

Module 10: Robotic Arm

Contents

Module 10 - Robotic Arm	7
Overview	7
Historical Development	7
Robot Classifications	7
By Mechanical Configuration	7
By Application	8
Key Specifications	8
System Architecture	9
Workspace and Envelope	9
Applications in CNC Integration	9
Economic Considerations	9
Build vs. Buy Decision	10
Design Considerations	10
Module Objectives	11
Prerequisites	11
Safety Warning	11
References	11
Module 10 - Robotic Arm	12
Maintenance Schedule	12
Daily (Operator Checks)	12
Weekly Maintenance	12
Monthly Maintenance	12
Quarterly Maintenance (90 Days)	13
Annual Maintenance (Major Service)	13
Lubrication	13
Gearbox Lubrication	13
Joint Bearings	14
Linear Axes and Cable Carriers	14
Wear Items and Replacement	14
Consumables	14
Components with Limited Life	15
Encoders and Feedback	16
Encoder Maintenance	16
Encoder Battery Replacement	16
Calibration Maintenance	16

TCP Calibration Verification	16
Payload Calibration	17
Preventive Maintenance Best Practices	17
Record Keeping	17
Condition-Based Maintenance	17
Scheduled Downtime	18
Software Maintenance	18
Program Backups	18
Firmware Updates	18
Controller Maintenance	19
References	19
Module 10 - Robotic Arm	19
Systematic Approach	20
Troubleshooting Methodology	20
Motion and Positioning Issues	20
Robot Does Not Move	20
Positioning Errors	21
Erratic Motion or Vibration	22
Following Error Alarms	23
Electrical Issues	23
Drive Faults	23
Power Supply Issues	24
Cable and Connector Problems	24
Software and Programming Issues	24
Program Errors	24
Communication Failures	25
Gripper and Tool Issues	25
Vacuum Gripper Failures	25
Mechanical Gripper Issues	26
Controller and Software Issues	26
Controller Crashes or Freezes	26
Data Loss	26
Calibration Lost	27
Emergency Procedures	27
Collision Recovery	27
Runaway Robot	27
Electrical Fire or Smoke	28
Diagnostic Tools	28
References	29
Module 10 - Robotic Arm	29
Summary	29
Practical Implementation	30
Economic Considerations	30
Comparison with Pick and Place Systems	31
Future Trends	31
Learning Pathways	31

Resources for Continued Learning	32
Closing Thoughts	32
Next Steps	32
References	33
Module 10 - Robotic Arm	33
Coordinate Systems and Transformations	34
Forward Kinematics	34
Inverse Kinematics	36
Singularities	37
Workspace Analysis	37
Link Design	38
Joint Design	38
Mechanical Transmission	39
Gravity Compensation	39
Example Robot Design	39
References	40
Module 10 - Robotic Arm	40
Motor Selection	40
Servo Motors	40
Stepper Motors	41
Gear Reducers	41
Harmonic Drives	41
Planetary Gearboxes	42
Cycloidal Drives	42
Timing Belt Reduction	42
Bearings	42
Types for Robot Joints	42
Preload	43
Sealing	43
Brakes	44
Encoders and Feedback	44
Encoder Types	44
Encoder Mounting	45
Cable and Hose Routing	45
Joint Assembly Example	46
Performance Considerations	46
References	47
Module 10 - Robotic Arm	47
Tool Mounting Interface	47
ISO 9409 Standard	47
Custom Flanges	47
Grippers	47
Parallel Jaw Grippers (From Module 9, Applied to Arms)	47
Angular Grippers	48
Three-Jaw Grippers	48

Vacuum Grippers for Robotic Arms	48
Magnetic Grippers	49
Process Tools	49
Welding Tools	49
Cutting and Grinding Tools	49
Dispensing Tools	50
Inspection Tools	50
Tool Changers	51
Manual Tool Change	51
Automatic Tool Changers (ATC)	51
Tool Changer Sizing	52
Tool Design Considerations	52
Multi-Function End Effectors	53
References	53
Module 10 - Robotic Arm	53
Control Architecture	53
Coordinate Space Selection	54
Joint Space Motion	54
Cartesian Space Motion (Linear)	54
Circular Motion	54
Trajectory Generation	55
Velocity Profiling	55
Time-Optimal Trajectory	55
Path Planning	55
Point-to-Point Planning	55
Continuous Path Planning	56
Collision Avoidance	56
Advanced Path Planning	57
Control Algorithms	57
PID Control (Per Joint)	57
Advanced Control	57
Singularity Handling	58
Real-Time Considerations	58
Calibration	59
References	60
Module 10 - Robotic Arm	60
Force Sensing	60
Wrist-Mounted Force/Torque Sensors	60
Joint Torque Sensing	61
Force Control Strategies	61
Position Control with Force Limit	61
Hybrid Position/Force Control	62
Impedance Control	62
Admittance Control	63
Passive Compliance	63
Mechanical Compliance Devices	63

Passive vs. Active Compliance	64
Applications of Force Control	64
Assembly	64
Surface Finishing	65
Contact Inspection	65
Implementation Considerations	65
Control Loop Architecture	65
Stability	66
Calibration	66
Safety Considerations	66
References	67
Module 10 - Robotic Arm	67
Programming Methods	67
Teach Pendant Programming	67
Offline Programming (OLP)	68
Text-Based Programming	69
High-Level Languages	70
Coordinate Systems and Frames	71
Frame Definitions	71
Frame Transformations	71
Program Structure	72
Modular Programming	72
Variables and Data	72
Control Flow	73
Simulation	73
Collision Detection	73
Cycle Time Analysis	74
Reachability Analysis	74
Virtual Commissioning	74
Path Generation	74
CAD-Based Paths	74
Teach by Demonstration	75
Advanced Features	75
Conveyor Tracking	75
Vision Integration	75
Multi-Robot Coordination	76
References	76
Module 10 - Robotic Arm	76
Integration Architectures	76
Standalone Robot + CNC	76
PLC-Coordinated Cell	77
Integrated Control	77
Communication Protocols	77
Digital I/O (Hardwired)	77
Serial Communication	78
Industrial Ethernet	78

CNC Machine Tending	79
Door Control	79
Part Handling	80
Process Integration	80
Multi-Machine Cells	81
Single Robot, Multiple CNCs	81
Robot on Rail	81
Dual-Arm Cells	81
Auxiliary Equipment	82
Part Conveyors	82
Part Washers	82
Inspection Stations	82
Tool Changers and Racks	83
Cell Control Software	83
Supervisory Control	83
HMI (Human-Machine Interface)	83
Workcell Design Best Practices	84
Layout Optimization	84
Workflow Optimization	84
Maintenance Considerations	84
References	85
Module 10 - Robotic Arm	85
Applicable Standards	85
International Standards	85
North American Standards	86
Risk Assessment	86
Hazard Identification	86
Risk Evaluation	87
Risk Reduction	87
Safeguarding Methods	88
Physical Barriers	88
Safety Functions	89
Operating Modes	90
Automatic Mode	90
Manual Mode (Teach/Programming)	90
Maintenance Mode	90
Collaborative Operation	90
Collaborative Operating Spaces (ISO 10218-2)	90
Collaborative Methods (ISO/TS 15066)	90
Collaborative Robot Examples	91
Safety Validation	91
Pre-Commissioning Checks	91
Functional Testing	92
Performance Level Validation	92
Documentation	93
Safety Manual	93
Validation Report	93

Training Records	93
Common Safety Violations	93
Best Practices	93
References	93

Module 10 - Robotic Arm

Overview

Robotic arms represent the most versatile form of industrial automation, capable of complex manipulation tasks requiring multiple degrees of freedom and full spatial orientation control. Unlike the pick-and-place systems covered in Module 9, multi-axis robotic arms can perform welding, assembly, material removal, painting, inspection, and other tasks requiring dexterity and programmable tool paths in three-dimensional space.

Integration of robotic arms with CNC machining centers creates flexible manufacturing cells that combine subtractive manufacturing with automated handling, assembly, and finishing operations. This module explores articulated robot design, kinematics, control systems, and practical integration with CNC equipment.

Historical Development

Industrial Robot Evolution

1960s - First Generation: - Unimate (1961): First industrial robot, General Motors installation - Hydraulic actuation, limited programming - Fixed sequence operations - Heavy, dangerous, expensive

1970s - Second Generation: - Electric servo motors replace hydraulics - Microprocessor control enables complex programming - Teach pendant interfaces - Expanded applications (welding, painting)

1980s - Third Generation: - Sensor integration (vision, force) - Offline programming capabilities - Improved accuracy and repeatability - Reduced costs, wider adoption

1990s-2000s - Fourth Generation: - Advanced control algorithms - Network connectivity - Simulation and virtual commissioning - Collaborative safety features emerging

2010s-Present - Fifth Generation: - Collaborative robots (cobots) - AI and machine learning integration - Cloud connectivity and IoT - Simplified programming interfaces - Affordable systems for small manufacturers

Robot Classifications

By Mechanical Configuration

Articulated (6-Axis) Robots - Six revolute (rotational) joints - Spherical workspace - Maximum flexibility and reach - Most common industrial configuration - Examples: Fanuc, ABB, KUKA, Yaskawa

SCARA Robots - Selective Compliance Assembly Robot Arm - Covered in Module 9, but larger versions used for assembly - Fast horizontal motion, rigid vertical - Cylindrical workspace - Excellent for assembly operations

Cartesian/Gantry Robots - Three linear axes - Also covered in Module 9 - Large workspace scaling - Simple kinematics

Collaborative Robots (Cobots) - Designed for safe human interaction - Force/torque sensing in all joints - Rounded surfaces, limited force/speed - Examples: Universal Robots, Franka Emika, Doosan

Dual-Arm Robots - Two arms on common base or independent - Coordinated manipulation - Human-like task execution - Complex programming and control

By Application

Assembly Robots - Medium payload (5-20 kg) - High precision (+/-0.05mm) - Force control capabilities - Tool changer support

Material Handling Robots - High payload (50-500+ kg) - Moderate precision (+/-0.2mm) - Large reach (2-4m) - Robust construction

Welding Robots - Reach: 1.5-3m - Payload: 5-20 kg (torch weight) - Integrated welding controls - Wire feed and gas control

Painting/Coating Robots - Explosion-proof construction - Smooth motion profiles - Path accuracy over precision - Intrinsically safe (pneumatic options)

Machine Tending Robots - Payload: 5-50 kg - Integration with CNC machines - Door opening mechanisms - Part handling grippers

Inspection Robots - High precision - Sensor mounting (CMM probes, cameras) - Smooth, vibration-free motion - Repeatability critical

Key Specifications

Degrees of Freedom (DOF) - 6 DOF: Full position and orientation control (standard) - 7+ DOF: Redundant, allows obstacle avoidance while maintaining end effector pose - 4-5 DOF: Limited applications, lower cost

Payload Capacity - Collaborative: 3-35 kg - Industrial light: 5-20 kg - Industrial medium: 20-100 kg - Industrial heavy: 100-1000+ kg

Reach - Collaborative: 500-1300 mm - Industrial: 600-3500 mm - Long-reach: Up to 4500 mm

Repeatability - Collaborative: +/-0.02-0.1 mm - Precision industrial: +/-0.02-0.05 mm - Standard industrial: +/-0.05-0.2 mm - Heavy-duty: +/-0.2-0.5 mm

Speed - Joint angular velocity: 90-250 deg/s - TCP (Tool Center Point) velocity: 1-3 m/s - Cycle time dependent on motion profile

Protection Rating - IP40: Standard indoor - IP54: Dusty environments - IP65/67: Washdown, food industry - ATEX: Explosive atmospheres

System Architecture

Mechanical Components - Base (mounting to floor or gantry) - Links (arm segments) - Joints (6 rotational axes) - Wrist (last 3 axes for orientation) - Tool mounting flange (ISO 9409)

Actuation System - Servo motors (AC or DC brushless) - Harmonic drives or planetary gearboxes - Timing belts for wrist axes - Brakes for safety and position holding

Sensors - Joint encoders (absolute or incremental) - Force/torque sensors (optional) - Vision systems (external or end-effector mounted) - Proximity sensors, touch probes

Control System - Robot controller (dedicated computer) - Servo drives (one per axis) - Power supply - I/O interfaces - Safety controller

Programming Interface - Teach pendant (handheld) - PC-based software - Offline programming and simulation - High-level scripting (Python, etc.)

Workspace and Envelope

Reachable Workspace - All points the TCP can reach with any orientation - Typically a partial sphere or toroid shape - Workspace volume determined by link lengths

Dexterous Workspace - Points reachable with full orientation capability - Smaller than reachable workspace - Critical for assembly and manipulation tasks

Singularities - Configurations where robot loses a degree of freedom - Infinite joint velocities for certain TCP motions - Wrist singularity, shoulder singularity, elbow singularity - Must be avoided or handled carefully in path planning

Workspace Optimization - Position robot base to maximize useful workspace - Consider task requirements (approach angles, clearances) - Avoid workspace boundaries and singularities - Mounting options: floor, wall, ceiling, rail

Applications in CNC Integration

Machine Tending - Load raw stock into CNC - Unload finished parts - Part flipping for multi-operation machining - Inspection and measurement integration

Deburring and Finishing - Automated grinding after machining - Polishing and surface finishing - Force-controlled material removal - Consistent quality and cycle time

Assembly - Combine machined parts with purchased components - Press-fit, screwing, riveting operations - Vision-guided pick and place - Quality verification

Welding - Post-machining welding operations - Fixture-to-fixture transfers - Automated tack welding - Seam tracking with sensors

Inspection - CMM-style measurement with touch probe - Vision-based dimensional inspection - Surface defect detection - Barcode/label verification

Economic Considerations

Capital Costs - Collaborative robots: \$25,000-\$50,000 - Industrial robots (small): \$30,000-\$80,000 - Industrial robots (medium): \$80,000-\$150,000 - Industrial robots (large): \$150,000-

\$500,000+ - Controller, teach pendant, cables included - End effectors and tooling: Additional \$2,000-\$20,000

Integration Costs - Mechanical installation: \$5,000-\$30,000 - Electrical integration: \$5,000-\$20,000 - Programming and commissioning: \$10,000-\$50,000 - Safety systems and guarding: \$10,000-\$100,000 - Total project cost typically 1.5-3x robot purchase price

Operating Costs - Electricity: \$0.50-\$2.00/day - Maintenance: \$1,000-\$5,000/year - Programming updates: Variable - Periodic recalibration: \$1,000-\$3,000

Return on Investment - Labor replacement: 1-3 operators (\$60,000-\$180,000/year) - Increased throughput: 20-50% typical - Quality improvement: Reduced scrap and rework - Flexibility: Quick changeover vs. dedicated automation - Typical payback: 1-3 years for machine tending, 6 months-2 years for assembly

Build vs. Buy Decision

Commercial Industrial Robots

Advantages: - Proven reliability and performance - Comprehensive support and documentation - Safety certifications - Offline programming tools - Established spare parts supply

Disadvantages: - High initial cost - Proprietary interfaces - Vendor lock-in - Limited customization

DIY/Open-Source Robots

Advantages: - Lower cost (30-50% of commercial) - Full customization and control - Educational value - Open-source software (ROS - Robot Operating System)

Disadvantages: - Higher technical skill required - No warranty or support - Safety certification challenges - Longer development time - Examples: AR2, AR3, AR4 (open-source designs)

Collaborative Robots (Middle Ground) - Simplified programming (graphical interfaces) - Easier integration (plug-and-play) - Lower safety costs (certified safe operation) - Growing ecosystem of accessories - Examples: Universal Robots UR series, Doosan collaborative line

Design Considerations

Payload Requirements - Part weight plus end effector weight - Dynamic loading during acceleration - Safety factor (typically 2x) - Payload vs. reach trade-off (less payload at full extension)

Reach Requirements - Maximum distance from base to work area - Consider mounting height and position - Allow margin for approach angles - Verify no interference with surroundings

Cycle Time - Required throughput determines speed - Motion time vs. process time (welding, assembly) - Optimize robot placement to minimize travel - Balance speed with precision requirements

Precision and Repeatability - Assembly: +/-0.05mm or better - Machine tending: +/-0.1-0.2mm acceptable - Consider cumulative errors in kinematic chain - Calibration and maintenance requirements

Environment - Temperature range (operating and storage) - Humidity and moisture exposure
- Dust, chips, and contamination - Washdown requirements (food, pharma) - Explosive atmospheres (ATEX rating)

Integration Requirements - Communication protocols (Ethernet, fieldbus) - I/O requirements (digital, analog) - Vision system compatibility - CNC interface standards - Safety system integration

Module Objectives

By completing this module, you will be able to:

1. Understand articulated robot kinematics (forward and inverse)
2. Select appropriate robot configuration for specific applications
3. Design and integrate end effectors and tool changers
4. Program robot motion using teach pendant and offline methods
5. Implement force control for assembly and finishing
6. Integrate robots with CNC machines and manufacturing cells
7. Design safety systems compliant with ISO 10218 and ISO/TS 15066
8. Perform calibration, maintenance, and troubleshooting
9. Optimize performance for cycle time and quality
10. Evaluate build vs. buy decisions for robot integration

Prerequisites

This module builds on: - **Module 3:** Linear motion principles (some joints use linear actuators) - **Module 4:** Control electronics (servo drives, sensors, I/O) - **Module 9:** Pick and place systems (foundation for robotic manipulation) - **Module 14:** LinuxCNC HAL (if using open-source control) - **Module 15:** G-code fundamentals (some robots use G-code derivatives)

Familiarity with coordinate systems, transformations, and basic kinematics is helpful.

Safety Warning

Industrial robots can cause serious injury or death through crushing, impact, entanglement, or other hazards. All robotic systems must comply with ISO 10218 (industrial robots), ISO/TS 15066 (collaborative robots), ANSI/RIA R15.06, and applicable local regulations. Proper risk assessment, safeguarding, emergency stops, and operator training are mandatory. Never bypass safety systems or enter operating robot workspace without proper lockout/tagout procedures.

Next: 10.2 Arm Kinematics and Design

References

1. **ISO 10218-1:2011** - Robots and robotic devices - Safety requirements
2. **ISO 9283:1998** - Manipulating industrial robots - Performance criteria
3. **Denavit, J. & Hartenberg, R.S. (1955).** "A Kinematic Notation for Lower-Pair Mechanisms." *ASME Journal of Applied Mechanics*, 22, 215-221

4. **Craig, J.J. (2017).** *Introduction to Robotics: Mechanics and Control* (4th ed.). Pearson
 5. **Lynch, K.M. & Park, F.C. (2017).** *Modern Robotics*. Cambridge University Press
 6. **ABB Robot Studio Software** - Robot simulation and programming
 7. **KUKA System Software (KSS)** - Robot control and motion planning
-

Module 10 - Robotic Arm

Regular maintenance ensures reliable operation, prevents costly downtime, and extends robot lifespan. This section covers preventive maintenance schedules, common wear items, and best practices.

Maintenance Schedule

Daily (Operator Checks)

Visual Inspection - Check for unusual sounds or vibrations - Inspect cables and hoses for visible damage - Verify no fluid leaks (oil, grease, hydraulic) - Ensure workspace clear of debris - Check guarding intact

Functional Checks - Test emergency stop function - Verify normal startup sequence - Check teach pendant operation - Observe motion smoothness

Cleanliness - Wipe exterior surfaces - Remove chips or debris from robot - Clean gripper or tool surfaces - Keep work area organized

Weekly Maintenance

Cable and Hose Inspection - Check for abrasion or cracks - Verify strain reliefs secure - Inspect cable carriers (drag chains) - Test connector tightness

Tool and Gripper - Inspect gripper jaws for wear - Check vacuum cups (if applicable) - Test gripper actuation - Verify tool mounting secure

Sensors - Clean vision system camera lens - Check proximity sensor alignment - Test force/torque sensor (if equipped) - Verify encoder signals

Safety Systems - Test all interlock switches - Verify light curtains unobstructed - Check safety mat function - Inspect fence and gates

Monthly Maintenance

Lubrication - Lubricate grease points per manufacturer schedule - Check oil levels (gearboxes if accessible) - Apply light oil to cable carriers - Lubricate linear axes (if external)

Mechanical Inspection - Check all visible fasteners (tighten if loose) - Inspect belts for wear and tension - Listen for bearing noise - Verify joint motion smoothness

Calibration Check - Move to known reference positions - Measure accuracy with dial indicator or gauge - Compare to baseline measurements - Recalibrate if drift >0.1 mm

Electrical - Inspect control cabinet interior - Check for loose connections - Verify fan operation and clean filters - Inspect motor connections

Software - Backup robot programs - Review error logs - Clear old log files - Update position data backups

Quarterly Maintenance (90 Days)

Detailed Mechanical - Inspect gearbox condition (noise, temperature) - Check brake function and wear - Measure backlash (should be <0.05mm at TCP) - Inspect pneumatic components

Electrical Testing - Test emergency stop response time - Measure motor winding resistance - Check encoder signal quality (oscilloscope) - Test battery backup (if equipped)

Gripper Maintenance - Replace worn vacuum cups - Inspect pneumatic cylinder seals - Calibrate gripper force (if measurable) - Check parallel jaw alignment

Calibration - Full TCP calibration (four-point method) - Verify all tool offsets - Check work object frames - Mastering (joint zero positions if accessible)

Annual Maintenance (Major Service)

Gearbox Service - Inspect harmonic drive flexsplines for wear - Check planetary gearboxes for backlash - Replace gearbox oil (if applicable) - Measure gear noise and vibration

Bearing Inspection - Check all joint bearings for play - Listen for roughness or noise - Measure bearing temperature under load - Replace if excessive wear detected

Timing Belt Replacement - Replace all timing belts (preventive, every 2000-5000 hours) - Inspect pulleys for wear - Re-tension and verify alignment

Cable Replacement - Replace motor and encoder cables (if showing wear) - Inspect power cables for damage - Check flexible conduits and cable carriers

Brake Testing - Measure brake holding torque - Test brake release/engage timing - Inspect brake pads or discs - Adjust or replace if worn

Encoder Verification - Check encoder alignment - Verify count accuracy (rotate 360°, check counts) - Test index pulse (if incremental) - Replace if erratic behavior

Controller Maintenance - Clean control cabinet thoroughly - Replace cooling fans if noisy - Check capacitor condition (power supply) - Firmware/software updates if available

Safety System Validation - Complete functional test of all safety functions - Measure stop times and distances - Verify performance levels maintained - Document results

Lubrication

Gearbox Lubrication

Harmonic Drives

Lubricant: - Manufacturer-specified grease - Typically NLGI Grade 0 or 1 - Molybdenum disulfide or lithium-based

Interval: - Initial: 200-500 hours - Ongoing: 2000-5000 hours (varies by model and load) - More frequent if high-duty cycle

Procedure: - Remove drain plug (if accessible) - Inject grease via grease nipple - Rotate joint through full range - Wipe excess grease - Reinstall drain plug

Planetary Gearboxes

Oil Bath: - Change oil every 10,000-20,000 hours - Use specified viscosity (ISO VG 220 typical) - Check oil level and condition

Grease-Packed: - Sealed for life or 20,000+ hours - Monitor temperature and noise - Replace gearbox if signs of failure

Joint Bearings

Grease-Lubricated Bearings

Schedule: - Every 2000-5000 hours depending on speed and load - More frequent in harsh environments

Grease Type: - Lithium or polyurea-based - NLGI Grade 2 - Temperature range: -40°C to 120°C - Low noise specification

Application: - Remove old grease if accessible - Apply fresh grease via grease fitting - Rotate joint to distribute - Do not over-pack (increases friction)

Sealed Bearings

Maintenance: - No lubrication possible - Monitor for roughness or noise - Replace when worn (typically 20,000+ hours)

Linear Axes and Cable Carriers

External Linear Guides

If robot has external linear axes: - Oil or grease per guide manufacturer - Typically every 100 km of travel or monthly - Wipe rails clean first - Apply thin, even coat

Cable Carriers

Light oil on pivot points: - Prevents wear and squeaking - Every 6 months - Use dry lubricant in dusty environments

Wear Items and Replacement

Consumables

Vacuum Cups

Inspection: - Check for cracks, tears, or hardening - Test suction with vacuum gauge - Replace if grip force reduced

Lifespan: - 6-12 months typical - Shorter if handling abrasive parts - Keep spares on hand

Gripper Jaws

Soft Jaws: - Urethane pads: 6-12 months - Rubber: 3-6 months - Depends on part characteristics

Hard Jaws: - Machined aluminum: Years - Replace if damaged or worn

Timing Belts

Inspection: - Check for cracks on tooth surface - Look for fraying at edges - Measure belt tension

Replacement: - Preventive: Every 2000-5000 hours - Immediate if damage visible - Replace both belt and idlers

Filters

Control Cabinet Air Filters: - Check monthly - Replace when clogged or dirty - Typically every 3-6 months

Pneumatic Filters: - Drain daily - Replace element every 6-12 months

Components with Limited Life

Flexsplines (Harmonic Drives)

Lifespan: - 3,000 to 10,000 hours typical - Depends on torque, speed, and reversals

Symptoms of Wear: - Increased backlash - Positioning errors - Unusual noise - Higher motor current

Replacement: - Requires gearbox disassembly - Factory service or experienced technician - Replace with OEM parts

Brakes

Wear Items: - Friction pads or discs - Springs (set brakes)

Inspection: - Measure holding torque (should meet spec) - Check brake engagement time - Listen for unusual noise

Replacement: - Every 10,000-50,000 actuations - Or when holding torque drops below spec

Batteries (Controller Backup)

Purpose: - Maintain encoder position memory - Preserve programs during power loss

Replacement: - Every 1-3 years - Typically lithium 3.6V cells - Low battery warning from controller

Procedure: - Power on robot controller - Replace battery while powered (preserves data) - Or backup/restore after battery change

Cooling Fans

Lifespan: - 30,000-50,000 hours (3-6 years continuous) - Bearings wear out

Symptoms: - Noisy operation - Reduced airflow - Overheat alarms

Replacement: - Standard 120mm or 80mm fans - Match voltage (24V DC or 120V AC) - Verify CFM rating

Encoders and Feedback

Encoder Maintenance

Cleaning

Optical Encoders: - Keep code disc clean (if exposed) - Use compressed air only - Never touch disc surface

Magnetic Encoders: - Less sensitive to contamination - Wipe sensor head if accessible

Alignment Verification

Symptoms of Misalignment: - Erratic position readings - Intermittent errors - Following error alarms

Check: - Encoder coupling for looseness - Alignment between motor and encoder shafts - Mounting security

Signal Quality

Test with Oscilloscope: - Channels A and B should be clean square waves - 90° phase shift between A and B - Amplitude within specification (typically 5V or differential)

Poor Signals Indicate: - Contamination - Misalignment - Cable damage - Failing encoder

Encoder Battery Replacement

Absolute Encoders with Battery Backup

Purpose: - Maintain position data during power loss - Avoid re-mastering (homing)

Procedure: 1. Note current position data (backup) 2. Power on controller 3. Replace battery while powered (no data loss) 4. Verify position retained 5. Or: Replace battery unpowered, then re-master

Mastering Procedure (if position lost)

Different per manufacturer: 1. Mechanical mastering: Move to physical reference marks 2. Electronic mastering: Use master tool or controller function 3. Store new zero positions in controller

Critical: - Mastering errors cause positioning errors - Follow manufacturer procedure exactly - Verify with test program after mastering

Calibration Maintenance

TCP Calibration Verification

Procedure

1. Jog robot to known reference point from four different orientations
2. Measure variation in reference point position
3. Variation should be <0.1 mm
4. If larger, re-calibrate TCP

Automated Calibration

Some systems: - Touch probe on robot - Probe calibrated artifact (sphere, plane) - Controller calculates TCP automatically

Payload Calibration

Purpose

Accurate payload data improves: - Motion control (feedforward compensation) - Collision detection sensitivity - Energy efficiency

Update When

- Tool or gripper changed
- Payload significantly different
- Poor motion quality observed

Method

1. Mount tool with known mass and center of gravity
2. Run identification routine (moves through poses, measures currents)
3. Controller calculates payload parameters
4. Store in tool data

Preventive Maintenance Best Practices

Record Keeping

Maintenance Log

Document: - Date and time - Maintenance performed - Parts replaced - Measurements taken - Technician name - Next service due

Trend Monitoring

Track over time: - Positioning accuracy - Backlash measurements - Motor currents - Temperature - Cycle times

Detect degradation before failure.

Parts Inventory

Maintain Stock: - Critical spares (gripper jaws, vacuum cups) - Long-lead items (gearboxes, motors) - Consumables (filters, belts, grease)

Track: - Part numbers - Quantity on hand - Reorder levels - Vendor and lead time

Condition-Based Maintenance

Vibration Monitoring

Baseline Measurement: - When robot new or after maintenance - Measure at each joint

Periodic Checks: - Monthly or quarterly - Increased vibration indicates bearing wear

Current Monitoring

Normal Operating Current: - Record baseline for each joint - Higher current indicates friction or binding - Lower current may indicate belt slippage

Temperature Monitoring

Thermal Camera: - Check gearbox and motor temperatures - Hot spots indicate problems - Typical: 40-60°C above ambient

Scheduled Downtime

Planning

Coordinate with Production: - Schedule major maintenance during planned shutdowns - Avoid unscheduled breakdowns - Bundle tasks to minimize downtime

Critical Spare Parts

Keep on Site: - Motors (base, shoulder joints) - Gearboxes (most critical joints) - Cables (motor power, encoder, teach pendant) - Controller backup or spare boards

Software Maintenance

Program Backups

Frequency

Daily or Weekly: - Automated backup to network drive - Versioned backups (keep last 10)

After Changes: - Immediately after program modifications - Before firmware updates - Before major maintenance

Backup Contents

- All robot programs
- Position data (taught points)
- Tool and work object calibrations
- Configuration files
- Safety parameters

Storage

Multiple Locations: - On robot controller - Network server - USB drive or external media - Off-site backup

Firmware Updates

When to Update

Good Reasons: - Bug fixes for known issues - New features needed - Security patches

Caution: - If system working well, consider not updating - Test on non-production robot first - Have rollback plan

Procedure

1. Backup everything (programs, configs)
2. Read release notes thoroughly
3. Verify compatibility (robot model, options)
4. Schedule downtime
5. Perform update
6. Test all functions
7. Verify programs run correctly
8. Keep previous firmware available for rollback

Controller Maintenance

Hard Drive / SSD

Industrial Controller: - Hard drive lifespan: 5-7 years - SSD more reliable (10+ years)

Preventive: - Replace hard drive every 5 years - Image backup before replacement - Some systems: Compact Flash or SD card (replace every 3 years)

Battery Backup (CMOS)

Controller PC Motherboard: - CR2032 battery - Replace every 2-3 years - Prevents BIOS settings loss

Next: 10.11 Troubleshooting

References

1. ISO 10218-1:2011 - Robots and robotic devices - Safety requirements
 2. ISO 9283:1998 - Manipulating industrial robots - Performance criteria
 3. Denavit, J. & Hartenberg, R.S. (1955). "A Kinematic Notation for Lower-Pair Mechanisms." *ASME Journal of Applied Mechanics*, 22, 215-221
 4. Craig, J.J. (2017). *Introduction to Robotics: Mechanics and Control* (4th ed.). Pearson
 5. Lynch, K.M. & Park, F.C. (2017). *Modern Robotics*. Cambridge University Press
 6. ABB Robot Studio Software - Robot simulation and programming
 7. KUKA System Software (KSS) - Robot control and motion planning
-

Module 10 - Robotic Arm

Systematic troubleshooting minimizes downtime and identifies root causes. This section covers common problems, diagnostic procedures, and solutions.

Systematic Approach

Troubleshooting Methodology

- 1. Gather Information** - What is the symptom? - When did it start? - What changed recently? - Is it constant or intermittent? - Any error codes or alarms?
- 2. Verify the Problem** - Reproduce the issue - Note exact conditions - Document error messages - Capture screenshots or videos
- 3. Check Obvious Causes** - Emergency stop engaged? - Safety devices triggered? - Power supply issues? - Correct operating mode?
- 4. Isolate the Subsystem** - Mechanical, electrical, or software? - Specific joint or axis? - Controller, drive, or motor?
- 5. Form Hypothesis** - List possible causes (most to least likely) - Based on symptoms and experience
- 6. Test Hypothesis** - Make one change at a time - Verify result before proceeding - Use measurements, not assumptions
- 7. Implement Solution** - Fix root cause, not symptoms - Verify complete resolution - Document solution and preventive measures

Motion and Positioning Issues

Robot Does Not Move

Symptom: No motion when commanded

Check List

- 1. Emergency Stop**
 - Is e-stop pressed?
 - Reset and verify release
 - Check all e-stops in system
- 2. Operating Mode**
 - Correct mode selected (Auto/Manual)?
 - Enabling device active (if manual mode)?
 - Mode selector key switch position
- 3. Safety Circuits**
 - Are all guards closed?
 - Light curtains clear?
 - Safety relay status
 - Check safety circuit wiring
- 4. Controller Status**
 - Controller powered on?
 - No fault codes displayed?
 - Motors enabled?
 - Servo drives ready (LED indicators)
- 5. Motor Power**
 - Main contactor energized?

- Drive enable signal present?
- Check drive fault LEDs
- Measure motor supply voltage

6. Software Limits

- Position within software limits?
- Check limit status in controller
- Jog away from limit if at boundary

Diagnostic Procedure

1. Press e-stop, release, observe behavior
 - No change: E-stop circuit issue
2. Check controller display for faults
 - Drive fault: Investigate specific drive
 - Communication fault: Check cables
3. Manually jog single joint (if possible)
 - Works: Program or command issue
 - Fails: Hardware problem
4. Monitor drive enable signal (oscilloscope/multimeter)
 - Not present: Controller or safety circuit
 - Present but no motion: Drive or motor fault

Positioning Errors

Symptom: Robot doesn't reach commanded position accurately

Possible Causes

1. **Calibration Drift**
 - Check: Move to known taught position, measure deviation
 - Solution: Re-calibrate TCP or re-master joints
 - Preventive: Regular calibration checks
2. **Encoder Issues**
 - Check: Encoder signal quality (oscilloscope)
 - Symptoms: Erratic positioning, intermittent errors
 - Solution: Re-align or replace encoder
 - Check encoder cable integrity
3. **Mechanical Backlash**
 - Check: Reverse direction, measure position change
 - Sources: Belt stretch, gear wear, loose couplings
 - Solution: Tighten belts, replace worn gears, adjust preload
4. **Gearbox Wear**
 - Check: Measure backlash at output
 - Harmonic drive: <1 arcmin acceptable, >3 arcmin replace
 - Planetary: <5 arcmin acceptable
 - Solution: Replace or rebuild gearbox
5. **Structural Deflection**

- Check: Position error correlates with payload or orientation
- Measure: Use dial indicator under load
- Solution: Reduce payload, reinforce structure, or accept limitation

6. Thermal Expansion

- Check: Worse after warmup?
- Measure: Position when cold vs. hot
- Solution: Allow warmup period, temperature compensation

Diagnostic Test

Repeatability Test: 1. Move to target position 20 times 2. Measure position each time 3. Calculate standard deviation 4. Repeatability should be <0.05 mm for most robots

If repeatable but inaccurate: - Calibration issue - Kinematic parameter error

If not repeatable: - Mechanical looseness - Encoder problems - Backlash

Erratic Motion or Vibration

Symptom: Robot shakes, oscillates, or moves roughly

Possible Causes

1. Control Tuning

- PID gains too high
- Check: Reduce proportional and derivative gains
- Tune: Start low, increase gradually
- Use auto-tune if available

2. Mechanical Resonance

- Structure vibrates at specific frequency
- Check: Occurs at certain speeds or positions?
- Solution: Change acceleration, add damping, adjust gains

3. Encoder Noise

- Electrical interference
- Check: Encoder signals with oscilloscope
- Solution: Shield cables, separate from power, ground properly

4. Loose Components

- Bolts, couplings, bearings
- Check: Inspect and torque all fasteners
- Listen: Rattle or clicking indicates looseness

5. Worn Bearings

- Check: Roughness when manually rotating joint
- Listen: Grinding noise
- Feel: Excessive play
- Solution: Replace bearings

6. Belt Issues

- Too loose: Vibration, tooth jumping
- Too tight: Excessive bearing load, motor strain
- Check: Belt tension (1-2% deflection)
- Solution: Adjust tensioner

Diagnostic Procedure

Isolate Affected Joint: 1. Move only one joint at a time 2. Identify which joint(s) vibrate 3. Vary speed and acceleration 4. Note when vibration occurs

Test with No Load: - Remove tool or payload - If vibration stops: Inertia or balance issue - If continues: Internal robot problem

Following Error Alarms

Symptom: Position error exceeds limit, robot stops

Causes

1. Excessive Load

- Payload beyond capacity
- Friction or binding
- Check: Reduce speed, check load rating

2. Acceleration Too High

- Commanded acceleration exceeds capability
- Solution: Reduce acceleration in program

3. Servo Drive Fault

- Insufficient torque output
- Check: Drive status LEDs, error codes
- Test: Swap drive with known good

4. Encoder Problems

- Skipped counts, poor signals
- Check: Signal quality
- Solution: Replace encoder or cable

5. Gearbox Binding

- Damaged gears, lack of lubrication
- Check: Motor current (high indicates resistance)
- Listen: Grinding noise
- Solution: Service or replace gearbox

Electrical Issues

Drive Faults

Ovvovoltage - Cause: Excessive regenerative energy during deceleration - Check: DC bus voltage - Solution: Slower deceleration, regenerative resistor

Ovvocurrent - Cause: Short circuit, motor fault, overload - Check: Motor winding resistance (should be few ohms, balanced) - Inspect: Cable damage, motor contamination - Solution: Repair or replace motor, check load

Ovvotemperature - Cause: Excessive duty cycle, inadequate cooling - Check: Drive and motor temperatures - Verify: Fan operation, clean heat sinks - Solution: Reduce duty cycle, improve cooling

Encoder Fault - Cause: Cable damage, encoder failure, noise - Check: Encoder cable continuity - Test: Swap with another axis encoder - Solution: Replace cable or encoder

Communication Error - Cause: Network issue, cable fault, EMI - Check: Network cable integrity
- Verify: Termination resistors - Solution: Replace cable, add shielding

Power Supply Issues

Controller Won't Power On

Check: 1. Main power supply voltage (measure at input) 2. Circuit breakers and fuses 3. Control transformer (if present) 4. Power supply outputs (24V DC, etc.)

Intermittent Power Loss

Causes: - Loose connections - Overload (breaker trips) - Failing power supply

Diagnosis: - Monitor voltages during operation - Check current draw - Thermal image of connections (hot spots)

Cable and Connector Problems

Intermittent Faults

Symptoms: - Errors appear and disappear - Occur during specific motions - Vibration-related

Diagnosis: - Wiggle cables while monitoring - Flex cables through normal range - Inspect for kinks or sharp bends

Solutions: - Re-terminate connectors - Replace damaged cables - Improve cable routing and strain relief

Open or Short Circuits

Test: - Multimeter continuity check - Measure resistance (each conductor to ground, conductor to conductor) - Compare to good cable

Software and Programming Issues

Program Errors

Unreachable Position

Error: Inverse kinematics has no solution

Causes: - Position outside workspace - Orientation not achievable - Near singularity

Solutions: - Adjust target position - Use different joint configuration - Add intermediate via points

Singularity Errors

Symptom: Cannot execute Cartesian move

Causes: - Path passes through or near singularity - Wrist, shoulder, or elbow singularity

Solutions: - Use joint space motion (MOVEJ instead of MOVEL) - Reorient approach to avoid singularity - Add via point to guide path around singularity

Syntax Errors

Specific to robot programming language:

- Check documentation for correct syntax
- Verify variable declarations
- Match brackets, parentheses, END statements

Logic Errors

Program runs but incorrect behavior:

- Step through program in simulation or slow motion
- Check variable values
- Verify conditional logic
- Add logging or status messages

Communication Failures

Robot-to-CNC Communication

Symptoms:

- Timeout errors
- No handshaking
- Wrong data received

Diagnosis:

1. Verify network settings (IP addresses, subnet masks)
2. Test connectivity (ping)
3. Monitor traffic (Wireshark or network analyzer)
4. Check protocol settings (baud rate, parity for serial)

Solutions:

- Correct IP address conflicts
- Replace damaged network cables
- Verify protocol configuration matches

I/O Not Working

Digital Output:

- Check: Physical wiring to output terminal
- Measure: Voltage at terminal (should be 24V when on, 0V when off)
- Test: Toggle output from controller
- Verify: Correct output number in program

Digital Input:

- Check: Sensor wiring and power
- Test: Sensor independently (continuity or voltage)
- Measure: Input terminal voltage
- Verify: Correct input number in program

Gripper and Tool Issues

Vacuum Gripper Failures

No Vacuum

Check:

1. Compressed air supply pressure (6 bar typical)
2. Venturi generator condition (clogged?)
3. Solenoid valve operation (manual test)
4. Tubing for leaks (hissing sound)
5. Vacuum gauge reading

Weak Vacuum

Causes:

- Worn vacuum cups (cracks, hardening)
- Air leaks in system
- Low supply pressure
- Venturi wear

Solutions:

- Replace vacuum cups
- Find and seal leaks (soap solution test)
- Increase supply pressure
- Replace venturi generator

Part Drops

Causes:

- Insufficient vacuum
- Part too heavy or porous
- Excessive acceleration
- Vibration

Solutions:

- Larger or more vacuum cups
- Increase vacuum (higher flow venturi)
- Reduce acceleration
- Improve grip surface (foam vs. bellows cups)

Mechanical Gripper Issues

Won't Close or Open

Pneumatic: - Check air pressure - Verify solenoid valve switching - Inspect cylinder seals - Look for obstructions

Electric: - Check motor power - Verify control signals - Test encoder feedback - Inspect mechanical jam

Weak Grip

Pneumatic: - Increase air pressure (within gripper rating) - Check for air leaks - Inspect jaws for wear

Electric: - Increase current limit (if adjustable) - Check motor and gearbox

Jaws Misaligned

Causes: - Bent guide rods - Worn bushings - Loose mounting

Solutions: - Replace damaged parts - Realign and secure

Controller and Software Issues

Controller Crashes or Freezes

Symptoms - Unresponsive teach pendant - Controller display frozen - Program stops unexpectedly

Diagnosis

Check: 1. CPU load (if monitoring available) 2. Memory usage 3. Error logs for software exceptions 4. Network traffic (broadcast storms?)

Solutions

- Reboot controller (last resort)
- Identify and fix software bug
- Optimize program (reduce loop complexity)
- Update firmware (if known bug)

Preventive

- Regular software backups
- Avoid overly complex programs
- Monitor system resources
- Keep firmware updated

Data Loss

Programs or Positions Lost

Causes: - Battery failure (encoder or controller) - Hard drive failure - Corruption during save

Solutions: - Restore from backup - Re-teach positions (if no backup) - Replace battery/hard drive

Preventive: - Regular backups (automated) - Replace batteries on schedule - Use redundant storage

Calibration Lost

Mastering Lost

Symptoms: - Position offsets by consistent amount - Robot moves to wrong absolute positions

Causes: - Encoder battery dead - Power loss during operation - Encoder replaced

Solution: - Re-master all joints (follow manufacturer procedure) - Verify with test program

TCP Calibration Incorrect

Symptoms: - Position accuracy varies with orientation - Circular paths not circular

Solution: - Re-calibrate TCP using four-point method - Verify all tool offsets

Emergency Procedures

Collision Recovery

After Impact

1. Immediate:

- Press emergency stop
- Ensure no personnel injured
- Assess damage visually

2. Inspection:

- Check for bent links or joints
- Inspect gearboxes (listen for damage)
- Verify encoder alignment
- Look for cracked welds or fasteners

3. Functional Test:

- Manually move each joint (power off, brakes released)
- Check for binding or unusual resistance
- Power on, jog each joint slowly
- Run test program at low speed

4. Recalibration:

- Verify accuracy at known positions
- Re-calibrate if needed
- Check for new vibrations or noises

5. Root Cause:

- Identify why collision occurred
- Fix programming error, sensor failure, or procedural issue
- Implement preventive measures

Runaway Robot

Uncontrolled Motion

Actions: 1. Press emergency stop immediately 2. Do not attempt to manually stop robot 3. Evacuate area 4. Lockout/tagout after motion stops

Causes: - Encoder failure (motor doesn't know position) - Drive fault (continuous torque output) - Software error

Investigation: - Do NOT restart until root cause identified - Check encoder signals - Inspect drive LEDs and error codes - Review program logic - Consult manufacturer support

Electrical Fire or Smoke

Response

1. Press emergency stop
2. If safe, disconnect main power
3. Evacuate personnel
4. Call emergency services
5. Use Class C fire extinguisher (electrical) if small and safe

Do NOT: - Use water - Open control cabinet if fire inside (oxygen feeds fire) - Restart until inspected by qualified electrician

Diagnostic Tools

Essential Tools

- Multimeter (voltage, resistance, continuity)
- Oscilloscope (encoder signals, noise)
- Dial indicator (positioning accuracy)
- Teach pendant (built-in diagnostics)
- Laptop with controller software

Helpful Tools

- Thermal camera (hot spots, bearing failures)
- Vibration analyzer (bearing condition)
- Vacuum gauge (gripper diagnosis)
- Network cable tester
- Spare cables (for swapping tests)

Documentation

- Electrical schematics
- Mechanical drawings
- Software manuals
- Error code lists
- Manufacturer support contacts

Next: 10.12 Conclusion

References

1. **ISO 10218-1:2011** - Robots and robotic devices - Safety requirements
 2. **ISO 9283:1998** - Manipulating industrial robots - Performance criteria
 3. **Denavit, J. & Hartenberg, R.S. (1955).** "A Kinematic Notation for Lower-Pair Mechanisms." *ASME Journal of Applied Mechanics*, 22, 215-221
 4. **Craig, J.J. (2017).** *Introduction to Robotics: Mechanics and Control* (4th ed.). Pearson
 5. **Lynch, K.M. & Park, F.C. (2017).** *Modern Robotics*. Cambridge University Press
 6. **ABB Robot Studio Software** - Robot simulation and programming
 7. **KUKA System Software (KSS)** - Robot control and motion planning
-

Module 10 - Robotic Arm

Summary

Robotic arms represent the pinnacle of flexible automation, combining mechanical design, advanced control, sensors, and programming to perform complex manipulation tasks. This module covered the complete spectrum of robotic arm technology from fundamental kinematics through practical workcell integration.

Key Topics Covered:

Kinematics and Design - Understanding forward and inverse kinematics enables accurate positioning and motion planning. The Denavit-Hartenberg convention provides a systematic framework for describing robot geometry. Singularities, workspace analysis, and mechanical design considerations determine robot capability and performance.

Joint Mechanisms and Actuators - Harmonic drives, planetary gearboxes, and timing belts transmit motor torque to joints. Bearing selection, sealing, and brake design impact precision, reliability, and safety. Proper motor sizing balances performance with cost and energy consumption.

End Effectors and Tooling - Grippers, process tools, and sensors enable robots to interact with parts and perform work. Automatic tool changers provide flexibility for multi-task operations. Proper TCP calibration ensures accurate tool positioning across all robot orientations.

Motion Planning and Control - Trajectory generation, path planning, and control algorithms determine how smoothly and efficiently robots move. PID control, feedforward compensation, and advanced methods like computed torque control enable precise tracking. Collision avoidance and singularity handling ensure safe and reliable operation.

Force Control and Compliance - Force/torque sensing and impedance control enable robots to interact safely with environments and perform contact tasks like assembly and finishing. Passive compliance devices offer simple mechanical solutions, while active force control provides programmable behavior.

Programming and Simulation - Multiple programming methods serve different needs: teach pendants for intuitive position teaching, offline programming for complex paths without production interruption, and text-based languages for sophisticated logic. Simulation validates programs before deployment, reducing risk and optimizing cycle time.

CNC and Workcell Integration - Robotic arms integrated with CNC machines create flexible manufacturing cells. Communication protocols, handshaking sequences, and cell control software coordinate equipment. Multi-machine cells maximize utilization and enable lights-out manufacturing.

Safety Systems - Comprehensive safety design protects personnel while maintaining productivity. ISO 10218, ISO/TS 15066, and ANSI/RIA R15.06 establish requirements for industrial and collaborative robots. Safeguarding methods include physical barriers, interlocks, presence-sensing devices, and safety-rated control functions. Collaborative operation enables safe human-robot interaction.

Maintenance and Troubleshooting - Preventive maintenance prevents failures and extends equipment life. Regular lubrication, inspection, and calibration maintain performance. Systematic troubleshooting approaches minimize downtime when issues occur. Understanding common failure modes and diagnostic procedures enables rapid recovery.

Practical Implementation

Successfully implementing a robotic arm system requires integration across disciplines:

Mechanical Design - Select robot architecture matching application requirements - Design or specify fixtures and work presentation - Ensure adequate workspace and accessibility - Account for payload and reach requirements - Integrate end effectors and tool changers

Electrical Integration - Size power supply and circuit protection - Implement safety circuits (emergency stop, interlocks) - Route cables properly (shielding, separation from power) - Integrate I/O, sensors, and networks - Provide grounding and surge protection

Software and Control - Configure controller for robot kinematics - Implement motion planning and trajectory generation - Program task sequences and decision logic - Integrate vision, force sensing, and process equipment - Develop operator interface and diagnostics

Safety and Compliance - Perform comprehensive risk assessment - Implement appropriate safeguarding (guards, interlocks, sensors) - Configure safety-rated control functions (STO, SLS, etc.) - Validate safety system performance - Train operators and maintain documentation

Economic Considerations

Robotic arms deliver significant value when properly applied:

Costs - Robot system: \$25,000-\$500,000+ depending on size and features - Integration: 1.5-3x robot cost (mechanical, electrical, programming, safety) - Operating costs: Maintenance, energy, programming updates - Total project: \$50,000-\$1,500,000 for complete workcell

Benefits - Labor savings: Eliminate 1-3 operators per shift - Increased throughput: 20-50% improvement over manual - Quality improvement: Consistent positioning, reduced scrap - Flexibility: Quick changeover between part types - Safety: Remove humans from hazardous operations - Typical ROI: 1-3 years for machine tending, 6 months-2 years for assembly

Build vs. Buy - DIY open-source robots (AR2, AR3): Educational value, 30-50% cost savings, higher technical burden - Commercial industrial robots: Proven reliability, support, safety certifica-

tions, higher cost - Collaborative robots: Easier integration, lower safety costs, growing capability
- Decision factors: Budget, technical capability, support requirements, volume

Comparison with Pick and Place Systems

Robotic Arms (Module 10) vs. Pick and Place (Module 9)

Robotic Arms: - 6+ degrees of freedom (full orientation control) - Complex manipulation and process tasks - Higher cost and complexity - Versatile (welding, assembly, machining, inspection)
- Longer programming and setup time

Pick and Place: - 3-4 degrees of freedom (limited orientation) - Focused on material handling - Simpler and lower cost - Optimized for speed (especially Delta robots) - Easier to program and deploy

Selection: - Pick and place: If task is primarily material handling - Robotic arm: If task requires complex orientation, multiple tools, or process integration

Future Trends

AI and Machine Learning - Vision-based learning (recognize parts without programming) - Adaptive motion planning (learn optimal paths from experience) - Predictive maintenance (detect failures before they occur) - Natural language programming (describe task verbally)

Advanced Sensing - Improved force/torque sensing (higher resolution, lower cost) - 3D vision for bin picking and part recognition - Tactile sensing (artificial skin) - Multi-modal fusion (vision + force + sound)

Collaborative Capabilities - Higher payloads while maintaining safety - Better human intent prediction - Intuitive programming (demonstration, AR/VR) - Mobile collaborative robots (combine mobility with manipulation)

Connectivity and Integration - OPC UA and other Industry 4.0 standards - Digital twins (real-time simulation matches reality) - Cloud-based programming and monitoring - Fleet management for multiple robots

Accessibility - Continued cost reduction - Simplified programming (graphical, no-code) - Plug-and-play accessories ecosystem - Rental and Robot-as-a-Service models

Learning Pathways

Beginner Projects 1. Study commercial robot kinematics and simulation 2. Build simple 3-DOF arm (desktop scale) 3. Program pick-and-place sequence 4. Integrate basic sensors (proximity, force)

Intermediate Projects 1. Build or acquire 6-DOF robot (AR3, used industrial) 2. Implement inverse kinematics and trajectory planning 3. Integrate vision system for part localization 4. Design CNC machine tending cell 5. Implement automatic tool changer

Advanced Projects 1. Force-controlled assembly or finishing application 2. Multi-robot coordinated workcell 3. Collaborative robot implementation (safety validation) 4. Custom end effector with embedded intelligence 5. Integration with MES/ERP systems

Resources for Continued Learning

Standards and Regulations - ISO 10218-1/2: Robot safety (industrial) - ISO/TS 15066: Collaborative robots - ANSI/RIA R15.06: North American robot safety - ISO 13849-1: Safety control systems

Technical References - “Introduction to Robotics” by John J. Craig (kinematics and control) - “Robot Modeling and Control” by Spong, Hutchinson, Vidyasagar - “Springer Handbook of Robotics” (comprehensive reference) - Manufacturer technical documentation (ABB, KUKA, Fanuc, Universal Robots)

Software Tools - ROS (Robot Operating System): Open-source framework - RoboDK: Affordable simulation and offline programming - Manufacturer software (RobotStudio, Sim Pro, RoboGuide) - OpenRAVE, MoveIt: Research and advanced development

Online Communities - ROS Discourse and GitHub - Robotics Stack Exchange - CNCzone robotics forum - Manufacturer user forums - r/robotics subreddit

Professional Organizations - Robotic Industries Association (RIA) - IEEE Robotics and Automation Society - International Federation of Robotics (IFR)

Closing Thoughts

Robotic arms extend human capability, performing tasks with precision, repeatability, and endurance that manual operation cannot match. Success requires understanding the fundamentals of kinematics, mechanics, control, and integration while maintaining rigorous attention to safety.

The skills developed in this module—coordinate transformations, motion planning, sensor integration, programming, and system design—apply broadly to automation, mechatronics, and advanced manufacturing. Robotic arms serve as an excellent platform for learning these disciplines because they integrate so many technologies in one system.

As manufacturing evolves toward greater flexibility and customization, robotic arms become increasingly central. Understanding their design, programming, integration, and optimization positions you to contribute to this transformation. The combination of declining costs, improving capabilities, and simplified interfaces makes robotic automation accessible to organizations of all sizes.

Whether integrating a commercial industrial robot, building an open-source system for learning, or deploying collaborative robots for flexible manufacturing, the principles remain consistent: design for the application, integrate thoughtfully, prioritize safety, and continuously optimize.

Next Steps

After completing this module:

1. **Explore Simulation:** Download RoboDK or manufacturer simulation software, import robot models, and practice programming
2. **Study Commercial Systems:** Visit manufacturer websites (ABB, KUKA, Fanuc, Universal Robots), watch application videos, understand specifications

3. **Hands-On Project:** Build or acquire a small robot arm (open-source design or educational kit), implement kinematics, program simple tasks
4. **Integration Design:** Specify a robotic cell for a real application, layout workcell, select equipment, estimate costs and ROI
5. **Safety Study:** Review ISO 10218 and ISO/TS 15066, understand risk assessment methodology, design compliant safety system
6. **Advanced Topics:** Explore force control, vision integration, advanced motion planning, or specialized applications (welding, inspection, etc.)

The journey from understanding basic kinematics to designing and operating complete robotic workcells is challenging but rewarding. Each successful project builds capability and confidence. Start with achievable goals, learn from failures, and progressively tackle more complex challenges.

Robotic arms represent some of the most sophisticated automation technology in modern manufacturing. Mastering their design, programming, and integration opens doors to high-value engineering work and positions you at the forefront of manufacturing technology.

Module 10 Complete

Previous Module: Module 9 - Pick and Place Robot Systems

Next Module: Module 11 - Large-Format FDM 3D Printing

Return to: Course Overview

References

1. **ISO 10218-1:2011** - Robots and robotic devices - Safety requirements
 2. **ISO 9283:1998** - Manipulating industrial robots - Performance criteria
 3. **Denavit, J. & Hartenberg, R.S. (1955).** "A Kinematic Notation for Lower-Pair Mechanisms." *ASME Journal of Applied Mechanics*, 22, 215-221
 4. **Craig, J.J. (2017).** *Introduction to Robotics: Mechanics and Control* (4th ed.). Pearson
 5. **Lynch, K.M. & Park, F.C. (2017).** *Modern Robotics*. Cambridge University Press
 6. **ABB Robot Studio Software** - Robot simulation and programming
 7. **KUKA System Software (KSS)** - Robot control and motion planning
-

Module 10 - Robotic Arm

Understanding robot kinematics is essential for design, programming, and control of articulated arms. This section covers coordinate transformations, forward and inverse kinematics, and mechanical design considerations.

Coordinate Systems and Transformations

Reference Frames

Base Frame: - Fixed to robot mounting surface - Origin typically at base center - Z-axis points up (convention)

Joint Frames: - One per joint, following Denavit-Hartenberg (D-H) convention - Defines position and orientation of each link - Standardized parameterization

Tool Frame (TCP - Tool Center Point): - Attached to end effector - Origin at functional point (gripper center, torch tip, probe ball) - Orientation defines tool direction

World Frame: - External reference (workcell, part fixture) - Robot base frame positioned relative to world frame - Multiple robots share common world frame

Homogeneous Transformations

Represent position and orientation in single 4×4 matrix:

$$T = \begin{bmatrix} R & p \\ \theta & 1 \end{bmatrix}$$

Where: - $R = 3 \times 3$ rotation matrix - $p = 3 \times 1$ position vector

Combine rotations and translations:

$$T_{\text{total}} = T_1 \times T_2 \times T_3 \times \dots \times T_n$$

Denavit-Hartenberg (D-H) Parameters

Standard method to describe robot kinematics with four parameters per joint:

- theta (theta): Joint angle (variable for revolute joint)
- d: Link offset along previous Z-axis
- a (alpha): Link length along X-axis
- α (alpha): Link twist about X-axis

Transformation from joint i-1 to joint i:

$$T_i = \text{Rot}_Z(\theta_i) \times \text{Trans}_Z(d_i) \times \text{Trans}_X(a_i) \times \text{Rot}_X(\alpha_i)$$

Forward Kinematics

Definition: Given joint angles, calculate end effector position and orientation.

6-Axis Articulated Robot Example

Standard industrial robot configuration (similar to Fanuc, ABB):

Joint 1: Base rotation (waist) Joint 2: Shoulder rotation Joint 3: Elbow rotation Joint 4: Wrist roll
Joint 5: Wrist pitch Joint 6: Wrist yaw (tool flange)

D-H Parameter Table (Example)

Joint	θ_i	d_i	a_i	α_i
1	θ_1	d_1	0	90°

Joint	theta_i	d_i	a_i	α_i
2	theta2-90°	0	a2	0°
3	theta3	0	a3	90°
4	theta4	d4	0	-90°
5	theta5	0	0	90°
6	theta6	d6	0	0°

Typical values:
- d1 = 400 mm (base height)
- a2 = 500 mm (upper arm)
- a3 = 400 mm (forearm)
- d4 = 450 mm (wrist length)
- d6 = 100 mm (flange to TCP)

Forward Kinematics Calculation

$$T_{0_6} = T_{0_1} \times T_{1_2} \times T_{2_3} \times T_{3_4} \times T_{4_5} \times T_{5_6}$$

Result is 4×4 matrix giving TCP position and orientation:

$$T_{0_6} = \begin{bmatrix} R_{0_6} & p_{0_6} \\ 0 & 1 \end{bmatrix}$$

Position: $p_{0_6} = [X, Y, Z]$ Orientation: R_{0_6} (can be converted to Euler angles or quaternion)

Implementation

Typically implemented in controller firmware. For verification or custom controllers:

```
import numpy as np

def dh_transform(theta, d, a, alpha):
    """Create D-H transformation matrix"""
    ct = np.cos(theta)
    st = np.sin(theta)
    ca = np.cos(alpha)
    sa = np.sin(alpha)

    return np.array([
        [ct, -st*ca, st*sa, a*ct],
        [st, ct*ca, -ct*sa, a*st],
        [0, sa, ca, d],
        [0, 0, 0, 1]
    ])

def forward_kinematics(joint_angles, dh_params):
    """Calculate TCP pose from joint angles"""
    T = np.eye(4)
    for i, (theta, d, a, alpha) in enumerate(dh_params):
        theta_i = theta + joint_angles[i] # Add variable joint angle
        T = T @ dh_transform(theta_i, d, a, alpha)
    return T
```

Inverse Kinematics

Definition: Given desired TCP position and orientation, calculate required joint angles.

Challenges - Non-linear equations - Multiple solutions (typically 8 for 6-axis robot) - Singularities where solutions undefined - Computational complexity

Solution Methods

Analytical (Closed-Form): - Derive equations algebraically - Fast computation - Specific to robot geometry - Most industrial robots have analytical solutions

Numerical: - Iterative optimization (Newton-Raphson, Jacobian-based) - General approach for any geometry - Slower, may not converge - Used when analytical solution unavailable

Analytical Inverse Kinematics (Simplified)

For standard 6-axis articulated robot:

Step 1: Position of Wrist Center

```
p_wrist = p_tcp - d6 * R_tcp * [0, 0, 1]
```

Subtract tool length along tool Z-axis.

Step 2: Joint 1 (Base Rotation)

```
theta1 = atan2(p_wrist_y, p_wrist_x)
```

Step 3: Joints 2 and 3 (Arm Triangle)

Solve triangle formed by shoulder, elbow, wrist:

```
r = sqrt(p_wrist_x^2 + p_wrist_y^2)
```

```
s = p_wrist_z - d1
```

```
D = sqrt(r^2 + s^2)
```

```
costheta3 = (D^2 - a2^2 - a3^2) / (2 * a2 * a3)
```

```
theta3 = +/-acos(costheta3) # Elbow up or down
```

```
theta2 = atan2(s, r) - atan2(a3*sin(theta3), a2 + a3*cos(theta3))
```

Step 4: Joints 4, 5, 6 (Wrist Orientation)

Calculate rotation from current orientation to desired:

```
R_3_6 = R_0_3^(-1) * R_0_6
```

Extract Euler angles from R_3_6:

```
theta5 = acos(R_3_6[2,2])
```

```
theta4 = atan2(R_3_6[1,2], R_3_6[0,2])
```

```
theta6 = atan2(R_3_6[2,1], -R_3_6[2,0])
```

Multiple Solutions

Typical 8 solutions arise from: - Joint 1: +/- configuration (front/back) - Joint 3: +/- (elbow up/down) - Joint 5: +/- (wrist flip)

Controller selects solution based on:

- Closest to current position (minimum joint motion)
- Avoids joint limits
- Avoids singularities
- User-specified preferences

Singularities

Definition: Configurations where robot loses a degree of freedom, making certain TCP motions impossible.

Types

Wrist Singularity (Gimbal Lock):

- Joint 5 = 0° or 180°
- Joints 4 and 6 axes align
- Cannot rotate about one axis
- Small TCP rotation requires large J4, J6 motion

Shoulder Singularity:

- Wrist center directly above or below base axis
- Joint 1 undefined
- Infinite solutions for J1

Elbow Singularity:

- Arm fully extended or folded
- Wrist center at boundary of reachable workspace
- Joint 3 at limit

Handling Singularities

Avoidance:

- Path planning keeps away from singular configurations
- Define keep-out zones

Damped Least Squares:

- Numerical method to gracefully handle near-singularities
- Slightly deviates from desired path to maintain stability

Reorientation:

- Temporarily change tool orientation to exit singularity
- Resume desired orientation after clear

Workspace Analysis

Reachable Workspace

Maximum extent robot can reach:

$$R_{\max} = a_2 + a_3 + d_4 + d_6 \text{ (fully extended)}$$

$$R_{\min} = |a_2 - a_3| - d_4 \text{ (fully folded)}$$

Workspace typically partial toroid shape with voids near singularities.

Dexterous Workspace

Region where full orientation control available:

- Smaller than reachable workspace
- Excludes singular configurations
- Critical for assembly tasks

Workspace Optimization

Robot Placement:

- Position base to maximize useful workspace coverage
- Consider multiple tasks and work areas
- Allow clearance for approach angles

Mounting Options:

- Floor mount: Standard, stable
- Inverted ceiling mount: Frees floor space, gravity aids downward force
- Wall mount: Side access to workspace
- Rail mount: Extended linear reach, multi-station service

Link Design

Material Selection

Aluminum Alloys: - 6061-T6, 7075-T6 - Lightweight, good strength-to-weight - Easy machining and welding - Standard for small to medium robots

Carbon Fiber Composites: - Highest strength-to-weight ratio - Reduces inertia, increases speed - More expensive, harder to manufacture - Used in high-performance robots

Steel: - Heavy-duty robots and bases - High stiffness and strength - Increased inertia

Structural Considerations

Stiffness: - Minimize deflection under load - Use structural shapes (box section, I-beam) - FEA analysis for optimization

Resonance: - Natural frequencies above servo bandwidth (>50 Hz) - Damping (material selection, damping pads)

Cable Routing: - Internal channels for cables and hoses - Rotating joints require cable management - Strain relief at entry/exit points

Mass Distribution

Center of Gravity: - Keep close to joint axes - Reduces torque requirements - Improves dynamic performance

Inertia: - Minimize rotational inertia - Concentrate mass near joints - Affects acceleration capability

Joint Design

Revolute Joint Construction

Components: - Bearings (crossed roller, angular contact pairs) - Motor and gearbox (harmonic drive, planetary) - Encoder (absolute or incremental) - Brake (holding position when unpowered) - Housing and seals

Bearing Selection

Load Types: - Radial (perpendicular to shaft) - Axial (parallel to shaft) - Moment (tilting)

Bearing Types: - Deep groove ball: Radial load, low cost - Angular contact: Combined radial and axial - Crossed roller: High rigidity, compact - Tapered roller: Heavy loads

Preload: - Eliminates play for precision - Dual bearing configuration - Springs or shims for adjustment

Joint Sealing

Protection Rating: - IP40: Basic dust protection - IP54: Enhanced dust, splash resistant - IP65: Full dust seal, water jet resistant

Seal Types: - Lip seals (elastomer) - Labyrinth seals (non-contact) - Combination for high protection

Mechanical Transmission

Harmonic Drives

Advantages: - Zero backlash - High reduction ratio (50:1 to 160:1) - Compact, coaxial design - Excellent for precision joints

Disadvantages: - Expensive - Finite life (flexspline fatigue) - Requires periodic inspection

Planetary Gearboxes

Advantages: - Lower cost than harmonic drives - High efficiency (>95%) - Robust and long-lasting

Disadvantages: - Some backlash (can be minimized with preload) - Larger package - Lower reduction ratios per stage

Timing Belts

Used for wrist joints: - Lightweight - Compliant (some shock absorption) - Low backlash with proper tension - Reduction via pulley diameter ratio

Direct Drive

No gearbox: - Zero backlash - High bandwidth control - Simplified mechanics

Requires: - Large, high-torque motors - More expensive - Lower reduction in inertia reflected to motor

Gravity Compensation

Purpose: Counteract gravity forces on links to reduce motor torque requirements.

Spring Balancing - Extension springs or gas springs - Connect between fixed base and moving link - Tune spring rate and position for balance

Counterweights - Masses added opposite load - Simple, no wear - Increases total moving mass and inertia

Active Compensation - Motor applies counter-torque based on joint angles - Requires continuous power - Most flexible, no added mechanical complexity

Example Robot Design

Small 6-Axis Desktop Robot

Specifications: - Reach: 500 mm - Payload: 2 kg - Repeatability: +/-0.05 mm

Link Lengths: - Upper arm (a2): 250 mm - Forearm (a3): 250 mm - Wrist length (d4): 150 mm

Motors: - J1, J2: NEMA 23 servo, 100W - J3, J4: NEMA 17 servo, 50W - J5, J6: NEMA 14 servo, 25W

Gearing: - J1, J2, J3: 100:1 harmonic drive - J4, J5, J6: 50:1 planetary or timing belt

Materials: - Links: 6061-T6 aluminum - Base: Steel plate and structural aluminum - Fasteners: Stainless steel

Next: 10.3 Joint Mechanisms and Actuators

References

1. **ISO 10218-1:2011** - Robots and robotic devices - Safety requirements
 2. **ISO 9283:1998** - Manipulating industrial robots - Performance criteria
 3. **Denavit, J. & Hartenberg, R.S. (1955).** "A Kinematic Notation for Lower-Pair Mechanisms." *ASME Journal of Applied Mechanics*, 22, 215-221
 4. **Craig, J.J. (2017).** *Introduction to Robotics: Mechanics and Control* (4th ed.). Pearson
 5. **Lynch, K.M. & Park, F.C. (2017).** *Modern Robotics*. Cambridge University Press
 6. **ABB Robot Studio Software** - Robot simulation and programming
 7. **KUKA System Software (KSS)** - Robot control and motion planning
-

Module 10 - Robotic Arm

Joint design determines robot performance, precision, and reliability. This section covers actuators, transmissions, bearings, and sensors for articulated robot joints.

Motor Selection

Servo Motors

AC Servo Motors - Three-phase brushless permanent magnet - High power density - Integrated encoder (typically 20-bit absolute) - Standard for industrial robots - Rated by continuous torque and peak torque - Typical sizes: 100W to 5kW per joint

DC Brushless Servo Motors - Similar performance to AC servos - Simpler drive electronics (for small systems) - Direct battery operation possible - Common in collaborative robots

Motor Specifications

Torque Requirements:

$$T_{motor} = (T_{load} \times GR) / \eta + T_{accel}$$

Where: - T_{load} = Static torque from gravity and forces - GR = Gear ratio - η = Transmission efficiency (0.85-0.95) - T_{accel} = Torque for acceleration

Example Calculation (Shoulder Joint): - Arm mass: 5 kg - Center of gravity: 300 mm from joint - Load: 2 kg at 500 mm - Static torque: $(5 \times 0.3 + 2 \times 0.5) \times 9.81 = 24.5 \text{ Nm}$ - Gear ratio: 100:1 - Required motor torque: $24.5/100/0.9 = 0.27 \text{ Nm}$ continuous

Add dynamic torque: - Desired acceleration: 2 rad/s² - Link inertia: $I = 0.5 \text{ kg m}^2$ - Dynamic torque: $0.5 \times 2 = 1 \text{ Nm}$ at joint - Motor torque: $1/100 = 0.01 \text{ Nm}$ - Total: $0.27 + 0.01 = 0.28 \text{ Nm}$

Select motor with 0.5-1.0 Nm continuous rating for margin.

Speed Requirements - Typical joint speeds: 90-180 deg/s - Motor speed = Joint speed × Gear ratio - Example: 180 deg/s × 100 = 18,000 deg/s = 3000 RPM

Thermal Considerations - RMS torque over duty cycle - Intermittent rating vs. continuous - Cooling (natural convection, forced air) - Derating for elevated temperatures

Stepper Motors

Applicability to Robots

Rarely used in articulated arms due to: - Lower torque-to-weight ratio - Step loss under high loads - No inherent position feedback - Lower efficiency

Acceptable for: - Small desktop robots (<1 kg payload) - Closed-loop steppers (with encoder) - Educational/hobby projects

Sizing - Safety factor: 3-5× for open-loop - Closed-loop steppers similar to servos

Gear Reducers

Harmonic Drives

Principle of Operation

Components: - Wave generator (elliptical cam with bearings) - Flexspline (thin flexible cup with external teeth) - Circular spline (rigid ring with internal teeth)

Operation: - Wave generator rotates inside flexspline - Flexspline deforms and meshes with circular spline - Two teeth difference creates reduction

Specifications

Reduction Ratios: - Common: 50:1, 80:1, 100:1, 120:1, 160:1 - Single-stage reduction

Backlash: - Typically <1 arcmin (0.017°) - Near-zero for precision applications

Efficiency: - 65-85% typical - Lower than planetary gears - Heat generation consideration

Torque Capacity: - Range: 0.5 Nm to 1000+ Nm - Rated and peak torque specifications

Life Expectancy: - 3,000 to 10,000 hours typical - Depends on load, speed, lubrication - Flexspline is wear component

Sizing

Safety Factor: - Rated torque = Peak load / 2 (minimum) - Account for shock loads and accelerations

Mounting: - Typically hollow bore (shaft through center) - Bolt pattern for motor and output - Compact package (diameter similar to motor)

Manufacturers - Harmonic Drive LLC (original patent holder) - Leaderdrive (Chinese alternative, lower cost) - Sumitomo (SHD series)

Planetary Gearboxes

Construction

Components: - Sun gear (motor input) - Planet gears (3-4, rotating around sun) - Ring gear (fixed housing) - Planet carrier (output)

Advantages: - Robust and reliable - High efficiency (95-98%) - Lower cost than harmonic drives
- Long service life

Disadvantages: - Higher backlash (3-10 arcmin typical) - Can be minimized with preload or dual-stage designs - Larger package for same reduction

Reduction Ratios - Single stage: 3:1 to 10:1 - Two-stage: 10:1 to 100:1 - Common: 10:1, 20:1, 50:1, 100:1

Applications in Robots - Base and shoulder joints (high torque, backlash acceptable) - Wrist joints (lighter loads, lower precision requirements) - Collaborative robots (lower speeds, compliance)

Manufacturers - Apex Dynamics - Neugart - Wittenstein - SEW-Eurodrive

Cycloidal Drives

Operation - Eccentric input rotates cycloidal disc - Disc engages with ring gear pins - High contact ratio (all teeth engaged)

Characteristics - Very low backlash (<1 arcmin) - High shock load capacity - Compact and robust
- Efficiency: 85-90%

Applications - Alternative to harmonic drives - Heavy-duty robots - Less common in standard designs

Timing Belt Reduction

Wrist Joint Applications

Advantages: - Lightweight - Remote motor mounting (reduces wrist inertia) - Low backlash with proper tension - Quiet operation

Design: - GT2 or GT3 profile belts - Aluminum or steel pulleys - Reduction via pulley diameter ratio - Example: 20-tooth drive, 100-tooth driven = 5:1

Tension and Backlash - Spring-loaded tensioner - Initial tension: 2-3% of belt strength - Check and adjust periodically - Backlash <0.1° with proper tension

Belt Selection - Width based on torque - Length to accommodate pulley spacing - Reinforced belts (steel or Kevlar cords)

Bearings

Types for Robot Joints

Crossed Roller Bearings

Construction: - Rollers arranged perpendicular to each other - Single compact unit - High rigidity in all directions

Advantages: - Extremely compact - High load capacity (radial, axial, moment) - Low friction - Excellent for rotary tables and joints

Disadvantages: - Expensive - Sensitive to contamination - Requires precision housing

Angular Contact Ball Bearings (Paired)

Configuration: - Two bearings in back-to-back or face-to-face arrangement - Preloaded to eliminate play

Advantages: - Lower cost than crossed roller - High speed capability - Wide availability

Applications: - Mid-size robot joints - Speed-critical applications

Tapered Roller Bearings

Characteristics: - Heavy-duty loads - Adjustable preload via shimming - Large robots (payloads >50 kg)

Deep Groove Ball Bearings

Use: - Light-duty joints - Where some play acceptable - Low cost

Preload

Purpose - Eliminate play and backlash - Increase stiffness - Improve positioning accuracy

Methods

Spring Preload: - Belleville washers or wave springs - Maintains preload despite wear or temperature - Moderate stiffness

Rigid Preload: - Shims or spacers set during assembly - Maximum stiffness - Requires precision assembly

Adjustment: - Too little: Play and vibration - Too much: Friction, heat, premature wear - Follow manufacturer specifications

Sealing

Protection Requirements - Dust and chips (machining environments) - Liquids (washdown, coolant splash) - Grease retention

Seal Types

Contact Seals: - Elastomer lip seals - Effective sealing - Some friction - Typical for IP54+

Non-Contact Seals: - Labyrinth or gap seals - Lower friction - Suitable for clean environments

Combination: - Primary labyrinth seal - Secondary lip seal - Best protection for harsh environments

Brakes

Purpose - Hold position when power removed (safety) - Prevent back-driving under gravity - Parking brake for transport/maintenance

Types

Electromagnetic Spring-Set Brakes: - Spring applies brake, electromagnet releases - Fail-safe (brake on power loss) - Integrated with motor or gearbox - Holding torque rated

Permanent Magnet Brakes: - Active control (PWM) to engage/release - No holding power required - More complex control

Sizing - Holding torque $\geq 1.5 \times$ maximum static joint torque - Thermal capacity for repeated operations - Response time (<100 ms typical)

Applications - Vertical joints (shoulder, elbow) - Safety requirement per ISO 10218 - Optional for horizontal joints (base rotation)

Encoders and Feedback

Encoder Types

Incremental Encoders

Operation: - Outputs pulses for motion - A and B channels (quadrature) for direction - Index pulse for homing reference

Resolution: - 1000-10,000 CPR (counts per revolution) typical - Higher resolution via quadrature (4x multiplication) - After gearing: Very high resolution at joint

Advantages: - Lower cost - Simple interface - High resolution available

Disadvantages: - Lose position on power loss - Requires homing sequence on startup - Susceptible to noise (pulse loss)

Absolute Encoders

Operation: - Outputs unique position code (no homing needed) - Retains position through power cycles - Multi-turn versions count full rotations

Protocols: - Parallel (individual wires per bit, rare) - Serial: SSI, BiSS, EnDat - Networked: EtherCAT, PROFINET

Resolution: - Single-turn: 12-20 bits (4096 to 1M positions/rev) - Multi-turn: Tracks 12-16 additional bits (4096-65536 turns)

Advantages: - No homing required - Reliable position data - Standard for industrial robots

Disadvantages: - Higher cost - More complex interfacing

Resolvers

Operation: - Analog sensor with sine/cosine outputs - Rotating transformer principle - Requires resolver-to-digital converter

Advantages: - Very robust (harsh environments) - Immune to electrical noise - Infinite resolution (analog)

Disadvantages: - Lower accuracy than optical encoders - Bulkier - Requires specialized interface

Applications: - Military, aerospace, harsh environments - Less common in modern industrial robots

Encoder Mounting

Motor-Mounted - Integral or attached to motor shaft - Measures motor position - Gear backlash and compliance not measured - Standard for servo motors

Joint-Mounted (Dual Encoder) - Second encoder at joint output - Measures actual joint position - Compensates for transmission errors - Used in high-precision systems

Calibration - Encoder zero relative to joint mechanical zero - Teach mode to define home position - Store offsets in controller non-volatile memory

Cable and Hose Routing

Internal Routing

Advantages: - Protected from environment - Clean appearance - No snagging on obstacles

Design Requirements: - Sufficient channel cross-section - Smooth bends (no sharp corners) - Support and strain relief - Access for replacement

External Routing

Cable Carriers (Drag Chains): - Manage cables through joint motion - Prevent twisting and strain - Sized for cable bundle diameter

Spiral Cable Wrap: - Flexible protective sleeving - Allows limited rotation - Simpler than drag chains

Rotary Joint Considerations

Continuous Rotation: - Slip rings for electrical signals - Rotary unions for pneumatic/hydraulic lines - Expensive, limit maintenance

Limited Rotation: - Cables routed to allow twist - Mechanical stops prevent over-rotation - Typical: $\pm 350^\circ$ for base joint - Wrist: Cables through hollow shafts and gearboxes

Cable Types

Power Cables: - Flexible stranded conductors - Oil-resistant insulation - Shield for EMI protection

Signal Cables: - Encoder: Shielded twisted pair - I/O: Multi-conductor - Network: Industrial Ethernet (Cat5e/6)

Pneumatic/Hydraulic: - Flexible hoses rated for pressure - Push-to-connect or threaded fittings - Routing to minimize bending stress

Joint Assembly Example

Shoulder Joint (Joint 2) for 5kg Payload Robot

Components: - Motor: 400W AC servo, 1.3 Nm continuous - Gearbox: 100:1 harmonic drive - Bearings: Crossed roller bearing, 150 mm OD - Encoder: 20-bit absolute, motor-mounted - Brake: 15 Nm holding torque - Housing: Aluminum, machined and welded

Assembly: 1. Install crossed roller bearing in housing 2. Mount harmonic drive to bearing outer race 3. Mount motor to harmonic drive input 4. Install brake between motor and gearbox 5. Route cables through hollow shaft 6. Seal housing with gasket and cover 7. Install encoder and calibrate zero position

Testing: - Friction test (motor current at slow speed) - Backlash measurement (<1 arcmin target) - Encoder functionality and calibration - Brake holding torque verification - Joint temperature under load

Performance Considerations

Backdrivability

High Ratio Gearboxes: - Harmonic drives with >100:1 difficult to backdrive - Planetary gears with <50:1 easily backdriven

Safety Implications: - Non-backdrivable requires brakes for safety - Backdrivable allows manual movement (teaching, collision recovery) - Collaborative robots typically backdrivable

Friction and Efficiency

Total Friction: - Bearing friction (seal type, preload) - Gear friction (tooth mesh, lubrication) - Cable drag

Efficiency Chain:

$$\eta_{\text{total}} = \eta_{\text{bearings}} \times \eta_{\text{gears}} \times \eta_{\text{seals}}$$

Example: $0.98 \times 0.85 \times 0.95 = 0.79$ (79%)

Low friction important for: - Energy efficiency - Heat generation - Backdrivability - Force control sensitivity

Inertia Matching

Optimal motor-to-load inertia ratio:

$$\text{Ratio} = J_{\text{load_reflected}} / J_{\text{motor}}$$

Ideal: 1:1 to 5:1 - Lower ratio: Better dynamic response - Higher ratio: Motor undersized or load too high

Reflected inertia:

$$J_{\text{reflected}} = J_{\text{load}} / GR^2$$

High gear ratios reduce reflected inertia, allowing smaller motors.

Next: 10.4 End Effectors and Tool Changers

References

1. **ISO 10218-1:2011** - Robots and robotic devices - Safety requirements
 2. **ISO 9283:1998** - Manipulating industrial robots - Performance criteria
 3. **Denavit, J. & Hartenberg, R.S. (1955).** "A Kinematic Notation for Lower-Pair Mechanisms." *ASME Journal of Applied Mechanics*, 22, 215-221
 4. **Craig, J.J. (2017).** *Introduction to Robotics: Mechanics and Control* (4th ed.). Pearson
 5. **Lynch, K.M. & Park, F.C. (2017).** *Modern Robotics*. Cambridge University Press
 6. **ABB Robot Studio Software** - Robot simulation and programming
 7. **KUKA System Software (KSS)** - Robot control and motion planning
-

Module 10 - Robotic Arm

End effectors are the interface between robot and task. This section covers grippers, process tools, sensors, and automatic tool changing systems.

Tool Mounting Interface

ISO 9409 Standard

Flange Specifications

Common sizes: - ISO 9409-1-31.5: 31.5 mm diameter (small robots) - ISO 9409-1-40: 40 mm diameter - ISO 9409-1-50: 50 mm diameter (most common) - ISO 9409-1-63: 63 mm diameter - ISO 9409-1-80: 80 mm diameter - ISO 9409-1-100: 100 mm diameter (large robots)

Features: - Bolt circle diameter - Centering pilot diameter - Dowel pin holes for orientation - Standard bolt pattern (M5, M6, M8)

Tool Coordinate System - Origin at center of flange - Z-axis perpendicular to flange (tool direction) - X and Y in flange plane

Custom Flanges

For non-standard applications: - Adapter plates to convert between sizes - Custom mounting for legacy tools - Quick-disconnect features - Alignment features (keys, pins)

Grippers

Parallel Jaw Grippers (From Module 9, Applied to Arms)

Pneumatic Parallel Grippers

Advantages for robotic arms: - Fast actuation (0.1-0.3 sec) - Simple control (on/off valve) - High grip force - Reliable and robust

Sizing: - Stroke: Part size variation \times 2 (jaw travel each side) - Force: Calculate per Module 9.3 - Typical sizes: 16mm, 25mm, 40mm, 63mm stroke

Integration: - Mount to tool flange via adapter plate - Pneumatic lines through robot arm - Solenoid valve at base or on gripper - Proximity sensors for open/closed feedback

Electric Parallel Grippers

Features: - Variable position and force control - Encoder feedback (jaw position) - Force sensing (current monitoring) - Slower than pneumatic

Advantages: - Adaptive grasping (different part sizes without tool change) - Force monitoring (detect part presence, prevent damage) - No compressed air required

Control: - Digital or analog signal for position - Current limit for force control - RS-485, EtherCAT, or CAN communication

Gripper Finger Design

Modular Jaws: - Quick-change interface on gripper body - Part-specific soft jaws - 3D printed for rapid prototyping - Machined aluminum for production

Materials: - Aluminum: General purpose, lightweight - Urethane: Soft, high friction, no part damage - Rubber: Conformable to shape - Steel: Heavy-duty, wear-resistant

Features: - V-grooves for round parts - Form-fitting profiles for shaped parts - Serrations for high friction - Vacuum integration (combination gripper)

Angular Grippers

Design: - Jaws pivot about axis - Suitable for round parts (internal or external grip) - More compact than parallel for same stroke

Applications: - Shaft handling - Tube gripping - Turning parts in assembly

Three-Jaw Grippers

Characteristics: - Self-centering (like lathe chuck) - Excellent for round parts - More complex mechanism - Typically pneumatic actuation

Vacuum Grippers for Robotic Arms

Suction Cup Arrays

Design: - Multiple cups on rigid or flexible base - Independent spring-loaded cups compensate for surface variation - Bellows cups for uneven surfaces

Advantages: - Handle large flat parts (sheet metal, panels, glass) - No grip force, gentle on parts - Fast pick/release

Vacuum Distribution: - Manifold distributes vacuum to all cups - Independent zones for different part sizes - Valves for zone control

Integrated Vacuum Generators

Venturi Ejector on Tool: - Compact, mounts near suction cups - Minimal tubing (reduces lag) - Only compressed air line to tool

Centralized Vacuum Pump: - Single pump serves multiple tools/robots - Vacuum reservoir for quick response - More energy efficient for continuous use

Magnetic Grippers

Electromagnets

Design: - Coil wrapped around iron core - Controllable via relay or PWM - Strong holding force when energized

Power Requirements: - Continuous power to maintain grip - Power through robot umbilical - Fail-safe requires backup power or safety catching device

Demagnetization: - Residual magnetism may prevent release - Active demagnetization cycle (reverse polarity pulse) - May interfere with machining or inspection

Permanent Magnet Systems

Switchable Permanent Magnets: - Mechanical or pneumatic actuation rotates magnet - No electrical power required - Robust and simple

Applications: - Steel part handling - Large sheet metal - Scrap handling

Process Tools

Welding Tools

Spot Welding Guns

Components: - Pneumatic or servo-electric actuation - Water-cooled electrodes - Transformer (if integrated) - Force control

Integration: - Heavy (10-25 kg typical) - High current and water lines - Force feedback for quality - Electrode wear compensation

Programming: - Approach position - Clamp force and weld time - Retract and move to next spot

Arc Welding Torches

MIG/MAG Torch: - Wire feed mechanism (may be robot-mounted or external) - Contact tip and nozzle - Gas shroud - Typically 3-6 kg

TIG Torch: - Tungsten electrode - Separate filler wire feed (if used) - Precision gas control

Control Integration: - Welding power source (separate unit) - Start/stop signals from robot - Wire feed speed control - Arc voltage and current monitoring - Seam tracking sensors (optional)

Cutting and Grinding Tools

Deburring Spindles

Specifications: - Speed: 10,000-60,000 RPM - Power: 0.5-3 kW - Air or electric drive - Collet or quick-change tooling

Force Control: - Constant force against surface - Adapt to part variation - Requires force/torque sensor

Applications: - Edge deburring after machining - Weld seam grinding - Surface finishing

Routing and Milling

High-Speed Spindle: - Similar to CNC spindle (see Module 6) - Lighter weight for robot mounting
- Rigid mounting required (robot stiffness limits)

Limitations: - Robot compliance reduces achievable tolerances - Best for non-precision material removal - Foam, composites, wood suitable - Metal milling requires extremely rigid robot

Dispensing Tools

Adhesive and Sealant Dispensers

Types: - Pneumatic pressure pots - Positive displacement pumps - Cartridge dispensers

Control: - On/off valves - Flow rate control - Pressure regulation

Programming: - Continuous path (bead along seam) - Dot patterns (adhesive dots) - Speed coordination (consistent bead width)

Spray Guns

Paint/Coating: - HVLP (High Volume Low Pressure) - Electrostatic for efficiency - Explosion-proof construction

Robot Requirements: - Smooth motion (avoid texture from motion artifacts) - Speed control (consistent coating thickness) - Rotational wrist for complex geometries

Inspection Tools

Touch Probes

CMM-Style Probes: - Contact trigger probe - 3-axis or 5-axis measurement - Repeatability: 0.01-0.02 mm

Operation: - Robot moves probe to surface - Contact triggers measurement - Record position and calculate dimensions

Limitations: - Robot repeatability limits accuracy - Suitable for go/no-go inspection - Not precision metrology

Vision Systems

End-Effector Mounted Camera: - Inspect parts at multiple angles - Defect detection - Barcode/OCR reading - Position verification

Integration: - Lighting (LED ring, structured light) - Image processing computer - Communication to robot controller

Force/Torque Sensors

Wrist-Mounted F/T Sensor: - Measures forces and torques in 6 axes - Mounts between flange and tool - Enables force-controlled tasks

Applications: - Assembly (insertion with force feedback) - Contour following (grinding, deburring)
- Contact detection - Product testing (button feel, latch engagement)

Tool Changers

Manual Tool Change

Simple Systems: - Bolt tool to flange manually - Disconnect utilities (air, electrical) - For infrequent changes or simple applications

Quick-Disconnect Couplings: - Pneumatic: Push-to-connect - Electrical: Threaded connectors or pogo pins - Reduces changeover time to 2-5 minutes

Automatic Tool Changers (ATC)

Master/Tool Side

Master Side (on robot): - Mechanical coupling (bayonet, tapered, or kinematic) - Pneumatic actuation for lock/unlock - Utility pass-throughs (air, electrical, data) - Pogo pins or inductive coupling for signals

Tool Side (on end effector): - Complementary coupling - Receptacles for utilities - Alignment features (pins, tapers)

Coupling Mechanisms

Bayonet Coupling: - Rotate and lock (quarter-turn) - High repeatability - Moderate force coupling

Tapered Coupling: - Pull tool into taper (Hirth coupling) - Very high stiffness and repeatability - Used in precision applications

Kinematic Coupling: - Three ball-and-groove pairs - Highest repeatability (+/-0.005 mm) - Expensive

Utility Pass-Through

Pneumatic: - Quick-disconnect valves in coupling - Multiple circuits (typically 2-4) - Pressure rating: 6-10 bar

Electrical: - Pogo pins (spring-loaded contacts) - Inductive coupling (contactless) - Power and signal lines (typically 6-16 circuits)

Data: - Ethernet (M12 connectors) - Serial communication - Tool identification (RFID or coded resistors)

Tool Rack

Design: - Holds multiple tools - Passive or active retention (locking mechanisms) - Organized layout for robot access - Typically 4-12 tool capacity

Tool Identification: - RFID tags on each tool - Robot reads tag to verify correct tool - Load tool-specific parameters (TCP offset, mass, etc.)

Change Sequence

Return Old Tool: 1. Move to tool rack position 2. Insert tool into holder 3. Unlock coupling (pneumatic signal) 4. Retract robot 5. Verify tool released (sensor)

Pick New Tool: 1. Move to new tool position 2. Engage coupling 3. Lock coupling (pneumatic signal) 4. Verify locked (sensor) 5. Load tool parameters (TCP, weight) 6. Move to ready position

Cycle Time: - Typical: 5-15 seconds per tool change - Add time to move between rack and work area

Tool Changer Sizing

Payload Capacity - Static payload: Tool weight - Dynamic payload: Tool plus part plus forces - Safety factor: 2-3x

Moment Capacity - Torque from tool extending from flange - Example: 5 kg tool at 150 mm = 7.5 Nm moment - Changer rated for moment loading

Repeatability - Critical for precision applications - Typical: +/-0.01-0.05 mm - Kinematic coupling: +/-0.005 mm

Manufacturers - ATI Industrial Automation - Schunk - Destaco - Stäubli - Zimmer Group

Tool Design Considerations

Weight and Center of Gravity

Minimize Weight: - Reduce motor torque requirements - Increase speed and acceleration - Extend robot reach capability

Balance: - Center tool mass along wrist axis - Reduce moments on wrist joints - Counterweights if necessary

Tool Center Point (TCP) Definition

TCP Location: - Functional point of tool (gripper center, torch tip, probe ball) - Defined relative to flange coordinate system - Entered in robot controller

TCP Calibration: - Four-point method: Touch reference point from four different robot orientations - Controller calculates TCP position - Accuracy critical for path precision

Reach Extension

Trade-offs: - Longer tool extends reach - Increases moment on wrist - Reduces effective payload - May increase deflection and vibration

Optimize: - Minimize tool length while meeting functional requirements - Use lightweight materials (carbon fiber, aluminum)

Collision Geometry

Robot Path Planning: - Software needs tool collision geometry - Define as simple shapes (cylinders, boxes) - Conservative envelope for safety - Allows collision-free path generation

Cable Management

Routing: - Through or along robot arm - Strain relief at wrist - Protection from abrasion and snagging

Connectors: - Secure and reliable - Industrial grade (M12, M8 for sensors) - Polarized to prevent miswiring

Multi-Function End Effectors

Combination Tools

Gripper + Sensor: - Gripper with integrated force sensor - Camera on gripper for visual feedback
- Vacuum and mechanical grip combined

Welding + Gripper: - Pick part, place in fixture, weld - Reduces tool changes - Complex integration

Tradeoffs - Increased complexity - Higher weight - More failure modes - Justified if reduces cycle time

Next: 10.5 Motion Planning and Control

References

1. ISO 10218-1:2011 - Robots and robotic devices - Safety requirements
 2. ISO 9283:1998 - Manipulating industrial robots - Performance criteria
 3. Denavit, J. & Hartenberg, R.S. (1955). "A Kinematic Notation for Lower-Pair Mechanisms." *ASME Journal of Applied Mechanics*, 22, 215-221
 4. Craig, J.J. (2017). *Introduction to Robotics: Mechanics and Control* (4th ed.). Pearson
 5. Lynch, K.M. & Park, F.C. (2017). *Modern Robotics*. Cambridge University Press
 6. ABB Robot Studio Software (KSS) - Robot simulation and programming
 7. KUKA System Software (KSS) - Robot control and motion planning
-

Module 10 - Robotic Arm

Motion planning determines how the robot moves between positions efficiently, smoothly, and safely while respecting joint limits, avoiding collisions, and achieving desired tool paths.

Control Architecture

Hierarchical Control Structure

Task Level: - High-level goals (pick part, weld seam, assemble component) - Sequence of operations - Decision logic

Path Planning Level: - Generate trajectory (positions over time) - Collision checking - Optimization (time, smoothness, energy)

Trajectory Generation Level: - Convert path to joint space commands - Velocity and acceleration profiling - Interpolation between waypoints

Servo Control Level: - PID or advanced control loops - Individual joint control - Update rate: 1-4 kHz typical

Coordinate Space Selection

Joint Space Motion

Characteristics - Each joint moves independently from start to end angle - Path in Cartesian space not controlled (tool may follow curved path) - Simple inverse kinematics (only at endpoints) - Fastest motion between points

Applications - Pick and place (path doesn't matter, only start/end) - Retract motions - Moving to home position

Programming

```
MOVEJ P1 SPEED=50% ; Joint space move to position 1
```

Advantages - No singularity issues - Fastest execution - Simple computation

Disadvantages - Tool path unpredictable - May cause collisions if obstacles present

Cartesian Space Motion (Linear)

Characteristics - Tool moves in straight line from start to end - Constant orientation (or linearly interpolated) - Requires inverse kinematics at every timestep - Smooth, predictable tool path

Applications - Welding seams - Gluing beads - Cutting paths - Assembly approach/retract

Programming

```
MOVEL P1 SPEED=500 mm/s ; Linear move to position 1
```

Interpolation

Position:

$$P(t) = P_{\text{start}} + (P_{\text{end}} - P_{\text{start}}) \times t$$

Where t = normalized time (0 to 1)

Orientation: - SLERP (Spherical Linear Interpolation) for quaternions - Ensures smooth rotation

Challenges - May encounter singularities along path - Joint velocities may exceed limits near singularities - Computationally intensive (continuous inverse kinematics)

Circular Motion

Application - Arc welding - Rounded corners - Smooth blending

Definition - Three points define circle (start, via, end) - Or center, start, angle

Programming

MOVEC VIA_P, END_P SPEED=300 mm/s ; Circular move through via point to end

Interpolation - Calculate circle parameters from three points - Generate points along arc - Inverse kinematics for each point

Trajectory Generation

Velocity Profiling

Trapezoidal Profile

Phases: 1. Acceleration (constant a) 2. Constant velocity (v_{max}) 3. Deceleration (constant a)

If distance too short for v_{max} : - Triangular profile (accelerate then immediately decelerate)

S-Curve Profile (Smoothen)

Jerk-Limited: - Acceleration increases smoothly - Reduces vibration and wear - Longer motion time than trapezoidal

Phases: 1. Jerk (increasing acceleration) 2. Constant acceleration 3. Jerk (decreasing acceleration) 4. Constant velocity 5. (Reverse for deceleration)

Multi-Joint Coordination

Synchronized Motion: - Scale velocities so all joints finish simultaneously - Smooth coordinated motion

Master-Slave: - One joint (master) runs at full capability - Others (slaves) scaled to finish at same time

Time-Optimal Trajectory

Objective: Minimize motion time while respecting limits

Constraints: - Joint velocity limits: $\dot{\theta}_i \leq v_{max,i}$ - Joint acceleration limits: $\ddot{\theta}_i \leq a_{max,i}$
- Torque limits (dynamic)

Algorithm: - Phase plane method - Dynamic programming - Numerical optimization

Result: - Each joint runs at limit for portions of trajectory - Complex velocity profiles - Typically 10-30% faster than simple profiling

Path Planning

Point-to-Point Planning

Simple Case: - Start and end positions only - No intermediate constraints - Joint space or Cartesian linear

Obstacle Avoidance via Via Points

Define intermediate positions:

```
MOVEJ HOME  
MOVEJ APPROACH_1  
MOVEL PICK_POS  
MOVEL APPROACH_1  
MOVEJ APPROACH_2  
MOVEL PLACE_POS
```

Via points guide robot around obstacles.

Continuous Path Planning

For Welding, Cutting, Dispensing

Requirements: - Smooth, continuous tool path - Constant orientation or programmed rotation - Velocity control (affects process quality)

Spline Interpolation

Cubic Splines: - Smooth curve through waypoints - Continuous velocity and acceleration - Adjustable tension/smoothness

B-Splines: - Curve influenced by but doesn't pass through all control points - Smoother than cubic splines - Used in CAD/CAM path generation

Path Smoothing (Blending)

Sharp Corners: - Stop at waypoint, change direction (slow)

Blended Corners: - Round corner with specified radius - Don't stop at waypoint - Faster, smoother

```
MOVEL P1 BLEND=10mm ; Blend within 10mm of P1  
MOVEL P2 BLEND=10mm  
MOVEL P3 ; Stop at P3
```

Collision Avoidance

Static Obstacles

Known Environment: - Pre-defined keep-out zones - Robot model includes tool geometry - Path planner checks all configurations along path

Methods: - Swept volume calculation - Discrete configuration checking - Distance field methods

Dynamic Collision Checking

Real-Time Monitoring: - Safety-rated laser scanners - Vision-based workspace monitoring - Detect humans or moving obstacles - Slow or stop robot as needed

Self-Collision

Links Interfering: - Possible in complex poses or long tools - Check during path planning - Joint limits may prevent some self-collisions

Advanced Path Planning

Sampling-Based Planners

RRT (Rapidly-Exploring Random Tree): - Randomly sample configurations - Build tree from start toward goal - Fast, probabilistically complete - Used in complex environments

PRM (Probabilistic Roadmap): - Build graph of collision-free configurations - Query for paths between start/goal - Pre-computation for known environment

Optimization-Based

Minimize Cost Function: - Path length - Smoothness (jerk) - Energy consumption - Joint wear

Subject to Constraints: - Collision-free - Kinematic limits - Dynamic limits

Control Algorithms

PID Control (Per Joint)

Structure

$$u(t) = K_p \times e(t) + K_i \times \int e(t) dt + K_d \times de(t)/dt$$

Where: - $e(t)$ = error (desired - actual position) - K_p = proportional gain - K_i = integral gain - K_d = derivative gain - $u(t)$ = control output (torque command)

Tuning

Proportional (K_p): - Higher K_p : Faster response, stiffer - Too high: Oscillation, instability

Integral (K_i): - Eliminates steady-state error - Compensates for gravity, friction - Too high: Over- overshoot, oscillation

Derivative (K_d): - Damping (resists velocity) - Smooths response - Sensitive to noise

Methods: - Ziegler-Nichols - Manual tuning - Auto-tuning algorithms (commercial controllers)

Feed-Forward

Add Gravity Compensation:

$$u = u_{PID} + u_{gravity}$$

Gravity torque calculated from: - Link masses and centers of gravity - Current joint angles (from kinematics)

Reduces PID error, improves tracking.

Advanced Control

Computed Torque Control

Full Dynamics Model:

$$\tau = M(\theta)\dot{\theta} + C(\theta, \dot{\theta})\ddot{\theta} + G(\theta)$$

Where: - $M(\theta)$ = inertia matrix - $C(\theta, \dot{\theta})$ = Coriolis and centrifugal forces - $G(\theta)$ = gravity vector

Linearizes system, allows aggressive control.

Requires: - Accurate robot model (mass, inertia) - High computational power - Typically implemented in high-end controllers

Adaptive Control

Parameters Adjust Online: - Payload changes (gripper empty vs. holding part) - Wear and friction changes - Temperature effects

Methods: - Model reference adaptive control (MRAC) - Self-tuning regulators

Impedance Control

Target Behavior:

$$M(\ddot{x}_{actual} - \ddot{x}_{desired}) + B(\dot{x}_{actual} - \dot{x}_{desired}) + K(x_{actual} - x_{desired}) = F_{external}$$

Mimics mass-spring-damper system: - M = virtual mass - B = virtual damping - K = virtual stiffness - $F_{external}$ = measured external force

Applications: - Assembly (compliant insertion) - Contact tasks (deburring, polishing) - Human-robot interaction (safe, compliant)

Requires: - Force/torque sensor - High update rate control loop

Singularity Handling

Detection

Jacobian Matrix: - Maps joint velocities to TCP velocities - Singular when determinant near zero

Condition Number: - Measure of proximity to singularity - High value indicates near-singular

Avoidance Strategies

Path Planning: - Route around singular configurations - Add via points to guide path

Joint Limits: - Restrict joint ranges to exclude singularities - Example: Limit wrist joint to 10° to 170° (avoid 0° and 180°)

Damped Least Squares

Numerical Method: - Regularize inverse kinematics near singularities - Allows motion through singularity with small path deviation

Trade-off: - Path accuracy vs. singularity handling

Real-Time Considerations

Control Loop Frequency

Typical Rates: - Position loop: 1-4 kHz - Velocity loop: 10-20 kHz (in drive) - Current loop: 20-50 kHz (in drive)

Requirements: - Deterministic timing (real-time OS) - Sufficient computational power - Low-latency communication (EtherCAT, etc.)

Communication Architecture

Master-Slave: - Robot controller (master) sends commands - Servo drives (slaves) execute - Synchronous updates

Protocols: - EtherCAT: 1 kHz to 10 kHz cycle rate - PROFINET IRT: Similar performance - CAN bus: Lower performance, simpler

Trajectory Buffering

Look-Ahead: - Controller buffers upcoming trajectory - Allows velocity optimization through corners - Prevents starvation (waiting for next command)

Streaming: - External PC sends trajectory points continuously - Controller interpolates and executes - Used for complex paths (CAM-generated)

Calibration

Kinematic Calibration

Sources of Error: - Link length tolerances - Joint zero position errors - Encoder mounting misalignment - Compliance and deflection

Measurement: - Laser tracker or CMM - Measure TCP position at many configurations - Optimize kinematic parameters to minimize error

Improvement: - Accuracy: 0.5mm ± 0.1mm typical - Depends on robot quality and calibration effort

TCP Calibration

Four-Point Method: 1. Define fixed reference point (sharp tool against stationary pin) 2. Touch reference from four robot orientations 3. Controller solves for TCP position (intersection of four spheres)

Automated: - Touch probe on robot gripper - Probe known artifact (sphere or precision plane) - Controller calculates TCP

Payload Identification

For Gravity Compensation: - Mount known payload (calibration weight) - Move through various poses - Measure joint currents/torques - Identify payload mass and center of gravity

Improves: - Tracking accuracy - Energy efficiency - Torque headroom

Next: 10.6 Force Control and Compliance

References

1. **ISO 10218-1:2011** - Robots and robotic devices - Safety requirements
 2. **ISO 9283:1998** - Manipulating industrial robots - Performance criteria
 3. **Denavit, J. & Hartenberg, R.S. (1955).** "A Kinematic Notation for Lower-Pair Mechanisms." *ASME Journal of Applied Mechanics*, 22, 215-221
 4. **Craig, J.J. (2017).** *Introduction to Robotics: Mechanics and Control* (4th ed.). Pearson
 5. **Lynch, K.M. & Park, F.C. (2017).** *Modern Robotics*. Cambridge University Press
 6. **ABB Robot Studio Software** - Robot simulation and programming
 7. **KUKA System Software (KSS)** - Robot control and motion planning
-

Module 10 - Robotic Arm

Force control enables robots to interact with the environment through regulated contact forces, essential for assembly, finishing, and collaborative operations.

Force Sensing

Wrist-Mounted Force/Torque Sensors

Six-Axis F/T Sensor

Measurement Axes: - Three forces: Fx, Fy, Fz - Three torques: Tx, Ty, Tz - All referenced to sensor coordinate frame

Construction: - Strain gauges on flexure element - Wheatstone bridge circuits - Temperature compensation - Overload protection (mechanical stops)

Specifications: - Force range: +/-10N to +/-1000N (typical) - Torque range: +/-1Nm to +/-100Nm
- Resolution: 0.1% to 0.01% of full scale - Accuracy: 0.5% to 2% of full scale - Sample rate: 1-10 kHz

Mounting: - Between robot flange and tool - Adds 10-30mm height - Weight: 0.2-2 kg depending on capacity - Calibration matrix provided by manufacturer

Signal Processing

Raw Sensor Data: - Six voltage channels (Fx, Fy, Fz, Tx, Ty, Tz) - Amplified and digitized - Transmitted via analog, serial, or Ethernet

Calibration: - Converts voltages to force/torque values - Accounts for cross-talk between axes - Temperature compensation

Filtering: - Low-pass filter (10-100 Hz typical) - Remove sensor noise and vibration - Balance responsiveness vs. noise

Bias Removal: - Zero sensor with tool mounted (no contact) - Subtract tool weight and inertia - Leaves only external contact forces

Manufacturers - ATI Industrial Automation (most common) - OnRobot - Schunk - Robotiq

Joint Torque Sensing

Direct Torque Measurement

Methods: - Strain gauges on joint shafts - Measure torsional deflection - Convert to torque

Advantages: - No external sensor needed - Measures actual joint torque - Can detect collisions anywhere on robot

Disadvantages: - Complex installation - Temperature sensitivity - Calibration challenges

Motor Current Sensing

Indirect Force Measurement: - Motor current proportional to torque - Already measured by servo drives - No additional hardware

Calculation:

$$\tau_{\text{joint}} = K_t \times I_{\text{motor}} \times GR$$

Where: - K_t = motor torque constant - I_{motor} = motor current - GR = gear ratio

Limitations: - Includes friction and inertia (not just external forces) - Less accurate than dedicated sensors - Requires good dynamic model

Model-Based Estimation:

$$\tau_{\text{external}} = \tau_{\text{measured}} - \tau_{\text{model}}$$

Subtract expected torques from model, leaving external forces.

Collaborative Robots

Integrated Torque Sensing: - All joints have torque sensors - Safety-rated monitoring - Detect contact forces throughout workspace - Examples: KUKA LBR iiwa, Franka Emika, ABB GoFa

Force Control Strategies

Position Control with Force Limit

Simple Force Limiting

Operation: - Normal position control - Monitor contact force - Stop or retract if force exceeds limit

Example - Assembly:

```
move_to(insertion_approach)
while force_z < max_force:
    move_down(speed=10 mm/s)
    if force_z > max_force:
        retract()
        signal_error()
        break
```

Applications: - Collision detection - Protect parts from damage - Simple contact tasks

Hybrid Position/Force Control

Concept

Control Different Axes Independently: - Position control on some axes - Force control on others - Selection matrix defines which

Example - Surface Following: - X, Y: Position control (follow programmed path) - Z: Force control (maintain contact force)

Implementation

Selection Matrix S:

```
S = diag([1, 1, 0, 1, 1, 1])
```

Where 1 = position control, 0 = force control.

Control Law:

$$\text{Command} = S \times \text{Position_Control} + (I - S) \times \text{Force_Control}$$

Application - Deburring

Setup: - Tool follows edge path (X, Y position controlled) - Z-axis force controlled (press against surface) - Adapt to surface height variation

Programming:

```
SET_FORCE_CONTROL(Z_AXIS, TARGET_FORCE=20N, STIFFNESS=LOW)
MOVE START_POS
ENABLE_FORCE_CONTROL()
MOVE END_POS SPEED=50mm/s
DISABLE_FORCE_CONTROL()
```

Impedance Control

Mass-Spring-Damper Model

Target Behavior:

$$F_{\text{external}} = M \times (\ddot{x} - \ddot{x}_{\text{desired}}) + B \times (\dot{x} - \dot{x}_{\text{desired}}) + K \times (x - x_{\text{desired}})$$

Parameters: - M: Virtual mass (inertia) - B: Damping coefficient - K: Stiffness - F_external: Measured external force

Physical Interpretation

High Stiffness (K): - Resists displacement from target - Similar to position control - Good for precise positioning

Low Stiffness: - Compliant, easily displaced - Safe for contact - Good for assembly, human interaction

High Damping (B): - Resists velocity - Smooth, controlled motion - Prevents oscillation

Low Mass (M): - Responsive to forces - Quick adaptation

Tuning

Typical Values (Cartesian axes): - K: 100-5000 N/m - B: 10-500 Ns/m - M: 0.1-10 kg (virtual)

Adjustment: - Start with low stiffness and high damping - Increase stiffness for precision - Reduce damping if too sluggish

Applications

Assembly: - Compliant insertion (peg-in-hole) - Chamfer finding - Snap-fit operations

Collaborative Operation: - Robot yields to human push - Safe, intuitive interaction

Admittance Control

Concept

Opposite of Impedance: - Measure force - Calculate desired motion - Position control to achieve motion

Admittance:

$$\ddot{x} = (1/M) \times (F_{\text{external}} - B\dot{x} - Kx)$$

Suitable When: - Position control already excellent (stiff robot) - Add compliance via software - No need to modify low-level controller

Implementation

In control loop: 1. Read force sensor 2. Calculate desired velocity from force 3. Integrate to get position 4. Send position command to robot

Passive Compliance

Mechanical Compliance Devices

Remote Center Compliance (RCC)

Design: - Flexure-based mechanism - Allows lateral and angular deflection - Spring returns to center when released

Function: - Enables peg-in-hole insertion despite misalignment - No sensors or active control - Robust and reliable

Specifications: - Lateral compliance: 0.5-5 mm typical - Angular compliance: 1-5 degrees - Center of compliance at tool tip (peg end)

Applications: - High-speed assembly - Screw insertion - Pin insertion

Elastic Buffers

Simple Springs or Rubber: - Absorb impacts - Reduce peak forces - Protect robot and part

Design: - Mount between flange and tool - Tune stiffness for application - May include damping

Passive vs. Active Compliance

Passive (RCC, Springs)

Advantages: - No power or sensors required - Very fast response (mechanical) - Reliable, no software bugs - Lower cost

Disadvantages: - Fixed compliance (not adjustable) - No force feedback or data logging - Limited to specific tasks

Active (Force Control)

Advantages: - Programmable compliance - Force data available - Adaptive to different tasks - More sophisticated behavior

Disadvantages: - Requires force sensor - Complex software - Limited by control loop bandwidth - Higher cost

Hybrid Approach

Combination: - Passive compliance for fast, local corrections - Active control for overall force regulation - Best of both worlds

Applications of Force Control

Assembly

Peg-in-Hole Insertion

Challenges: - Tight clearances (0.05-0.2 mm) - Jamming if misaligned - Position control alone insufficient

Force Control Strategy: 1. Approach hole with position control 2. Switch to compliant mode (low stiffness) 3. Search for hole (spiral or raster pattern) 4. Detect insertion (force drop) 5. Insert with force limit 6. Detect full insertion (force rise or distance)

Spiral Search:

```
set_compliance(lateral_stiffness=100, vertical_stiffness=500)
radius = 0
while not inserted():
    angle += 10 degrees
    radius += 0.1 mm
    move_to(center + radius * [cos(angle), sin(angle)])
    if force_z < threshold:
        # Hole found
        break
```

Snap-Fit Assembly

Process: 1. Align parts with force feedback 2. Apply increasing force 3. Detect snap (force spike then drop) 4. Verify engagement

Force Profile: - Gradual increase - Spike at snap point (10-50N typical) - Lower holding force after snap

Press-Fit

Controlled Force Application: - Monitor force vs. displacement - Ensure part fully seated - Detect anomalies (cross-threading, missing parts)

Quality Assurance: - Log force profile - Compare to reference - Accept/reject based on criteria

Surface Finishing

Deburring

Setup: - Grinding or cutting tool on robot - Force control maintains contact - Follow edge path

Control: - Target force: 10-50N (depends on material and tool) - Adapt to surface variations - Constant material removal rate

Polishing

Requirements: - Consistent surface pressure - Smooth, continuous motion - Overlapping tool paths

Force Control: - Light force (5-20N) - Compliant to follow contours - May use compliant backing pad (passive)

Challenges

Surface Variations: - Height changes require Z-axis compliance - Curvature requires 3D path following - Irregular shapes benefit from force control

Robot Stiffness: - Industrial robots less stiff than dedicated machines - Force control partially compensates - Best results with stiff robots and light forces

Contact Inspection

Product Testing

Button Feel: - Press button with controlled force - Measure displacement and force profile - Verify tactile response (click, resistance)

Latch Testing: - Engage and disengage latches - Measure actuation force - Detect proper engagement

Dimensional Probing

Touch Probe with Force Control: - Approach surface slowly - Detect contact (force threshold) - Record position - Compliant approach prevents damage

Implementation Considerations

Control Loop Architecture

Rate Requirements

Force Sensor Sampling: - 1-10 kHz typical - Higher rate for stiff contact

Force Control Update: - 100-1000 Hz typical - Match or exceed position control rate - Faster = better stability and responsiveness

Software Structure

Real-Time Layer: - Force sensor reading - Control law calculation - Position command output - Deterministic timing

Non-Real-Time Layer: - Task planning - User interface - Data logging - Can tolerate latency

Stability

Control Stability

Potential Issues: - Oscillation from high gains - Instability from time delays - Contact/release transitions

Tuning: - Start with low gains (compliant) - Increase gradually - Test with actual contact conditions

Contact Stability

Hard Contact: - Stiff environment (metal-to-metal) - Requires lower control gains or higher damping - Prevent chattering

Soft Contact: - Compliant materials (foam, rubber) - More forgiving - Higher gains possible

Calibration

Force Sensor Zero

Procedure: 1. Mount tool on robot 2. Move to known orientation (typically vertical) 3. Capture force reading (tool weight) 4. Store as bias offset 5. Subtract from all future readings

Gravity Compensation: - Account for tool weight at any robot orientation - Calculate from tool mass, center of gravity, and orientation - Subtract from sensor reading

Tool Center Point

Force Application Point: - For torque calculations, need TCP position relative to sensor - Enter TCP offset in controller - Affects torque interpretation

Safety Considerations

Force Limits

Maximum Safe Force: - Depends on application and hazards - Collaborative robots: ISO/TS 15066 limits (see Section 10.9) - Industrial applications: Risk assessment

Monitoring: - Continuous force monitoring - Threshold triggers stop or retract - Safety-rated implementation for collaborative mode

Fault Detection

Sensor Failures: - Out-of-range readings - Communication loss - Inconsistent data

Response: - Disable force control - Revert to position control or stop - Alert operator

Next: 10.7 Programming and Simulation

References

1. **ISO 10218-1:2011** - Robots and robotic devices - Safety requirements
 2. **ISO 9283:1998** - Manipulating industrial robots - Performance criteria
 3. **Denavit, J. & Hartenberg, R.S. (1955).** "A Kinematic Notation for Lower-Pair Mechanisms." *ASME Journal of Applied Mechanics*, 22, 215-221
 4. **Craig, J.J. (2017).** *Introduction to Robotics: Mechanics and Control* (4th ed.). Pearson
 5. **Lynch, K.M. & Park, F.C. (2017).** *Modern Robotics*. Cambridge University Press
 6. **ABB Robot Studio Software** - Robot simulation and programming
 7. **KUKA System Software (KSS)** - Robot control and motion planning
-

Module 10 - Robotic Arm

Robot programming defines motion sequences, process parameters, and decision logic. This section covers teach pendant programming, offline programming, simulation, and common robot languages.

Programming Methods

Teach Pendant Programming

Hardware

Teach Pendant: - Handheld device with LCD screen - Directional jog buttons - Numeric keypad - Function keys - Three-position enabling switch (dead-man)

Connection: - Wired to robot controller - Wireless options (less common for safety)

Teaching Process

1. Enter Manual Mode:
 - Key switch to “Teach” or “Manual”
 - Reduces robot speed (typically 250 mm/s max per ISO)
 - Enabling switch must be held
2. Jog Robot:
 - Joint mode: Move individual joints
 - World mode: Move in X, Y, Z of world frame
 - Tool mode: Move in tool coordinate system
 - Adjust speed (1-100%)
3. Record Positions:
 - Move to desired position
 - Press “Teach” or “Record Point”
 - System saves position (all joint angles or TCP pose)

- Assign point name or number
4. Build Program:
 - Select motion type (joint, linear, circular)
 - Select taught points
 - Set speed and acceleration
 - Add I/O commands (gripper, process signals)
 5. Test and Refine:
 - Run program at low speed
 - Verify positions and motions
 - Adjust as needed
 - Gradually increase speed

Advantages - Intuitive, no programming knowledge needed - Immediate feedback (see actual robot motion) - Easy to adjust positions

Disadvantages - Time-consuming (robot occupied during teaching) - Production downtime - Difficult for complex logic or calculations - Ergonomically challenging for large programs

Offline Programming (OLP)

Concept

Program Development: - Create program on PC away from robot - 3D CAD environment - Simulate and verify before deployment - Download to robot for execution

Workflow

1. Import CAD Models:
 - Robot model (from manufacturer)
 - Workcell layout (fixtures, machines, obstacles)
 - Part geometry
2. Define Coordinate Systems:
 - Robot base frame
 - Part frame
 - Tool frame (TCP)
3. Create Path:
 - Manual point teaching (click in 3D)
 - Import from CAD (edges, surfaces)
 - Automatic path generation (tool paths for machining)
4. Add Process Parameters:
 - Speed, acceleration
 - I/O signals
 - Tool changes
 - Wait commands
5. Simulate:
 - Visualize robot motion
 - Check for collisions
 - Verify cycle time
 - Optimize trajectory
6. Post-Process:

- Convert generic path to robot-specific code
 - Generate native robot language (KRL, RAPID, etc.)
7. Download and Test:
- Transfer program to robot controller
 - Test at low speed
 - Fine-tune as needed

Software Packages

Commercial: - RoboDK: Multi-brand, affordable (\$500-\$3000) - ABB RobotStudio: Free for ABB robots - KUKA Sim Pro: For KUKA robots - Fanuc RoboGuide: For Fanuc robots - Delmia (Dassault): High-end, expensive - Process Simulate (Siemens): High-end

Open-Source: - ROS MoveIt: Powerful, requires programming - OpenRAVE: Research-oriented

Advantages - No production downtime - Complex paths (CAD-driven) - Collision detection before deployment - Multiple robot brands from single environment - Optimization tools

Disadvantages - Calibration differences (simulation vs. real robot) - Learning curve - Software cost (for commercial packages) - Final on-robot testing still required

Text-Based Programming

Robot-Specific Languages

KUKA Robot Language (KRL)

```
DEF MyProgram()
  PTP HOME Vel=100% DEFAULT
  LIN P1 CONT Vel=1.5 m/s TOOL[1] BASE[0]
  LIN P2 CONT Vel=1.5 m/s
  LIN P3 Vel=0.5 m/s
  WAIT SEC 1.0
  PTP HOME Vel=100%
END
```

Features: - Pascal-like syntax - Data types: INT, REAL, BOOL, POS, FRAME - Loops, conditionals, functions - Inline motion and I/O

ABB RAPID

```
PROC Main()
  MoveJ Home, v1000, z50, Tool0;
  MoveL p10, v500, z10, Tool1\WObj:=wobj1;
  MoveL p20, v500, z10, Tool1;
  WaitTime 0.5;
  SetDO D0_Gripper, 1;
  MoveJ Home, v1000, z50, Tool0;
ENDPROC
```

Features: - Structured, modern syntax - Strong typing - Built-in motion commands - Interrupt handling

Fanuc KAREL

```

PROGRAM MyProgram
VAR
    status : INTEGER
BEGIN
    MOVE_TO(1, 100) -- Move to position 1 at 100mm/s
    WAIT_FOR(DI[1])
    SET_DO(1, ON)
    DELAY(500) -- 500ms
    MOVE_TO(2, 200)
END MyProgram

```

And TP (Teach Pendant) language: - Graphical point-and-click style - Limited text editing

Universal Robots URScript

```

def my_program():
    movej(home_pose, a=1.4, v=1.05)
    movel(p[0.3, 0.2, 0.4, 0, 3.14, 0], a=1.2, v=0.25)
    set_digital_out(0, True)
    sleep(0.5)
    movel(p[0.3, 0.2, 0.5, 0, 3.14, 0], a=1.2, v=0.25)
end

```

Features: - Python-like syntax - Simple and accessible - Less powerful than industrial languages

Advantages of Text Programming - Version control (git) - Copy/paste, find/replace - Complex logic and calculations - Automated code generation

Disadvantages - Requires programming knowledge - Less intuitive than teach pendant - Syntax varies by manufacturer

High-Level Languages

Python with Robot Framework

ROS (Robot Operating System):

```

import rospy
from moveit_commander import MoveGroupCommander

rospy.init_node('robot_control')
arm = MoveGroupCommander('manipulator')

# Move to named pose
arm.set_named_target('home')
arm.go()

# Move to Cartesian position
pose_target = arm.get_current_pose().pose
pose_target.position.z += 0.1
arm.set_pose_target(pose_target)
arm.go()

```

Advantages: - Powerful Python ecosystem - Works with multiple robot brands - Advanced features (motion planning, perception)

Disadvantages: - Setup complexity - Not real-time capable (use with caution)

C++ with Robot SDK

Some manufacturers provide C++ libraries: - More control than scripting - Real-time capable - Complex integration

Coordinate Systems and Frames

Frame Definitions

Base Frame - Fixed to robot mounting surface - Default reference for all motion

World Frame - User-defined external reference - May differ from base (robot mounted at angle, on rail, etc.)

Tool Frame - Origin at TCP (Tool Center Point) - Z-axis typically points out of tool - X and Y in tool plane

Work Object (Part) Frame - Fixed to workpiece or fixture - Allows programming relative to part - Robot motion compensates for part orientation

Frame Transformations

Teaching Frames

Three-Point Method (Work Object): 1. Teach origin point 2. Teach point on X-axis 3. Teach point in XY-plane 4. Controller calculates frame transformation

Four-Point Method (Tool TCP): - Touch reference point from four orientations - Controller solves for TCP position

Frame Math

Transform point from tool frame to base frame:

$$P_{\text{base}} = T_{\text{base_tool}} \times P_{\text{tool}}$$

Where T is 4x4 homogeneous transformation matrix.

Programming with Frames

Advantage: - Part programs independent of robot position - Move fixture □ update frame □ program unchanged

Example (ABB RAPID):

```
MoveL Offs(p10, 50, 0, 0), v100, fine, Tool1\WObj:=Part1;
```

Moves to p10 offset by 50mm in X, all relative to Part1 frame.

Program Structure

Modular Programming

Main Program

```
DEF Main()
    Initialize()
    LOOP
        PickPart()
        ProcessPart()
        PlacePart()
        UpdateCounter()
    ENDLOOP
END
```

Subroutines

```
DEF PickPart()
    MoveToApproach(PickLocation)
    MoveToGrasp(PickLocation)
    CloseGripper()
    IF NOT PartPresent() THEN
        HALT
    ENDIF
    Retract()
END
```

Benefits: - Reusable code - Easier debugging - Clearer structure

Variables and Data

Position Variables

VAR robtarget p1 := [[500, 100, 300], [1, 0, 0, 0], [0, 0, 0, 0], [9E9, 9E9, 9E9, 9E9,

Components: - Position: [X, Y, Z] - Orientation: Quaternion [q1, q2, q3, q4] - Robot configuration
(joint angles that achieve this pose)

Arrays

Part Locations:

```
DECL POS PartLocations[10]
PartLocations[1] = {X 100, Y 50, Z 20, A 0, B 0, C 0}
...
FOR i = 1 TO 10
    LIN PartLocations[i]
...
ENDFOR
```

Control Flow

Conditionals

```
IF DI_PartPresent = 1 THEN
    ProcessPart;
ELSE
    WaitForPart;
ENDIF
```

Loops

```
FOR i = 1 TO 10
    PickAndPlace(i)
ENDFOR
```

```
WHILE NOT Done()
    Step()
ENDWHILE
```

Error Handling

```
VAR errnum err_var;
...
GripPart;
IF ERRNO <> ERR_NO_ERR THEN
    ! Handle gripper error
    TPWrite "Gripper failed";
    Stop;
ENDIF
```

Simulation

Collision Detection

Geometric Modeling

Robot Model: - Links represented as meshes or primitive shapes - Accurate geometry from CAD

Environment: - Fixtures, machines, obstacles - Simplified collision geometry (convex hulls, bounding boxes)

Collision Checking

Methods: - Bounding box intersection (fast, conservative) - Mesh-to-mesh distance calculation (accurate, slower) - Swept volume (checks entire motion path)

Configuration: - Define safety margins (e.g., 10mm clearance) - Warning vs. error levels - Specific link pairs to check

Self-Collision

Robot Against Itself: - Cable carriers against base - Tool against shoulder - Common in complex poses

Prevention: - Joint limits - Path planning avoidance

Cycle Time Analysis

Simulation Benefits

Accurate Prediction: - Run simulated program - Measure time for each motion - Total cycle time

Bottleneck Identification: - Which motions are slowest? - Where are dwell times?

Optimization: - Adjust speeds - Change approach paths - Reorder operations

Comparison

Simulation vs. Reality: - Typically within 5-10% if well-calibrated - Acceleration/deceleration details differ - Useful for relative comparisons

Reachability Analysis

Workspace Verification

Check All Points: - Can robot reach all programmed positions? - Any near joint limits? - Any singularities?

Visualization: - Color code: green (good), yellow (marginal), red (unreachable) - Adjust robot placement or part orientation

Dexterity Analysis

Manipulability: - How far from singularities? - How many joint configurations available? - Used to optimize robot placement

Virtual Commissioning

Digital Twin

Complete Workcell: - Robot, CNC machines, conveyors - Control logic (PLC simulation) - Part flow and timing

Benefits: - Test integration before physical build - Identify issues early - Train operators on virtual system

Hardware-in-Loop (HIL)

Real Controller, Simulated Robot: - Connect actual robot controller to simulation - Controller thinks it's controlling real robot - Test real programs safely

Path Generation

CAD-Based Paths

Surface Machining - Select CAD surface - Define tool, stepover, feed rate - Generate tool path - Convert to robot motion

Edge Following - Select edges (for welding, gluing, etc.) - Define approach/retract - Generate continuous path

Point Grids - Define array of positions (palletizing, inspection) - Generate repeated motion pattern

Teach by Demonstration

Lead-Through Programming

Collaborative Robots: - Physically move robot through desired path - System records positions - Playback recorded path

Advantages: - Very intuitive (no programming) - Good for complex 3D paths

Disadvantages: - Limited precision - Hard to achieve smooth motion - May need refinement

Kinesthetic Teaching

Gravity Compensation: - Robot becomes weightless (zero-gravity mode) - Operator guides robot - Record waypoints or continuous path

Advanced Features

Conveyor Tracking

Setup - Encoder on conveyor belt - Measures belt position continuously - Robot synchronizes motion to moving part

Programming

```
DropWObj := CnvToObj(WObj_Conveyor);
MoveL RelTool(p1, 0, 0, -50), v500, z10, Tool1\WObj:=DropWObj;
MoveL p1, v500, fine, Tool1\WObj:=DropWObj;
MoveL RelTool(p1, 0, 0, 50), v500, z10, Tool1;
DropWObj := WObj0; ! Stop tracking
```

Applications: - Pick from moving conveyor - Place on moving line - Inspection on moving parts

Vision Integration

Vision-Guided Motion

Process: 1. Trigger camera capture 2. Vision system locates part 3. Sends offset to robot 4. Robot adjusts programmed position

Programming

```
# Trigger camera
trigger_camera()

# Get offset from vision system
offset = get_vision_offset()

# Apply offset to nominal position
```

```
actual_pos = nominal_pos + offset  
  
# Move to actual position  
robot.move(actual_pos)
```

Calibration

Camera-to-Robot Transform: - Teach robot to several fiducial points - Camera measures same points in image - Calculate transformation matrix

Multi-Robot Coordination

Independent Control - Separate programs - Coordinate via I/O signals (handshaking) - Simpler but less efficient

Coordinated Motion - Single controller manages multiple robots - Synchronized trajectories - Shared workspace management - Complex programming

Example Application - Two robots weld same large part - One robot holds part, other welds - Coordinated motion required

Next: 10.8 CNC and Workcell Integration

References

1. ISO 10218-1:2011 - Robots and robotic devices - Safety requirements
2. ISO 9283:1998 - Manipulating industrial robots - Performance criteria
3. Denavit, J. & Hartenberg, R.S. (1955). "A Kinematic Notation for Lower-Pair Mechanisms." *ASME Journal of Applied Mechanics*, 22, 215-221
4. Craig, J.J. (2017). *Introduction to Robotics: Mechanics and Control* (4th ed.). Pearson
5. Lynch, K.M. & Park, F.C. (2017). *Modern Robotics*. Cambridge University Press
6. ABB Robot Studio Software - Robot simulation and programming
7. KUKA System Software (KSS) - Robot control and motion planning

Module 10 - Robotic Arm

Integrating robotic arms with CNC machines and other equipment creates flexible manufacturing cells. This section covers communication protocols, coordination strategies, and complete workcell design.

Integration Architectures

Standalone Robot + CNC

Independent Controllers

Configuration: - Robot has dedicated controller - CNC has separate controller - Communication via digital I/O or network - Each system runs independently

Coordination Methods: - Handshaking signals (simple I/O) - Serial communication (RS-232) - Industrial Ethernet (Modbus TCP, EtherCAT)

Advantages: - Maximum flexibility - Standard equipment (no custom integration) - Easily expandable

Disadvantages: - More complex programming - Synchronization limited to signal exchange - No true motion coordination

PLC-Coordinated Cell

Central Control

PLC as Master: - Controls robot, CNC, conveyors, fixtures - Ladder logic or structured text programming - Sequences operations - Manages production flow

Communication: - PLC to robot: Ethernet/IP, PROFINET, or I/O - PLC to CNC: Similar industrial protocols - HMI connects to PLC

Advantages: - Centralized logic - Standard factory automation approach - Good for multi-machine cells - Familiar to maintenance staff

Disadvantages: - Additional hardware cost - PLC programming expertise required - Another point of failure

Integrated Control

Single Controller

Rare Configuration: - One controller manages robot and CNC axes - Examples: CNC with robotic arm add-on

Advantages: - Synchronized motion possible - Single programming environment - Reduced hardware

Disadvantages: - Limited vendor options - Less flexible - Specialized knowledge required

Communication Protocols

Digital I/O (Hardwired)

Signal Types

CNC Outputs □ Robot Inputs: - Cycle complete (CNC finished machining) - Door open (safe for robot entry) - Part present (part in fixture) - Ready for load/unload - Error/fault

Robot Outputs □ CNC Inputs: - Part loaded (robot placed workpiece) - Part unloaded (robot removed part) - Robot busy (do not close door) - Load/unload complete - Robot error/fault

Electrical Interface

Voltage Levels: - 24V DC (most common industrial standard) - Sinking vs. sourcing I/O - Verify compatibility between devices

Wiring: - Shielded cable for noise immunity - Separate from power cables - Terminal blocks for connections - Fused or protected circuits

Handshaking Sequence Example

Machine Tending: 1. CNC completes part, signals "Cycle Complete" 2. CNC opens door, signals "Door Open" 3. Robot detects signals, signals "Robot Busy" 4. Robot enters work area, unloads finished part 5. Robot loads new workpiece 6. Robot exits, signals "Part Loaded" and clears "Robot Busy" 7. CNC closes door 8. CNC starts machining cycle

Timing: - Typical: 5-20 second robot cycle - CNC may take 2-10 minutes - Robot waits for CNC completion

Advantages - Simple, reliable - No special protocols - Easy troubleshooting (multimeter)

Disadvantages - Limited data (only on/off signals) - Many wires for complex cells - No feedback beyond binary states

Serial Communication

RS-232

Characteristics: - Point-to-point (one-to-one) - Distance: Up to 15 meters - Speed: 9600 to 115200 baud typical - Simple protocol

Message Format (Example):

```
Robot → CNC: "READY\r\n"
CNC → Robot: "LOAD_PART\r\n"
Robot → CNC: "LOADING\r\n"
Robot → CNC: "LOAD_COMPLETE\r\n"
CNC → Robot: "START_CYCLE\r\n"
```

Applications: - Send part numbers or programs - Request status information - Transmit measurement data

RS-485

Characteristics: - Multi-drop (one-to-many) - Distance: Up to 1200 meters - Higher noise immunity than RS-232 - Requires termination resistors

Use: - Multiple devices on one bus - Fieldbus protocols (Modbus RTU)

Industrial Ethernet

Modbus TCP

Overview: - Client-server model - Read/write registers and coils - Simple, widely supported

Implementation:

```
from pymodbus.client import ModbusTcpClient
```

```

client = ModbusTcpClient('192.168.1.100')
client.write_coil(0, True) # Signal start
status = client.read_coils(10, 1) # Read complete flag
client.close()

```

Data Types: - Coils: Binary outputs (write) - Discrete inputs: Binary inputs (read) - Holding registers: 16-bit read/write - Input registers: 16-bit read-only

EtherCAT

Characteristics: - Real-time deterministic communication - Microsecond synchronization - Ring or line topology - Specialized hardware required

Applications: - Coordinated multi-axis motion - High-speed I/O - Advanced robot systems

Performance: - Update rate: 1-10 kHz - Jitter: <1 μs

PROFINET

Industrial Ethernet: - Similar to EtherCAT (real-time variant: PROFINET IRT) - Standard Ethernet hardware with PROFINET stack - Common in Siemens PLCs

Ethernet/IP

CIP Protocol: - Used by Allen-Bradley/Rockwell - Standard Ethernet - Moderate performance (not hard real-time)

OPC UA

Modern Standard: - Platform-independent - Secure (encryption, authentication) - Supports complex data structures - Growing adoption in Industry 4.0

Example:

```

from opcua import Client

client = Client("opc.tcp://192.168.1.100:4840")
client.connect()
node = client.get_node("ns=2;i=10") # Part count
value = node.get_value()
client.disconnect()

```

CNC Machine Tending

Door Control

Automatic Door Opening

Methods: - Robot triggers door open signal (if CNC supports) - Robot physically opens manual door (handle gripper) - Door already open (no door, or permanent access)

Integration: - Interlock: Robot cannot enter while CNC running - Safety circuit: Door open disables CNC spindle - Sensor confirms door fully open

Gripper vs. Door Actuator

Robot Gripper Opens Door: - Special gripper or tool changer to door handle gripper - Flexible but slow

Pneumatic/Electric Actuator: - Dedicated door opener - Robot signals actuator - Faster and more reliable

Part Handling

Workpiece Loading

Gripper Design: - Grips raw stock securely - Clearance for fixture or chuck - Release after placement

Fixture Interfaces: - Vise jaws (manual or pneumatic) - Chuck (pneumatic or hydraulic) - Vacuum table - Custom fixture with clamps

Loading Sequence: 1. Approach position (above fixture, offset) 2. Lower to load position 3. Place part (may require force control for seating) 4. Signal fixture to clamp (I/O to CNC or PLC) 5. Wait for clamp confirmation 6. Open gripper 7. Retract

Part Flipping

Multi-Operation Machining: - Machine OP1 (one side) - Robot unloads, flips 180° - Robot reloads for OP2 (other side)

Challenges: - Gripper must access part from both orientations - Flipping fixture or two grippers (tool change) - Registration for precise alignment

Part Identification

Methods: - Barcode/QR code reading (camera on robot) - RFID tags on fixture or pallet - Manual operator input

Use: - Select appropriate CNC program - Track part serial numbers - Routing (different part types to different machines)

Process Integration

CNC Program Selection

Automatic Selection: - Robot identifies part type - Robot signals program number to CNC - CNC loads correct program

Implementation: - Modbus write to program number register - Serial command (depends on CNC control) - PLC intermediary

Tool Offset Updates

From Measurement: - Robot measures part dimension (touch probe) - Calculates required tool offset - Sends offset to CNC - CNC adjusts subsequent parts

Chip Management

Challenges: - Chips on part prevent accurate placement - Chip buildup in fixture

Solutions: - Air blast (robot-mounted nozzle) - Brush station (robot sweeps part) - High-pressure coolant washdown - Vacuum pickup station

Multi-Machine Cells

Single Robot, Multiple CNCs

Layout

Linear: - Machines in row - Robot on floor or rail - Simple accessibility

U-Shape: - Machines around robot - Compact footprint - Robot centered for equal reach

Scheduling

Fixed Sequence: - Robot visits Machine 1, then 2, then 3, repeat - Simple, predictable - May leave machines idle

Dynamic Priority: - Service machine with longest wait first - Or machine closest to completion - Better utilization

Optimization: - Minimize robot travel time - Balance CNC cycle times - Buffer parts to smooth variations

Buffer Stations

Part Queues: - Input queue (raw stock ready to load) - Output queue (finished parts awaiting pickup) - Intermediate buffers between operations

Design: - Gravity-fed racks - Pallet stands - Conveyor sections

Benefits: - Smooth production flow - Decouple machine cycles - Allow robot to service multiple machines efficiently

Robot on Rail

Linear Seventh Axis

Configuration: - Robot mounted on linear slide - Extends reach horizontally - Serves machines along line

Specifications: - Rail length: 5-30 meters typical - Speed: 1-3 m/s - Positioning accuracy: +/-0.5-2 mm - Servo or rack-and-pinion drive

Programming: - Seventh axis coordinated with robot - Positions robot at each machine station - May move during robot operation for long parts

Dual-Arm Cells

Two Robots

Independent: - Each robot has own tasks - Coordinate via signals (collision avoidance)

Collaborative: - Hold part together - One positions, other machines/assembles - Complex coordination required

Workspace Sharing

Safety Zones: - Define exclusive zones (only one robot at a time) - Shared zones (collision monitoring required) - Handoff zones (part transfer)

Collision Avoidance: - Signal-based (Robot 1 in Zone A → Robot 2 waits) - Real-time monitoring (laser scanners) - Simulation verification

Auxiliary Equipment

Part Conveyors

Input Conveyor

Function: - Delivers raw stock to robot - May include part separation or orientation

Integration: - Sensor detects part arrival - Signals robot to pick - Conveyor stops or robot tracks moving part

Output Conveyor

Function: - Carries finished parts away - May route to inspection, packaging, or storage

Integration: - Robot places part on conveyor - Conveyor sensor confirms placement - Accumulation zone for batch collection

Part Washers

Purpose - Remove chips and coolant - Clean parts for inspection or assembly

Integration

Transfer: - Robot loads part into washer - Washer cycles (spray, rinse, dry) - Robot unloads clean part

Timing: - Washer cycle: 1-5 minutes typical - Robot continues serving other machines - Returns when wash complete

Inspection Stations

CMM or Vision Inspection

Process: 1. Robot picks part from CNC 2. Robot presents part to CMM or camera 3. Measurement system inspects dimensions 4. Results logged and communicated 5. Pass: Robot places in accept bin 6. Fail: Robot places in reject bin or rework

In-Process Measurement

Touch Probe on Robot: - Measure part while in fixture - Quicker than separate station - Less accurate than dedicated CMM

Data Feedback: - Send measurements to CNC for offset adjustment - SPC charting and trend analysis

Tool Changers and Racks

Tool Magazine

Multiple End Effectors: - Different grippers for part types - Process tools (welding, grinding) - Inspection tools (probe, camera)

Automatic Tool Change: - Robot docks at tool rack - Releases current tool - Picks new tool - Loads tool parameters (TCP, mass)

Benefits: - Flexibility (handle multiple part types) - Reduces downtime (vs. manual changes) - Enables complex processes

Cell Control Software

Supervisory Control

Function

Cell Controller: - Coordinates robot, CNCs, conveyors, etc. - Production scheduling - Part tracking - Data logging

Implementation: - Industrial PC running custom software - PLC with SCADA/HMI - Commercial MES (Manufacturing Execution System)

Communication

To Equipment: - Robot: Ethernet (OPC UA, Modbus) - CNC: MTConnect, Modbus, proprietary - PLC: Industrial Ethernet - Sensors: I/O, Ethernet

Features

Production Scheduling: - Queue of jobs (part types and quantities) - Assign to machines - Optimize throughput

Part Tracking: - Serial numbers or batch IDs - Which machine, which operation - Quality data association - Genealogy and traceability

Reporting: - Machine utilization (OEE - Overall Equipment Effectiveness) - Cycle times - Downtime and reasons - Quality metrics

HMI (Human-Machine Interface)

Operator Interface

Display: - Current status (running, waiting, error) - Part counts (produced, remaining) - Cycle time - Alarms and messages

Controls: - Start/stop production - Select part program - Reset errors - Manual jog (with appropriate safety)

Remote Monitoring

Access: - Web interface or mobile app - View status from anywhere - Alerts via email or SMS

Workcell Design Best Practices

Layout Optimization

Accessibility

Robot Placement: - Maximize useful workspace - Minimize travel distance - Avoid reaching over obstacles

Operator Access: - Loading feedstock - Unloading finished parts - Maintenance and troubleshooting

Service Access: - Clear paths for technicians - Access to control panels - Room for forklift/pallet jack

Safety Compliance

Guarding: - Physical barriers around robot workspace - Interlocked gates for operator access - Separation from traffic areas

Visibility: - Clear sightlines for monitoring - Good lighting - Status indicators visible

Workflow Optimization

Material Flow

One-Way Flow: - Raw material in one side - Finished parts out opposite side - Minimize cross-traffic

Minimize Handling: - Direct transfer where possible - Avoid intermediate storage unless buffering needed

Cycle Time Balance

Bottleneck Analysis: - Identify slowest operation (limits throughput) - Option 1: Speed up bottleneck - Option 2: Add parallel capacity - Option 3: Rebalance tasks

Utilization: - Keep high-value equipment (CNCs) running maximum time - Robot idle time acceptable if CNCs fully utilized

Maintenance Considerations

Accessibility

Design for Maintenance: - Easy access to wear items (grippers, tool changers) - Removable panels - Service positions (clear area for major work)

Spare Parts: - Storage nearby - Quick-change designs - Minimize downtime

Utilities

Power Distribution: - Adequate capacity - Clean power for controls - Emergency shutoff accessible

Compressed Air: - Filtration and regulation - Adequate flow for all pneumatic devices - FRL (Filter, Regulator, Lubricator) per device or zone

Network: - Separate control network (isolated from office) - Managed switches - Proper grounding and shielding

Next: 10.9 Safety Systems

References

1. **ISO 10218-1:2011** - Robots and robotic devices - Safety requirements
 2. **ISO 9283:1998** - Manipulating industrial robots - Performance criteria
 3. **Denavit, J. & Hartenberg, R.S. (1955).** "A Kinematic Notation for Lower-Pair Mechanisms." *ASME Journal of Applied Mechanics*, 22, 215-221
 4. **Craig, J.J. (2017).** *Introduction to Robotics: Mechanics and Control* (4th ed.). Pearson
 5. **Lynch, K.M. & Park, F.C. (2017).** *Modern Robotics*. Cambridge University Press
 6. **ABB Robot Studio Software** - Robot simulation and programming
 7. **KUKA System Software (KSS)** - Robot control and motion planning
-

Module 10 - Robotic Arm

Robotic arms present serious hazards requiring comprehensive safety design. This section covers applicable standards, safeguarding methods, collaborative operation, and safety validation.

Applicable Standards

International Standards

ISO 10218-1: Robots and Robotic Devices - Safety Requirements for Industrial Robots

Scope: - Design and construction of industrial robots - Inherent safety measures - Safeguarding requirements - Information for use

Key Requirements: - Maximum speed in manual mode: 250 mm/s - Three-position enabling device - Emergency stop on teach pendant - Protective stop function - Safety-rated monitoring

ISO 10218-2: Robot Systems and Integration

Scope: - Integration of robots into complete systems - Safeguarding of robot cells - Risk assessment - Installation and commissioning

Key Concepts: - Collaborative operation spaces - Safety functions (STO, SS1, SS2) - Operator interaction modes - Performance level requirements

ISO/TS 15066: Robots and Robotic Devices - Collaborative Robots

Collaborative Operation: - Safety-rated monitored stop - Hand guiding - Speed and separation monitoring - Power and force limiting

Biomechanical Limits: - Maximum allowable contact forces and pressures - Varies by body region
- Transient vs. quasi-static contact - Published force/pressure tables

ISO 13849-1: Safety-Related Parts of Control Systems

Performance Levels (PL): - PL_a (lowest) to PL_e (highest) - Based on reliability and fault tolerance
- Robot safety functions typically require PL_d or PL_e

Categories: - Category B: Basic (single channel) - Category 1: Well-tried components - Category 2: Test at intervals - Category 3: Single fault safe (dual channel) - Category 4: Single fault safe with fault detection

ISO 13850: Emergency Stop

Requirements: - Category 0 (immediate power removal) or Category 1 (controlled stop then power removal) - Red mushroom button on yellow background - Direct opening action (mechanically linked) - Manual reset required - Overrides all other functions

North American Standards

ANSI/RIA R15.06: Industrial Robots and Robot Systems

Harmonized with ISO 10218: - Safeguarding methods - Operating modes - Collaborative operation (references ISO/TS 15066)

Additional Requirements: - Pendant enabling device specifications - Lockout/tagout provisions - Training requirements

OSHA Regulations

29 CFR 1910.212: - Machine guarding general requirements

29 CFR 1910.147: - Control of Hazardous Energy (lockout/tagout)

NFPA 79: Electrical Standard for Industrial Machinery

Electrical Safety: - Circuit protection - Grounding and bonding - Control circuits - Emergency stop circuits

Risk Assessment

Hazard Identification

Mechanical Hazards

Crushing: - Between robot and fixed objects - Between robot links - In gripper or tool

Impact: - Moving robot strikes person - High speeds increase severity - Unexpected motion

Shearing: - At joints - Between gripper jaws - Tool cutting edges

Entanglement: - Cables and hoses - Rotating joints - Moving conveyors

Electrical Hazards

Contact: - Live parts in control cabinet - Motor power connections - Damaged cables

Arc Flash: - High-power systems (>480V) - Proper PPE and procedures required

Process Hazards

Welding: - Arc flash, UV radiation - Fumes and gases - Fire hazard

Cutting/Grinding: - Flying debris - Sparks - Dust

Chemical Dispensing: - Toxic or corrosive materials - Vapor exposure - Skin contact

Risk Evaluation

Severity Assessment

S1 - Slight Injury: - Bruising, minor cuts - First aid treatment

S2 - Serious Injury: - Fractures, lacerations requiring stitches - Medical treatment, possible hospitalization

S3 - Fatal or Permanent Disability: - Crushing, amputation - Head or internal injuries

Frequency and Exposure

F1 - Rare: - Less than once per year - Brief exposure

F2 - Frequent: - Daily or continuous exposure - Long duration

Probability of Occurrence

P1 - Low: - Unlikely under normal operation - Multiple safeguards present

P2 - Medium: - Possible, has occurred elsewhere

P3 - High: - Has occurred or likely to occur

Risk Level Calculation

Risk = Severity × Frequency × Probability

High Risk: - Requires additional safeguarding - Cannot operate until mitigated

Medium Risk: - Consider additional measures - Document justification if accepted

Low Risk: - Acceptable with standard safeguards - Monitor and maintain

Risk Reduction

Hierarchy of Controls (ISO 12100)

1. Inherently Safe Design:

- Limit force and speed
- Rounded edges (no pinch points)
- Eliminate hazards if possible

2. Safeguarding:

- Physical guards
- Interlocks
- Presence-sensing devices
- Safety-rated control functions

3. Warnings and Signage:
 - Visual indicators (stack lights, strobes)
 - Audible alarms
 - Safety signs and labels
4. Training and Procedures:
 - Operator training
 - Maintenance procedures
 - Personal protective equipment (PPE)

Safeguarding Methods

Physical Barriers

Perimeter Fencing

Specifications: - Height: 1800 mm minimum (2000-2400 mm typical) - Mesh size: <=50 mm (prevent reach-through) - Distance from robot: >=500 mm (when fully extended) - Vertical bars (not horizontal, prevent climbing)

Materials: - Welded wire mesh (most common) - Expanded metal - Polycarbonate panels (visibility)
- Powder-coated steel frame

Interlocked Gates

Access Gates: - Self-closing - Safety switches (ISO 14119) - Dual-channel monitoring - Trapped key systems (for maintenance)

Gate Configurations: - Pedestrian access (operator entry) - Material access (part loading) - Maintenance access (larger opening)

Interlock Function: - Robot stops when gate opens (Category 1 stop) - Cannot restart while gate open - Manual reset after gate closes

Fixed Guards

Non-Removable Panels: - Require tools to remove - Protect specific hazards (pinch points, sharp edges) - Attached with tamper-resistant fasteners

Light Curtains

Type 4 Safety Light Curtains (IEC 61496): - Infrared beam array - Detects intrusion - Safety-rated output

Specifications: - Protected height: 300-1800 mm typical - Resolution: 14mm, 30mm, or 50mm - Response time: <20 ms - Safety category 4, PLe

Placement: - At access points where physical gates not practical - Muting for part passage (requires careful design)

Distance Calculation:

$$\text{Safety distance} = K \times T + C$$

Where: - K = hand speed constant (1600 mm/s) - T = total response time (sensor + robot stop) - C = additional distance based on resolution

Safety Laser Scanners

2D/3D Area Monitoring: - Configurable zones (warning and protective) - Automatic zone switching
- Detect personnel entry

Features: - Range: 3-70 meters - Angular resolution: 0.1-1 degree - Update rate: 20-40 Hz -
Multiple zone sets

Applications: - Large workcells - Mobile robots - Dynamic access control

Safety Functions

Safe Torque Off (STO)

Function: - Removes power to motor drives - Robot coasts to stop - Category 0 stop (uncontrolled)

Implementation: - Dual-channel monitoring - Directly disables drive power stage - No reliance on software

Performance Level: - PLd or PLe depending on architecture

Use: - Emergency stop - Guard door interlock - Basic safety stop

Safe Stop 1 (SS1)

Function: - Controlled deceleration - Power removed after stop - Category 1 stop

Advantage: - Prevents uncontrolled motion - Protects workpiece and robot

Requirements: - Safe motion monitoring during stop - Verify standstill before removing power

Safe Stop 2 (SS2)

Function: - Controlled stop - Power maintained (holds position)

Use: - Temporary stops (gate open briefly) - Operator loads part - Robot ready to resume

Requirements: - Continuous position monitoring - Detect unexpected motion

Safely-Limited Speed (SLS)

Function: - Monitors robot speed - Stops if exceeds programmed limit

Parameters: - Typically 250 mm/s for manual mode (ISO 10218) - Lower limits for collaborative operation

Implementation: - Safety-rated encoders - Continuous velocity calculation - Safe comparison to limit

Safe Reduced Speed (SRS)

Function: - Limits maximum speed in certain modes or zones

Use: - Teaching mode - Collaborative zones - Maintenance mode

Safe Operating Stop (SOS)

Function: - Monitors that robot is stationary - Allows safe work near robot - Power maintained

Example: - Robot holds part while operator inspects - No motion permitted - Triggers stop if any movement detected

Operating Modes

Automatic Mode

Configuration: - Full production speed - All guards closed - No personnel in restricted space - Safety functions: STO via e-stop or interlocks

Access: - Only via interlocked gates (stops robot) - Emergency stop if entry detected

Manual Mode (Teach/Programming)

Requirements per ISO 10218: - Speed limited to 250 mm/s (SLS function) - Three-position enabling device required - Hold middle position: Robot enabled - Release or full press: Robot stops - Pendant must be within restricted space

Enabling Device: - Three positions (off, on, panic) - Panic-stop if squeezed too hard - Deadman switch principle

Safety: - Operator has direct control - Can stop instantly - Limited speed reduces injury severity

Maintenance Mode

Lockout/Tagout (LOTO): - Disconnect all energy sources - Lock main disconnect in "off" position - Tag to identify who locked it - Verify zero energy state

Procedures: - Only authorized personnel - Written procedures - Test before starting work - Restore and test after work

Collaborative Operation

Collaborative Operating Spaces (ISO 10218-2)

Collaborative Workspace - Where robot and operator share space - Requires safety-rated monitoring

Non-Collaborative Space - Robot operates alone - Traditional safeguarding

Transition Zone - Robot slows or stops when operator approaches

Collaborative Methods (ISO/TS 15066)

1. Safety-Rated Monitored Stop

Operation: - Operator enters collaborative space - Robot stops and remains stopped - Safety-rated monitoring of standstill - Operator performs task - Operator exits, robot resumes

Requirements: - SOS (Safe Operating Stop) function - Detection of operator entry (light curtain, scanner, mat) - Manual restart or automatic after exit

Applications: - Part loading/unloading - Inspection - Adjustment

2. Hand Guiding

Operation: - Operator physically guides robot - Force sensor or enabling device on tool - Robot moves with operator - Three-position enabling device required

Requirements: - SLS (speed limited to 250 mm/s or less) - Force/torque sensor or enabling device - Protective stop if device released

Applications: - Teaching positions - Manual operation in constrained spaces

3. Speed and Separation Monitoring (SSM)

Operation: - Minimum distance maintained between robot and operator - Robot slows as operator approaches - Stops if separation too small

Implementation: - Safety laser scanners or vision system - Real-time position monitoring (robot and human) - Graduated response (slow □ slower □ stop)

Separation Distance:

$$S = Sh + Sr + Ss + C$$

Where: - Sh = distance human can move during robot reaction time - Sr = distance robot moves during stop - Ss = safety margin - C = position uncertainty

4. Power and Force Limiting (PFL)

Operation: - Contact between robot and operator permitted - Forces limited to safe values per ISO/TS 15066 - No sharp edges or pinch points

Force Limits (Examples from ISO/TS 15066): - Skull: 130 N (transient), 65 N (quasi-static) - Face: 140 N / 65 N - Chest: 140 N / 110 N - Abdomen: 110 N / 65 N - Hand/fingers: 140 N / 40 N

Requirements: - Inherent safe design (limited power, compliant surfaces) - Force/torque sensing in all joints - Safety-rated force monitoring - Validated biomechanical assessment

Design Considerations: - Padding or soft covers - Rounded edges - Limited mass and speed - Force-limited motors or clutches

Collaborative Robot Examples

Universal Robots (UR series) - Force/torque sensing in all joints - Configurable safety zones and speeds - Hand-guiding capable - Primarily SSM and PFL modes

KUKA LBR iiwa - Torque sensors in all 7 joints - High sensitivity - Medical and precision applications - All collaborative modes

ABB GoFa/SWIFTI - Dual-encoder safety - SSM and PFL - Higher payload than typical cobots

Franka Emika - Integrated force sensing - Research and precision applications

Safety Validation

Pre-Commissioning Checks

Mechanical - Guards installed per design - Interlocks functional - No sharp edges or pinch points - Cable routing secure

Electrical - Emergency stops wired correctly - Safety circuits dual-channel - Proper grounding - Circuit protection (fuses, breakers)

Control System - Safety functions configured - Speed limits set correctly - Zones and parameters validated

Functional Testing

Emergency Stop

Test Each E-Stop: 1. Start robot motion (manual or auto) 2. Press e-stop button 3. Verify immediate stop 4. Verify power removed (STO) 5. Attempt to restart (should fail) 6. Reset e-stop 7. Verify manual restart required

Measure stop time and distance.

Interlock Testing

Each Interlocked Gate: 1. Start robot motion 2. Open gate 3. Verify robot stops (Category 1) 4. Verify cannot restart with gate open 5. Close gate 6. Verify restart possible 7. Attempt to defeat interlock (tape, bypass) 8. Verify robot stops if defeated

Light Curtain / Scanner

Intrusion Test: 1. Start robot motion 2. Break light curtain with test piece 3. Verify robot stops 4. Measure response time 5. Verify safety distance adequate 6. Test muting functions (if applicable)

Speed Monitoring (SLS)

Manual Mode Test: 1. Enter manual mode 2. Attempt to exceed 250 mm/s 3. Verify robot stops or limits speed 4. Test with different motion types

Collaborative Mode: 1. Configure speed limit (e.g., 100 mm/s) 2. Run test program 3. Verify limit enforced

Performance Level Validation

Calculate Performance Level

For each safety function: - Identify architecture (Category 1-4) - Calculate MTTF (mean time to failure) - Determine diagnostic coverage - Use ISO 13849-1 graphs/tables - Verify achieves required PL (typically PLd or PLe)

Common Mode Failures

Check for: - Single point failures that defeat safety - Environmental factors (EMI, temperature) - Software errors (use certified software)

Periodic Testing

Requirements: - Frequency based on risk assessment - Typically: Daily (e-stop), Weekly (interlocks), Annually (full validation) - Document all tests

Documentation

Safety Manual

Required Contents: - Risk assessment results - Safety functions and PLs - Operating modes and restrictions - Training requirements - Maintenance procedures - Emergency procedures

Validation Report

Include: - Test procedures - Test results and measurements - Deviations and resolutions - Date and personnel - Acceptance signatures

Training Records

Document: - Operators trained - Topics covered - Competency verification - Dates and refresher schedule

Common Safety Violations

Bypassing Safety Devices - Jumper wires around interlocks - Taping guard switches - Disabling light curtains - Holding enabling device in active position

Inadequate Guarding - Gaps allowing reach-in - Insufficient fence height - Guards easily removable without tools

Improper E-Stop Implementation - Single-channel only - No mechanical latching - Insufficient quantity or placement

Working Inside Safeguards - Without LOTO - Robot powered - Relying on software safeties

Lack of Training - Untrained operators - No emergency procedures - Unaware of hazards

Best Practices

- Design inherently safe first (reduce force, speed, mass)
- Use safety-rated components throughout
- Dual-channel monitoring for critical functions
- Regular testing and maintenance
- Comprehensive training program
- Document everything
- Review and update after modifications
- Involve safety professionals

Next: 10.10 Maintenance

References

1. ISO 10218-1:2011 - Robots and robotic devices - Safety requirements

2. **ISO 9283:1998** - Manipulating industrial robots - Performance criteria
 3. **Denavit, J. & Hartenberg, R.S. (1955).** "A Kinematic Notation for Lower-Pair Mechanisms." *ASME Journal of Applied Mechanics*, 22, 215-221
 4. **Craig, J.J. (2017).** *Introduction to Robotics: Mechanics and Control* (4th ed.). Pearson
 5. **Lynch, K.M. & Park, F.C. (2017).** *Modern Robotics*. Cambridge University Press
 6. **ABB Robot Studio Software** - Robot simulation and programming
 7. **KUKA System Software (KSS)** - Robot control and motion planning
-