of variable materials, contextual judgement, and recovery from novel situations. The allocation must also consider cycle synchronisation, cognitive workload, training, and the consequences of an incorrect handover.

Table 11.2 summarises these differences as design tendencies rather than absolutes. Capability depends on the specific robot, sensor suite, tooling, operator population, and production environment.

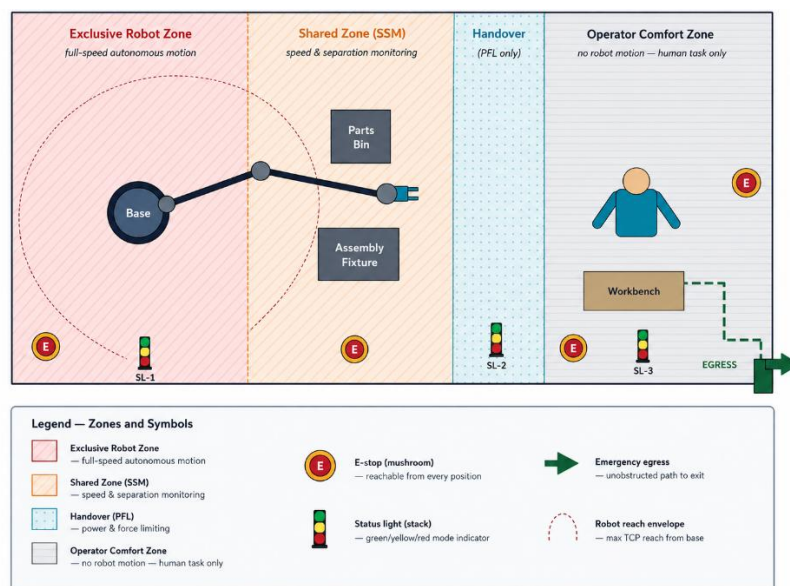
Table 11.2. Human and robot capabilities relevant to task allocation

Capability	Human advantage	Robot advantage
Dexterity	Adaptive manipulation; deformable and variable materials	Repeatable trajectories; controlled force application
Perception	Context-rich scene interpretation; anomaly recognition	Calibrated measurement; high-rate, consistent inspection

Capability	Human advantage	Robot advantage
Adaptability	Handles unforeseen conditions and ambiguous exceptions	Rapid repeat of validated variants within programmed limits
Endurance	Strategic problem-solving and flexible re-planning	Sustained repetitive work without physiological fatigue
Decision-making	Contextual and ethical judgement; responsibility	Consistent rule execution and data-driven optimisation
Strength	Fine force modulation with rich tactile feedback	Sustained payload and force without musculoskeletal strain

The preferred allocation assigns repetitive, heavy, precision-critical, or exposure-related subtasks to the robot while reserving deformable-material handling, contextual quality judgement, and exception management for the human. The handover point must specify position, timing, confirmation logic, safe state, and recovery from a missed or delayed transfer.

Figure 11.2 shows a top-view collaborative assembly concept with three functional zones: a robot-only region, a shared task region, and an operator region. The maximum robot reach, operator reach, material flow, and escape route are considered simultaneously rather than added after the equipment has been placed.



Colour bands are explanatory, not safety devices. Actual SSM boundaries follow the validated protective-separation-distance calculation, measured stopping performance, and safety-sensor coverage. The robot may run faster in a verified robot-only region, decelerate as separation decreases, and enter an SRMS before the minimum permitted distance is breached.

WORKED EXAMPLE 11.2 — TASK ALLOCATION FOR MOTOR ASSEMBLY

Problem: Allocate seven steps in small-motor assembly: shaft insertion, guarded bearing-press loading/unloading, winding placement, connector soldering, functional test, labelling, and packaging. Available: a UR10e-class cobot and a guarded press.

Robot: shaft insertion; press loading/unloading (the guarded press generates the force); soldering with fume extraction and process monitoring; label placement.

Human: winding placement because the material is flexible; interpretation of functional-test exceptions and ambiguous results.

Shared: packaging, with the robot presenting the motor in a defined pose and the operator closing and stacking the pack.

Cycle-time estimate: robot-only work = 32 s and human-only work = 28 s in parallel; shared packaging = 6 s sequentially. Effective cycle time = $\max(32, 28) + 6 = 38$ s. The human may use the 4 s balance to prepare the next winding only if the handover remains unaffected.

Key insight: Model the precedence network. Ignoring shared sequential work produces an optimistic cycle time.

11.3 Artificial Intelligence in Robotic Production

The term 'AI' covers techniques with very different levels of determinism, data dependence, deployment maturity, and verification burden. The following four-layer model is a useful engineering classification, not a safety hierarchy. In production systems the layers are commonly combined, while safety-related functions remain independent.

11.3.1 Layer 1 — Classical Machine Vision

Rule-based machine vision remains a core industrial technology. Blob analysis, template matching, edge detection, geometric fitting, and calibrated measurement execute according to explicit parameters and are highly traceable. Processing time ranges from milliseconds to longer depending on image size, algorithm, and hardware. Their main

limitation is sensitivity to uncontrolled lighting, presentation, surface variation, and cases not represented by the programmed rules.

11.3.2 Layer 2 — Deep Learning for Perception

Convolutional neural networks and vision transformers complement classical vision when relevant variation is difficult to encode explicitly and representative training data are available. Three tasks dominate industrial use:

- **Classification** — assigning one or more labels to an image, such as accepted/rejected or part family. Representative architectures include residual networks, EfficientNet-type models, and compact vision transformers.
- **Object detection** — localising and classifying instances using bounding boxes. YOLO-family detectors, Faster R-CNN, and transformer-based detectors are used for part counting, random picking, and packaging verification.
- **Segmentation** — assigning a class to each pixel. U-Net, Mask R-CNN, and foundation-model-assisted workflows are used for defect localisation, weld-seam analysis, and bead or coating inspection.

Deep learning is valuable for scratches, contamination, variable textures, and complex visual context, but a single accuracy value is not sufficient evidence. Production validation should separate training, validation, and test data by lot, shift, supplier, and time; report class-specific precision and recall; assess calibration and out-of-distribution behaviour; and define an abstention or fallback path for low-confidence cases [11].

11.3.3 Vision System Architecture and Selection

Vision selection begins with the dimensional information required by the task, then considers field of view, material, lighting, motion, accuracy, cycle time, calibration stability, and integration effort (table 11.3).

Table 11.3. *Comparative selection criteria for 2D and 3D machine vision*

Aspect	2D vision	3D vision
Data	Intensity or colour in image coordinates (x, y)	Depth map or point cloud (x, y, z)
Best suited to	OCR, colour inspection, planar measurement, pattern matching	Random bin picking, pose estimation, volume and shape inspection

Aspect	2D vision	3D vision
Metrology	Potentially high in-plane accuracy with calibrated optics; strongly dependent on field of view and lens	Depth uncertainty depends on range, surface properties, occlusion, and sensing principle
Technologies	Area-scan or line-scan cameras; telecentric optics; controlled illumination	Stereo, structured light, time-of-flight, and laser triangulation
Throughput	Usually lower acquisition and processing burden	Usually higher acquisition, filtering, and point-cloud processing burden
Cost	Typically lower hardware and integration cost	Typically higher hardware, calibration, and computation cost
Integration	Well established; simpler calibration for planar tasks	More demanding calibration, registration, occlusion handling, and robot-frame transformation

Two mounting strategies dominate. In eye-to-hand configurations the camera is fixed relative to the cell, which provides a stable, wide field of view and straightforward robot-world calibration. In eye-in-hand configurations the camera is mounted on the robot, providing close and variable viewpoints but requiring hand-eye calibration and management of motion blur, cable routing, and changing illumination. Bin picking may use either strategy; selection depends on occlusion, required viewpoint diversity, and cycle time.

WORKED EXAMPLE 11.3 — VISION SYSTEM SELECTION FOR A SORTING CELL

Problem: A sorting cell must identify six bottle variants at 40 parts/minute. Bottles are upright on a conveyor, with three colours and two cap types. The robot needs planar position and orientation for pick-and-place.

Analysis: Variant identification requires colour and cap classification; guidance requires x-y position and in-plane angle; the available processing window is 1.5 s; and acquisition must be synchronised with conveyor position.

Solution: Use a 2D area-scan colour camera, controlled diffuse or ring illumination, encoder-triggered acquisition, and conveyor tracking. Colour can be classified in a calibrated colour space; cap type can use template matching or a compact classifier; centroid and principal-axis estimation provide pose.

Verification: Confirm pose error over the full field of view, conveyor-speed variation, motion blur, reflective labels, partly occluded caps, and all six variants. A nominal 15 ms image-processing result is acceptable only after end-to-end timing includes triggering, communication, robot tracking, and rejection logic.

Why not 3D? The bottles are upright and their height is determined by the identified variant. Depth is not required for the pick pose, so 3D would add calibration and processing burden without a corresponding information benefit.

Budgetary selection: An industrial smart-camera system with optics, lighting, encoder interface, mounting, and commissioning may be budgeted initially in the mid-thousands of euros. The final value should be based on supplier quotations and must include integration and validation rather than hardware alone.

11.3.4 Layer 3 — Reinforcement Learning and Learned Control

Reinforcement learning optimises a policy through interaction with an environment and a reward function. Research systems have demonstrated diverse-object grasping, manipulation of deformable items, and in-hand control, but robustness outside the training distribution remains a major limitation. Reward design, exploration, and hidden simulation assumptions can produce unsafe or unexpected behaviour; learned policies should therefore operate within deterministic constraints.

Sim-to-real transfer reduces the cost and risk of collecting experience on physical equipment:

- **Domain randomisation** — varying friction, mass, sensor noise, latency, lighting, textures, and object pose so that the policy is less dependent on a single simulated condition.
- **Progressive transfer** — pre-training in simulation, then adapting with a limited and controlled set of real observations under explicit change control.
- **Digital-twin-based learning** — using a calibrated system model as the training environment and updating model parameters from measured production data.

Common platforms include NVIDIA Isaac Sim, MuJoCo, Gazebo with ROS 2, and CoppeliaSim. Production deployment remains selective and is most credible when the learned policy is bounded by workspace, speed, force, collision, and process constraints that are enforced independently of the learning algorithm [12].

Figure 11.3 separates the pipeline into model calibration, policy training, domain randomisation, controlled transfer, and real-cell validation. Each stage produces evidence: model-error bounds, training coverage, stress-test results, approved command interfaces, and post-transfer performance.

The feedback loop from the physical cell must be governed. Real data can reveal the reality gap and support re-training, but a changed policy is a changed software component: it requires version control, regression testing, risk review, and renewed validation before production release.

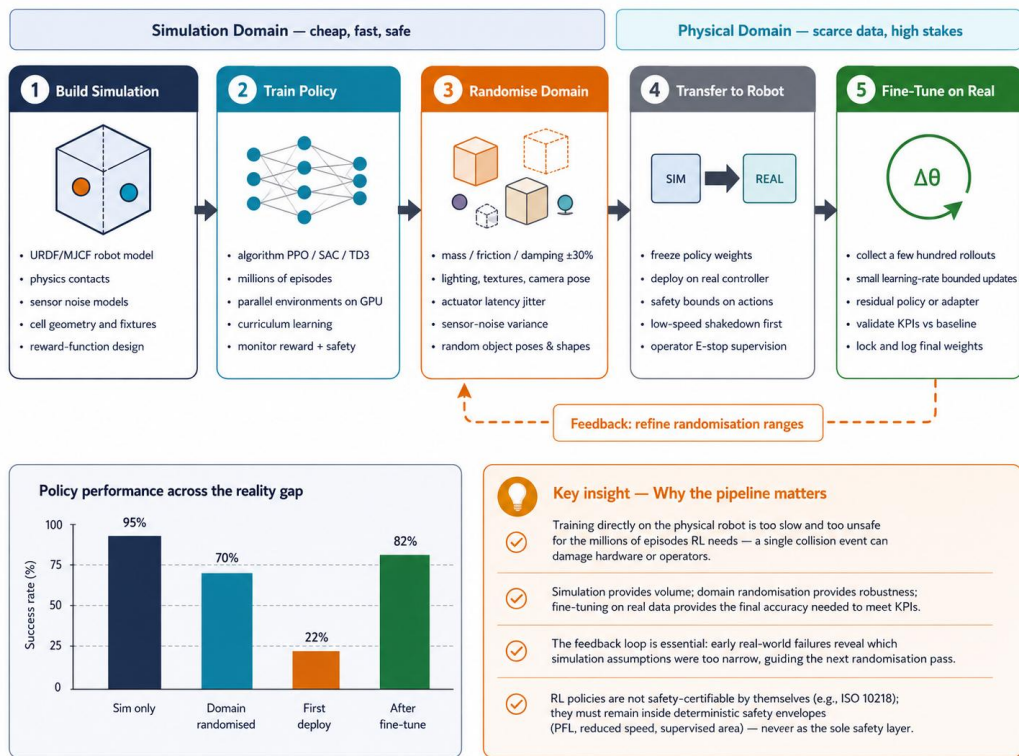


Figure 11.3. Controlled sim-to-real reinforcement-learning pipeline with an independent safety layer.

11.3.5 Layer 4 — LLM-Based Robot Programming

Large language models and vision-language-action models can translate task descriptions into code fragments, robot-program templates, test cases, or planning proposals. Research frameworks such as PaLM-E illustrate the integration of language and perception, while industrial tools increasingly use language models as programming assistants [13]. Their output is probabilistic and must be treated as untrusted until it has passed syntax checking, semantic constraints, simulation or sandbox execution, code review, and explicit human approval. Direct autonomous execution of safety-critical motions is not an acceptable default architecture.

Figure 11.4 arranges four AI layers according to typical production role and verification emphasis. Classical vision is highly traceable in structured tasks; deep-learning perception is deployed with statistical validation and fallbacks; reinforcement learning is used selectively under deterministic limits; and language-based systems are primarily assistive or bounded.

The layers should not be interpreted as a replacement sequence. A robust cell may use classical calibration, a deep-learning detector, a constrained learned grasp policy, and a language interface while retaining a conventional safety PLC and validated robot safety functions. Capability and verification burden generally increase together.

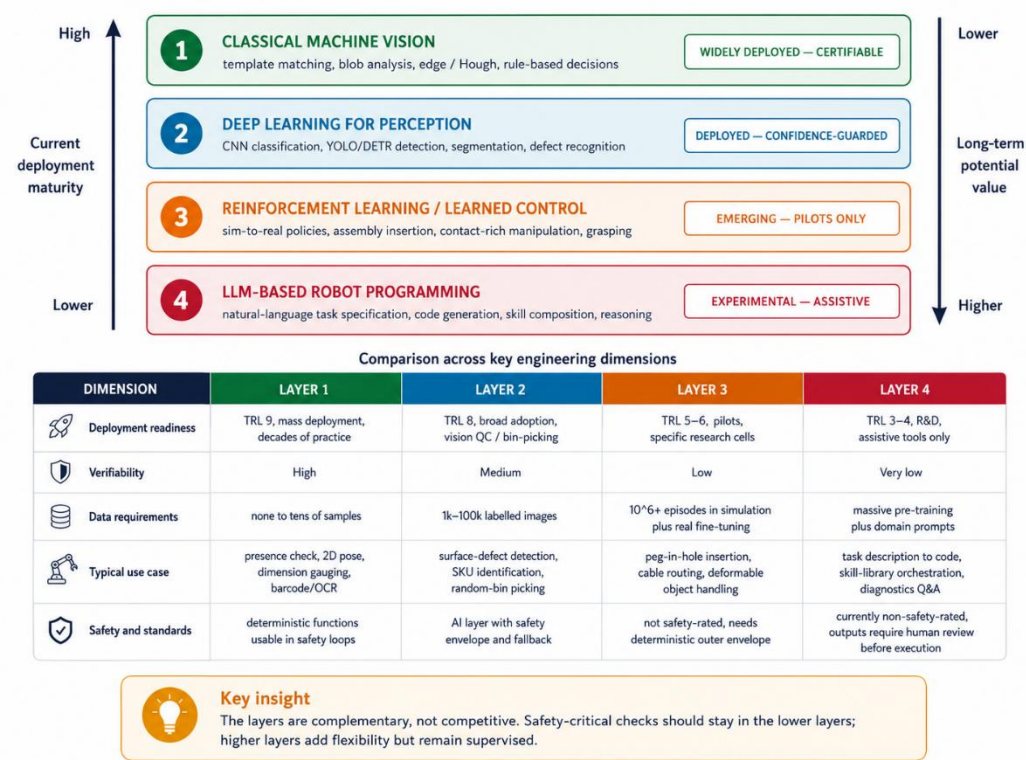


Figure 11.4. AI layers in industrial robotics: production role, verification emphasis, and deployment position.

INDUSTRIAL PRACTICE NOTE

IN PRACTICE: MONITOR AI MODEL DRIFT THROUGH A CONTROL PLAN

A deployed vision model should be monitored at four levels: (1) input characteristics such as illumination, focus, colour, and feature distributions; (2) model confidence, calibration, and abstention rate; (3) performance on a controlled 'golden' sample set; and (4) process outcomes such as false rejects, escapes, rework, and operator overrides. Alert and action limits must be established from the validated baseline using appropriate statistical methods; universal thresholds such as a fixed percentage change or a fixed re-training interval are not defensible. Re-training is triggered by evidence, and every released model passes documented change control and regression validation.

Note: the monitoring architecture is generic. Metrics, sampling frequency, control limits, and re-validation scope must be defined for the application and its risk classification.

11.4 Verification, Validation, and Liability of AI Components

Deterministic control software is normally verified against explicit requirements, defined states, timing constraints, and fault responses; formal analysis may be possible for selected functions. Machine-learning components add a different evidence problem because behaviour is inferred from data and cannot be represented completely by a short set of explicit rules. Three consequences follow:

- **Verification becomes statistical and scenario-based.** Evidence includes independent test sets, class-specific confusion matrices, calibration curves, robustness tests, data-lineage records, scenario coverage, and performance under distribution shift. These methods complement rather than eliminate requirements-based testing.
- **AI governance standards are published, but they do not confer machinery safety integrity.** ISO/IEC 22989, 23053, 23894, and 42001 provide terminology, system frameworks, risk-management guidance, and management-system requirements. A learned component used in a robot cell must still be integrated under machinery-safety standards, and safety-related risk reduction should remain in validated functions unless an accepted conformity route exists [3,4,8,14].
- **Regulatory responsibility follows the intended function and system integration.** Under the EU AI Act, an AI system that is itself a safety component, or performs

a safety function in machinery subject to third-party conformity assessment, may be classified as high-risk. The implementation timetable was still evolving in July 2026, and the EU Machinery Regulation applies from 20 January 2027. Integrators must document provider/deployer roles, substantial modifications, technical documentation, human oversight, post-market monitoring, and the interaction with CE conformity assessment; specialist legal review is appropriate [15,16].

11.4.1 The Human-in-the-Loop Pattern

Human oversight is useful when the AI identifies or recommends and a trained operator retains authority to accept, reject, or escalate. However, ‘human in the loop’ is not automatically a safety function. The operator must have adequate information, time, competence, workload capacity, and a safe means to intervene. If the human cannot reliably detect and correct an error before harm occurs, the architecture requires an independent automatic safeguard. Overrides and reviewed cases should be logged because they also provide labelled evidence for model improvement.

WORKED EXAMPLE 11.4 — ROI OF AN AI-ENHANCED VISION INSPECTION

A contract electronics manufacturer inspects 18,000 PCBs per shift for soldering defects. The current system has 94% defect-detection sensitivity; a deep-learning system is projected to achieve 99.4%. The quoted capital cost is €85,000.

Assumptions: two shifts per day, 36,000 boards/day; true defect prevalence = 1.0%; cost of an escaped defective board = €48; 250 operating days/year.

Defective boards/day = $36,000 \times 0.01 = 360$. Baseline escapes = $360 \times (1 - 0.94) = 21.6/\text{day}$. AI escapes = $360 \times (1 - 0.994) = 2.16/\text{day}$.

Avoided escape cost/day = $(21.6 - 2.16) \times €48 = €933.12$.

Annual gross saving = $€933.12 \times 250 = €233,280$. Annual data-curation, computing, maintenance, and re-validation cost = €16,000. Net annual benefit = €217,280.

Simple payback = $€85,000 / €217,280 = 0.39$ years, or approximately 98 operating days.

Lesson: Detection sensitivity must be applied to the defective population, not to all inspected boards. A defensible ROI also includes false-reject cost, manual review, downtime, ramp-up, data acquisition, and uncertainty in defect prevalence.

WORKED EXAMPLE 11.5 — BIOMECHANICAL LIMIT CHECK FOR AN ASSEMBLY COBOT

A cobot handles a 2 kg electronic subassembly at 250 mm/s. A 3 mm chamfer on the EoAT could contact the dorsal hand. The assessment is for one defined robot pose and contact scenario; other poses and body regions must be tested separately.

Step 1 — preliminary energy screen: assume a conservative effective mass of 11.2 kg obtained from the manufacturer's validated model for this pose. $E = \frac{1}{2}mv^2 = \frac{1}{2} \times 11.2 \times 0.25^2 = 0.35$ J. This screen does not establish compliance.

Step 2 — define the contact model: classify the contact as transient or quasi-static, select the relevant body region, and configure the measuring device with the specified spring and damping characteristics.

Step 3 — measure: an ISO/PAS 5672-compliant device records 130 N peak force and 460 N/cm² peak pressure. Pressure is measured; it is not calculated by treating the chamfer radius as the contact-patch radius.

Step 4 — compare: for the selected dorsal-hand transient pressure limit of 190 N/cm², 460 N/cm² is non-compliant even though the force result may be below the applicable force threshold.

Mitigation and re-test: increase compliance and contact area with a validated cover, reduce speed if required, and repeat the measurement in the worst-case pose. An illustrative re-test result of 87 N/cm² is compliant for this scenario only; residual trapping, sharp-edge, and workpiece hazards remain subject to the risk assessment.

Lesson: PFL compliance depends on robot dynamics, contact geometry, body region, transient versus quasi-static contact, and measurement-device characteristics. Simple area approximations are not a substitute for physical validation [5,6].

11.5 Human-Robot Interaction and Ergonomic Design

Ergonomic improvement is a frequent objective of collaborative automation. Repetitive reaching, awkward postures, high exertion, and insufficient recovery contribute to work-related musculoskeletal disorders. A cobot is beneficial only when the redesigned division

of labour reduces exposure without introducing new risks such as monitoring burden, constrained posture, increased pace, or difficult exception handling [10].

11.5.1 Ergonomic Assessment Methods

- **RULA (Rapid Upper Limb Assessment).** Scores upper-limb posture, muscle use, and force on a 1–7 action scale: 1–2 generally acceptable, 3–4 further investigation, 5–6 investigation and change soon, and 7 investigation and change immediately.
- **REBA (Rapid Entire Body Assessment).** Assesses whole-body posture and load. Scores 8–10 indicate high risk and 11 or more very high risk; intervention priority increases accordingly.
- **OCRA (Occupational Repetitive Actions).** ISO 11228-3 provides a framework for evaluating high-frequency low-load work using repetition, force, posture, recovery, and additional factors. The selected OCRA method and interpretation bands must be reported explicitly [9].

A credible ergonomic evaluation compares the original and redesigned work over representative production periods. At minimum, it should report posture scores, repetition and force exposure, perceived exertion or discomfort, cycle time, quality, operator interventions, near misses, and any task displaced to another workstation. A headline reduction in complaints is not sufficient without exposure and outcome data.

INDUSTRIAL PRACTICE NOTE

IN PRACTICE: COMPARE PROGRAMMING METHODS THROUGH A PILOT

Programming effort depends more on task complexity, sensing, tooling, and validation than on the number of waypoints. A short pilot should measure engineering time from an empty project to a validated production cycle.

- Hand guiding: rapid for simple point teaching, but still requires frame definition, payload configuration, logic, and safety validation.
- Teach pendant: precise and auditable for discrete sequences; effort rises with path complexity and variants.
- Graphical or flow-based programming: accessible and structured, but integration logic can still become complex.
- Offline programming: requires an accurate cell model and calibration, then scales well across variants and reduces production interruption.
- LLM-assisted programming: can accelerate first drafts, documentation, and test generation, but verification, simulation, code review, and commissioning remain mandatory.

Select the method using measured pilot data: engineering hours, changeover time, residual corrections on the real cell, defect or fault rate, maintainability, and competence required. Do not generalise programming-time claims from a demonstration.

Note: the comparison identifies evaluation criteria rather than universal time ranges. Results vary with robot platform, application, integrator experience, and required assurance level.

Common Pitfalls

Pitfall 1: Treating the cobot as an inherently safe device.

A robot may provide certified safety-related functions, but the application is safe only after the robot, EoAT, workpiece, fixtures, surrounding machinery, task, and foreseeable misuse have been assessed under ISO 12100 and ISO 10218-2 [4,7].

Pitfall 2: Copying a traditional-robot layout onto a cobot.

A conventional high-speed cell does not become collaborative when the fence is removed. Collaboration requires deliberate human-centred layout, suitable operation methods, validated transitions, and control of trapping and tool hazards.

Pitfall 3: Assuming AI vision will improve over time without effort.

Model performance can drift when parts, suppliers, lighting, optics, software, or process conditions change. Monitoring, labelled review samples, and controlled re-validation are continuing engineering responsibilities.

Pitfall 4: Under-provisioning integration effort for AI components.

In many AI projects, data acquisition, labelling, interfaces, cybersecurity, process integration, validation, and operational monitoring require more effort than training the model. Budget the complete lifecycle rather than the algorithm alone.

Pitfall 5: Using collaborative mode where SRMS would suffice.

PFL and SSM add measurement, validation, and sensor-coverage requirements. If intermittent access with SRMS meets the process target, it may be the simpler and more robust risk-reduction strategy.

Pitfall 6: Ignoring EoAT contact geometry in PFL assessments.

Small radii, hard surfaces, projecting workpieces, and trapped contacts can create unacceptable pressure even at modest speed. Measure force and pressure for the actual EoAT and workpiece; do not infer compliance from robot speed alone.

Pitfall 7: Over-specifying cobot payload.

Excess payload margin can increase cost and, depending on design, moving mass and stopping distance. Select payload, reach, inertia capability, stiffness, accuracy, and environmental rating together rather than maximising a single specification.

Pitfall 8: Deploying AI without a fallback strategy.

Every AI component requires a defined low-confidence and fault response: manual review, reject-to-safe bin, stop-and-alert, redundant classical inspection, or another validated fallback. The fallback must be tested under realistic failure conditions.

KEY TAKEAWAYS

Collaborative safety is an application property, not an intrinsic property of a robot. The risk assessment includes the robot, tool, workpiece, fixtures, layout, control system, and human task.

ISO 10218-2:2025 provides the main application framework; ISO/TS 15066:2016 remains current complementary guidance. SRMS, hand guiding, SSM, and PFL may be combined only through validated safety-related transitions.

Collaborative workstation design integrates ergonomics, process flow, visibility, egress, anti-trapping, sensor coverage, and stopping performance.

Task allocation should exploit complementary capabilities and must be checked as a precedence network. Shared handover steps can determine the true cycle time.

AI in robotics spans classical vision, deep-learning perception, reinforcement learning, and language-based assistance. These layers differ in data dependence and verification burden.

2D vision is usually appropriate for structured planar tasks; 3D vision is justified when depth, surface shape, occlusion, or unconstrained pose is essential.

AI outputs should not directly replace validated safety functions. Independent deterministic safeguards, version control, fallback behaviour, and change management remain necessary.

Human oversight is effective only when the operator has adequate information, time, authority, competence, and a safe intervention path.

AI ROI depends on defect prevalence, sensitivity, false rejects, error cost, integration cost, and lifecycle overhead. Detection rates must be applied to the relevant population.

PFL verification requires body-region-specific force and pressure measurement using appropriate test equipment; simplified contact-area calculations do not establish compliance.

Review Questions

Answer the following questions to consolidate your understanding of the chapter material. A thorough response should draw on concepts, formulas, and examples presented in the text.

1. Select representative 10 kg-payload cobot and conventional industrial-robot models from current manufacturer data. Compare mass, reach, speed, stopping information, and safety functions, and explain why payload alone is insufficient for comparison.
2. A cobot hands over welded parts to an operator for visual inspection every 40 s. Select the collaborative operation method or combination of methods and describe the evidence required to validate the application.
3. The cell in Q2 must occasionally move an 8 kg workpiece at 1 m/s. Propose a safe zoning and method-transition strategy that preserves productivity without allowing high-speed motion at an inadequate separation distance.
4. Design the task allocation for collaborative assembly of a small gearbox (shaft, gears, housing, screws, label, functional test). Represent precedence and shared tasks, then justify the allocation using table 11.2.
5. A deep-learning vision system reports 99.7% overall accuracy. What additional data, metrics, stress tests, and fallback evidence are required before deployment?
6. Recommend 2D or 3D vision and an appropriate processing method for: (a) colour-based sorting on a conveyor, (b) random bin picking, and (c) PCB solder-joint inspection. State the assumptions that could change each choice.
7. Under what conditions could an AI-based collision-avoidance function in a cobot application be classified as high-risk under the EU AI Act? Identify the integrator's technical and conformity-assessment responsibilities.
8. Compute the simple payback period for a cobot workstation with CAPEX €58,000, labour saving €24,000/year, estimated ergonomic benefit €11,000/year, and training plus maintenance cost €6,000/year. State two reasons why the result is not a complete investment appraisal.
9. A manufacturer claims that a cobot deployment reduced worker complaints by 78%. Propose at least three independent exposure, health, and process metrics that could confirm or contradict the claim.

References

- [1] International Federation of Robotics (2025) World Robotics 2025 — Industrial Robots. VDMA Services GmbH, Frankfurt.

- [2] International Federation of Robotics (2024) Collaborative Robots: How Robots Work alongside Humans. IFR position paper, updated December 2024.
- [3] ISO 10218-1:2025, Robotics — Safety requirements — Part 1: Industrial robots. ISO, Geneva.
- [4] ISO 10218-2:2025, Robotics — Safety requirements — Part 2: Industrial robot applications and robot cells. ISO, Geneva.
- [5] ISO/TS 15066:2016, Robots and robotic devices — Collaborative robots. ISO, Geneva; confirmed current in 2022.
- [6] ISO/PAS 5672:2023, Robotics — Collaborative applications — Test methods for measuring forces and pressures in human-robot contacts. ISO, Geneva.
- [7] ISO 12100:2010, Safety of machinery — General principles for design — Risk assessment and risk reduction. ISO, Geneva.
- [8] ISO 13849-1:2023, Safety of machinery — Safety-related parts of control systems — Part 1: General principles for design. ISO, Geneva.
- [9] ISO 11228-3:2007, Ergonomics — Manual handling — Part 3: Handling of low loads at high frequency. ISO, Geneva.
- [10] Villani, V., Pini, F., Leali, F., and Secchi, C. (2018) Survey on human-robot collaboration in industrial settings. *Mechatronics*, 55, pp. 248–266.
- [11] Goodfellow, I., Bengio, Y., and Courville, A. (2016) *Deep Learning*. MIT Press.
- [12] Sutton, R. S. and Barto, A. G. (2018) *Reinforcement Learning: An Introduction*, 2nd ed. MIT Press.
- [13] Driess, D. et al. (2023) PaLM-E: An Embodied Multimodal Language Model. *Proceedings of ICML 2023*.
- [14] ISO/IEC 22989:2022; ISO/IEC 23053:2022; ISO/IEC 23894:2023; ISO/IEC 42001:2023. Artificial-intelligence terminology, ML-system framework, risk management, and management systems. ISO/IEC, Geneva.
- [15] European Parliament and Council (2024) Regulation (EU) 2024/1689 laying down harmonised rules on artificial intelligence; European Commission implementation guidance and timeline, accessed 17 July 2026.
- [16] European Parliament and Council (2023) Regulation (EU) 2023/1230 on machinery, applicable from 20 January 2027.
- [17] Universal Robots, FANUC, ABB, KUKA, Doosan Robotics, Techman Robot, and OMRON official collaborative-robot product portfolios, accessed 17 July 2026.
- [18] Universal Robots (2020) Welcome to the Age of the Collaborative Industrial Robots. Corporate history article describing the December 2008 Linatex UR5 installation.
- [19] McAtamney, L. and Corlett, E. N. (1993) RULA: a survey method for the investigation of work-related upper limb disorders. *Applied Ergonomics*, 24(2), pp. 91–99.
- [20] Hignett, S. and McAtamney, L. (2000) Rapid Entire Body Assessment (REBA). *Applied Ergonomics*, 31(2), pp. 201–205.

CHAPTER

12

Adaptive, Reconfigurable, and Mobile Robotic Systems

From fixed lines to responsive, networked production

Twentieth-century manufacturing was built on a productive bargain: specialised equipment traded flexibility for cost-per-unit. A transfer line with seventy custom stations could produce millions of identical engine blocks at a marginal cost no batch job could match. The bargain worked as long as products and volumes remained stable for the decade-plus depreciation horizon of the line.

That bargain has progressively weakened as product lifecycles, demand profiles, and variant mixes have become less predictable. Consumer products are refreshed frequently, automotive platforms support numerous derivatives, and e-commerce increasingly combines high order volumes with product-level customisation. The manufacturing response has therefore been a migration from Dedicated Manufacturing Systems (DMS), through Flexible Manufacturing Systems (FMS), towards Reconfigurable Manufacturing Systems (RMS), complemented by mobile intralogistics that can alter routes and task allocation without rebuilding fixed transport infrastructure.

This chapter introduces the conceptual framework of adaptive production, the engineering characteristics that enable it, and the principal enabling technologies — Autonomous Mobile Robots (AMRs), the Industrial Internet of Things (IIoT), edge and cloud computing, private wireless networks, and digital twins. The discussion connects the classical manufacturing-systems taxonomy with contemporary connected-production architectures while distinguishing mature engineering practice from emerging capability.

LEARNING OBJECTIVES

1. Explain the evolution from Dedicated through Flexible to Reconfigurable Manufacturing Systems and position a given production problem on this spectrum.
2. Apply the six core RMS characteristics — modularity, convertibility, scalability, integrability, diagnosability, and customisation — to evaluate a system design.
3. Distinguish Automated Guided Vehicles (AGVs) from Autonomous Mobile Robots (AMRs), including navigation technology, fleet management, and typical applications.

4. Design a basic AMR-based intralogistics solution including fleet sizing, charging strategy, traffic management, and integration with production scheduling.
5. Evaluate the role of IoT, cloud computing, edge computing, and 5G in enabling connected and adaptive production.
6. Identify the digital-twin architecture required to close the loop between physical production and adaptive planning.
7. Apply ISO 3691-4:2023 and the relevant regional framework to mobile-robot deployments, and identify the additional integration requirements for a mobile manipulator.
8. Calculate key AMR fleet performance metrics including utilisation, throughput, and charging duty cycles.
9. Design a cybersecurity zoning scheme for a connected production environment following IEC 62443 principles.
10. Compare cloud robotics deployment models (on-premise, hybrid, RaaS) and evaluate their suitability for different manufacturing contexts.

12.1 The DMS–FMS–RMS Spectrum

Koren and colleagues formalised the classification of manufacturing systems in a series of papers starting in 1999. The three archetypes differ along two orthogonal dimensions: **capacity** (throughput volume) and **functionality** (variety of products producible).

Table 12.1 compares the three classical manufacturing-system paradigms and an AMR-enabled adaptive-production architecture. Figure 12.1 positions them conceptually on the capacity–functionality plane. The map is a design aid rather than a strict taxonomy: mobile intralogistics changes routing and resource allocation, but it does not by itself make the processing equipment reconfigurable.

Table 12.1. *Three manufacturing-system paradigms and an AMR-enabled adaptive-production architecture*

System Type	Capacity	Functionality	Reconfiguration Cost	Typical Application
DMS	Fixed, very high	Single product	Extreme (new line)	Mass production, cigarettes, beer cans

System Type	Capacity	Functionality	Reconfiguration Cost	Typical Application
FMS	Fixed	Pre-defined variety	Low (re-program)	Engine blocks, turbine blades
RMS	Adjustable	Adjustable within family	Moderate (re-module)	Automotive body-in-white, appliances
AMR-enabled adaptive architecture	Dynamically scalable within installed resources	High routing and logistics flexibility	Low for routing; process changes remain application-dependent	High-mix intralogistics, e-commerce fulfilment, small-batch assembly

The economic logic is straightforward. DMS is appropriate when volume is high, variety is low, and the investment horizon is stable. FMS accommodates a predefined product family mainly through programme, tooling, and fixture changes. RMS extends this capability by allowing the physical and control architecture to be modified — for example by adding or removing stations, exchanging process modules, or rerouting material flow — when capacity or functionality requirements change. AMR-enabled intralogistics complements these paradigms by replacing fixed transport paths with software-coordinated mobile resources; it improves routing flexibility but does not remove the need to reconfigure the production process itself.

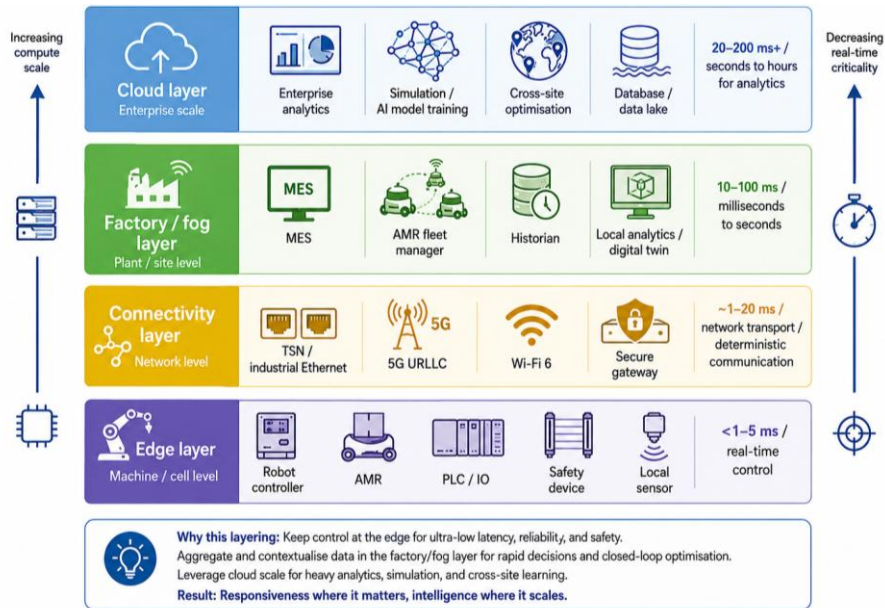


Figure 12.1. Conceptual capacity–functionality positioning of DMS, FMS, RMS, and AMR-enabled adaptive production.

WORKED EXAMPLE — Choosing Between DMS, FMS, and RMS

A consumer electronics manufacturer produces three product families (smartphones, tablets, smartwatches) with a total of 18 SKUs. Annual demand fluctuates $\pm 30\%$ between quarters due to seasonal trends. Product redesign cycles average 14 months. The current DMS line, installed in 2018, requires 6 weeks of retooling per product changeover at a cost of €180,000 per event. Evaluate the three paradigms.

Step 1 — Assess product-mix volatility. 18 SKUs across 3 families with 14-month redesign cycles and $\pm 30\%$ demand swings represent high volatility. DMS is clearly mismatched.

Step 2 — Evaluate FMS suitability. FMS can handle the 18 current SKUs through program changes, but it cannot accommodate new product families without mechanical reconfiguration. With 14-month redesign cycles, the pre-defined product set changes before the FMS has amortised.

Step 3 — RMS analysis. For this illustrative business case, assume modular stations can be reconfigured in 2–3 days at €15,000–25,000 per event, compared with 6 weeks and €180,000 for the legacy DMS. With four changeovers per year and a representative RMS event cost of €20,000, annual avoided changeover cost is $4 \times (\text{€}180,000 - \text{€}20,000) = \text{€}640,000/\text{year}$. Capacity modules and AMR mission planning can then be adjusted to the quarterly demand profile.

Recommendation. Select an RMS architecture with AMR-based intralogistics, subject to a detailed product-family definition and lifecycle cost model. The modular stations address product and capacity evolution; the AMR fleet changes delivery routes and priorities without rebuilding fixed transport infrastructure.

12.2 The Six Characteristics of Reconfigurable Systems

Koren's 2010 monograph identifies six characteristics that a system must possess to deliver genuine reconfigurability. Each adds engineering effort at the design stage and each pays back in reduced adaptation cost later.

12.2.1 Modularity

Functionally distinct subsystems are implemented as physically separable modules with standard interfaces. A welding cell is composed of modules for loading, welding, inspection, and unloading; each module can be replaced, upgraded, or repurposed without touching the others. Standard interfaces span mechanical (bolt patterns, utility connectors), electrical (power, I/O connector types), and informational (field-bus addressing, MES connection). The economic advantage is that only the affected module incurs cost during reconfiguration, while the rest of the system continues to operate.

12.2.2 Convertibility

The system can be converted from producing one member of a planned product family to another with controlled time, cost, and risk. Convertibility is enabled by programmable robots, quick-change end-effectors and fixtures, recipe-controlled process parameters, and modular tooling. The target changeover time is application-specific and must be derived from takt time, batch size, validation effort, and economic loss during changeover. Unlike FMS flexibility, RMS convertibility may require a planned physical module exchange as well as a software or parameter change.

12.2.3 Scalability

Capacity can be increased (or decreased) incrementally rather than in one big step. A palletising cell with two robots and four output conveyors can grow to four robots and eight conveyors by adding modules; a scalable controller architecture anticipates this growth and carries enough I/O headroom and CPU margin to accommodate it. Scalability must be designed into the floor plan (reserved footprint), the electrical infrastructure (oversized panels), and the control architecture (expandable bus) from the start.

12.2.4 Integrability

New modules, sensors, and products can be integrated rapidly through standard mechanical, electrical, communication, and information-model interfaces. OPC UA supports interoperable information exchange, IO-Link standardises communication with many field-level devices, and AutomationML can exchange engineering data between lifecycle tools. Integrability is achieved only when these technologies are implemented with common naming, versioning, addressing, and acceptance-test rules; a protocol alone does not provide plug-and-produce capability.

12.2.5 Diagnosability

The system can identify and localise its own faults quickly. At the module level this means every component reports health continuously; at the system level this means a diagnostics architecture that correlates module-level signals into root-cause narratives. SCADA/MES visibility (Chapter 9) and condition-based monitoring (Chapter 10) are the tactical realisations of diagnosability. Advanced implementations use AI-based anomaly detection to identify degradation before failure occurs.

12.2.6 Customisation

Each module is purpose-built for a specific family of parts, not universal. A truly universal robot can do anything but does nothing well; a customised module does one family of jobs excellently and accepts the restriction. The RMS approach seeks the Goldilocks zone: customised enough to be productive, general enough to support planned variants. This characteristic distinguishes RMS from fully flexible (but slow) job-shop configurations.

Table 12.2 compares the six characteristics in terms of design-stage investment, operational payback, and enabling technologies. Figure 12.2 converts the same six dimensions into a radar profile that can support structured comparison between a candidate design and a defined baseline.

Table 12.2. *The six RMS characteristics: design-stage investment, operational payback, and key enablers*

Characteristic	Design-Stage Investment	Operational Payback	Key Enabler
Modularity	Standard interfaces, module boundaries	Module-level replacement, no full shutdown	Bolt patterns, IO-Link, OPC UA
Convertibility	Quick-change tooling, parametric fixtures	Controlled, shorter changeover than conventional retooling	Robot recipes, quick couplings, automatic tool identification
Scalability	Headroom in floor, power, control	Incremental capacity adjustment	Modular I/O, expandable bus
Integrability	Standard protocols, data models	Reduced module-integration and validation effort	OPC UA, AutomationML, IO-Link
Diagnosability	Sensor coverage, diagnostic software	Fast root-cause, predictive maintenance	CBM, SCADA, AI anomaly detection
Customisation	Family-specific module design	High productivity within product family	Parametric design, modular tooling

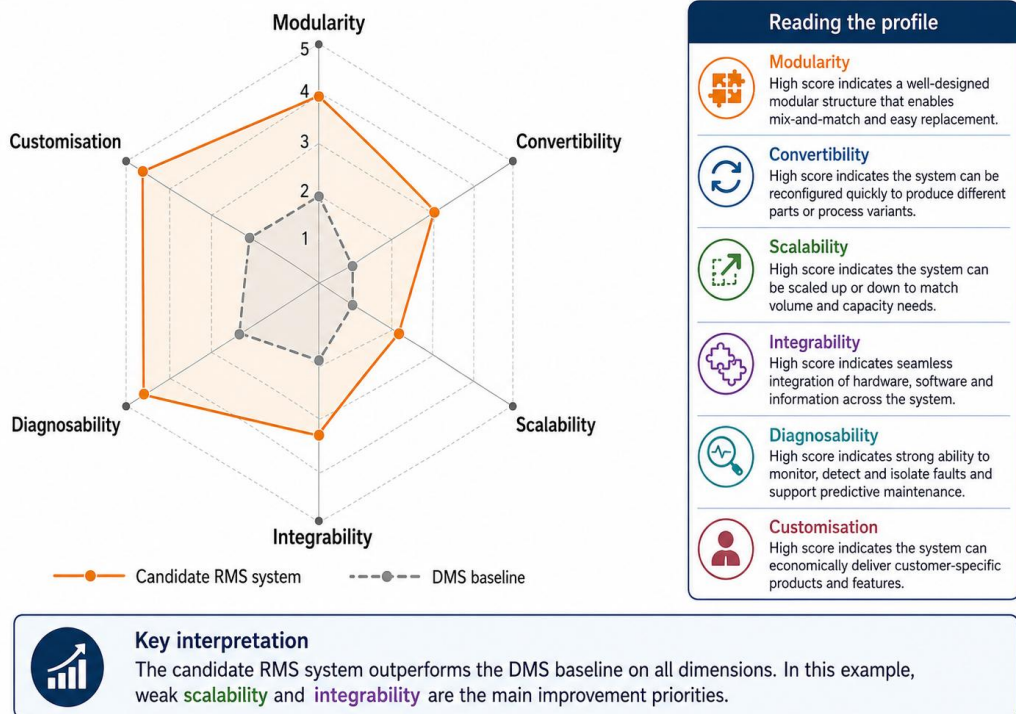


Figure 12.2. Illustrative RMS maturity profile. Scores from 1 to 5 require explicit assessment criteria for each characteristic; the radar plot highlights relative strengths and weaknesses rather than proving reconfigurability.

WORKED EXAMPLE — RMS Maturity Assessment of a Palletising Cell

A palletising cell currently handles two product families (beverage cartons and pet-food bags). Assess its RMS maturity using the six characteristics on a 1–5 scale.

Assessment: Modularity = 4 (separate loading, palletising, and wrapping modules with standard power connectors, but custom fieldbus addressing). Convertibility = 3 (EOAT change takes 25 min; the project target is 15 min). Scalability = 2 (the floor plan permits one additional robot, but PLC I/O is fully allocated). Integrability = 3 (OPC UA is available on the robot controller, whereas the conveyor PLC uses a proprietary interface). Diagnosability = 4 (motor condition signals and a SCADA dashboard are available, but failure isolation remains partly manual). Customisation = 4 (the gripper is optimised for cartons and accepts bags after a minor jaw adjustment).

Overall RMS score: $(4 + 3 + 2 + 3 + 4 + 4)/6 = 3.33$ out of 5. Priority actions, using illustrative budget estimates, are: (1) expand PLC I/O to restore scalability headroom (€2,500); (2) add a standardised gateway and information model for the conveyor interface (€4,000); and (3) reduce EOAT changeover to the 15-min project target with a quick-change coupling (€3,500). The claimed post-improvement score of 4.0 should be confirmed by rescoring each dimension against the same criteria.

INDUSTRIAL PRACTICE NOTE — RMS Readiness Checklist for Existing Installations

Before converting legacy equipment, audit seven dimensions and define project-specific acceptance thresholds: (1) INTERFACE STANDARDISATION — identify every mechanical, electrical, safety, and data interface and reduce unnecessary variants; (2) MODULE BOUNDARIES — measure the time and resources required to isolate, disconnect, exchange, and recommission each functional unit; (3) FLOOR FOOTPRINT — reserve documented space and access for credible expansion scenarios; (4) ELECTRICAL HEADROOM — verify spare panel capacity, protective devices, thermal margin, and power quality; (5) CONTROL EXPANDABILITY — assess spare I/O, CPU and memory utilisation, address space, bus load, and licence limits; (6) DOCUMENTATION CURRENCY — ensure that as-built drawings, software versions, backups, and interface descriptions match the installed system; and (7) WORKFORCE CAPABILITY — confirm that maintenance and production personnel can execute the defined reconfiguration procedure safely. A low score should trigger a lifecycle-cost comparison between retrofit and replacement rather than an automatic RMS label.

Note: This example is illustrative. Numerical values are engineering assumptions for teaching purposes and must not be treated as normative limits or vendor guarantees.

12.3 AGVs and AMRs: The Mobile Revolution

Fixed conveyors are among the largest and most inflexible fixtures in a factory. Every conveyor represents a decision made years before the products it currently moves were designed; every changed route demands structural work. Mobile robots dissolve this

rigidity by replacing fixed paths with responsive vehicles. Two generations of mobile robots exist in production today.

12.3.1 Automated Guided Vehicles (AGVs)

AGVs execute transport missions along validated routes. Guidance may use magnetic tape, embedded wires, optical or QR markers, reflectors, or mapped virtual paths, depending on the generation of the system. Route changes normally require engineering modification and revalidation. When an unexpected obstacle enters the protective field, a conventional AGV generally performs a protective stop and waits for the route to clear, although engineered bypass logic is possible in some systems.

12.3.2 Autonomous Mobile Robots (AMRs)

AMRs combine map-based localisation with autonomous path planning. LiDAR, cameras, wheel odometry, and inertial measurements are fused to estimate pose within a mapped operating area. When an obstacle blocks the nominal path, the vehicle may calculate an alternative route within validated operational constraints. Fleet-management software allocates missions, manages traffic and charging, and exchanges production information with WMS, MES, or orchestration systems. Achievable localisation accuracy and replanning behaviour are application- and vendor-dependent and must be verified in the actual environment.

Table 12.3 summarises typical engineering distinctions between AGVs and AMRs. These labels are not strict regulatory categories and product capabilities overlap. Figure 12.3 shows a representative AMR sensor and control architecture in which navigation sensors, odometry, mission computing, and safety-rated protective devices remain functionally separated.

Table 12.3. Typical engineering distinctions between AGV and AMR solutions; capabilities overlap by product

Feature	AGV	AMR
Navigation	Validated fixed or virtual route; tape, markers, reflectors, or mapped guidance	Map-based localisation with autonomous path planning
Route flexibility	Route modification requires engineering and revalidation	Software remapping and revalidation within the operating map

Feature	AGV	AMR
Obstacle handling	Protective stop; limited predefined diversion where engineered	Local replanning within validated operating constraints
Installation and commissioning	Infrastructure and validation typically require days to weeks	Mapping and validation typically require days; hours only for simple pilots
Cost structure	Lower onboard sensing; route infrastructure and changes can dominate	Higher onboard sensing/software; lower fixed-route infrastructure
Speed	Application, payload, traffic, and safety configuration dependent	Application, payload, traffic, and safety configuration dependent
Payload	Light carts through multi-tonne industrial carriers	Small carts through industrial carriers; product-dependent
Fleet scalability	Constrained by route topology and traffic-control architecture	Software-coordinated, but limited by congestion, charging, and shared resources

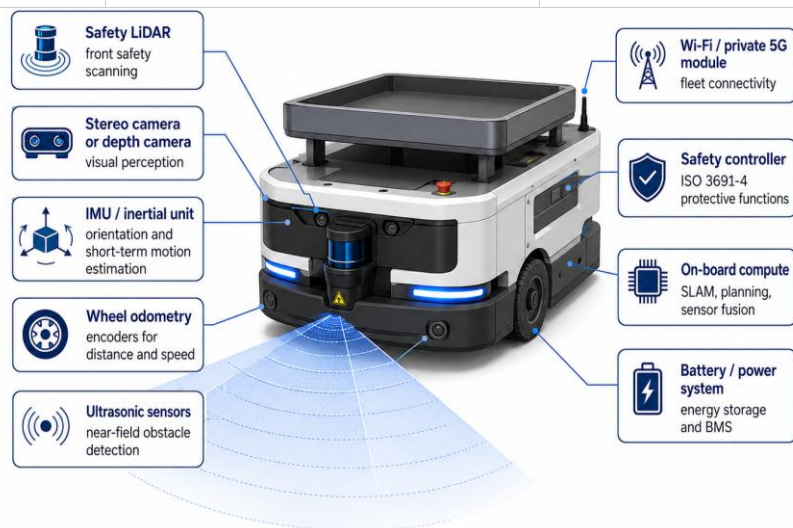


Figure 12.3. Typical AMR architecture.

12.3.3 Safety Standards for Mobile Robots

ISO 3691-4:2023 specifies safety requirements and verification methods for driverless industrial trucks and their systems, explicitly including products commonly described as AGVs and AMRs. It addresses the vehicle and system, including control, braking, stability, protective devices, operating zones, and information for use. In the United States, ANSI/ITSDF B56.5-2024 provides the corresponding framework for driverless guided industrial vehicles; industrial mobile-robot applications may also use ANSI/A3 R15.08. In

the European Union, machinery placed on the market in July 2026 remains subject to Directive 2006/42/EC; Regulation (EU) 2023/1230 applies from 20 January 2027. For a mobile manipulator, the mobile base and the industrial robot application require coordinated risk assessment: ISO 3691-4 applies to the vehicle, while ISO 10218-2:2025 applies to the integrated industrial robot application. ISO/TS 15066 is relevant only to collaborative operation of the manipulator, not as a substitute for mobile-platform safety requirements.

12.3.4 AMR Navigation Technologies

Modern AMRs employ complementary navigation technologies to obtain robust localisation. LiDAR-based SLAM is widely used for map generation and pose estimation in indoor environments. Visual SLAM can add features and semantic information, particularly where geometry alone is insufficient. Wheel odometry and inertial measurement units provide short-term motion propagation but accumulate drift. Outdoors or in suitable semi-outdoor areas, GNSS with real-time kinematic corrections can provide centimetre-level positioning under favourable satellite visibility. Sensor-fusion algorithms combine these measurements, but the resulting accuracy depends on calibration, floor conditions, scene geometry, speed, payload, and the validation method.

Table 12.4 compares the technologies using indicative performance classes and dominant limitations. The values are planning ranges, not guarantees; final selection requires vendor data, an environmental survey, and acceptance testing under representative load, speed, lighting, and traffic conditions.

Table 12.4. AMR navigation technologies: indicative performance, operating conditions, and principal limitations

Technology	Typical Performance	Operational Range	Primary Limitations	Selection Note
2D LiDAR SLAM	Centimetre-level in well-mapped indoor environments	Typically tens of metres; sensor-dependent	Glass, sparse geometry, occlusion, map changes	Common indoor baseline; verify localisation and protective fields separately
3D LiDAR SLAM	Centimetre-level with vertical geometry	Longer and three-dimensional coverage; sensor-dependent	Compute load, dust/fog, reflective surfaces, calibration	Useful where ramps, racks, overhangs, or outdoor transitions matter

Technology	Typical Performance	Operational Range	Primary Limitations	Selection Note
Visual SLAM	Centimetre-to-decimetres class; scene-dependent	Camera field of view and feature visibility	Lighting variation, glare, motion blur, texture-poor surfaces	Cost-effective complement; requires lighting and lens-maintenance control
Wheel odometry + IMU	Accurate short-term propagation; unbounded drift without correction	No absolute range; error grows with distance and time	Wheel slip, tyre wear, vibration, payload-dependent dynamics	Essential fusion input, but not an absolute localisation source
RTK-GNSS	Typically 1–3 cm horizontal under favourable conditions	Outdoor or selected semi-outdoor areas	Indoor signal loss, multipath, correction-link availability	Useful for yards and outdoor logistics; combine with local sensing

WORKED EXAMPLE — Sizing an AMR Fleet for a Production Line

A multi-station assembly line requires component kits to be delivered from a central supermarket to six workstations. Each station consumes one kit every 12 min. Average round-trip (pickup at supermarket, delivery to station, return to supermarket) is 7 min. Average charging downtime per AMR is 15% of operational time. Size the fleet.

Delivery demand. $6 \text{ stations} \times (60/12) \text{ kits/h} = 30 \text{ kits/h} \rightarrow$ average inter-delivery interval 2 min.

Throughput per AMR. $(60 \text{ min/h}) / 7 \text{ min/trip} = 8.57 \text{ trips/h per AMR}$ (before charging).

Availability factor. $1 - 0.15 = 0.85$.

Effective throughput. $8.57 \times 0.85 = 7.29 \text{ trips/h per AMR}$.

Theoretical fleet size. $30 / 7.29 = 4.12 \text{ AMRs}$.

Practical sizing. Round up to five active AMRs. Their average utilisation is $30 / (5 \times 7.29) = 82\%$. Add a sixth vehicle only if the required service level, maintenance policy, or peak-demand analysis justifies a standby. The acceptability of 82% cannot be inferred from a universal congestion threshold; it must be tested against route conflicts, mission-time variability, and charging behaviour.

Charging layout. Size the number and power of charging stations from battery energy, charger power, state-of-charge limits, peak simultaneous demand, and the dispatching policy. A discrete-event simulation is preferable when charging queues share the same aisles or docking resources as production missions.

INDUSTRIAL PRACTICE NOTE — AMR Floor Readiness Checklist

Before commissioning an AMR fleet, audit: (1) FLOOR GEOMETRY — compare slopes, steps, joints, gaps, and local damage with the vehicle manufacturer's limits at maximum payload; (2) TRACTION AND SURFACE CONDITION — verify friction, contamination, drainage, and any reflective or transparent surfaces that affect sensors; (3) CLEARANCES AND CROSSINGS — define pedestrian routes, intersections, escape paths, and speed zones through the application risk assessment; (4) LIGHTING — test camera-based functions across the actual illumination range, glare, flicker, and day/night transitions; (5) WIRELESS PERFORMANCE — survey coverage, roaming, interference, redundancy, and recovery time for every service that depends on connectivity; (6) DOCKING AND LOAD TRANSFER — validate mechanical alignment, fiducials, tolerances, load stability, and failed-docking recovery; and (7) MAP GOVERNANCE — establish controlled procedures for layout changes, temporary obstacles, software versions, and revalidation. Numerical limits must come from the selected AMR and site risk assessment rather than from a generic checklist.

Note: This example is illustrative. Numerical values are engineering assumptions for teaching purposes and must not be treated as normative limits or vendor guarantees.

12.4 IIoT, Cloud, and the 5G Enabler

Adaptive production systems require a communication and computing architecture that preserves real-time control while making selected information available across cells, production areas, and sites. Four technical pillars are particularly important: semantic interoperability, workload partitioning across edge and cloud resources, fit-for-purpose wired and wireless networking, and cybersecurity engineered throughout the lifecycle.

12.4.1 OPC UA and the Semantic Layer

OPC Unified Architecture (IEC 62541) provides a platform-independent framework for secure communication and information modelling between devices, control systems, and enterprise software. Rather than transmitting only raw values, OPC UA can expose structured objects with types, units, quality information, events, and relationships. Domain-specific Companion Specifications improve semantic interoperability. OPC UA for Robotics, Part 1: Vertical Integration (OPC 40010-1, version 1.02, 2025) defines

information for supervising and monitoring motion-device systems, including identification, operating state, condition data, and related diagnostics; it does not standardise robot motion commands or replace a real-time robot interface.

12.4.2 MQTT and Lightweight Publish-Subscribe

OPC UA and MQTT serve complementary roles rather than a strict client-server versus publish-subscribe split. OPC UA provides information modelling, services, security, and both client-server and PubSub mechanisms. MQTT is a lightweight broker-based transport frequently used for telemetry and cloud integration, with QoS levels 0, 1, and 2. A common architecture exposes semantically defined machine data through OPC UA and forwards selected telemetry through MQTT, sometimes using OPC UA PubSub over MQTT. Safety and hard real-time control should remain independent of a general-purpose cloud messaging path.

12.4.3 Edge, Fog, and Cloud Computing

Data can be processed at three successive layers:

- **Edge** (on the device, controller, or nearby industrial computer): deterministic access to local signals, minimal dependence on external networks, and bounded compute resources. Suitable for motion execution, safety-related functions implemented on certified hardware, local data acquisition, and time-critical vision.
- **Fog** (factory- or site-level compute): low-latency access over the plant network with sufficient resources for MES functions, cross-cell coordination, local historians, fleet orchestration, and site analytics.
- **Cloud** (remote or geographically distributed datacentre): elastic compute and storage with variable end-to-end latency and external-network dependency. Suitable for cross-site analytics, large-scale model training, software distribution, and non-time-critical optimisation.

The engineering task is to place each function at the layer that satisfies its response-time, safety-integrity, availability, data-governance, and maintainability requirements. Safety functions and servo loops remain local and deterministic. Factory orchestration can use site-level compute, while cross-site model training benefits from cloud resources. Hybrid architectures are therefore the norm; “edge”, “fog”, and “cloud” describe responsibility boundaries rather than fixed latency numbers.

Table 12.5 gives an indicative partitioning of common functions. The latency entries are engineering classes, not normative limits; the validated requirement must come from the machine risk assessment, control-loop design, and network architecture.

Table 12.5. *Recommended edge–fog–cloud partitioning for common industrial functions*

Function	Recommended Layer	Latency Requirement	Data Volume
Emergency-stop and protective-stop logic	Local safety-rated control	Response time derived from risk assessment and safety design	Small, deterministic signals
Motion execution and servo coordination	Edge	Microseconds to milliseconds, control-dependent	Continuous cyclic data
Vision inspection	Edge/Fog	Milliseconds to seconds, application-dependent	Images or point clouds
OEE calculation	Fog	Seconds to minutes	Events and counters
Production scheduling	Fog/Cloud	Seconds to hours	Orders, resources, constraints
Cross-site model training	Cloud or central compute	Hours to days	Large historical datasets
Fleet route optimisation	Fog/Cloud	Sub-second to minutes, mission-dependent	Maps, missions, traffic state

INDUSTRIAL PRACTICE NOTE — 5G Site Survey for Factory Deployment

A private-5G survey must cover more than radio signal strength. (1) USE-CASE REQUIREMENTS — define end-to-end latency, jitter, availability, device density, mobility, uplink demand, and recovery behaviour for each traffic class. (2) PROPAGATION AND MOBILITY — map metal structures, racks, moving cranes, doors, outdoor transitions, handover zones, and antenna diversity using representative drive tests. (3) INTERFERENCE AND EMC — measure the radio environment and assess coexistence with other wireless systems; do not assume that industrial electrical equipment interferes specifically at the 5G carrier frequency without measurement. (4) CORE AND TRANSPORT RESILIENCE — design local breakout, time synchronisation, power backup, redundant backhaul,

monitoring, and degraded-mode operation. (5) SPECTRUM AND AUTHORISATION — confirm the current Romanian allocation and authorisation route with ANCOM for the selected deployment model and band. Availability of local industrial spectrum, licence conditions, and lead time can change; regulatory confirmation belongs at project initiation, not after radio design. Site-survey effort and cost are site-specific and should be quoted against a defined acceptance-test plan.

Note: This example is illustrative. Numerical values are engineering assumptions for teaching purposes and must not be treated as normative limits or vendor guarantees.

12.4.4 5G and Time-Sensitive Networking

Conventional wireless networks have often been unsuitable for deterministic or safety-related industrial control because end-to-end latency, jitter, roaming, and availability are difficult to bound. Private 5G can provide managed quality of service, mobility, traffic isolation, and support for demanding industrial service profiles. However, the frequently quoted “1 ms and 99.999%” figures are scenario-specific requirements rather than universal end-to-end guarantees. Time-Sensitive Networking (TSN) is a family of IEEE 802.1 mechanisms for time synchronisation, traffic scheduling, policing, and redundancy in bridged Ethernet networks. 5G–TSN integration can extend deterministic networking concepts to mobile equipment, but the complete path — device, radio, core network, gateway, TSN domain, and application — must be engineered and validated. Safety functions should not depend solely on a non-safety-rated wireless service.

Figure 12.4 integrates these concepts into a layered reference architecture: safety-rated and real-time functions at the edge; a wired TSN and managed wireless connectivity layer; factory-level orchestration and data services; and cloud resources for cross-site analytics and model lifecycle management. The arrows should distinguish control paths from telemetry paths and show that only suitable workloads cross each boundary.

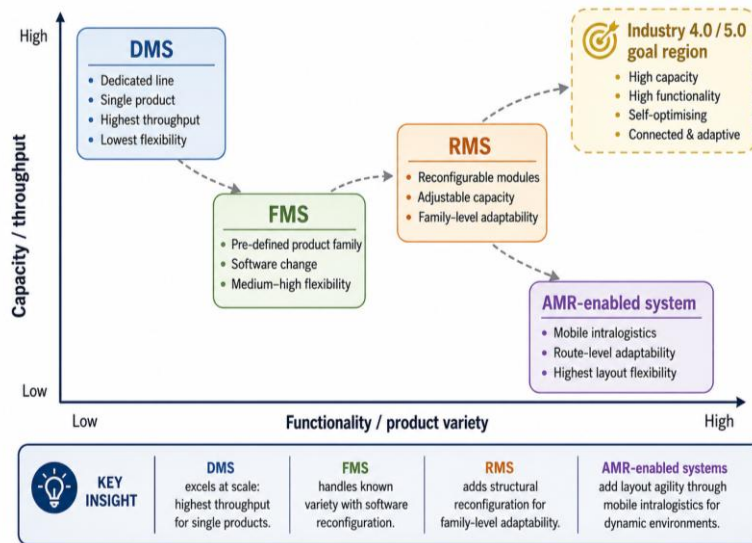


Figure 12.4. Layered architecture of an adaptive production system.

WORKED EXAMPLE — AMR Throughput with Opportunity Charging

Compute the steady-state throughput of a five-AMR fleet. Each AMR provides 4 h (240 min) of usable run time and requires 30 min for a full recharge. Each 6-min mission already includes two 30-s pick/drop dwell periods during which opportunity charging is available. Assume, initially, an ideal linear charging rate derived from the full-charge specification and ignore taper and losses.

Nominal mission capacity. Five AMRs $\times$ (60 min/h)/(6 min/mission) = 50 missions/h, provided that charging does not add cycle time.

Without opportunity charging. The battery duty factor is $240/(240 + 30) = 0.889$. Average fleet throughput is therefore $50 \times 0.889 = 44.4$ missions/h if full-recharge downtime is distributed evenly.

With opportunity charging. A 30-min full charge restores 240 min of run time, corresponding to an ideal ratio of 8 min of run time per minute of charging. The two dwell periods provide 1 min of charging per mission and therefore 8 min of equivalent run time, exceeding the 6 min consumed by the mission. Under the stated ideal assumptions, the fleet is energy-neutral and can sustain the nominal 50 missions/h without scheduled full-charge downtime. This is 12.5% higher than

44.4 missions/h. In practice, charger efficiency, battery taper at high state of charge, thermal limits, and unavailable docks reduce this margin and must be modelled from manufacturer data.

Business implication. Opportunity charging is valuable when it is concurrent with unavoidable dwell time and does not create docking queues. Battery and charger sizing should be based on an energy balance and state-of-charge simulation, not on battery capacity alone.

12.5 Digital Twin for Adaptive Production

Chapter 10 introduced the digital twin as a lifecycle-management concept. In adaptive production, the twin may support monitoring, diagnosis, prediction, optimisation, or controlled adaptation. ISO 23247 provides a manufacturing-oriented digital-twin framework, but terminology in commercial literature remains inconsistent. A useful engineering distinction is based on data flow and authority: a digital model has no automatic live connection; a digital shadow receives automatic data from the physical system; and a closed-loop digital twin supports governed bidirectional interaction. Bidirectionality does not imply that an optimisation algorithm may directly command safety-critical equipment without validation and an independent safety architecture.

- 1. Digital model.** Static virtual representation, no automatic data flow. Used for training and offline simulation.
- 2. Digital shadow.** One-way automatic update from physical to virtual. Enables monitoring, diagnostics, post-hoc analysis.
- 3. Digital twin.** Bidirectional, synchronised update. The virtual system influences the physical one through control actions or parameter adjustments.

Table 12.6 summarises these three working levels. Update rate alone does not determine whether a system is a twin: semantic fidelity, synchronisation, model validation, and governed influence on the physical asset are equally important. Figure 12.5 should therefore show both data direction and the decision/approval boundary between analytics and execution.

Commercial platforms cover different portions of the digital-twin lifecycle. NVIDIA Omniverse and Isaac Sim support physically based simulation, synthetic data generation,

and robot-policy development. Siemens Tecnomatix and Insights Hub combine manufacturing simulation with industrial data integration and operations analytics. Dassault Systèmes 3DEXPERIENCE connects product, process, simulation, and operational information. Selection should follow validated use cases, data ownership, integration effort, lifecycle cost, and the engineering team’s installed toolchain rather than a claim of universal platform superiority.

Table 12.6. *Three levels of digital-twin sophistication with their data-flow characteristics and typical applications*

Level	Data Flow	Update Frequency	Typical Use	Example Platform
Digital model	None (manual)	On demand	Training, offline sim	SolidWorks, RoboDK
Digital shadow	Physical → Virtual	Event-driven or periodic	Monitoring, dashboards	Siemens Insights Hub, industrial historian
Digital twin	Bidirectional	Use-case-driven; sufficiently synchronised	Adaptive control, optimisation	NVIDIA Omniverse/Isaac Sim, integrated plant platforms

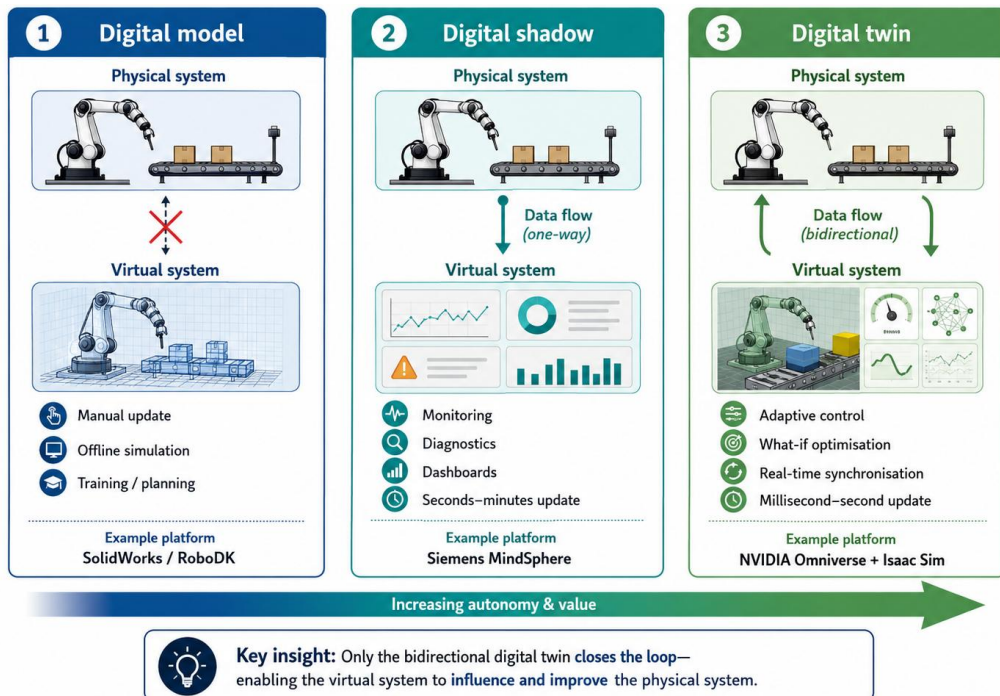


Figure 12.5. Digital twin data-flow architecture showing the three levels.

WORKED EXAMPLE — Digital Twin ROI for a Multi-Cell Production System

A manufacturer operates four robotic welding cells. Unplanned downtime averages 120 h/year per cell at an estimated contribution loss of €850/h. A digital-shadow deployment costs €45,000 per cell and is expected to reduce unplanned downtime by 35%. A closed-loop digital-twin deployment costs €110,000 per cell and is expected to reduce downtime by 60%. It may also increase saleable output by 4%, but the financial value of that output depends on market demand and contribution margin. Evaluate both options over three years, ignoring discounting and recurring operating costs.

Current downtime cost. $4 \text{ cells} \times 120 \text{ h} \times €850/\text{h} = €408,000/\text{year}$.

Digital shadow (Level 2). Investment: $4 \times €45,000 = €180,000$. Annual saving: $€408,000 \times 0.35 = €142,800$. Simple payback: $€180,000 / €142,800 = 1.26 \text{ years}$. 3-year ROI: $(€142,800 \times 3 - €180,000) / €180,000 = 138\%$.

Digital twin. Investment: $4 \times \text{€}110,000 = \text{€}440,000$. Downtime saving: $\text{€}408,000 \times 0.60 = \text{€}244,800/\text{year}$. Conservative case, with no monetised throughput gain: payback = $\text{€}440,000/\text{€}244,800 = 1.80$ years; three-year ROI = $(\text{€}244,800 \times 3 - \text{€}440,000)/\text{€}440,000 = 67\%$. Capacity-constrained case: if annual saleable output is €5 million and incremental output earns a 30% contribution margin, the 4% output gain contributes $\text{€}5,000,000 \times 0.04 \times 0.30 = \text{€}60,000/\text{year}$. Total annual benefit becomes €304,800; payback = 1.44 years; three-year ROI = 108%.

Recommendation. The digital shadow has the lower implementation risk and a 1.26-year simple payback under the stated assumptions. The digital twin may produce the higher strategic value, but its business case is sensitive to demand, contribution margin, validation effort, recurring software cost, and authority to act on the physical process. A staged deployment on two cells is justified before scaling the closed loop.

12.6 Cloud Robotics and Fleet Intelligence

A connected robot can participate in shared monitoring, software lifecycle management, fleet optimisation, and cross-site analytics. The value comes from governed data and services, not from direct unrestricted Internet access. Five common capability classes are:

- **Remote monitoring and diagnostics.** Authorised service personnel access selected data through controlled gateways, audited sessions, and role-based permissions, reducing diagnosis time without exposing controllers directly to the public Internet.
- **Managed software distribution.** Controller firmware, application code, and vision models can be deployed in staged, signed, and rollback-capable waves. Safety-related configuration changes require the validation and change-control process defined for the machine; they should not be treated as ordinary over-the-air updates.
- **Cross-fleet learning.** Anomaly patterns detected on one robot trigger preventive action on others; successful programs developed on one cell propagate to siblings.
- **SaaS-based planning.** Fleet-optimisation, route-planning, and scheduling services consumed from the cloud rather than licensed on premise.

- **Robotics-as-a-Service (RaaS).** Subscription, availability-based, or pay-per-use commercial models shift part of the integration, maintenance, and asset-performance responsibility to the provider. Pricing and included service levels vary substantially with payload, operating hours, site conditions, and contractual risk allocation.

12.6.1 Cybersecurity of Connected Production

Connectivity expands the attack surface of industrial automation and control systems (IACS). The IEC 62443 series provides a lifecycle framework for asset owners, integrators, service providers, and component suppliers. IEC 62443-3-2 requires the system under consideration to be partitioned into zones and conduits, followed by risk assessment and assignment of target security levels; IEC 62443-3-3 defines system security requirements. Good practice includes least-privilege access, authenticated and encrypted communications where appropriate, network segmentation, controlled remote access, asset inventory, logging, backup and recovery, vulnerability management, and patching planned around operational risk. Security-level targets must be derived from risk; they should not be assigned merely because a component is located at a particular Purdue level.

INDUSTRIAL PRACTICE NOTE — IEC 62443 Zoning for a Connected Robotic Factory

An IEC 62443-inspired design for five robotic cells can be structured as follows. **CELL ZONES:** each cell contains the robot controllers, safety PLC, local HMI, drives, and cell devices; traffic leaving the zone passes through an industrial firewall or security gateway. **PRODUCTION-OPERATIONS ZONE:** MES interfaces, SCADA, historian, fleet manager, and engineering services are segmented from individual cells and use allow-listed conduits. **INDUSTRIAL DMZ:** brokers, update repositories, remote-access jump hosts, and data-transfer services mediate traffic between operations and enterprise/cloud networks; direct enterprise-to-controller sessions are prohibited. **ENTERPRISE/CLOUD ZONES:** ERP, business analytics, and approved cloud services receive only the data and services required for their use cases. **CONDUITS** specify protocol, direction, identity, encryption, logging, rate limits, and fail-safe behaviour — for example OPC UA with certificate management for

semantic data, vendor-approved engineering protocols through a jump host, and brokered MQTT for selected telemetry. Target security levels are assigned separately to each zone and conduit after the IEC 62443-3-2 risk assessment. A data diode is appropriate only for genuinely one-way flows; remote diagnostics or command paths require a controlled bidirectional architecture. Incident response should support isolation of an affected cell while preserving safe operation of unaffected zones.

Note: This example is illustrative. Numerical values are engineering assumptions for teaching purposes and must not be treated as normative limits or vendor guarantees.

Common Pitfalls

Pitfall 1: Equating connectivity with adaptability.

A factory that connects every device to the cloud but never changes its operations in response is digitised but not adaptive. Adaptive production requires closed-loop decisions, not only open-loop data collection.

Pitfall 2: Under-investing in the semantic layer.

OPC UA information models and MES schemas are not decorations; they are the vocabulary in which the adaptive system thinks. A factory that pumps raw numeric values into the cloud without semantic structure is pumping noise.

Pitfall 3: Deploying AMRs into unprepared floors.

AMRs need predictable floors (level, non-reflective, uncluttered), coherent WiFi coverage, and a digital map that matches reality. Deploying AMRs before the floor is ready yields erratic behaviour and operator rejection.

Pitfall 4: Treating cybersecurity as an IT problem.

OT cybersecurity is a distinct discipline. It shares principles with IT but differs in priorities: availability matters more than confidentiality, patching windows are rare and short, and consequences of failure are physical.

Pitfall 5: Over-promising the digital twin.

A static virtual model labelled ‘digital twin’ that is not synchronised with reality is a marketing slide, not an engineering tool. Genuine digital twins demand sensor investment, data pipelines, and fidelity-calibration discipline.

Pitfall 6: Sizing AMR fleets for average demand only.

Average-demand sizing can leave no reserve for peaks, charging, traffic conflicts, blocked routes, or maintenance. Fleet sizing should therefore evaluate peak and percentile demand, mission-time variability, congestion through simulation or queueing analysis, and an explicit resilience policy. There is no universal utilisation threshold: acceptable utilisation depends on layout, dispatching logic, charging strategy, and service-level requirements.

Pitfall 7: Ignoring human factors in AMR deployments.

Operators who feel unsafe around AMRs will find ways to disable or avoid them. Early involvement of floor operators in route planning, safety briefings, and feedback cycles is essential for acceptance.

Pitfall 8: Confusing FMS flexibility with RMS reconfigurability.

FMS handles variety within a pre-defined product set by program change; RMS handles unforeseen product evolution by physical reconfiguration. The distinction matters when planning capital investment horizons.

Pitfall 9: Neglecting scalability headroom in initial design.

A cell designed with zero spare I/O slots, zero floor footprint margin, and a fully loaded PLC cannot scale without a complete control-system replacement — destroying the economic case for incremental growth.

Pitfall 10: Assuming 5G replaces wired connections everywhere.

Private 5G can support mobile platforms, temporary equipment, and selected time-critical coordination functions, but it does not automatically replace safety-rated wired networks or local servo buses. Fixed equipment with tightly bounded control cycles will often continue to use EtherCAT, PROFINET IRT, or another validated wired architecture, while 5G carries supervisory control, mission data, video, and mobility-dependent traffic.

KEY TAKEAWAYS

- DMS, FMS, and RMS describe production-system architectures; AMR-enabled intralogistics is a complementary capability that increases routing and resource-allocation flexibility.
- RMS is defined by six characteristics — modularity, convertibility, scalability, integrability, diagnosability, customisation — each contributing to adaptive capability.
- AMRs replace fixed conveyors with navigation-intelligent mobile platforms; fleet management software is as important as the vehicles themselves.
- Mobile-robot safety is governed by ISO 3691-4:2023 and the applicable regional framework. For mobile manipulators, the vehicle and industrial-robot application standards must be integrated through one risk assessment.
- OPC UA provides semantic interoperability and MQTT provides lightweight brokered messaging; neither should be treated as the sole path for safety or hard real-time control.
- Edge–fog–cloud partitioning assigns workloads according to safety, determinism, availability, data governance, and compute requirements. Private 5G and TSN can support converged architectures only when the complete end-to-end path is engineered and validated.
- Digital models, digital shadows, and closed-loop twins represent increasing connectivity and authority, but update frequency alone does not establish a genuine digital twin.
- Cloud robotics enables cross-fleet learning and Robotics-as-a-Service business models, but introduces cybersecurity obligations governed by IEC 62443.
- AMR fleet sizing must account for charging energy, peak and percentile demand, mission-time variability, route conflicts, maintenance, and the required service level — not only average utilisation.
- Scalability headroom (floor footprint, electrical capacity, control I/O, communication bandwidth) must be designed into every RMS installation from the start.

Review Questions

Answer the following questions to consolidate your understanding of the chapter material. A thorough response should draw on concepts, formulas, and examples presented in the text.

1. Explain the economic logic of the DMS → FMS → RMS progression in terms of product lifecycle length and variety. Name one industry that remains best served by DMS and justify your choice.
2. A body-in-white line currently produces one car model. The OEM wants to add a second derivative. Which of the six RMS characteristics does this change primarily stress, and what engineering interventions would you prioritise?
3. Compare an AGV and an AMR solution for inter-station material delivery in a factory with 40 workstations spread over 8,000 m². Include installation cost, reconfiguration cost, and long-term flexibility in your analysis.
4. A fleet of AMRs must serve 10 workstations with average delivery demand of 60 trips/hour, average mission time of 8 min, and 20% charging downtime. Compute the theoretical minimum fleet size. Then propose a practical fleet size with operational reserve and explain how congestion, blocked routes, and mission-time variability affect the decision.
5. A new production site is being built with strong adaptive-production ambitions. Specify the communication architecture (network segments, protocols, determinism requirements, wireless) in enough detail for a vendor RFP.
6. Explain the edge–fog–cloud model and place the following functions at the appropriate layer: motion control, vision inspection, OEE calculation, cross-site model training, emergency stop logic, production scheduling.
7. Compare private 5G with Wi-Fi 6/6E for an AMR fleet. Address coverage, roaming, quality of service, core-network availability, time synchronisation, cybersecurity, and spectrum authorisation. Explain why radio latency alone does not establish end-to-end determinism and how the Romanian regulatory environment should be checked during project planning.
8. A digital-twin platform is proposed that updates the virtual model every 5 minutes. Classify it on the model/shadow/twin scale and identify the applications it can and cannot support. What sensor investment would be needed to upgrade it to a true digital twin?
9. Outline an IEC 62443-inspired zoning scheme for a factory with five robotic cells, a central MES, and a cloud telemetry link. Identify the conduits between zones and the type of security controls required at each.

References

- [1] Koren, Y. (2010) *The Global Manufacturing Revolution: Product–Process–Business Integration and Reconfigurable Systems*. Wiley, Hoboken, NJ. ISBN 978-0-470-58377-7.
- [2] Koren, Y., Gu, X., and Guo, W. (2018) Reconfigurable manufacturing systems: Principles, design, and future trends. *Frontiers of Mechanical Engineering*, 13(2), pp. 121–136.
- [3] ISO 3691-4:2023, *Industrial trucks — Safety requirements and verification — Part 4: Driverless industrial trucks and their systems*. ISO, Geneva.
- [4] ANSI/ITSDF B56.5-2024, *Safety Standard for Driverless, Automatic Guided Industrial Vehicles and Automated Functions of Manned Industrial Vehicles*. ITSDF/ANSI, Washington, DC.
- [5] ISO 10218-2:2025, *Robotics — Safety requirements — Part 2: Industrial robot applications and robot cells*. ISO, Geneva.
- [6] IEC 62541-1:2020, *OPC Unified Architecture — Part 1: Overview and concepts*. IEC, Geneva.
- [7] OPC Foundation (2025) *OPC 40010-1, OPC UA for Robotics — Part 1: Vertical Integration*, Release 1.02. OPC Foundation.
- [8] ISO/IEC 20922:2016, *Information technology — Message Queuing Telemetry Transport (MQTT) v3.1.1*. ISO/IEC, Geneva.
- [9] IEEE 802.1Q-2022, *IEEE Standard for Local and Metropolitan Area Networks — Bridges and Bridged Networks*. IEEE, Piscataway, NJ.
- [10] IEC 62443-3-2:2020, *Security for industrial automation and control systems — Part 3-2: Security risk assessment for system design*. IEC, Geneva.
- [11] IEC 62443-3-3:2013, *Industrial communication networks — Network and system security — Part 3-3: System security requirements and security levels*. IEC, Geneva.
- [12] 3GPP TS 22.104 (Release 17), *Service requirements for cyber-physical control applications in vertical domains*. 3rd Generation Partnership Project.
- [13] ISO 23247-1:2021, *Automation systems and integration — Digital twin framework for manufacturing — Part 1: Overview and general principles*. ISO, Geneva.
- [14] Regulation (EU) 2023/1230 of the European Parliament and of the Council on machinery and repealing Directive 2006/42/EC.
- [15] Grieves, M. and Vickers, J. (2017) Digital twin: Mitigating unpredictable, undesirable emergent behavior in complex systems. In: *Transdisciplinary Perspectives on Complex Systems*. Springer, pp. 85–113.
- [16] ElMaraghy, H. A. (ed.) (2009) *Changeable and Reconfigurable Manufacturing Systems*. Springer, London. ISBN 978-1-84882-066-1.
- [17] Wurman, P. R., D’Andrea, R., and Mountz, M. (2008) Coordinating hundreds of cooperative, autonomous vehicles in warehouses. *AI Magazine*, 29(1), pp. 9–20.
- [18] Groover, M. P. (2020) *Automation, Production Systems, and Computer-Integrated Manufacturing*, 5th ed. Pearson. ISBN 978-0-13-460546-3.

CHAPTER

13

Industrial Case Studies and Best Practices

Five sectors, seven lessons, and the patterns that transfer

Textbook principles acquire meaning through the installations that embody them. A cycle-time formula, a safety-mode decision tree, or an OEE calculation becomes concrete when traced through a production system in which engineers must balance performance, safety, quality, cost, and organisational constraints. This chapter examines five composite case studies from five industrial sectors, selected to span the design-to-operation framework developed in the preceding twelve chapters.

The selection covers sectors in which robotic production systems are sufficiently mature to reveal distinct engineering priorities: automotive body-in-white, electronics inspection, food packaging, pharmaceutical inspection, and logistics. Each case is analysed through the same framework and then compared for recurring patterns of success, failure, and post-commissioning improvement.

LEARNING OBJECTIVES

1. Analyse a robotic installation by its design decisions, operational performance, and economic outcome, identifying the engineering choices most responsible for success or failure.
2. Extract transferable principles from sector-specific case studies and apply them to new design problems in unfamiliar industries.
3. Evaluate ROI of automation investments using payback period, NPV, and IRR, selecting the appropriate metric for the decision context.
4. Benchmark proposed performance against industry norms and identify outliers that deserve additional scrutiny.
5. Recognise the seven common patterns of robotic-system success and failure that recur across industrial sectors.
6. Formulate a post-deployment continuous-improvement plan that evolves the system from commissioning performance toward benchmark performance.
7. Perform sensitivity analysis on automation business cases to identify the assumptions that most affect project viability and risk.
8. Design a structured post-commissioning review process that captures lessons learned and feeds them into the organisation's design standards.

13.1 A Framework for Case-Study Analysis

Every robotic installation can be described through a six-element framework that facilitates comparison across installations and across sectors:

1. **Context.** The business problem, the product, the volume, the alternative considered.
2. **System design.** Layout type, number of robots, robot types, EoAT, auxiliary equipment, control architecture, safety approach.
3. **Performance achieved.** Cycle time, throughput, OEE, yield, and comparison with the predecessor system.
4. **Economic result.** Investment, payback period, ROI, any hidden costs or benefits that emerged only in operation.
5. **Lessons learned.** What worked as designed, what surprised the team, what they would do differently.

6. Transferable principles. Which aspects generalise to other sectors and which are irreducibly sector specific.

Figure 13.1 visualises these six elements as a directed flow with a learning feedback loop from Lessons back to Context. Table 13.1 then previews the five cases examined in this chapter, summarising their sectors, indicative investment scales, economic metrics, key technologies, and principal business drivers.

The cases that follow adopt this framework. Some cases emphasise certain elements more than others — the pharmaceutical case gives weight to regulatory compliance, the logistics case to fleet operations — but all six elements are addressed for each installation.

Methodological note. The five cases are composite teaching cases assembled from published technical literature, vendor documentation, standards, and representative industrial ranges. They are not claims about named production sites. Brand names, where retained, identify technology classes rather than verified project configurations. Unless stated otherwise, performance values represent stabilised post-ramp-up conditions rather than commissioning-day results; all economic results depend on the assumptions declared in each example.

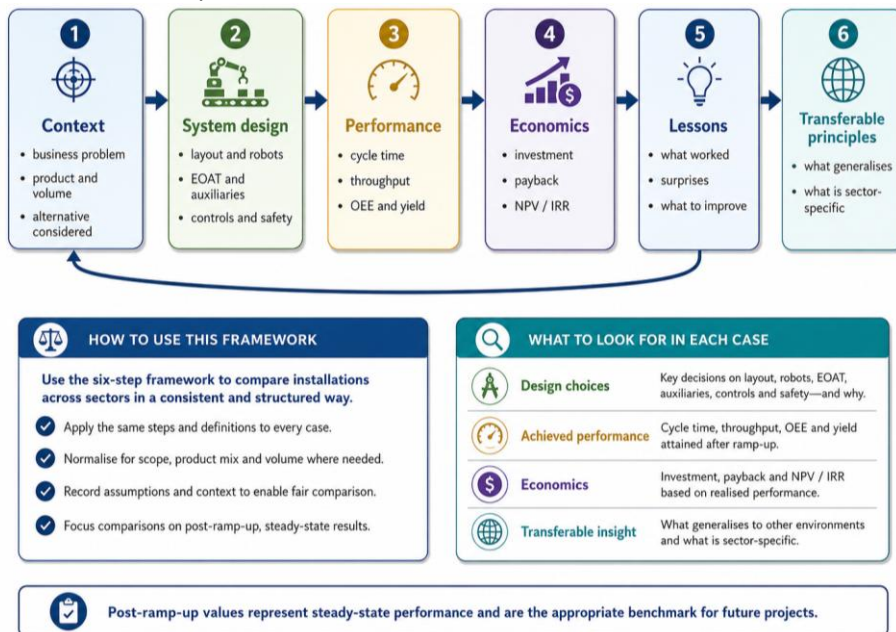


Figure 13.1. The six-element case-study analysis framework: Context → System Design → Performance → Economics → Lessons → Transferable Principles. The feedback loop from Lessons to Context represents organisational learning transferred to future projects.

Table 13.1 compares five deliberately different automation contexts. The range extends from a capital-intensive body-in-white refurbishment to an AI-assisted electronics-inspection upgrade and an AMR fleet procured through an operating-expenditure model. The comparison is intended to show that the appropriate economic metric depends on the business model: simple payback may be informative for a capital project, whereas an RaaS decision requires discounted cash-flow and service-scope comparison rather than a nominal payback figure.

Across the five cases, the technical driver shifts from throughput and variant capability to quality, hygienic design, validated data integrity, ergonomics, and logistical responsiveness. The common engineering problem is not the selection of a robot in isolation, but the design of a complete socio-technical production system whose interfaces, validation evidence, operator practices, and lifecycle costs remain coherent.

Table 13.1. *Summary comparison of the five case studies examined in this chapter*

Case	Sector	Investment	Payback	Key Technology	Primary Driver
Case 1	Automotive BIW	€18.5M	3.0 yr	Vision-guided welding	Throughput + variants
Case 2	Electronics	€185k	1.2 yr	Deep-learning AOI	Quality + labour
Case 3	Food packaging	€2.1M	2.1 yr	Hygienic delta robots	Labour + waste
Case 4	Pharmaceutical	€340k	3.8 yr	Cobot + hyperspectral	Inspection evidence
Case 5	Logistics	€1.58M/yr RaaS	NPV-based	120 AMRs + fleet mgmt	Productivity + ergonomics

13.2 Case 1 — Automotive Body-in-White Spot Welding

Automotive body-in-white (BIW) production is the longest-established large-scale application of industrial robotics. The following composite case represents the refurbishment of a European high-volume line originally commissioned in the early 2000s and modernised approximately two decades later.

Context

The original line ran 45 JPH for a single model, with 98 robots. The business demand was to increase throughput to 60 JPH while accommodating three model variants on the same floor, within the existing building envelope.

System Design

The refurbished line uses 120 high-payload KUKA KR QUANTEC-class robots across 14 welding stations. Flexible fixturing and quick-change tooling support an engineered 15-minute changeover. Vision and weld-process monitoring verify part position, gun alignment, and selected quality indicators at critical stations. Time-critical cell communication uses PROFINET IRT, while safety-oriented workspace and axis monitoring are implemented with KUKA.SafeOperation and fail-safe control functions.

Table 13.2 quantifies the impact of this refurbishment across seven performance metrics, comparing the 2002 baseline to the 2020 refurbished state.

Performance Achieved

Table 13.2 compares the illustrative legacy and refurbished states. Throughput rises from 45 to 62 jobs per hour, while three variants replace the former single-model configuration. The most consequential flexibility gain is the reduction of changeover time from 6 hours to 15 minutes, because variant production becomes operationally feasible only when changeover losses no longer dominate the schedule.

Table 13.2. Performance comparison for the 2002 → 2020 BIW line refurbishment

Metric	Before (2002)	After (2020)	Improvement
Throughput (JPH)	45	62	+38%
Model variants	1	3	+200%
OEE	72%	91%	+19 pp
Defect rate (PPM)	4,000	400	−90%
Changeover time	6 h	15 min	−96%
Energy per body (kWh)	42	31	−26%

The quality, energy, and OEE improvements should be interpreted as system-level outcomes rather than as the effect of a single technology. Better fixturing, process monitoring, robot motion, preventive maintenance, operator routines, and production control all contribute. The increase from 72% to 91% OEE is particularly significant

because it combines availability, performance, and quality; the component contributions must therefore be decomposed multiplicatively rather than by adding their percentage-point changes directly.

Economic Result

Illustrative investment: €18.5 million. Illustrative net annual benefit: €6.2 million from additional contribution margin, energy savings, and avoided quality cost. Simple payback = $€18.5\text{M} / €6.2\text{M} = 2.98$ years. With a 10-year horizon, constant annual net benefit, no residual value, and no tax effects, the project IRR is approximately 31%; this result is highly sensitive to realised volume and contribution margin.

Lessons Learned

- Vision-guided process verification justified its cost through earlier detection of positioning and weld-quality deviations.
- The 15-minute changeover was achieved only after systematic SMED analysis; the initial post-installation value was approximately 45 minutes.
- SafeOperation zones and associated safety validation required several engineering iterations, extending the schedule by approximately six weeks in the illustrative case.
- Operator training was the biggest risk: 22 operators required eight weeks of structured retraining.

Figure 13.2 shows the functional architecture behind these values: fourteen stations, one hundred twenty robots, safety-monitored workspaces, and a central MES that manages recipes and genealogy for three body variants.

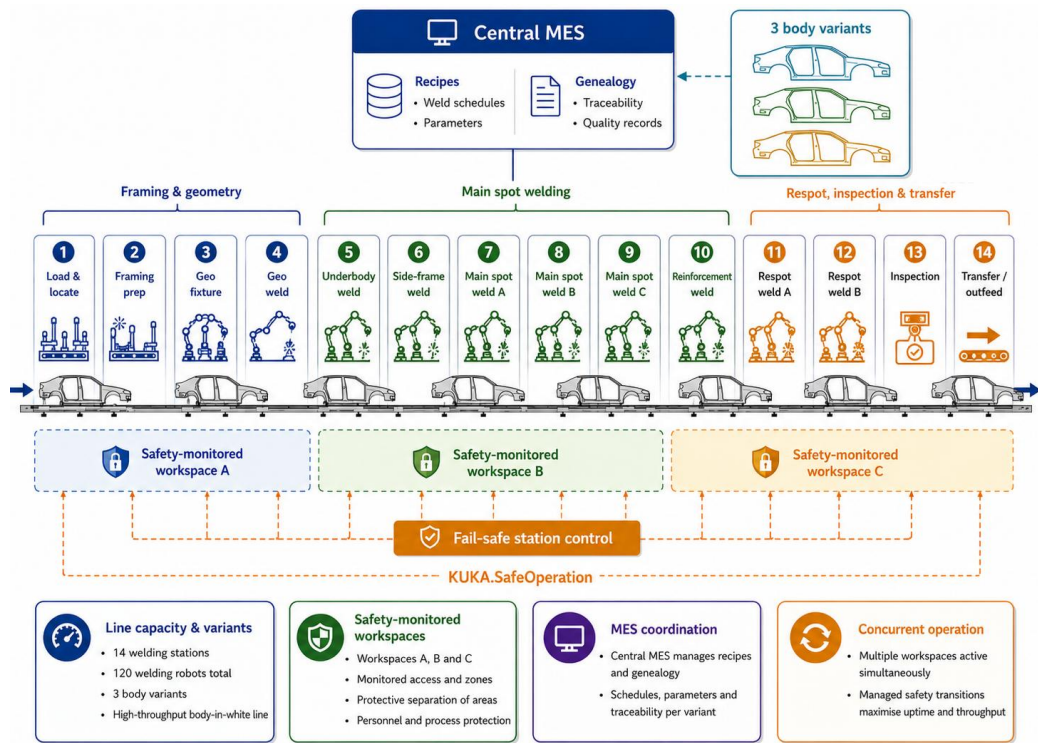


Figure 13.2. BIW spot-welding line schematic: 14 stations with 120 robots. A central MES manages recipes for three variants; KUKA.SafeOperation and fail-safe station control supervise safety-related workspaces and transitions.

INDUSTRIAL PRACTICE NOTE — Case 1 — BIW Control Architecture Deep Dive

The illustrative architecture uses PROFINET IRT for time-critical cyclic communication and PROFINET RT for less time-critical diagnostics and supervisory data. Fourteen fail-safe station PLCs coordinate local sequences, with PROFIsafe carrying safety-related signals. Network loading must be calculated from frame size, update time, topology, synchronisation, and protocol overhead; multiplying a count of variables by cycle time does not produce a valid bandwidth assessment. Standard MRP permits recovery times up to 200 ms and is therefore not a bumpless redundancy mechanism for isochronous traffic. Where continuity of IRT communication is required, an architecture such as MRPD or controller/system redundancy must be engineered and validated. Safety functions must transition to a defined safe state if communication assumptions are violated.

WORKED EXAMPLE — Decomposing the BIW OEE Improvement

The BIW refurbishment improved OEE from 72% to 91%. Decompose into Availability (A), Performance (P), and Quality (Q).

Before: A=82%, P=92%, Q=95.5%. $OEE = 0.82 \times 0.92 \times 0.955 = 72.1\%$.

After: A=95%, P=97%, Q=98.8%. $OEE = 0.95 \times 0.97 \times 0.988 = 91.0\%$.

Factor contributions: A Shapley decomposition attributes approximately +11.9 percentage points to availability, +4.3 points to performance, and +2.8 points to quality. Availability is the largest contributor, and the three contributions sum to the 19.0-point OEE increase.

Note: this case is illustrative; figures and timings are typical industry ranges, not data from a specific named site.

13.3 Case 2 — Electronics: AI-Based PCB Inspection

A composite electronics case considers the replacement of classical automated optical inspection (AOI) with deep-learning-assisted inspection on a line producing 36,000 PCBs per day. The baseline system detects 94% of known defects and generates a 2% false-positive rate. Because detection rate is not the same as defect prevalence, economic calculations must use the number of genuinely defective boards, not the total production volume.

System Design and Performance

Twenty-four existing SCARA robots are retained. A CNN ensemble is fine-tuned on 62,000 labelled images collected over six months and deployed on an industrial GPU server with approximately 12 ms inference time per image. Low-confidence or out-of-distribution cases, representing about 0.8% of volume in the illustrative dataset, are routed to human review. The rule-based AOI remains available as a comparison channel and fallback diagnostic aid, but it is not treated as an independent safety function.

Table 13.3 summarises the before-and-after metrics for the AI upgrade — throughput, defect-escape rate, false-positive rate, and operator workload — alongside the classical AOI baseline.

Table 13.3. *PCB inspection performance before and after AI upgrade*

Metric	Classical AOI	AI-Based AOI	Change
Detection rate	94%	99.4%	+5.4 pp
False positive rate	2.0%	0.3%	–85%
Estimated defect escapes/day (0.3% prevalence)	6.48	0.65	–90%
Throughput (PCBs/h)	1,500	1,500	0
Operator labour	3×2 ops	3×1 op	–50%

Economic Result and Lessons

Illustrative investment: €185,000. Under the most-likely assumptions used in the sensitivity analysis, net annual benefit is approximately €150,000 and simple payback is about 1.2 years. Model performance is reviewed continuously; retraining is triggered by validated drift, performance degradation, or introduction of a new product variant rather than by calendar frequency alone. Human review remains mandatory for low-confidence cases and for release decisions defined by the site quality plan.

INDUSTRIAL PRACTICE NOTE — Case 2 — AI Model Lifecycle Management in Production

The model requires a controlled lifecycle. Production images and metadata are sampled according to a documented data-retention and privacy policy. Drift monitoring combines operational KPIs, confidence calibration, class balance, and representative labelled audits; any statistical threshold is established from validation data and is site specific. Retraining is triggered by a sustained drop below the validated acceptance criterion, a confirmed distribution shift, or introduction of a new PCB variant. Candidate models are tested on frozen hold-out and challenge sets, run in shadow mode, and released through version-controlled rollback-capable deployment. The example assumes a six-month formal review, but actual retraining frequency is determined by evidence rather than calendar alone.

WORKED EXAMPLE — PCB AI Investment Sensitivity

Perform sensitivity analysis across defect prevalence, avoided escape cost, detection-rate gain, volume, labour saving, and recurring model-lifecycle cost. Investment: €185,000; 250 production days/year.

Pessimistic: Defect prevalence 0.2%, volume 30,000/day, detection gain 3 pp, avoided cost €20, labour saving €60,000/year, recurring cost €20,000/year. Avoided quality cost = $30,000 \times 0.002 \times 0.03 \times €20 \times 250 = €9,000/\text{year}$. Net annual benefit = €49,000; payback = 3.78 years.

Most likely: Defect prevalence 0.3%, volume 36,000/day, detection gain 5.4 pp, avoided cost €48, labour saving €96,000/year, recurring cost €16,000/year. Avoided quality cost = €69,984/year. Net annual benefit = €149,984; payback = 1.23 years.

Optimistic: Defect prevalence 0.5%, volume 40,000/day, detection gain 5.7 pp, avoided cost €65, labour saving €140,000/year, recurring cost €16,000/year. Avoided quality cost = €185,250/year. Net annual benefit = €309,250; payback = 0.60 years.

Key insight: Payback varies from approximately 0.6 to 3.8 years once the calculation distinguishes total production volume from defective-board prevalence. The most influential assumptions are defect prevalence, attributable escape cost, and sustainable labour saving.

Note: this case is illustrative; figures and timings are typical industry ranges, not data from a specific named site.

13.4 Case 3 — Food: Delta-Robot Packaging Line

A composite dairy-packaging installation uses eight high-speed delta robots, each rated for approximately 200 picks per minute under favourable product-spacing conditions, to place yoghurt pots into retail multipacks. The line is designed for hygienic operation and high-pressure wash-down. Ingress protection, hygienic cleanability, material compatibility, and cleaning validation are treated as separate requirements rather than being inferred from a single enclosure rating.

Illustrative steady-state throughput is 1,600 pots per minute and 120 trays per minute. The €2.1 million investment has an indicative payback of 2.1 years. The hygienic-design

premium increases initial capital expenditure but reduces corrosion risk, cleaning downtime, contamination traps, and premature replacement over the lifecycle.

Lessons

- Hygienic-design premiums pay back through extended life and reduced cleaning downtime.
- Product variability is the greatest source of unexpected stops; vision tuning iterates 12 weeks post-deployment.
- Seasonal campaigns introduce pattern changes requiring formal MES variant management.

Figure 13.3 shows eight hygienic delta robots working in parallel above a vision-tracked infeed conveyor and placing products into indexed thermoformed trays. The layout identifies wet-cleaning zones, protected electrical enclosures, drainage paths, and inspection points relevant to hygienic risk assessment.

INDUSTRIAL PRACTICE NOTE — Case 3 — Hygienic Design Standards for Food Robotics

Hygienic design requires several complementary assessments. EHEDG Guideline 62 Part 1 (2026) addresses fundamental hygienic-design requirements for robotic systems in open food-processing environments. 3-A Sanitary Standard 103-00 addresses industrial robot systems used where cleaning and sanitisation are required. An IP69 or IPX9K enclosure rating addresses resistance to ingress during high-pressure, high-temperature cleaning; it does not by itself demonstrate hygienic cleanability, material suitability, drainage, or freedom from microbial harborage. Materials, surface finish, seals, lubricants, fillet radii, drainage, and cleaning cycles must be selected through a hygienic risk assessment and validated against the actual product and cleaning chemistry. 316L stainless steel may be appropriate where chloride exposure justifies it, but material and elastomer selection is application specific.

Note: this case is illustrative; figures and timings are typical industry ranges, not data from a specific named site.

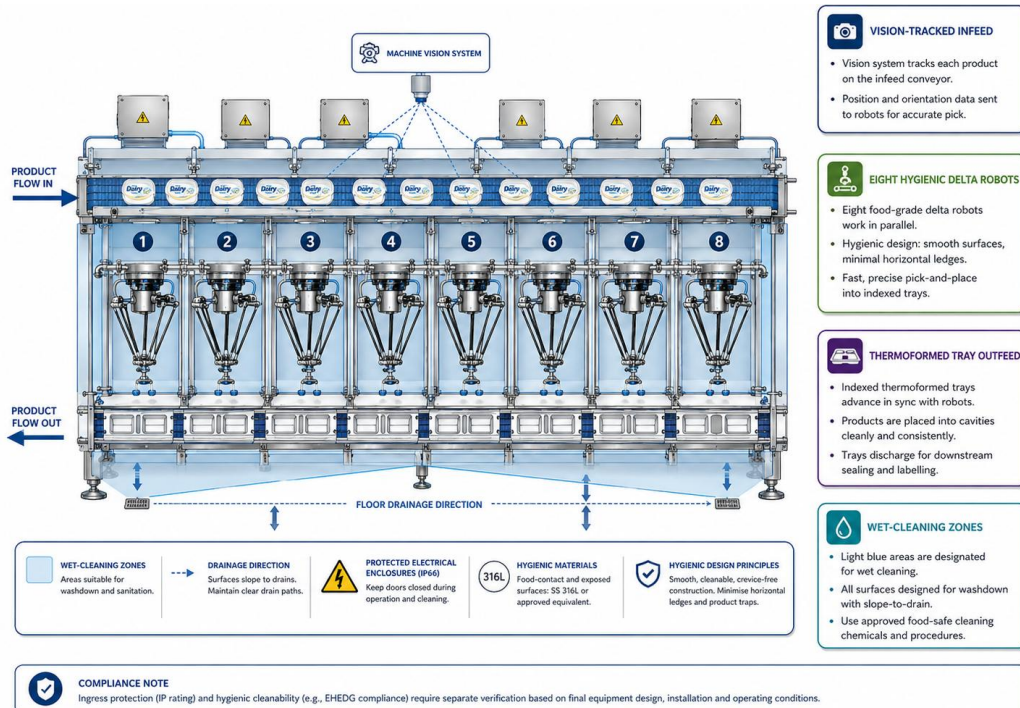


Figure 13.3. Delta-robot packaging cell layout: eight hygienic delta robots in parallel pick-and-place operation, with vision-tracked infeed, thermoformed-tray outfeed, and designated wet-cleaning zones. Ingress protection and hygienic cleanability require separate verification.

WORKED EXAMPLE — 15-Year TCO: Hygienic vs. Standard Design

Compare the undiscounted 15-year TCO of a hygienic design (€2.1M, 15-year life, €45k/year maintenance) with a standard design (€1.82M, 8-year life, €55k/year maintenance, replacement at year 8 for €1.95M). The example excludes downtime, residual value, discounting, production growth, and the cost of cleaning-validation failure.

Hygienic 15-year TCO: €2,100,000 + 15 × €45,000 = €2,775,000.

Standard 15-year TCO: First install: €1,820,000 + 8 × €55,000 = €2,260,000.

Replacement: €1,950,000 + 7 × €55,000 = €2,335,000. Total: €4,595,000.

Hygienic advantage: €4,595,000 – €2,775,000 = €1,820,000 lower undiscounted TCO. The lifecycle saving is approximately 6.5 times the €280,000 initial premium;

this is not an investment-return multiple because timing and discounting are omitted.

13.5 Case 4 — Pharmaceutical: Blister-Pack Inspection

Pharmaceutical production combines high unit volumes with strict requirements for product quality, data integrity, validation, and controlled change. The following composite case considers a European generics manufacturer adding cobot-assisted hyperspectral inspection to a blister-pack line. The project is governed by the site pharmaceutical quality system, EU GMP requirements, and a risk-based computerised-system lifecycle informed by GAMP 5 and Annex 11.

Context and System Design

A UR5e-class cobot equipped with hyperspectral imaging inspects blister packs for missing, broken, or incorrect tablets. A human operator handles exceptions and makes the final disposition decision for cases outside the validated acceptance envelope. Illustrative detection requirements are $\geq 99.95\%$ for missing tablets and $\geq 99.9\%$ for wrong-tablet events. Model or software changes undergo documented impact assessment, regression testing, approval, and, where required, requalification; not every update requires an external regulatory submission, but significant changes may require months of controlled validation work.

Performance and Economic Result

Illustrative throughput is 90 blisters per minute, matching the predecessor process while reducing shift-to-shift variation. Inspection-record review time falls from 48 hours to approximately 4 hours because evidence is generated in line; final batch release still depends on completion of all applicable quality-control and quality-assurance activities. Investment is €340,000, with an indicative payback of 3.8 years. Validation and qualification consume approximately 11 months and €180,000 in the teaching case, illustrating that compliance engineering can equal or exceed the equipment-integration effort.

Lessons

- Validation and qualification are managed as a parallel engineering workstream with their own requirements, traceability, testing, and approval gates.
- Model updates are deliberately infrequent because every change must preserve the validated state and data-integrity controls.
- Human-in-the-loop operation limits autonomous decision authority and provides a clearer, risk-based route to validation than fully autonomous batch disposition.

INDUSTRIAL PRACTICE NOTE — GAMP-5 Validation Cost Breakdown

Phase 1 — Specification and risk assessment (illustrative 2.5 months, 900 h): user requirements, functional/design specifications, intended use, data-integrity controls, and quality-risk assessment. Phase 2 — Qualification (4 months, 1,400 h): installation, operational, and performance qualification with traceable test evidence. Phase 3 — Software and model validation (2.5 months, 800 h): risk-based software classification, model-lifecycle documentation, challenge sets, access control, audit trails, backup/restore, and change-control evidence. Phase 4 — Release (2 months, 400 h): validation summary, SOPs, training, quality approval, and regulatory-impact assessment. Total illustrative effort: 3,500 h at €50/h = €175,000 plus €5,000 tools. GAMP 5 is a good-practice framework, not a product certification.

WORKED EXAMPLE — RaaS vs. Capital Purchase: 5-Year Cost Comparison

120 AMRs: RaaS at €1,100/AMR/month with 3% annual escalation versus a €12M purchase plus €300k/year maintenance. Discount rate: 8%. All values are teaching assumptions and service scope must be normalised before comparison.

RaaS over five years: nominal payments are €1.584M, €1.632M, €1.681M, €1.731M, and €1.783M. Nominal total = €8.410M; NPV at 8% = €6.685M.

Purchase over five years: €12.000M upfront plus five annual maintenance payments of €0.300M. Nominal total = €13.500M; NPV at 8% = €13.198M.

Interpretation: Five-year NPV favours RaaS by approximately €6.513M under the stated assumptions. Nominal cumulative costs cross during year 9; discounted cumulative costs cross around year 13. The decision rule is therefore horizon- and contract-dependent, not a universal eight-year threshold.

Note: this case is illustrative; figures and timings are typical industry ranges, not data from a specific named site.

13.6 Case 5 — Logistics: AMR Fleet in E-Commerce Fulfilment

E-commerce fulfilment combines high labour content, volatile demand, and strong seasonal peaks. Traditional picker-to-goods workflows can devote a large share of each cycle to travel, while training and ergonomic fatigue constrain rapid scaling. The following composite case evaluates a shelf-carrying goods-to-person AMR system rather than attributing the numerical values to a named vendor or site.

The illustrative deployment comprises 120 AMRs in a 45,000 m² Central European fulfilment centre serving clothing and small-goods orders. Mobile storage units are transported from a central inventory zone to fixed pick stations, reducing operator travel and decoupling picker productivity from walking distance. Fleet orchestration, WMS integration, charging strategy, radio coverage, and local robot safety are all treated as separate design workstreams.

Table 13.4 presents the illustrative before-and-after performance. Figure 13.4 visualises a central inventory zone served by 120 AMRs and a perimeter of stationary pick positions under a managed Wi-Fi 6 network. The wireless network supports task dispatch, telemetry, and fleet coordination; safety-rated obstacle detection and protective stopping remain local to each AMR and must tolerate temporary network loss.

Performance

Table 13.4 shows the operational effect of a goods-to-person architecture. Operator walking falls sharply because storage is brought to the workstation, while picking productivity increases from 85 to 260 units per hour in the illustrative case. These values should be treated as site-specific outcomes that depend on order profile, slotting, replenishment, station design, labour standards, and the definition of a completed pick.

Daily throughput increases by 74% while pickers per shift fall by 36%, and training time falls from three weeks to three days. The figures demonstrate the potential of workflow simplification, but they do not prove that AMRs alone caused the entire improvement; WMS logic, process redesign, slotting policy, operator interfaces, and ramp-up discipline contribute materially.

Table 13.4. *AMR fulfilment centre performance*

Metric	Before	After	Change
Picks/hr/picker	85	260	+206%
Daily throughput	67,200	117,000	+74%
Pickers per shift	280	180	–36%
Training time	3 weeks	3 days	–86%
Walking/shift	16 km	< 0.5 km	–97%

Lessons

The case highlights the interdependence of financing, digital integration, local safety, radio engineering, and human factors. The principal lessons are:

- Financial alignment via RaaS: a subscription model can align expenditure with uncertain demand and transfer part of the maintenance and obsolescence risk, but its value must be tested through discounted cash flow and contract-scope analysis.
- Ergonomics and operator acceptance: goods-to-person picking can reduce walking and fatigue, but stationary work introduces new risks associated with repetition, reach frequency, and workstation posture.
- Network infrastructure: a comprehensive RF site survey and monitored coverage are essential for task dispatch and fleet coordination. The AMRs must nevertheless maintain safe local behaviour during communication degradation.

- Integration complexity: WMS-to-fleet-manager interfaces, exception handling, inventory reconciliation, and cutover testing frequently dominate the schedule and should be planned as a major workstream.

Figure 13.4 illustrates a high-density goods-to-person fulfilment architecture in which AMRs move storage units between a central inventory zone and stationary pick positions. The layout reduces travel but introduces fleet-traffic, charging, replenishment, egress, and radio-coverage constraints that must be validated under peak demand.

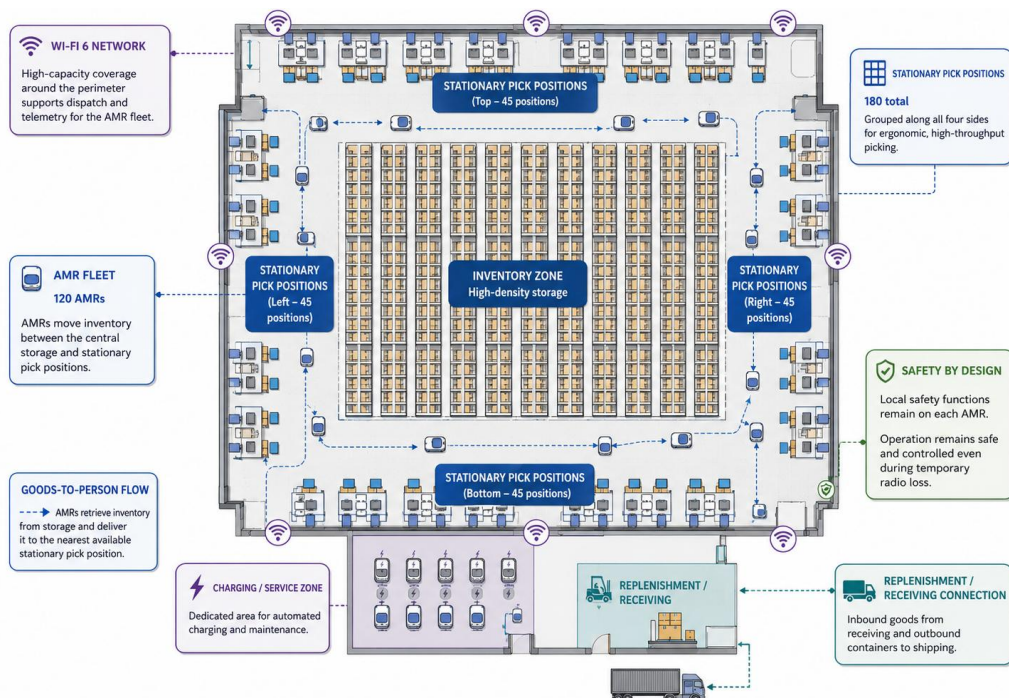


Figure 13.4. Illustrative AMR fulfilment-centre layout: 120 AMRs connect a central inventory zone to stationary pick positions. Wi-Fi 6 supports dispatch and telemetry, while local safety functions remain independent of the radio network.

A managed wireless network provides fleet-level task updates, telemetry, and coordination. It should be engineered for coverage, roaming, interference, capacity, cybersecurity, and graceful degradation rather than described as a safety channel. Navigation, obstacle detection, protective fields, and emergency stopping remain on-board so that loss of communication leads to a controlled operational response rather than loss of protective function.

INDUSTRIAL PRACTICE NOTE — Case 5 — WMS-to-Fleet-Manager Integration

In the illustrative project, WMS-to-fleet-manager integration consumes approximately 40% of the calendar. The work includes: (1) API contract and versioning; (2) reconciliation of order, task, location, inventory, and exception data models; (3) asynchronous acknowledgements, idempotency, retry limits, and dead-letter handling; (4) periodic full-state reconciliation; (5) identity, certificate, and network-security controls; and (6) parallel operation before cutover. Specific middleware, field counts, and latency limits depend on the selected platforms. The transferable lesson is to budget 30–40% of project time for digital integration when legacy WMS interfaces are involved, then validate that estimate during front-end engineering.

Note: this case is illustrative; figures and timings are typical industry ranges, not data from a specific named site.

For the economic comparison below, €1,100 per AMR per month, 3% annual escalation, a €12 million purchase alternative, and €300,000 annual maintenance are explicit teaching assumptions rather than market quotations. RaaS includes service, software, fleet management, and support; a capital-purchase alternative may differ in scope, residual value, staffing, upgrade entitlement, and capacity flexibility. A valid decision therefore compares like-for-like discounted cash flows and contractual risk allocation.

Figure 13.5 compares cumulative nominal and discounted costs. Under the stated assumptions, nominal cumulative RaaS cost exceeds the purchase alternative during year 9, while the discounted-cost crossover at an 8% discount rate occurs around year 13. The result changes materially with escalation, residual value, fleet utilisation, service scope, and the ability to scale the fleet.

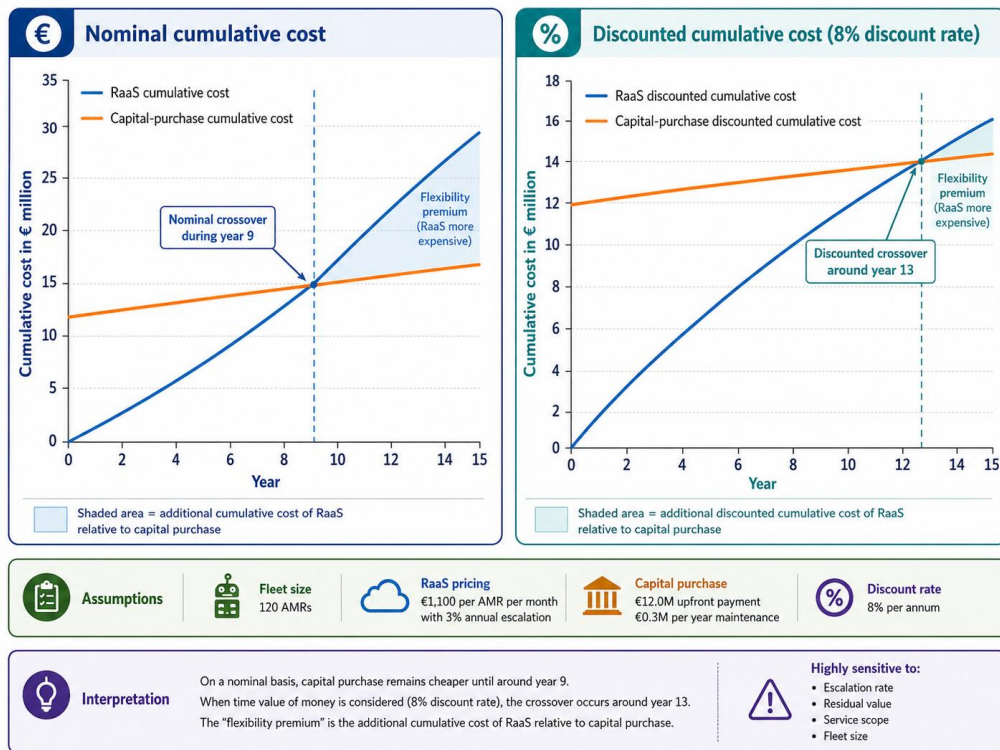


Figure 13.5. RaaS versus capital-purchase cost comparison under explicit teaching assumptions. Nominal cumulative costs cross during year 9; discounted costs at 8% cross at approximately year 13. Sensitivity to escalation, residual value, service scope, and fleet size must be reported.

13.7 Cross-Sector Synthesis: Seven Patterns That Transfer

Across the five composite cases, seven recurring observations appear with sufficient frequency to support structured discussion. They are not universal laws and should not be generalised beyond the evidence available; instead, they function as hypotheses to test during planning and post-project review.

The synthesis uses the illustrative schedule, cost, performance, and organisational data defined in the five teaching cases. A pattern is labelled recurrent when it appears strongly in at least three cases; patterns concentrated in one or two sectors are treated as context dependent.

Figure 13.6 presents the illustrative effort allocation as a Sankey diagram. Flow widths represent the assumed engineering hours in the teaching dataset, not measured industry

averages. Integration is prominent in all cases, whereas regulatory and validation effort is concentrated in pharmaceutical and automotive applications.

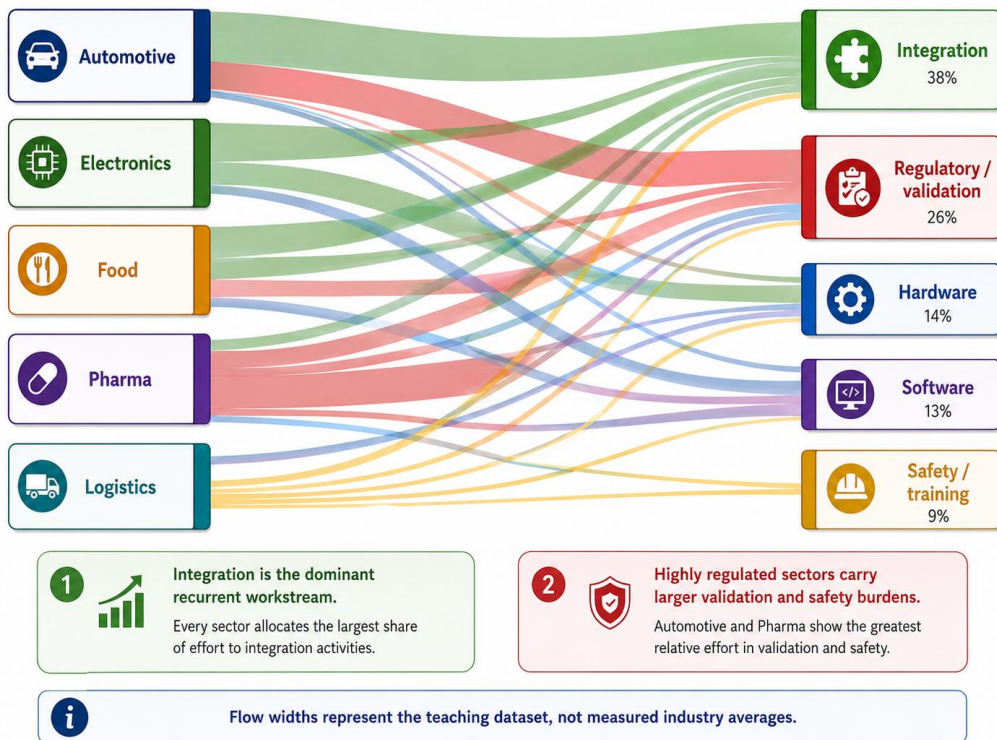


Figure 13.6. Illustrative Sankey diagram of project-effort allocation across the five composite cases. Flow widths represent the teaching dataset; integration is prominent across sectors, while regulatory, validation, and safety effort is concentrated in the more highly regulated cases.

Pattern 1 — Integration Dominates Effort

In the composite dataset, integration consumes approximately 35–60% of project effort. The transferable lesson is not the exact percentage, but the need to budget interfaces, testing, data reconciliation, and cutover as first-class work packages.

Pattern 2 — Data Collection Precedes AI

In both AI-enabled cases, data collection, labelling, validation, and lifecycle controls require at least as much effort as model development. AI schedules should therefore begin with the data plan, not the model architecture.

Pattern 3 — Human Factors and Ergonomics Shape Adoption

Ergonomic benefit, workload redesign, operator trust, and exception-handling quality materially influence adoption. Direct labour reduction alone rarely captures the full human-system business case.

Pattern 4 — Changeover Time Determines Variant Economy

The automotive case's 15-minute changeover enabled the three-variant business model. A 6-hour changeover would have forfeited the business case.

Pattern 5 — Operator Transition Is the Hidden Project

Each case requires structured operator training, role clarification, and transition management. The hidden project is often the redesign of work practices rather than the installation of hardware.

Pattern 6 — Safety and Validation Effort Is Commonly Underestimated

In the illustrative automotive, pharmaceutical, and logistics cases, safety, validation, or compliance work extends the schedule. The transferable lesson is to plan evidence generation and acceptance testing from concept design, not to add them after integration.

Pattern 7 — Continuous Improvement Requires Instrumentation

Systems without instrumentation, ownership, and review cadence may stagnate or regress after handover; systems with reliable KPIs and assigned engineering ownership can continue to improve.

Table 13.5 summarises the intensity of the seven patterns within the composite teaching cases. Figure 13.7 renders the same qualitative scores as overlaid radar polygons, distinguishing recurrent patterns from sector-dependent ones.

Table 13.5. Cross-case pattern intensity matrix

Pattern	Automotive	Electronics	Food	Pharma	Logistics
Integration effort	High (55%)	Med (40%)	Med (35%)	High (50%+)	High (60%)
Data collection	N/A	High (4 mo)	Medium	Medium	Low
Human factors / ergonomics	Low	Medium	High	High	High
Changeover impact	Critical	Low	Medium	Low	Low

Pattern	Automotive	Electronics	Food	Pharma	Logistics
Operator transition	High (8 wk)	Medium	Medium	Medium	High
Safety / validation effort	Yes (+6 wk)	No	No	Yes (+3 wk)	Yes (+4 wk)
CI instrumentation	Full SCADA	AI drift track	OEE dashboard	Batch KPI	Fleet analytics

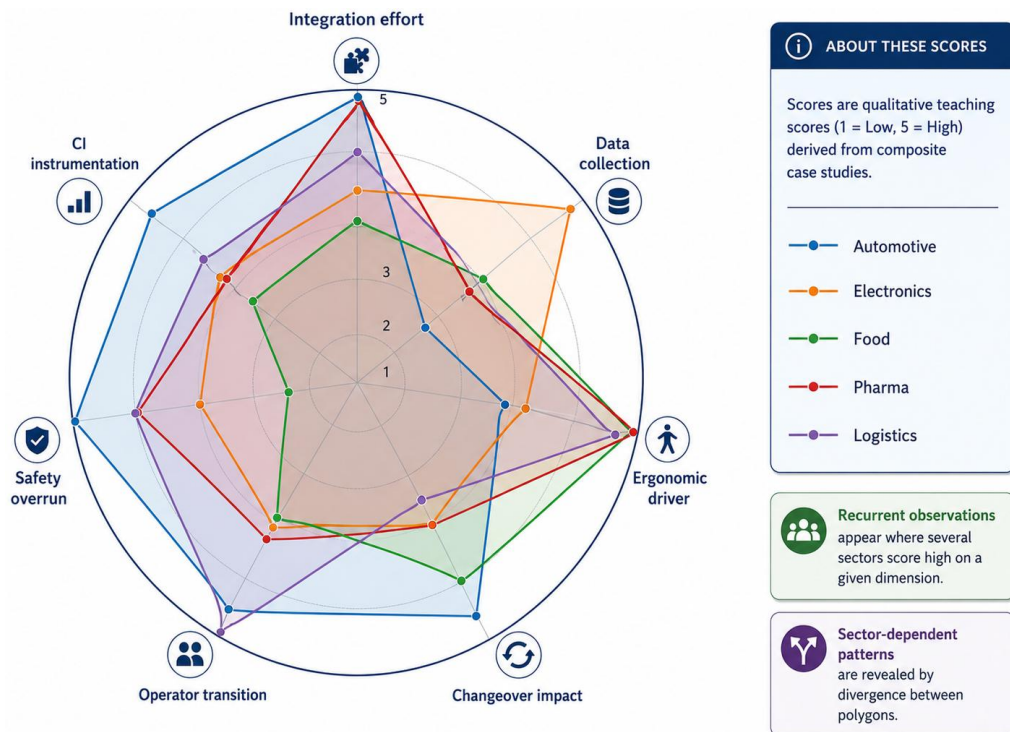


Figure 13.7. Radar chart of seven cross-case patterns across five composite sectors. High scores in several sectors indicate recurrent observations within the teaching dataset, not universal industry laws.

WORKED EXAMPLE — Designing a 90-Day Post-Commissioning Review

Design a structured 90-day review for the BIW case to detect regression before an 18-month slide becomes entrenched.

Weeks 1–4 (Intensive): Daily OEE by shift. Daily defect-rate SPC charts. Weekly actual vs. predicted cycle times. Root-cause analysis for any OEE drop >2 pp. Dedicated engineer full-time.

Weeks 5–8 (Stabilisation): Weekly OEE aggregation. Changeover-time audit vs. 15-min target. Operator competence assessment. SafeOperation zone-violation log review.

Weeks 9–12 (Handover readiness): Monthly reports established. Predictive-maintenance baseline calibrated. Lessons-learned workshop. As-built documentation verified.

Red flags: (1) OEE <88% for 3 shifts. (2) Defect >600 PPM for 2 days. (3) Changeover >20 min. (4) Unplanned stop >45 min not in risk register. Any red flag triggers 8D report within 24 hours.

INDUSTRIAL PRACTICE NOTE — Post-Project Lessons-Learned Template

Seven categories: (1) SCOPE ACCURACY — planned vs. actual scope. (2) TIMELINE FIDELITY — planned vs. actual durations per phase. (3) BUDGET VARIANCE — by category, flag any >+15%. (4) TECHNICAL SURPRISES — top 3 unanticipated issues and resolution. (5) HUMAN FACTORS — operator acceptance, training effectiveness, near-misses. (6) SUPPLIER PERFORMANCE — delivery, support, documentation quality. (7) RECOMMENDATIONS — 3 specific, actionable changes to standard methodology. Complete within 30 days of commissioning sign-off.

Note: this case is illustrative; figures and timings are typical industry ranges, not data from a specific named site.

Common Pitfalls

Pitfall 1: Benchmarking against the wrong baseline.

Always benchmark against best-in-class achievable, not merely against what was there before.

Pitfall 2: Reporting payback only on cash-flow improvements.

A rigorous analysis also accounts for ergonomic injury avoidance, strategic flexibility, and data option value.

Pitfall 3: Borrowing a layout without borrowing the constraints.

Layout is a response to constraints, not a universal pattern. What works in one context may fail in another.

Pitfall 4: Ignoring operator resistance as a technical problem.

Operator complaints are data points about operational reality. Teams that dismiss them miss genuine design flaws.

Pitfall 5: Harvesting first-year gains and disinvesting.

The continuous-improvement budget must be protected from operational pressures. Systems regress without it.

Pitfall 6: Assuming one sector's success factors apply universally.

Automotive's emphasis on cycle time is less relevant in pharmaceutical, where regulatory compliance dominates.

Pitfall 7: Underestimating WMS/MES integration effort.

In several composite cases, integration accounts for a large share of project effort. Treat WMS, MES, data models, interface testing, and cutover as major workstreams rather than incidental software tasks.

Pitfall 8: Neglecting production drift in AI-based systems.

AI performance can drift when products, lighting, materials, cameras, or operating practices change. Use validated performance and drift triggers, supported by periodic review; a fixed retraining interval is not a substitute for monitoring.

Pitfall 9: Confusing cobot deployment with plug-and-play.

Cobots require the same risk assessment rigour as traditional robots. Uncertified deployment is not 'collaborative.'

Pitfall 10: Treating the business case as a one-time exercise.

Organisations that revisit the business case at 6, 12, and 24 months learn faster and approve future projects more accurately.

KEY TAKEAWAYS

- A common six-element framework structures cross-sector case analysis.
- The five installations are composite teaching cases; numerical results are assumptions to be tested, not claims about named sites.
- Recurrent observations include integration effort, data lifecycle work, human factors, changeover economics, transition management, underestimated safety/validation effort, and the need for instrumentation.
- Economic analysis should use consistent cash-flow definitions, payback, NPV, IRR, and sensitivity analysis rather than a single headline metric.
- Detection rate must not be multiplied by total production volume without accounting for defect prevalence.
- OEE changes are multiplicative and require an appropriate decomposition method.
- Hygienic design, ingress protection, and cleaning validation are related but distinct requirements.
- GAMP 5 supports a risk-based validated lifecycle; it is not a product certification.
- RaaS-versus-purchase decisions depend on horizon, discounting, service scope, residual value, and capacity uncertainty.
- Sustained performance depends on post-commissioning ownership, monitoring, and controlled improvement.

Review Questions

Answer the following questions to consolidate your understanding of the chapter material. A thorough response should draw on concepts, formulas, and examples presented in the text.

1. Apply the six-element framework to a robotic installation you are familiar with. For each element, identify whether sufficient information exists to evaluate it.
2. The BIW case reports a 19-percentage-point OEE increase. Using the before-and-after availability, performance, and quality values, decompose the change with a method that respects the multiplicative OEE relationship.
3. Recalculate the PCB-inspection business case using defect prevalence, detection-rate improvement, avoided escape cost, labour saving, and recurring model-lifecycle cost. Which assumption has the greatest effect on payback?
4. Describe three ways the food-packaging case's hygienic constraints would change your engineering approach relative to a dry-goods application.

5. The pharmaceutical case allocates 50% of effort to validation. Would this ratio be higher, lower, or same in (a) medical-device assembly, (b) cosmetics packaging, (c) aerospace machining?
6. Compute the five-year nominal and discounted RaaS-versus-purchase comparison for the AMR case. Under the stated assumptions, in which year do nominal cumulative costs cross, and in which year do discounted costs cross?
7. Select two of the seven cross-case patterns and describe how you would incorporate them into a project-management template.
8. Identify one pattern that is industry-specific rather than universal. Justify why it does not transfer.
9. Propose three metrics that would detect post-commissioning regression within 90 days, allowing intervention before an 18-month slide.

References

- [1] International Federation of Robotics (2024) World Robotics 2024. VDMA Services GmbH, Frankfurt.
- [2] KUKA (current edition) KUKA.SafeOperation: Safety-Oriented Monitoring of Workspaces and Robot Motion. KUKA Deutschland GmbH, Augsburg.
- [3] KUKA (2025) Automotive Spot-Welding Applications with KR QUANTEC-Class Robots. KUKA Deutschland GmbH, Augsburg.
- [4] PROFIBUS & PROFINET International (2023) PROFINET Planning Guideline: Media Redundancy. PI, Karlsruhe.
- [5] PROFIBUS & PROFINET International (2026) PROFINET Technology: The Easy Way. PI, Karlsruhe.
- [6] ISPE (2022) GAMP 5: A Risk-Based Approach to Compliant GxP Computerized Systems, 2nd ed. International Society for Pharmaceutical Engineering.
- [7] European Commission (2011) EudraLex Volume 4, Annex 11: Computerised Systems. Brussels.
- [8] EHEDG (2026) Guideline 62 Part 1: Hygienic Design Criteria for Robotic Systems in Food Processing Environments — Fundamental Requirements for Robots.
- [9] 3-A Sanitary Standards, Inc. (current edition) 3-A Sanitary Standard 103-00: Industrial Robot Systems.
- [10] IEC 60529:1989+A1:1999+A2:2013, Degrees of protection provided by enclosures (IP Code). IEC, Geneva.
- [11] ISO 12100:2010, Safety of machinery — General principles for design — Risk assessment and risk reduction. ISO, Geneva.
- [12] Groover, M.P. (2020) Automation, Production Systems, and Computer-Integrated Manufacturing, 5th ed. Pearson.

- [13] Siciliano, B., Sciavicco, L., Villani, L., and Oriolo, G. (2019) *Robotics: Modelling, Planning and Control*, 2nd ed. Springer.
- [14] Koren, Y. (2010) *The Global Manufacturing Revolution: Product-Process-Business Integration and Reconfigurable Systems*. Wiley.
- [15] Wurman, P.R., D'Andrea, R., and Mountz, M. (2008) Coordinating hundreds of cooperative, autonomous vehicles in warehouses. *AI Magazine*, 29(1), pp. 9–20.

CHAPTER

14

Commissioning and Course Synthesis

From virtual commissioning to steady-state production, and the design-to-operation framework revisited

Commissioning is the engineering phase during which a robotic production system transitions from an assembled installation into an operational production system. The distinction is important: assembly verifies that components are present, connected, and mechanically complete; commissioning verifies that the system **does what it was designed to do**. It is one of the most risk-laden phases of the project because design decisions made over preceding months are tested simultaneously against real equipment, materials, interfaces, and operating conditions, while schedule and customer acceptance depend on the outcome.

This chapter traces the commissioning process from virtual pre-validation through Factory Acceptance Testing, installation, Site Acceptance Testing, and ramp-up to steady-state production. It then synthesises the full design-to-operation framework built over thirteen preceding chapters, showing how the methods and concepts interlock to produce robotic production systems that are safe, productive, maintainable, and adaptive. The closing pages look outward at the career of a robotics engineer and at the trajectories — collaborative, AI-enhanced, reconfigurable, networked — that will reshape the discipline over the coming decade.

LEARNING OBJECTIVES

1. Distinguish virtual commissioning, Factory Acceptance Test (FAT), and Site Acceptance Test (SAT) and describe the deliverables expected at each gate.
2. Structure a commissioning plan that sequences functional, performance, safety, and endurance tests to minimise rework and schedule risk.
3. Apply the four ramp-up phases (readiness, verification, stabilisation, optimisation) and recognise the engineering tasks typical of each.
4. Calculate virtual commissioning ROI across multiple projects and identify the programme-level break-even point.
5. Integrate the full design-to-operation framework — from requirements analysis through reconfigurable adaptive production — into a coherent engineering workflow.

6. Identify the principal trajectories shaping the next decade of robotic production systems and evaluate their implications for engineering practice.
7. Design a FAT/SAT test protocol with numbered test cases, pass/fail criteria, and non-conformance management procedures.
8. Articulate the professional responsibilities of a robotic-systems engineer across safety, quality, operational continuity, and lifelong learning.
9. Compare Software-in-the-Loop (SIL) and Hardware-in-the-Loop (HIL) virtual commissioning approaches and select the appropriate method for a given project complexity.
10. Evaluate the contractual, technical, and organisational dimensions of the integrator–customer relationship during commissioning.

14.1 The Commissioning Problem

A newly assembled robotic cell is a hypothesis: that the design will deliver the specified cycle time, quality, and availability when confronted with real material, real operators, and real variance. Commissioning is the phase that tests this hypothesis. Three classes of defect surface during commissioning:

- **Mechanical defects.** Misalignments, interference, wear patterns that emerge only under motion; tool drops; cable chafing; utility routing errors.
- **Controls defects.** I/O wiring errors, sequencing bugs, timing hazards, handshake failures at cell boundaries, safety-logic errors.
- **Integration defects.** The cell works, but when connected to upstream and downstream equipment, synchronisation fails. MES commands arrive in unexpected formats. Real parts differ from CAD. Real operators interact differently from assumptions.

Each class has distinctive symptoms, root-cause patterns, and remediation costs. In complex robotic projects, integration defects often form the largest category because interfaces between controllers, equipment, information systems, utilities, and operators are tested together for the first time. The exact distribution varies substantially with system architecture and project maturity. The core commissioning strategy is therefore

to move as much verification as practicable upstream — into virtual, simulated, and staged environments — so that interface problems surface earlier and at lower cost.

Table 14.1 provides indicative planning ranges for the three defect classes, including typical resolution effort and the cost differential between discovery at FAT and at SAT. These values are engineering estimates for budgeting and risk analysis, not universal industry benchmarks.

Table 14.1. *Indicative commissioning-defect classification, resolution effort, and cost comparison between FAT and SAT discovery*

Defect Class	Indicative Share	Avg. Resolution Time	Cost at FAT	Cost at SAT
Mechanical	15–25%	2–4 hours	€200–500	€800–2,000
Controls	15–25%	4–8 hours	€400–1,000	€1,500–4,000
Integration	50–65%	1–3 days	€600–1,500	€3,000–10,000

14.2 Virtual Commissioning

Virtual commissioning connects the real controllers — PLC, robot controller, safety controller — to a virtual plant model that simulates the mechanical and sensing environment. The controllers execute their actual code; the virtual plant responds as the real plant would; inputs and outputs flow across the controller/plant boundary as if the cell were built. Integration defects that would otherwise surface weeks later during physical commissioning become visible in the engineering office, days after the code is written.

14.2.1 Hardware-in-the-Loop vs Software-in-the-Loop

- **Software-in-the-Loop (SIL).** A virtual or emulated controller runs on a PC and exchanges signals with the virtual plant. SIL is comparatively fast to configure, requires little or no production hardware, and scales well to large models. Its fidelity depends on the controller emulator and communication model; by itself, it cannot verify the behaviour of physical I/O, real fieldbus hardware, or machine-level safety performance.
- **Hardware-in-the-Loop (HIL).** Physical controllers and, where required, representative I/O are connected through the intended communication architecture to a virtual plant running with deterministic or sufficiently fast real-time behaviour. HIL improves coverage of controller timing, fieldbus interaction,

and hardware-dependent faults, but it still does not replace validation of the completed safety functions and stopping performance on the physical installation.

Table 14.2 compares SIL and HIL across seven criteria: setup cost and time, timing fidelity, fieldbus coverage, safety-logic coverage, hardware requirements, and typical application. Figure 14.1 shows the underlying architecture: physical or virtual controllers exchange I/O and communication signals with a real-time plant model. The quality of the result depends on controller emulation, solver timing, model fidelity, and interface configuration; equivalence with the physical system must therefore be demonstrated rather than assumed.

Table 14.2. Indicative SIL versus HIL virtual-commissioning comparison

Criterion	SIL	HIL
Setup cost	€5,000–15,000 (indicative)	€25,000–85,000 (indicative)
Setup time	1–2 weeks (typical)	3–6 weeks (typical)
Timing fidelity	Model-dependent; non-real-time or real-time	Controller- and fieldbus-level timing; model-dependent
Fieldbus testing	Virtual or emulated interface	Physical interface and real fieldbus
Safety PLC testing	Safety-logic verification; limited physical coverage	Controller-level safety logic and timing; physical validation still required
Hardware required	PC and software licences	Physical controllers and representative I/O
Best suited for	Early design, simple cells, scalable regression testing	Complex multi-controller systems and hardware-dependent timing risks

Commercial environments for virtual commissioning include Siemens Tecnomatix Process Simulate with S7-PLCSIM Advanced, Dassault Systèmes DELMIA, and Rockwell Automation FactoryTalk DesignSuite Emulate3D. Industrial case studies report substantial reductions in on-site commissioning effort, particularly for complex multi-device systems, but the realised benefit depends on model scope, interface coverage, reuse, and project maturity. Reported percentage savings should therefore be treated as case-specific evidence rather than a guaranteed outcome.

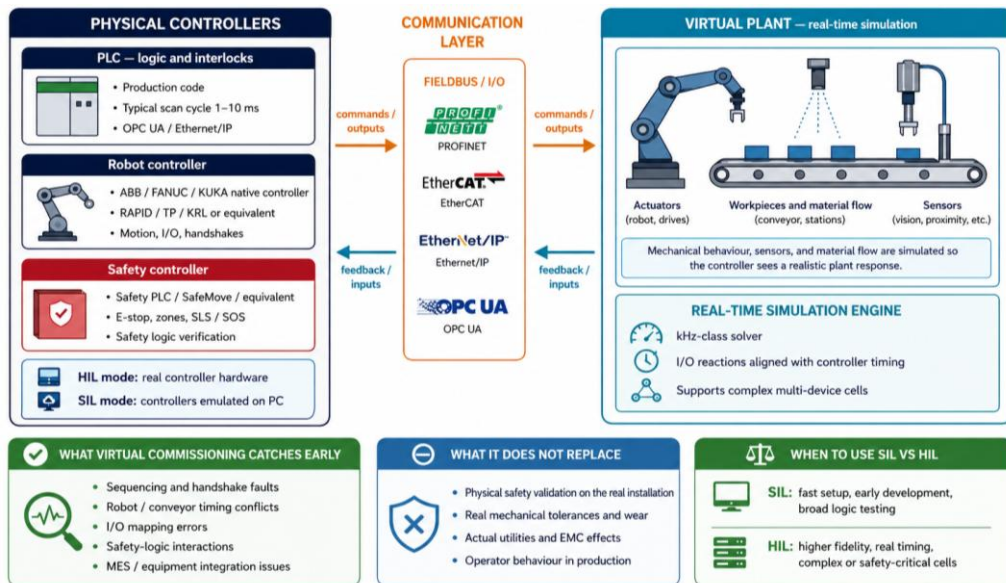


Figure 14.1. Virtual-commissioning architecture: physical or virtual PLC, robot, and safety controllers exchange I/O and fieldbus signals with a real-time virtual plant model for early verification of control logic, sequencing, interfaces, and timing.

WORKED EXAMPLE — Virtual Commissioning ROI Calculation

A six-robot cell integrator must decide whether to invest €85,000 in virtual commissioning (software licences, plant-model engineering, and a HIL test bench). Baseline without VC: FAT lasts 3 weeks, SAT lasts 5 weeks, and 12 integration defects are discovered during SAT, each requiring an average of 2 days at €1,800/day.

Baseline SAT exposure. Direct defect-resolution cost is $12 \times 2 \times €1,800 = €43,200$. Separately, 25 SAT days expose the project to site engineering and customer downtime of $25 \times (€1,800 + €4,000) = €145,000$. These figures describe different cost views and should not be added if the day-rate elements overlap.

VC savings. Assume that 5 of the 12 defects are found in the engineering office. Their office-resolution cost is $5 \times 2 \times €600 = €6,000$ instead of $5 \times 2 \times €1,800 = €18,000$ on site, saving €12,000. Shortening SAT from 5 to 3 weeks saves $10 \times (€1,800 + €4,000) = €58,000$. Shortening FAT from 3 to 2 weeks saves an assumed €6,000. Total estimated saving: €76,000.

Single-project ROI: €76,000 – €85,000 = –€9,000. Under these assumptions, the first project nearly breaks even but does not fully recover the initial investment.

Programme-level analysis (four similar projects with reuse): Total investment = €85,000 + €25,000 + €15,000 + €10,000 = €135,000. Total estimated saving = 4 × €76,000 = €304,000. Net programme benefit = €169,000. Break-even is reached during the second project under the stated assumptions. **Virtual commissioning should therefore be evaluated as a reusable engineering capability, not assumed to be profitable on every individual project.**

14.3 Factory Acceptance Test (FAT)

FAT is the formal verification, conducted at the integrator's facility before shipment, that the assembled cell meets its specification. The cell runs through a test protocol jointly prepared by the customer and integrator during design. Successful FAT is typically a contractual gate — commercially significant, because shipping without a clean FAT means the customer inherits the problems.

Table 14.3 lays out five core categories of FAT testing — functional, performance, safety, endurance, and diagnostic — each with its verification objective, representative test examples, and an indicative duration allowance. Integration and edge-case scenarios should be embedded within these categories or added as project-specific test groups where interface risk warrants them.

14.3.1 FAT Test Categories

The **Factory Acceptance Test (FAT)** serves as the definitive bridge between the design-build phase and the final on-site installation. It is a critical milestone where the customer and the integrator collaborate to verify that the system meets every predefined functional requirement, safety standard, and performance benchmark within the controlled environment of the builder's shop. Conducting these tests before the equipment leaves the floor is vital; it prevents the "exponential cost of change" that occurs once a system is bolted to the production floor, where logistics and downtime constraints make modifications significantly more difficult.

Table 14.3 summarises the five core test categories and indicative durations, providing a structured basis for planning and for agreeing customer witness points.

Table 14.3. *FAT test categories. Total FAT duration for a typical cell: 7–10 working days; complex/regulated: 3–4 weeks*

Test Category	Objective	Example Tests	Typical Duration
Functional	Every function operates correctly	Part presence, reject chute, alarms	2–3 days
Performance	Cycle time and throughput	Sustained 1-hour run at design rate	1–2 days
Safety	Safety functions meet specified PLr and stopping requirements	E-stop reaction, light curtain, safety-rated motion	2–3 days
Endurance	Stability over extended operation	8-hour continuous run with part variance	1–2 days
Diagnostic	Errors detected, reported, logged	Induced faults: missing part, jam, gripper fail	1 day

INDUSTRIAL PRACTICE NOTE — FAT Protocol Template

A production-grade FAT protocol is a controlled document with numbered test cases signed off by both integrator and customer. Structure: (1) HEADER — project, cell ID, date, attendees, and scope. (2) PRE-TEST CHECKLIST — safety documentation available, calibration certificates current, test material and spares ready, and documentation controlled. (3) TEST CASES — each has an ID (e.g. FAT-WELD-042), prerequisite state, steps, expected result, actual result, pass/fail decision, tester signature, and customer witness signature where required. (4) NON-CONFORMANCE REGISTER — each failure is logged with severity, containment, corrective action, owner, re-test date, and verification sign-off. (5) SIGN-OFF — integrator test manager, customer witness, and date. The number of test cases and document pages must be derived from system functions and risk: a compact cell may require dozens of cases, while a large line may require several hundred.

WORKED EXAMPLE — Designing a FAT Protocol for a Palletising Cell

Design the test protocol structure for a 2-robot palletising cell handling two product variants (A: 5 kg cartons, B: 12 kg bags). The cell includes a vision system, dual-pattern gripper, and safety fence with light curtain access.

Functional tests (15 test cases): FAT-F-001 to FAT-F-005: Vision system identifies product A and B correctly (5 tests, 10 parts each). FAT-F-006 to FAT-F-008: Gripper pattern selection for A and B. FAT-F-009 to FAT-F-012: Pallet pattern formation (4 patterns). FAT-F-013: Reject chute activation on unrecognised product. FAT-F-014: Pallet-complete signal to downstream wrapper. FAT-F-015: Alarm triggering on all fault conditions.

Performance tests (5 test cases): FAT-P-001: Product A sustained run, 60 min, target 8 cycles/min. FAT-P-002: Product B sustained run, 60 min, target 6 cycles/min. FAT-P-003: Mixed A/B run, 30 min. FAT-P-004: Changeover A→B, target <3 min. FAT-P-005: Changeover B→A, target <3 min.

Safety tests (8 test cases): FAT-S-001 to FAT-S-003: E-stop from three locations; verify stopping time and distance against the validated limits in the risk assessment. FAT-S-004: Light-curtain interruption initiates the specified safe state. FAT-S-005: Safety-PLC diagnostic and fault-reaction verification. FAT-S-006: Muting sequence during pallet exchange. FAT-S-007: Safety-rated speed monitoring at reduced speed. FAT-S-008: Reset after a protective stop requires deliberate manual action from outside the safeguarded space and does not initiate hazardous motion.

Endurance + diagnostic (4 test cases): FAT-E-001: 8-hour continuous run with random A/B mix. FAT-D-001: Induced gripper failure. FAT-D-002: Induced conveyor jam. FAT-D-003: Vision camera obstruction. Total: 32 test cases, estimated FAT duration: 8 working days.

Note: this case is illustrative; figures and timings are typical industry ranges, not data from a specific named site.

14.4 Installation and Site Acceptance Test (SAT)

After FAT, the cell is dismantled, shipped, and reassembled at the customer site. Installation typically requires two to five weeks depending on cell size and whether site preparation (flooring, utilities, network) was completed in parallel. Once reassembled, a second test round — Site Acceptance Test — verifies that the cell continues to meet specification in its production environment.

14.4.1 What Differs Between FAT and SAT

- Integration with upstream and downstream equipment (physical, electrical, informational).
- Site-specific utility quality (voltage stability, compressed-air moisture, ambient temperature, dust).
- Interaction with real plant operators and their established work practices.
- Connection to the customer's production MES/ERP/SCADA infrastructure.
- Safety integration with the wider plant safety system (site-wide E-stop, fire alarm).

SAT sign-off is commonly linked to contractual acceptance and release of a retained payment, but the amount and conditions are project-specific and must be defined explicitly in the contract. Because SAT exposes site-dependent interface risks, the integrator should also maintain a documented contingency based on utility quality, interface novelty, installation readiness, and customer responsibilities rather than rely on a universal percentage.

INDUSTRIAL PRACTICE NOTE — The SAT Discovery Log

Every SAT reveals site-specific issues. A DISCOVERY LOG captures them systematically: each entry has a unique ID, description, evidence, root-cause category (for example utilities, integration, operator interaction, documentation, fixture, or other), containment action, corrective action, owner, target date, status, and verification record. Review the log at a fixed cadence with the cross-functional team and distinguish issues that block safe operation, contractual acceptance, performance, or documentation closure. Entry counts and category distributions are project-specific; the key management indicators are severity, ageing, recurrence, and the number of unresolved items carried into ramp-up.

WORKED EXAMPLE — Estimating SAT Duration from Cell Complexity

Estimate the SAT duration for a 4-robot assembly cell with 3 upstream stations, MES integration, and a new safety fence design. Use the complexity-based estimation method.

Base SAT duration (single robot, no integration): 2 weeks.

Complexity multipliers: Robot count factor: 4 robots = $\times 1.6$ (each additional robot adds 0.2). Upstream integration factor: 3 stations = $\times 1.4$. MES integration: $\times 1.3$. New safety design (not proven in prior project): $\times 1.2$.

Estimated SAT: $2 \times 1.6 \times 1.4 \times 1.3 \times 1.2 = 6.99$ weeks, rounded to 7 weeks. Applying a project-specific contingency of 10–20% for novel interfaces gives approximately 7.7–8.4 weeks; plan 8 weeks unless stronger historical data are available.

Validation: This is an internal planning heuristic, not an industry standard. The multipliers should be recalibrated using completed-project data. A discovery rate materially above the project baseline during the first two weeks indicates deeper integration problems and should trigger technical and management review.

Note: this case is illustrative; figures and timings are typical industry ranges, not data from a specific named site.

14.5 Ramp-up to Steady-State Production

SAT acceptance is not the end of the commissioning journey; it is the handoff to ramp-up, the phase during which production volume builds from pilot runs to design throughput. Ramp-up typically takes 8–16 weeks for a complex cell and is characterised by four overlapping phases:

- 1. Readiness.** Operators complete training. Spare parts arrive. Maintenance procedures inspected. Scheduled test-production days.
- 2. Verification.** Daily production at 30–50% of design rate. Quality issues traced to root cause and corrected.
- 3. Stabilisation.** Production climbs to 70–90% of design rate. Residual issues surface and are resolved. OEE rises week-on-week.
- 4. Optimisation.** At design rate, marginal improvements through cycle-time optimisation, additional SMED on changeovers, and focused-improvement kaizen on top three downtime reasons.

Figure 14.2 plots a representative ramp-up trajectory alongside a problematic one: a well-prepared cell may reach approximately 20% of design output on day one, 50% by week four, 80% by week eight, and design rate by week twelve. Persistent deviation from

the planned curve often traces to incomplete operator preparation, unstable process parameters, insufficient maintenance readiness, or FAT/SAT residuals carried into go-live.

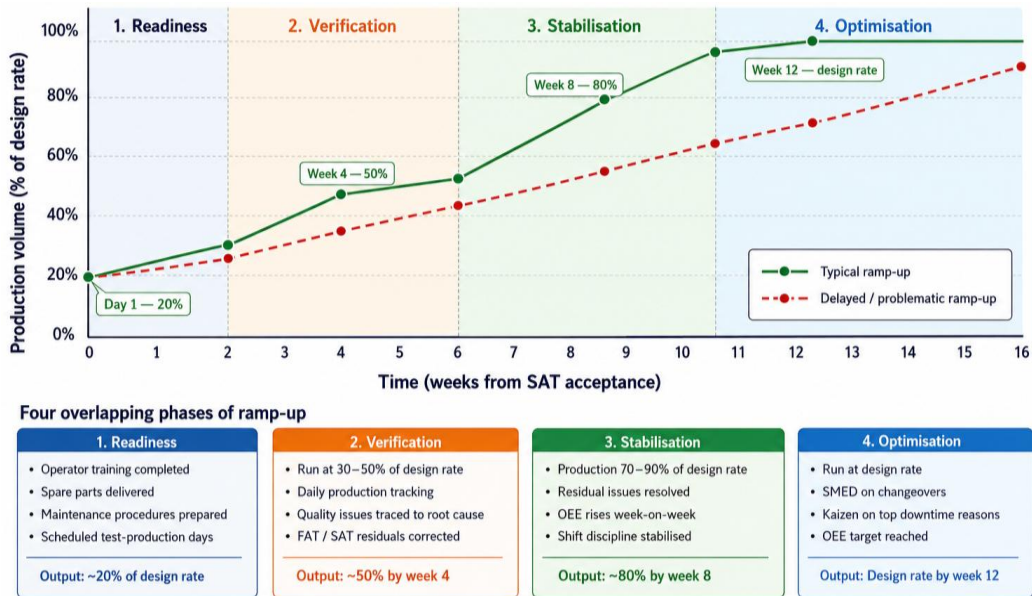


Figure 14.2. Ramp-up curve: production volume vs. time from SAT acceptance through steady-state.

WORKED EXAMPLE — Predicting the Ramp-up OEE Trajectory

A palletising cell has a commissioning target OEE of 85%. Predict the week-by-week OEE trajectory during the 12-week ramp-up and identify the engineering milestones that drive each step-change.

Weeks 1–2 (Readiness): OEE ≈45%. A representative decomposition is Availability 83%, Performance 60%, and Quality 90%: $0.83 \times 0.60 \times 0.90 = 44.8\%$. Frequent stops are expected for operator learning, parameter adjustment, and vision calibration.

Weeks 3–4 (Early Verification): OEE ≈60%. Availability 80%, Performance 80%, and Quality 94% give $0.80 \times 0.80 \times 0.94 = 60.2\%$. Milestone: the first full-shift run without a major unplanned stop.

Weeks 5–8 (Stabilisation): OEE $\approx 78\%$. Availability 88%, Performance 92%, and Quality 96% give $0.88 \times 0.92 \times 0.96 = 77.7\%$. Milestone: three consecutive production days above the stabilisation target with no recurring critical fault.

Weeks 9–12 (Optimisation): OEE reaches the 85% target. Availability 92%, Performance 94%, and Quality 98.3% give $0.92 \times 0.94 \times 0.983 = 85.0\%$. Milestone: formal handover from commissioning engineering to production ownership. A representative SMED action reduces changeover from 5 min to 2.5 min and improves availability.

INDUSTRIAL PRACTICE NOTE — Ramp-up Staffing Model

Ramp-up staffing normally tapers as stability increases. An illustrative model is: Weeks 1–4 — full commissioning team on site; Weeks 5–8 — reduced core team with specialist support on call; Weeks 9–12 — mentoring and controlled handover; post-week 12 — warranty support under the agreed service level. Staffing cost and duration must be derived from system complexity, risk, and contractual scope. Removing engineering support before stabilisation is demonstrated increases the risk of OEE regression and recurring faults during early production.

Note: this case is illustrative; figures and timings are typical industry ranges, not data from a specific named site.

14.6 Course Synthesis: The Design-to-Operation Framework

Thirteen chapters have introduced the concepts, methods, and technologies of robotic production systems. It is useful, at the close of the course, to step back and examine how they fit together — how the design decisions in Part I propagate through the integration decisions in Part II, the operational disciplines in Part III, and the adaptive and industrial lessons of Part IV.

14.6.1 From Customer Need to System Concept

The journey starts with a customer need stated in business terms. The engineering translation — Chapter 4's methodology for designing robotic cells — turns these into quantitative specifications: cycle time, throughput, reliability, flexibility, budget. Chapter

1 situates the cell in modern manufacturing (Industry 4.0/5.0); Chapter 2 defines its components and flow architecture; Chapter 3 establishes its layout topology.

14.6.2 From System Concept to Realisable Cell

Chapter 5 selects the robot and EoAT using multi-criteria methods (AHP, weighted scoring). Chapter 6 designs the control and communication architecture. Chapter 7 validates the design through simulation and digital twin — catching collisions, cycle-time shortfalls, and reachability issues before metal is cut.

14.6.3 From Realisable Cell to Operating System

The operating cell demands safety engineering (Chapter 8), continuous operational monitoring (Chapter 9), and reliability-focused maintenance strategy (Chapter 10). These three disciplines together determine whether the cell achieves merely nominal performance or world-class performance, and whether it performs sustainably for a decade or degrades within eighteen months.

14.6.4 From Operating System to Adaptive Production

The final module examines how individual cells combine into adaptive, networked production systems. Collaborative robots and AI (Chapter 11) expand what tasks can be automated; reconfigurable systems and mobile robots (Chapter 12) enable responsive production; case studies (Chapter 13) ground the principles in practice; commissioning (this chapter) closes the loop between virtual design and living operation.

Figure 14.3 assembles the fourteen chapters into a single workflow from customer need on the left to steady-state adaptive production on the right, organised into four modules. The dashed feedback arrows mark the knowledge transfers that make each subsequent project sharper: commissioning lessons update the design methodology, operational OEE data refines robot-selection criteria, and maintenance experience feeds back into safety engineering.

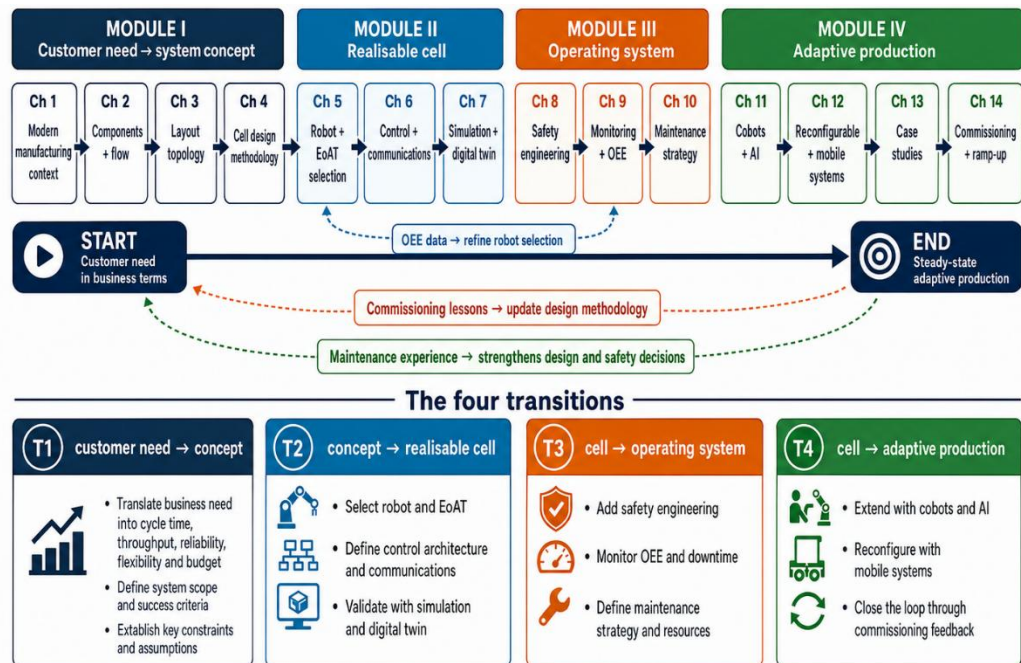


Figure 14.3. The complete design-to-operation framework: 14 chapters arranged as a workflow from customer need (left) to steady-state adaptive production (right).

WORKED EXAMPLE — Mapping a Real Project to the 14-Chapter Framework

A food manufacturer commissions a new packaging line for snack bars. Map each project phase to the relevant textbook chapter and identify the key deliverable from each.

Chapters 1–3: Context analysis (Chapter 1: market demand, Industry 4.0 readiness assessment). System architecture (Chapter 2: 3 delta robots, conveyor, vision, wrapper). Layout design (Chapter 3: U-shaped cell, 4.5 × 8 m, operator access from concave side). Deliverable: System Requirements Specification.

Chapters 4–7: Cell design methodology (Chapter 4: 12-step design process, cycle time 4.2 s). Robot selection (Chapter 5: AHP scores three delta robots; Omron Quattro selected). Control architecture (Chapter 6: S7-1500F + PROFINET + ProfiSafe). Simulation (Chapter 7: Visual Components model validates 14 packs/min). Deliverable: Preliminary Design Review package.

Chapters 8–10: Safety (Chapter 8: ISO 12100 risk assessment, light curtain + muting for pallet exchange). OEE framework (Chapter 9: target 82%, tracking via SCADA). Maintenance plan (Chapter 10: preventive schedule, spare-parts list, first CBM sensor on main drive). Deliverable: Safety file + O&M manual.

Chapters 11–14: AI vision for label inspection (Chapter 11). Future AMR integration for raw-material delivery (Chapter 12). Lessons from dairy case study (Chapter 13: hygienic design, seasonal variant management). Commissioning (Chapter 14: FAT 8 days, SAT 5 weeks, ramp-up 10 weeks). Deliverable: Commissioning Report + Lessons Learned.

14.7 Trajectories for the Next Decade

An engineer graduating in 2026 may spend the next forty years in a discipline whose frontier continues to move rapidly. Five trajectories are likely to influence robotic production systems strongly between now and 2035, although their pace and industrial impact will vary by sector, regulation, economics, and technical maturity:

- 1. Collaboration becomes more pervasive.** The distinction between cobots and traditional industrial robots is likely to become less rigid as torque sensing, lower-mass mechanisms, safety-rated motion, and application-specific risk reduction become more widely available. Fenced cells will remain necessary for many high-speed, high-payload, or process-hazard applications.
- 2. AI moves from assistant towards bounded operational autonomy.** Deep learning is established in many inspection applications; reinforcement learning and foundation-model-assisted programming are developing for grasp planning, adaptation, and integration support. Wider deployment will depend on data quality, verification, cybersecurity, functional safety, and human oversight.
- 3. Mobility becomes a mainstream material-flow option.** AMRs are likely to replace or complement fixed conveyors in applications where routing flexibility and product variety justify the additional coordination and safety complexity. High-volume, highly deterministic flows will continue to favour fixed transport in many cases.

4. Digital twins become persistent production infrastructure. For suitable assets, models created during engineering are likely to be maintained into operation for planning, diagnostics, training, and change validation. Adoption will depend on model governance, connectivity, cybersecurity, and the economic value of keeping the twin synchronised with the physical system.

5. Sustainability metrics become first-class engineering variables. Energy per unit, material efficiency, embodied carbon, and end-of-life recovery are likely to join OEE, quality, and cost in design trade-offs. EU sustainability-reporting requirements remain an important driver, but their scope and implementation have evolved and must be interpreted using the current legal framework rather than the original 2022 timetable alone.

14.7.1 What Does Not Change

Engineering tools become more sophisticated, but physical laws and disciplined verification remain unchanged. A 10 kg mass moving at 2 m/s still carries 20 J of kinetic energy. Biomechanical limits likewise constrain safe human–robot interaction. TPM (Total Productive Maintenance), SMED (Single-Minute Exchange of Die), Pareto analysis, and the PDCA (Plan–Do–Check–Act) cycle remain durable foundations of high OEE. Rigorous specification, robust design, iterative verification, and disciplined operation form the irreducible core of the profession, including in systems with increasing autonomy. The kinetic-energy relation used in the example is given below.

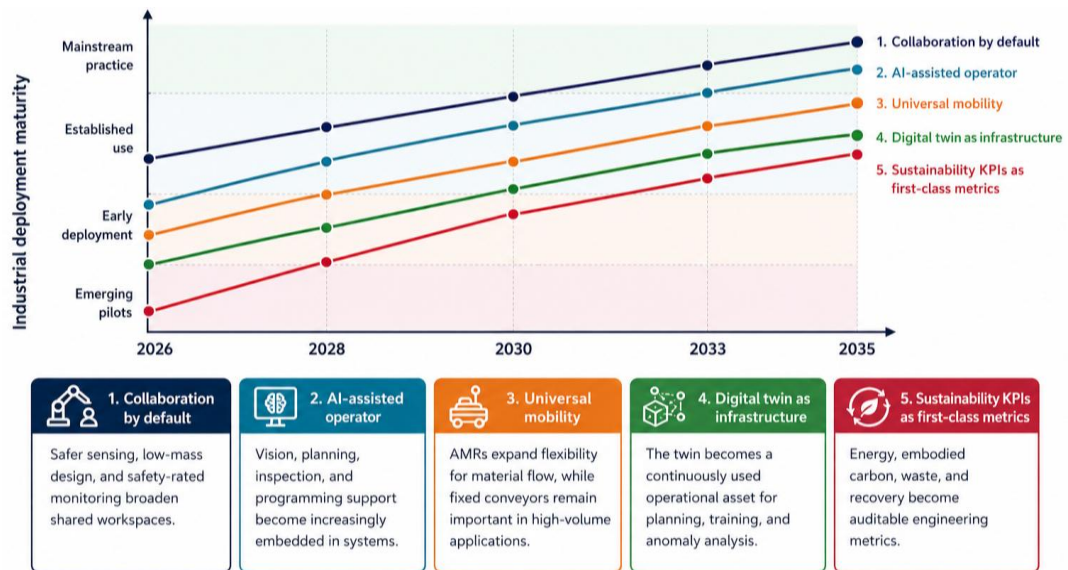
$$E_k = \frac{1}{2}mv^2$$

Building on this foundation, emerging technologies must be assessed through both maturity and adoption evidence. Modern engineers increasingly need fluency in data science, cybersecurity, and energy modelling alongside kinematics and process engineering. The transition from a static production line to a dynamic, self-optimising production system is not a departure from classical engineering; it is a more demanding expression of it, requiring higher-fidelity models, stronger data governance, and more explicit risk management.

Figure 14.4 presents a qualitative industrial-readiness and adoption outlook for five technology trajectories from 2025 to 2035. It is not a formal **Technology Readiness Level (TRL)** assessment: the curves compare broad technology families rather than single, well-

defined technologies validated against a common TRL method. Collaborative robotics and AI-enabled vision are already deployed in production in selected applications, while wider adoption of mobile, twin-based, and autonomous functions will depend on standardisation, integration cost, cybersecurity, and demonstrable operational value.

Sustainability KPIs may advance rapidly because regulation, customer requirements, and energy cost are turning environmental data into auditable engineering information. The Corporate Sustainability Reporting Directive has itself been amended, including by Directive (EU) 2026/470; engineers must therefore distinguish the enduring need for reliable sustainability data from the changing legal scope and timetable. By 2035, the most credible outcome is not a uniformly autonomous or carbon-neutral factory, but a more connected production environment in which selected technologies are integrated where their lifecycle value and risk case are justified.



Indicative outlook — actual adoption depends on sector, regulation, standards, and project economics.

Figure 14.4. Qualitative industrial-readiness and adoption outlook for five trajectories (2025–2035); not a formal TRL assessment.

14.8 Responsibilities of a Robotics Engineer

The engineers who design and operate robotic production systems bear responsibilities that extend beyond technical correctness:

- 1. Safety is not negotiable.** Pressure to ship on schedule or cut costs is never an acceptable justification for compromised safety.

2. **Honest reporting of uncertainty.** An engineer who presents numbers without uncertainty misleads decision-makers.
3. **Respect for operators.** A design that is technically correct but operationally intolerable is a failed design.
4. **Long-term thinking.** A cell runs 8–15 years. Optimising for commissioning often misaligns with optimising for operation.
5. **Commitment to lifelong learning.** The robotics discipline of 2040 will be substantially different from that of 2026. This course is a starting point.

Common Pitfalls

Pitfall 1: Shipping to site before FAT is clean.

Moving unresolved defects from the integrator's facility to the customer site usually increases remediation cost because access, logistics, production constraints, and customer downtime are added. A transparent FAT that records and closes non-conformances is better than an apparently clean FAT that conceals them.

Pitfall 2: Treating ramp-up as a maintenance responsibility.

The 8–16-week ramp-up is engineering work. Staffing it with operators who inherit an incompletely commissioned system causes chronic underperformance.

Pitfall 3: Deferring virtual commissioning to 'if we have time.'

Virtual commissioning can produce strong returns when interface risk is high and models are reused, but the business case is project- and programme-dependent. Skipping it solely to recover schedule may transfer work into later phases, where debugging is slower and more disruptive.

Pitfall 4: Contractual rigidity during SAT.

Site discoveries frequently include issues that were not fully anticipated. Collaborative problem-solving normally supports faster technical convergence than relying on adversarial contract interpretation as the primary response.

Pitfall 5: Harvesting lessons learned informally.

Organisations without a formal lessons-learned practice are more likely to repeat avoidable mistakes. Even a concise, structured debrief that is reviewed at the next project kickoff can provide disproportionate value.

Pitfall 6: Underestimating operator training duration.

Operators often need several weeks of structured training and supported production to become proficient on a new cell. Compressing training into a brief handover increases the risk that routine disturbances cannot be diagnosed or recovered safely.

Pitfall 7: Ignoring site utility quality during design.

Voltage sags, compressed-air moisture, and temperature swings cause intermittent failures that are extremely difficult to diagnose. Specify utility requirements in the contract and verify during site preparation.

Pitfall 8: Treating commissioning documentation as overhead.

FAT/SAT test records, discovery logs, and as-built drawings are the foundation for future maintenance and modifications. Incomplete documentation multiplies the cost of every future intervention.

Pitfall 9: Optimising cycle time at the expense of maintainability.

A cell with 0.2 s faster cycle time but no maintenance access to the gripper pneumatics will lose the time advantage in the first month of unplanned stops.

Pitfall 10: Assuming FAT results transfer perfectly to SAT.

Environmental factors (vibration from adjacent equipment, lighting, operator practices) can invalidate FAT results. SAT must re-verify all performance-critical tests.

KEY TAKEAWAYS

- Commissioning transforms an assembled installation into an operational production system. Integration and interface defects often form the largest and most schedule-sensitive defect category, but the distribution is project-specific.
- Virtual commissioning shifts control and integration verification upstream. Savings can be substantial on complex systems, but depend on model fidelity, test coverage, reuse, and project maturity.

- FAT validates the cell at the integrator; SAT validates it in production context. Both are contractual gates with formal documentation.
- Ramp-up to steady-state takes 8–16 weeks and demands continued engineering ownership; staffing with operations alone causes chronic underperformance.
- SIL is faster and more scalable for early and repeated software testing; HIL adds physical-controller and fieldbus coverage. Neither replaces validation of the completed safety functions on the physical installation.
- The design-to-operation framework integrates 14 chapters into a single workflow: customer need → specifications → cell → operating system → adaptive production.
- Five likely trajectories towards 2035 are more pervasive collaboration, bounded AI autonomy, mainstream mobile logistics, persistent digital twins, and sustainability as a first-class engineering variable.
- Engineering responsibilities — safety, honest reporting, respect for operators, long-term thinking, lifelong learning — outlast any specific technology.
- Virtual-commissioning ROI should be calculated from explicit assumptions. Reuse across similar projects can move break-even from marginal on the first project to strongly positive at programme level.
- Post-commissioning documentation (test records, discovery logs, as-built drawings, lessons learned) is the foundation for the system’s entire operational life.

Review Questions

Answer the following questions to consolidate your understanding of the chapter material. A thorough response should draw on concepts, formulas, and examples presented in the text.

1. Explain the three classes of defect that surface during commissioning. Using the indicative ranges in Table 14.1, discuss which class is likely to dominate frequency, schedule risk, and remediation cost in a complex cell.
2. Compare SIL and HIL virtual commissioning in terms of fidelity, cost, and defects caught. For a small single-robot cell, which would you select and why?
3. Outline a FAT protocol for a palletising cell: list at minimum three functional, two performance, three safety, and one endurance test with pass/fail criteria.

4. Identify five site-specific factors that can cause a FAT-accepted cell to fail SAT. For each, propose a mitigation that could be built into the FAT protocol.
5. Sketch a typical ramp-up curve from SAT acceptance to steady-state and mark the four phases with engineering activities expected in each.
6. The chapter claims ‘optimising for the commissioning metric often misaligns with optimising for the operational metric.’ Give two concrete examples.
7. Select one of the five 2035 trajectories and argue that it will advance faster than the others. Then argue the opposite position.
8. The course introduces approximately 50 engineering methods and frameworks. Identify the five you expect to use most often in the first two years of your career and justify.
9. A colleague proposes to skip virtual commissioning because the project is ‘running behind schedule.’ Formulate the argument you would make to the project manager.

References

- [1] Hoffmann, P., Schumann, R., Maksoud, T.M.A., and Premier, G.C. (2010) Virtual commissioning of manufacturing systems: A review and new approaches. Proc. 24th European Conf. on Modelling and Simulation, pp. 175–181.
- [2] Reinhart, G. and Wunsch, G. (2007) Economic application of virtual commissioning to mechatronic production systems. *Production Engineering*, 1(4), pp. 371–379.
- [3] Siemens AG (2022) SIMATIC PLCSIM Advanced — Virtual Commissioning Guide. Document ASE44101962-01, Nuremberg.
- [4] Dassault Systèmes (2022) DELMIA Robotics: Virtual and Physical Commissioning. 3DS Press, Vélizy.
- [5] ISPE (2022) Commissioning and Qualification — Baseline Guide, Vol. 5, 2nd ed. ISPE, Tampa FL.
- [6] Groover, M.P. (2020) *Automation, Production Systems, and Computer-Integrated Manufacturing*, 5th ed. Pearson. ISBN 978-0-13-460546-3.
- [7] International Federation of Robotics (2024) *World Robotics 2024*. VDMA, Frankfurt.
- [8] European Union (2022, amended 2026) Directive (EU) 2022/2464 on corporate sustainability reporting (CSRD), as amended inter alia by Directive (EU) 2026/470. Official Journal of the European Union.
- [9] NSPE (2019) *Code of Ethics for Engineers*. Publication 1102, Alexandria VA.
- [10] Mocan, B. (2025) *Design and Operation of Robotic Production Systems — Course Notes*. Technical University of Cluj-Napoca.
- [11] Siciliano, B., Sciacivco, L., Villani, L., and Oriolo, G. (2019) *Robotics: Modelling, Planning and Control*, 2nd ed. Springer. ISBN 978-1-84628-641-4.

- [12] ISO 12100:2010, Safety of machinery — General principles for design — Risk assessment and risk reduction. ISO, Geneva.
- [13] IEC 62443-3-3:2013, Industrial communication networks — System security requirements and security levels. IEC, Geneva.
- [14] Lee, J., Bagheri, B., and Kao, H.-A. (2015) A cyber-physical systems architecture for Industry 4.0-based manufacturing systems. *Manufacturing Letters*, 3, pp. 18–23.
- [15] Kagermann, H., Wahlster, W., and Helbig, J. (2013) Recommendations for implementing the strategic initiative INDUSTRIE 4.0. Acatech, Munich.
- [16] Rockwell Automation (2026) FactoryTalk DesignSuite Emulate3D — Digital Twin Software for Simulation and Virtual Commissioning. Milwaukee, WI.

A Closing Word

Thirteen chapters ago, this book opened with the assertion that robots generate value through their integration into well-designed production systems. Between that opening and this closing, we have examined concepts, methods, technologies, standards, and case studies — the working machinery of engineering.

A robotic production system is a machine that makes things. It is also a workplace for the operators who tend it, an asset that may remain in service for a decade or more, and a statement about what we, as designers, believe good production should be. The satisfaction of designing one well comes not only from cycle time or OEE, but from seeing something first conceived in requirements, models, and drawings operate safely, reliably, and productively for the colleagues who depend on its quality.

Go build them well.

— *B. M.*

APPENDICES

Design and Operation of Robotic Production Systems



Appendix A • *Mathematical Foundations*

Appendix B • *Standards Compendium*

Appendix C • *Glossary and Acronyms*

Appendix D • *Reference Data Tables*

Appendix E • *Project Templates and Checklists*

Appendix A — Mathematical Foundations

This appendix consolidates the mathematical tools used throughout the textbook. It is intended as a refresher and a reference, not as a first introduction. Readers who have not encountered these topics before should consult a dedicated text (Siciliano et al. for robotics kinematics, Law for queueing theory, Trivedi for reliability mathematics). The presentation prioritises computational recipes and engineering interpretation over full derivations.

A.1 Vectors, Matrices, and Homogeneous Transformations

Industrial robotics uses 3D vectors to represent positions and velocities, and homogeneous 4×4 matrices to represent rigid-body transformations. A homogeneous transformation T combines a 3×3 rotation R and a 3×1 translation p into a single matrix:

$$T = \begin{bmatrix} R & p \\ 0 & 1 \end{bmatrix} \quad (4 \times 4, \text{ bottom row} = [0 \ 0 \ 0 \ 1])$$

A rotation about the z-axis by angle θ has the matrix $R_z(\theta)$ with $\cos(\theta)$ and $\sin(\theta)$ components; similarly for $R_x(\theta)$ and $R_y(\theta)$. Any 3D rotation can be expressed as a product of three elementary rotations (Euler angles, roll-pitch-yaw) or as a single axis-angle pair or unit quaternion.

A.1.1 Composition of Transformations

If T_1 is the frame of link 1 relative to base, and T_2 is the frame of link 2 relative to link 1, then the frame of link 2 relative to base is $T = T_1 \cdot T_2$. Order matters: $T_1 \cdot T_2 \neq T_2 \cdot T_1$ in general. A rigid-body point p_{link} expressed in link coordinates is obtained in base coordinates by:

$$p_{\text{base}} = T \cdot [p_{\text{link}} ; 1] \quad (\text{stacked as } 4 \times 1)$$

A.1.2 Worked Example

WORKED EXAMPLE

Transforming a Grasp Pose

A point on the tool is $p_{\text{tool}} = (0.12, 0, 0)$ m. The tool frame is rotated 30° about the z-axis relative to the flange and translated by $p = (0, 0, 0.10)$ m.

Step 1 — build $T_{\text{flange\_tool}}$: $R = R_z(30^\circ)$, $p = (0, 0, 0.10)$.

Step 2 — transform the point: $p_{\text{flange}} = T_{\text{flange\_tool}} \cdot (0.12, 0, 0, 1)^T = (0.104, 0.060, 0.100, 1)^T$ m.

Sanity check: rotation preserves the 0.12 m radial distance in the xy-plane, while the translation sets $z = 0.10$ m. A point on the tool z-axis would be unaffected by a z-axis rotation.

A.2 Forward Kinematics with Denavit-Hartenberg

The Denavit-Hartenberg (DH) convention systematises the description of serial robot kinematics using four parameters per link: link length a_i , link twist α_i , link offset d_i , and joint angle θ_i . The transformation from link $i-1$ to link i is:

$$A_i = \text{Rot}_z(\theta_i) \cdot \text{Trans}_z(d_i) \cdot \text{Trans}_x(a_i) \cdot \text{Rot}_x(\alpha_i)$$

The forward kinematics of an n -DOF robot is the ordered product $T_{0_n} = A_1 \cdot A_2 \cdot \dots \cdot A_n$. Manufacturer documentation normally provides link dimensions, coordinate-system definitions, and calibrated kinematic data; it does not always publish a ready-to-use DH table. For engineering calculations, the selected standard or modified DH convention must therefore be stated explicitly and checked against the controller or CAD model.

A.2.1 Standard DH Table — Puma Example

Table A.1. Denavit-Hartenberg parameters for a Puma-class 6-DOF arm.

Joint i	a_i (m)	α_i (rad)	d_i (m)	θ_i (rad)
1	0	$\pi/2$	0	q_1
2	0.432	0	0	q_2
3	0.020	$-\pi/2$	0.150	q_3
4	0	$\pi/2$	0.432	q_4
5	0	$-\pi/2$	0	q_5
6	0	0	0.100	q_6

Once the A_i matrices are defined, forward kinematics is pure matrix multiplication. Symbolic tools (Robotics Toolbox for MATLAB, Peter Corke; the Python Robotics Toolbox) generate the closed-form expression automatically. For numerical computation at run-time, the cost is $6 \times (4 \times 4 \text{ matrix multiply}) \approx 450$ floating-point operations — trivial on modern controllers.

A.3 Inverse Kinematics

The inverse kinematics (IK) problem asks: given a desired pose T_{desired} of the end-effector, find joint angles q that achieve it. Unlike forward kinematics, IK is generally non-linear and may have multiple solutions, no solution, or infinitely many (singular configurations).

For many 6-DOF industrial arms with a spherical wrist, position and orientation can be separated and closed-form inverse-kinematics solutions are available. Other geometries require numerical methods. For 7-DOF redundant arms, such as the KUKA LBR iiwa or

Franka Research 3, a secondary criterion—elbow posture, distance from joint limits, collision clearance, or energy—selects one solution from the redundant set.

A.3.1 Jacobian-Based IK for Small Corrections

When the robot is close to a desired pose and small corrections are needed (e.g. visual-servoing, force-following), the Jacobian J relates joint velocities to end-effector velocities:

$$\mathbf{v} = J(\mathbf{q}) \cdot \dot{\mathbf{q}} \quad (\mathbf{v} \text{ is } 6 \times 1, \dot{\mathbf{q}} \text{ is } n \times 1, J \text{ is } 6 \times n)$$

For a square, full-rank Jacobian, $\dot{\mathbf{q}} = J^{-1}\mathbf{v}$. More generally, use the pseudo-inverse $\dot{\mathbf{q}} = J^+\mathbf{v}$. A numerical IK iteration computes the pose error $\mathbf{e}$, sets $\mathbf{v} = K \cdot \mathbf{e}$, and updates $\mathbf{q}_{\text{new}} = \mathbf{q} + J^+\mathbf{v} \cdot \Delta t$. Near singularities, the smallest singular value approaches zero and the undamped pseudo-inverse can amplify noise and demand excessive joint velocity. Damped least squares uses $J^+_{\lambda} = J^T(J \cdot J^T + \lambda^2 \cdot I)^{-1}$, trading some tracking accuracy for numerical robustness.

A.4 Robot Dynamics: Lagrangian Formulation

The dynamic equations of a rigid-body manipulator follow from the Euler-Lagrange equation:

$$M(\mathbf{q}) \cdot \ddot{\mathbf{q}} + C(\mathbf{q}, \dot{\mathbf{q}}) \cdot \dot{\mathbf{q}} + G(\mathbf{q}) = \boldsymbol{\tau}$$

where $M(\mathbf{q})$ is the $n \times n$ symmetric positive-definite mass matrix, $C(\mathbf{q}, \dot{\mathbf{q}}) \cdot \dot{\mathbf{q}}$ represents Coriolis and centrifugal terms, $G(\mathbf{q})$ is the gravity vector, and $\boldsymbol{\tau}$ is the vector of joint torques. The model supports joint-torque and actuator sizing, feed-forward control, and reflected-inertia calculations. Natural frequencies require an additional stiffness model and are obtained from a generalised eigenvalue problem involving both stiffness and mass.

A.4.1 Engineering Interpretation

- **$M(\mathbf{q})$:** the inertia matrix captures how fast the robot can accelerate given bounded torque. Joints close to the base carry more inertia; distal joints are lighter.
- **$C(\mathbf{q}, \dot{\mathbf{q}})$:** Coriolis and centrifugal effects depend on configuration and joint speed. **Commercial robot controllers account for the manipulator dynamics within their servo and feed-forward algorithms; they should not be discarded using a universal TCP-speed threshold.**
- **$G(\mathbf{q})$:** gravity compensation is essential for positioning; most controllers compute and inject gravity torques feed-forward.

For payload verification, evaluate joint torque, speed, acceleration, centre of gravity, and load inertia over the complete trajectory. Manufacturer load-identification and payload-checking tools support this process, but their names and functions differ by brand; the

final permissible envelope must be confirmed against the current robot and controller documentation.

A.5 Trajectory Planning

A trajectory is a time-parameterised sequence of poses $q(t)$ that the robot tracks. Three profile families dominate practice.

A.5.1 Polynomial (Cubic, Quintic)

Between two points in joint space, a cubic polynomial $q(t) = a_0 + a_1 \cdot t + a_2 \cdot t^2 + a_3 \cdot t^3$ can satisfy position and velocity boundary conditions. A quintic polynomial also satisfies acceleration boundary conditions and can provide smoother endpoint behaviour. Commercial controllers use proprietary blends of polynomial, trapezoidal, and jerk-limited interpolation; a teach-pendant move should not be assumed to implement one universal profile.

A.5.2 Trapezoidal Velocity Profile

The trapezoidal profile accelerates at constant rate $a_{\max}$ to reach cruise velocity $v_{\max}$, maintains it, and decelerates at $-a_{\max}$ to stop. For a joint displacement Δq :

If $\Delta q \leq v_{\max}^2/a_{\max}$: triangular profile, no cruise phase
Else: trapezoidal, cruise time = $\Delta q/v_{\max} - v_{\max}/a_{\max}$

The trapezoidal profile is time-optimal given hard velocity and acceleration limits, but produces instantaneous changes in acceleration (jerk = ∞) at profile transitions. These jerk spikes excite structural vibration in the robot and its payload.

A.5.3 S-Curve (Jerk-Limited) Profile

An S-curve profile bounds jerk by replacing instantaneous acceleration changes with finite ramps. Depending on move length and limits, the profile can contain jerk, constant-acceleration, constant-velocity, and symmetric deceleration phases. The benefit is lower excitation of structural modes and reduced load shock; the cycle-time penalty is application-dependent and must be evaluated together with settling time and process accuracy.

WORKED EXAMPLE

Trapezoidal vs. S-Curve Cycle-Time Comparison

Joint move of $\Delta q = 1.0$ rad, $v_{\max} = 2$ rad/s, $a_{\max} = 5$ rad/s², $j_{\max} = 50$ rad/s³ (S-curve).

Trapezoidal: Acceleration time $t_a = v_{\max}/a_{\max} = 0.4$ s. Distance during accel = $\frac{1}{2} \cdot a \cdot t^2 = 0.4$ rad. Distance during decel = 0.4 rad. Cruise = 0.2 rad, cruise time = 0.1 s. Total = 0.4 + 0.1 + 0.4 = 0.9 s.

S-curve: Jerk phase $t_j = a_{\max}/j_{\max} = 0.1$ s. Distance during accel phase = $v_{\max} \cdot (t_a + t_j) / 2 = 2 \cdot 0.5 / 2 = 0.5$ rad. Total accel = 0.5 s, decel = 0.5 s. Cruise = 0 rad (triangular). Total = 1.0 s.

Conclusion: in this idealised move the jerk-limited profile takes 1.0 s, approximately 11% longer than the trapezoidal profile. The practical cell-cycle effect may be smaller or larger because settling, path blending, process dwell, and controller-specific limits are not included.

A.6 Reliability Mathematics

Reliability engineering (Chapter 10) relies on the probability distribution of time-to-failure. Three distributions dominate practice.

A.6.1 Exponential Distribution (Constant Failure Rate)

If the failure rate λ is constant (the 'random' region of the bathtub curve), the probability of surviving beyond time t is:

$$R(t) = \exp(-\lambda t), \quad F(t) = 1 - \exp(-\lambda t)$$

For a non-repairable item following an exponential distribution, the mean time to failure is $MTTF = 1/\lambda$. For a repairable system, MTBF is the mean operating time between failures and equals $1/\lambda$ only under the constant-rate assumptions used here. The exponential distribution is memoryless; wear-out, ageing, and maintenance effects violate that assumption.

A.6.2 Weibull Distribution (Full Bathtub)

The Weibull distribution fits all three regions of the bathtub via its shape parameter β and scale parameter η :

$$F(t) = 1 - \exp(-(t/\eta)^\beta)$$

Shape $\beta < 1$ indicates a decreasing failure rate, $\beta = 1$ gives the exponential case, and $\beta > 1$ indicates an increasing failure rate. Numerical β ranges are component- and dataset-specific; maintenance decisions should be based on fitted confidence intervals, failure modes, and the consequences of failure rather than on a generic β threshold alone.

A.6.3 System Reliability

For n components in series (any single failure brings down the system):

$$R_{\text{sys}}(t) = R_1(t) \cdot R_2(t) \cdot \dots \cdot R_n(t)$$

For n identical components in parallel (all must fail):

$$R_{\text{sys}}(t) = 1 - (1 - R(t))^n$$

Mixed configurations are analysed by reduction to these two forms. For complex networks with common-cause failures, more advanced techniques (fault trees, Markov models) apply.

A.7 Queuing Theory for Production Systems

A.7.1 Little's Law

Little's Law is a remarkably general relationship, independent of arrival/service distribution, applying to any stable queuing system:

$$L = \lambda \cdot W$$

where L is the mean number in the system (WIP), λ is the mean arrival rate (throughput), and W is the mean time in the system (lead time). The law's power is its generality: it applies to a single station, a full line, or an entire factory. If any two are known, the third follows.

A.7.2 M/M/1 Queue

For a station with exponential inter-arrival times (rate λ) and exponential service times (rate μ), utilisation $\rho = \lambda/\mu$. The mean queue length and waiting time are:

$$L_q = \rho^2 / (1 - \rho), \quad W_q = \rho / (\mu - \lambda)$$

The $1/(1-\rho)$ term creates a steep utilisation effect as ρ approaches 1. The often-used design target $\rho \leq 0.85$ is only a rule of thumb: the appropriate margin depends on variability, service-level targets, buffer capacity, and the cost of idle capacity.

A.7.3 Kingman's Approximation

For general distributions (G/G/1), Kingman's formula provides a workable approximation:

$$W_q \approx [\rho / (1 - \rho)] \cdot [(C_a^2 + C_s^2) / 2] \cdot (1/\mu)$$

where C_a and C_s are the coefficients of variation of inter-arrival and service times. Kingman's approximation is useful for early buffer and waiting-time estimates, but it assumes a stable single-server queue and should be checked against discrete-event simulation when blocking, batching, failures, priorities, or multiple servers are important.

A.8 Probability and Statistics for Process Control

A.8.1 Process Capability Indices

The capability of a process to meet specification limits is quantified by Cp and Cpk:

$$C_p = (USL - LSL) / (6 \cdot \sigma)$$

$$C_{pk} = \min(USL - \mu, \mu - LSL) / (3 \cdot \sigma)$$

Cp measures potential capability relative to specification width; Cpk also accounts for process centring. Common organisational targets include 1.33 for capable routine processes and higher values for critical characteristics, but the acceptance threshold must be defined by the customer and control plan. Under a centred normal model, Cpk = 1.0 corresponds to about 2,700 nonconforming parts per million outside two-sided specification limits; the result changes with distribution shape and process drift.

A.8.2 Control Charts

A control chart tracks a quality characteristic over time with statistically derived control limits, commonly at $\pm 3\sigma$ for an individuals or Shewhart chart. Western Electric rules detect different non-random patterns, but combining several rules raises the overall false-alarm probability above the 0.27% associated with a single point outside $\pm 3\sigma$. The selected rule set should therefore match the process risk and sampling plan.

A.9 AHP Mathematics

Chapter 5 introduces the Analytic Hierarchy Process. Its mathematics reduces to an eigenvalue problem applied to pairwise comparison matrices.

A.9.1 Pairwise Comparison Matrix

For n criteria, build an n×n matrix A where a_{ij} is the relative importance of criterion i vs j on Saaty's 1–9 scale (1 = equal, 3 = moderately more, 5 = strongly more, 7 = very strongly, 9 = extremely). By construction $a_{ij} = 1/a_{ji}$ and $a_{ii} = 1$.

A.9.2 Weight Extraction — Eigenvector Method

The vector of weights w is the normalised principal eigenvector of A. For small n, a good approximation is the geometric mean of each row divided by the sum of those geometric means:

$$w_i = (\prod_j a_{ij})^{(1/n)} / \sum_k (\prod_j a_{kj})^{(1/n)}$$

Weights sum to 1 by construction.

A.9.3 Consistency Ratio

A perfectly consistent matrix would have principal eigenvalue $\lambda_{\max} = n$. Departures from consistency are quantified by the Consistency Index $CI = (\lambda_{\max} - n) / (n - 1)$. The

Consistency Ratio is CI divided by a reference RI (Random Index) tabulated by Saaty; for $n = 3$: RI = 0.58; $n = 4$: 0.90; $n = 5$: 1.12; $n = 6$: 1.24; $n = 7$: 1.32.

A common AHP rule of thumb is $CR \leq 0.10$; higher values indicate that the pairwise judgements should be reviewed. The threshold is guidance rather than a physical law, and sensitivity analysis remains necessary when rankings are close.

A.10 Control Theory Essentials

A.10.1 PID Controller

The PID controller outputs a control signal $u(t)$ based on tracking error $e(t) = \text{reference} - \text{measurement}$:

$$u(t) = K_p \cdot e(t) + K_i \cdot \int e(\tau) d\tau + K_d \cdot (de/dt)$$

Proportional (K_p) reacts to current error; Integral (K_i) eliminates steady-state error by accumulating; Derivative (K_d) anticipates trend and adds damping. In industrial robot control, PID appears at the joint level (inner loop, servo-drive parameters) and at the task level (outer loop, e.g. force control). Gain tuning methods include Ziegler-Nichols (classical), Cohen-Coon (for processes with delay), and modern auto-tune algorithms.

A.10.2 Stability and Performance Margins

For a stable loop, gain and phase margins quantify robustness to modelling error and delay. Values such as 6 dB gain margin and 30–45° phase margin are common starting targets, not universal acceptance criteria. Robot-control bandwidth depends on which loop is being discussed (current, velocity, position, or task level), the controller architecture, payload, sampling rate, and structural resonances; bandwidth figures must therefore state the measurement point and test conditions.

A.11 Numerical Methods for Engineering Calculations

A.11.1 Newton-Raphson for Root Finding

Used throughout robotics for inverse kinematics, trajectory time-scaling, and optimal control. Given a function $f(x)$ whose root is sought, Newton iteration is:

$$x_{(k+1)} = x_k - f(x_k) / f'(x_k)$$

Convergence is quadratic near the root when the derivative is non-zero and the initial estimate is sufficiently close. In multi-dimensional problems, solve $J \Delta x = f$ rather than explicitly forming J^{-1} . For robot IK, rank-revealing or damped least-squares methods are preferred near singularities, and warm-starting from the previous solution often improves convergence.

A.11.2 Discrete Integration

Sensor data is sampled discretely. Trapezoidal integration estimates the integral of a signal $s(t)$ from N samples $s_0, s_1, \dots, s_N$ at interval Δt :

$$I \approx \Delta t \cdot (s_0/2 + s_1 + s_2 + \dots + s_{(N-1)} + s_N/2)$$

For uniformly sampled data, the composite trapezoidal rule has global error $O(\Delta t^2)$ when the signal is sufficiently smooth. Higher sampling rates can reduce numerical-integration error, but sensor noise, timestamp jitter, missing samples, and aliasing may dominate in real systems.

A.11.3 Polynomial Interpolation

For sparse reference data (datasheet points), linear interpolation is usually sufficient. Given (x_1, y_1) and (x_2, y_2) , the value at x is:

$$y = y_1 + (y_2 - y_1) \cdot (x - x_1) / (x_2 - x_1)$$

Higher-order interpolation, including cubic splines, is widely used for smooth reference generation. Controller interpolation and servo-update periods are product-specific and may differ by orders of magnitude; use the current controller manual rather than a generic 125 μ s or 250 μ s assumption.

A.12 Linear Algebra Essentials

A.12.1 Eigenvalues and Principal Components

An eigenvector v of a matrix A satisfies $A \cdot v = \lambda \cdot v$. Manipulator natural frequencies are not the eigenvalues of the mass matrix alone; with a linearised stiffness matrix K they follow the generalised problem $K\phi = \omega^2 M\phi$. For the Jacobian, singular values—not simply eigenvalues of $J^T J$ —provide a numerically robust measure of proximity to rank loss. Principal Component Analysis uses an eigendecomposition or SVD of centred data to identify dominant variance directions.

A.12.2 Singular Value Decomposition (SVD)

Every matrix A can be decomposed as $A = U \cdot \Sigma \cdot V^T$, where U and V are orthogonal and Σ is diagonal with singular values $\sigma_i \geq 0$. The pseudo-inverse $A^+ = V \cdot \Sigma^+ \cdot U^T$ provides least-squares solutions to over- or under-determined systems. For a rectangular robot Jacobian, the singular values indicate manipulability: when the smallest σ approaches zero, the robot is near a singularity and small end-effector motions require unbounded joint velocities.

A.12.3 Condition Number

The condition number $\kappa = \sigma_{\max}/\sigma_{\min}$ measures sensitivity of a linear-system solution to perturbations. There is no universal threshold such as 10^4 : acceptable conditioning

depends on units, scaling, noise, and the required accuracy. Near robot singularities, regularisation or task scaling is used when $\sigma_{\min}$ becomes too small for the application.

A.13 Optimisation Formulations

A.13.1 Linear Programming for Resource Allocation

Capacity-planning problems are often formulated as linear programmes: minimise $c^T x$ subject to $Ax \leq b$ and $x \geq 0$. Modern solvers can handle very large sparse models, but practical performance depends strongly on formulation, sparsity, numerical scaling, and the computing environment; modelling and data quality are usually more limiting than the LP algorithm itself.

A.13.2 Mixed-Integer Programming for Line Balancing

Assigning tasks to stations under precedence and cycle-time constraints is a classical mixed-integer model. Tractability depends on the number and structure of tasks, stations, side constraints, symmetry, and required optimality gap. Large industrial instances may need decomposition, valid inequalities, warm starts, or heuristics rather than a fixed task-count cut-off.

A.13.3 Multi-Objective Optimisation (Pareto)

Engineering decisions often involve competing objectives. A Pareto-optimal solution cannot improve one objective without worsening at least one other objective. Weighted-sum, ϵ -constraint, and evolutionary methods such as NSGA-II are common. The number of alternatives presented to decision-makers should be chosen to expose meaningful trade-offs without obscuring them.

KEY POINTS

Appendix A — Essentials to Remember

- Homogeneous 4×4 matrices are the universal language of robot kinematics; always compose left-to-right from base to tool.
- DH parameters give closed-form forward kinematics. Inverse kinematics is harder and may have 2, 4, 8, or more solutions depending on robot geometry.
- Jerk-limited profiles reduce acceleration discontinuities and structural excitation; the cycle-time effect is trajectory- and controller-dependent.
- Weibull β identifies decreasing, constant, or increasing failure-rate behaviour, but maintenance decisions also require uncertainty, failure-mode, and consequence analysis.
- Little's Law ($L = \lambda \cdot W$) is general for stable systems. Queue growth depends on utilisation and variability; 0.85 is a planning heuristic, not a universal limit.
- AHP consistency ratio $CR \leq 0.10$ separates defensible pairwise judgements from noise; $CR > 0.10$ demands re-examination.
- Near kinematic singularities, damped-least-squares IK trades accuracy for stability. Always regularise; never compute J^{-1} directly.

Appendix B — Standards Compendium

This appendix catalogues the principal standards cited in the textbook and selected related documents used by robotics integrators. The edition status was checked for the July 2026 publication, but standards, amendments, national adoptions, and harmonisation listings continue to change. Always verify the current edition and legal status through ISO, IEC, the relevant national standards body, and—where EU presumption of conformity is claimed—the Official Journal of the European Union.

KEY POINTS

How to Use This Compendium

- Begin a project by identifying the applicable standards via risk assessment (ISO 12100) rather than by assumption.
- For EU machinery placed on the market before 20 January 2027, Directive 2006/42/EC remains applicable; Regulation (EU) 2023/1230 applies from 20 January 2027. Use the transition plan appropriate to the placing-on-the-market date.
- Always cite the edition and amendment status and re-check them before each design review, procurement decision, and conformity assessment.
- Many standards are 'horizontal' (apply generally) or 'vertical' (apply to specific machines). ISO 12100 is horizontal; ISO 10218 is vertical for industrial robots.
- A standard provides presumption of conformity in the EU only for the requirements and period covered by its current Official Journal citation; standards remain voluntary means of demonstrating compliance.

B.1 Safety Standards — The Core Framework

Table B.1. Core safety standards applicable to robotic systems.

Reference	Title	Relevance to Robotic Systems
ISO 12100:2010	Safety of machinery — General principles for design — Risk assessment and risk reduction	Type A standard establishing risk assessment and the three-step risk-reduction method: inherently safe design, safeguarding/complementary measures, and information for use.
ISO 10218-1:2025	Robotics — Safety requirements — Part 1: Industrial robots	Requirements for the industrial robot as a product, including inherent safety, safety functions, modes, limits, and information for integration and use.

Reference	Title	Relevance to Robotic Systems
ISO 10218-2:2025	Robotics — Safety requirements — Part 2: Industrial robot applications and robot cells	Integrator requirements across design, integration, commissioning, operation, maintenance, decommissioning, safeguarding, and validation of robot applications and cells.
ISO/TS 15066:2016	Robots and robotic devices — Collaborative robots	Supplementary guidance for collaborative robot applications, including biomechanical contact data. The collaborative application—not the robot alone—must be risk assessed and validated.
ISO 13849-1:2023	Safety of machinery — Safety-related parts of control systems — Part 1: General principles for design	Functional-safety framework for mechanical and electrical safety controls. Defines Performance Levels PLa through PLe and Categories B, 1–4.
ISO 13849-2:2012	Safety of machinery — Safety-related parts of control systems — Part 2: Validation	Validation methods for safety-related control systems: testing, analysis, documentation requirements.
IEC 62061:2021+A1:2024+A2:2026	Safety of machinery — Functional safety of safety-related control systems	Machinery-sector functional-safety standard for safety-related control systems. Uses SIL concepts and applies across electrical and non-electrical technologies within its scope.
IEC 61508 series	Functional safety of electrical/electronic/programmable electronic safety-related systems	Parent standard of IEC 62061. Provides SIL allocation methodology for wider industry sectors.
ISO 13855:2024	Safety of machinery — Positioning of safeguards with respect to the approach of the human body	Methodology and approach-speed parameters for positioning electro-sensitive protective equipment, interlocking guards, pressure-sensitive devices, and two-hand controls.
ISO 13857:2019	Safety of machinery — Safety distances to prevent hazard zones being reached by upper and lower limbs	Tabulated safety distances for fixed and movable guards. The 'reach-over' and 'reach-through' geometric analyses.
ISO 14118:2017	Safety of machinery — Prevention of unexpected start-up	Lockout/tagout (LOTO) requirements. Critical during maintenance and intervention.

Reference	Title	Relevance to Robotic Systems
ISO 14119:2024	Safety of machinery — Interlocking devices associated with guards — Principles for design and selection	Selection and application of guard-interlocking devices, including trapped-key systems and measures to minimise reasonably foreseeable defeat.

B.2 Robot Performance and Terminology Standards

Table B.2. Robot performance and terminology standards.

Reference	Title	Relevance
ISO 8373:2021	Robotics — Vocabulary	Standardised vocabulary for industrial and service robotics. Essential for precise specification writing.
ISO 9283:1998	Manipulating industrial robots — Performance criteria and related test methods	Definitions and measurement procedures for accuracy, repeatability, overshoot, settling time. The reference for datasheet numerics.
ISO 9409-1:2004	Manipulating industrial robots — Mechanical interfaces — Part 1: Plates	Standardised mounting flange for tool changers and end-effectors. Avoids proprietary-mount lock-in.
ISO 9946:1999	Manipulating industrial robots — Presentation of characteristics	Datasheet presentation conventions. Enables like-for-like comparison across manufacturers.
ISO 11593:2022	Robots for industrial environments — Automatic end-effector exchange systems — Vocabulary	Defines terminology for automatic end-effector exchange systems used as part of robot systems; use vendor data for load, locking, accuracy, utilities, and safety performance.
ISO 3691-4:2023	Industrial trucks — Safety requirements and verification — Part 4: Driverless industrial trucks and their systems	Safety requirements and verification for driverless industrial trucks and systems, including vehicles commonly described as AGVs or AMRs.

B.3 Communication and Integration Standards

Table B.3. *Communication and integration standards.*

Reference	Scope	Engineering Notes
IEC 61158 series	Industrial communication networks — Fieldbus specifications	Umbrella series defining multiple industrial communication protocol types. Check the relevant technology-specific parts and profiles rather than treating the series as one protocol.
IEC 61784-1/2	Industrial communication networks — Profiles	Profile definitions: which fieldbus protocol with which physical layer. CPF 3 = PROFIBUS, CPF 2 = CIP (DeviceNet, EtherNet/IP), CPF 9 = EtherCAT.
IEC 62541 series	OPC UA — Open Platform Communications Unified Architecture	Semantic-layer integration protocol. Base standard plus companion specifications for specific domains. Underpinning Industry 4.0 integration.
MQTT v5.0 (OASIS)	Message Queue Telemetry Transport	Lightweight publish-subscribe protocol for IoT. Broker-based; three QoS levels. Increasingly used in factory-floor telemetry and cloud upload.
IEEE 802.1 TSN family	Time-Sensitive Networking	A family of IEEE 802.1 mechanisms for time synchronisation, traffic scheduling, redundancy, and bounded latency. Determinism depends on the selected profiles and complete network configuration.
ISA-95 / IEC 62264	Enterprise-control system integration	Functional hierarchy (Levels 0–4) and object models for MES–ERP integration. B2MML is the XML schema.
ANSI/ISA-18.2 series	Management of alarm systems	Lifecycle framework for alarm philosophy, identification, rationalisation, design, implementation, operation, monitoring, maintenance, and audit.
ISA-88 / IEC 61512 series	Batch control	Models and terminology for batch control, including separation of procedural and physical models; widely used in pharmaceutical, food, and chemical production.

B.4 Electrical, EMC, and Installation Standards

Table B.4. *Electrical, EMC, and installation standards.*

Reference	Title	Notes
IEC 60204-1:2016+A1:2021	Safety of machinery — Electrical equipment of machines — Part 1: General requirements	Widely used for machine electrical equipment from the supply connection onwards. It is not automatically mandatory; legal or contractual applicability and any EU harmonisation status must be checked.
IEC 61000 series	Electromagnetic compatibility (EMC)	60+ parts. Key ones: 61000-6-2 (immunity for industrial environments), 61000-6-4 (emission for industrial environments).
IEC 62443 series	Security for industrial automation and control systems	Cybersecurity lifecycle, policies, system security requirements, component requirements, zones and conduits, and security levels for industrial automation and control systems.
IEC 60947 series	Low-voltage switchgear and controlgear	Contactors, motor starters, circuit breakers. Component-level requirements for the control panel.
IEC 60529 (current consolidated edition)	Degrees of protection provided by enclosures (IP Code)	Defines enclosure protection against access, solid objects, and water. Do not use “IP69K” as an IEC 60529 designation without checking the applicable standard; ISO 20653:2023 defines IP69K for road-vehicle electrical equipment.
IEC 61131-3:2013	Programmable controllers — Part 3: Programming languages	Defines LD, FBD, ST, and SFC; Instruction List is retained mainly for legacy systems and is deprecated in the third edition.
UL 508A	Industrial Control Panels	US industrial-control-panel standard commonly specified by customers and authorities having jurisdiction. Required certification or listing depends on the market, installation, and contractual route.

B.5 Ergonomics and Human Factors Standards

Table B.5. *Ergonomics and human factors standards.*

Reference	Scope	Application
EN 1005-3:2008	Recommended force limits for machinery operation	Psychophysical force limits for repetitive and occasional operator actions. Used in cobot design and operator-station ergonomics.
ISO 11228-1:2021	Manual handling — Lifting, lowering, carrying	NIOSH-derived lift-equation adaptation. Determines whether a manual handling task requires mechanisation or robotic assistance.
ISO 11226:2000	Ergonomics — Evaluation of static working postures	Provides recommendations for evaluating static postures using body angles, duration, and support conditions. RULA and REBA are separate observational assessment methods, not methods defined by ISO 11226.
ISO/TR 23482-2:2019	Robotics — Application of ISO 13482 — Part 2: Application guidelines	Guidance for applying ISO 13482 to personal-care robots. It is not a substitute for industrial-robot application standards such as ISO 10218.
ISO 9241 series	Ergonomics of human-system interaction	Multi-part ergonomics series covering human-centred design, dialogue principles, displays, input devices, and usability; relevant to HMI, SCADA, and operator-interface design.

B.6 Quality and Reliability Standards

Table B.6. *Quality and reliability standards.*

Reference	Title	Notes
ISO 9001:2015	Quality management systems — Requirements	General QMS. Prerequisite for many supplier-qualification audits.
IATF 16949:2016	Quality management — Automotive sector	Automotive-industry QMS extending ISO 9001. Almost universal in automotive-tier supply.
IEC 60300 series	Dependability management	Reliability, availability, maintainability, safety. 60300-3-1 application guide for dependability analysis techniques.

Reference	Title	Notes
SAE JA1011	Evaluation criteria for RCM processes	Reliability-Centered Maintenance methodology. Seven canonical questions per asset.
ISO 55000:2024	Asset management — Vocabulary, overview and principles	Foundational terminology and principles for realising value from assets across their lifecycle; used with ISO 55001 and related guidance where an asset-management system is required.
GAMP 5, 2nd edition (ISPE, 2022)	A Risk-Based Approach to Compliant GxP Computerized Systems	Industry good-practice guide for risk-based lifecycle management of GxP computerised systems. It is not an ISO standard, regulation, or product certification.

B.7 Regulatory Frameworks

Table B.7. Regulatory frameworks by region.

Reference	Region	Scope
Machinery Directive 2006/42/EC	EU	Applies to machinery placed on the EU market until 19 January 2027. Establishes essential health and safety requirements and conformity-assessment obligations.
Machinery Regulation (EU) 2023/1230	EU	Applies from 20 January 2027 and replaces Directive 2006/42/EC. Addresses digital instructions, software, cybersecurity-related safety, and machinery incorporating evolving or AI-based behaviour.
HG 1029/2008	Romania	Romanian transposition of Machinery Directive 2006/42/EC.
EN 60204-1 (current cited edition)	EU	May provide presumption of conformity for the essential requirements covered only when the relevant reference is currently cited in the Official Journal; check the applicable transition dates and restrictions.
OSHA 29 CFR 1910.212	USA	General requirements for machine guarding.
ANSI/A3 R15.06-2025	USA	Revised US industrial-robot safety standard organised in parts for industrial robots, robot applications/cells, and use of robot cells.
JIS B 8433 series	Japan	Japanese industrial-robot safety standards aligned in structure with ISO 10218; verify the current national edition and legal context.

Reference	Region	Scope
GB 11291 series	China	Chinese industrial-robot safety standards; verify current national editions, compulsory status, and market-access requirements for the target installation.
Radio Equipment Directive 2014/53/EU	EU	Applies to radio equipment incorporated in machinery, including Wi-Fi, cellular, and Bluetooth devices; assess interaction with machinery, EMC, cybersecurity, and product-specific legislation.
RoHS Directive 2011/65/EU	EU	Restriction of hazardous substances in electrical/electronic equipment.
WEEE Directive 2012/19/EU	EU	Waste electrical and electronic equipment; disposal and take-back obligations.

Appendix C — Glossary and Acronyms

This appendix provides expanded acronym definitions and a glossary of technical terms used throughout the textbook. Terms are cross-referenced where useful. The acronym list extends the one presented in the front matter; the glossary clarifies terms whose meaning is contested, overloaded, or domain-specific.

C.1 Acronyms

Table C.1. *Acronyms used throughout the textbook.*

Acronym	Expansion and note
AGV	Automated Guided Vehicle — driverless industrial vehicle operating under defined guidance, path, or traffic rules; capabilities vary by system
AHP	Analytic Hierarchy Process — pairwise-comparison multi-criteria decision method
AMR	Autonomous Mobile Robot — mobile robot with autonomous localisation and route planning; the boundary from AGV terminology is not universal
AI/ML	Artificial Intelligence / Machine Learning
AoA	Angle of approach / attack — application-specific term; define the welding or process convention used
API	Application Programming Interface
ATC	Automatic Tool Changer
B2MML	Business to Manufacturing Markup Language — XML schema for ISA-95 data exchange
BIW	Body-in-White — the automotive welded body before painting
CAGR	Compound Annual Growth Rate
CAN	Controller Area Network — differential serial bus widely used in vehicles and industrial devices
CANopen	CAN-based open protocol for industrial automation
CBM	Condition-Based Maintenance — maintenance triggered by assessed asset condition rather than fixed intervals alone
CE	Conformité Européenne — EU conformity marking; not a third-party quality approval
CIM	Computer-Integrated Manufacturing

Acronym	Expansion and note
CIP	Common Industrial Protocol (used in EtherNet/IP, DeviceNet). Also Clean-in-Place in food industry.
CMMS	Computerised Maintenance Management System
COGS	Cost of Goods Sold
COTS	Commercial Off-The-Shelf
CP	Continuous Path motion; versus PTP
Cp / Cpk	Process Capability Indices — see Appendix A.8.1
CPS	Cyber-Physical System
CT	Cycle Time
DCS	Distributed Control System
DH	Denavit-Hartenberg parameters (Appendix A.2)
DMAIC	Define-Measure-Analyse-Improve-Control — Six Sigma improvement methodology
DMS	Dedicated Manufacturing System
DO/DI	Digital Output / Digital Input
DOF	Degree(s) of Freedom
DP	Decentralised Periphery (PROFIBUS)
DTS	Data Transport Service
EHEDG	European Hygienic Engineering & Design Group
EMC	Electromagnetic Compatibility
EoAT	End-of-Arm Tooling
ERP	Enterprise Resource Planning
ESPE	Electro-Sensitive Protective Equipment — e.g. light curtains
EtherCAT	Ethernet for Control Automation Technology — Beckhoff-originated industrial Ethernet
EUC	Equipment Under Control (IEC 61508)
F-DI / F-DO	Failsafe Digital Input / Output (safety-rated)
FAT	Factory Acceptance Test
FBD	Function Block Diagram — IEC 61131-3 programming language

Acronym	Expansion and note
FMEA	Failure Mode and Effects Analysis
FMS	Flexible Manufacturing System
FPY	First-Pass Yield
F/T	Force/Torque (sensor)
GAMP	Good Automated Manufacturing Practice
GMP	Good Manufacturing Practice
HAZOP	Hazard and Operability study
HIL / SIL*	Hardware-in-the-Loop / Software-in-the-Loop. *Not to be confused with Safety Integrity Level.
HMI	Human-Machine Interface
HRC	Human-Robot Collaboration
HRI	Human-Robot Interaction
ICS	Industrial Control System
IFR	International Federation of Robotics
IIoT	Industrial Internet of Things
IK / FK	Inverse / Forward Kinematics
IoT	Internet of Things
IP	Ingress Protection (e.g. IP65); also Internet Protocol
IRT	Isochronous Real-Time (PROFINET)
ISA	International Society of Automation
ISO	International Organization for Standardization — “ISO” is the organisation’s short name, not an acronym
IT / OT	Information Technology / Operational Technology
KPI	Key Performance Indicator
LD	Ladder Diagram — IEC 61131-3 programming language
LiDAR	LiDAR — Light Detection and Ranging; 2D or 3D optical ranging sensor
LiPo	Lithium Polymer (battery type)
LLM	Large Language Model
LOTO	Lockout-Tagout

Acronym	Expansion and note
LT	Lead Time
MES	Manufacturing Execution System
MoSCoW	Must/Should/Could/Won't — requirements prioritisation
MQTT	Message Queue Telemetry Transport
MRP	Media Redundancy Protocol (PROFINET); also Material Requirements Planning
MTBF	Mean Time Between Failures
MTTR	Mean Time To Repair (or Restore)
MTTF	Mean Time To Failure (for non-repairable items)
MTTFd	Mean Time To Dangerous Failure (ISO 13849-1)
NDT	Non-Destructive Testing
NEMA	National Electrical Manufacturers Association (USA)
NFC	Near Field Communication
NVA / VA	Non-Value-Added / Value-Added
OEE	Overall Equipment Effectiveness
OEM	Original Equipment Manufacturer
OLP	Offline Programming
OPC UA	Open Platform Communications Unified Architecture
OSSD	Output Signal Switching Device — safety-rated output
OT	Operational Technology — shop-floor systems (vs. IT)
PCB	Printed Circuit Board
PdM	Predictive Maintenance
PDCA	Plan-Do-Check-Act
PFHd	PFH_D — average probability of a dangerous failure per hour
PFL	Power and Force Limiting — collaborative-operation method requiring application-specific risk assessment and validation
PL	Performance Level (ISO 13849-1)
PLC	Programmable Logic Controller
PM	Preventive Maintenance; also Project Management

Acronym	Expansion and note
PoE	PoE — Power over Ethernet
IQ / OQ / PQ	IQ / OQ / PQ — Installation / Operational / Performance Qualification in regulated validation
PROFINET	PROFINET — industrial Ethernet system; use the official brand name rather than expanding it as an acronym
PTP	Point-to-Point motion; also Precision Time Protocol
QMS	Quality Management System
REBA / RULA	Rapid Entire Body Assessment / Rapid Upper Limb Assessment
RCM	Reliability-Centered Maintenance
RMS	Reconfigurable Manufacturing System; also Root Mean Square
ROI	Return on Investment
RoHS	Restriction of Hazardous Substances
ROP	Reorder Point
RTOS	Real-Time Operating System
SAT	Site Acceptance Test
SCADA	Supervisory Control and Data Acquisition
SCARA	Selective Compliance Assembly Robot Arm
SF	Safety Function; also Safety Factor — define the intended meaning locally
SFC	Sequential Function Chart
SIL	Safety Integrity Level (IEC 61508/62061)
SLAM	Simultaneous Localisation and Mapping
SLP	Systematic Layout Planning
SMED	Single-Minute Exchange of Die
SOP	Standard Operating Procedure
SPC	Statistical Process Control
SRMS	Safety-Rated Monitored Stop
SSM	Speed and Separation Monitoring
ST	Structured Text — IEC 61131-3 programming language
TCO	Total Cost of Ownership

Acronym	Expansion and note
TCP	Tool Centre Point; also Transmission Control Protocol
TEEP	Total Effective Equipment Performance
TPM	Total Productive Maintenance
TSN	Time-Sensitive Networking
TT	Takt Time
UR	Universal Robots
URS	User Requirements Specification
V&V	Verification and Validation
VFD	Variable Frequency Drive
VLAN	Virtual Local Area Network
WEEE	Waste Electrical and Electronic Equipment
WIP	Work in Progress
WMS	Warehouse Management System

C.2 Glossary of Key Terms

Table C.2. *Glossary of key terms.*

Term	Definition
Accuracy (positional)	The closeness of the robot-commanded pose to a reference pose in absolute coordinates. Contrasts with repeatability.
Availability	Fraction of planned production time during which equipment is running: $A = MTBF / (MTBF + MTTR)$. One of three OEE factors.
Bathtub curve	Plot of failure rate $\lambda(t)$ over equipment life showing infant mortality, random, and wear-out regions.
Brownfield / Greenfield	Brownfield: installation or modification within an existing facility and its constraints. Greenfield: installation in a new or substantially unconstrained facility. The additional brownfield effort is project-specific.
Cartesian robot	Gantry/linear-axis configuration with three or more orthogonal prismatic joints.
Cobot (collaborative robot)	Informal term for an industrial robot designed with features that may support collaborative applications. A robot is not “collaborative” by

Term	Definition
	itself; the complete application, task, tooling, workpiece, environment, and safeguards must be risk assessed.
Collaborative workspace	Workspace in which a person and robot system can perform tasks concurrently or sequentially under defined collaborative-operation measures. Boundaries and permissible interaction are established by the application risk assessment.
Controlled stop	Stop in which motion is decelerated under control. Depending on the safety function, energy may be retained (stop category 2) or removed after the controlled stop (category 1).
Cycle time	Elapsed time between the start of processing one part and the start of processing the next. A measurement.
Dead reckoning	Position estimation by integrating encoder or IMU data from a known initial pose. Accumulates drift; requires periodic correction from absolute sensors.
Delta robot	Parallel-kinematics robot with three or four parallelogram arms sharing a common end-effector platform. Fast, light-payload.
Digital twin	Synchronised virtual representation with governed data exchange between physical and virtual systems. A true operational twin supports bidirectional interaction; one-way monitoring alone is more accurately described as a digital shadow.
Duty cycle	Fraction of time an actuator or system is active versus idle. Affects thermal derating and component life.
Effective mass	Direction-dependent apparent inertia of the robot, tool, and payload at a potential contact point. It depends on configuration, contact direction, and dynamic model, not only total robot mass.
End-effector	Synonym for EoAT — the tool attached to the robot flange.
Enabling device (three-position)	Operator control whose intermediate position permits motion; both fully-pressed and fully-released states trigger a protective stop.
Fieldbus	Industrial communication system for sensors, actuators, drives, and controllers. Classical fieldbuses and industrial Ethernet technologies have different timing, topology, and diagnostic characteristics.
Footprint	Floor space occupied by the cell or system, typically measured in square metres. Includes safety perimeter, operator access, and logistics paths.
Hand guiding	Collaborative-operation method in which the operator commands robot motion through a hand-operated device or force/torque interface, subject to validated enabling, speed, and safety functions.

Term	Definition
Hard / soft real-time	Hard: missed deadline is a system failure (safety, motion sync). Soft: degraded performance but not failure (diagnostic logging).
Industry 4.0 / 5.0	Industry 4.0: cyber-physical integration with connectivity, data, simulation. Industry 5.0: adds human-centricity, sustainability, resilience.
Isochronous	Communication or execution occurring at controlled, phase-related intervals. PROFINET IRT and selected TSN profiles can support isochronous operation when the complete network is engineered accordingly.
Jacobian	Matrix relating joint rates to task-space velocity, $v = J(q)\dot{q}$. Rank and singular values determine which instantaneous task motions are feasible and how strongly joint motion is amplified.
Kanban	Pull-based inventory-signalling system from Toyota Production System. Limits WIP to a defined maximum.
Lean manufacturing	Production philosophy emphasising waste elimination, pull systems, and continuous improvement (kaizen).
Lights-out operation	Unattended operation, typically overnight or weekends. Requires high reliability and autonomous error recovery.
Line balancing	Distribution of tasks across stations so cycle times are approximately equal. Reduces wait losses.
Manipulability	Configuration-dependent measure derived from the Jacobian that describes motion or force capability. Yoshikawa's $v_{det}(JJ^T)$ is one common form, but its interpretation depends on scaling and task units.
OEE	Overall Equipment Effectiveness = Availability $\times$ Performance $\times$ Quality. Useful as an aggregate production metric when the time base, ideal cycle, and quality definitions are controlled.
Palletising	Arranging products on a pallet for storage or shipment. One of the oldest and most ubiquitous robot applications.
Payload	Allowable load at the robot flange under specified centre-of-gravity, inertia, orientation, speed, acceleration, and duty conditions. Selection must use the manufacturer load diagram or payload tool, not mass alone.
Perirobotic system	Equipment adjacent to the robot (fixtures, feeders, sensors, safety devices) — the 'peripheral' half of the integrated system. Romanian: SPR.

Term	Definition
Pick-and-place	Canonical handling operation: grip an object at source, move it, release at destination. The single most common robotic task.
Pose	Combined position (3 numbers) and orientation (3 numbers) of a rigid body in 3D space. Specified as a homogeneous transformation or a 6-vector.
Protective stop	Safety-function response that brings hazardous motion to a safe state. The stop category and whether drive power is retained or removed depend on the validated safety architecture and applicable standard.
Pull / Push production	Pull: downstream demand triggers upstream production (just-in-time). Push: upstream schedule drives production regardless of immediate demand.
Reach	Maximum distance from the robot's mounting axis that the TCP can access. Typically quoted for horizontal reach in a standard posture.
Reconfigurable Manufacturing System (RMS)	System designed for rapid reconfiguration between product families via modular, scalable, convertible architecture.
Repeatability	The spread of achieved poses when the robot is commanded to the same reference pose repeatedly. A precision metric, distinct from accuracy.
SCARA	Selective Compliance Assembly Robot Arm: 4-DOF robot (two horizontal rotary + vertical linear + wrist rotary). Fast for top-down assembly.
Singularity (kinematic)	Configuration where the Jacobian loses rank and certain end-effector motions become impossible. Wrist singularity, elbow singularity, shoulder singularity.
Skeleton model (digital twin)	Simplified kinematic model updated from sensor data. Distinct from full-physics digital twin, which models dynamics.
SMED	Single-Minute Exchange of Die — Shigeo Shingo's method for reducing changeover time, targeting < 10 minutes.
Takt time	Available production time divided by customer demand; the target pace. A demand-driven target, not a measurement.
Teach pendant	Portable operator terminal for programming the robot by moving it to reference poses and recording them. The primary human-to-robot interface for manual mode.
Throughput	Good parts produced per unit time. Constrained by the bottleneck station in an imbalanced line.

Term	Definition
Utilisation	Fraction of available capacity demanded or active. In an M/M/1 queue, $\rho = \lambda/\mu$ and waiting time rises steeply as ρ approaches 1; no single utilisation limit applies to every production system.
Value Stream	End-to-end flow from raw material to shipped product, analysed for waste and improvement opportunities.
Virtual commissioning	Pre-deployment verification of control logic, sequences, interfaces, and selected performance assumptions using a simulated plant or emulated equipment. Benefits depend on model fidelity, test coverage, and issue-management discipline.
Workspace	The full set of poses the robot can achieve; typically a complex 3D volume with shape determined by kinematics.
Zero position	Reference joint configuration with all joints at their zero angle; published by the robot manufacturer as the calibration datum.

Appendix D — Reference Data Tables

This appendix provides non-normative engineering reference data for early concept studies. Robot models, catalogue values, commercial availability, and prices change rapidly; every project must verify the current manufacturer datasheet, controller configuration, environmental variant, payload centre of gravity, and supplier quotation. Monetary ranges are indicative European budget figures for teaching purposes and exclude VAT, options, tooling, safety equipment, integration, and commissioning unless stated otherwise.

D.1 Typical Industrial Robot Specifications

D.1.1 6-Axis Articulated Robots (Small Payload, < 10 kg)

Table D.1. 6-axis articulated robots, small payload (≤ 10 kg).

Model	Payload (kg)	Reach (mm)	Repeatability (mm)	Indicative budget* (€)
ABB IRB 1100-4/0.475	4	475	± 0.01	14,000–18,000
ABB IRB 1200-7/0.7	7	700	± 0.02	16,000–21,000
FANUC LR Mate 200iD/7L	7	911	± 0.02	18,000–22,000
KUKA KR 6 R900-2	6	901	± 0.02	20,000–25,000
Yaskawa GP8	8	727	± 0.02	17,000–21,000
UR5e (cobot)	5	850	± 0.03	28,000–34,000
Omron TM5-700 (cobot)	6	700	± 0.05	26,000–30,000

D.1.2 6-Axis Articulated Robots (Medium Payload, 10–50 kg)

Table D.2. 6-axis articulated robots, medium payload (10–50 kg).

Model	Payload (kg)	Reach (mm)	Repeatability (mm)	Indicative budget* (€)
ABB IRB 2600-20/1.65	20	1650	± 0.04	38,000–45,000
ABB IRB 4600-45/2.05	45	2050	± 0.05	60,000–72,000
FANUC M-20iD/25	25	1831	± 0.02	52,000–60,000
FANUC M-710iC/50	50	2050	± 0.07	65,000–78,000
KUKA KR 20 R1810	20	1813	± 0.05	45,000–52,000
KUKA KR 50 R2500	50	2500	± 0.08	70,000–82,000

Model	Payload (kg)	Reach (mm)	Repeatability (mm)	Indicative budget* (€)
Yaskawa GP25	25	1730	±0.03	46,000–54,000
Yaskawa GP50	50	2061	±0.07	68,000–78,000

D.1.3 6-Axis Articulated Robots (Heavy Payload, > 100 kg)

Table D.3. 6-axis articulated robots, heavy payload (> 100 kg).

Model	Payload (kg)	Reach (mm)	Repeatability (mm)	Indicative budget* (€)
ABB IRB 6700-200/2.60	200	2600	±0.05	95,000–115,000
ABB IRB 7600-325/3.10	325	3100	±0.08	140,000–170,000
FANUC M-2000iA/900L	900	4680	±0.30	350,000–420,000
KUKA KR 210 R2700	210	2700	±0.06	115,000–135,000
KUKA KR 500-3 MT	500	2830	±0.08	180,000–220,000
Yaskawa MH215	215	2700	±0.05	105,000–125,000

D.1.4 SCARA Robots

Table D.4. SCARA robots.

Model	Payload (kg)	Reach (mm)	Repeatability (mm)	Indicative budget* (€)
Epson G6-651S	6	650	±0.015	14,000–18,000
Epson LS6-B602S	6	600	±0.020	9,000–12,000
Omron Cobra 800	5.5	800	±0.025	19,000–24,000
Stäubli TS2-100	14	1000	±0.020	28,000–35,000
Yamaha YK500XGL	5	500	±0.010	11,000–14,000

D.1.5 Delta (Parallel) Robots

Table D.5. Delta (parallel) robots. Cycle-rate values are vendor test figures under specified load and motion conditions and are not directly comparable.

Model	Payload (kg)	Reach dia. (mm)	Illustrative cycle rate* (picks/min)	Indicative budget* (€)
ABB IRB 360-6	6	1600	120	55,000–70,000
FANUC M-3iA/6S	6	1350	150	50,000–65,000
Omron Quattro 650HS	15	1300	143–200*	45,000–58,000

Model	Payload (kg)	Reach dia. (mm)	Illustrative cycle rate* (picks/min)	Indicative budget* (€)
Stäubli TP80 FAST	1	800	200	35,000–45,000

Commercial note. The budget columns are indicative teaching ranges, not manufacturer list prices or procurement quotations. Configuration, controller, software options, environmental protection, warranty, delivery, and negotiated discount can change the price materially. System-level investment is often several times the arm-and-controller cost once tooling, safeguarding, peripherals, engineering, validation, and commissioning are included. Refurbished equipment must be assessed individually using service history, accumulated operating hours, spare-parts availability, safety obsolescence, and warranty—not a generic age or hour threshold.

D.2 End-of-Arm Tooling — Typical Specifications

D.2.1 Vacuum Grippers

Table D.6. Vacuum gripper types and indicative characteristics; verify effective area, vacuum level, leakage, acceleration, orientation, and safety factor.

Parameter	Bellows cup	Flat cup	Foam pad	Vacuum tube
Typical diameter (mm)	20–80	10–100	100–400	30–60
Theoretical holding force at $\Delta p = 70$ kPa (N)	25–200	10–500	50–2000	50–180
Surface requirement	Uneven/flexible	Flat rigid	Uneven/porous OK	Rigid/flat
Cycle life (million)	3–8	5–15	0.5–2	2–5
Indicative component cost* (€)	8–30	5–40	80–250	25–80

D.2.2 Mechanical Grippers

Table D.7. Mechanical gripper types and indicative characteristics; verify jaw geometry, duty cycle, part tolerance, and fail-safe behaviour.

Parameter	Pneumatic 2-finger	Electric 2-finger	Pneumatic 3-finger	Soft gripper
Grip force (N)	50–500	50–400	100–800	5–30
Stroke (mm)	10–80	20–100	30–60	variable
Open/close time (ms)	50–200	100–400	80–250	200–500
Position repeatability	±0.1 mm	±0.02 mm	±0.1 mm	—

Parameter	Pneumatic 2-finger	Electric 2-finger	Pneumatic 3-finger	Soft gripper
Typical part types	Boxes, cylinders	Precision parts	Irregular/heavy	Delicate/variable
Indicative component cost* (€)	200–800	1,200–3,500	600–1,800	800–2,500

D.2.3 Force/Torque Sensors

Table D.8. Force/torque sensor classes and indicative commercial ranges; verify overload, bandwidth, cross-talk, environmental rating, and calibration.

Model class	Force range (N)	Torque range (Nm)	Resolution	Indicative budget* (€)
Basic 6-axis	±200 / ±500	±8 / ±20	0.2 % FS	3,000–6,000
Mid-range 6-axis	±500 / ±2000	±20 / ±100	0.05 % FS	6,000–12,000
High-precision 6-axis	±200	±8	0.001 N / 0.0001 Nm	15,000–25,000
Integrated joint-torque sensing	joint-dependent	joint-dependent	manufacturer-specific	included / option

D.3 Industrial Protocol Comparison

Table D.9. Industrial communication comparison. Timing ranges are illustrative and depend on network load, device count, topology, configuration, and implementation.

Protocol	Media	Typical update / cycle range	Topology	Typical use
PROFIBUS-DP	Twisted pair (RS485)	1–10 ms	Linear bus	Legacy factories; sensor/actuator networks
DeviceNet	CAN (shielded pair)	5–20 ms	Linear with drops	NA automotive, OEM equipment
PROFINET RT	Ethernet	1–10 ms	Star/line/ring	Modern PROFIBUS replacement; standard RT
PROFINET IRT	Industrial Ethernet with scheduled isochronous communication	0.25–1 ms*	Line/star/ring	Motion and tightly coordinated automation; requires compatible devices and engineered topology

Protocol	Media	Typical update / cycle range	Topology	Typical use
EtherNet/IP	Ethernet	1–10 ms	Star	NA factories; based on CIP
EtherCAT	Industrial Ethernet with on-the-fly frame processing	0.1–1 ms*	Line/tree/ring	Distributed motion and fast I/O; performance depends on device and master configuration
CC-Link IE	Ethernet	1–10 ms	Line/star	Japanese factories; Mitsubishi ecosystem
Modbus TCP	Ethernet/TCP	Application-dependent	Star/switched	Simple device integration and supervisory exchange; not inherently deterministic
OPC UA	TCP or PubSub transports	Application-dependent	Any IP topology	Semantic interoperability, information modelling, and supervisory/enterprise integration; not a generic motion bus
MQTT	TCP/TLS publish-subscribe	Application-dependent	Broker-based	Telemetry and event distribution with loose coupling; timing depends on broker, QoS, network, and application

D.4 Friction Coefficients and Gripping Forces

For a friction grip that must support a vertically oriented part, the available tangential friction must exceed gravity and the worst-case inertial load. A conservative normal-force requirement is:

$$F_{N,\text{total}} \geq [m \cdot (g + |a_t|) / (\mu)] \cdot SF$$

where $F_{N,\text{total}}$ is the total normal force across all effective contacts, m is workpiece mass, a_t is the acceleration component that loads the friction interface, μ is the verified static friction coefficient, and SF is an application-specific safety factor. If the force is shared equally by N contacts, the required normal force per contact is $F_{N,\text{total}}/N$. For other orientations, use a free-body diagram and include moments, centre-of-gravity offset, jaw geometry, impact, contamination, and loss of one contact where relevant.

Table D.10. Friction coefficients for common gripper–workpiece pairs.

Gripper surface / Workpiece	Static μ	Kinetic μ	Notes
Nitrile rubber / clean steel	0.9–1.2	0.7–0.9	Industry standard, good general purpose
Nitrile rubber / oily steel	0.3–0.5	0.2–0.3	Painted/oiled; force multiplier needed
Urethane / aluminium	0.8–1.0	0.6–0.7	Softer than nitrile, better for delicate parts
Steel / steel (bare)	0.4–0.6	0.3–0.4	Serrated jaws increase effective grip
Foam pad / cardboard	0.6–0.8	0.5–0.7	Compression adds conformance area
Silicone / glass	0.5–0.8	0.4–0.5	Non-marking; electronics applications
Vacuum (suction) / flat smooth	N/A	N/A	Normal holding force is $\Delta p \cdot \text{effective area}$; friction is relevant for lateral loads and must be checked separately.

D.5 Energy Consumption Reference Data

Energy demand varies with robot size, trajectory, payload, regenerative capability, auxiliaries, and duty cycle. The values below are broad planning ranges only; procurement and energy calculations require measured duty-cycle data or manufacturer power-consumption information for the selected system.

Table D.11. Energy consumption reference data for robotic cells.

Category	Small (< 10 kg payload)	Medium (10–50 kg)	Heavy (> 100 kg)
Idle (servo on)	0.2–0.5 kW	0.4–0.8 kW	0.8–1.5 kW
Typical operation	0.5–1.2 kW	1.0–2.5 kW	3.0–8.0 kW
Peak (acceleration)	2–4 kW	4–10 kW	15–40 kW
Compressed air (cell)	30–80 L/min	80–200 L/min	200–500 L/min

Compressed-air consumption includes EoAT, fixtures, valves, and blow-off. The example 200 L/min free-air flow operated for 4,000 h/year corresponds to approximately 48,000 Nm³/year. At €0.02–0.04 per Nm³, the direct compressed-air cost is about €960–1,920/year, before leakage and compressor-system losses. Site-specific tariffs and compressor efficiency must be used for a business case.

D.6 Reliability Planning Ranges for Robotic Components

Table D.12. Illustrative reliability-planning ranges. Use current vendor B10d, MTTFd, PFH_D, mission-time, service-interval, and field-return data for design decisions.

Component class	Typical MTBF	Dominant failure mode	MTTR (hours)
6-axis arm (mid-range)	60,000–80,000 h	Encoder, bearing	4–12
Servo drive	50,000–70,000 h	Electrolytic capacitor	1–4
Servo motor	70,000–100,000 h	Bearing, encoder seal	2–6
Harmonic reducer	30,000–50,000 h	Flex-spline fatigue	8–24
Cycloidal reducer	40,000–60,000 h	Bearing wear	8–16
Pneumatic gripper	3–8 M cycles	Seal, spring	0.5–2
Electric gripper	5–10 M cycles*	Gear train, encoder, cable, seals	1–4
Vacuum cup (natural rubber)	1–3 M cycles	Wear, cut damage	0.1–0.3
Pneumatic tool changer	500,000–1 M mates	Locking cone wear	0.5–2
Safety laser scanner	Use vendor reliability data	Optics contamination, electronics, scanning mechanism where applicable	1–3
Safety light curtain	Use vendor PFH_D / mission-time data	Emitter/receiver electronics, alignment, cabling	0.5–2
Industrial PC	50,000–70,000 h	HDD/SSD, PSU	1–4
Safety PLC	Use certified PFH_D / mission-time data	Power supply, I/O, communication, environmental stress	2–8

D.7 Cycle Time Benchmarks

The ranges below support preliminary cycle-time modelling only. They are not contractual benchmarks: travel distance, payload inertia, approach strategy, process dwell, vision retries, safety limits, blending, operator interaction, and quality criteria can dominate the result.

Table D.13. *Cycle-time benchmarks for common robotic operations.*

Operation	Typical cycle time	Notes
Pick-and-place (small, < 1 kg)	0.3–2.0 s	Delta robot best; 1.0–1.5 s typical for SCARA
Pick-and-place (medium, 5 kg)	1.5–4.0 s	6-axis arm; includes short travel
Pick-and-place (heavy, 50 kg)	3–10 s	Large robot, longer deceleration
Single spot-weld operation	2.5–4.0 s	Illustrative positioning + squeeze/weld/hold + retract; line cycle depends on parallelism and robot travel
Arc weld (linear seam, 300 mm)	8–20 s	At 0.6–1.5 m/min weld speed
Dispense bead (100 mm)	2–5 s	Plus pre-cure nozzle travel
Bin pick (3D vision, first try)	3–8 s	Plus occasional retry cycles
Palletise (single box)	3–6 s	Including approach/depart/stack
Screw driving (one fastener)	3–7 s	Including torque-verify dwell
Quality inspection (2D vision)	0.1–0.5 s	Image capture + inference
Quality inspection (3D scan)	1–4 s	Scan acquisition + processing

Appendix E — Project Templates and Checklists

This appendix provides concise working templates for the lifecycle of a robotic production system. Project teams should expand them into controlled Word, spreadsheet, requirements-management, or quality-system records as appropriate. Each template needs an owner, document identifier, revision, approval status, and retention rule. Concision is valuable, but required regulatory, contractual, and traceability fields must not be omitted.

E.1 User Requirements Specification (URS) Template

The URS is the project's first formal document and the basis for all downstream decisions. Minimum sections:

Table E.1. *User Requirements Specification (URS) template.*

Section	Content
1. Project identification	Project code, owner, stakeholders, signatory, date.
2. Business objectives	Why this project? Expected ROI, strategic rationale, constraints.
3. Scope	In-scope and explicitly out-of-scope elements. Interfaces with adjacent systems.
4. Functional requirements	What the system must do. Each requirement has a unique ID (e.g. FR-001), priority (MoSCoW), source, acceptance criterion.
5. Performance requirements	Quantitative targets: throughput, cycle time, OEE, availability, quality.
6. Non-functional requirements	Safety (reference standards), ergonomic constraints, noise limits, maintainability targets, cleanability, environmental tolerances.
7. Constraints	Physical footprint, utility limits (power, compressed air, cooling), integration with existing equipment, budget, schedule.
8. Assumptions	Working assumptions that, if false, invalidate the requirement. Document to avoid later blame-shifting.
9. Acceptance criteria	How the delivered system will be verified against the URS. Typically references FAT and SAT protocols.
10. Change control	Procedure for modifying the URS once frozen (who authorises, what impact analysis).

E.2 Risk Assessment Register Template (ISO 12100)

For each identified hazard, the register records the task and hazardous situation, the initial risk estimate, the three-step risk-reduction measures, any safety functions and their required performance, the residual risk, and the verification evidence. ISO 12100 does not prescribe one universal numerical risk matrix; the project must state and apply its chosen method consistently.

Table E.2. Risk Assessment Register template (ISO 12100).

Column	Content
Hazard ID	Unique identifier, revision status, and link to drawings or task analysis.
Hazard category	Mechanical, electrical, thermal, noise, vibration, radiation, substances/materials, ergonomic, environmental, or combined hazard.
Task / lifecycle phase	Transport, assembly, commissioning, automatic operation, setup, teaching, cleaning, fault recovery, maintenance, decommissioning, or reasonably foreseeable misuse.
Hazardous situation and sequence	Persons exposed, initiating event, hazardous event, possible harm, and existing controls.
Initial risk estimate	Severity and probability/exposure/avoidance factors using the project-defined method; record assumptions and rationale.
Risk-reduction measures	Apply and document the ISO 12100 three-step method: inherently safe design; safeguarding/complementary measures; information for use.
Safety function / SRP/CS	Define each safety function, trigger, safe state, response time, reset/restart behaviour, interfaces, and fault reaction where control-based risk reduction is used.
Required PLr or SIL	Determine separately for each safety function using ISO 13849-1 or IEC 62061, with documented rationale. PLr/SIL is not a residual-risk rating.
Residual risk	Re-estimate risk after measures. Identify remaining hazards, limitations, PPE, training, warnings, and information for use.
Verification / validation evidence	Reference calculations, drawings, SISTEMA or SIL verification, stopping-time tests, inspections, software tests, and validation records.
Owner / status / date	Named owner, action status, target and closure dates, approver, and links to evidence.
Change / review trigger	Conditions requiring reassessment: design change, incident, new product, altered speed/payload, software update, or new information.

E.3 FAT/SAT Test-Case Template

Each test case is an independent numbered record. The full FAT/SAT protocol is the ordered collection of all test cases, plus pre-test checklist and signoffs.

Table E.3. *FAT/SAT test-case template.*

Field	Content
Test case ID	Unique (e.g. FAT-WELD-042); sequence matters for traceability
Test category	Safety / Functional / Performance / Documentation / Integration
Requirement ID	Link to URS requirement being verified (e.g. FR-018)
Prerequisites	State required before test: power on, guards closed, specific product loaded, etc.
Procedure	Numbered steps. Keep actions atomic; one action per step.
Expected result	Specific, measurable, observable. Avoid vague ('works correctly').
Actual result	Filled during execution
Pass / Fail / Conditional	Use predefined acceptance logic. Any deviation requires a traceable NCR, concession, or approved retest before closure.
Tester name & signature	Who performed the test
Witness name & signature	Customer or independent witness where contractually or procedurally required.
Date	Execution date (and time for critical cases)
NCR reference	If Fail, reference to Non-Conformance Register

E.4 Change Control Register Template

Once the design is baselined, changes must be controlled through a formal register.

Table E.4. *Change Control Register template.*

Field	Content
Change ID	Unique (e.g. CR-007)
Raised by / date	Originator and date
Description	Clear statement of the proposed change
Rationale	Why is it needed? Business, technical, regulatory?
Impact analysis	Documents affected (URS, FS, DS, schematics, test cases), re-test required, schedule impact, cost impact, risk-reassessment required

Field	Content
Approval route	Approval authority and segregation of duties as defined by the project change-control procedure, quality system, and contract.
Approvers / dates	Signed approvals for each reviewer
Implementation owner	Who will execute the change
Re-verification	Test cases added/re-run; reference to execution records
Closure date	When change is fully implemented and verified

E.5 Pre-Commissioning Checklist

Complete this checklist before formal FAT execution. Any open item must be entered in an approved punch list with owner, risk assessment, due date, and explicit authorisation to proceed; otherwise FAT is deferred.

Table E.5. *Pre-commissioning checklist.*

Category	Checklist items
Mechanical	All mechanical assembly complete and torqued to spec. Fasteners marked as torqued. Robots installed on baseplates with calibrated anchor bolts. Safety fences secured. EoAT mounted. Cables dressed and strain-relieved.
Electrical	Supply, protection, isolation, protective bonding, phase sequence, panel wiring, labels, emergency-stop circuits, and drive parameters verified to the applicable design and test procedure. Record measured values and instrument calibration.
Pneumatic / hydraulic	Supply, isolation and stored-energy release, regulators, filtration, valves, hoses, fittings, pressure limits, and leak test verified against project-specific acceptance criteria.
Safety	Risk assessment approved for FAT scope. Safeguards and safety functions installed, configured, verified, and ready for validation. Safety software/version controlled; calculations and tool files available; emergency procedures posted.
Programming	All robot programs loaded. All PLC programs loaded. HMI screens tested for navigation. Operator manuals drafted.
Utilities	Power, compressed air, cooling water, exhaust/ventilation all verified at the cell interface.
Documentation	As-built drawings updated. Software backup taken. Calibration certificates filed. Spare parts delivered and stored. Manufacturer warranties registered.
Personnel	All test personnel identified and trained. PPE available. Witnesses scheduled. Emergency contact list posted.

E.6 SAT Discovery Log Template

Used during SAT and early production ramp-up to capture site-specific issues not caught at FAT. Reviewed weekly by a cross-functional team.

Table E.6. SAT Discovery Log template.

Field	Content
Log ID	Unique (e.g. DL-023)
Discovery date	When the issue was first observed
Observer	Who saw it (often an operator rather than an engineer)
Description	What happens, under what conditions, with what effect on production
Category	Integration / Utilities / Operator / Documentation / Fixture / Other
Severity	Critical (blocks production) / Major (significant impact) / Minor (cosmetic/minor)
Root cause	Once determined; blank until investigation complete
Resolution action	What will be done to close the entry
Owner	Single named individual responsible
Target date	Committed closure date
Verification	How closure is confirmed (retest, observation, signoff)
Status / Date closed	Open / In Progress / Verified / Closed

E.7 Validation Summary Report Outline (Regulated Industry)

For GMP-regulated environments, a Validation Summary Report (VSR) consolidates risk-based qualification and validation evidence into a controlled, approved record. The applicable scope and signatories depend on the regulated product, quality system, intended use, and current GxP requirements; the outline below is therefore a starting point rather than a universal regulatory form.

Table E.7. Validation Summary Report outline (regulated industries).

Section	Content
1. Executive summary	One-page overview: what was validated, outcome, signoff status.
2. Scope and purpose	System boundary, intended use, GxP impact and criticality, applicable regulations/guidance, and justified software/system classification where used.

Section	Content
3. Validation approach	Methodology (traditional V-model or agile-adapted); risk-based rationale per ICH Q9.
4. Qualification summary	IQ summary: installation verified. OQ summary: operational qualification results. PQ summary: performance qualification over representative production.
5. Test results	Summary of all test cases with pass/fail; reference to detailed protocols.
6. Non-conformances	All NCRs raised during validation, disposition, impact analysis.
7. Training records	Who was trained on what; evidence of competence.
8. Change control summary	All changes made during validation, through change control.
9. Conclusions and recommendations	Overall validation statement; any residual risks; any conditions on release to production.
10. Approvals	Signed by the roles required by the organisation's quality system and applicable regulation, typically System Owner and Quality Assurance; include Validation, Engineering, IT, or Qualified Person only where required.

E.8 Design Review Meeting Minutes Template

Each design review (concept, preliminary, critical, or pre-release) should produce controlled minutes that capture the evidence reviewed, decisions, actions, unresolved issues, and approvals. The record forms part of the project technical file and, where applicable, the regulated design-history or quality record.

Table E.8. Design review meeting minutes template.

Section	Content
Header	Project, review type (Concept/Preliminary/Critical/Pre-release), date, location, attendees with roles.
Agenda	Topics covered, time allocations. Deviations from agenda noted.
Materials presented	Reference to documents reviewed (drawing numbers, presentation file, simulation runs). Attach or link all.
Discussion summary	Key points raised, not minute-by-minute transcript. Focus on issues, decisions, trade-offs.
Decisions	Each decision with: wording, decider, date, basis. Link to documents where the decision is recorded (ECN, URS revision).

Section	Content
Action items	Each with ID, description, owner, target date, priority. Reviewed at next design review.
Open questions	Items not resolved in this meeting; scheduled for resolution before next review.
Next review	Type and scheduled date; prerequisites for holding it.
Approval	Named approvers, decision status, signatures or controlled electronic approval, and any dissent or conditions. Define the review and approval window in the project procedure rather than relying on implicit agreement.



The templates collected here are necessary but not sufficient. A template is a scaffolding for thinking; the engineering judgement to fill it in well — to distinguish a material risk from a theoretical one, a real requirement from a wish-list item, a meaningful acceptance criterion from a placeholder — is the work that separates competent project execution from excellence. Use these templates as starting points, adapt them to your organisation's conventions, and, above all, resist the temptation to let documentation effort displace engineering effort.

About this book

This textbook provides a comprehensive, practice-oriented guide to the design, integration, operation, and optimisation of robotic production systems. Written for advanced engineering students and practising engineers, it covers the complete lifecycle — from requirements analysis and layout design through robot selection, control architecture, safety engineering, validation, and commissioning.

Worked examples, common pitfalls, and industrial practice notes drawn from real installations bridge the gap between academic theory and factory-floor reality.

Key features

- Structured six-step design methodology for robotic cells
- AHP-based robot and EOAT selection with sensitivity analysis
- Safety engineering aligned with ISO 12100 and the ISO 10218:2025 series
- OEE, MTBF, and TCO calculations with worked examples
- Simulation-based validation and digital twin roadmap
- Industry 4.0 and 5.0 integration throughout