This PPT is based on a series of videos from FANUC engineer

What is a Robot and its Purpose

Robots have a variety of applications, including material handling, palletizing, pick and place, material removal (such as grinding, painting), etc..

In the post-epidemic era, the application of robots will be more extensive due to their reliability, not getting sick, ability to maintain cleanliness, and production of high-quality parts.

Learning robot technology is to be at the forefront of future development. Robots are being used in various industries such as the RV industry, automotive industry, pharmaceutical industry (integrated with cameras), etc..

Do not be limited by the robot applications shown in the course. There are many future application areas that have not yet been created.

Material handling is one of the largest application areas for robots, followed by palletizing and material removal. Welding is also an important application.

Robot Space

Maximum Space: The furthest range that the robot arm can reach when fully extended, even if operations may not be possible at that position. Imagine the furthest distance the robot can reach when fully "lying down" and extended.

Restricted Space: An area set for safety or practical work needs, which may be limited by protective measures or is within the maximum space but suitable for actual work.

Operating Space: The area where the robot actually works. It can be understood as the range within the restricted space where the robot actually performs operations.

Maximum Envelope: All areas swept by all moving parts of the robot during motion.

Operating Envelope: The area where the robot performs actual work, usually concentrated in one area to reduce cycle time.

Safety

Operator safety is always the top priority. Robots can be replaced, parts can be remade, but people cannot be replicated.

The safety priority of a robot system from high to low is: operator, robot, external equipment (such as signal lights, fixtures, controllers), tooling, workpiece.

The Teach Pendant has an Emergency Stop (E-stop) button, which is used to stop all movements in an emergency.

The Dead Man Switch has three positions: fully released (no movement), fully pressed (emergency stop), middle position (allows movement). The operator needs to keep the dead man switch in the middle position when manually operating the robot.

Common safety measures include: Light Curtains (stop upon entry), Safety Fences, Pressure Mats (triggered when stepped on), Interlocks (e.g., prohibits startup if the door is not closed), Warning Lights, Motion Limits (physical limit screws).

Collaborative Robots have additional pressure, speed, and safety sensors and can stop upon collision.

When entering the robot work area, be sure to carry the teach pendant or some kind of emergency stop device. Ideally, it should be the teach pendant.

There are usually safety doors and guardrails with interlocks.

The Robotics Industries Association (RIA) has developed relevant safety guidelines, which can be viewed on its website.

Maximum envelope is the area that all moving parts of the robot may reach.

Anti-Tie Down Logic is used to prevent people from bypassing safety mechanisms for operational convenience. For example, it is absolutely forbidden to place heavy objects on pressure sensors to keep them activated.

Multiple safety mechanisms need to be considered during programming to prevent all safety measures from being bypassed simultaneously.

Entering the opened robot work envelope is absolutely prohibited.

In an educational environment, even small robots can cause injury, so never start the robot in automatic mode before the teacher checks.

When moving the robot in automatic mode, be sure to close the safety door. Be careful when entering the work area in manual (teach) mode for observation or setting coordinate systems, and do not stay in the robot work cell for too long.

The robot can be subjected to Lock Out Tag Out to ensure safety during maintenance work and prevent accidental startup. This is especially important when replacing servo motors and other scenarios requiring power disconnection.

When programming, procedural precautions should be taken to ensure handshaking signals between the robot and external equipment to ensure safe operation and avoid damage to equipment or workpieces.

Pay attention to Override Speed, visualize the robot's motion trajectory, keep the work area clean, and ensure circuit protection is in place during programming.

Home Position and Interference Zones and other functions can be used in programming to improve safety.

Axes and Motion

Industrial robots usually have six axes, providing six degrees of freedom, allowing them to operate flexibly.

Major Axes:

- Axis 1 (J1): The large rotating axis at the base, providing torsional motion (hip).
- Axis 2 (J2): Usually located above Axis 1, providing bending motion (waist bends forward).
- Axis 3 (J3): Provides shoulder up and down movement.

Minor Axes: Used for fine movements.

Axis 4 (J4): Usually provides a torsional movement (wrist twist).
Axis 5 (J5): Provides wrist bending or swinging motion (wrist).
Axis 6 (J6): The rotation of the end flange, used for tool posture adjustment (face rotation).

Sometimes robots have additional axes, such as Axis 7 (J7) mounted on a gantry or with a turntable, which will not be covered in this course.

Each axis can move independently, but the robot's greatest strength lies in its ability to coordinate the simultaneous movement of multiple axes.

The motion range (Axis Limits) of each axis can be limited by programming. The motion range of an axis is usually expressed in angles (e.g., a circle is 360 degrees). The manufacturer presets the motion range of the axes, which can be adjusted as needed, but caution should be exercised, and a robot restart may be required after adjustment.

Fanuc robots use the Dual Check Safety (DCS) system to provide an additional layer of safety by checking the robot's speed and position. DCS can set position checks, speed checks, and speed checks in the Cartesian coordinate system. DCS can recognize whether the robot is about to leave the set safety area and stop its motion.

Controller

The controller is the brain of the robot and runs the software.

The Teach Pendant is usually connected to the controller via a cable.

The controller internally contains a power supply, transformer, memory, I/O board, etc..

Fanuc offers different sizes of controllers, such as the A-type controller, B-type controller, Open Air Mate controller, and Mate controller, suitable for different application scenarios and installation spaces. We use the A-type controller.

The controller usually has an Operating Panel containing a fault reset button, teach/run mode switch key, start button, and fault indicator light, etc. The B-type controller has these.

There are also user-defined buttons on the operating panel that can be used to start macro programs, etc..

The controller has Local and Remote two operation modes. Local mode is used for testing and teaching, and remote mode is used for production operation, usually controlled by a PLC or other controllers.

Software

The software is pre-installed in the robot controller.

Be sure to back up the software image regularly to prevent system failures.

There are three main operating modes for the robot:

T1 Mode: Limits the robot's maximum speed to 250 mm/s, used for safe teaching.
T2 Mode: Test run mode, allows the robot to run at up to 100% speed, but still in teach mode and controlled by the teach pendant.
Automatic (Auto) Mode: Full-speed operation mode.

In automatic mode, if the safety fence is breached (Class 1 Stop), the servo power will be disconnected after a short delay, and the robot will stop and brake.

In T1 and T2 modes, safety measures such as safety fences may be bypassed because the operator may be inside the work cell for operation, requiring attention to safety.

Robot System Components

Mechanical Unit: The robot's mechanical arm, driven by AC Servo Motors. Servo motors precisely know their position through serial pulse counters.

Each servo motor has a Mechanical Brake. When the servo is powered on, the brake is released; when powered off, the brake engages to prevent the robot from falling due to gravity. The robot's joints need to be lubricated regularly.

There is a Rotary Pulse Encoder inside the servo motor.

The robot records the "Home" position, which is a preset reference point. At the factory, the robot has an Absolute Zero Setup, usually with the arm fully extended and upright, used for calibration. However, in actual applications, the "Home" position may be arbitrarily set.

There are batteries (usually four C-type batteries) inside the robot to power the encoders when the power is off, maintaining the robot's position memory. If the batteries run out after a power outage, the robot will issue an alarm after being turned on because it does not know its position.

Controller: The control center of the robot.

Peripheral Equipment: Such as Teach Pendant, Programmable Logic Controller (PLC), End of Arm Tooling, Human Machine Interface (HMI), cameras, and vision systems.

Teach Pendant

The teach pendant is the main tool for programming robots, similar to a large game controller.

The programming process usually involves manually Jogging the robot to the target position and then pressing a button to store the position and motion method.

The teach pendant has an emergency stop button (E-stop) and a dead man switch for safe operation.

Each button has its unique function, which will be explained in detail in subsequent courses.

The Shift key is usually used in combination with other buttons to execute the blue-labeled functions.

The Menu button is used to access various settings and configurations.

Cursor up and down keys, hold the Shift key and use them together to jump quickly.

The Step mode button is used for program debugging and can execute the program step by step or continuously. It is recommended to use step mode when testing a program for the first time.

The Reset button is used to clear alarms.

Commonly used buttons include: Previous, Reset, Enter.

The functions of Soft Keys change according to the current screen content.

Buttons such as Position and I/O (Input/Output) are used to view relevant information.

Older controllers have indicator lights on the panel, while newer controllers have the status indicator lights concentrated at the top (LED). We use the newer touchscreen teach pendant, which can theoretically connect a mouse and keyboard.

Forward and Backward buttons can be used to move through program instructions.

When manually operating the robot, you usually need to press and hold the Shift key while using the arrow keys to Jog the robot.

Manual Operation (Jogging)

Jog speed can be adjusted and is displayed as an Override Percentage, indicating the percentage of the current allowable speed. If the speed limit in T1 mode is 250 mm/s, a 50% override speed is 125 mm/s.

There are multiple Jog speeds, and even Fine Speed can be achieved, moving only one servo pulse at a time.

When jogging the robot in Joint mode, press and hold the Shift key and press the buttons labeled J1, J2, etc. to control the forward or reverse movement of each joint separately. Joint mode is usually used to unjam or reconfigure the robot.

When jogging the robot under other coordinate systems, the movement trajectory of the robot's end effector (TCP) may not be a straight line.

Fanuc robots have five different coordinate systems, which can be switched by pressing the COORD button.

Joint coordinate system: Each joint moves independently.
World coordinate system: Moves along the X, Y, Z axes of the Cartesian coordinate system, relative to the origin of the world coordinate system.
Tool coordinate system: Moves relative to the Tool Center Point (TCP).
User coordinate system: User-defined coordinate system.
Jog Frame coordinate system.

In the Cartesian Jogging system (such as the world coordinate system), the robot's movement is related to the X, Y, Z axes in mathematics, and all movements are relative to a zero point (origin).

The origin of the world coordinate system is usually located at the intersection of the robot base J1 and J2.

The right-hand rule is used to determine the positive directions of the X, Y, and Z axes in the world coordinate system.

Positions can be represented using joint coordinates, world coordinates, or user coordinates.

Singularity

A singularity may occur when certain axes of the robot become too aligned. In the world coordinate system, it is easy to occur when the J4 and J6 axes are roughly in a straight line.

At a singularity, the robot controller has difficulty calculating how to achieve the target TCP position because there are multiple combinations of joint angles that can reach the same position.

The solution to the singularity problem is to switch to the Joint coordinate system and slightly jog the J5 axis (approximately ±10-15 degrees), and then switch back to the world coordinate system. This can break the straight line state of the axes.

Hope this note helps you.

Error Types and Causes

Hardware issues: Dead battery, switch failure, broken cable, tool not properly connected. Uncommon but may occur. Fanuc systems have hardware fault diagnosis functions.

Software issues: Incorrect programming, such as operational errors by third-party personnel, motion errors in the program (such as circular arc programming errors).

External errors: Safety-related errors, such as the emergency stop being pressed, or stops triggered by interlocks with other automation equipment (such as opening a safety door). Many external errors can be cleared with the reset button.

Clearing Errors

When an error occurs, the teach pendant will display an error message, such as a DCS error.

The Reset button (usually above the number keys) can clear many errors.

Detailed alarm codes and descriptions can be viewed through Menu button -> Alarm.

The Diagnostic button can also provide fault information.

Fanuc provides electronic documents containing a complete list of alarm codes. Alarm codes usually consist of a four-digit prefix and a three-digit code, along with a description.

Press F1 to view motion errors and application-specific errors.

Press F5 (Detail) to view more detailed error information.

Clear all error messages: Press and hold the Shift key and then press the F4 soft key on the teach pendant.

The Dead Man Switch usually requires a reset to clear the error after being released. However, in this video, the system variable is set to automatically reset when the dead man switch is pressed again.

Pressing and releasing the Emergency Stop (E-stop) will also be recorded as an error.

DCS (Dual Check Safety) errors: Monitor the robot's speed and position to prevent collisions.

Steps to clear DCS errors:

1. Release all buttons.
2. Re-enable the dead man switch (press to the middle position).
3. Press and hold the Shift key.
4. While holding the Shift key, press the Reset button.
5. Release the Reset button, but continue holding the Shift key.

6. Move the robot out of the dangerous area in manual (Jog) mode, moving in the opposite direction of what caused the error.
7. Be clear about what you are doing when operating in manual mode and avoid blindly pressing buttons.

Program Management

If you want to run another program after testing one, you need to Abort all running programs.

Go to the Function menu (top right of the teach pendant) and select Abort All. This is a common operation to ensure the system is in a clean state.

If a program is paused, it cannot be deleted. The system is like a room with someone inside; it needs to be cleared before operations can be performed.

Aborting all programs does not change the robot's current position unless there are program or manual operation commands.

If you see an error message like INF i i n t f 105, it indicates that a program is currently running.

Singularity

In the World Coordinate System, the robot may enter a singularity when the fourth axis (J4) and the sixth axis (J6) are roughly in a straight line.

At a singularity, the robot controller encounters difficulty in calculating how to move the Tool Center Point (TCP) because there are an infinite number of joint angle combinations that can achieve the same end position.

Symptoms of Singularity: When attempting to move the robot along certain directions (e.g., the Z-axis) in the world coordinate system, the controller will display a MOTN-230 Singularity error, and the robot will not move.

Methods to resolve singularity issues:

1. Switch to the Joint Coordinate System.
2. Slightly rotate the fifth axis (J5) by approximately 5 degrees.
3. Switch back to the World Coordinate System, at which point the robot should be able to move normally. This operation can break the alignment of J4 and J6 into a straight line.

4. This method can solve the singularity problem while keeping the Tool Center Point roughly unchanged.

Okay, I'm happy to teach you how to program Fanuc robots. Based on the information I have reviewed, here are some basics and key concepts of Fanuc robot programming.

1. Creating and Editing Programs

To create a new program, you need to enter the SELECT menu and then choose CREATE. You can name the program through the KEYBOARD in the OPTION menu. It is recommended that program names do not start with a number and do not contain spaces.

After creating the program, you need to click EDIT to enter the program editing interface.

In the editing interface, you can add various instructions, such as motion instructions, logic control instructions, I/O instructions, etc., through the INSTRUCTION menu (usually the F1 key on the teach pendant).

You can use the EDIT COMMAND menu for basic editing operations such as COPY, PASTE, REPLACE, INSERT, DELETE, REMARK. When using the replace function, be aware that it usually only applies to the current line where the cursor is located and the code after it.

2. Motion Types

Fanuc robots support a variety of motion types:

Joint Motion: This is the fastest way of moving. Each robot joint moves simultaneously to the target angle at its maximum possible speed. The path is not defined. In the program, you can add this by selecting the JOINT motion type. Joint motion is usually used to quickly move the robot to an approximate position or as a transition between TEACH points. When recording joint positions, the data is based on the movement angle of each joint. It is recommended to record the HOME (origin) position as a joint position because it is independent of the tool coordinate system and user coordinate system.

Linear Motion: The robot's Tool Center Point (TCP) moves from the starting point to the target point at a constant speed along a straight line path. In the program, you can select the LINEAR motion type. Linear motion is often used in applications requiring precise paths, such as welding, spraying, material handling, etc.. The speed unit for linear motion is usually millimeters per second (mm/sec).

Circular Motion: The robot TCP moves along a circular arc path. Fanuc's circular motion usually consists of one linear segment and two arc segments. To use the circular command, the motion type needs to be changed to CIRCLE or CIRCULAR ARC when teaching the points. CIRCULAR ARC motion (A command) allows defining arcs with free curve shapes by defining the start point, intermediate point, and end point of the arc. This is very useful for irregular curved paths, such as welding or spraying. The ordinary CIRCLE command (C command) usually requires defining two points on the circle.

3. Recording Positions

To record the robot's current position, you usually need to press the SHIFT key and the POINT (or TEACH) key simultaneously.

When recording a position, you can choose different POSITION REPRESENTATION methods as needed. Common methods include:

Cartesian Coordinates (XYZWPR): Based on the currently active User Frame and Tool Frame. If you try to run a position recorded under this coordinate system under another user coordinate system, problems may occur.

Joint Coordinates (JOINT): Based on the angle of each joint of the robot. Joint coordinates are more versatile because they do not depend on a specific tool coordinate system or user coordinate system.

Position Registers (PR): Position registers are a data type used to store robot position data. You can enter the DATA menu and then select Position Registers to create and edit position registers. You can name the position registers for easy identification. To record the current robot position into the selected position register, you can press the SHIFT and RECORD keys when the robot is at the target position. In the program, you can choose to use position registers to call previously recorded positions.

4. Coordinate Systems (Frames)

Understanding coordinate systems is crucial for precise programming.

World Frame: This is the robot's reference coordinate system, usually located at the center of the robot base. All other coordinate systems are defined relative to the world coordinate system.

Tool Frame (UTOOL): Also known as the Tool Center Point (TCP), defines the position and orientation of the tool tip relative to the robot's sixth axis flange. Correctly setting the tool coordinate system is essential because it determines the robot's actual movement trajectory when performing tasks. The tool coordinate system can be set by various methods, such as the Three Point Method or the Direct Entry Method. In Roboguide, the tool coordinate system can be set by editing the tool properties and moving the coordinate system icon. By default, the tool coordinate system is located at the center of the robot flange. The active tool coordinate system can be switched using the SHIFT + COORD key combination.

User Frame (UFRAME): Used to define the position and orientation of the workpiece or fixture in the world coordinate system. By defining the user coordinate system, you can program based on the workpiece coordinates. Even if the position of the workpiece in the workspace changes, you only need to re-teach the user coordinate system without changing all the points in the program. The user coordinate system can also be set by various methods, such as the Three Point Method. In Roboguide, it can be done by adding a Fixture and defining its position and orientation, and then creating a user coordinate system based on that fixture. The active user coordinate system can be switched using the SHIFT + COORD key combination.

5. Basic Commands

Setting Position Registers: You can use the `PR[n] = P[m]` instruction to assign the data of a position (`P[m]`) to a position register (`PR[n]`).

Data Registers (R): Used to store numerical values (integers, floating-point numbers). You can enter the DATA menu and then select Data Registers to create and edit data registers. Data registers are often used for counting, logical judgments, etc.. For example, the `R = R + 1` instruction will add 1 to the value of data register R.

Input/Output (I/O): Used to communicate with external devices.

Digital Input (DI) and Digital Output (DO): Used to control the on/off state of external devices or receive signals from external devices. You can use the `DO[n] = ON/OFF` instruction to control digital output and use the `IF DI[m] = ON THEN ...` instruction to check the digital input status.

Robot I/O (RI): Usually used for some functions of the robot itself, such as controlling the gripper or sensors of the end effector.

Calling Programs: You can use the `CALL program_name` instruction to call another subroutine within the current program.

Jumps and Labels: Used to control the program flow.

Label: Use `LBL[n]` to define a label in the program.

Jump: Use the `JMP LBL[n]` instruction to unconditionally jump to the specified label.

Conditional Branching: Execute different program segments based on conditions.

IF Statement: Use the `IF condition THEN ...` structure for conditional judgment. The condition can be comparing the values of registers, I/O status, etc..

SELECT Statement: Used to execute different program segments based on different values of a register, similar to the `switch` or `case` statements in other programming languages. You can use the `IF SELECT R[n] = constant THEN CALL program_name` structure.

Wait: Pauses the robot program execution until a specific condition is met or a certain period of time has passed. You can use the `WAIT condition` instruction to wait for a condition to become true (e.g., wait for a digital input signal to turn ON). You can also use the `WAIT time (sec)` instruction to wait for a specified amount of time.

Payload: Used to define the weight and center of gravity of the robot's end effector and workpiece. Correctly setting the payload can improve the robot's motion accuracy and safety. You can use the `PAYLOAD[n]` instruction to activate predefined payload parameters.

FOR Loop: Used to repeat a segment of code a specified number of times. Use the `FOR R[n] = start TO end BY step` and `ENDFOR` structure to define the loop.

6. Looping

You can use `IF` statements and `JMP` instructions to create basic loop structures. In addition, Fanuc also provides more convenient `FOR/ENDFOR` loop structures to repeatedly execute blocks of code.

7. Macros

Macros are small programs that can run in the background. They are usually used to perform logical operations or simple motion commands and can be quickly executed by assigning them to buttons on the teach pendant or manual functions. When creating a macro program, if it only contains logic and no motion, you need to ensure that the Group Mask is set to an asterisk (``). You can use the Macro option in the SETUP menu to set up macros.

8. Conditional Monitoring

Conditional monitoring allows you to monitor specific conditions (e.g., input signal status) in the background while the main program is running. If the set condition is met, it can trigger the robot to stop or perform other actions. You can use the MONITOR and MONITOR END instructions in the INSTRUCTION menu to define the program segment to be monitored and call another non-motion program (Non-Motion Job), such as a user alarm program, when the condition is met.

9. Menu Utility

The menu utility allows you to interact with the operator through the teach pendant during production. You can create custom Prompt Boxes, Yes/No Select boxes, or Select from List boxes. These interactions can be configured through the Menu Utility option in the SETUP menu. In the program, you can use the Macro (actually calling the menu utility function) under the INSTRUCTION menu to call these custom menus and get input from the operator. The operator's selection is usually stored in the specified data register, and you can use these return values for conditional branching in the program.

10. Offsets

Offsets allow you to make small adjustments to existing positions or repeat the same motion trajectory at different locations without re-teaching. You can use Position Offset Registers to add offsets to existing position data.

11. Program Checks and Production Setup

Before putting the robot into production, it is very important to perform program checks and setup. You can set program check items in MENU -> SETUP -> Program Select, such as checking if the robot is in the HOME (origin) position and whether the correct speed override is enabled. More detailed system configuration options, such as power failure recovery settings, automatic program start settings, and whether forced I/O is allowed in automatic mode, can be found in MENU -> SYSTEM -> Config.

12. Program Adjust and Reference Positions

The program adjust function allows you to make small adjustments to the robot's trajectory during program execution without re-teaching. This is very useful for compensating for slight deviations in workpieces or fixtures. The Program Adjust option can be found under MENU -> SETUP. Reference Positions (such as HOME (origin)) are important positions for robot safety and calibration. Reference positions can be defined and calibrated in MENU -> SETUP -> Reference Position.

13. User Alarms

User alarms allow you to customize the robot's alarm messages and handling methods. You can create non-motion macro programs to define alarm trigger conditions and the actions to be performed (e.g., stop the program, cancel motion, display custom error messages). You can use the UAlarm (User Alarm) instruction under INSTRUCTION -> MISCELLANEOUS to trigger user-defined alarms in the program.

14. Roboguide (Simulation Software)

Roboguide is Fanuc's robot simulation software that can simulate real robot working environments on a computer. It allows you to create, test, and optimize robot programs offline, and also perform collision detection, cycle time analysis, etc.. You can create a Roboguide workcell from an actual robot backup. In Roboguide, you can more easily create and edit tool coordinate systems and user coordinate systems, and perform 3D visual programming and debugging.

15. Navigation and Basic Operations

Familiarity with the teach pendant's operating interface is the foundation of programming.

MENU: Used to access various functions and settings of the robot.
SELECT: Used to select and create programs (Jobs).
EDIT: Used to edit the currently selected program.
DATA: Used to view and edit various data, such as registers and position registers.
I/O (Input/Output): Used to monitor and control input/output signals.
SYSTEM: Contains the robot's system configuration and variables.
ALARM: Used to view current and historical robot alarm information.
POSITION: Used to view the robot's current position information, which can be displayed in different coordinate systems.
TEACH Mode (usually activated by the T1 or T2 mode button on the teach pendant) is the mode for program teaching and manual operation. In teach mode, the enabling switch (Deadman Switch) needs to be pressed to move the robot.
AUTO Mode is the mode for the robot to automatically run programs. In automatic mode, the program is usually started by the start button (Cycle Start).
STEP Mode allows the program to be executed step by step for debugging.

CONTINUOUS Mode allows the program to run continuously.
COORD key is used to switch the coordinate system (e.g., Joint, World, Tool, User) during manual jogging. Not all robot configurations allow jogging in all coordinate systems.

16. Error Codes

When the robot encounters an error, an error code will be displayed. Understanding basic error codes helps in diagnosing problems. For example, the MOTN-017 error usually accompanies a three-digit number, and you can use this number to determine which axis and what type of error occurred (e.g., motion limit error).

17. UOP (User Operator Panel) Communication with Allen Bradley CompactLogix via Ethernet/IP

Fanuc robots can communicate with external devices (e.g., Allen Bradley PLC) via Ethernet/IP. This requires corresponding configuration, including setting the robot's IP address, configuring the Ethernet/IP interface (usually done in MENU -> I/O -> Ethernet IP), and corresponding configuration on the PLC side. Through UOP (User Operator Panel) signals, the PLC can control robot operations such as start, stop, emergency stop, and monitor the robot's status.

Week 1: What is a Robot and its Purpose

Various Applications of Robots:

- Material Handling
- Palletizing
- Picking and Placing
- Material Removal: e.g., Grinding, Spray Painting
- Welding

Popularity of Robot Applications: Automotive Industry, Pharmaceutical Industry, RV Industry, etc..

Importance of Robots in the Post-Epidemic Era: Automation, Reliability, No Sick Leave, Maintaining Cleanliness, Improving Part Quality.

Discussion on whether robots replace or create jobs.

Do not be limited by known applications; more new robot applications may emerge in the future.

Material Handling is the largest application.

Robot Terminology

Maximum Space: The furthest range that all parts of the robot can reach, even if operation may not be possible at that position.

Restricted Space: An area that restricts robot activity for safety or other reasons. May be limited by protective devices or software.

Operating Space: The area where the robot actually works.

Robot Safety - First Principles

Operator safety is paramount. Robots and parts can be replaced, but people are irreplaceable.

Safety Priority of Robot Systems:

1. Operator
2. Robot

3. External Devices: e.g., Signal Lights, Fixtures, Controller
4. Tooling: Tools installed at the end of the robot

5. Workpiece: The object handled or moved by the robot

Safety Devices and Measures

Teach Pendant: With Emergency Stop (E-stop) button.

Dead Man Switch: Safety switch on the teach pendant with three states: fully released (stop), fully pressed (stop), middle position (allows movement).

Light Curtains: Can stop the robot when someone enters the work area.

Safety Fences.

Pressure Mats: Detect if someone enters a dangerous area.

Interlocks: e.g., requiring turning two keys simultaneously to start.

Warning Lights.

Motion Limits: Physical limit switches or software settings to prevent the robot from exceeding safe limits.

Collaborative Robots (Cobots)

Equipped with additional pressure sensors, safety sensors, and speed sensors.

Able to stop automatically if colliding with a person.

Safety Precautions for Entering the Robot Work Area

Always carry the teach pendant or some emergency stop device.

Usually have safety doors and guardrails with interlocks.

Follow the safety guidelines of the Robotics Industries Association (RIA).

Maximum Envelope is the robot's entire potential range of motion.

Operating Envelope is the area where the robot actually works.

Anti-Tie Down Logic: Prevents bypassing safety mechanisms through human intervention.

Axis Limits

The range of motion (angle or position) of each robot axis can be limited by programming.

Usually expressed in angles (e.g., +/- 180 degrees).

Limits can be further reduced based on manufacturer settings.

A robot restart is usually required after changing axis limits.

There are Hard Stops (physical limits) and Limit Switches (electrical detection limits).

Limits for Axes 4 and 6 are usually software limits.

Axis 5 has hard stops to prevent self-bending.

Fanuc Dual Check Safety (DCS)

An additional layer of safety.

Checks the robot's speed and position.

Has functions such as position check, speed check, and Cartesian coordinate system speed check.

DCS can stop the robot if it is about to enter an unsafe area or move too fast.

DCS Errors may occur.

DCS can define Restricted Zones, and the robot will stop when it leaves these zones.

Important Safety Precautions

Never wear loose clothing or jewelry.

Visually Inspect the robot and work cell.

Never enter the robot's work envelope while it is powered on.

In educational environments, never start the robot in automatic mode without permission.

Always close the safety door when moving the robot in automatic mode.

Exercise caution when entering the work area in manual or teach mode, and do not stay there for too long.

Lock Out Tag Out procedures can be used when maintenance is required on the robot.

Always perform lock out tag out and release any residual energy (flywheels, capacitors, counterweights, springs) when replacing components such as servo motors.

Programming Considerations

Take programming precautions to ensure safety.

Ensure Handshaking between the robot and external devices.

Use Motion Limits to protect equipment and personnel.

Pay attention to Override Speed and visualize the robot's motion trajectory during programming.

Keep the work area clean.

Ensure circuit protection is in place.

Home Position and Interference Zones are more advanced safety programming concepts.

General Robot System Components

Mechanical Unit: Robot body.

Controller: The brain of the robot, runs the software.

Peripheral Equipment: e.g., Teach Pendant.

Teach Pendant

Used to program the robot.

Similar to a large game controller.

Programming method: Teach the robot to reach a certain position, and then press a button to store the position and motion method.

Connected to the controller via a cable.

Has an Emergency Stop (E-stop) button and a Dead Man Switch.

Contains various buttons, each with a unique function.

The Shift key is used to execute the blue-labeled functions.

Menu is used to access various settings and configurations.

Cursor Keys are used to navigate menus.

Step Mode: Allows step-by-step execution of the program.

Reset button is used to clear alarms.

Enter button.

Previous button.

Soft Keys.

Position button and I/O button.

Older controllers have indicator lights, while newer controllers have status lights at the top (LED).

Forward and Backward buttons are used to run the program.

Combining the Shift key and direction keys (J1-J6) allows manual teaching (Jogging) of the robot.

Modern teach pendants are usually touchscreens and can connect mice and keyboards.

Mechanical Unit and Servo Motors

Robot motion is driven by AC Servo Motors.

Servo motors have Serial Pulse Counters that accurately know the motor's position.

When the robot is powered on, its position can be known through these counters.

Servo motors have a Mechanical Brake. The brake releases when power is on and engages when power is off.

The robot may drift if power is off and the brake fails.

Robot joints need lubrication, not the servo motors themselves.

The robot is programmed to know the Home position, but there is no true origin at the factory; it needs to be set by the user. Absolute Zero Setup is a physical reference position used for calibration.

The robot body contains batteries (C-type) to power the encoders when power is off, maintaining position memory.

Robot Axes

Axis 1 (J1): Rotational movement of the base (torsion).

Axis 2 (J2): Bending movement of the arm.

Axis 3 (J3): Up and down movement of the arm (shoulder).

Major Axes: Axes 1, 2, and 3, responsible for large movements.

Axis 4 (J4): Torsional movement of the wrist.

Axis 5 (J5): Bending movement of the wrist (pitch).

Axis 6 (J6): Rotational movement of the flange (roll).

Minor Axes: Axes 4, 5, and 6, responsible for fine posture adjustment.

Axis 7 (J7): e.g., the axis of a gantry or rotary table on which the robot is mounted (not covered in this course).

Each axis can move independently or in a coordinated manner.

Mentioned Spider Robots, usually with only three axes, used for high-speed picking and placing.

Axis Motion Range and Limits

Each axis has its range of motion, which may be limited by hard stops and limit switches.

The motion range of Axes 4 and 6 is usually limited by software.

The motion range of Axis 5 has hard stops to prevent excessive bending.

Controller

Contains power supply, transformer, memory, etc..

Contains an I/O Board for communication with external devices.

Different controller sizes: Type A, Type B, Open Air Mate, Type Mate.

The controller usually has an Operating Panel containing:

Fault Reset button.
Teach/Run mode switch key.

Go (Start) button.
Fault LED (Fault Indicator Light).

Operator Panel

Contains User Buttons that can be customized for functions (e.g., running macros).

Local/Remote mode switch: Local for testing, Remote for production running via PLC or other controllers.

Software

Pre-installed in the controller.

Be sure to Back Up with Images.

Mode Select:

T1 Mode: Limits speed to 250 mm/s.
T2 Mode: Test mode, allows up to 100% speed, but still in teach mode.
Auto Mode: Full-speed operation.

The behavior of the safety door when triggered differs in different modes: in automatic mode, it will stop and brake; in T1/T2 mode, the safety door can be bypassed (because someone may be inside operating).

Peripheral Equipment

PLC (Programmable Logic Controller).

End-of-Arm Tooling.

Human Machine Interface (HMI).

Cameras and Vision Systems.

More Details on the Teach Pendant

The middle position of the Dead Man Switch allows movement.

Each button has its specific function.

Session 1 has detailed descriptions of all menus.

Pressing and holding the Shift key and cursor keys allows quick skipping of menu items.

Step Mode is used to execute the program line by line.

Familiarize yourself with the three commonly used buttons: Previous, Reset, Enter.

The function of Soft Keys changes according to the current screen.

The Position button displays the current position.

The I/O button displays the status of input/output signals.

Starting the Robot

Usually done through a switch.

Jog Speed

Override Speed: Expressed as a percentage, it is the percentage of the current maximum allowable speed.

For example, in T1 mode (maximum 250 mm/s), a 50% override speed means the maximum speed is 125 mm/s.

There are different Jog speeds, including Fine Speed that can move one servo pulse at a time.

Manual Teaching (Jogging)

Joint Mode: Press and hold the Shift key and the buttons labeled J1-J6 to control the forward or reverse movement of each joint individually. Usually used when clearing jams or reconfiguring the robot.

World Mode: Moves the Tool Center Point (TCP) in the directions of the Cartesian coordinate system (XYZ), the movement trajectory is a straight line.

Tool Mode: Moves the TCP relative to the tool coordinate system.

User Mode: Moves the TCP relative to the user-defined coordinate system.

Jog Frame Mode: Not commonly used.

Switching between coordinate systems is done by pressing the Chord button (or the corresponding button and indicator light on older controllers).

Cartesian Coordinates

Used in World mode, Tool mode, and User mode.

All movements are relative to an Origin.

The origin of the world coordinate system is usually located at the intersection of the J1 and J2 axes.

The Right Hand Rule is used to determine the positive directions of the X, Y, and Z axes.

Singularity

Occurs when certain axes of the robot become too aligned.

Near a singularity, the robot may not move as expected.

The method to resolve singularity issues: In joint mode, slightly adjust the J5 axis (+/- 10-15 degrees).

Fanuc Robot Programming Basics

Creating and Editing Job (Program)

Create a new Job through the `SELECT` menu.

Use the `CREATE` option and name it (recommended not to start with a number and no spaces).

Add comments in `DETAIL`, set `Group Mask` (1 for motion job, for non-motion job).

Enter the editing interface using `EDIT`.

Add instructions through the `INSTRUCTION` menu.

Recording and Calling Positions (Position Registers)

Enter the `DATA` menu, select `POSITION REGISTERS` to create and name.

Use `SHIFT` + `RECORD` on the teach pendant to record the current position to the selected position register.

Position registers can be identified by name.

Position data can be stored in Cartesian coordinates (XYZ WPR) or joint angles (Joint).

Cartesian coordinates are based on the user coordinate system and tool coordinate system.

Joint angles are based on the angle of each joint of the robot and are more versatile.

Use `INSTRUCTION` -> `MOTION` -> `JOINT`/`LINEAR` and select `POSITION REGISTER` to call recorded positions in the Job.

Data Registers

Used to store numbers (integers, can be positive or negative).

Similar to dents, integers, word, double word in PLC programming.

The values of data registers can be viewed and modified through the `DATA` menu.

Mathematical operation instructions (`ADD`) can be used to manipulate data registers in the program.

Motion Types

Joint: Each joint moves independently to the target position at the fastest speed, the path is undefined.

Linear: The Tool Center Point (TCP) moves to the target position in a straight line path, speed is in millimeters per second (mm/sec).

Circular/Arc: Used to create arc trajectories, usually for welding applications.

An arc is defined by three points: start point (usually the end point of the previous Linear or Joint motion), intermediate point (a point on the arc, change the motion type to `CIRCLE` through `EDIT` -> `DEFINITION`), end point (defined by moving the robot and `TOUCH UP`).

Completing a full circle usually requires two consecutive circular arc motions.

Speed is similar to linear motion.

The `CIRCULAR ARC (A)` command can be used to create more free-form curved motions, requiring consideration of multiple intermediate points.

Speed

The speed of joint motion and circular motion is usually expressed as a percentage (%).

The speed of linear motion is usually expressed in millimeters per second (mm/sec).

Can be adjusted using the speed adjustment buttons on the teach pendant.

Termination Type

Fine: The robot will stop precisely at the target position.

Continuous: The robot will not completely stop at the target position but will smoothly pass through, used to improve motion efficiency.

Common Instructions

Motion: `JOINT`, `LINEAR`, `CIRCLE`.

I/O: `DO` (Digital Output) used to control external devices.

I/O status can be configured and viewed through the `I/O` menu.

Descriptive information for I/O can be set.

Register: Use mathematical instructions (`R[ ] = R[ ] + Number`) to operate data registers.

Branching: `JMP LBL` (Jump Label) used to jump to the specified label.

`LBL` (Label) defines the jump target in the program.

Conditional Branching: `IF`/`SELECT` executes different program segments based on conditions.

`IF R[ ] = Constant, JMP LBL [...]`.

`SELECT R[ ], CASE Constant, CALL Program [...] ELSE, JMP LBL [...]`.

Wait: `WAIT Condition` pauses the program until the specified condition is met.

Can wait for the status of digital input (`DI[ ] = ON/OFF`).

Can wait for the value of a register (`R[ ] =/>/< Constant`).

Multiple wait conditions can be combined using `AND` and `OR`.

A `TIMEOUT` can be set to perform a specified operation after the wait times out.

Payload: `PAYLOAD [Number]` is used to set the current robot payload, affecting motion performance.

Different payload parameters can be configured in `MENU` -> `SYSTEM` -> `MOTION` -> `PAYLOAD`.

Remark: Used to add comments to the program to improve readability.

Use `INSTRUCTION` -> `MISC` -> `REMARK` to add.

Use `EDIT COMMAND` -> `REMARK` to comment out one or more lines of code, making them not executed (displays `//` at the beginning of the line).

User Alarm: `U_ALARM [Number]` triggers a user-defined alarm.

User alarm macros need to be defined in a non-motion type Job (`GROUP MASK` as ``).

User alarm information can be configured in `MENU` -> `SYSTEM` -> `ALARM`.

Call: `CALL [Program Name]` is used to call other programs in the current program.

Offset: Used to perform offsets based on existing positions.

The `OFFSET, PR[ ]` instruction can be added after a motion instruction to add a position offset.

The `OFFSET CONDITION` can be used to define offset conditions at the beginning of the program.

For/Endfor: `FOR R[ ] = Start Value TO End Value STEP Step Size` and `ENDFOR` structures are used to create loops.

User Frames and Tool Frames

User Frame: Defines the reference coordinate system for the workpiece or work environment.

Can be set through `MENU` -> `SETUP` -> `FRAMES` -> `USER FRAME`.

Can be defined using the three-point method or other methods.

Defined by recording the origin, a point in the positive X-axis direction, and a point on the XY plane.

After setting, the robot's Cartesian motion will be based on this coordinate system.

Tool Frame: Defines the reference point (usually the Tool Center Point TCP) and orientation of the robot's end effector (tool).

Can be set through `MENU` -> `SETUP` -> `FRAMES` -> `TOOL FRAME`.

Can be defined using the three-point method (Approach Point 1, 2, 3) or by directly entering the offset.

A correct tool frame is crucial for precise robot motion.

Press `SHIFT` + `COORD` on the teach pendant to switch the currently active coordinate system (World, User, Tool, etc.).

Program Adjust

Allows fine-tuning of existing programs without re-teaching.

Can be set in `MENU` -> `SETUP` -> `ADJUST`.

Adjustments can be made relative to the tool frame in the XYZ axes and rotation directions.

Adjustments can be temporary or permanent.

Reference Positions

e.g., Home Position.

These predefined reference positions can be called in the program.

I/O (Input/Output)

Digital I/O

Used to communicate with external devices (such as PLC, HMI, sensors, cylinders, etc.).

Can be viewed and configured through `MENU` -> `I/O` -> `DIGITAL`.

Names and comments can be set for each digital I/O.

Use `DO[ ] = ON/OFF` to control digital outputs in the program.

Use `DI[ ] = ON/OFF` to detect the status of digital inputs.

Robot I/O

Refers to the dedicated I/O on the robot controller, usually used to connect devices on the end effector (such as gripper sensors, pneumatic valves, etc.).

Commands are similar to digital I/O.

Logical NOT can be represented using parentheses and exclamation marks `(!DI[ ])`.

Ethernet I/O

Requires Fanuc's Cat code (option) to enable.

Allows communication with devices such as PLCs (e.g., Allen Bradley CompactLogix) via Ethernet.

The `Ethernet IP` option will appear in `MENU` -> `I/O`.

Parameters such as IP address, Rack, Slot need to be configured.

Data is transmitted in units of Word (16 bits).

UOP (User Operator Panel) signals can be used to control robot operation (such as start, stop, enable, etc.) via Ethernet I/O.

Conditional Monitoring

Use `MONITOR START` and `MONITOR END` instructions to monitor input signals during specific parts of the program.

For example, check if the suction cup has picked up the object before picking it up.

Non-motion related monitoring logic is recommended to be placed in the main program, and motion commands in subroutines, to improve modularity.

Program Flow Control

Jump Label

Use `LBL [Number]` to define a label.

Use `JMP LBL [Number]` to jump to the specified label.

Can be used to implement program loops.

Conditional Branching

IF Statement: `IF Condition THEN Execute Operation`, can have an `ELSE` branch.

SELECT Statement: Executes different `CASE` branches based on the value of a register, can have an `ELSE` branch to handle unmatched cases.

Wait

`WAIT Condition` pauses the program until the condition is met.

Can wait for digital inputs, register values, etc..

Multiple conditions can be combined using `AND` and `OR`.

A timeout period can be set with `TIMEOUT LBL [Number]`, jumping to the specified label after the timeout.

Looping

Use `JMP LBL` to implement simple infinite loops or loops with a limited number of repetitions (combined with a counter).

Use the `FOR R[ ] = Start Value TO End Value STEP Step Size` and `ENDFOR` structures to implement structured loops, which are easier to manage and understand.

`FOR` loops can be nested.

The code inside the loop will be executed repeatedly for the specified number of times.

Roboguide Software

Installation and Setup

Requires downloading the installation file, which is large and may take a long time to install.

Different robot models and software versions can be selected during installation.

A new Cell can be created from scratch, or it can be imported from existing backup files.

Creating a Cell (Workstation)

A new workstation can be created through `File` -> `New Cell`.

A virtual workstation identical to the real robot configuration can be created from the robot controller's backup file (.zip).

On the real robot's teach pendant, go to `MENU` -> `FILE` -> `UTILITY` -> `SET DEVICE`, select the backup medium (e.g., USB), and then select `BACKUP EVERYTHING`.

In Roboguide, select `Create a file from backup` and choose the backup file.

The created Cell contains information such as the robot model and Dual Check Safety (DCS) areas.

Importing and Configuring Tools (Grippers)

Right-click on IOTool_1 (or other tools) under the Tooling tab and select `IOTool Properties`.

CAD models can be imported as the robot's end effector.

The scale and position of the CAD model can be adjusted.

Gripper actions (such as opening and closing) can be set.

Teach Pendant Simulation

Roboguide has a built-in virtual teach pendant.

You can set to use a physical teach pendant through `Window` -> `Tools` -> `Options` -> `Teach`.

Connecting a physical teach pendant requires specific connection cables and power adapters.

Press and hold `SHIFT` + `I` + `POSITION` while starting up on the physical teach pendant to enter the settings interface and select `ROBOGUIDE MODE`.

Ensure that the robot software version in Roboguide matches the version of the physical teach pendant.

The virtual teach pendant can switch between new and old interfaces (`iPendant`/older pendant).

The button functions on the virtual teach pendant are similar to those on the real teach pendant (such as Jogging, Step/Continuous, Fault Reset, etc.).

Coordinate System Navigation

You can Jog (manually move the robot) using `SHIFT` + axis buttons (X, Y, Z, W, P, R) on the virtual teach pendant.

The `COORD` menu is used to select different Jogging coordinate systems (World, Joint, Tool, User, Jog Frame).

World: Fixed coordinate system based on the robot base center.

Joint: Based on the independent movement of each joint.

Tool: Based on the currently active tool coordinate system.

User: Based on the currently active user coordinate system.

Jog Frame: User-defined Jogging coordinate system.

If the World coordinate system is not available, it can be enabled by modifying the system variable `$SCR_GRP.$COORD_MASK`.

This variable is a binary mask, with each bit corresponding to an available coordinate system.

For example, the decimal value 31 (binary 11111) indicates that all commonly used coordinate systems (Joint, Jog Frame, World, Tool, User) are enabled.

You can roughly move the robot model in Roboguide by dragging it with the mouse.

Program Node Map

You can toggle whether to display auxiliary elements such as points and lines in the program through `View` -> `Program Node Map`.

The displayed position marker types (`Position Triads`, `Position Connector Lines`) can be adjusted.

Double Monitor Display

You can use `SHIFT` + `DISPLAY` to simultaneously display two different information windows (such as program and I/O status) on the teach pendant screen.

Program Editing Tips and Utilities

Edit Command

Located in the `EDIT` menu at the bottom of the teach pendant screen.

Replace: You can batch replace specified motion attributes (such as speed) in the program.

The replace operation proceeds downwards from the current cursor position. To replace the entire program, the cursor needs to be moved to the top of the program.

Copy and Paste: You can copy and paste one or more program lines.

Use `SELECT` to select the lines to be copied.

`COPY` copies the selected lines.

`PASTE` pastes the copied content at the current cursor position. You can choose to paste logic (`LOGIC`) or position ID and logic (`POSITION ID`).

Insert: You can insert a new line at the current cursor position.

Delete: You can delete the line where the current cursor is located or multiple selected lines.

Remark: You can use `EDIT COMMAND` -> `REMARK` to comment out the selected lines of code, making them not executed (`//` is displayed at the beginning of the line).

Macros

Small programs used to perform tasks in the background, can contain logic or motion commands.

Macros can be set in `MENU` -> `SETUP` -> `MACRO`.

Macros can be assigned to user keys or manual functions on the teach pendant.

Non-motion type macros (`GROUP MASK` as ``) can be run under normal user keys.

Macros can be called using `INSTRUCTION` -> `MISC`.

Menu Utility

Custom menus can be created through `MENU` -> `SETUP` -> `MENU UTILITIES`.

You can create menu types such as `Prompt Message`, `Yes/No Select`, `Select from the list`.

Operators can interact with the program through these custom menus during runtime, for example, selecting subroutines to be executed.

Advanced Topics

Payload

Refers to the total weight of the robot's end effector and workpiece.

Correct payload settings can improve the robot's motion accuracy and performance and prevent overloading.

Can be configured in `MENU` -> `SYSTEM` -> `MOTION` -> `PAYLOAD`.

Different payload schemes can be set and switched in the program.

Introduction to Machine Vision

Machine vision systems are used to detect, identify, and measure objects.

Includes components such as light sources, cameras, and image processing software.

Light Source: Different types of light sources (diffuse light, specular light, backlight, etc.) and colors will affect image quality.

Camera: Used to capture images.

Filters: Can filter out specific wavelengths of light and enhance image contrast.

Selecting the appropriate light source and camera is crucial for the success of machine vision applications.

Using Fanuc UOP to Communicate with Allen Bradley CompactLogix via Ethernet I/O

Requires enabling the Ethernet I/O option.

Communication is achieved by mapping UOP signals (such as command enable, start, stop, etc.) between the PLC and the robot controller.

UOP signals correspond to specific digital input/output channels.

Corresponding configurations are required on both the PLC and the robot sides.

Coordinate Mask Setup

The system variable `$SCR_GRP.$COORD_MASK` controls the types of coordinate systems available in Jogging mode.

By modifying the value of this variable (integer, corresponding to different binary bit combinations), specific coordinate systems can be restricted or enabled (e.g., disabling World Jogging).