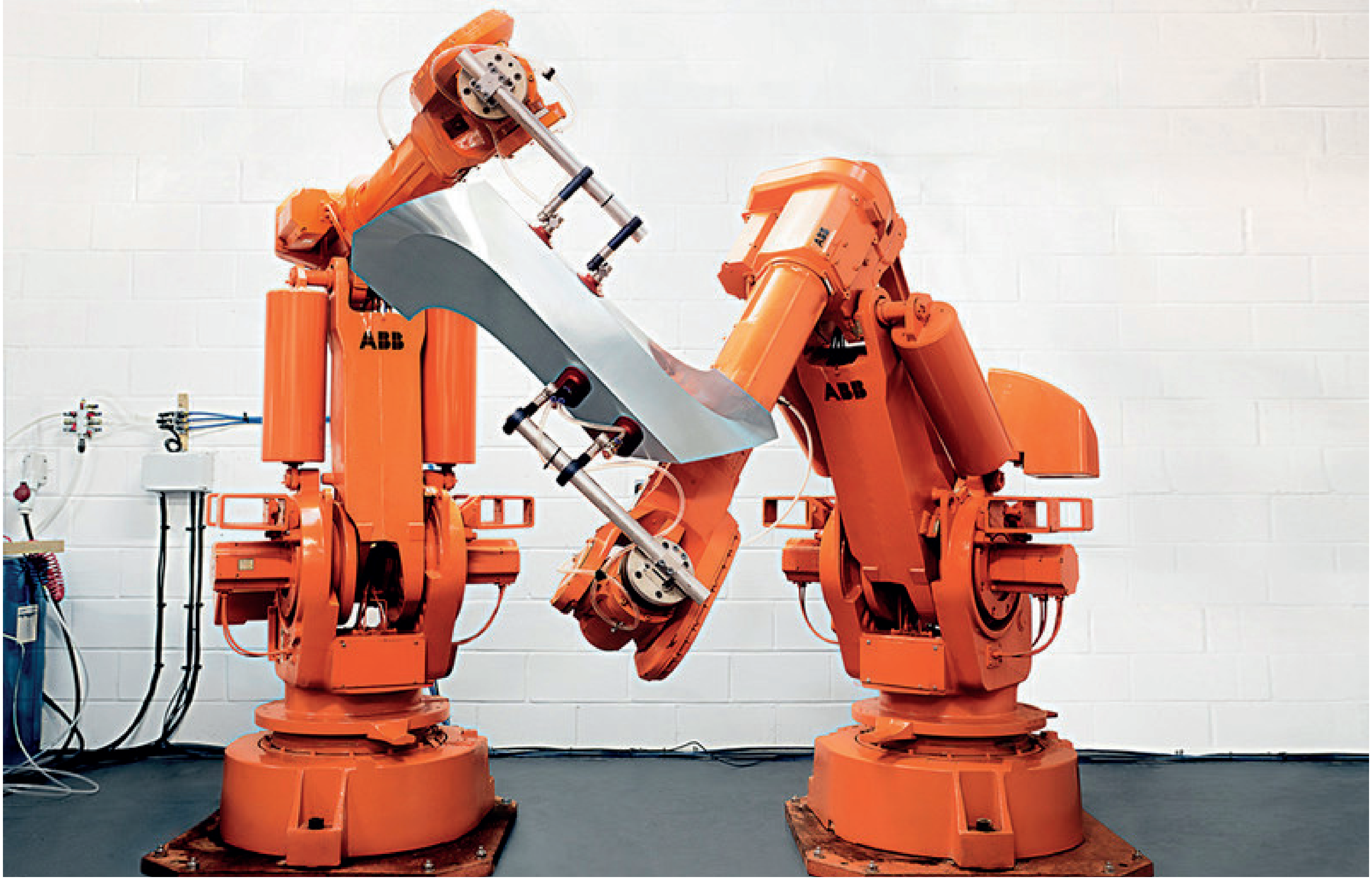
Robot induction

Table of contents

Introduction to robotics

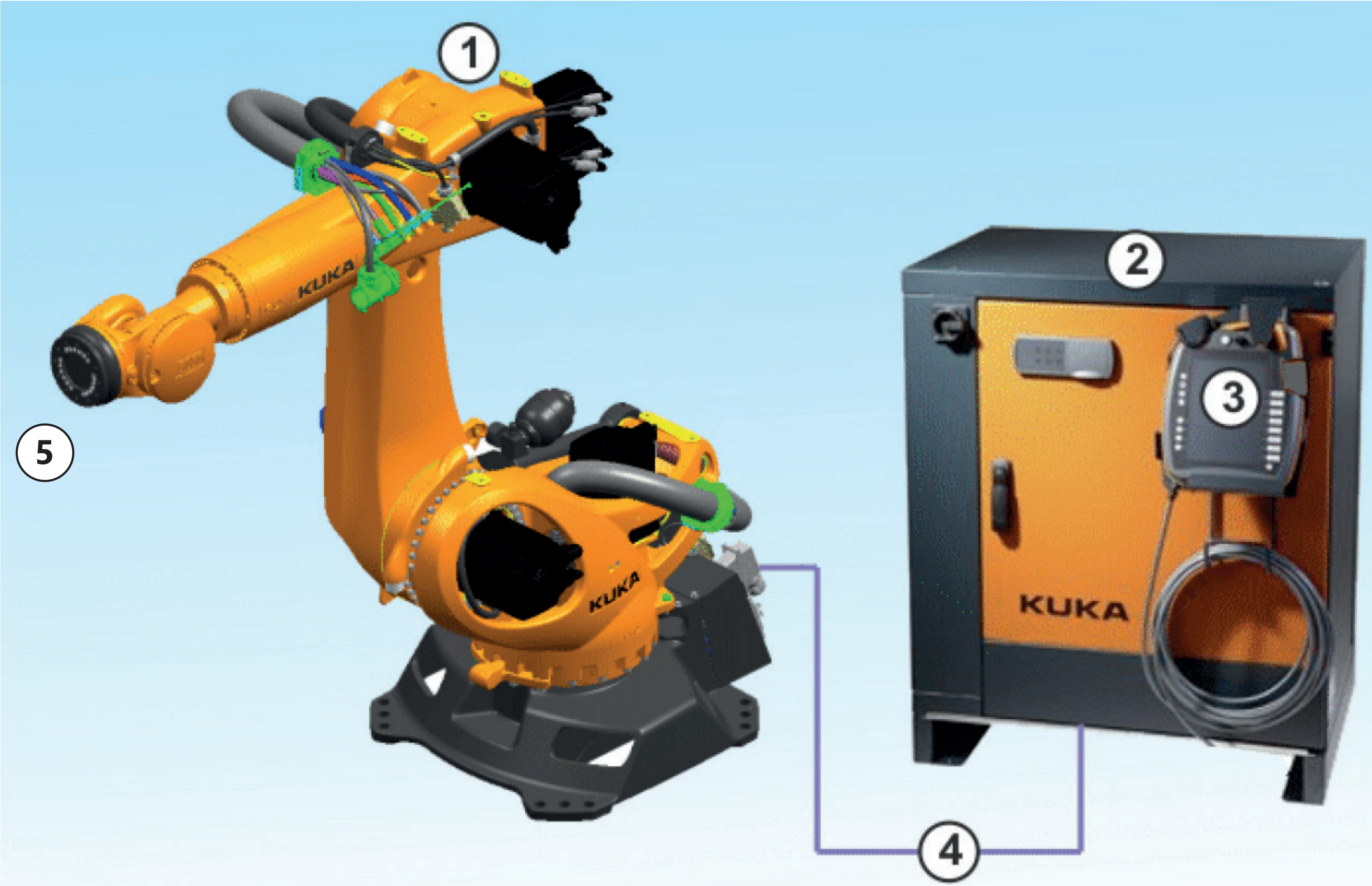


Milestones in robotics history

1947 — the first servoed electric powered teleoperator is developed
1948 — a teleoperator is developed incorporating force feedback
1949 — **research on numerically controlled milling machine is initiated**
1954 — George Devol designs the first programmable robot
1956 — Joseph Engelberger, a Columbia University physics student, buys the rights to Devol's robot and founds the Unimation Company
1961 — **the first Unimate robot is installed in a Trenton, New Jersey plant of General Motors to tend a die casting machine**
1961 — the first robot incorporating force feedback is developed
1963 — the first robot vision system is developed
1971 — **the Stanford Arm is developed at Stanford University**
1973 — the first robot programming language (WAVE) is developed at Stanford
1974 — Cincinnati Milacron introduced the T3 robot with computer control
1975 — Unimation Inc. registers its first financial profit
1976 — the Remote Center Compliance (RCC) device for part insertion in assembly is developed at Draper Labs in Boston
1976 — Robot arms are used on the Viking I and II space probes and land on Mars
1978 — **Unimation introduces the PUMA robot, based on designs from a General Motors study**
1979 — **the SCARA robot design is introduced in Japan**
1981 — the first direct-drive robot is developed at Carnegie-Mellon University
1982 — Fanuc of Japan and General Motors form GM Fanuc to market robots in North America
1983 — Adept Technology is founded and successfully markets the direct drive robot
1986 — the underwater robot, Jason, of the Woods Hole Oceanographic Institute, explores the wreck of the Titanic, found a year earlier by Dr. Robert Barnard.
1988 — Staubli Group purchases Unimation from Westinghouse
1988 — the IEEE Robotics and Automation Society is formed
1993 — the experimental robot, ROTEX, of the German Aerospace Agency (DLR) was flown aboard the space shuttle Columbia and performed a variety of tasks under both teleoperated and sensor-based offline programmed modes
1996 — Honda unveils its Humanoid robot; a project begun in secret in 1986
1997 — the first robot soccer competition, RoboCup-97, is held in Nagoya, Japan and draws 40 teams from around the world
1997 — the Sojourner mobile robot travels to Mars aboard NASA's Mars PathFinder mission
2001 — Sony begins to mass produce the first household robot, a robot dog named Aibo
2001 — **the Space Station Remote Manipulation System (SSRMS) is launched in space on board the space shuttle Endeavor to facilitate continued construction of the space station**
2001 — the first telesurgery is performed when surgeons in New York performed a laparoscopic gall bladder removal on a woman in Strasbourg, France
2001 — robots are used to search for victims at the World Trade Center site after the September 11th tragedy
2002 — Honda's Humanoid Robot ASIMO rings the opening bell at the New York Stock Exchange on February 15th

Types of robots

Anatomy of an industrial robot

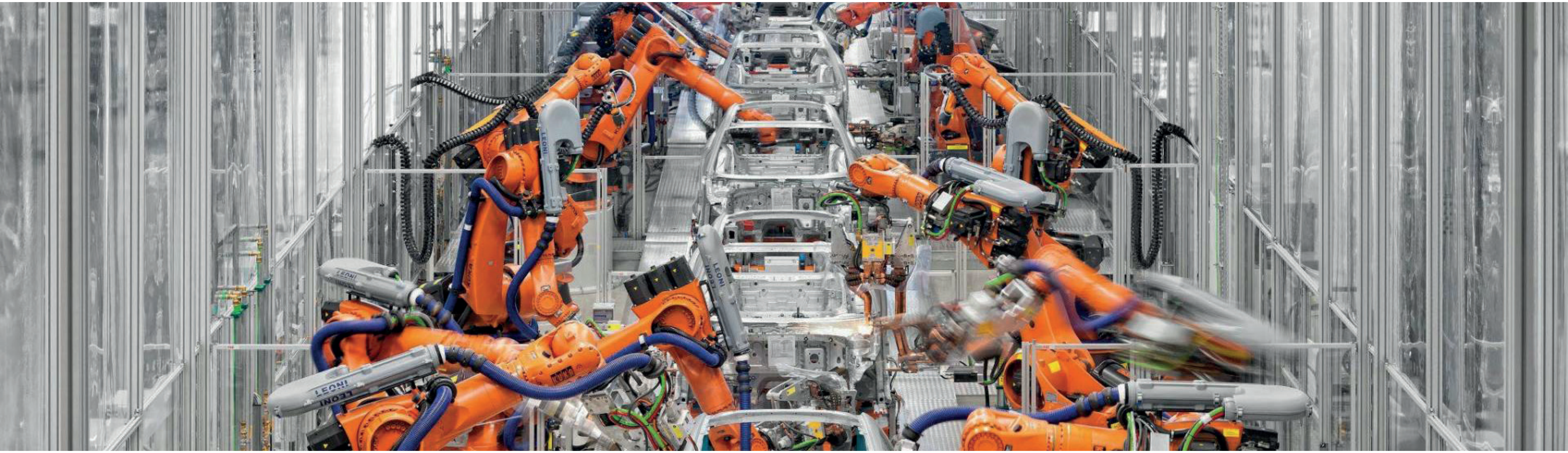


1. Manipulator
2. Controller
3. Pendant
4. Connecting cables
5. End effector (tool)

Industrial vs. Collaborative

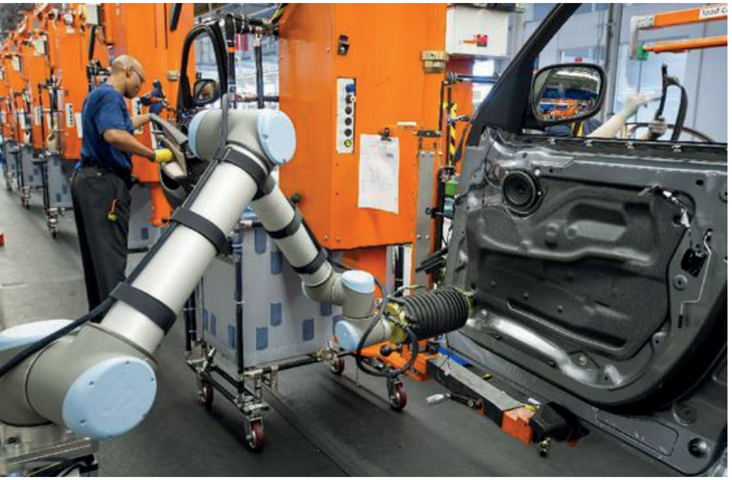
Industrial robots

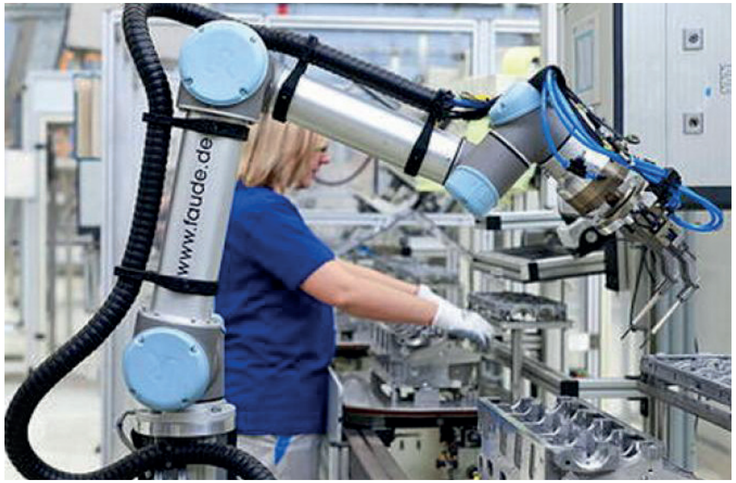
Dangerous (high speed, high torque) industrial machinery, can not be operated in human proximity, enclosure around the robot is required.



Collaborative robots

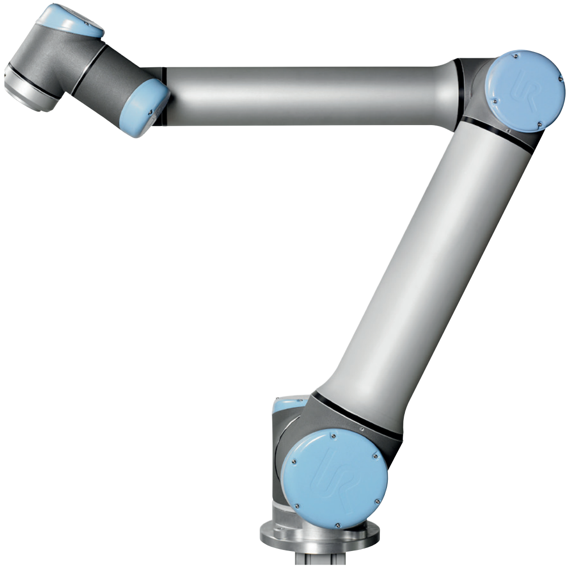
Accurate force torque monitoring and low speed makes robot safer, robot can be operated close proximity, enclosure is recommended but not required.





bMade robots Gordon street

ABB 120
Payload 3 kg



UR10
Payload 10 kg
Collaborative

Kuka Agilus kr6
Payload 6 kg

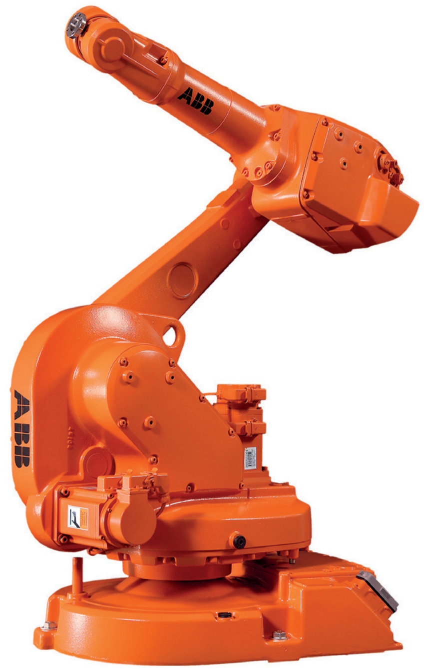


ABB IRB1600
Payload 10 kg

bMade robots Here East

Franka Emika Panda
Payload 3 kg
Collaborative

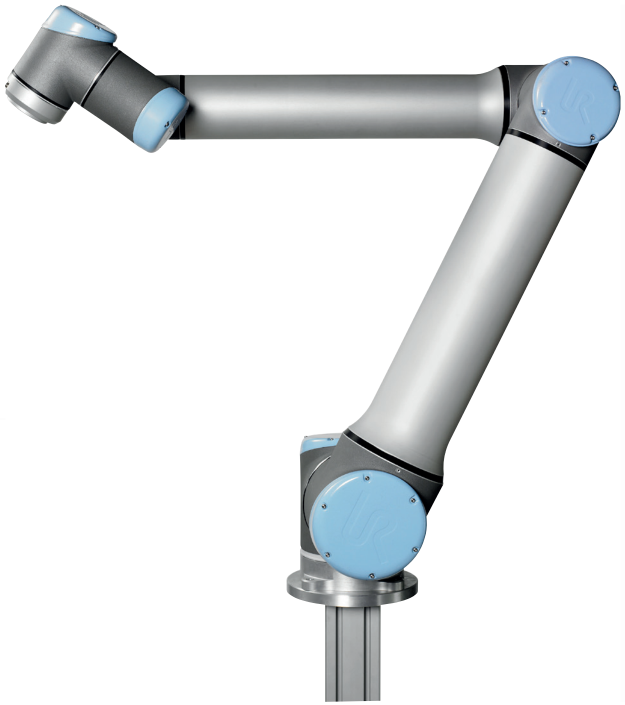
Staubli RX160
Payload 34 kg

Staubli TX200
Payload 150 kg

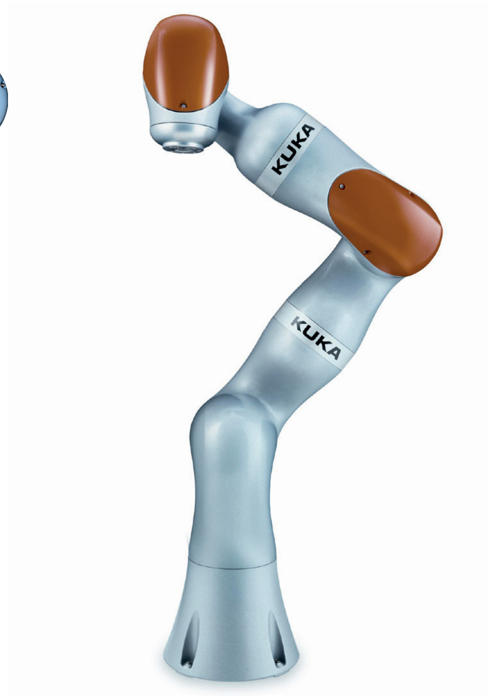
Kuka kr60 HA
Payload 60 kg

Robot joint configurations

6 axis
Spherical configuration



6 axis
Articulated configuration



7 axis
Articulated configuration

Delta robot

Robotic concepts & terminology

Joint limits and working range

Axis	Range of motion, software-limited	Speed with rated payload
1	+/-170°	360 °/s
2	+45° to -190°	300 °/s
3	+156° to -120°	360 °/s
4	+/-185°	381 °/s
5	+/-120°	388 °/s
6	+/-350°	615 °/s

The direction of motion and the arrangement of the individual axes may be noted from the diagram (>>> Fig. 4-1).

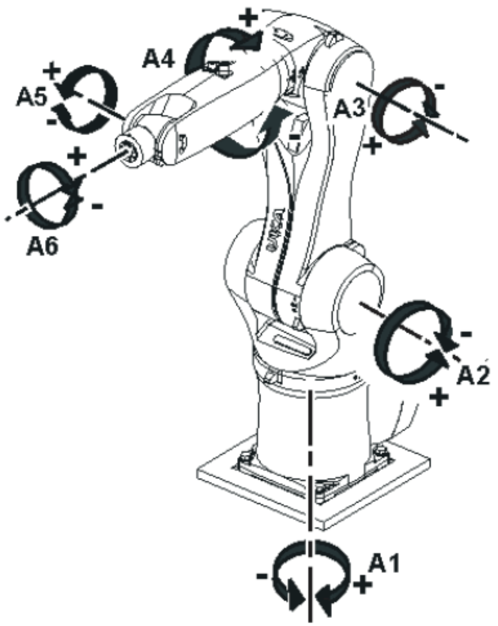
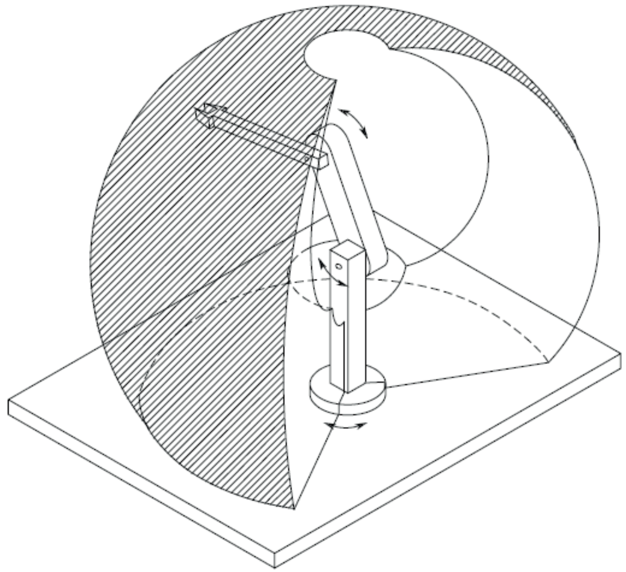
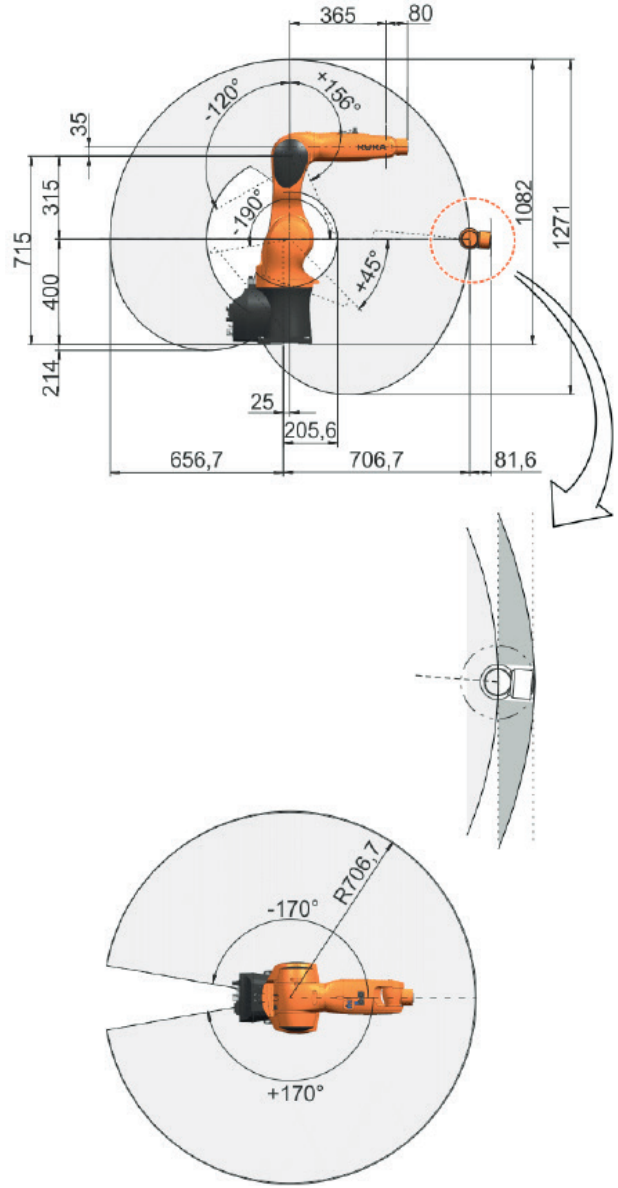


Fig. 4-1: Direction of rotation of robot axes

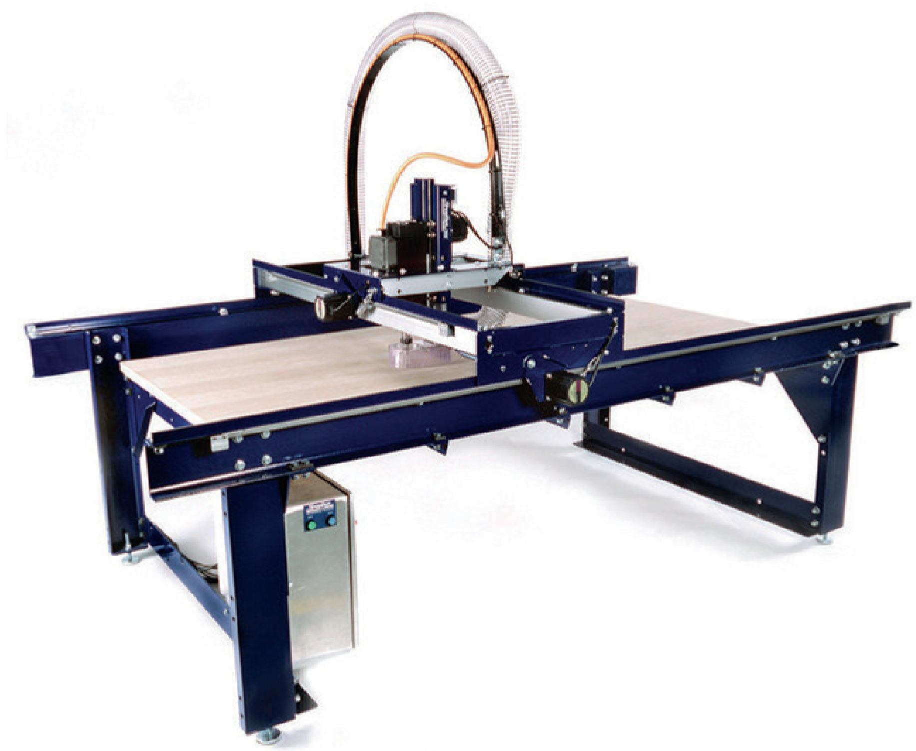




Robot reach is dependent on the desired end effector orientation, e.g. if the end effector needs to be orientated horizontally the work range will be different than if the end effector needs to be oriented vertically.

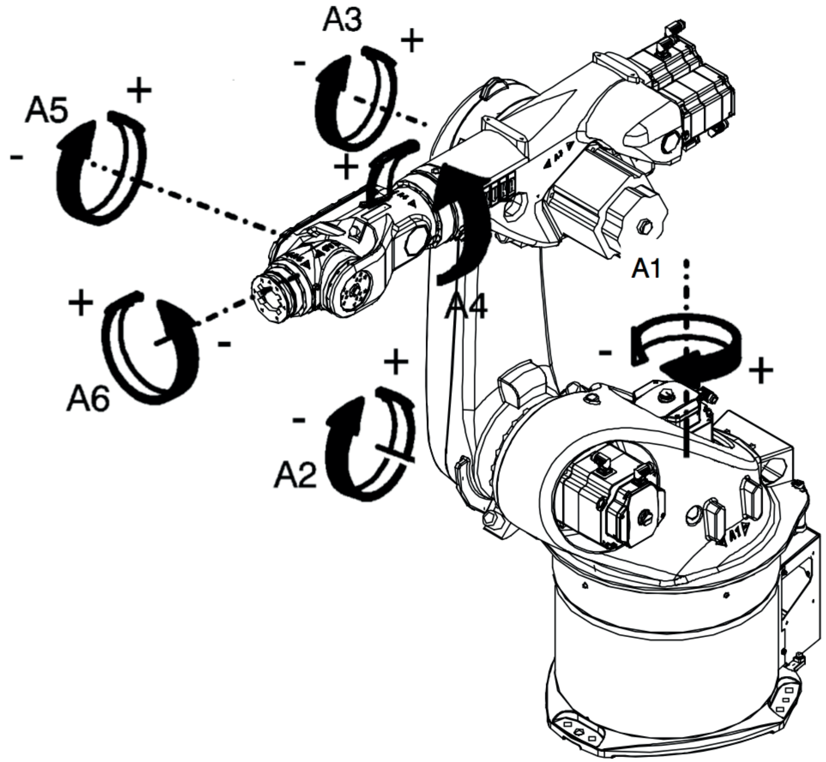
Robot coordinates or targets

3 axis CNC
3 degrees of freedom



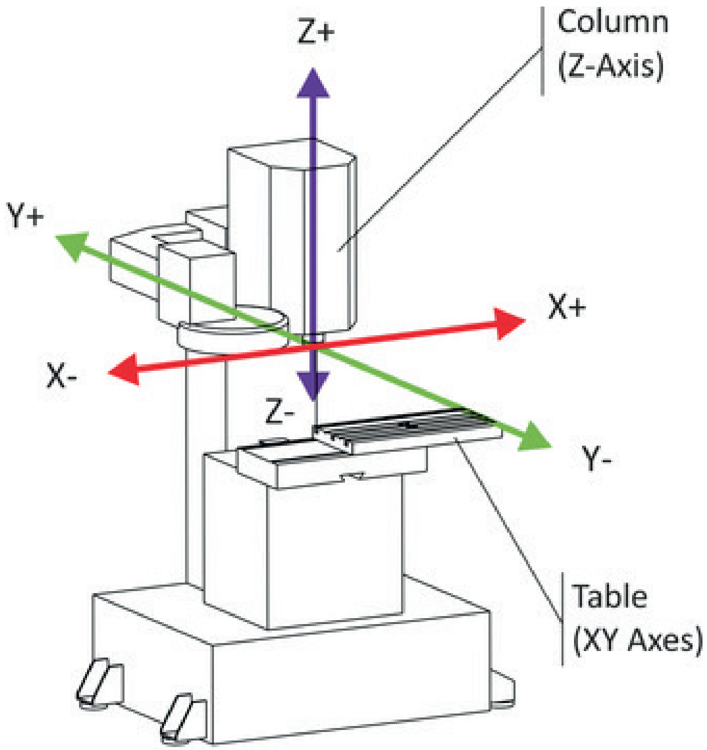
A three axis CNC requires 3 translation values **(X, Y, Z)** to define the pose of the machine.

Robot
6 or more degrees of freedom



A six axis robot requires one rotation value for each axis **(A1, A2, A3, A4, A5, A6)** to define the pose of the robot

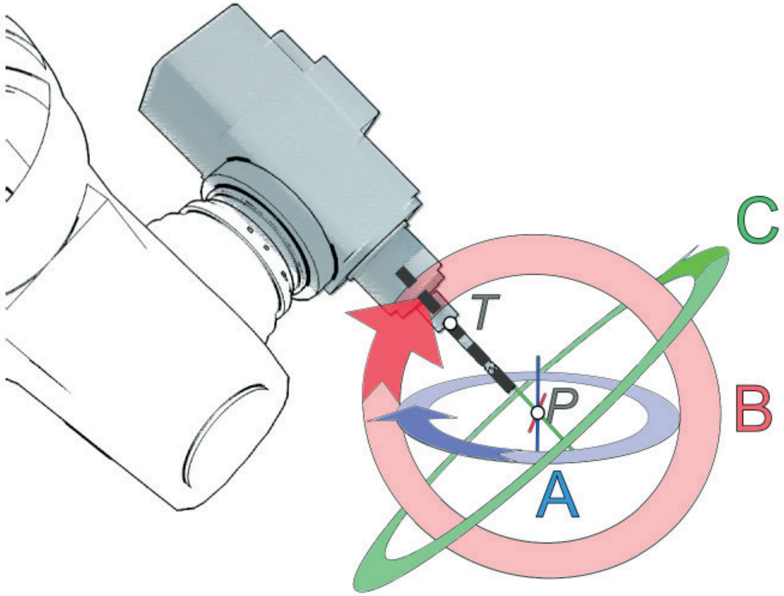
3 axis CNC



In software the target for a 3d printer or 3 axis CNC is defined by a point

(X, Y, Z)

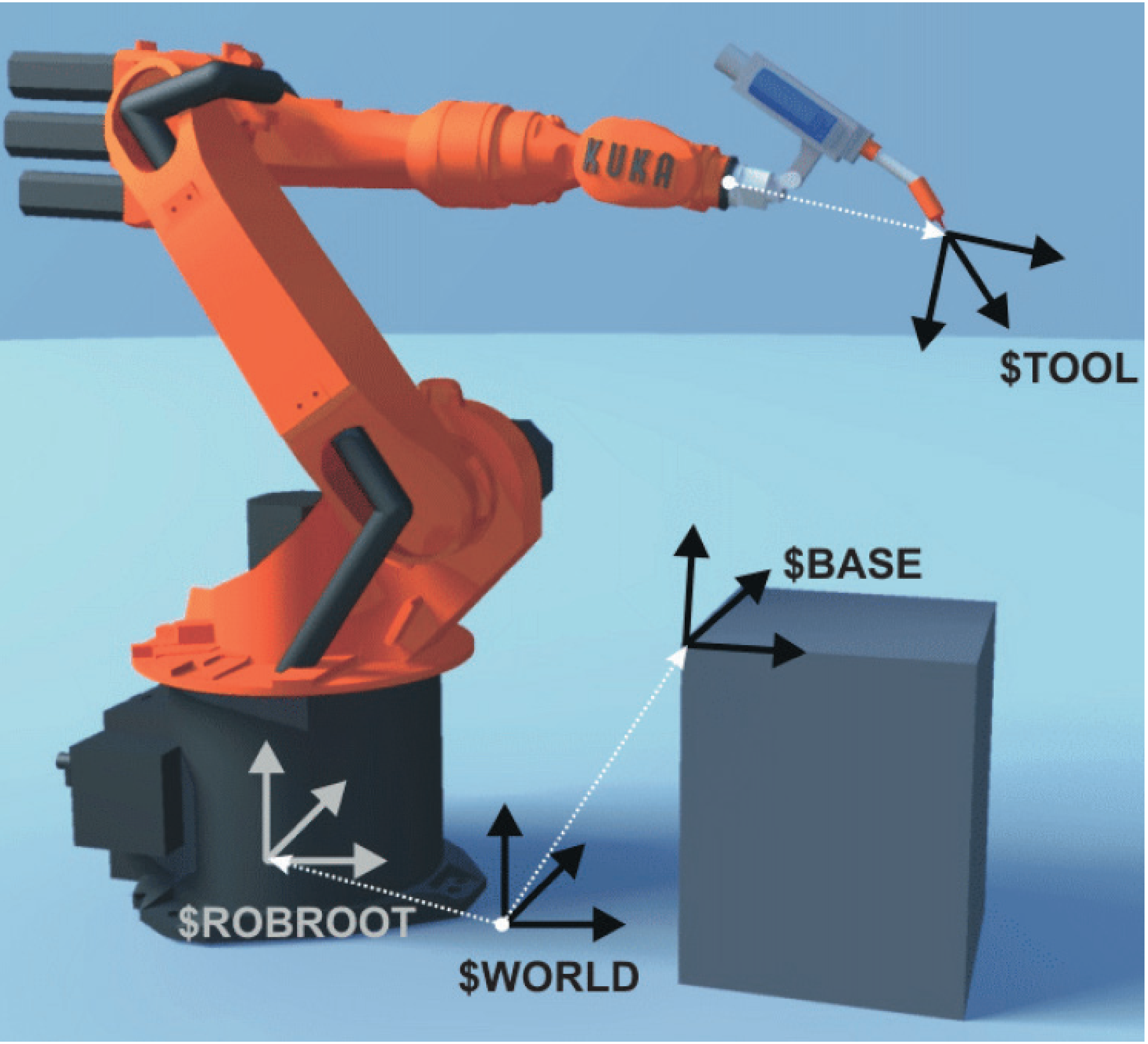
Robot



To define the position of the end effector in cartesian space a tool frame is used. This will contain a position as well as orientation, in software this is implemented trough a plane.

Euler notation used by KUKA
(X, Y, Z, A, B, C)
Axis Angle notation used by STAUBLI & UR
(X, Y, Z, rX , rY, rZ)
Quaternion notation used by ABB
(X, Y, Z, q1, q2, q3, q4)

Robot coordinate systems



World coordinate space:
In practice unless using a multi robot configuration or a robot on a linear axis the robot and world coordinate space share the same origin.

Robot coordinate space:
The robot root, defines the origin of the robot coordinate space in relation to the world coordinate space. The origin is located on the base of the robot under the first axis of rotation.
This is the coordinate space in which the robot targets are defined unless a base coordinate space is defined.

Base coordinate space:
The base root defines the origin of the base in relationship to the robot coordinate space.
When using a base the targets for the robot are defined in this coordinate space.

TCP or tool centre point
Defines where the tip of the end effector is in relationship to the centre of the flange of the robot.

Forward kinematics and inverse kinematic solutions

Forward kinematics refers to the use of the kinematic equations of a robot to compute the position of the end-effector from specified values for the joint parameters.
Maps from joint space to cartesian space.

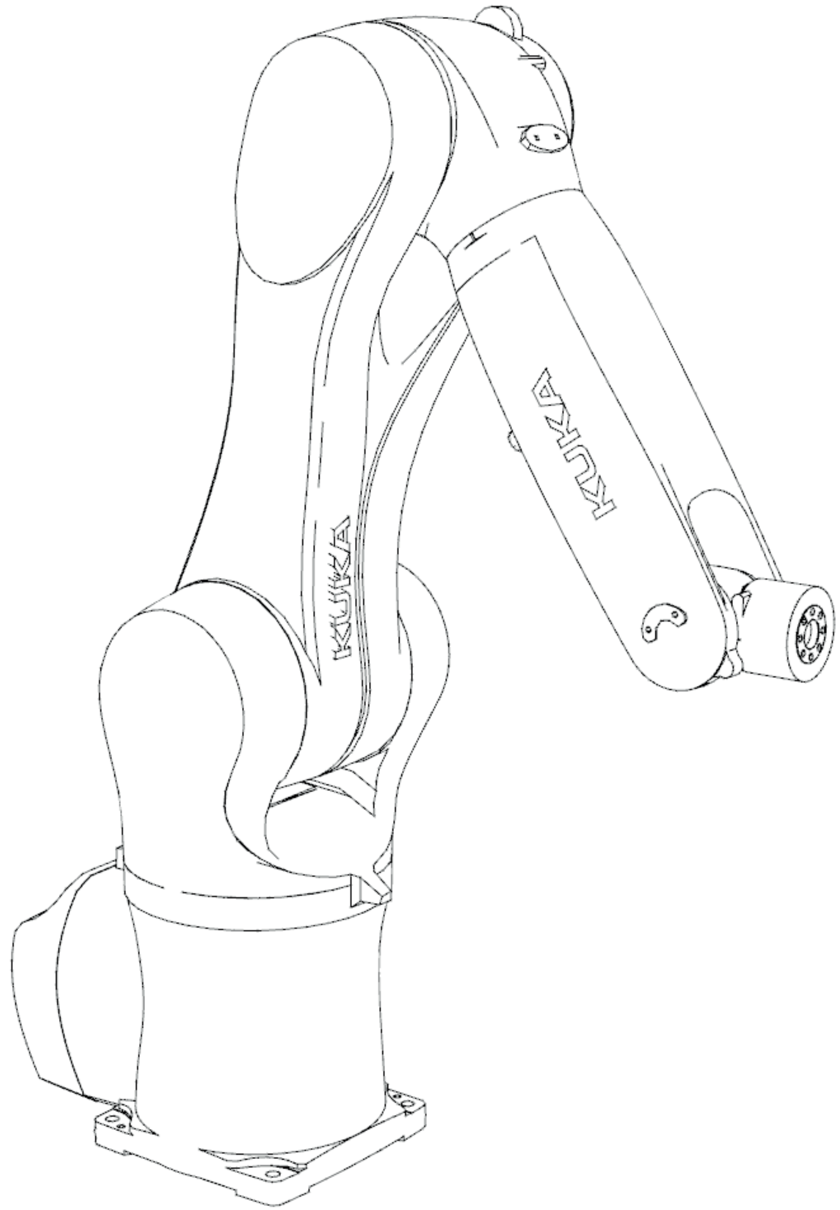
(A1, A2, A3, A4, A5, A6) is known,
(X, Y, Z, A, B, C) of TCP is unknown

e.g. (0,-72.4, 129.597, -180, 57.116, 90) to
(500,0,500,90,0,90)

Inverse kinematics refers to the use of the kinematics equations of a robot to determine the joint parameters that provide a desired position of the end-effector.
Maps from cartesian space to joint space.

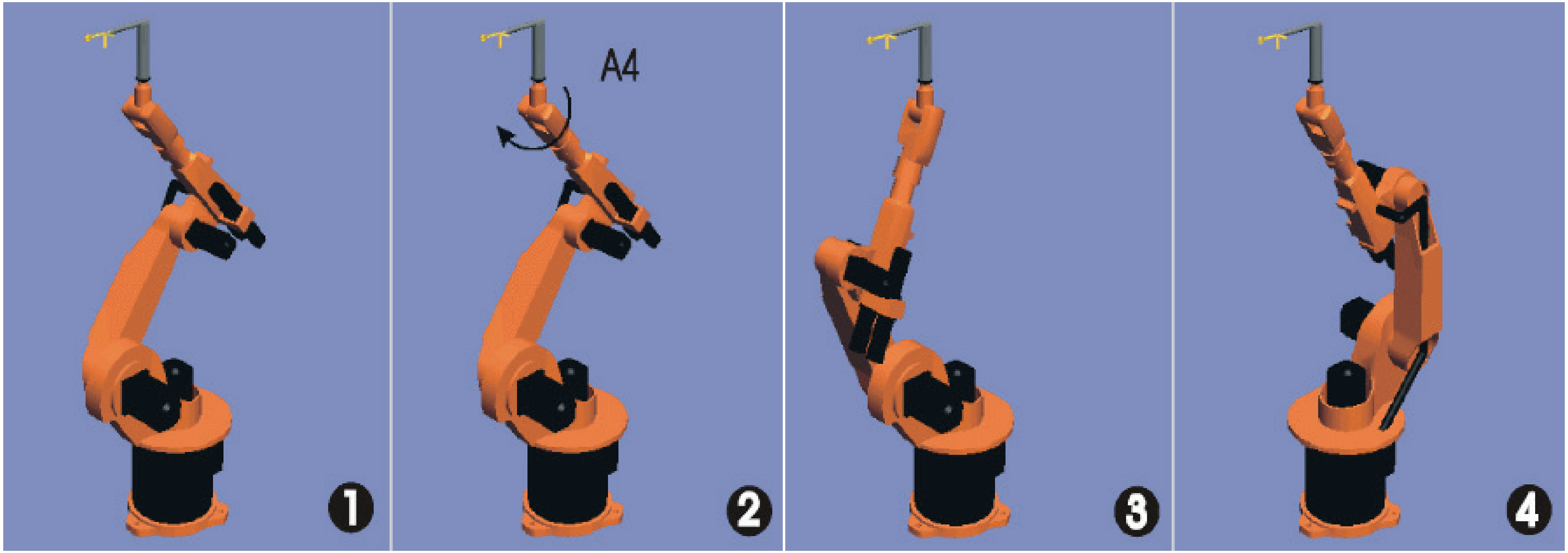
(X, Y, Z, A, B, C) of TCP is known,
(A1, A2, A3, A4, A5, A6) are unknown

e.g. (500,0,500,90,0,90) to
(0,-72.4, 129.597, -180, 57.116, 90)



Inverse kinematic solutions

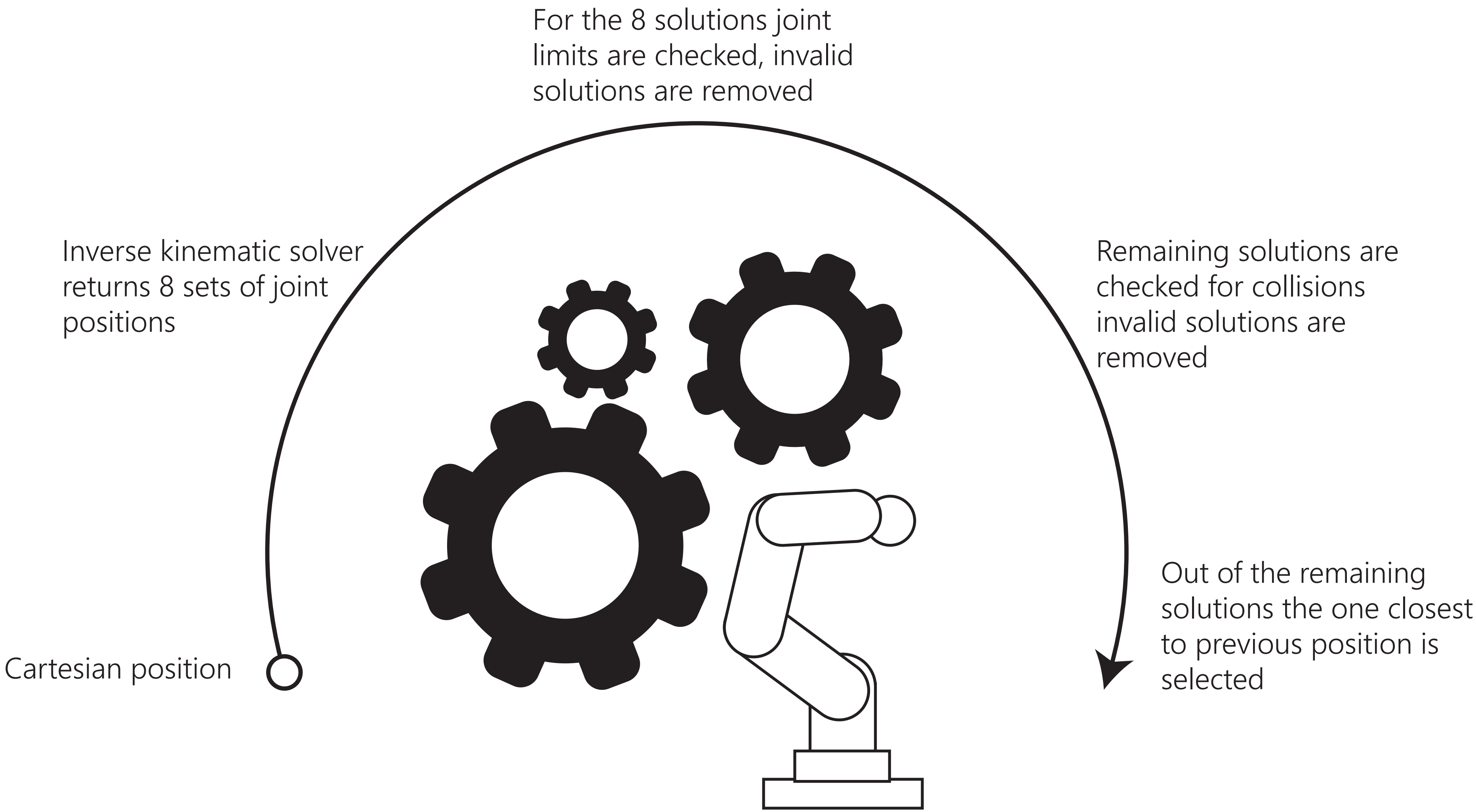
The robot can reach any cartesian position in multiple ways,
the inverse kinematics solver returns multiple solutions.



In the following images the TCP is in the same position but robot joint configuration is different

The number of inverse kinematic solutions is depended on the robot joint configuration e.g. for robots with a spherical configuration the number of solutions is equal 8. Some of these solutions will not be valid because they exceed the joint limits.

Inverse kinematic solver



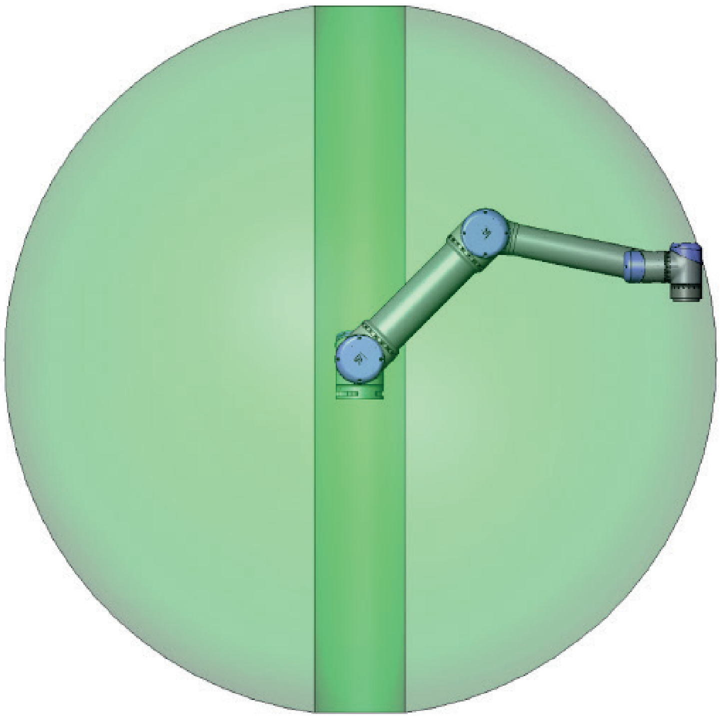
The kinematic solver will always pick the most efficient solution (requiring the least amount of rotation on each axis) when moving from one target to the next. Because changing poses requires big rotation changes on multiple axis the solver will always try and preserve the current pose.

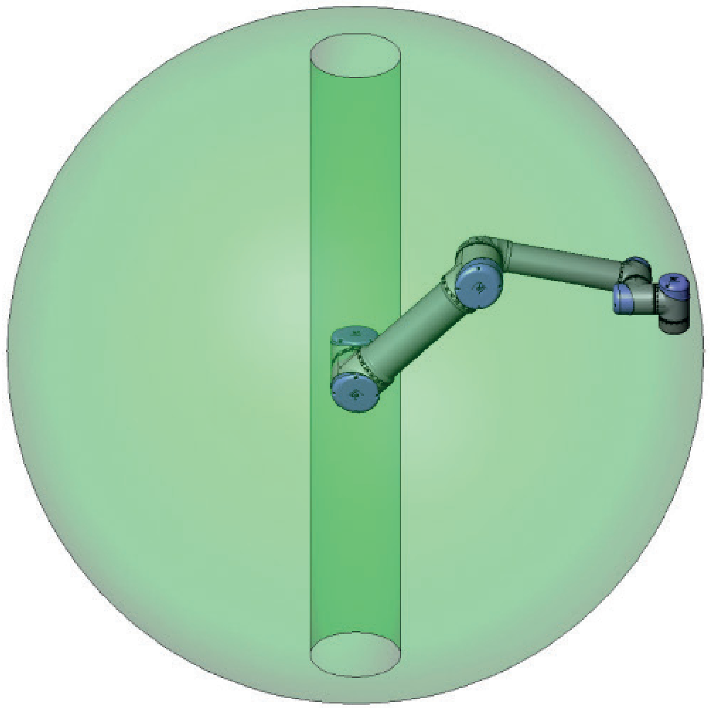
Singularities

Safety Requirements (ANSI/RIA R15.06-1999) defines a singularity as **“a condition caused by the collinear alignment of two or more robot axes resulting in unpredictable robot motion and velocities.”** It is most common in robot arms that utilize a “spherical wrist”. This is a wrist about which the three axes of the wrist, controlling yaw, pitch, and roll, all pass through a common point. An example of a wrist singularity is when the path through which the robot is traveling causes the first and third axes of the robot’s wrist (i.e. robot’s axes 4 and 6) to line up. The second wrist axis then attempts to spin 180° in zero time to maintain the orientation of the end effector. Another common term for this singularity is a “wrist flip”. The result of a singularity can be quite dramatic and can have adverse effects on the robot arm, the end effector, and the process. Some industrial robot manufacturers have attempted to side-step the situation by slightly altering the robot’s path to prevent this condition. Another method is to slow the robot’s travel speed, thus reducing the speed required for the wrist to make the transition.

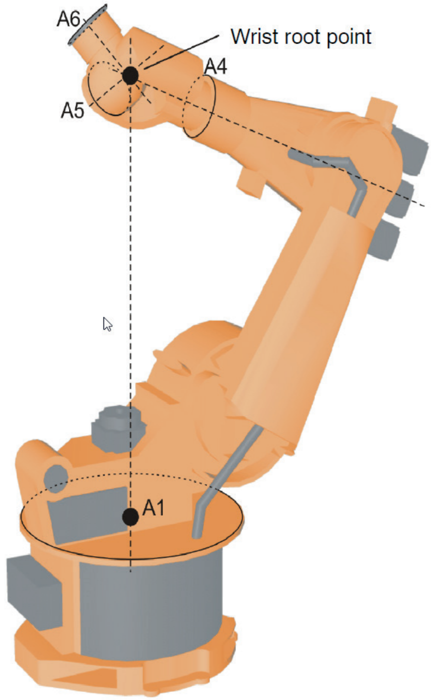
- Singularities are depended on the robot joint configuration.
- Singularities only occur when moving the robot in cartesian space.

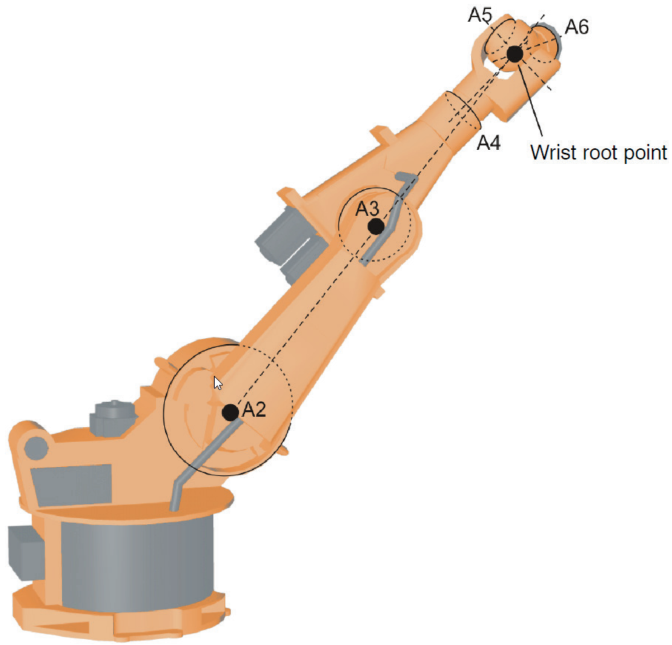
Singularities for a 6 axis articulated robot

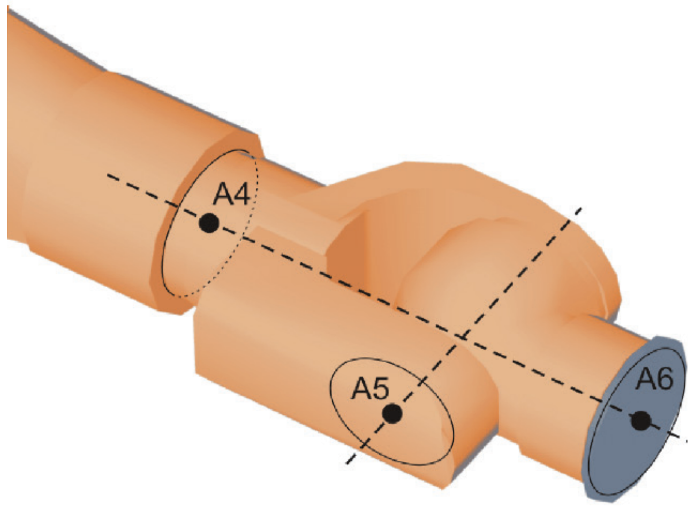




Singularities for a 6 axis spherical wrist robot







Shoulder singularity

Here, the wrist root point, located at the intersection of axes A4, A5 and A6, is positioned directly on axis 1. The position of axis 1 cannot be determined unambiguously by means of the reverse transformation and can thus take any value.

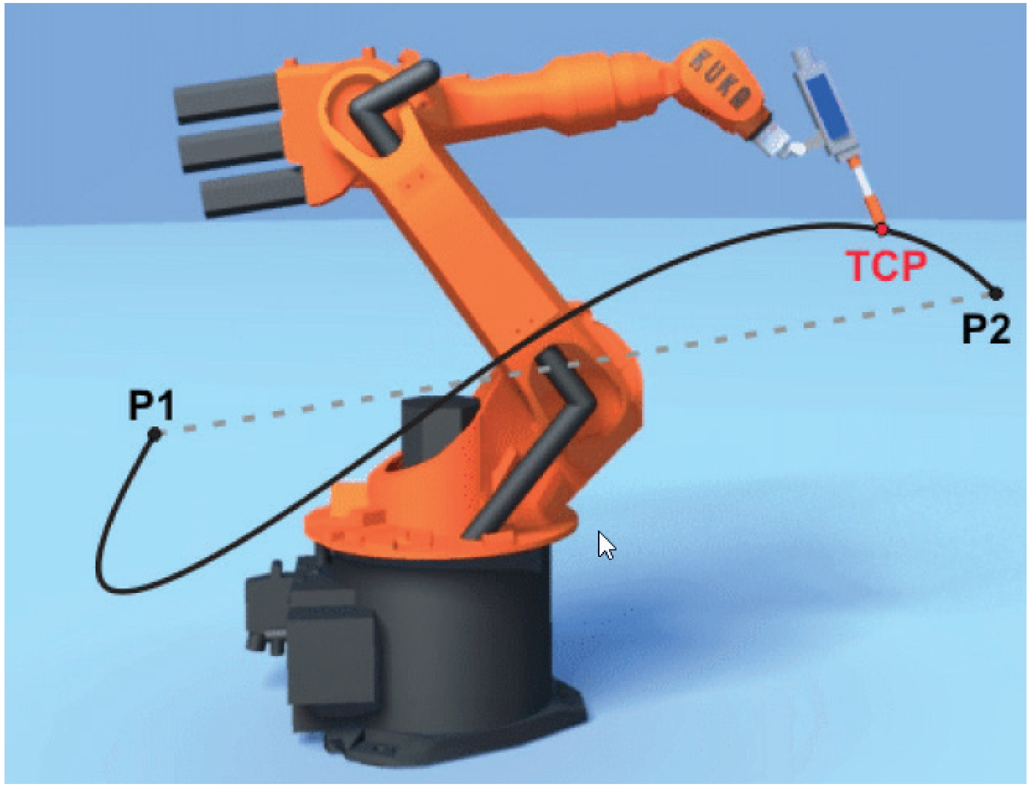
Elbow singularity

In this case the robot is at the limit of it's work envelope. Although reverse transformation does provide unambiguous axis angles, low Cartesian velocities result in high axis velocities for axes 2 and 3.

Wrist singularity

In this case, axes 4 and 6 are parallel. It is not possible to determine the positions of these two axes unambiguously by means of reverse transformation as there is an infinite number of axis positions for A4 and A6 for which the sum of the axis angles is identical.

Motion types

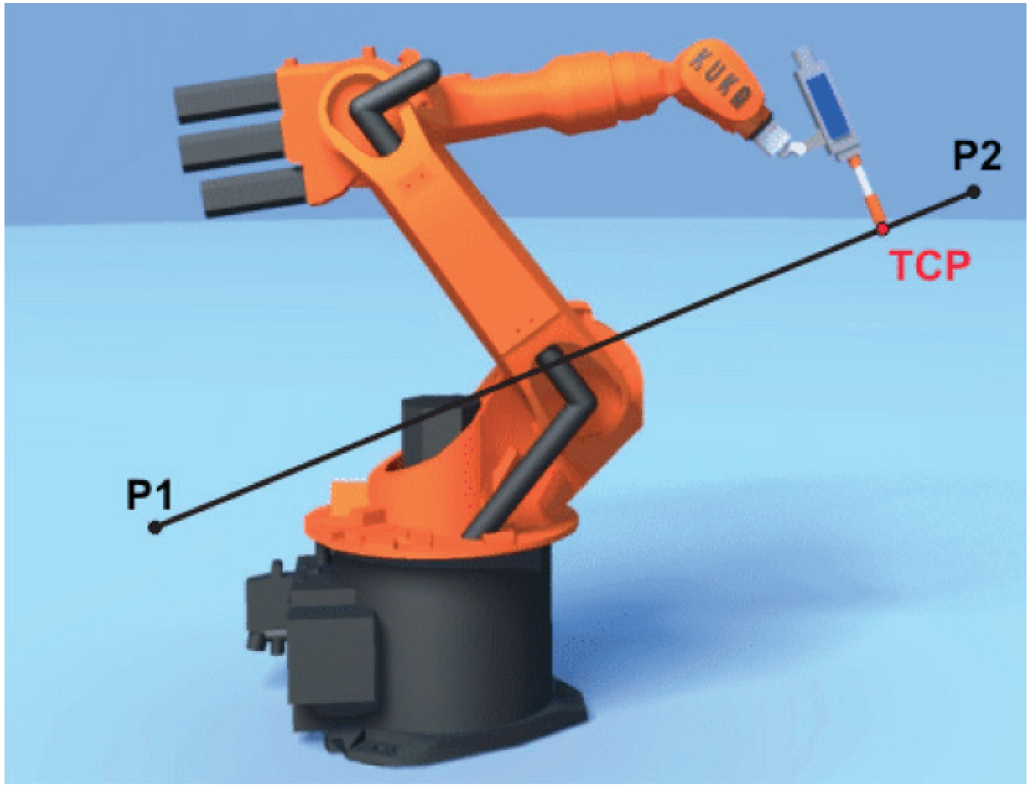


Move J **Joint space interpolation**

MoveJ(A1,A2,A3,A4,A5,A6)

\+ Good for posing the robot
\+ Predictable robot movement (vs. simulation)
\- Slightly lower precision

For moving trough the workspace



Move L **Linear interpolation**

MoveL(X,Y,Z,rX,rY,rZ), MoveL(X,Y,Z,A,B,C)
MoveL(X,Y,Z,q1,q2,q3,q4)

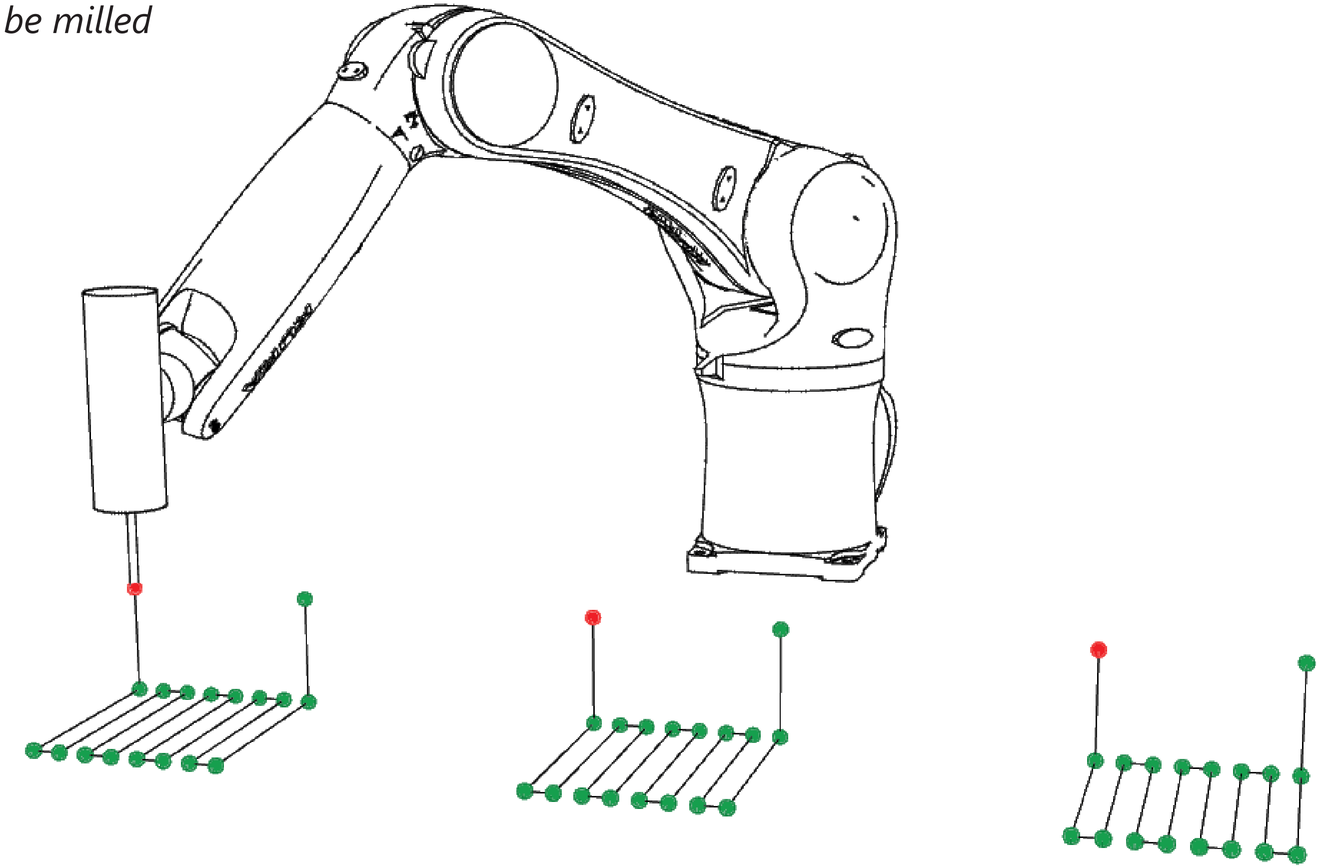
\+ Maximum precision
\- Can result in unpredictable robot movement (vs. simulation)

For toolpath operations

Additional movement types are available such as Move C (circular move), Kuka also support spline movements

Combining motion types

Example: 3 pockets need to be milled



Defining the first target of each operation as **Joint move** allows us to:
- Set the pose of the robot
- Avoid singularities when travelling in between the operations

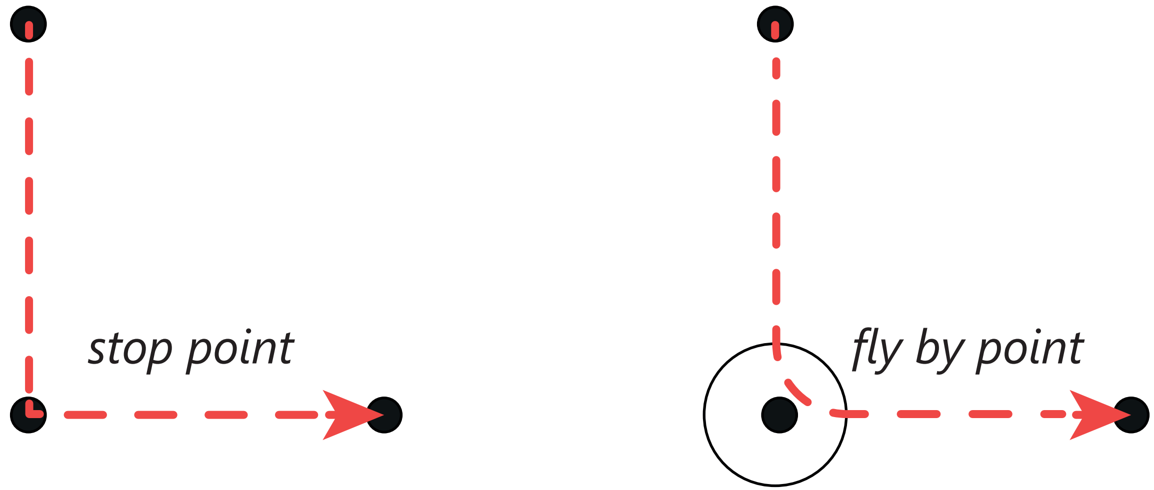
Defining the subsequent targets of each operation as **Linear movements** allows us to:
- Move the robot with maximum accuracy
- Move the robot in straight lines

Position approximation

We distinguish between:

- **Stop point:** (approximation radius = 0) high accuracy but robot slows down when nearing point
- **Fly by point:** (approximation radius > 0) lower accuracy but robot maintains constant speed

robot path



robot speed

Terminology by brand

KUKA	C_DIS
ABB	Zone
STAUBLI	Blend
UR	Blend

If consecutive targets have overlapping radii the robot will either skip the target or treat the target as stop point!

End effectors and IO

IO or inputs and outputs

IO allows for communication with end effector, other robots or auxiliary tools.

The following types of IO can be available depending on the specific robot:

- Digital in:	0 or 1	0V or 24V
- Analogue in:	0 to 1	0V to 10V
- Digital out:	0 or 1	0V or 24V
- Analogue out:	0 to 1	0V to 10V

- ABB IRB 1600:	DI, DO
- ABB IRB 120:	DI, DO
- KUKA KR6:	DI, DO, AnOut
- KUKA KR60HA:	DI, DO, AnOut
- UR 10:	DI, DO, AnIn, AnOut

Can be set by:
- Toggle manually on pendant
- Set through code by associating it with a target

Always turn off robot when connecting or disconnecting IO

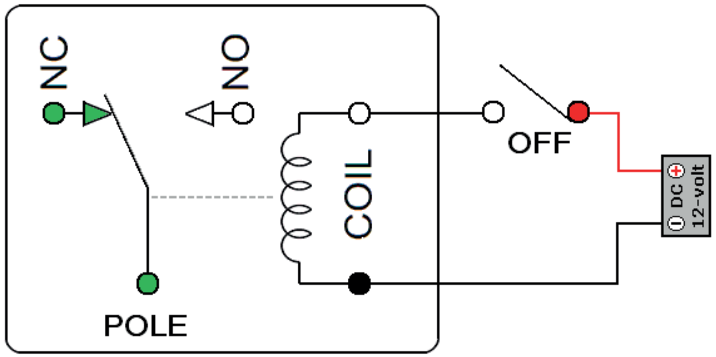
Strictly forbidden to wire any 230V ac

Interfacing with other systems

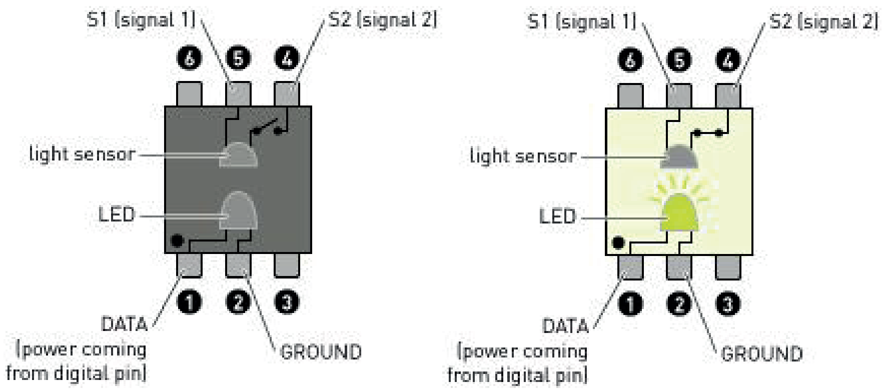
Because robot IO's are designed to interface with industrial equipment they operate on 24V.

If you wish to interface with systems (e.g. Arduino) operating on a lower or higher voltage you will need:

Relay (low switching frequency)

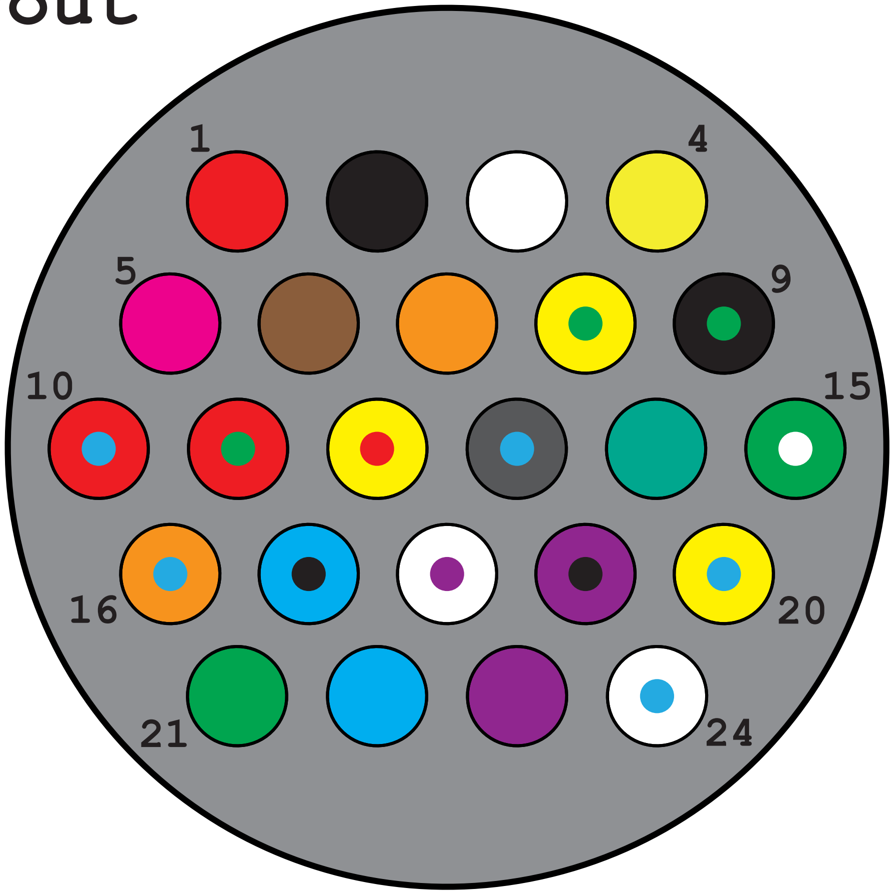


Opto coupler (high switching frequency)



IO plug for industrial robots Gordon street

IO pinLayout

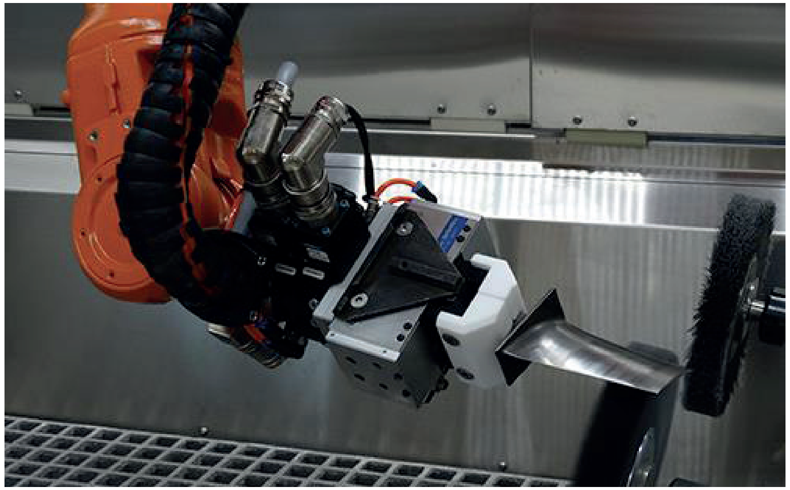


1> power
4> ground
5-9> dIn
10-15> dOut
16-20> dOut
21-22> aIn
23-24> aOut

Plug assembly can be purchased from bMade shop
Depending on the robot not all pins are wired in !!!

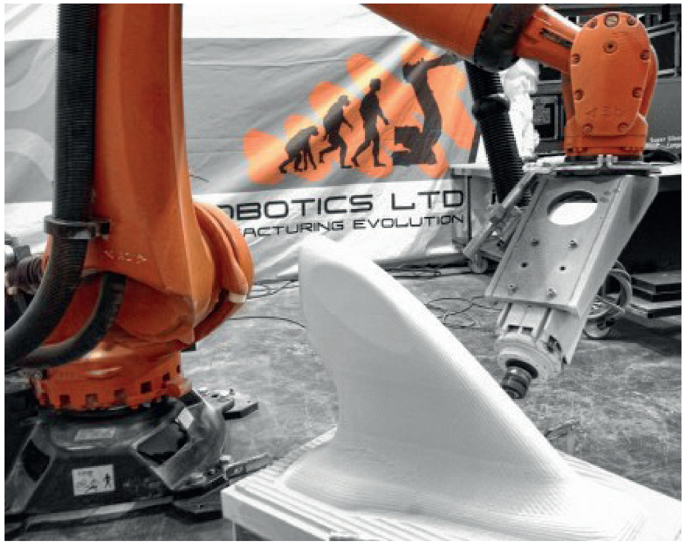
Considerations for end effectors design

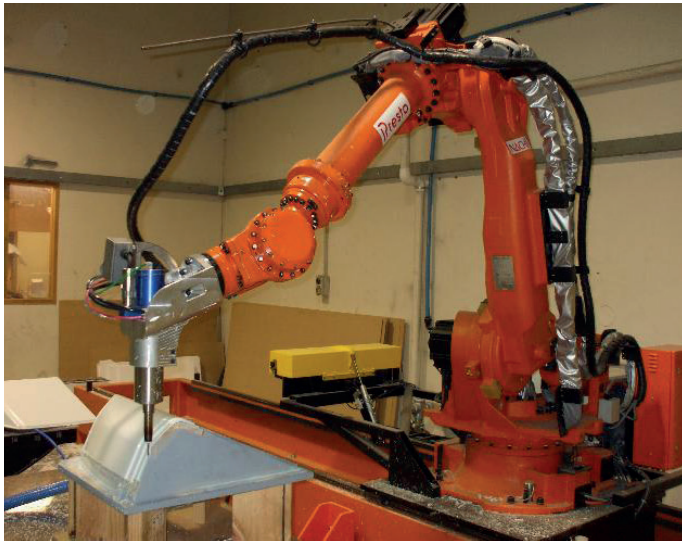
Tool to part
tool or end effector is fixed to robot flange
Part to tool
part is fixed to robot flange or held by gripper



part to tool

- Use soft connection (between flange and tool) where possible (when stiffness is not required), to protect your tool in case of crash.
- End effector orientation depends on tool requirement directional (knife) vs. uni-directional (spindle)
- End effector orientation will change the size of workspace and where singularities occur, it is highly recommended to run some simulations before deciding on a final orientation





45 degrees between flange and TCP to avoid singularities while milling

bMade end effectors

Can be rented trough the bMade shop

- large gripper
- medium gripper
- small gripper
- glue extruder
- solenoid valves
- spindle

Operation

Robot programming

There are multiple ways to program a robot

- Teaching: manually jogging the robot and recording positions, industry standard for maximum repeatability.
- Offline programming: A program is uploaded and executed by robot controller.

~ Offline programming with sensory inputs

- Real-time control: Commands are streamed to the robot controller.

Programming language by manufacturer

KUKA	KRL
ABB	RAPID
STAUBLI	VAL3
UR	URscript

For an in depth look at programming syntax consult official manuals
(Can be found in bmadeNM drive)

KUKA	Operating and programming instructions for system integrators
ABB	Operating manual intro to RAPID
STAUBLI	VAL3 Reference Manual
UR	Script manual

Calibration

4Point calibration for end effector:

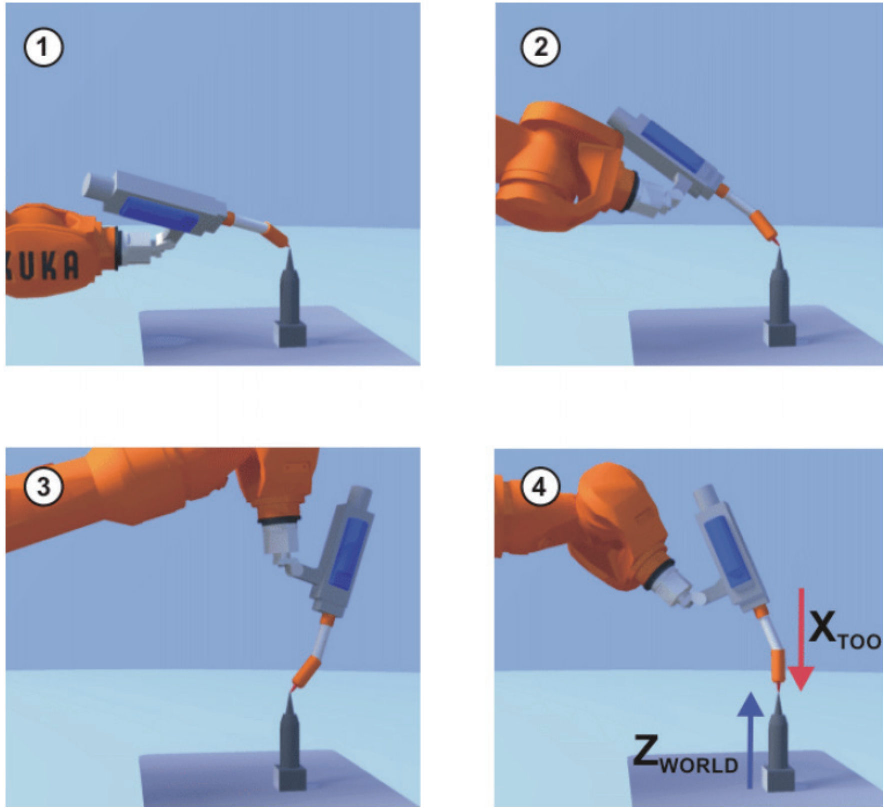
4 points define a unique sphere, the centre of which is the TCP. This calibration will return the TCP position.

4Point, long spike and short spike:

Two 4 point calibrations, one with a short spike and one with a long spike, used when tool axis accuracy is required. This calibration will return the TCP position and orientation.

3Point calibration for base:

3 points define a unique plane.



Universal robot:

Tool: Calibration has to be done through grasshopper

Base: Jog the robot to 3 points to define a plane, use the display to read the positions and input data in grasshopper

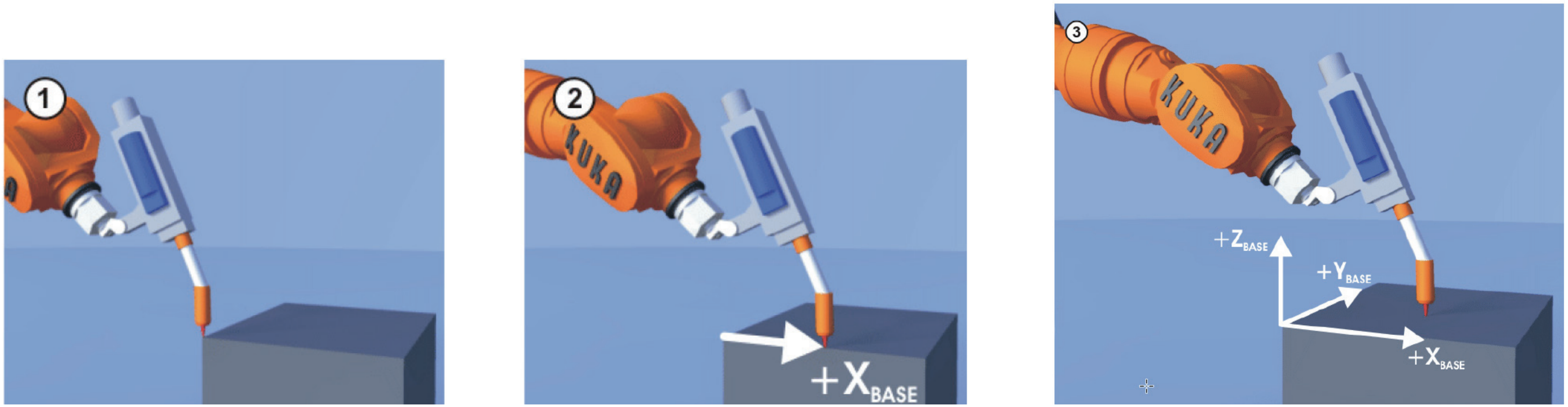
Kuka:

Tool: 4point calibration on pendant or in grasshopper, this returns the position of the TCP.

Base: 3point calibration method either on the pendant or in grasshopper (using the pendant to display the robot position)

ABB:

Idem Kuka



Loading a program onto the robot controller (UR)

Program is send directly from Grasshopper over Ethernet

1) On your computer

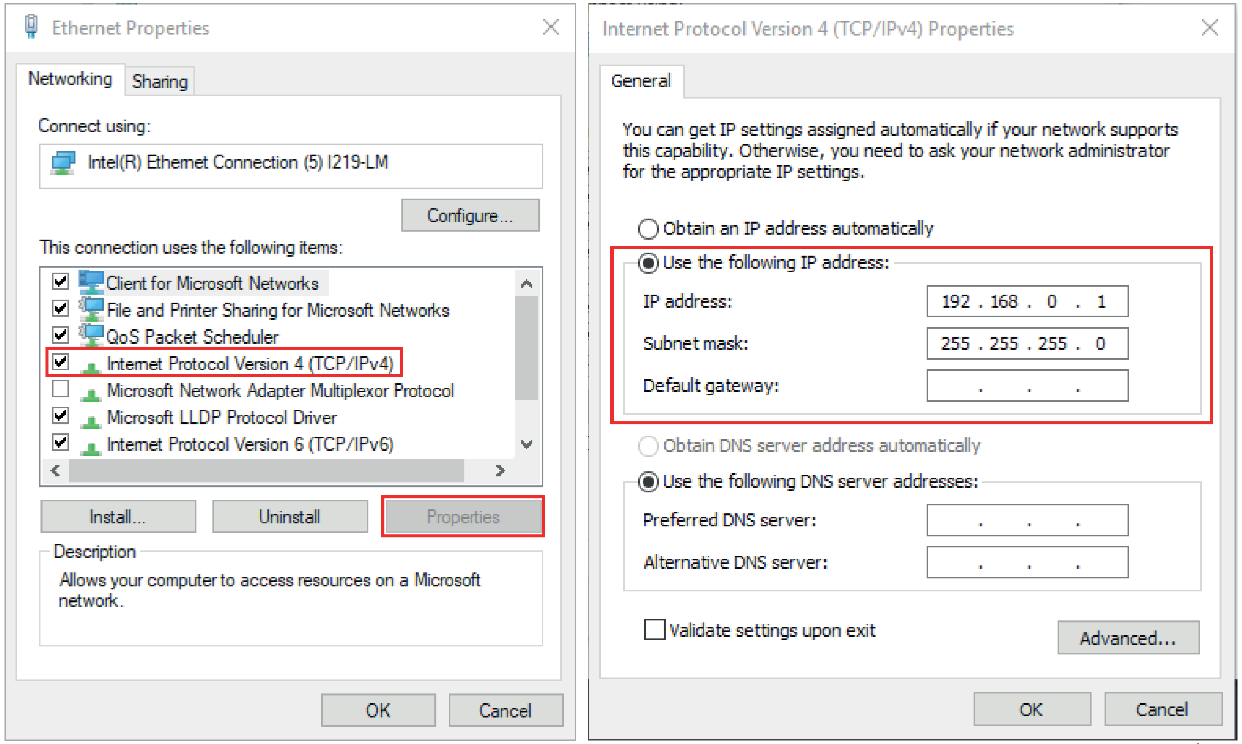
- Navigate to ***control panel*** > ***network connections***
- Right click on ***Ethernet*** and select ***properties***
- In the properties window select ***Internet Protocol Version 4 (TCP/IPv4)*** and click the ***properties*** button
- Set the option ***Use the following IP address***
- Set IP address to 192.168.0.1
- Set Subnet mask 255.255.255.0

2) On the robot

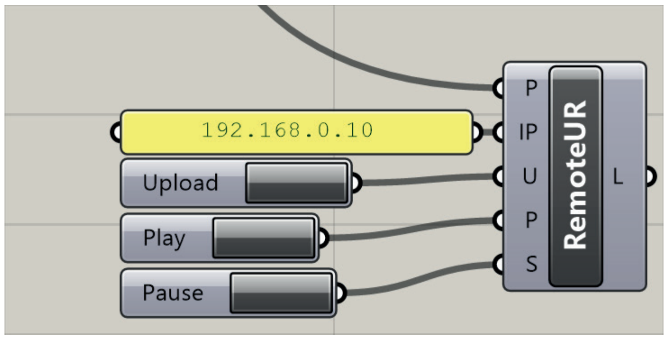
- Navigate to ***Setup robot*** > ***Setup network***
- Set IP to 192.168.0.10
- Set Subnet Mask to 255.255.255.0
- Click ***Apply***

3) In Grasshopper

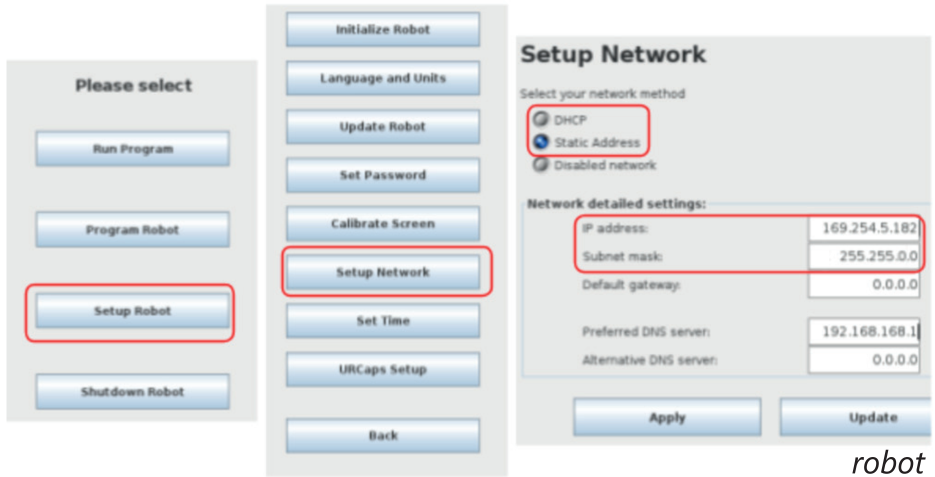
- Get a ***remoteUR*** component
- Set the IP to match the IP you have set in step 2



computer



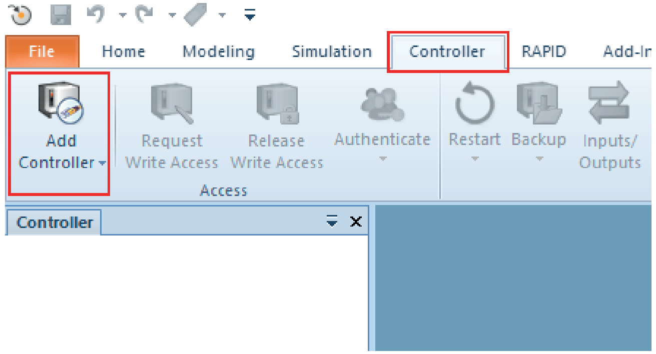
grasshopper



robot

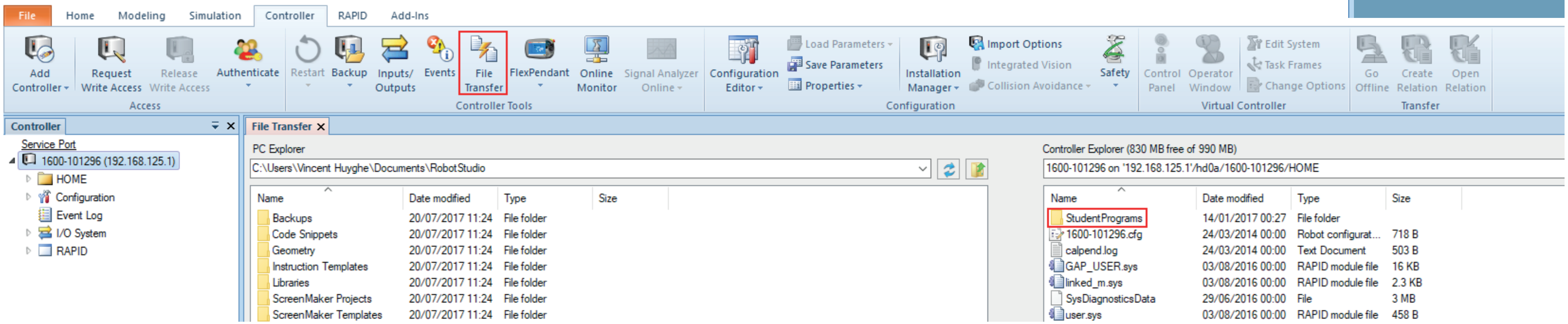
Loading a program onto the robot controller (ABB)

Using Robot Studio to transfer program to controller over Ethernet



1) On your computer:

- Start robot studio and navigate to the controller tab
- Click on ***add controller*** and select ***one click connect***
- Once the controller is recognised in Robot Studio click ***file transfer*** in tool bar
- Drag and drop the folder containing the .pfg and .MOD file to the ***StudentPrograms*** folder on the right side

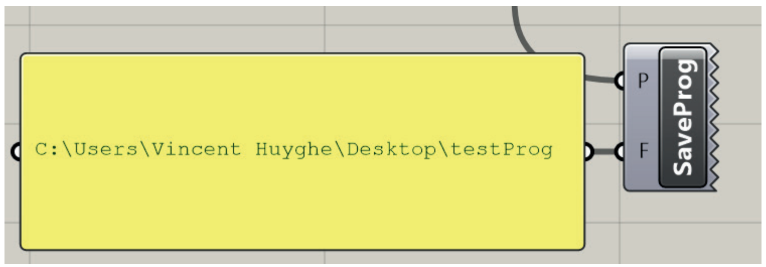


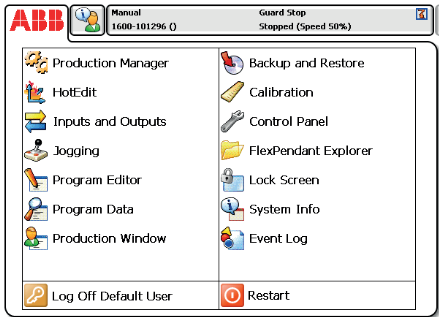
2) In Grasshopper:

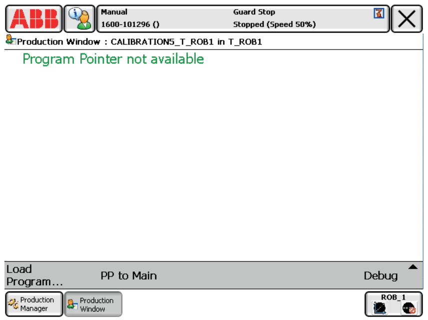
- Get a ***SaveProg*** component
- Specify a file path for saving

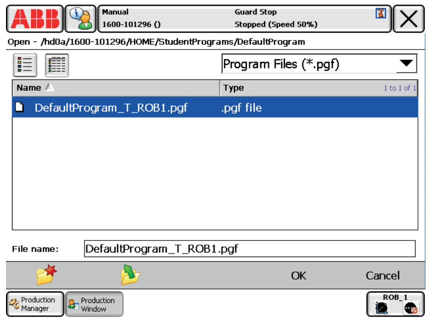
3) On the pendant:

- Press the ***ABB*** logo in the upper left corner
- Navigate to ***Production Window***
- Select ***Load program*** from bottom left corner
- Select your .pfg file



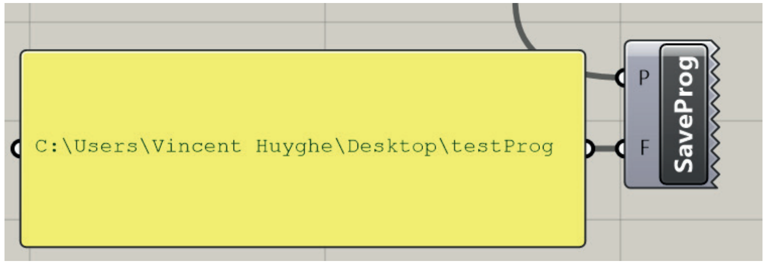






Loading a program onto the robot controller (Kuka)

Using a USB to transfer the program to the controller



1) In Grasshopper:

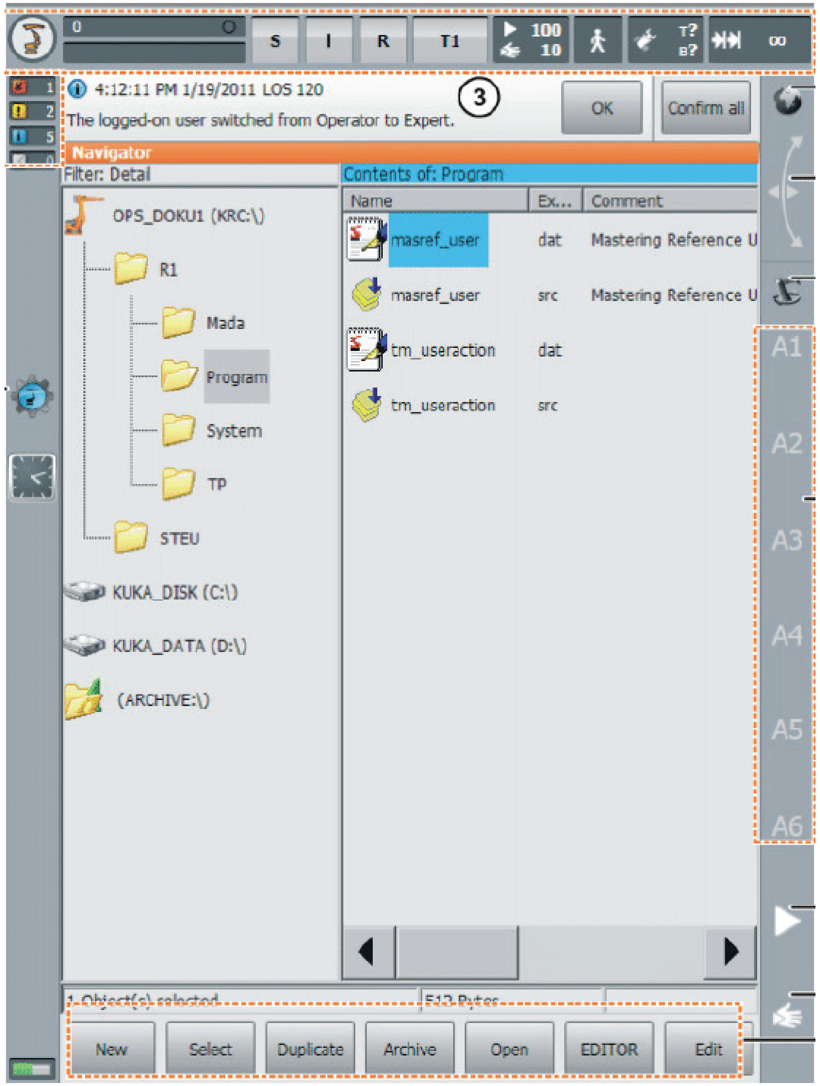
- Get a ***SaveProg*** component
- Specify a file path for saving

2) On your computer:

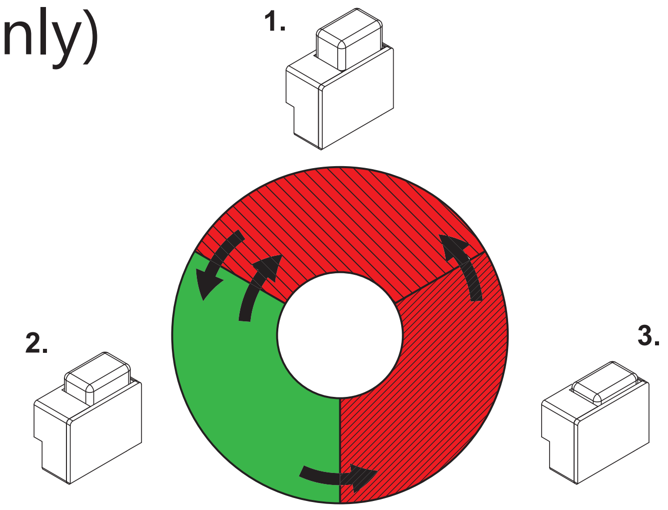
- Copy the 2 .src and .DAT file generated by Grasshopper to a USB stick

3) On the pendant:

- Click the robot symbol in top left corner
- Navigate to ***Configuration*** > ***User Group***
- Click ***Log-on*** in button bar (bottom)
- Select ***Expert*** and enter password “bmade”
- Navigate the tree on left side to your usb drive
- Select the file and click the ***edit*** in the button bar then select ***copy*** from menu
- Navigate to ***R1*** > **Program** and click ***edit*** button then select ***paste*** from menu
- Select the file you pasted and click ***edit*** button and select ***Load without parameters***



Program execution mode (industrial robots only)



Manual mode

- Speed is limited, operator can be inside enclosure
- Manual jogging of robot
- Kuka terminology: T1 mode
- Enabling switch required

Manual full speed

- Kuka terminology: T2 mode
- Enabling switch required
- Prohibited to use

Automatic mode

- Explicit permission from bMade staff is required
- Speed is unlimited
- Nobody is allowed in enclosure

Enabling device or dead man switch

Position 1 – "Standby position":

- Button free, i.e. not pressed in.
- The process can not be run.
- Waiting to be pressed to its midpoint position ("running position").

Position 2 – "Running position":

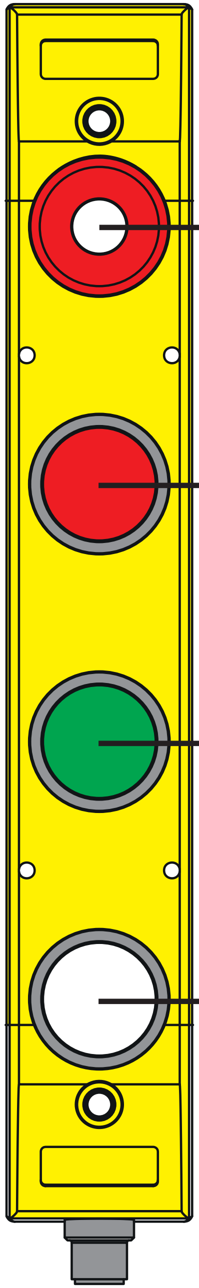
- The push button is pressed to its midpoint position.
- The process can be run.
- The process will stop if the button is released or is pushed to its end position ("stopped").

Position 3 – "Stop position":

- The push button is pressed to its end point.
- The process is stopped by the control device.
- Starting the process requires that the button is fully released ("standby"), and then pressed to midpoint position ("driving mode").

Robots at Bartlett

Gordon Street industrial robot cell



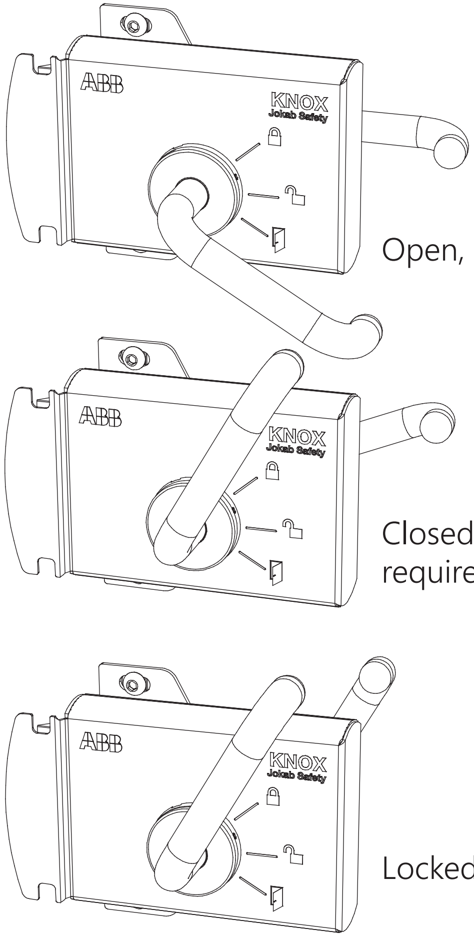


ABB procedure for automatic mode

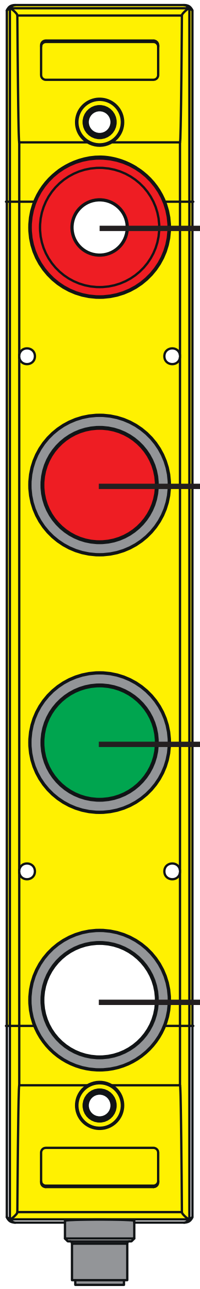
- Turn key on controller to automatic
- Lift door handle to closed position
- Press green reset button
- Press white Motors on

KUKA procedure for automatic mode

- Turn key on pendant to automatic
- Lift door handle to closed position
- Press green reset button
- Press white acknowledge operator safety

Activating automatic mode with somebody in the enclosure will result in automatic ban from the robot room

Gordon Street collaborative robot cell



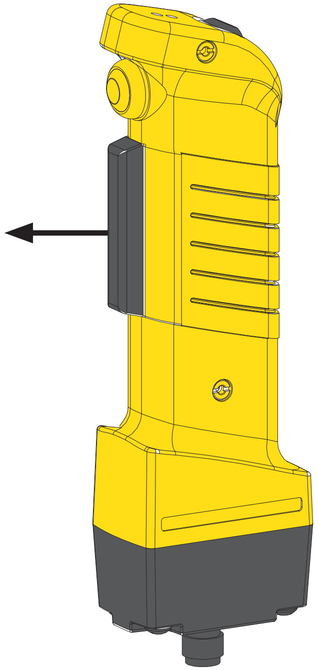
Emergency stop

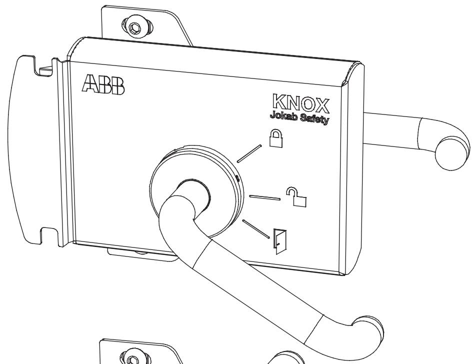
Enabling device/
Dead’s man switch

Request access
(unlock door)

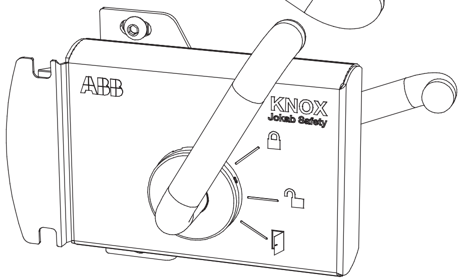
Reset,
blinking when
reset required

Reset enabling
device, blinking
when required

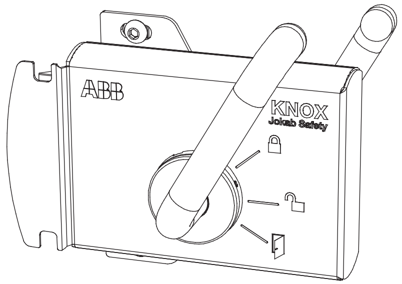




Open, unlocked



Closed, but reset
required to lock

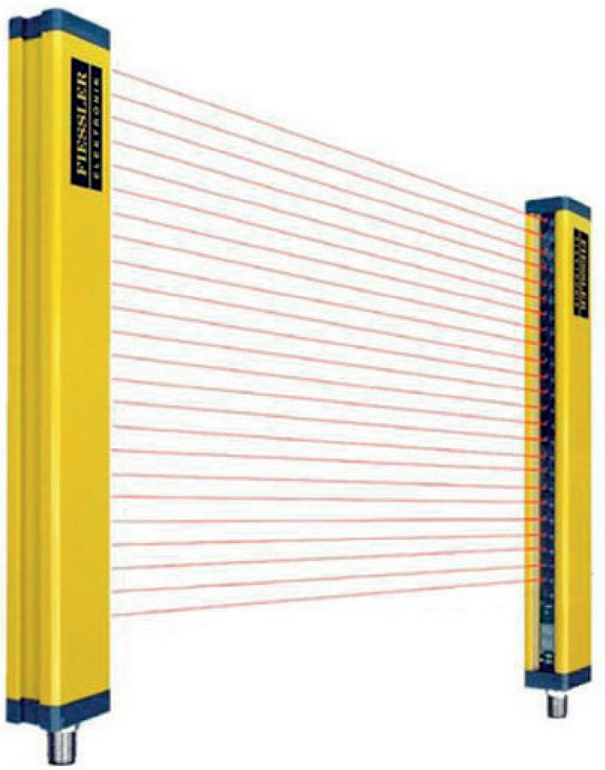


Locked

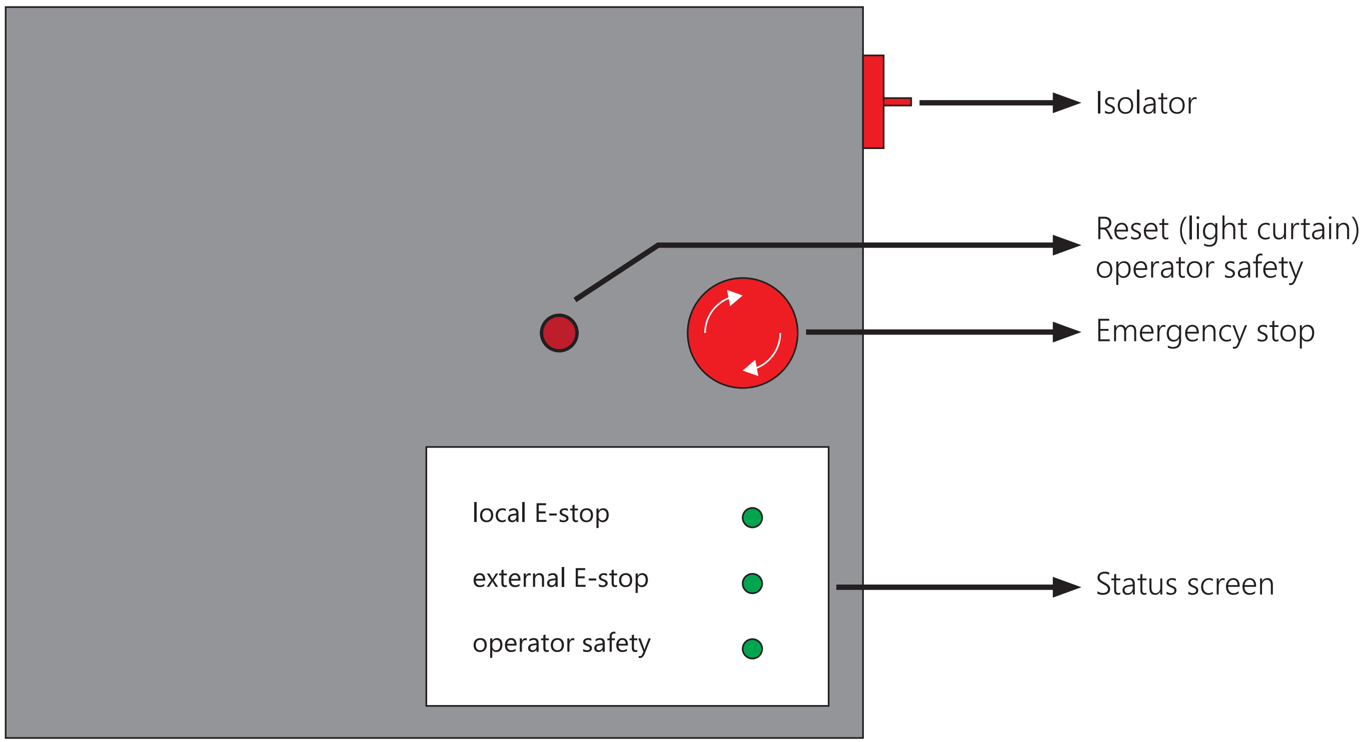
Operating UR10 with enclosure open
- Press and hold enabling device
- Press white button to reset safeguard stop

Operating UR10 with enclosure closed
- Lift door handle to closed position
- Press green reset button

Here East industrial cell



Light curtain



Safety controller (PLC)

Activating automatic mode with somebody in the cell will result in automatic ban from the robot room

Robot Health & Safety

- Robots are dangerous industrial machines any imprudent behaviour will result suspension of your access!
- A signed risk assessment form is required for any robotic project, if any changes are made to the process or participants a new RA form is required. The risk assessment needs to be on clipboard on front of cell at all times
- Robots must always be supervised by someone who has completed the induction for the appropriate robot and has signed the risk assessment form.
- When the robot is moving in manual mode only the operator (person holding the pendant) can be in the enclosure or perimeter.
- Manual full speed or T2 is strictly prohibited.
- It is forbidden to use automatic mode if anybody is inside the enclosure, explicit permission from bMade staff is required before using automatic mode.
- Max 24V 10A for any electronics in end effector.
- No 230Vac wiring allowed
- When leaving the robot room all end effector electronics must be powered down.

When in doubt ask bMade staff

Neglecting any of these rules will result in a suspension of your access to robots!

Robot terms of use

- When turning off a robot wait 2 minutes before restarting the robot !!!
- A bMade lab coat is required before entering the robot room / cell (can be obtained trough the bMade shop in exchange for a 10£ deposit)
- The robot rooms / cell operate on same hours as the workshop meaning 9:30 to 13:00 and 14:00 to 17:00
- In order to book a robot you will need the following:
    - An induction for the appropriate robot
    - A signed robot risk assessment
    - A working grasshopper file
- Robots need to be booked a week in advance, if you wish to cancel your booking please do so asap.
- Robot room booking can be requested trough:
    - Vincent Huyghe, vincent.huyghe@ucl.ac.uk
    - Alex McCann, alex.mccann@ucl.ac.uk
- No food is allowed in the robot room
- Only bring in the robot room what you need for project (no coats and bags)
- When leaving the room at the end of the day you must:
    - If you have the robot booked for the following day
        - Clean the room
    - Otherwise
        - Clean the room
        - Disconnect all IO and end effector
        - Return the robot to the home position
        - Remove everything you brought into the room
- By booking a robot you acknowledge you have read and understand the terms of use!!!

Breaking any of these rules will result in a loss of your deposit!

Robot risk assessment

The document and examples can be found the on bmadeNM drive under health and safety
The risk assessment needs to be approved by bmade staff first and signed by tutor
The risk assessment needs to be signed by any team member wanting to enter the robot room
You will be expected to provide a Material safety Data Sheet where applicable

Robot Risk Assessment Form V6

Health and Safety Document Type	Robot Risk Assessment Form
Document Title	Bmade Robot Risk Assessment Form V6
Issue date and version	October 2018 V6
Risk Assessment Form Review	Annually or sooner if required
Form Author	Vincent Huyghe

Robot Risk Assessment Form

Name of Assessor: ______

Project/Task Title: ______

Unit: ______
Robot: ______

Project Description:

______

Primary groups of people at risk: ______

Other groups of people at risk: ______

Hazards (click inside boxes to select):

- ☐ Substances (including chemicals)
- ☐ Biological materials
- ☐ Manual handling
- ☐ Compressed gases
- ☐ Electrical equipment
- ☐ Process generating sprays or Aerosols
- ☐ Any off-site work
- ☐ Significant noise levels (>75db)
- ☐ Other machinery
- ☐ Grippers
- ☐ Extruders

Robot Risk Assessment Form V6

To find the risk score, decide how severe (S) you think an outcome will be (minor to fatal) and note the score. Then decide the probability (P) of it occurring (from very unlikely to likely) and enter these in the risk assessment drop down boxes. A risk score will then be calculated. The raw (uncontrolled risk) will probably be high, but after you have considered the control measures, the residual risk should fall.

These tables are designed to help you gauge whether your risks are low or high. However, there are many factors that affect one's judgement of risk - for example, you may not have all the information you need to make a realistic assessment. Lack of space, lack of training, working to deadlines and working late and alone could all act as risk increasing factors.
Your assessment of risk will always be subjective, as it depends on not only your knowledge and experience, but also the information you have available. The more information you have, the more accurate the assessment will be. Remember, any effective appropriate control measures will lower risk.

Severity

Category	Examples	Score
Minor	Superficial injuries - cuts, bruises, mild skin irritation, mild aches and pains requiring first aid only. Minor property damage.	1
Serious	More serious injuries or ill-health, requiring time off work or study or a hospital visit, e.g. burns, sprains, strains and short-term musculoskeletal disorders, cuts requiring stitches, back injuries, fractures to fingers or toes. More serious property damage.	2
Major	Broken limbs, amputations, long-term health problems resulting from work, or acute illness requiring medical treatment, loss of consciousness, serious electric shock, loss of sight. Major property damage.	3
Fatal	Injury or ill-health which leads to death either at the time or soon after the incident, or eventually, as in the case of certain occupational diseases, such as asbestos-related cancers.	4

Probability

Category	Examples	Score
Very unlikely	Good control measures are in place. Controls do not rely on a person using them (i.e. personal compliance). Controls are very unlikely to break down. People are very rarely in this area or very rarely engage in this activity.	1
Unlikely	Reasonable control measures are in place but they do rely on a person using them (some room for human error). Controls unlikely to breakdown. People are not often in this area / do not often engage in this activity / this situation is unlikely.	2
Possible	Inadequate controls are in place, or likely to breakdown if not maintained. Controls rely on personal compliance. People are sometimes in this area or sometimes engage in this activity / this situation sometimes arises	3
Likely	Poor or no controls in place. Heavy reliance on personal compliance (lots of room for human error). People are often in this area / engage in this activity on a regular basis / this situation often arises.	4

Low risk:	Score between 1 and 3
Medium risk:	Score between 4 and 6
High risk:	Score between 7 and 8
Very high risk:	Score between 9 and 16

If the risk score is high look into measures to reduce the risk.
If the score is very high the process should not be carried out, review and reassess.

If in doubt seek advice!

Robot Risk Assessment Form V6

Briefly describe each hazard and harm that could happen in the boxes below:	S	P	R	Now describe the control measures	S	P	R

Assessor(s):	Student		
Name:		Signature:	
Name:		Signature:	
Name:		Signature:	
Name:		Signature:	

Approver:	Unit Tutor		
Name:		Signature:	
Date:			

Please pay special attention to:
- You are expected to be able to answer question about your risk assessment, don't just read it make sure you understand !!!!!
- Spelling errors

Robot Risk Assessment Form

Name of Assessor: ______________________________

Project/Task Title: ______________________________

Unit: ____________________
Robot: ____________________

Project Description:

____
____
____
____
____
____

Primary groups of people at risk: ______________________________

Other groups of people at risk: ______________________________

Hazards (click inside boxes to select):

- ☐ Substances (including chemicals)
- ☐ Biological materials
- ☐ Manual handling
- ☐ Compressed gases
- ☐ Electrical equipment
- ☐ Process generating sprays or Aerosols
- ☐ Any off-site work
- ☐ Significant noise levels (>75db)
- ☐ Other machinery
- ☐ Grippers
- ☐ Extruders

Severity

Category	Examples	Score
Minor	Superficial injuries - cuts, bruises, mild skin irritation, mild aches and pains requiring first aid only. Minor property damage.	1
Serious	More serious injuries or ill-health, requiring time off work or study or a hospital visit, e.g. burns, sprains, strains and short-term musculoskeletal disorders, cuts requiring stitches, back injuries, fractures to fingers or toes. More serious property damage.	2
Major	Broken limbs, amputations, long-term health problems resulting from work, or acute illness requiring medical treatment, loss of consciousness, serious electric shock, loss of sight. Major property damage.	3
Fatal	Injury or ill-health which leads to death either at the time or soon after the incident, or eventually, as in the case of certain occupational diseases, such as asbestos-related cancers.	4

Probability

Category	Examples	Score
Very unlikely	Good control measures are in place. Controls do not rely on a person using them (i.e. personal compliance). Controls are very unlikely to break down. People are very rarely in this area or very rarely engage in this activity.	1
Unlikely	Reasonable control measures are in place but they do rely on a person using them (some room for human error). Controls unlikely to breakdown. People are not often in this area / do not often engage in this activity / this situation is unlikely.	2
Possible	Inadequate controls are in place, or likely to breakdown if not maintained. Controls rely on personal compliance. People are sometimes in this area or sometimes engage in this activity / this situation sometimes arises	3
Likely	Poor or no controls in place. Heavy reliance on personal compliance (lots of room for human error). People are often in this area / engage in this activity on a regular basis / this situation often arises.	4

Low risk:	Score between 1 and 3
Medium risk:	Score between 4 and 6
High risk:	Score between 7 and 8
Very high risk:	Score between 9 and 16

Briefly describe each hazard and harm that could happen in the boxes below:	S	P	R	Now describe the control measures	S	P	R

Robots software

RoboDK
- post processing milling toolpaths for complex geometry's from Fusion360 or Powermill.
- Plasma cutting
- Easy wizard but not very flexible

Grasshopper + Robots plugin
- Custom applications
- Good way to learn robotics
- Super flexible

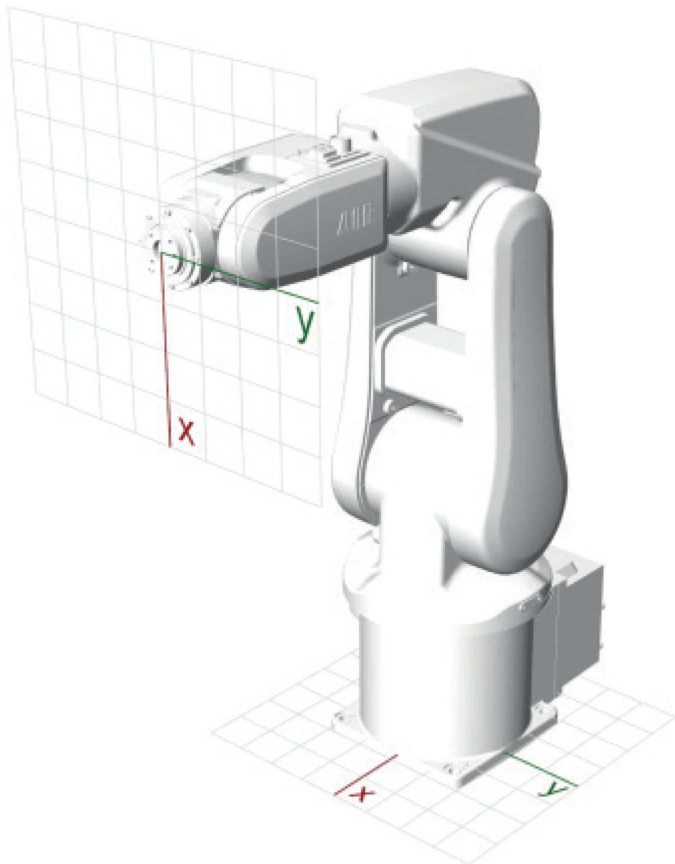
Robots Plugin for Grasshopper

Download from following link

https://github.com/visose/Robots

See wiki for instructions on installing

Plugin convections



As with Rhino, the plugin uses a right-handed coordinate system. The main coordinate systems are:

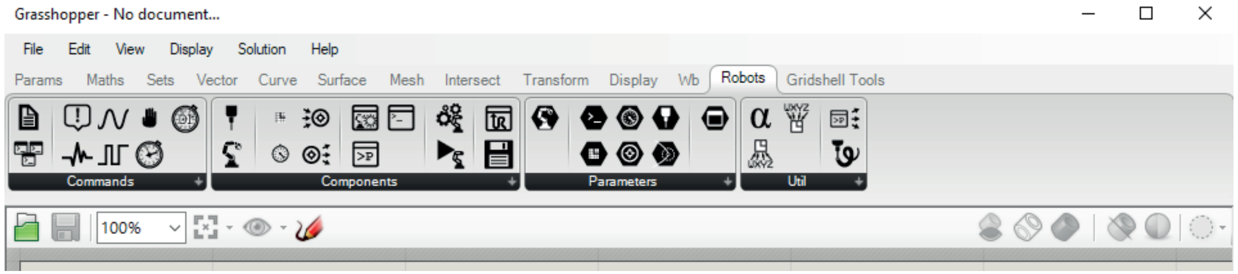
World coordinate system: It's the Rhino document's coordinate system. Cartesian robot targets are defined in this system. They're transformed to the robot coordinate system during post-processing.

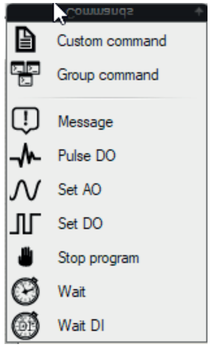
Robot coordinate system: Used to position the robot in reference to the world coordinate system. By default robots are place in the world XY plane. The X axis points away from the front of the robot, the Z axis points vertically.

Tool coordinate system: Used to define the position and orientation of the TCP relative to the flange. The Z axis points away from the flange (normal to the flange), the X axis points downwards.

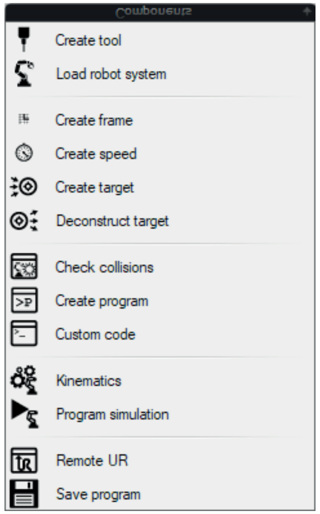
Robots Plugin

Toolbar

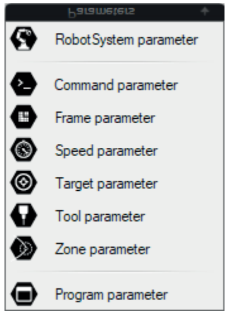




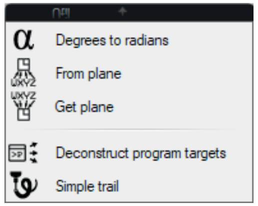
IO and commands



All the components for creating and processing toolpaths

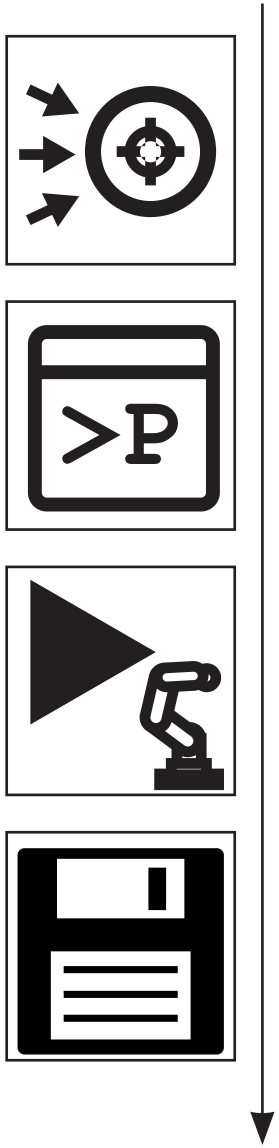


Containers for parameters, can also be used for casting parameters



Utility components

Workflow



Create robot targets and assign parameters

Create robot program

Validate robot program

Save program and upload to controller

Robots components

Load robot: select what robot you wish to use

Create tool: define an end effector

Create speed: define a speed

Construct target: define robot targets

Deconstruct target: extract parametrs from a robot target

Check collision: check toolpath for collisions

Create program: run kinematics and post process multiple targets

Custom command: This components allows you to use commands not implemented in the plugin

Kinematics: run kinematics for a single target

Program simulation: create a visual simulation of your tool path

Save program: save program to file

Remote UR: send program to controller

Set DO: set digital output

Set AO: set analog output

Group commands: group multiple commands into one, this useful for working with the Grasshopper data structure

Stop: stop program until operator restarted by operator

Create frame: create a base coordinate frame

Get plane: returns a plane in Rhino based on position and Euler-, Axis or Quaternion angles, useful when calibrating

From plane: returns coordinates and angles from plane

Right click on these components to expose more parameters