Cobots

Rapid implementation of Cobots in industrial environment

www.continental-corporation.com

Content

A. **Theoretical Module**

B. **Practical Module**

Content

A. Theoretical Module

- General Presentation
- Cobot Selection
- Mechanical Installation
- Griper Development
- Electrical Installation
- Software of Cobot
- Design of Cobot Program

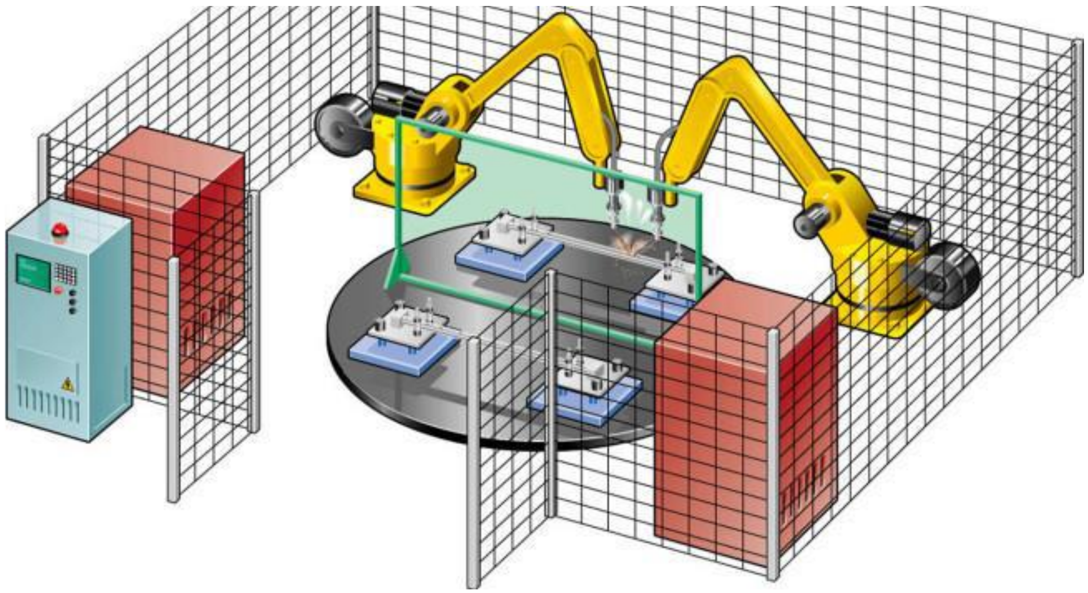
Content

A. Theoretical Module

- General Presentation
- Cobot Selection
- Mechanical Installation
- Griper Development
- Electrical Installation
- Software of Cobot
- Design of Cobot Program

General Presentation

- Mainly, **Cobots** or **Collaborative Robots** are considered as being complex machines which work hand in hand with human beings. In a shared work process, they support and relieve the human operator.
- **Cobots** are specifically designed for working with people in a shared work space.
- **Collaboration** is determined by application and workspace, not by the robot itself;



General Presentation

Broadly, there are three categories of robots:

1. The traditional **industrial robot** that typically operates in a caged environment, working with high payloads and/or speeds. It is relatively well-understood in terms of its strengths, opportunities, research challenges and growth potential.
2. **Cobots** or collaborative/cooperative robots which typically interact closely with humans and have much lower payloads. They can be fixed or mobile.
3. **Software bots** have few if any physical characteristics, but can respond to a variety of inputs such as automated queries. This category more generally includes developments in Robotic Process Automation (RPA).

General Presentation

- Comparation between Industrial robot and cobot:

	Robot	Cobot
Payload	+++	+
Reach	+++	+
Position Accuracy	+++	++
Speed	+++	+
Human-robot interaction (safety)	+	+++
Simplicity of programming	++	+++
'Plug & produce' in production	+	+++
Task Variation (flexibility)	++	+++
Availability	+++	++

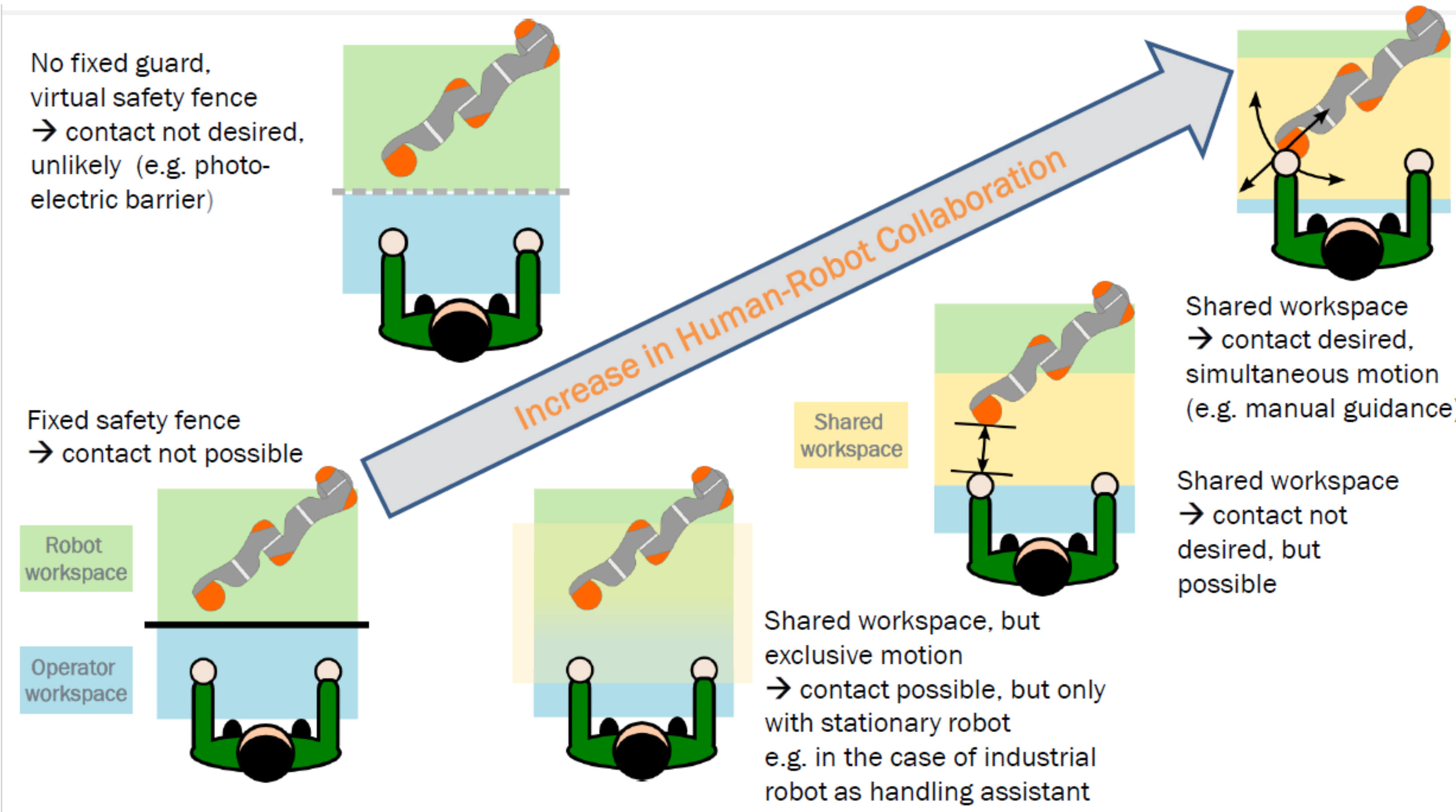
+: limited   ++: good   +++: very good

General Presentation

- The main advantages or **cobots** are:
  - Safe interaction with human (integrated sensors, passive compliance, overcurrent detection.);
  - Easy programming (can be easily taught by demonstration);
  - Flexibility of tasks (more dexterous and flexible).

- **Cobots** are coming with some limitations, also:
  - Lack of ability to detect abnormalities makes them less flexible compared to their human colleagues;
  - Limited operating speed;
  - Reduced payload;

- **Forms of Human-Robot Collaboration**



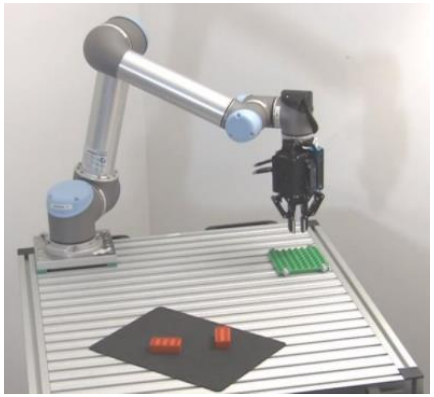
General Presentation

- **Human-Robot Collaboration:**

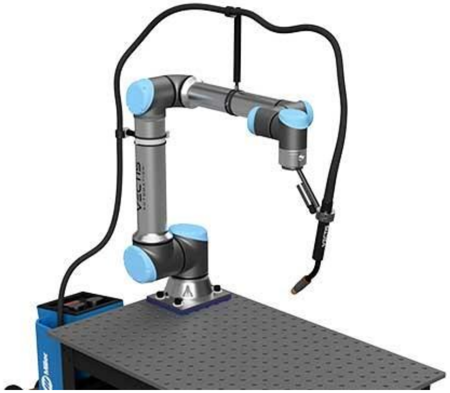
  - Enables variable automation of tasks for which complete automation is too expensive or too complex can thus be partially rationalized;
  - Rationalize tasks by combining strengths and dividing the work optimally between robot and human;
  - Non-ergonomic workstations can be improved by robots;
  - Reduction in repetitive and trivial work of operators, led to improved physical work conditions and give the workers room to do other potentially value creating tasks.

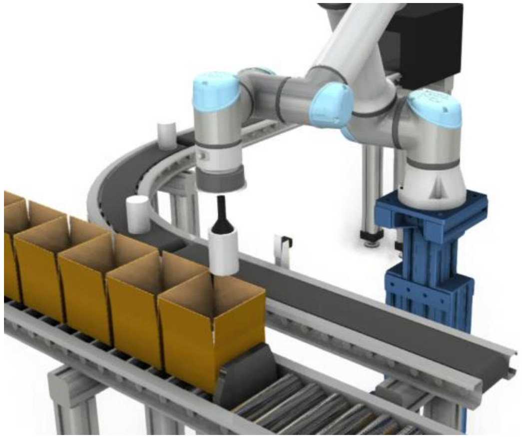
General Presentation

- **Common Applications for Cobots:**
  - **Pick and Place**: a workpiece is picked up and placed in a different location. Pick and place functions typical require an end-effector that can grasp the object. It could either be a gripper or vacuum cup effector.
  - **Machine Tending**: Means using a robot to load and unload a machine, such as a CNC milling machine, 3D printer, or labeling machine. These type of cobot applications may require the cobot to have input and output (I/O) interfacing hardware specific to the machine. The I/O hardware indicates to the robot the next cycle or when material needs to be replenished.



- **Common Applications for Cobots:**
  - **Packaging and Palletizing**: A subset of the pick and place is the packaging and palletizing of products: box assembly and loading; and box collating or placing onto a pallet for shipping. Conveyor tracking is required for this application to synchronize robotic movement with a conveyor. A vision system also may be needed for products with a non-uniform shape.
  - **Process Tasks**: any task that requires a tool to interact with a workpiece. Common examples are a gluing processing, dispensing, or welding. Each of these process tasks requires a tool to go down a fixed path repeatedly.





- **Common Applications for Cobots:**
  - **Finishing Tasks**: tasks that require operators require a manual tool and large amount of force. These finishing jobs can include polishing, grinding, and deburring. The robot can be taught manually or via computer programming methods. This is achieved through force sensing, either via the end-effector or internally.
  - **Quality Inspection**: usually involves full inspection of finished parts, high resolution images for precision machined parts, and part verification against CAD models. End-effectors with high-resolution cameras may be required for the inspection, as well as vision systems and software.

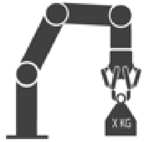
Content

A. Theoretical Module

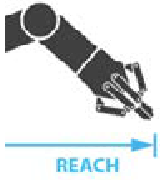
- General Presentation
- Cobot Selection
- Mechanical Installation
- Griper Development
- Electrical Installation
- Software of Cobot
- Design of Cobot Program

General Presentation

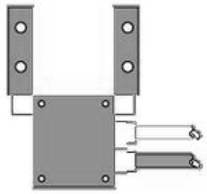
- Terms used related to cobots:
  - **Payload** is the weight that the robot can carry. Real payload = nominal payload minus the weight of the robot's end effector (gripper).



  - **Reach** is the distance that can be reached by the robot's wrist. This measurement is taken from the robot's base.

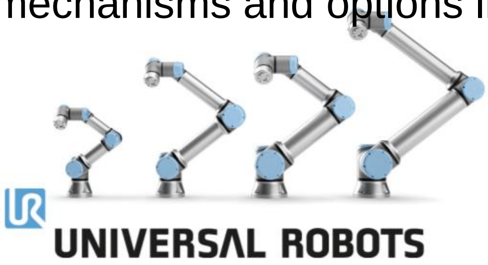


  - **End Effector** is the device at the end of a robotic arm, designed to interact with the environment. The exact nature of this device depends on the application of the cobot.



Cobot Selection

- **Selection** of the cobot is based on several considerations**:**
  - **Supplier**: for Continental, Universal Robot is standard supplier for cobots at this moment.
  - **Degrees of Freedom**: depending on complexity of cobot movements and other restrictions, an increased number of DoF must be required;
  - **Speed**: an increased speed is similar with higher output but require additional safety measures.
  - **Safety**: Depending upon project and how the Cobot interacts with humans you should look at the safety mechanisms and options like safety stop buttons, fall back capabilities etc.



Cobot Selection

- Based on all these requirements, most of the cobots used on our plant are UR5, with these caracteristics:

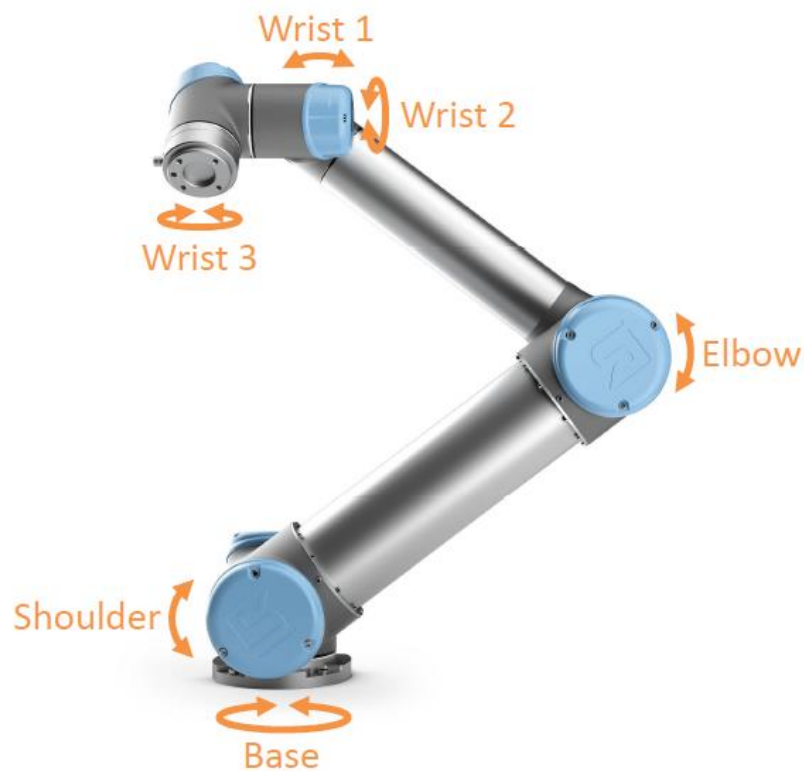
UR5

Degrees of freedom	6
Payload	5 kg
Weight	18.4 kg
Repeatability	+/- 0.1 mm
Reach	850 mm
Safety	TUV approved
Price	+/- 35,000 USD
Ease of programming	8/10

Cobot Selection

- UR5 Cobot is composed from 3 main elements:
  - Robotic arm;
  - Control box;
  - Teach penant.

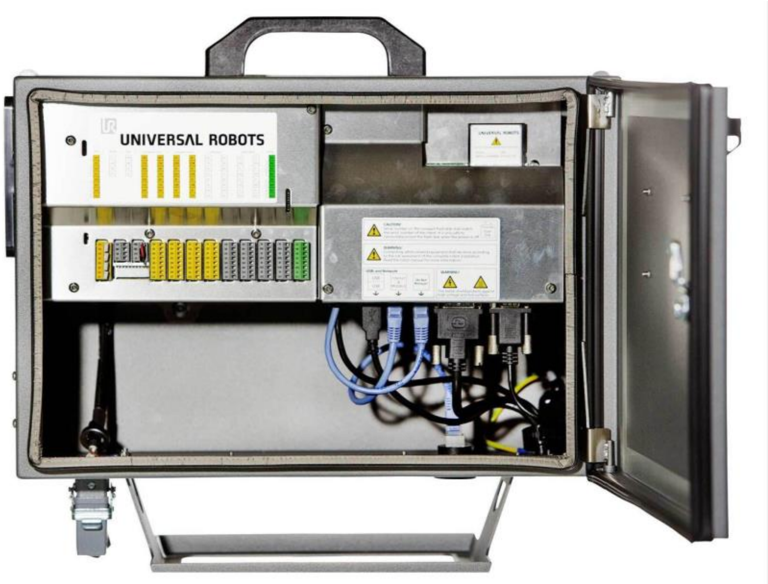
- UR5 robotic arm is composed of:
  - 6 axes;
  - Modular design
  - +/- 360° freedom;



Cobot Selection

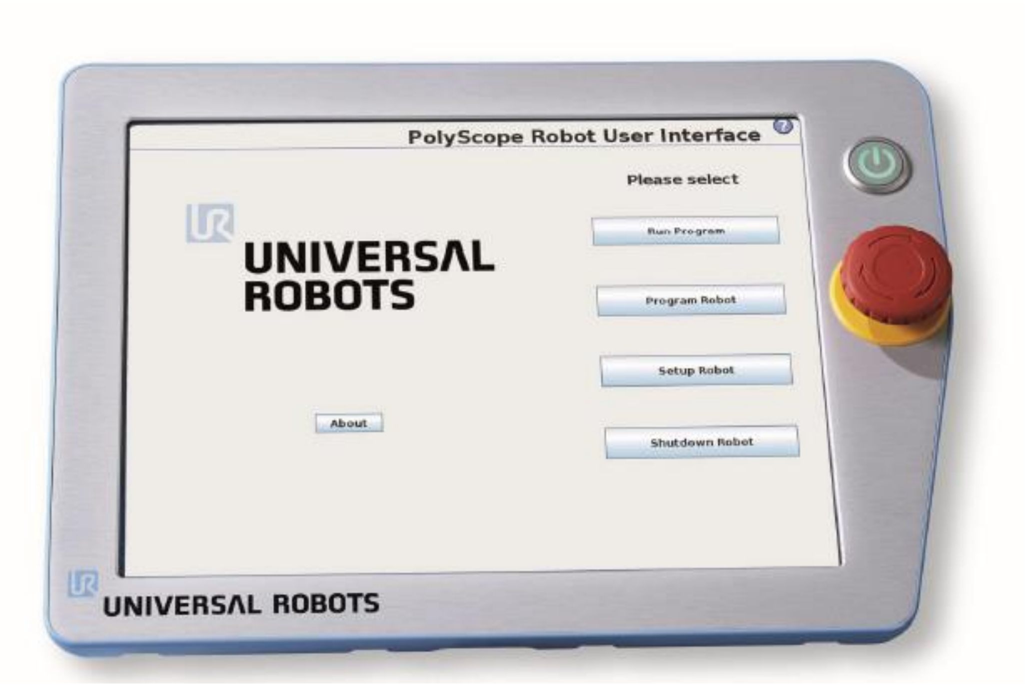
- Control box contain:
  - Flashcard/USB with software;
  - Power to robotic arm;
  - Safety system;
  - Communication to peripheral devices;

- Connectors:
  - Power 220/110 Vac;
  - Ethernet;
  - USB;
  - Robot arm.



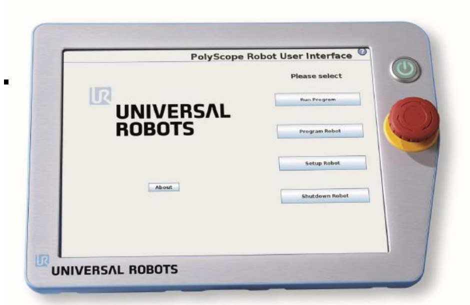
Cobot Selection

- Teach pendant – touch screen monitor:
  - Power button;
  - Emergency button
  - Freedrive button;
  - USB connector



Cobot Selection

- Teach pendant – touch screen monitor:
  - Power button – by pressing the power button at the front side of the panel with the touch screen, the Control Box will be turned On on. When the control box is turned on, text from the underlying operating system will appear on the touch screen. After about one minute, a few buttons appear on the screen and a popup guides the user to the initialization screen
  - Emergency button – activating the emergency stop button will immediately stop all robot motion. Emergency stop shall not be used as a risk reduction measure, but as a secondary protective device.
  - Freedrive button – activate Freedrive mode.
  - USB connector – for USB connecting drives/flash memories.



Content

A. Theoretical Module

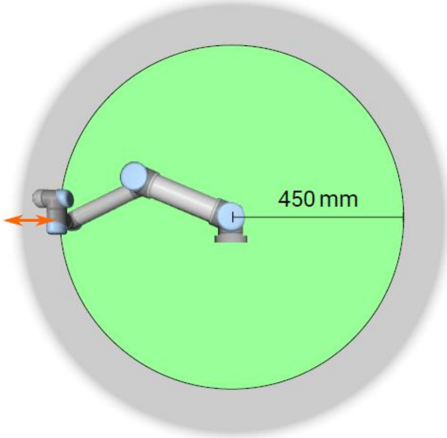
- General Presentation
- Cobot Selection
- Mechanical Installation
- Griper Development
- Electrical Installation
- Software of Cobot
- Design of Cobot Program

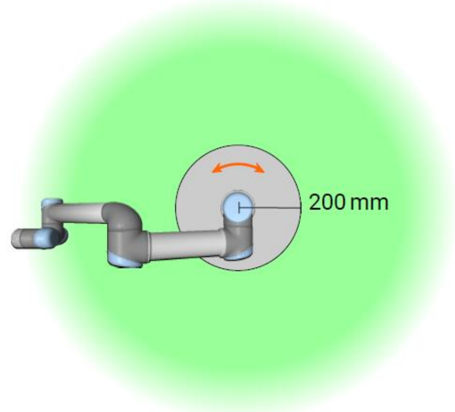
Mechanical Installation

- Before instalation of any cobot, we are responsible for ensuring that all the applicable safety laws and regulations are observed and that any significant hazards in the complete robot application are eliminated. This includes, but is not limited to:
  - Performing a risk assessment for the complete robot system;
  - Interfacing other machines and additional safety devices if defined by the risk assessment;
  - Setting up the appropriate safety settings in the software;
  - Ensuring that the user will not modify any safety measures;
  - Validating that the total robot system is designed and installed correctly;
  - Specifying instructions for use;
  - Marking the robot installation with relevant signs and contact information;
  - Collecting all documentation in a technical file; including the risk assessment and this manual;

Mechanical Installation

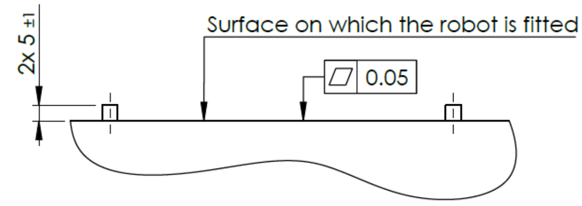
- There are two situation that must be taken into consideration at mounting, to be avoided due to pinching hazard:
  - One area is defined for radial motions when the wrist 1 joint is at least 450 mm from the base of the robot. As the robot stretches out, the knee-joint effect can give high forces in the radial direction (away from the base) at low speeds.
  - The other area (right) is within 200 mm of the base of the robot, when moving tangentially. Similarly, the short leverage arm, when the tool/end effector is close to the base and moving around the base, can cause high forces at low speeds.
- Pinching hazards can be avoided by removing obstacles in these areas, placing the robot differently, or by using a combination of safety planes and joint limits to eliminate the hazard by preventing the robot moving into this region of its workspace.



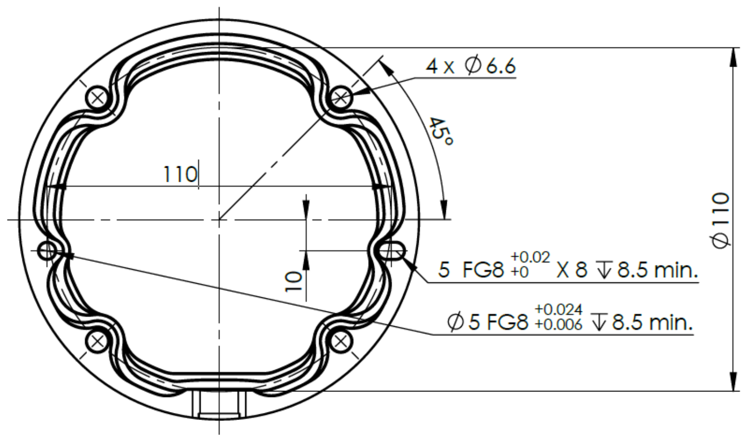


Mechanical Installation

- The **robot arm** must be mounted on a a sturdy, vibration-less, surface that can withstand at least ten times the full torque of the base joint and at least five times the weight of the Robot Arm.
- The mounting itself of the base of the **robot arm** require:
  - Solid surface;
  - Footprint 149 mm;
  - 4 M8 bolts.

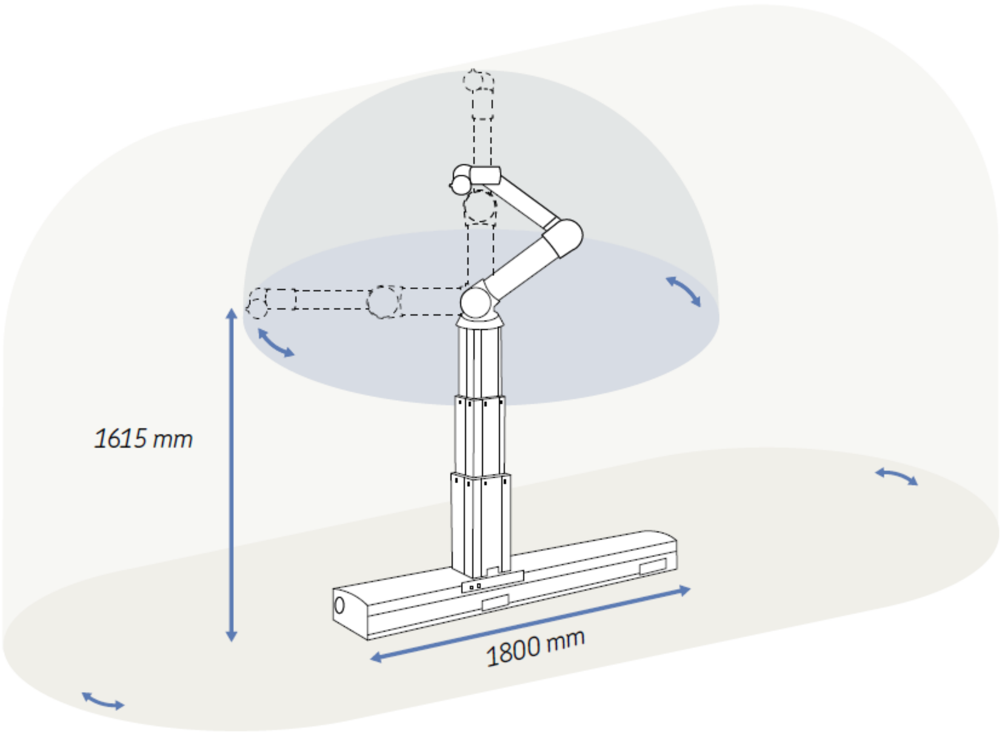


- The **Control Box** can be hung on a wall or placed on the ground. A clearance of 50 mm on each side of the Control Box is needed for sufficient airflow.
- The **Teach Pendant** can be hung on a wall or on the Control Box. Verify that the cable does not cause tripping hazard.



Mechanical Installation

- Due to the fact that the radius of action is usually limited by their reach, additional linear axes, vertical and/or horizontal, can significantly enhance the radius of action up to 5 times, by re-positioning the base of the robot during its working cycle.



Content

A. Theoretical Module

- General Presentation
- Cobot Selection
- Mechanical Installation
- Griper Development
- Electrical Installation
- Software of Cobot
- Design of Cobot Program

Griper Development

- **End Effector** is the device at the end of a robotic arm, designed to interact with the environment. The exact nature of this device depends on the application of the cobot.
- The **Pick And Place** is the most frequent application in Conti Sibiu plant. So, the grippers are most frequent end effectors.
- The **grippers** can be:
  - With 2-5 fingers;
  - Suction (vacuum) cups;
  - Magnetic;
  - Electrostatic;

Content

A. Theoretical Module

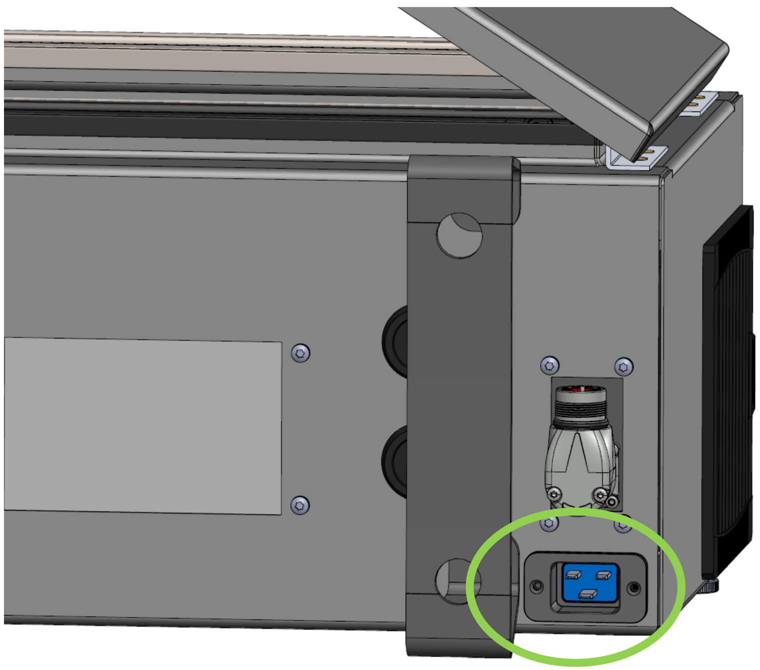
- General Presentation
- Cobot Selection
- Mechanical Installation
- Griper Development
- Electrical Installation
- Software of Cobot
- Design of Cobot Program

Electrical Installation

- Similar with mechanical installation, before any electrical works, we are responsible for ensuring that all the applicable safety laws and regulations are taken into consideration and that any significant hazards in the complete robot application are eliminated.
- The Control Box is conected with main power source and with additional equipment. From Control Box is powered the Robotic Arm, through a cable with locking connections, and the Teach Pedant.
- Main electrical interface groups for the Robot Arm are in the Control Box.

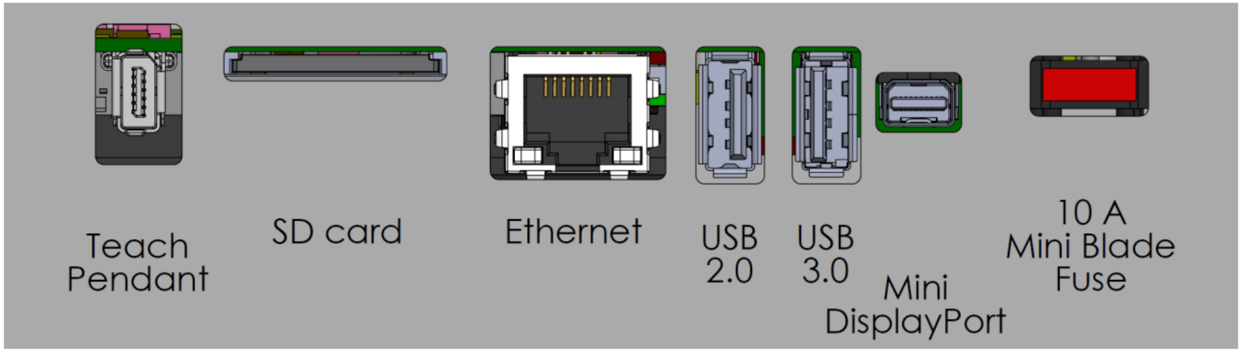
Electrical Installation

- Mains cable connect the Control Box to the energy source.
- It is recommended to install a main switch to power off all equipment in the robot application as an easy means for lockout-tagout under service.
- The mains supply is equipped with:
  - Connection to ground;
  - Main fuse;
  - Residual current device.



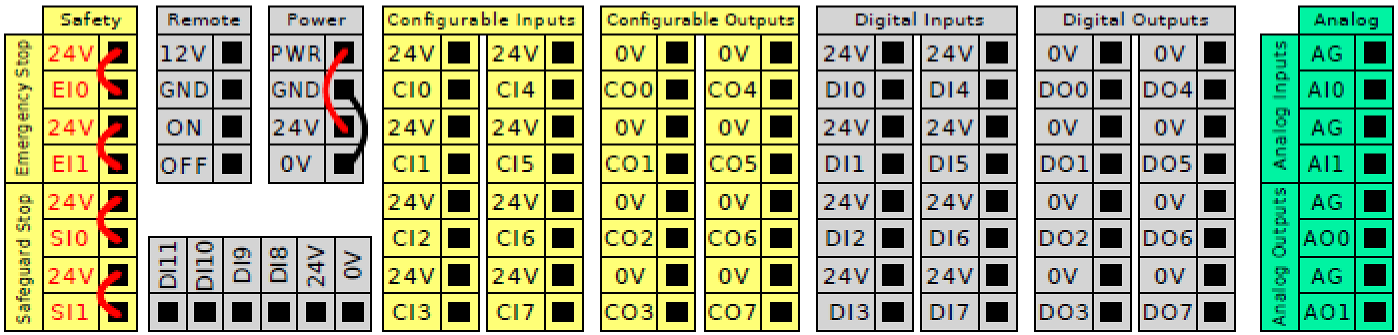
Electrical Installation

- Main electrical interface from Control Box are listed below:
  - Mains connection
  - Robot connection
  - Controller I/O
  - Tool I/O
  - Ethernet.
- The term I/O refers to both digital and analog control signals;



Electrical Installation

- Inside Control Box, there is a Controller **I/O** which allows a wide range of equipment including pneumatic to be conected to cobot software.



- The meaning of the color schemes listed below must be maintained:

Yellow with red text	Dedicated safety signals
Yellow with black text	Configurable for safety
Gray with black text	General purpose digital I/O
Green with black text	General purpose analog I/O

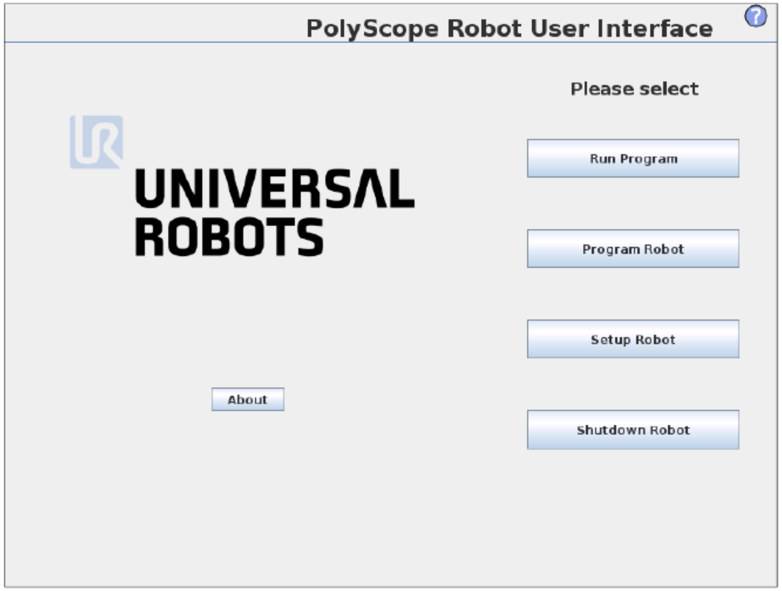
Content

A. Theoretical Module

- General Presentation
- Cobot Selection
- Mechanical Installation
- Griper Development
- Electrical Installation
- Software of Cobot
- Design of Cobot Program

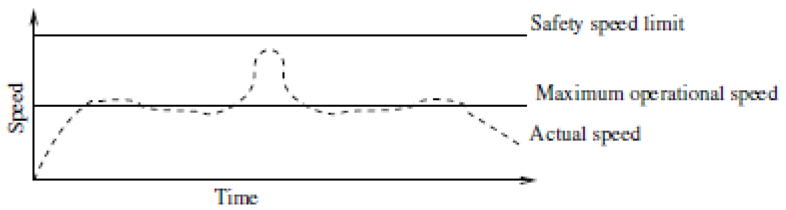
Software of Cobot

- The **PolyScope** or robot user interface is the touch screen on your Teach Pendant panel.
- It is developed by Universal Robots company.
- It is the graphical user interface (GUI) that operates the robot arm and control box, executes and creates robot programs.
- The PolyScope it is built based on Debian Linux



Software of Cobot

- The UR Cobots operates in several modes:
  - **Normal mode** – the cobots operates in a protective mode, safety limitations while the cobot is performing designated tasks;
  - **Reduced mode** – Active when the robot Tool Center Point is positioned beyond a Trigger Reduced mode plane or when triggered using a configurable input.
  - **Recovery mode** – when the robot is outside allowed area. This mode allows the robot to move slowly back to the allowed area using MoveTab or Freedrive. It is not possible to run programs for the robot in this mode.

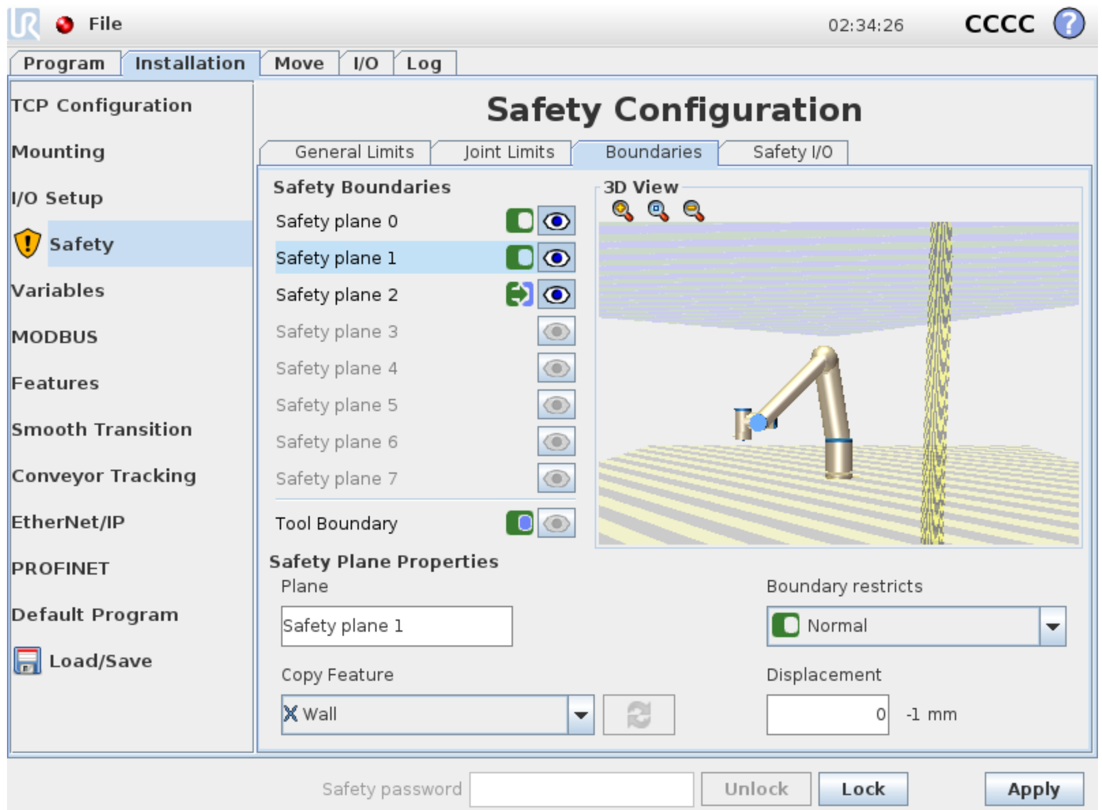


Software of Cobot

- The UR Cobots operates in several modes:
  - **Freedrive mode -** When in Freedrive mode and the movement of the robot arm comes close to certain limits, the user will feel a repelling force. This force is generated for limits on the position, orientation and speed of the robot TCP and the position and speed of the joints. The purpose of this repelling force is to inform the user that the current position or speed is close to a limit and to prevent the robot from violating that limit. However, if enough force is applied by the user to the robot arm, the limit can be violated. The magnitude of the force increases as the robot arm comes closer to the limit.
  - **Backdrive mode** – can be used to forcefully move specific joints to a desired position without releasing all brakes in the robot arm.

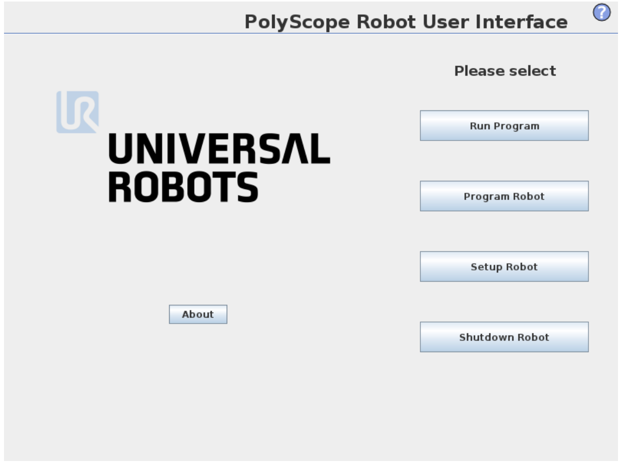
Software of Cobot

- **Safety Configuration**:
  - Prior to start programming the robot, safety parameters must be introduced into the cobot:
    - General Limits:
      - Force;
      - Power;
      - Speed;
      - Momentum.
    - Joint Limits:
      - Joint speed;
      - Joint position;
    - Boundaries;

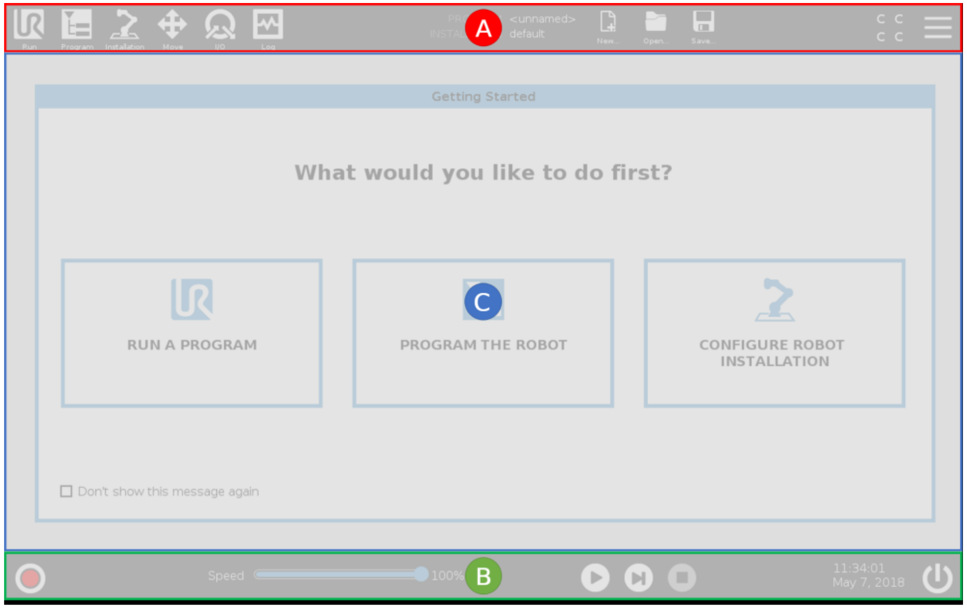


Software of Cobot

- **PolyScope** welcome screen is composed from next main buttons:
  - Run Program – choose and run an existing program. This is the simplest way to operate the robot arm and control box.;
  - Program Robot – change a program, or create a new program;
  - Setup Robot – change the language, set passwords, upgrade software, etc.;
  - Shut down – powers off the robot arm and shuts down the control box.;
  - About –PolyScope version, IP, host, etc;



- **PolyScope** comprises three zones:
  - Header with tabs/icons that make interactive screens available to you.
  - Footer with buttons that control your loaded program/s.
  - Screen with fields that manage and monitor robot actions.

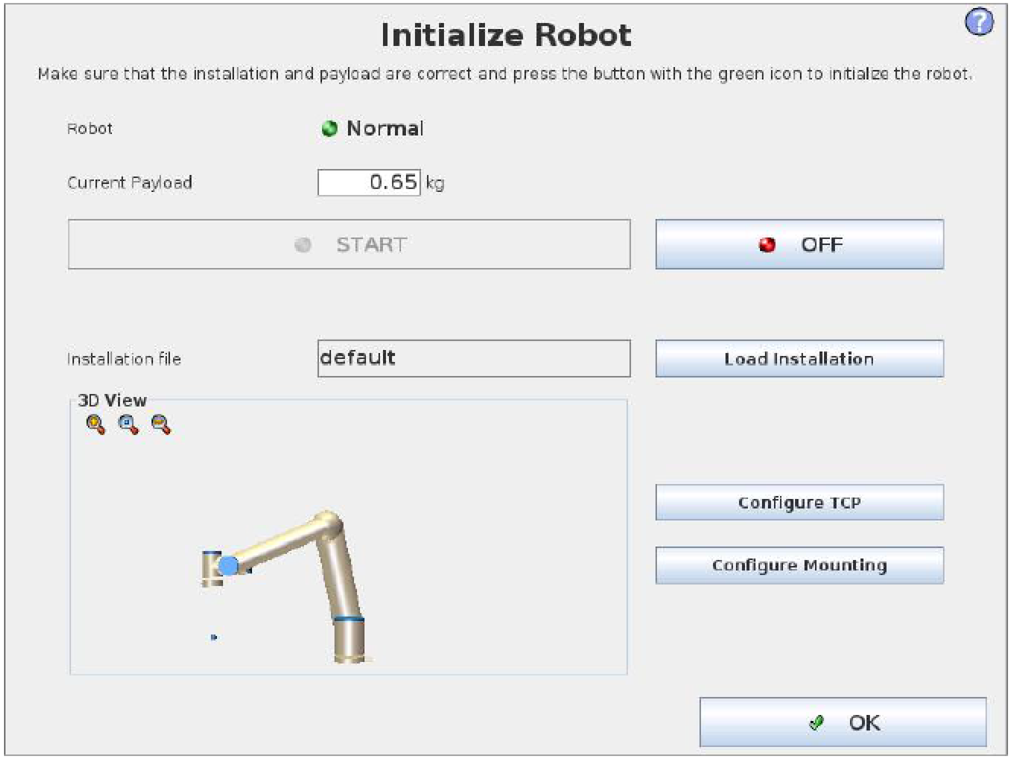


Software of Cobot

- Before any work of cobot, some general settings must be adjusted:
  - Initialize the robot;
  - Language and units;
  - Update soft, if updates exist;
  - Password;
  - Screen calibration;
  - Network setup;
  - Date and time;

Software of Cobot

- Robot initialization consist in:



- Initialize robot
  - Check payload setting
  - Click START: enables power
  - Click START: releases brakes

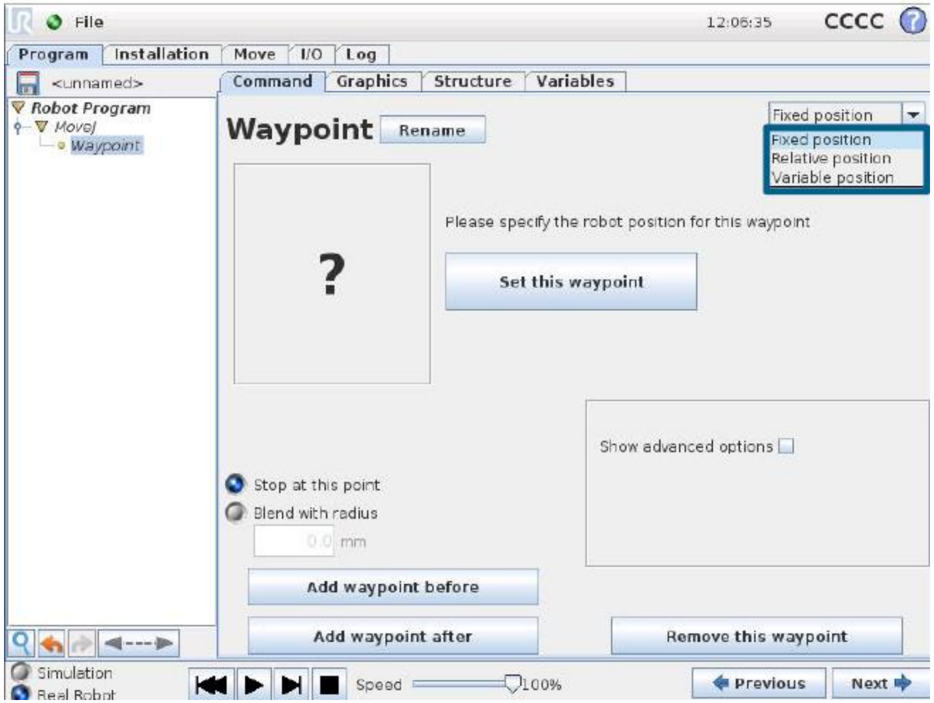
Content

A. Theoretical Module

- General Presentation
- Cobot Selection
- Mechanical Installation
- Griper Development
- Electrical Installation
- Software of Cobot
- Design of Cobot Program

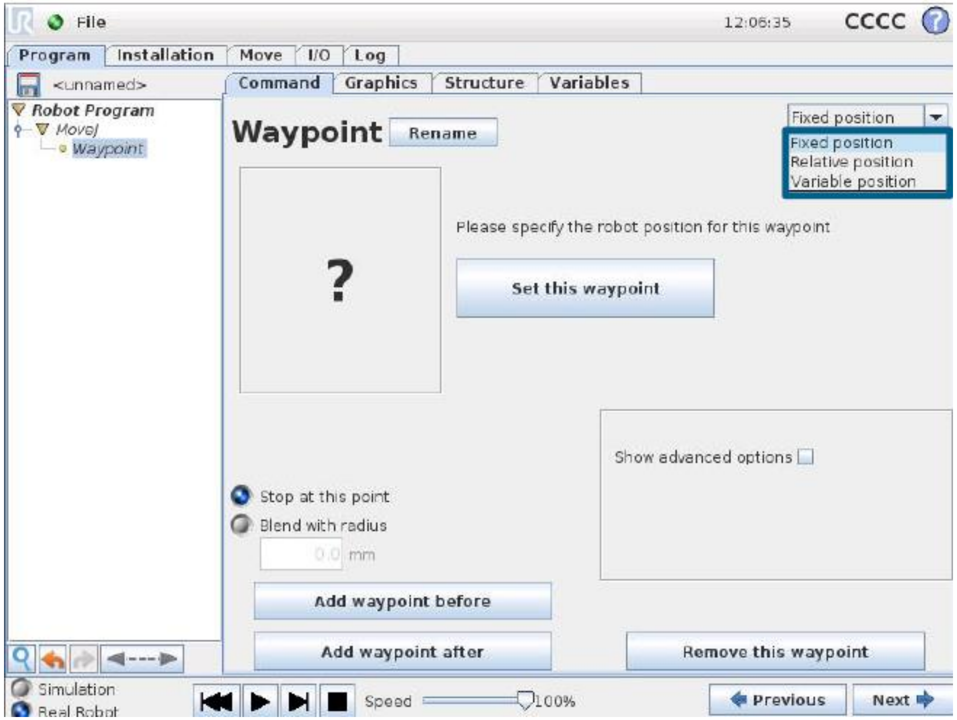
Design of Cobot Program

A ***Waypoint*** is a point on the robot path. Waypoints are the most central part of a robot program, telling the robot arm where to be. A ***fixed position waypoint*** is given by physically moving the robot arm to the position.



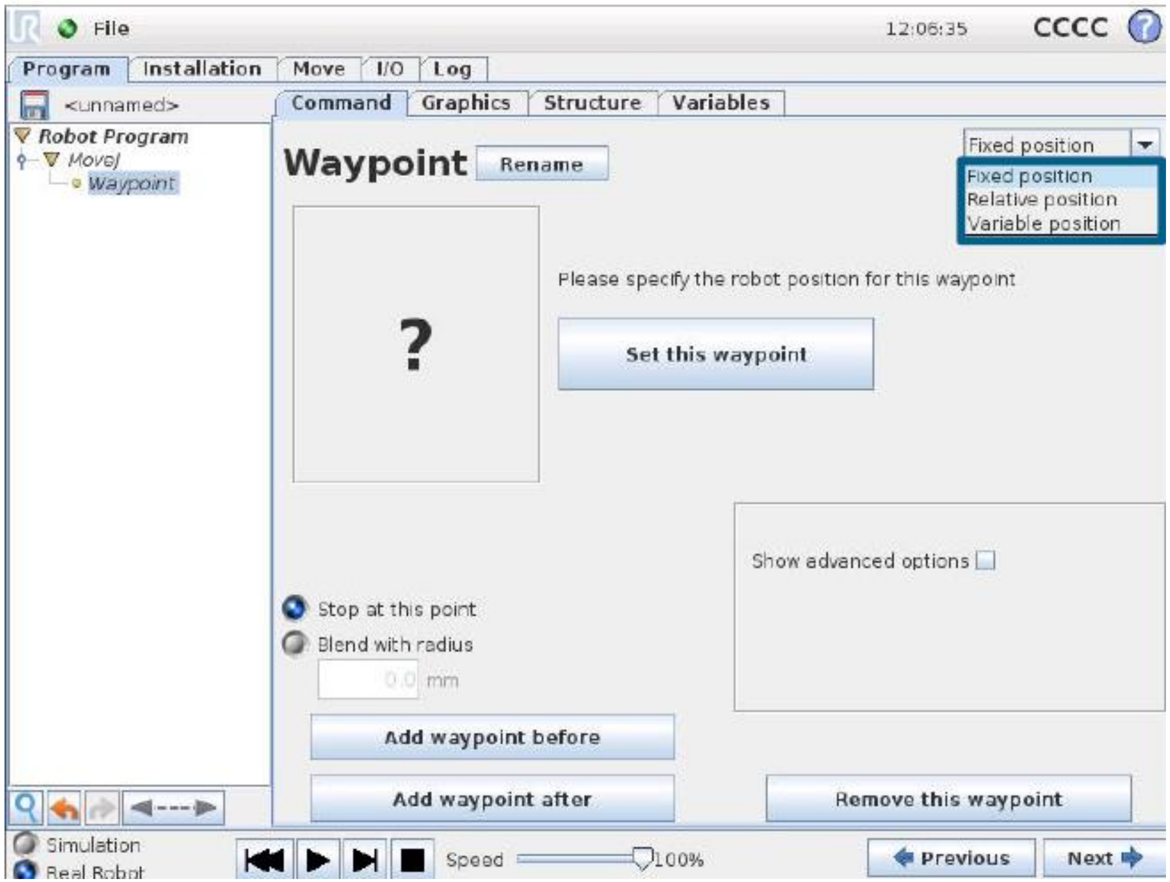
Design of Cobot Program

- A ***Relative Waypoint*** is a waypoint with the position given relative to the robot arm's previous position, such as "two centimeters to the left". The relative position is defined as the difference between the two given positions (left to right). Note that repeated relative positions can move the robot arm out of its workspace.



Design of Cobot Program

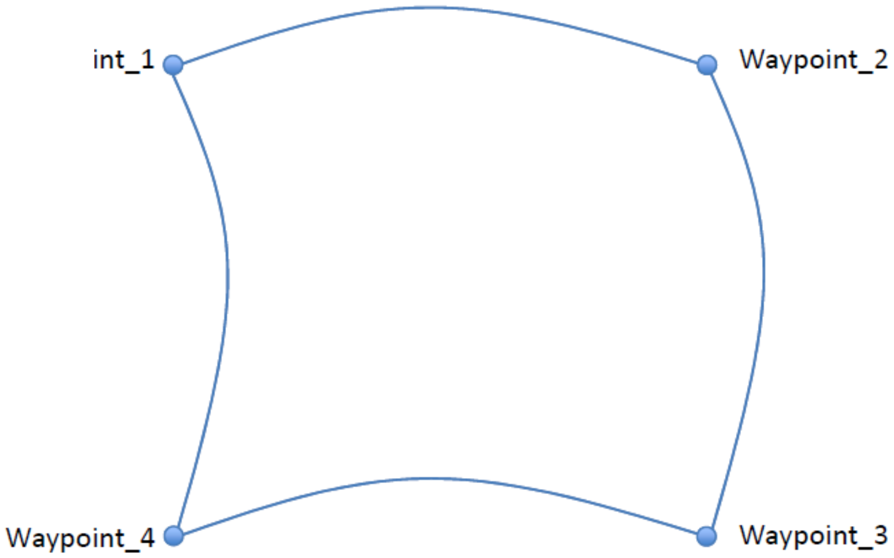
- A ***Variable Waypoint*** is a waypoint with the position given by a variable.



Design of Cobot Program

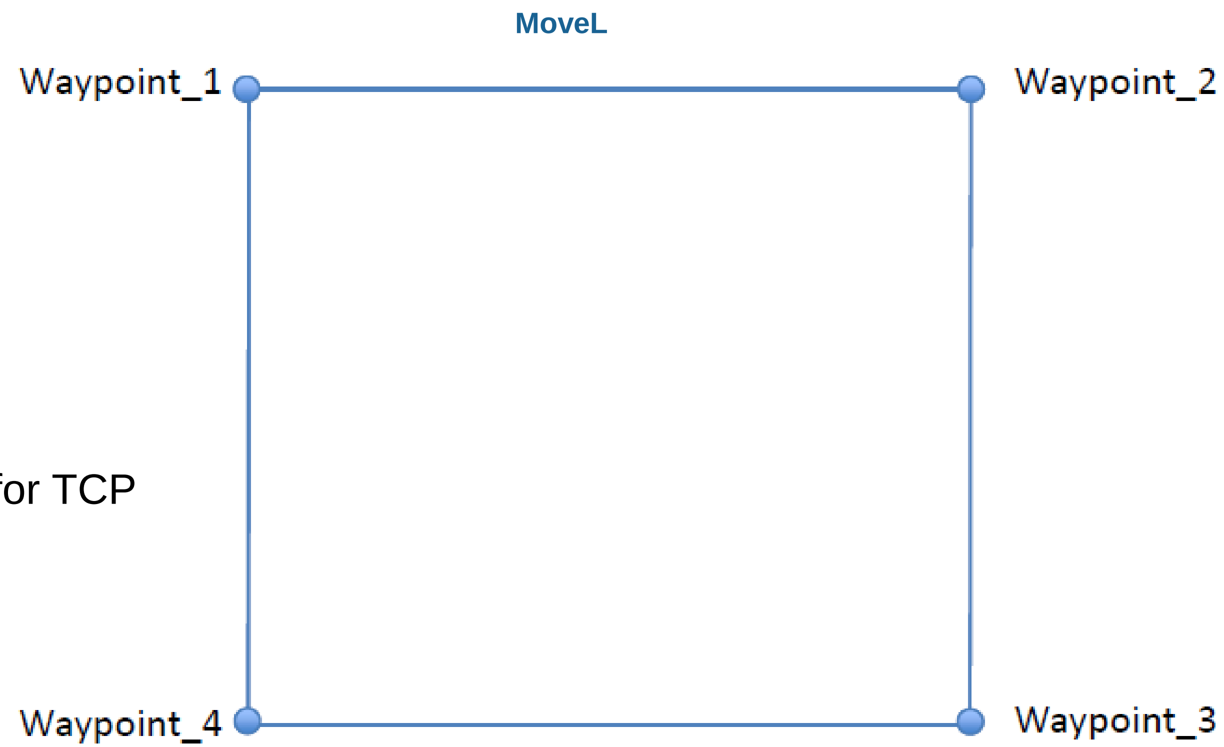
- **Types of movements:**
  - ***MoveJ*** will make movements that are calculated in the joint space of the robot arm. Each joint is controlled to reach the desired end location at the same time. This movement type results in a curved path for the tool.

- Joint movement
  - No interpolation
  - Fastest move type
  - Useful in ”free” space movements



Design of Cobot Program

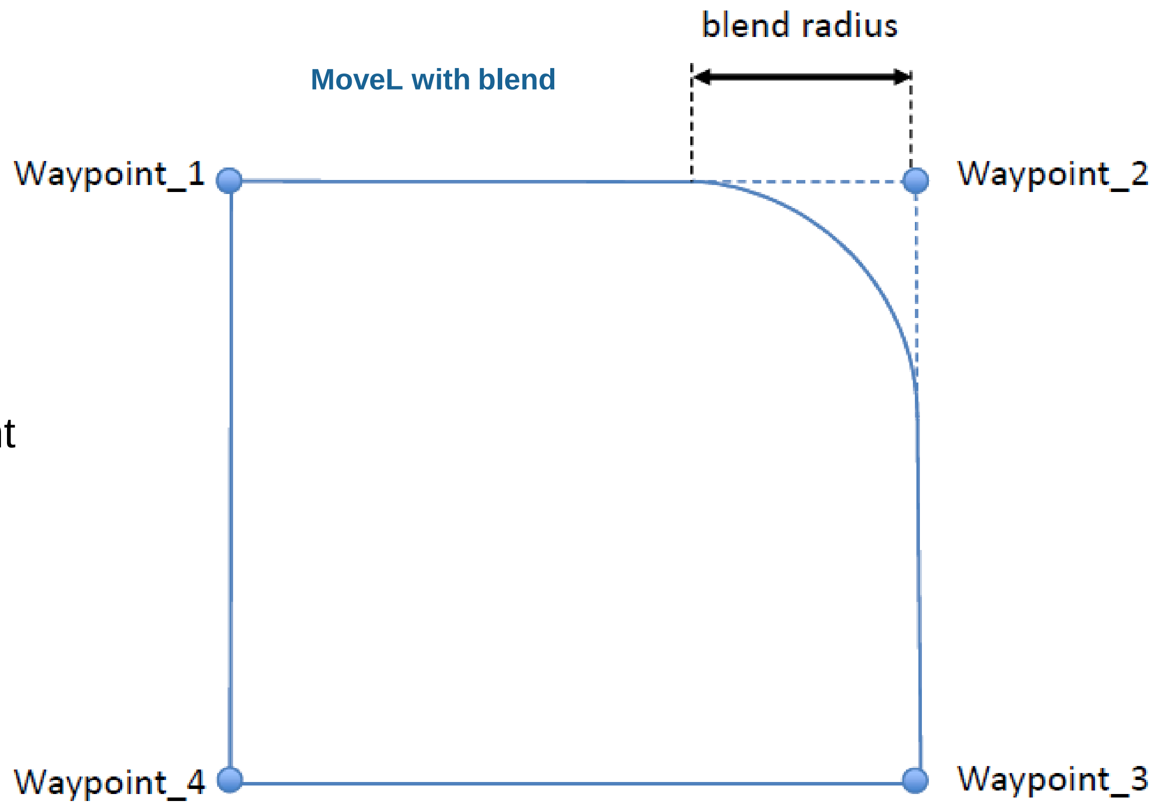
- ***moveL*** will make the tool move linearly between waypoints. This means that each joint performs a more complicated motion to keep the tool on a straight line path.



- Linear movement
  - Interpolation on
  - Linear trajectory for TCP

Design of Cobot Program

- **MoveL with blend** – if a *blend radius* (r) is set, the robot arm trajectory blends around the waypoint, allowing the robot arm not to stop at the point.

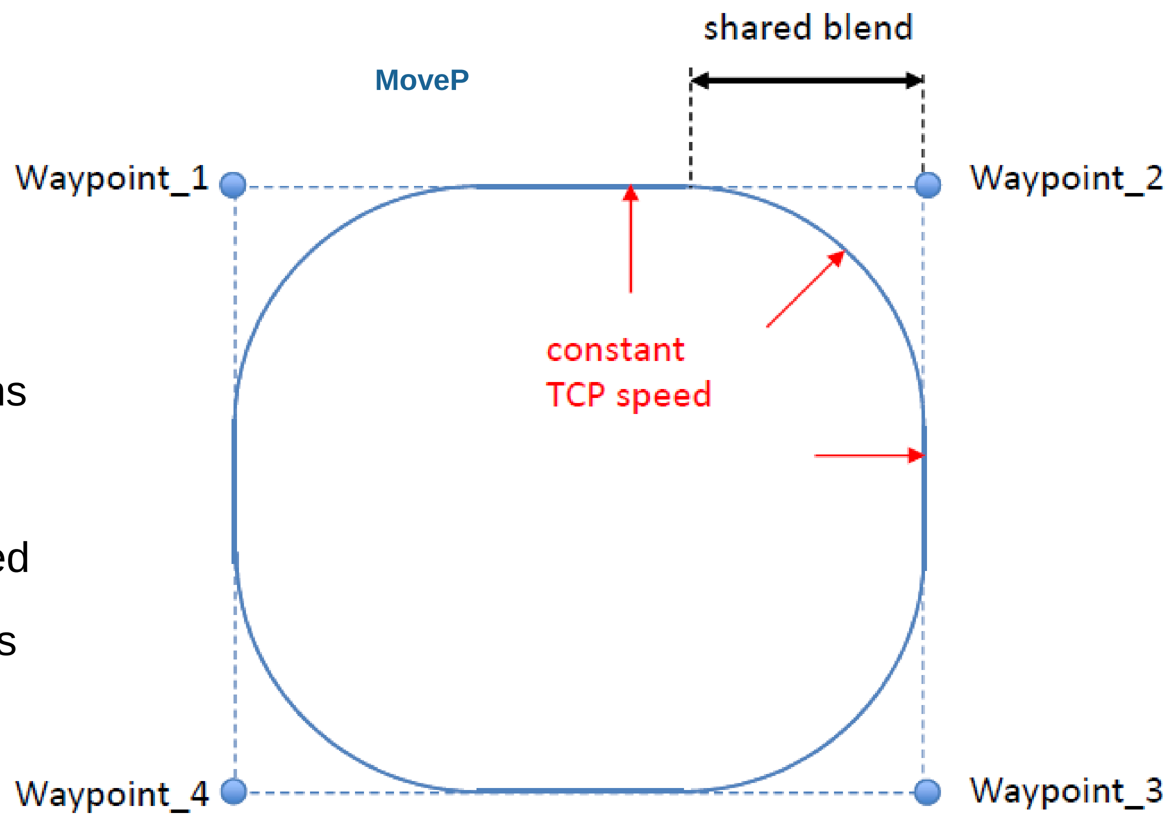


- Blend radius
  - Continuous movement
  - No stop in Waypoint

Design of Cobot Program

- ***moveP*** will move the tool linearly with constant speed with circular blends, and is intended for some process operations, like gluing or dispensing.

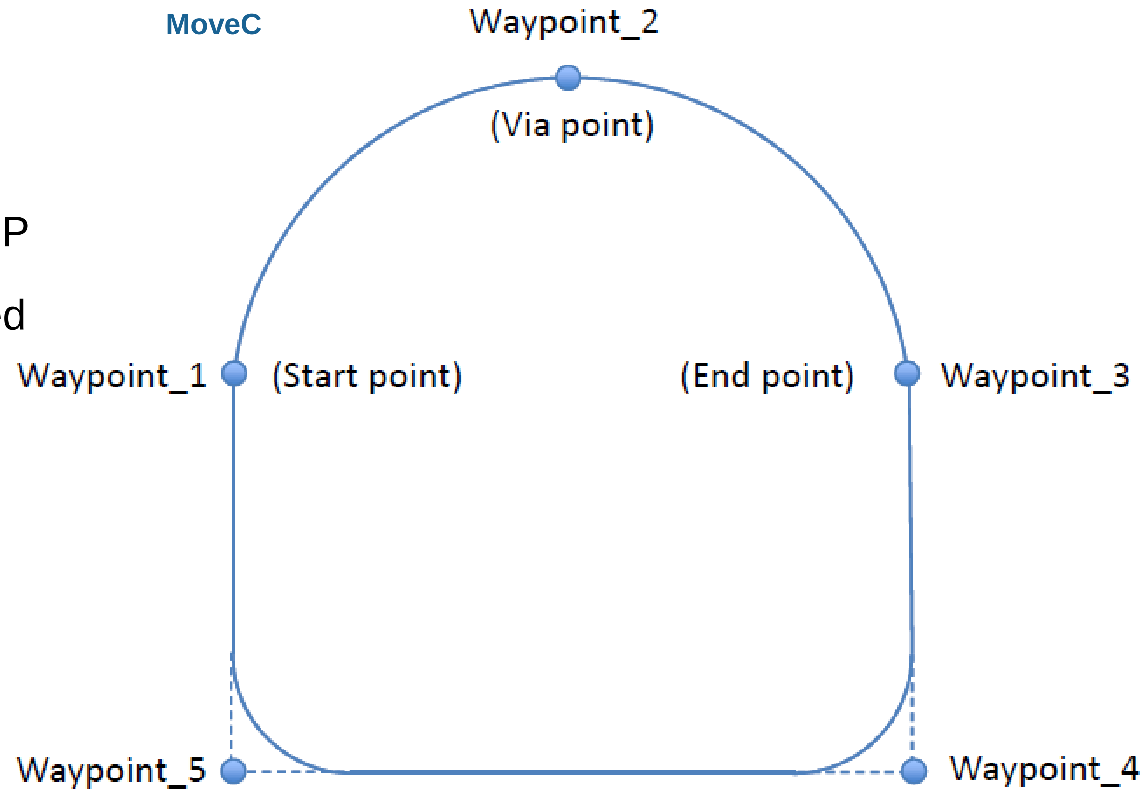
- Process Movement
  - Process applications
  - Linear movement
  - Constant TCP speed
  - Shared blend radius



Design of Cobot Program

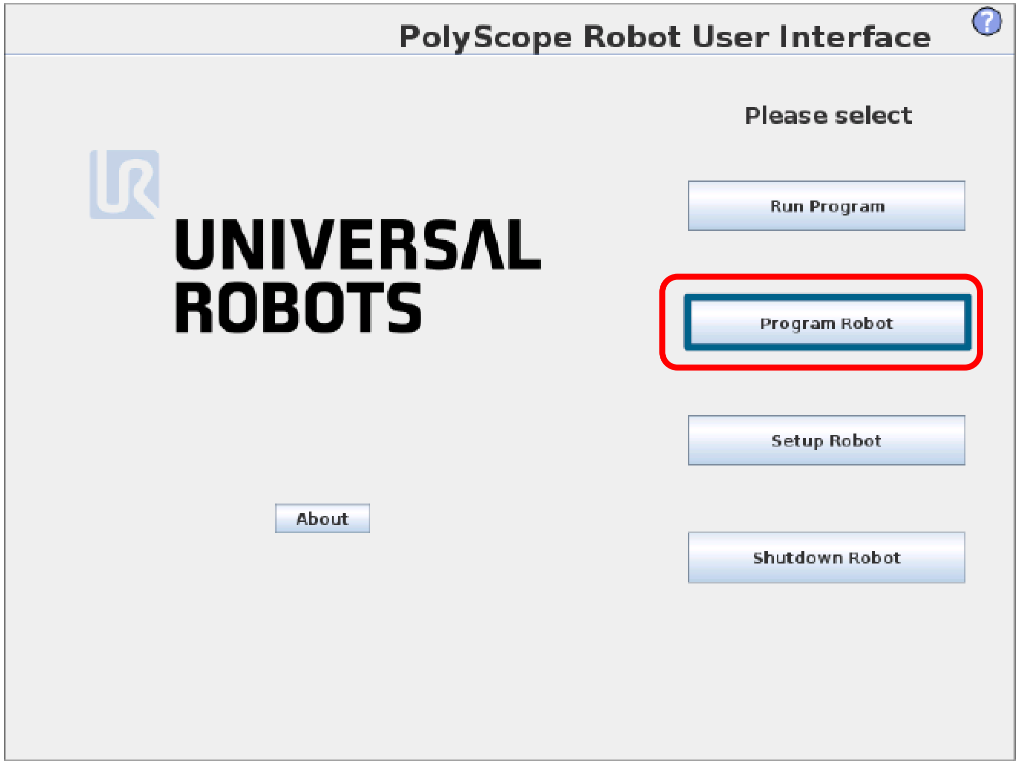
- A ***Circle Move (moveC)*** can be added to a *moveP* command, consisting of two waypoints: the first one specifying a via point on the circular arc, and the second one being the endpoint of the movement.

- Circular movement
  - Circular trajectory for TCP
  - Three Waypoints required
    - Start point
    - Via point
    - End point
  - Radius auto calculated



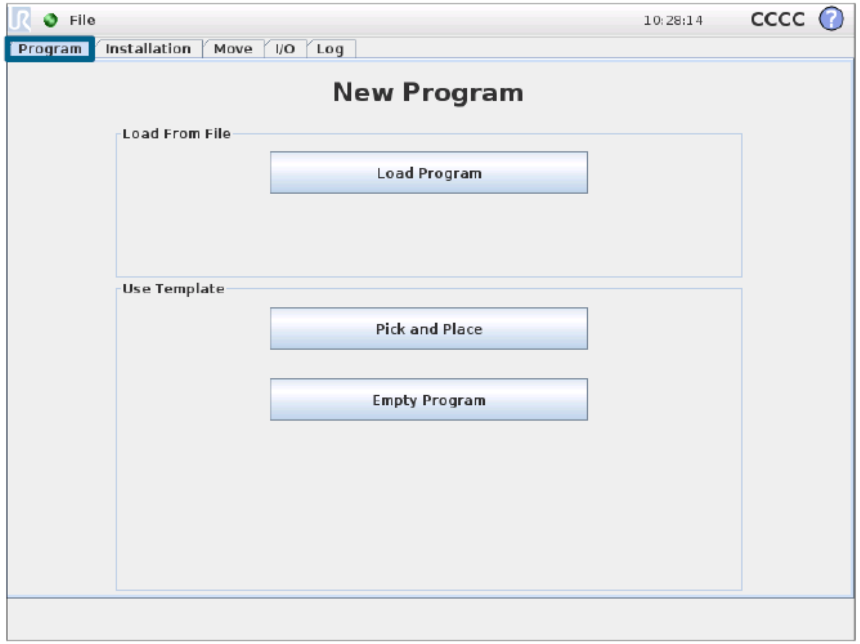
Design of Cobot Program

- Building a cobot program can be done in the **Program robot** page from the main interface.



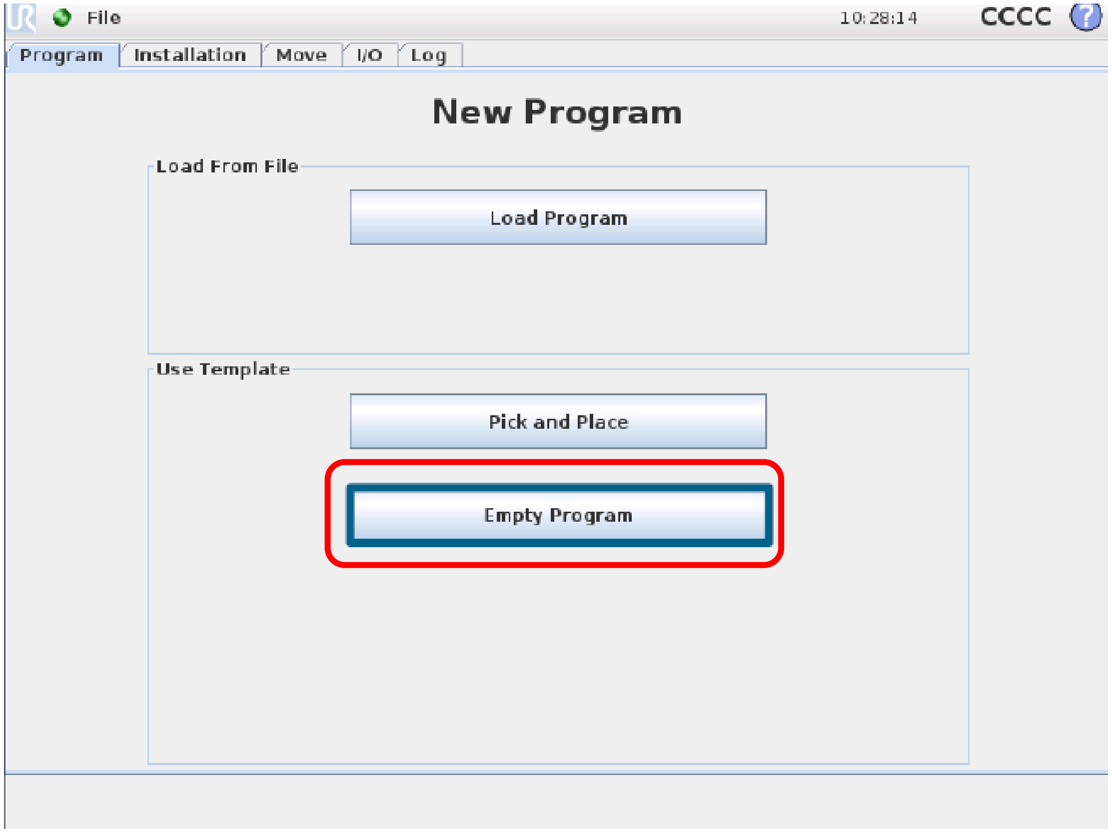
Design of Cobot Program

- A robot program can be created from scratch or from a template, like Pick and Place application.
- If the program has been created in an external computer, the program can be loaded using PolyScope.



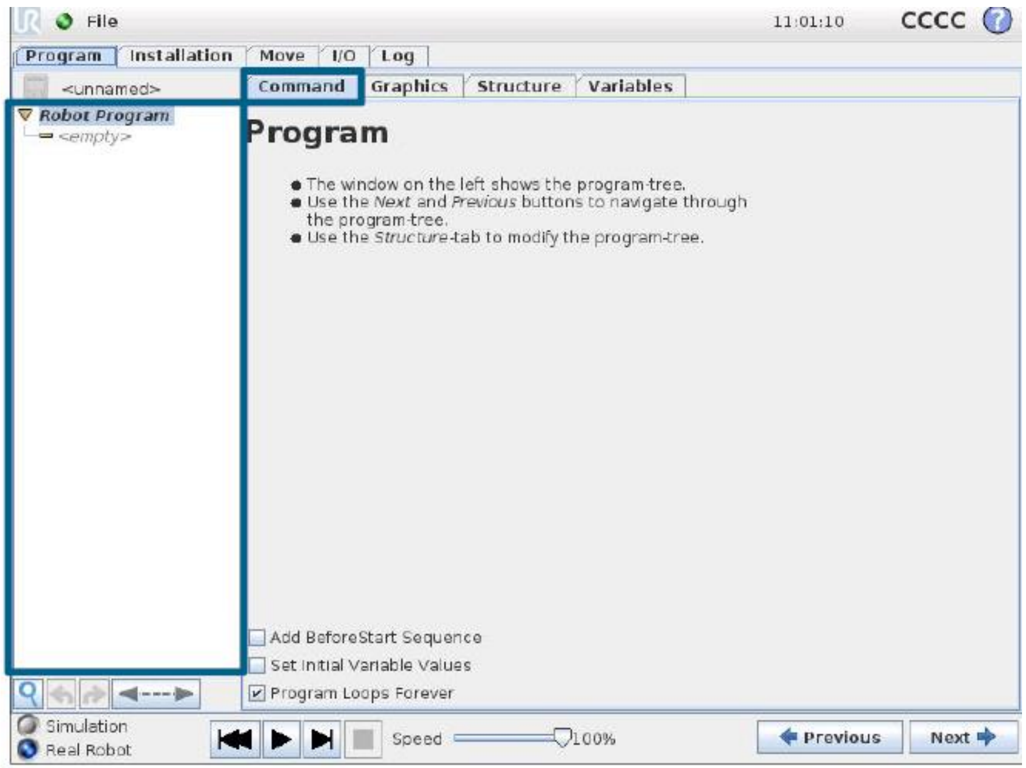
Design of Cobot Program

- A template can provide the overall program structure, so only the details of the program need to be filled in.



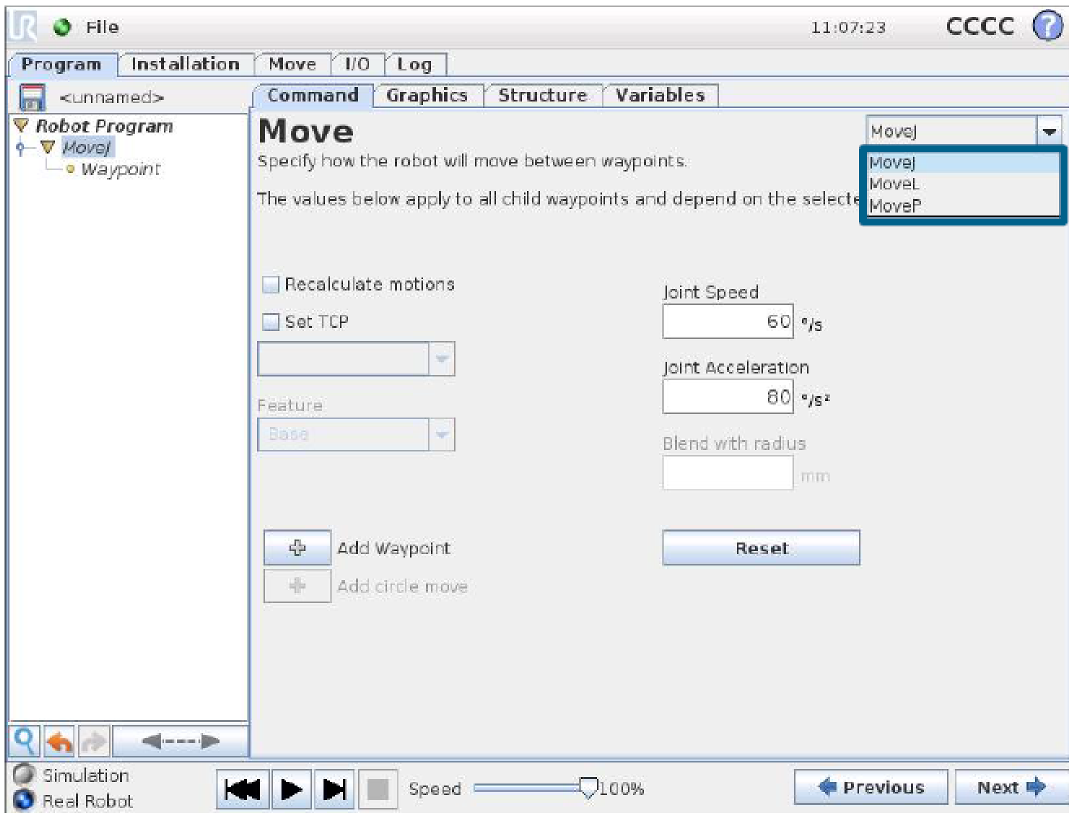
Design of Cobot Program

- The ***Program Tree*** contains visual cues informing about the command currently being executed by the robot controller.
- If a command is clicked while a program is running, the ***Command Tab*** will display the information related to the selected command.



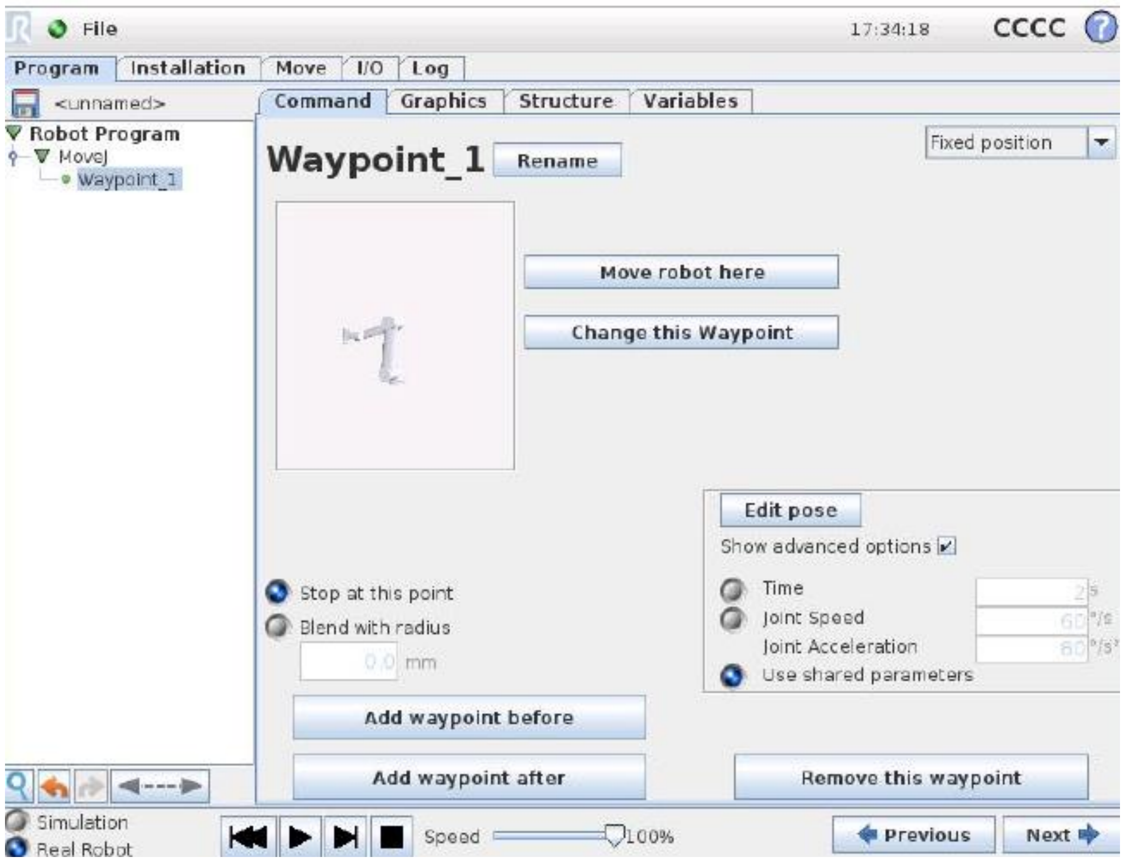
Design of Cobot Program

- The ***Move*** command controls the robot motion through the underlying waypoints. Waypoints have to be under a Move command. The Move command defines the acceleration and the speed at which the robot arm will move between those waypoints.



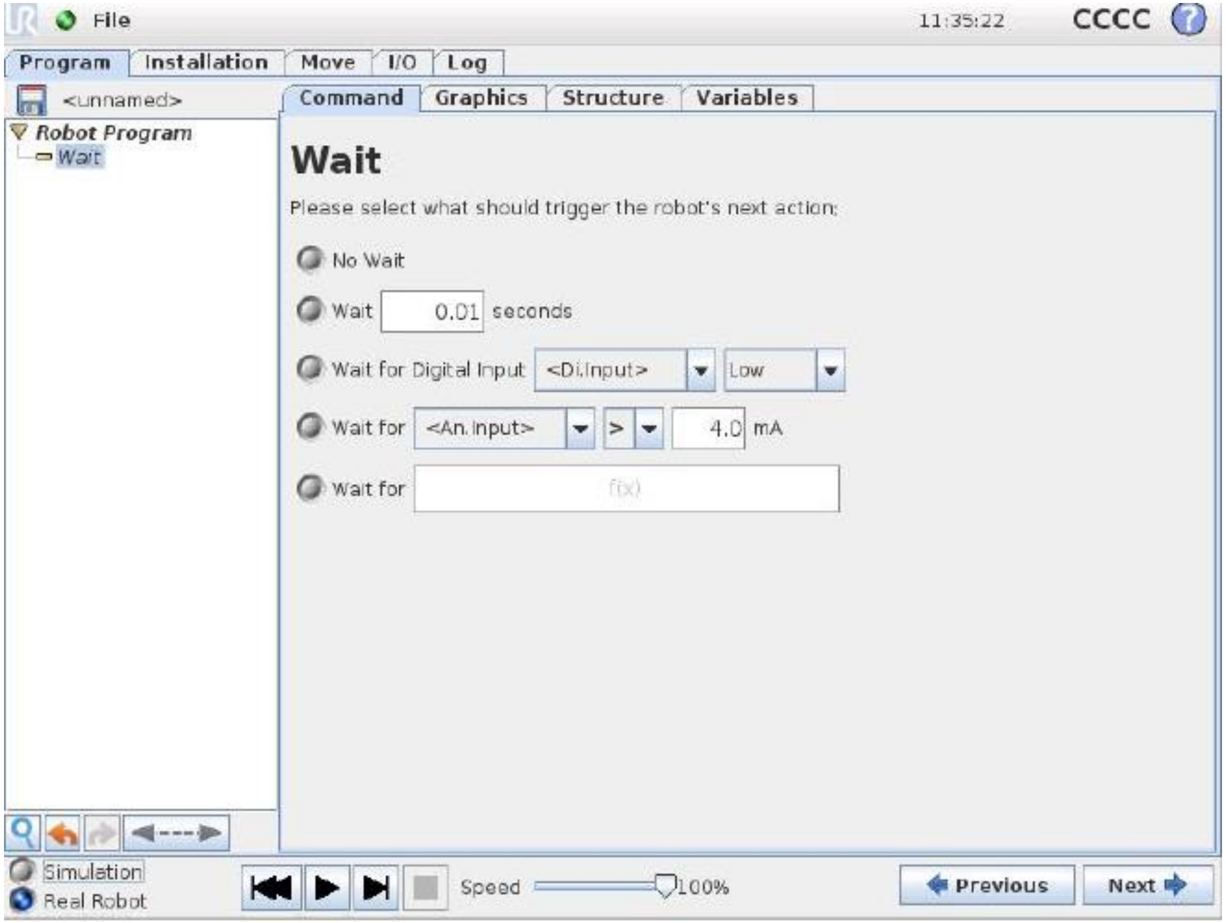
Design of Cobot Program

- You can also modify the waypoint’s name and position and by pressing the “Move robot here” button, the cobot moves to that position.



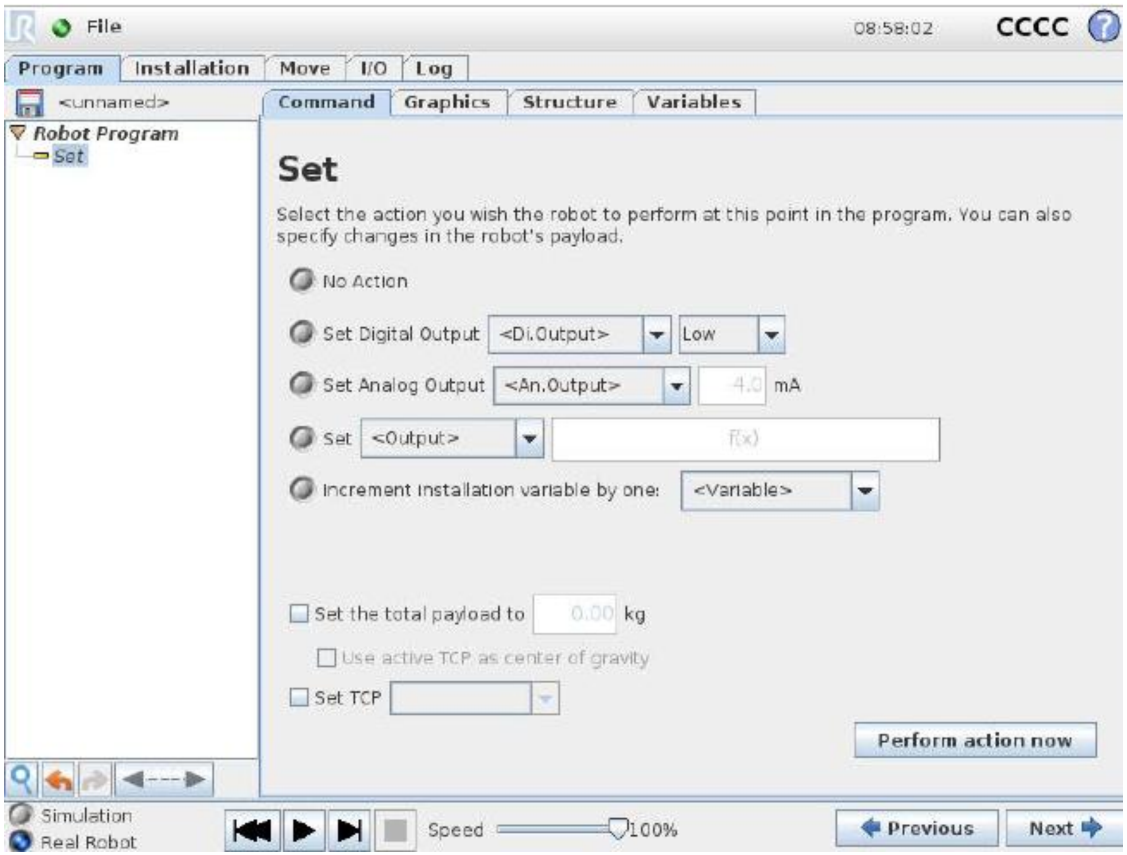
Design of Cobot Program

- The ***Wait*** command waits for a given amount of time or for an I/O signal.



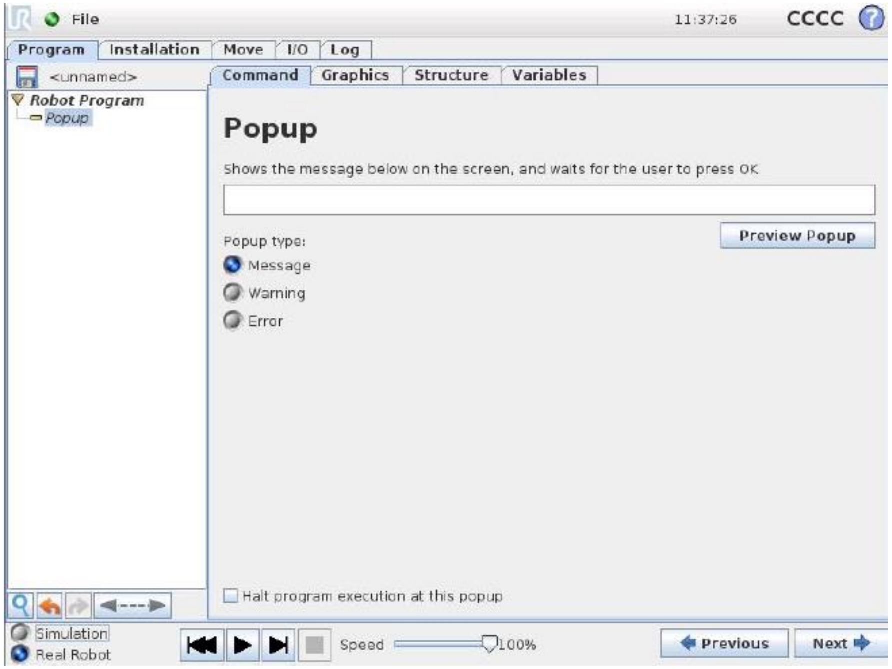
Design of Cobot Program

- The Set command sets either digital or analog outputs to a given value.



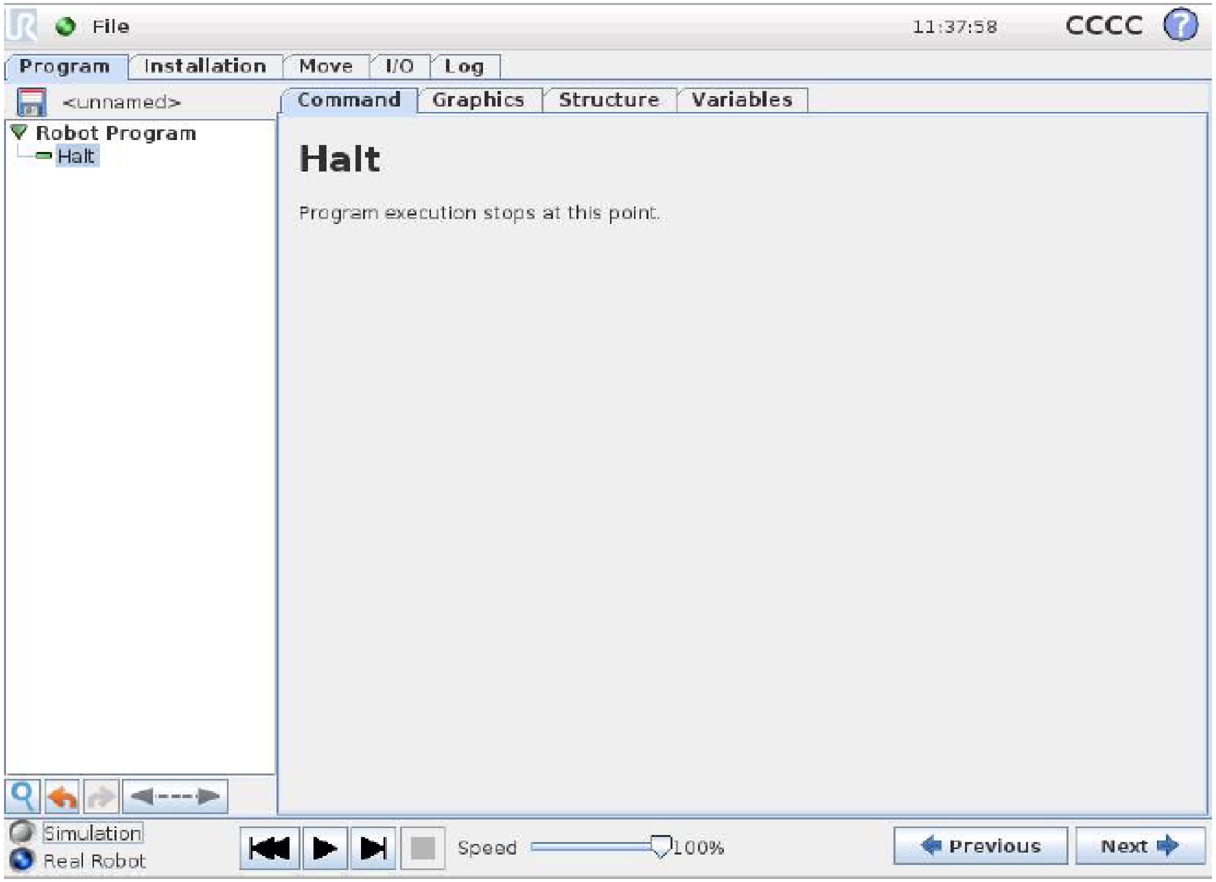
Design of Cobot Program

- The ***popup*** is a message that appears on the screen when the program reaches this command. The style of the message can be selected, and the text itself can be given using the on-screen keyboard. The robot waits for the user/operator to press the “OK” button under the popup before continuing the program. If the “Halt program execution” item is selected, the robot program halts at this popup.



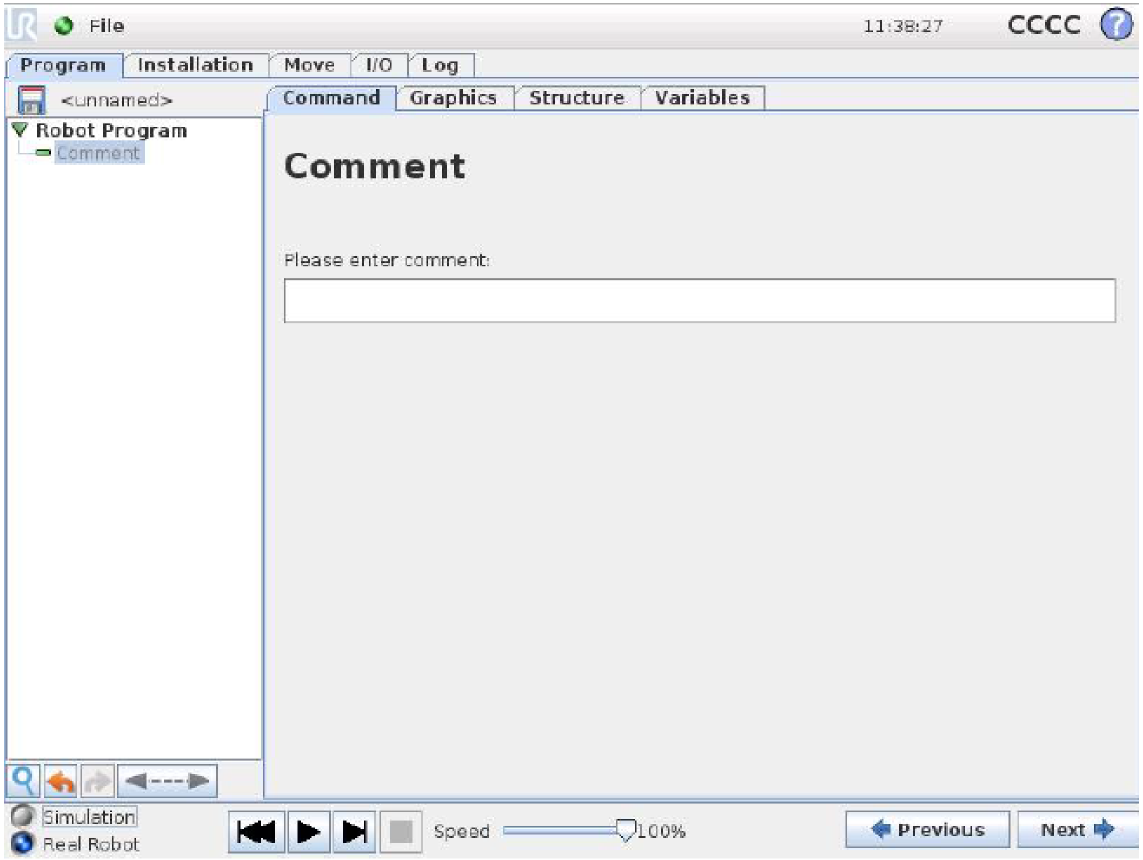
Design of Cobot Program

- The ***Halt*** command stops the program execution when it is reached in the Program Tree.



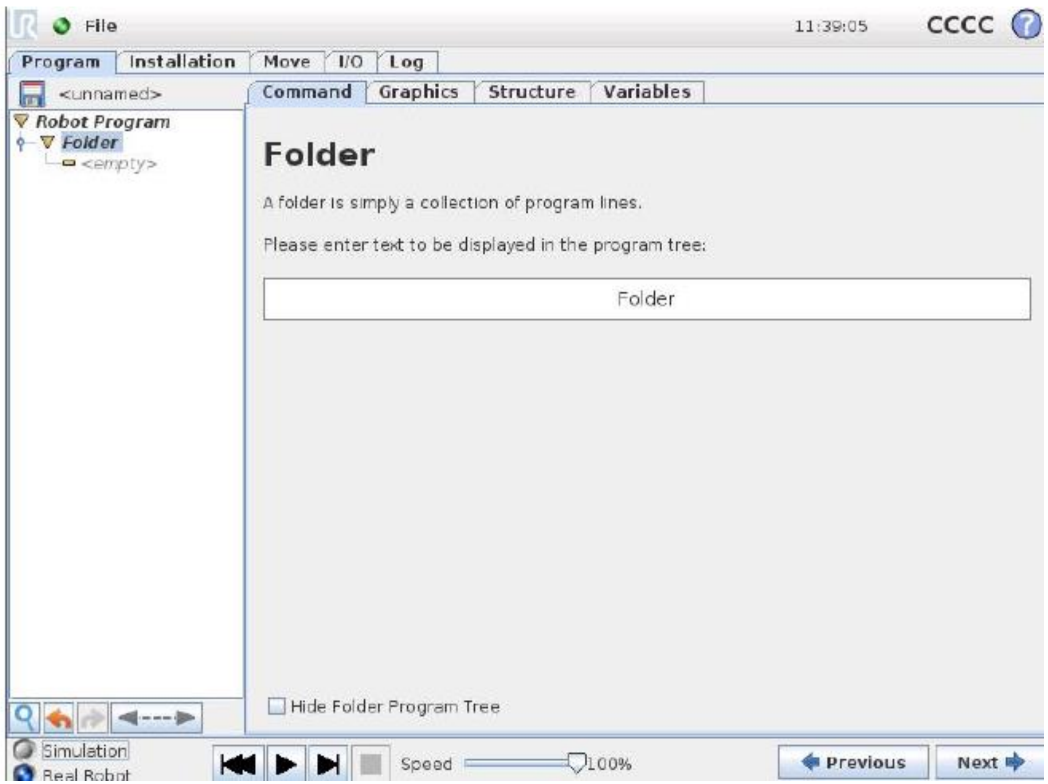
Design of Cobot Program

- Gives the programmer an option to add a line of text to the program. This line of text does not do anything during program execution.



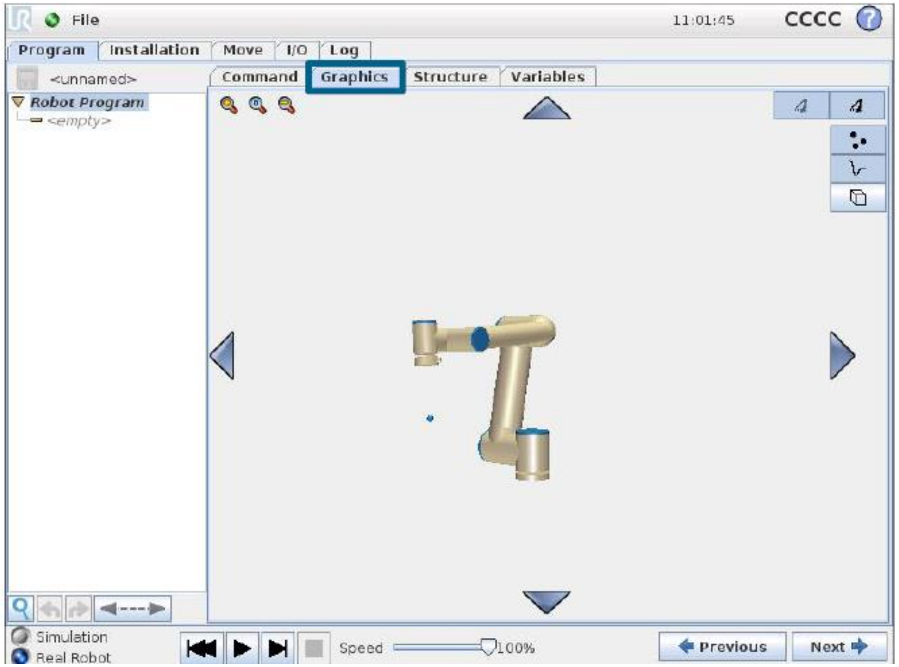
Design of Cobot Program

- A folder is used to organize and label specific parts of a program, to clean up the program tree, and to make the program easier to read and navigate. A folder does not in itself do anything.



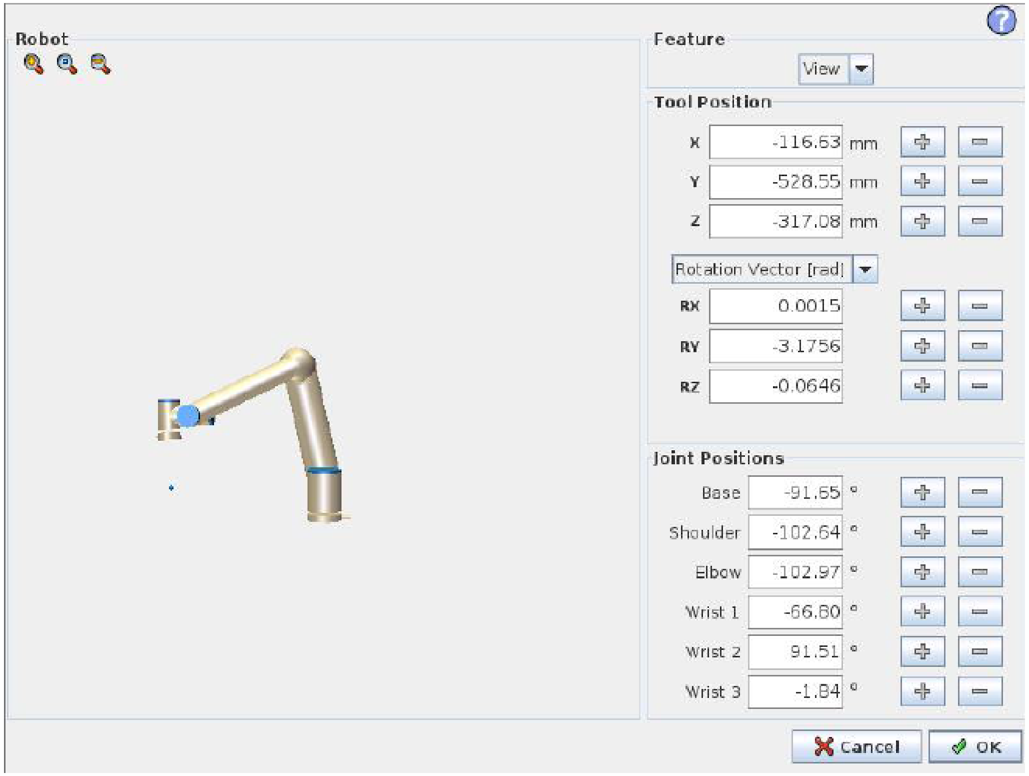
Design of Cobot Program

- The ***Graphics Tab*** shows the graphical representation of the current robot program.
- The 3D drawing of the robot arm shows the current position of the robot arm, and the “shadow” of the robot arm shows how the robot arm intends to reach the waypoint selected in the left hand side of the screen.



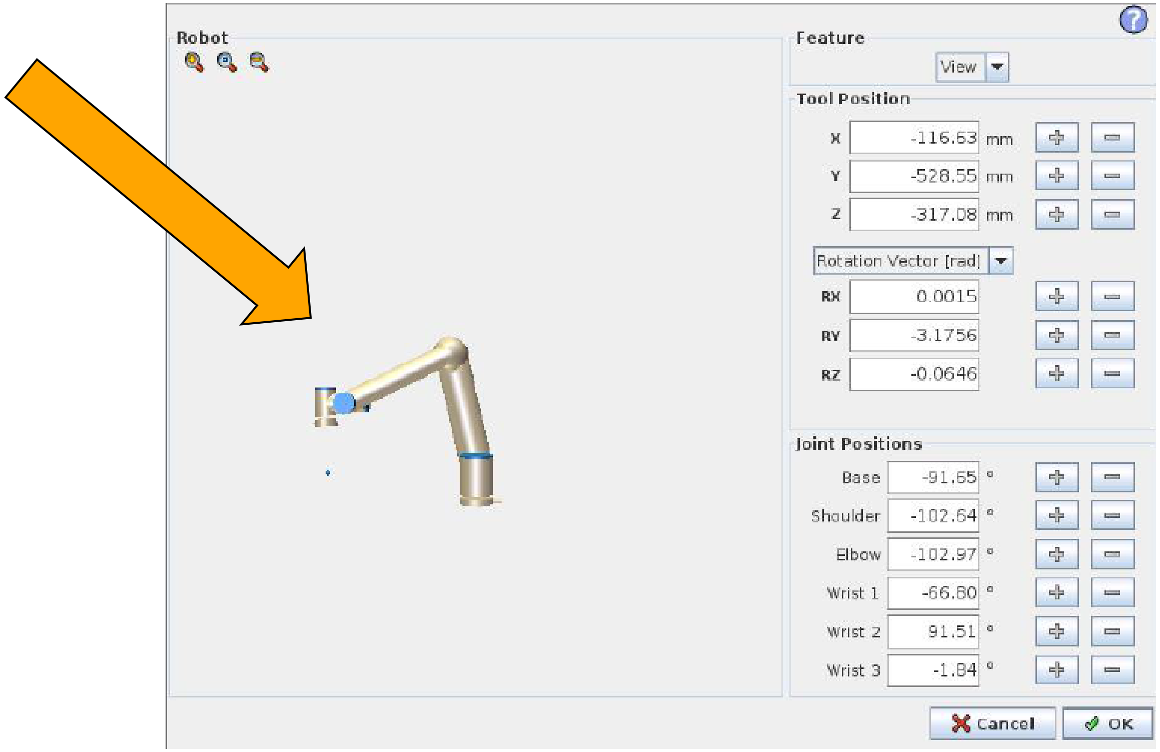
Design of Cobot Program

- On this screen you can specify target joint positions, or a target pose (position and orientation) of the robot tool. This screen is “offline” and does not control the robot arm directly.



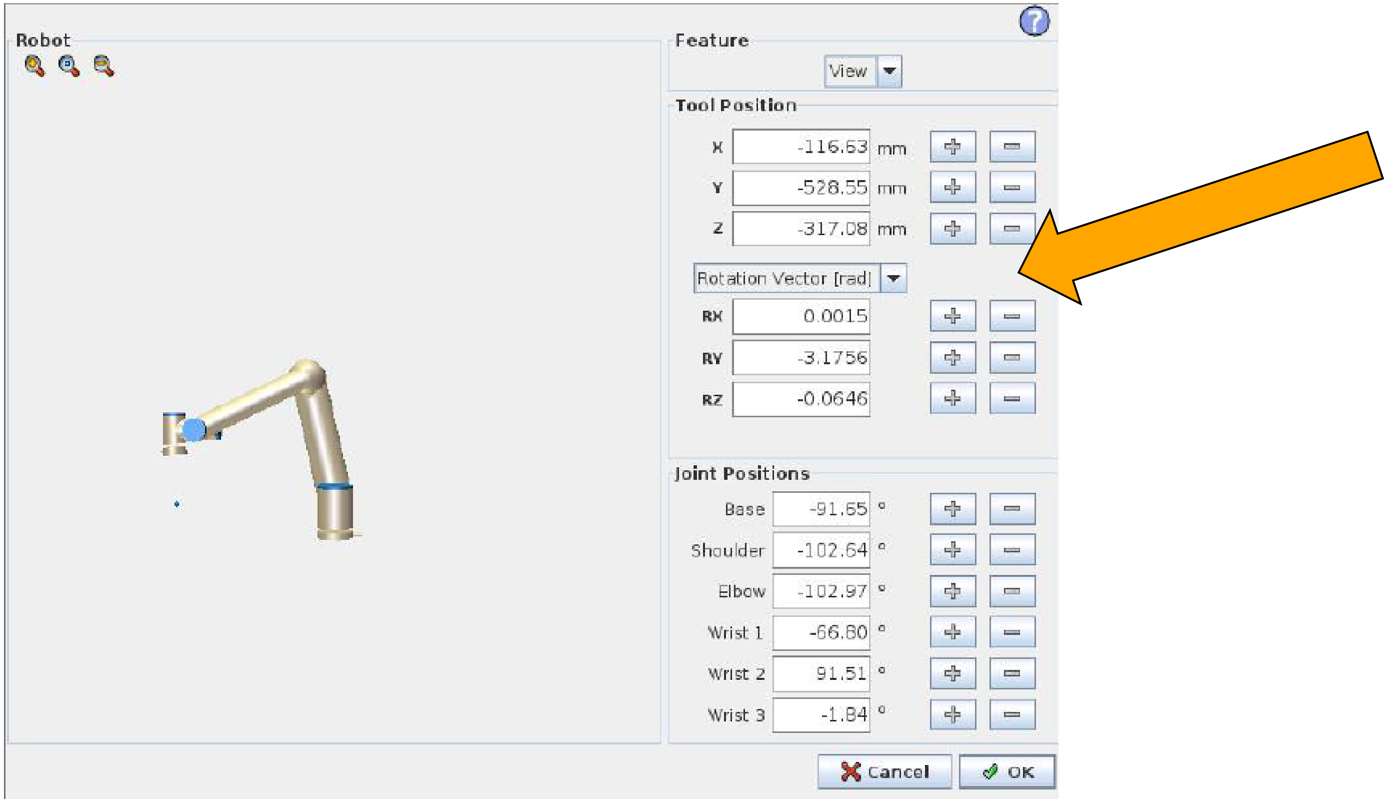
Design of Cobot Program

- The current position of the robot arm and the specified new target position are shown in 3D graphics. The 3D drawing of the robot arm shows the current position of the robot arm, and the "shadow" of the robot arm shows the target position of the robot arm controlled by the specified values on the right hand side of the screen.



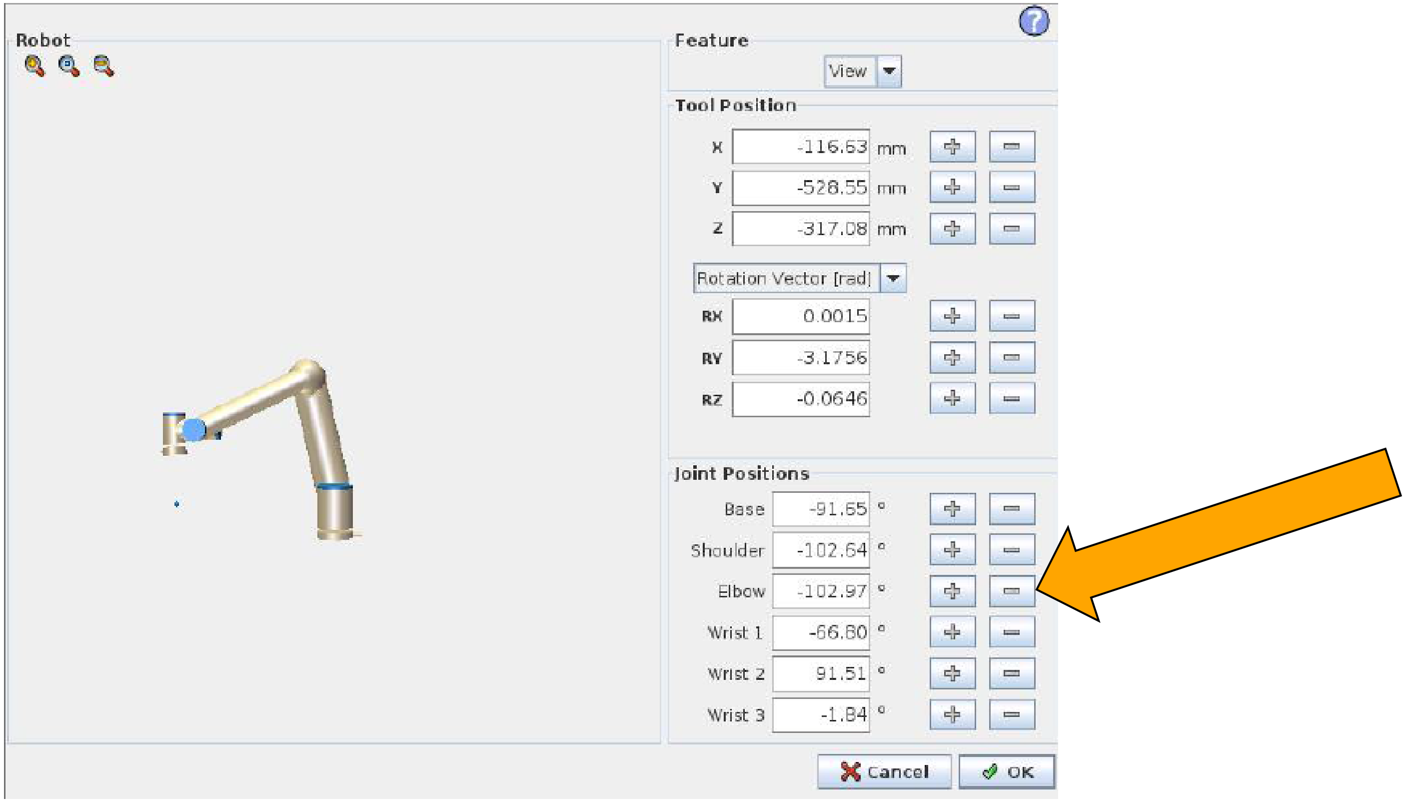
Design of Cobot Program

- The text boxes show the full coordinate values of that TCP (Tool Center Point) relative to the selected feature. X, Y and Z control the position of the tool, while RX, RY and RZ control the orientation of the tool.



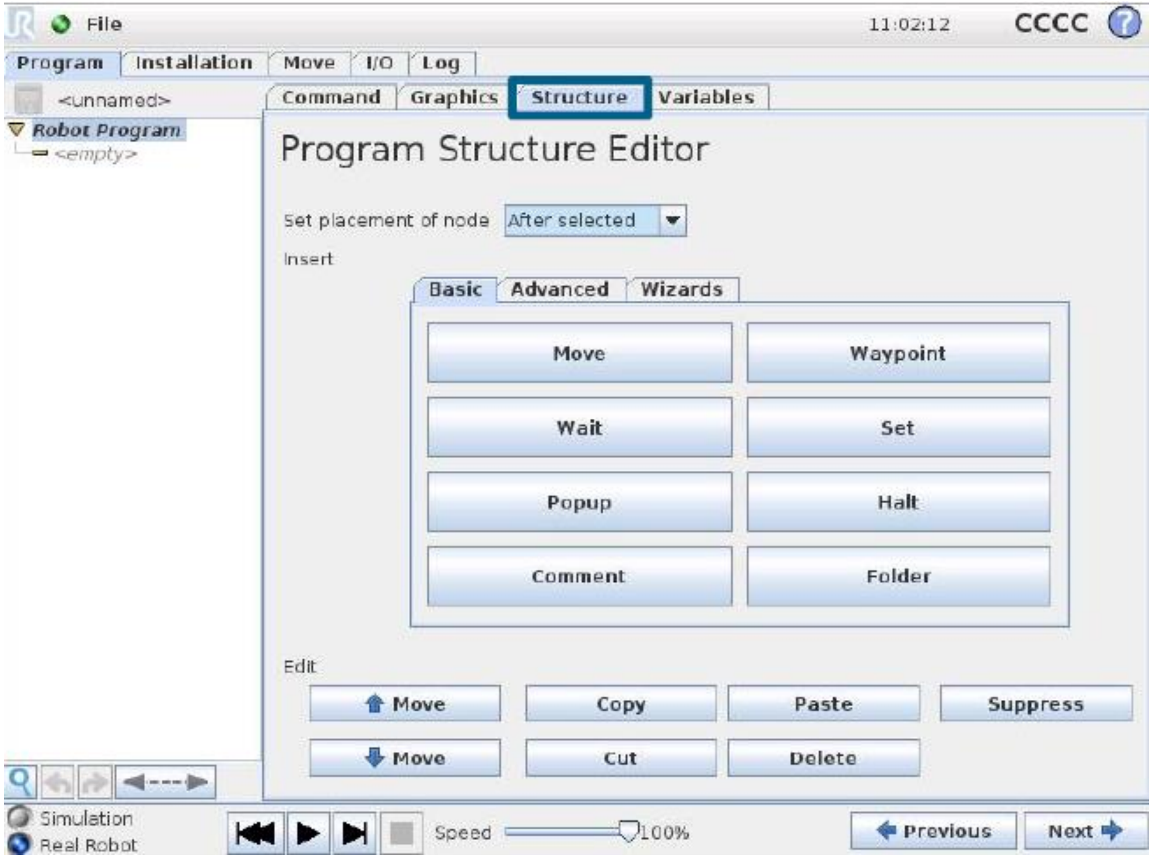
Design of Cobot Program

- Allows the individual joint positions to be specified directly. Each joint position can have a value in the range from −360◦ to +360◦ , which are the joint limits.



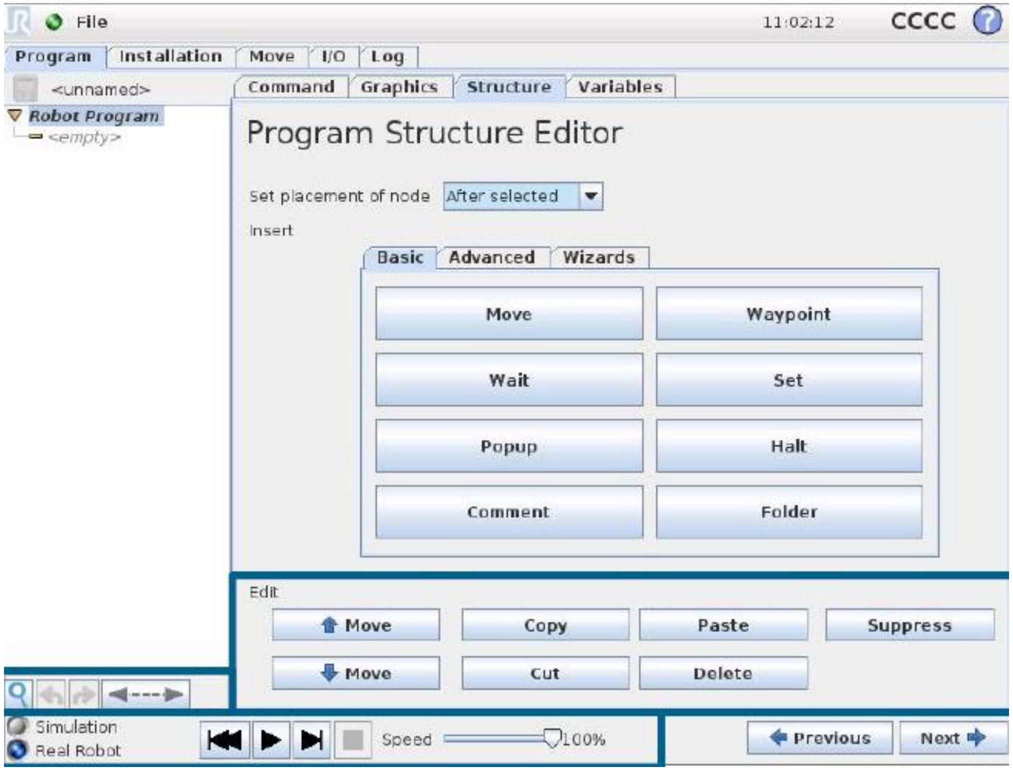
Design of Cobot Program

- The **Structure Tab** gives an opportunity for inserting, moving, copying and removing the various types of commands.



Design of Cobot Program

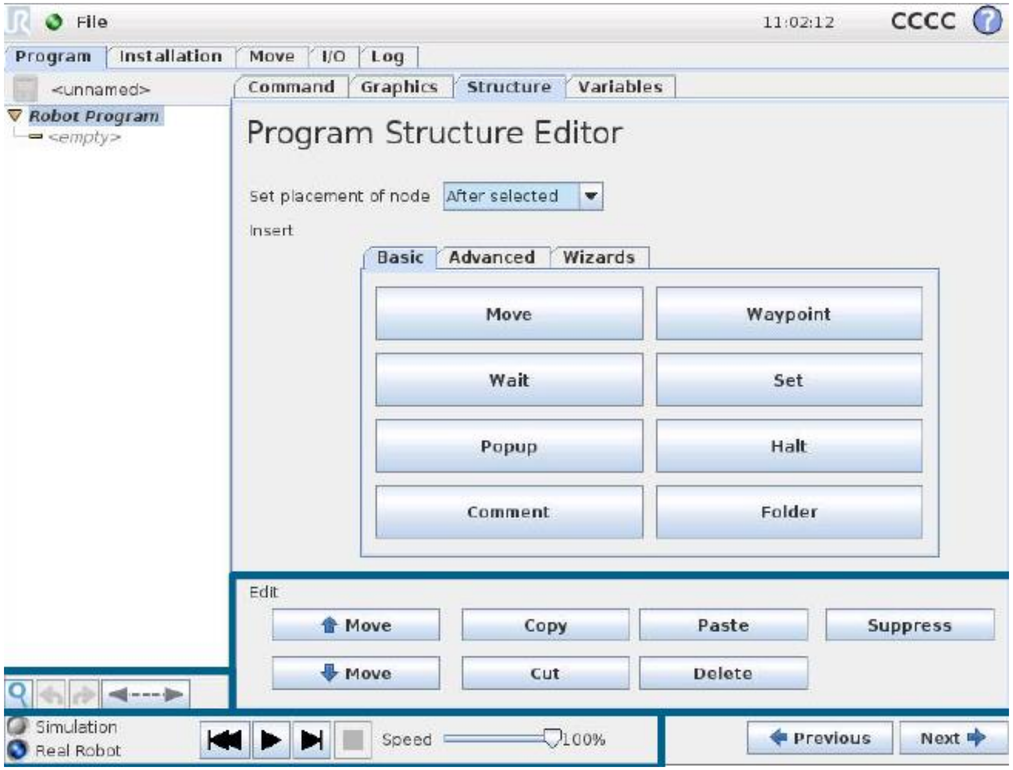
- The Edit area contains basic actions like Copy, Cut and Paste.



- *Suppressed program lines are simply skipped when the program is run. A suppressed line can be unsuppressed again at a later time. This is a quick way to make changes to a program without destroying the original contents.*

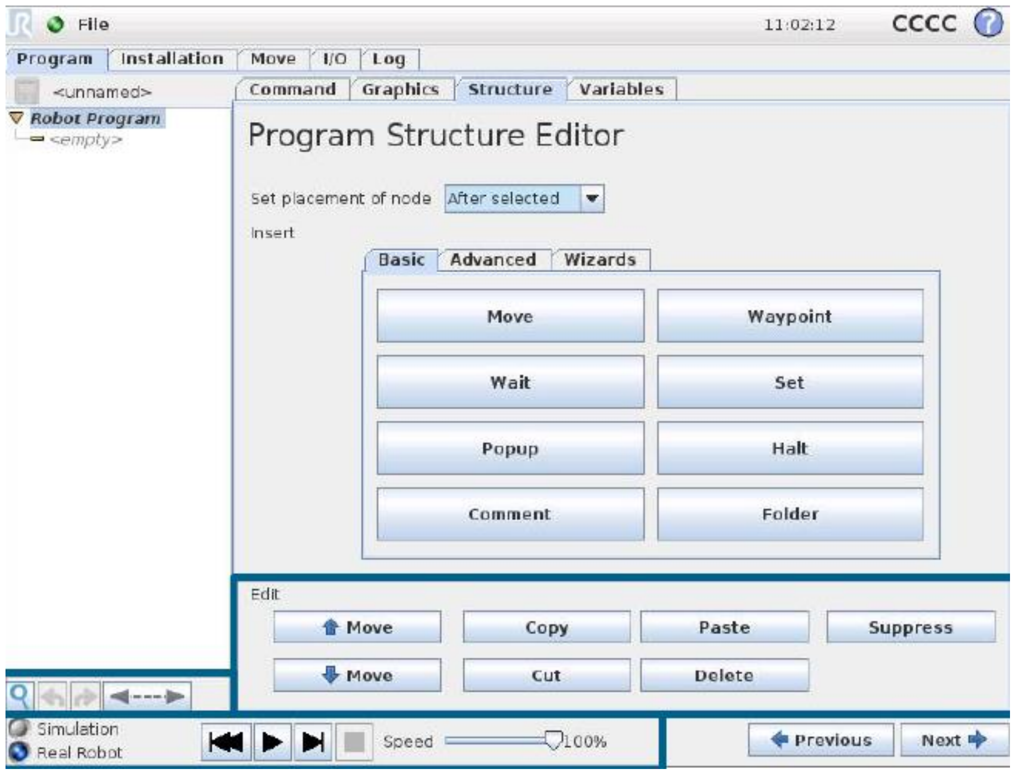
Design of Cobot Program

- The **Dashboard** (in the lowest part of the screen) features a set of buttons similar to an old-fashioned tape recorder, from which programs can be started and stopped, single-stepped and restarted.



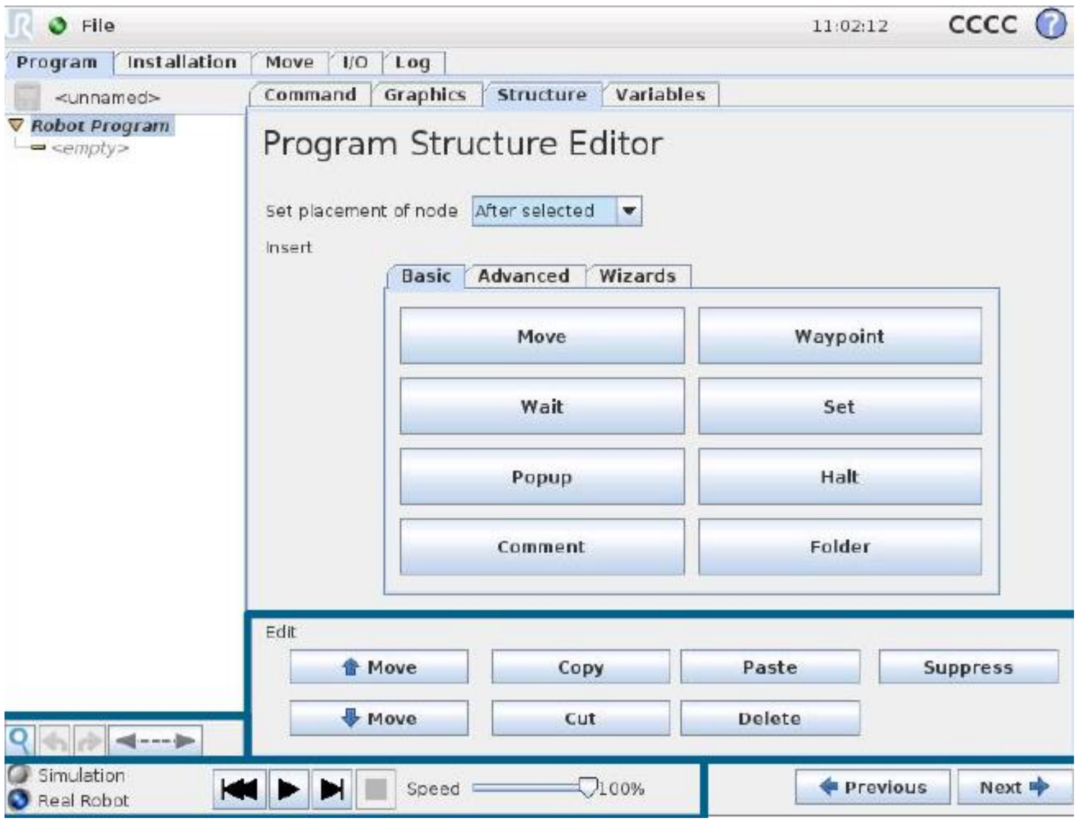
Design of Cobot Program

- The button with the icon can be used to perform a text search in the program tree. When clicked, a search text can be entered and program nodes that match will be highlighted in yellow. Press the icon to exit search mode.



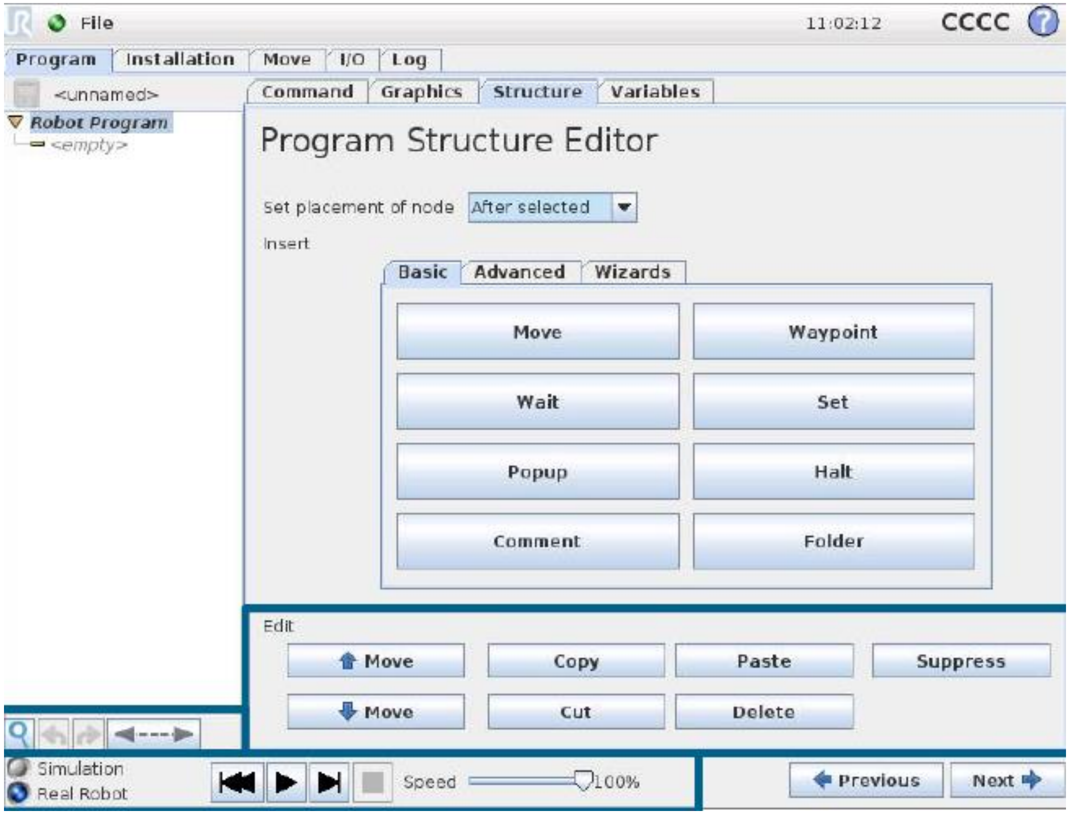
Design of Cobot Program

- The buttons with icons and below the program tree serve to undo and redo changes made in the program tree and in the commands it contains.



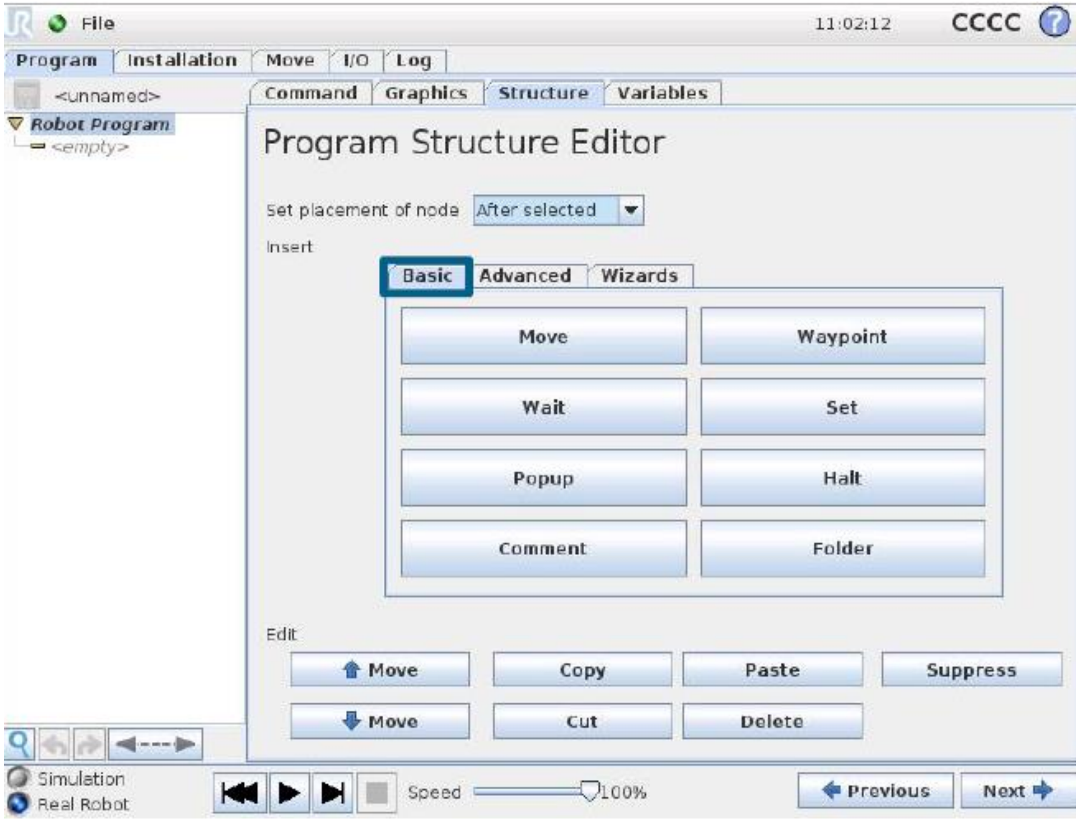
Design of Cobot Program

- Clicking the button with the ◄---► icon below the program tree jumps to the current executing or the last executed command in the tree.



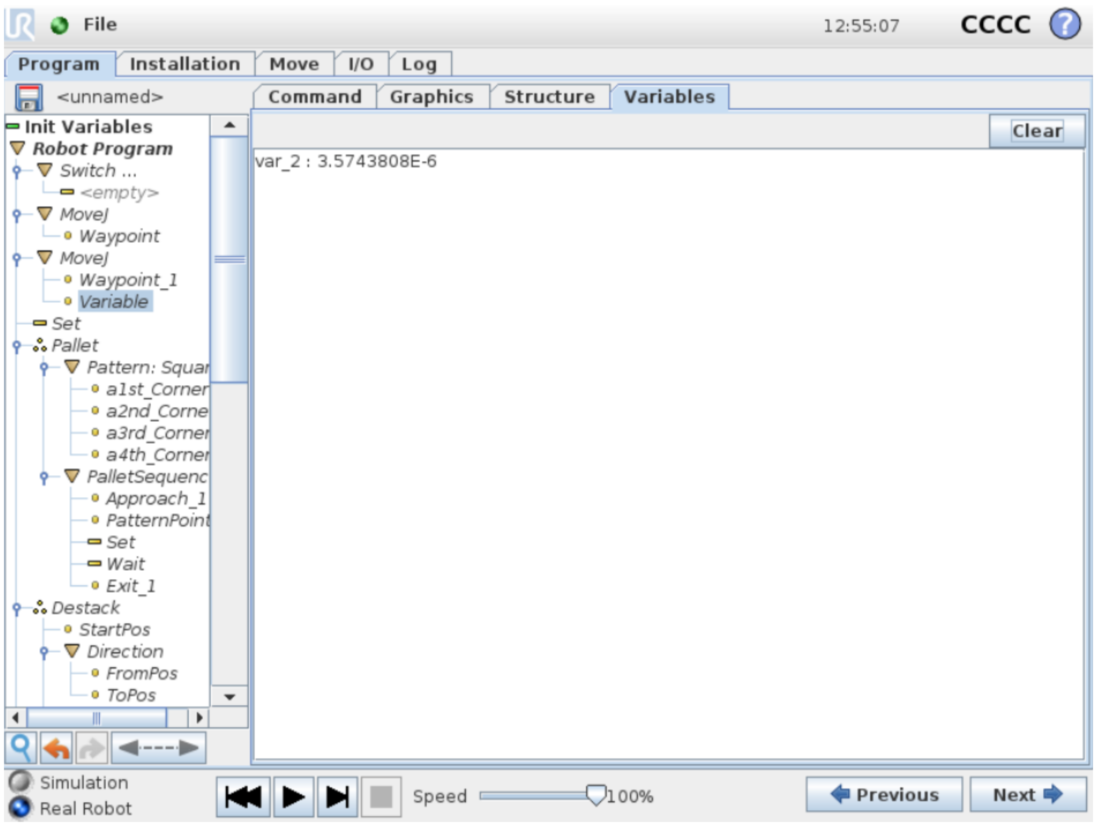
Design of Cobot Program

- There are three tabs in the **Structure Tab**, each one containing a variety of commands. They are classified based on the complexity of the command



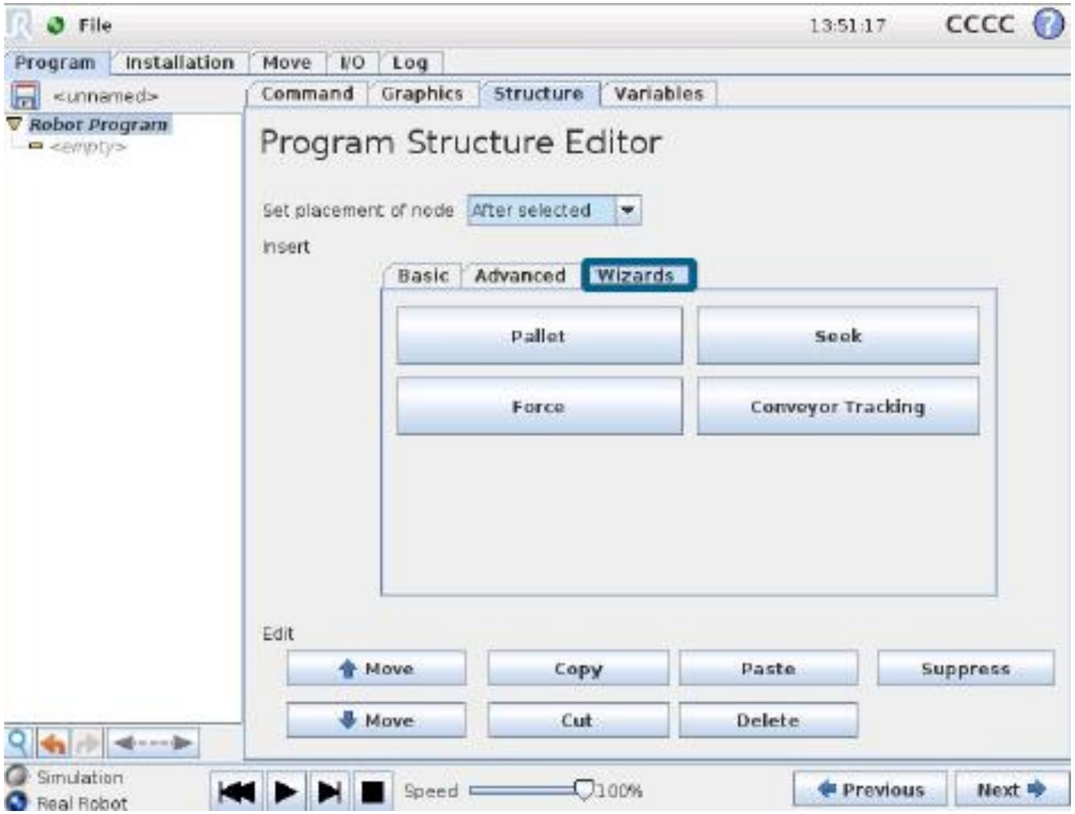
Design of Cobot Program

- The ***Variables Tab*** shows the live values of variables in the running program, and keeps a list of variables and values between program runs. It only appears when it has information to display.



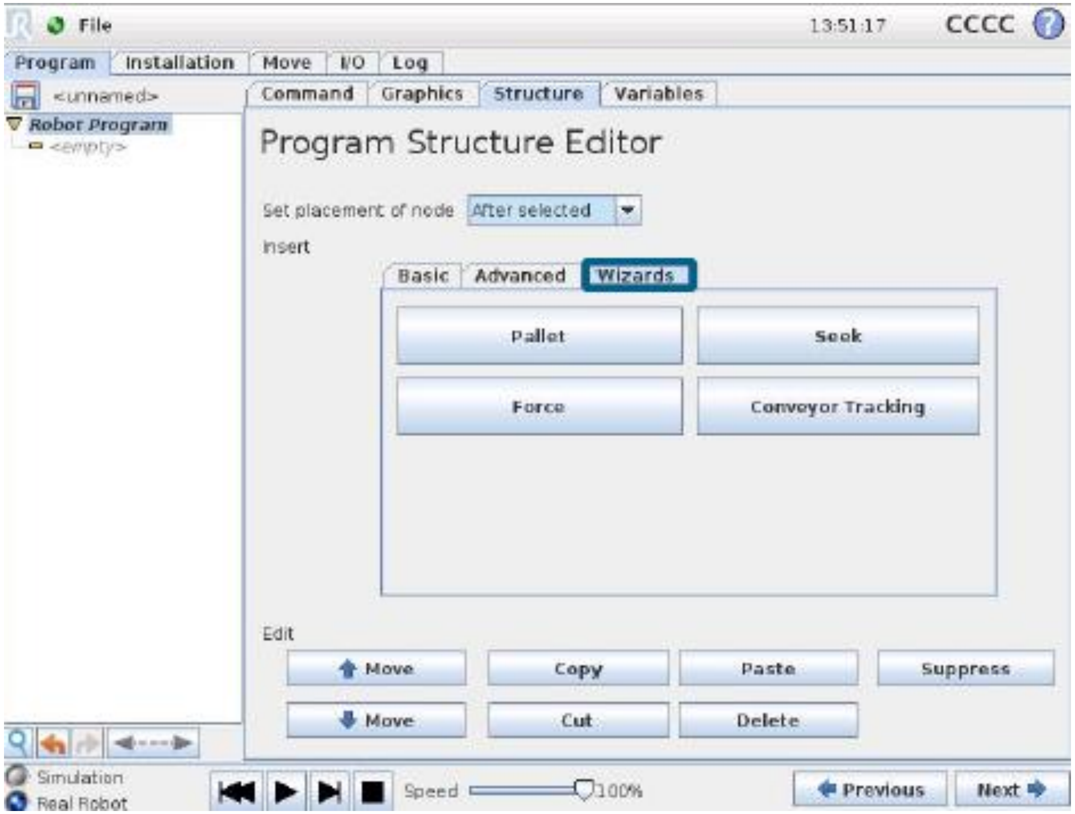
Design of Cobot Program

- A ***pallet*** operation can perform a sequence of motions in a set of places given as a pattern. At each of the positions in the pattern, the sequence of motions will be run relative to the pattern position.



Design of Cobot Program

- A ***seek*** function uses a sensor to determine when the correct position is reached to grab or drop an item. The sensor can be a push button switch, a pressure sensor or a capacitive sensor.

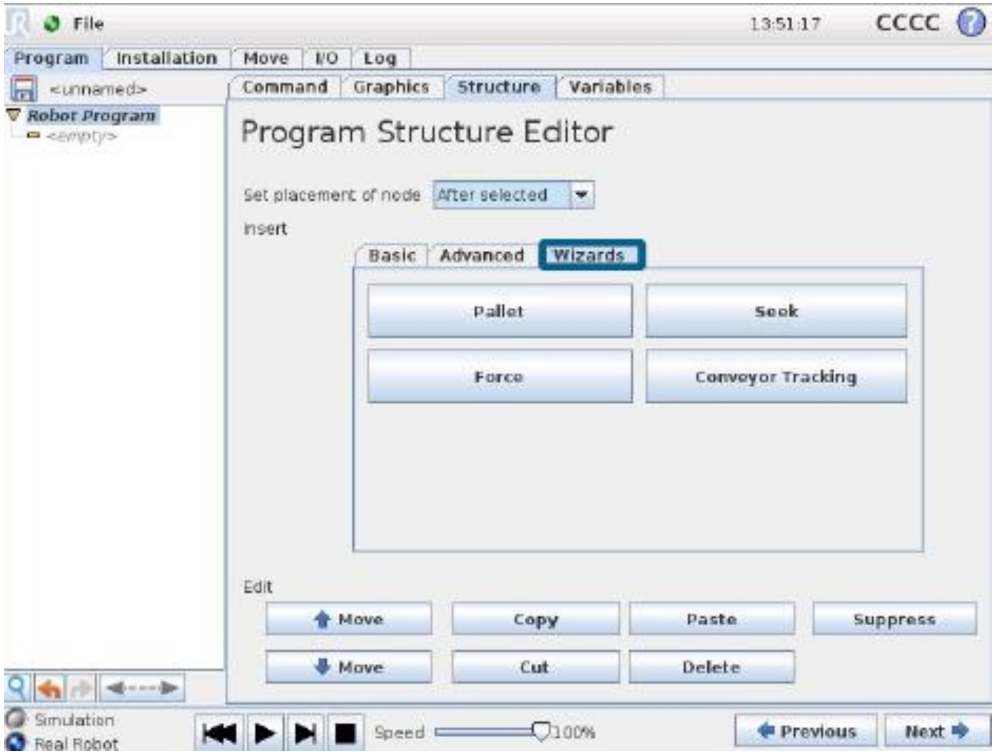


Continental

Design of Cobot Program

DIGIFOF — DIGITAL DESIGN SKILLS FOR FACTORIES OF THE FUTURE

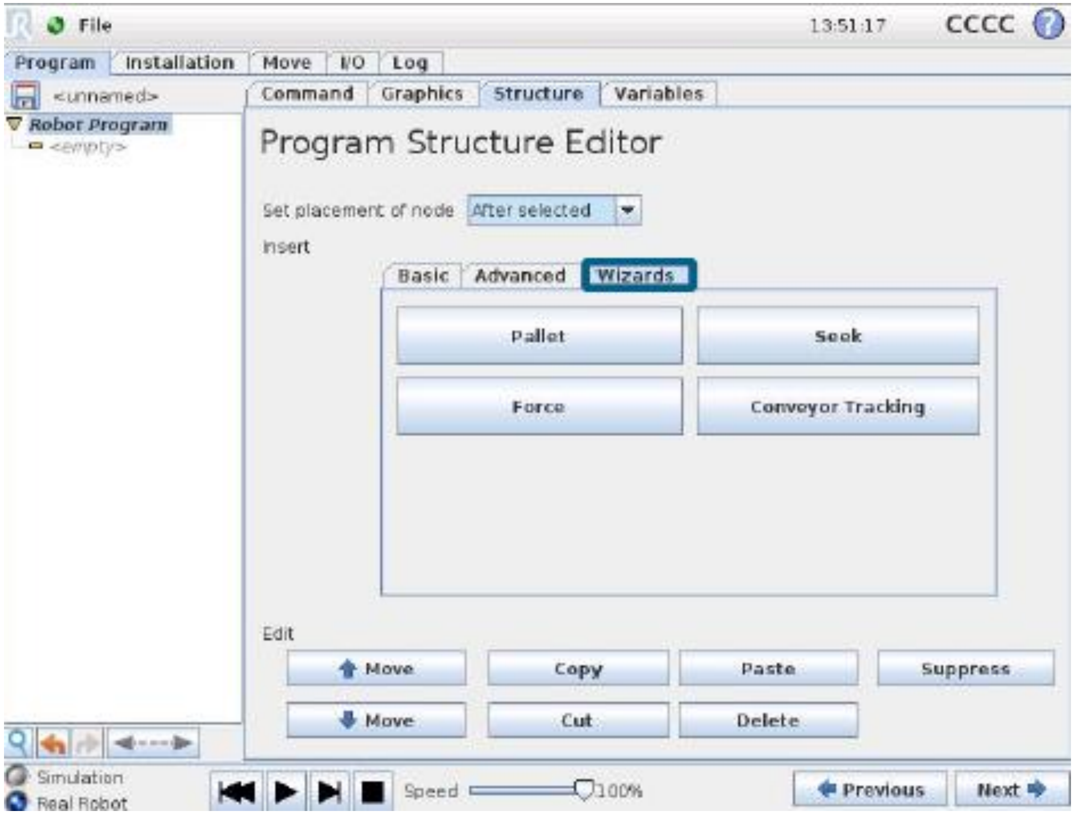
- ***Force*** mode allows for compliance and forces in selectable axis in the robot's workspace. All robot arm movements under a Force command will be in Force mode. When the robot arm is moving in force mode, it is possible to select one or more axes in which the robot arm is compliant.



- It is also possible to make the robot arm itself apply a force to its environment, e.g. a workpiece.

Design of Cobot Program

- When using a conveyor, the robot can be configured to track it's movement. While the program is executing under the ***Conveyor Tracking*** node, the robot will adjusts it's movements to follow the conveyor.



Content

B. **Practical Module**

- Program basic movement with the cobot from SAS Lab.

Content

B. Practical Module

- Implement a cobot – design solution for an allocated process:
  - Understand process goals
  - Create specification
  - Evaluate technologies
  - Design System
  - Optimize solution
- Practice with a cobot;

Thank you for your attention!

Questions?