

iRobotCAM User Manual 1.2

Intelligent Computer Aided Manufacturing for Robotics

Table of Contents

Chapter 1: iRobotCAM Installation Guide	1
1.1 Prerequisites	1
1.2 Installation Trial+Activation Version	1
Chapter 2: Product Key Features	
2.1 Robot production line design2.2 Mechatronics Integration Design with Physical Simulation	
2.3 Robot off-line programming	9
2.4 Robot virtual commissioning	10
Chapter 3: iRobotCAM User Interface	
3.0 New project file	
3.2 Mechatronic	
3.3 Gantt chart	
3.4 Assembly relationship	
3.5 Teach programming	
3.6 Collision detection	
3.7 Programming	
3.8 URDF Import and Export	
3.9 Workspace	24
3.10 Euler angle	24
3.11 Connect module	25
3.12 Simulation verification	26
3.13 Laser cutting	26
3.14 Welding	27
3.15 Screen recording function	
Chapter 4: Notes and Save	
Chapter 5: Define Basic Mechatronic Objects Chapter 6: Define Conveyor Belts	
Chapter 7: Define Motion Array Units	
Chapter 8: Create Robot	64
Chapter 9: Teach Program Draw Arc	
Chapter 10: Robot Pick And Place Chapter 11: Collision Detection	
Chapter 12: Configure Communication	
Chapter 13: Turntable Model	138
Chapter 14: Laser Cutting	
Chapter 15: Welding	

Chapter 1: iRobotCAM Installation Guide

This guide will walk you through the process of installing iRobotCAM on your Windows operating system.

1.1 Prerequisites

- Computer Windows specifications: Supports Win10 64bit or higher versions.
- ZW3D version requirements: The minimum support is ZW3D 2023X educational version, and ZW3D 2024 version has no requirements.
- This product is developed at the expert level of ZW3D software.

1.2 Installation Trial+Activation Version

Precautions before installation:

Here are the precautions you need to take before installing the software package:

1. Trial period:

- The software package has a 30-day trial period.
- After 30 days, you will need activation authorization to continue using it.

2. ZW3D installation:

• Make sure you have ZW3D software installed on your system (download from the official website).

3. Before installing the iROBOTCAM software package (English version), complete the following step:

• Close the ZW3D software.

4. Installing the software package:

- Double-click the software package to begin the installation process.
- Follow the on-screen instructions to complete the installation.

Additional notes:

• Ensure a stable Internet connection when activating authorization

• If you encounter any problems during installation, please refer to the software documentation or contact the software vendor.

Step-by-step instructions on IROBOTCAM plugin for ZW3D 2024 installation.

Step 1:

W

- Double-click the installation package file "IROBOTCAM-V1.2-plugins_en-for-ZW3D2024.exe".
- Click "Next" to proceed. (Figure 1-1)

Setup - IROBOTCAM	_	
	Velcome to the IROBOTCAL Setup V This will install IROBOTCAM on your computer. It is recommended that you close all other applicatio continuing. Click Next to continue, or Cancel to exit Setup.	
	Next >	Cancel

Figure 1–1: Setup

Step 2:

- Read the license information carefully.
- If you agree with the terms, click "I accept".
- Fill in your personal information and click "Next". (Figure 1-2)

Setup - IROBOTCAM -		B Setup - IROBOTCAM	– 🗆 X
License Agreement Please read the following important information before continuing.		User Information Please enter your information.	
Please read the following License Agreement. You must accept the terms of this agreement before continuing with the installation.		User Name: Supperman	
Software End User License Agreement	^	Organization:	
IMPORTANT: READ CAREFULLY THIS END USER LICENSE AGREEMENT (AGREEMENT) IS A LEGAL AGREEMENT BETWEEN YOU (AN INDIVIDUAL OR ENTITY) AND NANJING YUEQING INFORMATION TECHNOLOGY CO., LTD. (HEREINAFTER REFERRED TO AS YUEQING TECHNOLOGY) REGARDING SOFTWARE PRODUCTS. THE	~	nanjing/oeying	
< Back I accept	Cancel	< Back Ne	ext > Cancel

Figure 1–2: License and User Information

Step 3 (Figure 1-3):

• Select "installation path".



- Click the "Browse" button to change the installation directory if needed.
- Click "Next".
- Select start menu folder.

ect Destination Location		Betup - IROBOTCAM	- 0
Where should IROBOTCAM be installed?		Where should Setup place the program's shortcuts?	
Setup will install IROBOTCAM into the following fold		Setup will create the program's shortcuts in the follo	wing Start Menu folder.
To continue, click Next. If you would like to select a differe		To continue, click Next. If you would like to select a different	t folder, click Browse.
D:\Program Files\JROBOTCAM	Browse	IROBOTCAM	Browse
14 louis 708 A 107 of feas did course is sounded			
At least 708.0 MB of free disk space is required.			

Figure 1–3: Installation and Start menu folder

Step 4(Figure 1-4):

- A summary of the installation information will be displayed.
- You can click "Back" to make changes or "Install" to proceed.

	in installing IROBOTCAM		() ()
ick Install to continue with nange any settings.	h the installation, or clic	k Back if you want	to review or
Jser information: 896167864@qq.com			^
Destination location: D:\Program Files\IROB	OTCAM		
tart Menu folder: IROBOTCAM			
			~

Figure 1–4: Installation information

Step 5:

• Click "Finish" to complete the installation process. (Figure 1-5)

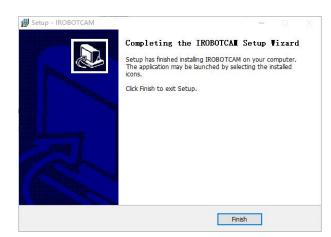


Figure 1–5: Complete installation

Step 6:

- When you first run ZW3D after installation, some default settings will be displayed.
- Click "OK" to acknowledge them. (Figure 1-6)

Millimeters	O Inches
Default user role	
Expert	

Figure 1–6: Default setting

Step 7:

- Open the ZW3D interface and click on the "IROBOTCAM" module.
- Click "Activation" to activate the software package.(Figure 1-7)

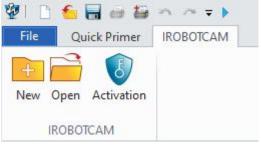


Figure 1–7: Activeation

Step 8:

- The first installation of the plugin provides a 30-day trial period.
- After the trial expires, you will need to activate the plugin using a valid activation code.
- In the "License" window, under the "IROBOTCAM" column, you will see your

¥

current status (trial or inactive).

- To activate the plugin, copy or enter the activation code you received from Yue Qing robot in the "Key ID" field.
- Click "Apply".
- The "IROBOTCAM" status will update to "activated" if the code is valid. (Figure 1-8)

License		
Activate		
Key ID		Apply
Trial license		
Directory		Load
Product infomation		
CAD status	Activated	

Figure 1–8: Activate Authorization

Step 9:

• You have successfully installed and activated the IROBOTCAM plugin for ZW3D 2024.

Note:

- The "-" symbol in the activation code is important and should not be omitted.
- If you encounter any problems during installation or activation, please refer to the software documentation or contact the software vendor.

Chapter 2: Product Key Features

iRobotCAM is dedicated to creating an open and comprehensive platform for digital solutions related to robotics, focusing on various aspects of production line mechatronic. This includes:

Design concept :

- Assists in the design of robot-based production lines.
- Provides tools for mechatronic concept design.

Robot off-line Programming & Simulation:

- Enables users to program and simulate robot machining processes.
- Offers advanced features for virtual commissioning , ensuring accurate and efficient program development.

virtual commissioning & Real-world Synchronization:

- Utilizes a reliable physical simulation engine for realistic virtual testing.
- Achieves seamless debugging by synchronizing virtual and actual robot movement.

Process Integration & Customization:

- Features convenient modules for integrating various industrial processes.
- Allows for the creation of customized process modules to suit specific needs.

Industrial Software Platform:

- Leverages the powerful ZW3D platform for robust and reliable operation.
- Offers dedicated solutions for robot welding, robot spraying, and so on.

Continuous Improvement:

• iRobotCAM is constantly evolving to enhance the capabilities and application value of industrial robots.

Benefits:

- Streamlined production line design and development.
- Improved accuracy and efficiency of robot programming.
- Reduced risk of errors and rework.
- Enhanced flexibility and customization options.

Target User:

- Manufacturers and engineers involved in robot-based production.
- Companies seeking to optimize their robotic processes.
- Developers of industrial automation solutions.

2.1 Robot production line design

1. Robot workstation unit equipment library, importing robots, workpieces, tools, additional axes (guide rails, displacement machines, gantry frames, etc.) from different manufacturers, as well as other related work units (floors, safety barriers, control cabinets, etc.).

2. Virtual sensor modeling, cylinder component modeling, claw component modeling, conveyor belt component modeling.

3. Work unit layout, precise control of the position layout and pose relationship of the equipment imported into the work unit.

Robot Workstation Unit Equipment Library:

- **Comprehensive Scope:** The library should include a wide range of equipment from various manufacturers, including robots, workpieces, tool hands, additional axes (guide rails, displacement machines, gantry frames), floors, safety barriers, control cabinets, and other relevant components.
- **Standardized Representation:** Equipment should be represented in a consistent and standardized format, including 3D models, kinematic parameters, dynamic properties, and simulation sequence.
- Interoperability: Equipment models could compatible with different simulation platforms and software tools to facilitate seamless integration into existing workflows.
- **Parametrization:** Equipment models could parametrically defined, allowing for customization and adaptation to specific application requirements.
- **Search and Filtering:** Efficient search and filtering functionalities could implemented to allow users to quickly find and access the equipment they need.

Virtual Sensor Modeling:

- **Sensor Types:** The library including various sensor types commonly used in robot workstations, such as force sensors, and proximity sensors.
- **Customization:** Sensor models could customize the parameters to match the specific properties
- Integration with Other Models: Sensor models could seamlessly integrated with other equipment models within the robot workstation, allowing for accurate simulation of sensor-based interactions.

7

Benefits:

- **Reduced Design Time:** Access to a comprehensive library of pre-built equipment models can significantly reduce the time and effort required for robot workstation design.
- **Improved Design Quality:** Virtual sensor modeling allows for early identification and mitigation of potential issues, leading to more robust and reliable robot systems.
- Enhanced Collaboration: The library can facilitate collaboration between engineers, designers, and other stakeholders involved in robot workstation development.
- **Cost Optimization:** By reducing design time and improving design quality, the library can significantly contribute to cost optimization in robot workstation projects.

2.2 Mechatronics Integration Design with Physical Simulation

Mechatronics integration design, by defining geometric bodies as rigid bodies and collision bodies, realistically restoring the physical properties of geometric bodies such as mass, inertia, friction, material, and collision, creating motion pairs and constraints, sensors and actuators, simulation sequences, and signal adapters, thereby achieving physical simulation of the model.

Key Aspects of Physical Simulation for Mechatronics Integration Design:

- Geometric Body Representation:
 - Rigid Bodies: Each physical component is modeled as rigid body and collision body, accurately capturing its mass, inertia, and collision properties.
 - Material Properties: Realistic material properties, including friction coefficients, are assigned to each body to simulate their interactions and behaviors.
- Motion Pairs and Constraints:
 - Motion Pairs: Define the permitted relative motion between two rigid bodies, encompassing joints, gears, and other kinematic connections.
 - Constraints: Restrict the motion of bodies in specific ways, limiting their movement to prescribed paths or preventing unwanted rotations.
- Sensors and Actuators:
 - Sensors: Simulated devices that gather information about the system's state, including position, velocity, and force, providing critical feedback for control algorithms.
 - Actuators: Virtual representation of robot, tool, positioner, cylinder, or other mechanisms that apply forces or torques to manipulate the motion of the system's components.
- Simulation Sequences:
 - ♦ Defining the sequence of events that occur during the simulation. This includes specifying initial conditions, control inputs, and IO interactions.
- Signal Adapters:

♦ Facilitate data exchange between different components by converting signals between compatible formats, such as translating analog sensor data to digital values for software processing.

Benefits of Physical Simulation:

- **Realistic Motion Prediction:** Accurately reflects the physical properties and interactions of the components, providing valuable insights into the system's behavior under various operating conditions.
- Early Design Verification: Potential design flaws and unforeseen interactions are identified early in the development process, saving time and resources by avoiding costly physical prototypes.
- **Performance Optimization:** Allows for iterative design refinement and parameter adjustments to improve system performance, efficiency, and robustness.
- **Reduced Development Costs:** Physical simulation can significantly reduce the need for expensive and time-consuming physical prototypes, streamlining the development process.

2.3 Robot off-line programming

Robot teaching, performing robot teaching operations in joint space.Cartesian space in base coordinate system, tool coordinate system, etc.Robot interpolation algorithms, including several basic interpolation algorithms such as lines, arcs, and joints.Robot job simulation run, actual animation demonstration process.

1. Robot Teaching:

- **Joint space teaching:** Define robot movements by specifying the desired joint angles at each point in the program.
- **Cartesian space teaching:** Teach robot movements directly by guiding the end-effector through the desired path using a virtual teach pendant or other input device.
- **Coordinate system selection:** Specify the coordinate system for teaching, such as base coordinate system, tool coordinate system, or user-defined coordinate systems.

2. Robot Interpolation Algorithms:

- Linear interpolation: Generate straight-line motions between programmed points.
- **Circular interpolation:** Generate smooth, circular motions between programmed points.
- **Joint interpolation:** Generate coordinated movements of all robot joints to reach the desired end-effector positions.

3. Robot Operation Simulation and Animation:

- **Simulate the robot's movement:** Utilize the programmed path and interpolation algorithms to visually simulate the robot's motion and interaction with its environment.
- Animation demonstrations: Create realistic animations of the programmed movements to demonstrate the robot's performance and identify potential collisions or issues.

Additional Features:

- **Collision detection:** Detect potential collisions between the robot and its environment during simulation, allowing for early identification and correction of potential problems.
- **Program editing and optimization:** Edit and refine the programmed path to improve efficiency, accuracy, and smoothness of robot movements.
- **Program generation:** Generate robot program code based on the simulated program, supporting various robot controller languages.
- Integration with CAD/CAM software: Import and export 3D models of robots and workcells from CAD/CAM software for enhanced simulation realism.

Benefits of Robot Off-line Programming:

- **Reduced Programming Time:** Develop and test robot programs without needing a physical robot, significantly reducing programming time and effort.
- **Improved Program Accuracy:** Simulate the program beforehand to identify and correct errors before deployment, leading to more accurate and reliable programs.
- Enhanced Efficiency: Streamline the robot programming process and optimize robot movements for better performance and efficiency.
- **Reduced Downtime:** Avoid production disruptions by testing and debugging programs offline without impacting real-world operations.
- **Improved Safety:** Identify potential safety hazards through virtual simulation before the robot operates in the real world.

2.4 Robot virtual commissioning

Equipped with virtual commissioning and monitoring, capable of synchronizing the actions of virtual robot system workstations with actual robot system workstations; Supports multi machine IO communication simulation, multi robot synchronization, and multi axis robot linkage planning.

1. Synchronization with Real Robot Systems:

• Connect the virtual robot system and the actual robot system, enabling synchronized execution of the program in both environments.

2. Multi-Machine I/O Communication Simulation:

- Simulate the communication between multiple machines involved in the robot system, including controllers, sensors, and actuators.
- Test and debug the program's interaction with various I/O devices and ensure proper data exchange and communication protocols.

3. Multi-Robot Synchronization:

- Simulate the coordinated movement of multiple robots operating in the same environment.
- Analyze potential collisions, coordination issues, and optimize robot paths for smooth and efficient collaboration.

4. Multi-Axis Robot Linkage Planning:

- Simulate the synchronized motion of robots with multiple axes, such as articulated arms or gantry robots.
- Verify the program's ability to control complex robot kinematics and ensure accurate and coordinated movements.

Benefits of Robot Virtual Debugging:

- Early Identification of Errors: Identify and correct program errors in real-time during simulation, significantly reducing debugging time and effort.
- Enhanced Program Robustness: Test the program under various simulated conditions and scenarios to ensure its robustness and adaptability in real-world situations.
- **Improved System Integration:** Verify the interaction and communication between robots and other machines within the system, facilitating smoother integration and deployment.
- **Reduced Downtime:** Eliminate the need for repeated testing and debugging on the physical robot, minimizing downtime and production disruptions.
- Enhanced Safety: Identify and mitigate potential safety hazards in the virtual environment before the robot operates in the real world, ensuring a safe working environment.

Chapter 3: iRobotCAM User Interface

The start interface allows users to create new projects or open existing ones. Additionally, it provides options to activate iRobotCAM. (Figure 3-1)



Figure 3–1: iRobotCAM User Interface

3.0 New project file

W

Under the "IROBOTCAM" module, click "New" to "Create a new project file". (Figure 3-2)

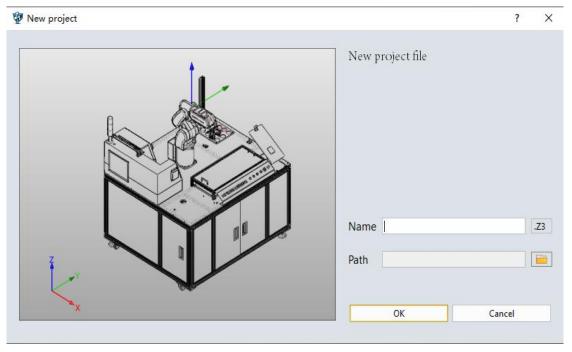


Figure 3–2: New Project

3.1 Component Library

On the "IROBOTCAM" toolbar, in the leftmost first position. (Figure 3-3)

210	6 🔒	ei 15	n 0	() =	•												ZW3D 2024 SF	x64 - [* test	Z3]			
File	Shape	Free	orm	Wirefram	ne Direct	Edit Asse	mbly She	et Metal	FTI W	eldments	Point Clou	d Data	Exchange	Heal	PMI 1	ools Visu	slize Inqui	e Electro	de IROBOTCAM	Арр	Mold	Simulatio
Model library	Import			Rave	Simulation	Collision detection	Mechatroni					Controller			Workspac	e Calibratio	n Laser cutting	Melding	About Help			
		File				Simula	tion		Mech	anical	Electrical			Robo	t		Process	Planning	Help			

Figure 3–3: Component library location

The Model library offers a vast selection of pre-built robot models, categorized for ease of access. Users can browse through the library, and readily incorporate desired components into their projects. (Figure 3-4)

ABB			Carl mar		
FANUC	a farmer		-	and the second s	
GSK				-	
KUKA			5	e	
отс	IRB1600-X_1	IRB_120	IRB_1300-10_1	IRB_2600-12_1	
PANASONIC					
UR			7		
YASKAWA					
		3			
	IRB_2600ID-15_1	IRB_4600-20_2	IRB_5710-110_2	IRB_6620-150_2	
				Select Clos	

Figure 3–4: Model library

3.2 Mechatronic

This section constitutes the heart of iRobotCAM, encompassing a diverse set of tools for building realistic and functional robot models. (Figure 3-5)

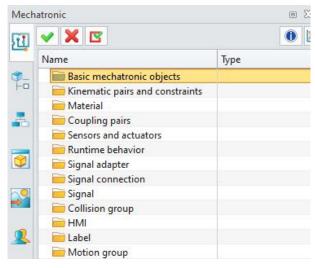


Figure 3–5: Basic Mechatronic Objects

- Rigid body: It can make geometric objects move under the control of physical systems, and any geometric object can only be affected by gravity or other forces if a rigid body component is added.
- Physical property: make the rigid body have a mass effect and respond to forces. Collision response: Collision with the environment or other objects during task processing. Support perception: Objects support being perceived. (Figure 3-6)

😨 Rigid bo	ody			?)	×
Rigid boo	ły				
				-	F
▼ Mass					
Material	Default		•		
Property	settings				
Collision	property response sensing				
▼ Label					
None					•
-					

Figure 3-6: Rigid Body

3

- Collision: Collision is a type of physical component that can only trigger collisions when added to geometric objects along with rigid bodies. It defines the collision mode between components and other colliding components. In physical simulation, rigid bodies without colliding bodies pass through each other.
- Collision shape: Convex hull calculation speed is fast, can envelop, and accuracy may be slightly poor. Convex decomposition is decomposed into many small envelopes to fit the model, and the time for convex decomposition is slow but very accurate.
- Collision material: defines the basic characteristics of the material in collision.(Figure 3-7)

😨 Collision				
Collision	Object			
				+
▼ Shape				
Collision sh	ape	Convex decompositio	n	•
Material	Default		(*)	
▼ Property s	settings			
Physical	property			
Collision	n response			
Support	sensing			
▼ Label				
None				
None				

Figure 3-7: Collision

- Revolute joint: Establish a motion pair between two objects, allowing a degree of freedom to rotate around a certain axis.
- Parent node: Select another rigid body connected to the connector.
- Child node: Select the rigid body that needs to add hinge constraints.
- Specify axis vector: Specify rotation axis, starting angle: The angle of the connector relative to the basic component before the simulation starts.(Figure 3-8)

4

V

😨 Revolute joir	nt			?	×
 Connection reli 	ation				
Select parent node	2				
					+
Select child node					
					+
					X
Axis and angle					ef-
Specify axis vecto	or X:	Υ:	Z:		1
Start angle	0				•
Limitations				5. 	
Upper limit					•
Lower limit					•
▼ Settings	<u></u>				
Motion type		Dyna	mics		•
▼ Name					
1					
-		10	or .		
			ОК Ар	ply (Cancel

Figure 3-8: Revolute joint

- Prismatic joint: A joint consisting of two components that can only move relative to each other in a certain direction, and has one degree of freedom for translation.
- Offset: The distance between the connector and the basic component before the simulation starts.(Figure 3-9)

SV.

💯 Prismatic joi	nt				?	×
 Connection reli 	ation					
Select parent node	2					
						+
Select child node						
						+
						X
Axis and offset						
		Y:		Z:		t
Specify axis vecto		٨:		2:	_	14
Offset	0					mm
Limitations						
Upper limit						mm
Lower limit						mm
▼ Settings						
Motion type		1	Dynamics			•
▼ Name						
		1	01	1		
			OK	Apply		Cancel

Figure 3-9: Prismatic joint

• Materials: The input friction coefficient and recovery coefficient of collision materials enable the object to have frictional force, which can more vividly simulate the simulation form of the object. (Figure 3-10)

🕎 Material			7	? X
 Property 				
Frictional coefficient Restitution coefficient Linear damping Angular damping Density	500			
▼ Name				
		ОК	Apply	Cancel

Figure 3-10: Materia

()

• Sensor: Use collision sensors to collect collision events. The size of the collision shape, length, height, and width depend on the type of collision shape. (Figure 3-11)

🐲 Sensor					?	×
🛡 Туре						
💼 Distance s	ensor					•
Property						
Collision shap	e		Straight I	ine		•
Specify coordi	nate X:	۷:		Z:		1.
Length	30					mm
Height	30					mm
Width	30					mm
▼ Name						
×						
Bind signal						1
			ОК	Ар	ply	Cancel

Figure 3-11: Sennor

- Transfer plane: It is a physical property that converts the selected plane into a conveyor belt.
- Specify vector: Specify the transmission direction of the transfer plane.
- Parallel: specifies the speed magnitude in the transmission direction.
- Vertical: Specify the speed magnitude perpendicular to the transmission direction.(Figure 3-12)

🐲 Tran <mark>s</mark> fer plane			?	×
▼ Transfer plane				
Basic element				+
Base coordinate				+
 Velocity and position 				
Specify vector		7	•	< *
▼ Velocity				
Axial vector		1	mm/s	•
Collision				
Create collision				
▼ Name				
	ОК	Apply	C	ancel

Figure 3-12: Trasfer plance

- Position control: Command to create an actuator that drives an axis defined by a kinematic pair to move to a predetermined position at a predetermined constant speed.
- Axis type-cylinder-angle path option: used to define the rotation scheme of the axis kinematic pair.
- Target: Specify a target location.
- Velocity: Specify a constant speed.
- Base marker: composed of the robot base point and coordinate position, it is the foundation of other coordinate systems of the robot. (Figure 3-13)

8

W

💯 Position Control	? ×
Required	
Kinematic pair	+
Rigid body	+
Optional	
Base coordinate	4
Flange coordinate	+
Signal	+
Constrains	C I desse de
Shaft type C	law 🔻
Target	mm
Velocity	mm/s *
Limit acceleration	
Max acceleration	mm/s² *
Max deceleration	mm/s² *
🖾 Limit jerk Maximum jerk	mm/s ^a *
Limit force	
Positive force	N *
	N T
Negative force	
Negative force	

Figure 3-13: Position control

Add joint kinematic pairs, base coordinates, flange coordinates, robot parameters, etc. to create a robot. (Figure 3-14)

W

Assem	bly robot	Robot parameter		
Robot N	ame			
V Joint				
Select	Delete			
1	Vame	Direction	Offset value	
🔲 Join	t coupling			
	t coupling ng coefficie	nt -1		
Couplir				
Couplir	ng coefficie			4
Couplir	ng coefficie			+
Couplir V Base Name	ng coefficie coordinate			•
Couplir V Base Name	ng coefficie			- + -
Couplir V Base Name	ng coefficie coordinate			-+
Couplir V Base Name	ng coefficie coordinate			- ∔ -
Couplir V Base Name V Flan <u>c</u>	ng coefficie coordinate			+

Figure 3-14: Create robot

Select joint positions, add external data, maximum acceleration, maximum acceleration, base coordinates, and tool center coordinates to create robot tools. (Figure 3-15)

🐲 Robot tool		?	×
▼ Requried			
Rigid body			+
Base coordinate			+
Tool center			+
▼ Optional			
Kinematic pair			+ ×
Sensing point			+
Signal			+
Constrains			
Velocity		Ĩ	
		-	m/s *
Max jerk		m	m/s³ ▼
Max force			
Positive force		N	
Negative force		N	*
▼ Name			
	ОК	Apply	Cancel

Y

Figure 3-15: Robot tool parameter interface

Add motion array units, select basic objects, array elements, add speed, and simulate conveyor belt operation. (Figure 3-16)

💱 Motion array unit	ĵ	? ×
 Basic Element 		
Basic element		+
Base coordinate		+
 Array collection 		
		+
		×
▼ Velocity		
Velocity		mm/s *
▼ Name		

Figure 3-16: Motion array unit parameter interface

Add an external axis to allow the robot to move on it by selecting the joints, base, base coordinates, and flange coordinates. (Figure 3-17)

😨 External axis			?	×
▼ Joint				
+ ×				
Name	Direction	Offset v	alue	
Base coordinate	2			
				+
Name				
5	ite			
Name Flange coordina	ite			
Flange coordina	ate			+
Flange coordina	ste			+
 Flange coordina Name 	ate			+
	ite			•

Figure 3-17: External axis

Add joint positions, select kinematic pairs, input type and speed, and establish internal signals. (Figure 3-18)

💯 Joint posit	ion			? ×
▼ Kinematic p	air			
				4
▼ Name				
Motion type:	Specify po	osition 🔻		
		ОК	Apply	Cancel

Figure 3-18: Joint position

Material production unit: Use materials to create multiple objects with the same appearance and attributes at specific time intervals. (Figure 3-19)

💯 Material production unit			? ×
 Object to copy 			
Production object:			+
Production position:			+
Production frequency:			Hz
Produce Once Time			
▼ Name			
	ОК	Apply	Cancel

Figure 3-19: Material production unit

Simulation sequence: It is a shortcut for logic control and simulation verification, which can associate various actuators with sensors to achieve logical control. (Figure 3-20)

🐲 Simulation seque	ence					?	×
Execution time							
Start time							s
Duration							5
 Signal configuration 	1						
Activation signal							+
Activation condition	True					Ŧ	
🗹 End signal							+
Execution signal							+
End position							
▼ Name							
		Г	ОК	1	Appl	v	Cancel
		Ļ	UK		Appl	Y	Cancel

Y



- Signal adapter: Encapsulates runtime formulas and signals using its commands, which are included under the "Mechatronic Modeling" window signals, creating signal objects that can be used to connect external signals.
- Formula: Assign the formula displayed in the formula box to the selected signal. (Figure 3-21)

gnal					
Assign as	Name	Data type	In/Out	Initial value	
					-
rmula					
Assign as		Formula		Note	
					 -
					2
nula					
nula					

Figure 3-21: Signal adapter

Signal connection: Connect internal signals with external signals, and control the internal signals through external signals on external devices to view the simulation effect of the robot. (Figure 3-22)

Y

	ection							?	×
ignal connection	n name				Sigr	nalConnect1			
roject name									
levice name									
V Signal									
Name	Adapter name	IO type	Data type		Name	IO type	Data type		
				0					
				Ø					
7 Signal connect	tion								
		Outer sign	sal						
 Signal connect Inner signal 	tion Direction	Outer sign	al						
 Signal connect Inner signal 		Outer sign	ıal						
		Outer sign	ıal						
		Outer sign	tal						
		Outer sign	xal						
		Outer sign	xal						
		Outer sign	xal						
		Outer sign	al						

Figure 3-22: Signal connection

Signal: Divided into output signal and input signal, creating different signals to generate different simulation motion forms for the robot. (Figure 3-23)

💯 Signal			7	? ×
▼ Settings				
🗹 Connect runtime param	eters			
Parameter name	An	gle		•
IO type		ĺn		•
Data type		BC	OOL	
Dimension		Ar	ngle	
Unit		۰		
Initial value			false	•
▼ Name				
Signal name				
Controller				•
Port				
		ок	Apply	Cancel

Figure 3-23: Signal

- Label: By adding labels, parent and child assemblies can be assembled.
- Parent Label: Select the parent label from the label type drop-down list, and select the corresponding rigid body for the parent assembly and child assembly respectively.(Figure 3-24)

Parent label				5
Parent label	None			
Threshold	0.05			
Parent assembly				+
Child assembly				+
🛚 Name				

Figure 3-24: Parent label

Child Label: Select a child label under the label type, and select the parent label that has already been created from the drop-down list in the parent label box.(Figure 3-25)

Label			?	>
Label type				
Child label				•
Parent label	None			-
Name				
Vame				
Name				

Figure 3-25: Child Label

3.3 Gantt chart

After adding the simulation sequence, perform simulation verification, and the realtime sequence time will be displayed in the Gantt chart. (Figure 3-26)

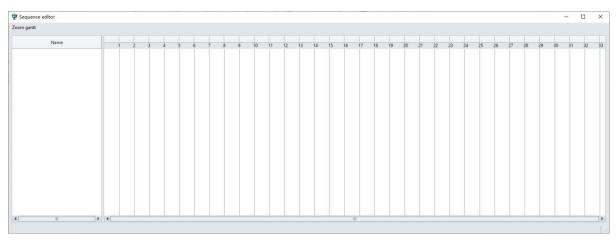


Figure 3-26: Gant chart

3.4 Assembly relationship

Y

Add assembly relationships, select child devices to install on a certain device, and adjust the posture to ensure a tight fit between the sub devices and the device. (Figure 3-27)

Child device Device installation		None None				
	ative pose	8	9		18	
х	þ	γ	0	Z	0	
RX	0	RY	0	RZ	0	

Figure 3-27: Assembly relationship

3.5 Teach programming

Manipulate the right angle space and joint space of the robot based on a 3D model, and add marker points. (Figure 3-28)

🖞 Rob	ot teach				?	×
Robot na	ame	GSK_Base				
Descar	tes space					
Base	coordinate	O Tool coordii	nate			
Linear in	crement[mi	m]	Angle inc	rement[deg	l	
10		;	1			÷
«X –	X++	1.052	RX-	RX+	0.000	
4Y-	Y+>	-0.000	RY-	RY+	1.570	
4Z-	Z+Þ	1.427	RZ-	RZ+	0.000	
Joint s	pace					
Axis1["]	0.000	-170]		170
Axis2[°]	0.000	-95				132
Axis3[°]	-0.000	-165				73
Axis4[°]	0.000	-180]		180
Axis5[°]	0.000	-133]		133
Axis6[°]	0.000	-360				360
Teach	programmir	20				
ل ماکم		<u>~ × (</u>		20		
1	₽ >>			رەرە		
Name	Notes		Param			

Figure 3-28: Teach

3.6 Collision detection

W

By adding collision groups and then collision pairs, you can see collision detection between robots and objects, which is highlighted. (Figure 3-29)

Collis	sion detection		E 23
-0	🗸 🗙 🖸	8	0 2
	Target object	•	
¶-=	Obstacle	all other collidable objects in 🔻	
		+	×
3			
2			

Figure 3-29: Collision Detection

3.7 Programming

V

In the "Programming" window, add a JOB program, add program program1, and then right-click to edit the program. (Figure 3-30)

Proga	m edit	
	✓ X ☑	0
*	🕨 📄 💽	
	1	
-	V RB120Controller	
	V 🔚 JOB1	
	program1	
2		

Figure 3-30: Programming

In the "Program Parameters" window, perform the add command. (Figure 3-31)

2 Program parameters	23
Robot GSK_Base	₿
▼ Required	
Name Type Data	
▼ Command	
System Motion Control IO	
Command Paramter	

Figure 3-31: Program Parameters

Control robot movement by adding instructions. Program instructions are classified as follows:

System instructions :

- 1. NOP (empty instruction, no action taken)
- 2. END (program end instruction, program stops when it appears, and even if there are instructions remaining, it will no longer run)
- 3. WAIT (system wait instruction, unit: ms)
- 4. SPEED
- 5. PAUSE (system stop instruction, when the program runs to this instruction, the system stops and no longer runs).

(Figure 3-32)



Figure 3-32: System instructions

Motion commands :MOVJ, MOVL, MOVA, MOVC, etc. (Figure 3-33)

Attention: Only one motion command can be used for each loc point, otherwise it may cause robot motion trajectory errors; If the same point needs to be used multiple times, please mark it multiple times with different names.

Sec. 168 111	1.22	1.000 00 00	1.00	
System	Motion	Control	10	

Figure 3-33: Motion commands

(1) HOME:

- > Function: Use joint interpolation motion to return the robot to zero point;
- **Format:** Simply click the HOME button;
- Explanation: When using HOME, it can only be added directly, and motion instructions cannot be added by inserting instructions;

(2) MOVJ:

- **Function:** Move to the target point through joint interpolation;
- Format: MOVJ {LOC (target point), VEL [] (velocity), ACC [] (acceleration), JERK [] (acceleration)};

Explanation: The unit of VEL is °/s; ACC and JERK both use percentage form, and the input range is a positive number. If it is less than or equal to 0, an error will be reported; (Figure 3-34)

WON 👰					?	×
Target:			•			
VEL:	200		°/s			
ACC:	100		%	1	2	
JERK:	100		%	//	R	
				J	Ø	

Figure 3-34: MOVJ

- (3) MOVL:
 - **Function:** Move to the target point through linear interpolation;
 - Format: MOVL {LOC (target point), VEL [] (velocity), ACC [] (acceleration), JERK [] (acceleration)};
 - Explanation: The unit of VEL is m/s; ACC and JERK both use percentage form, and the input range is a positive number. If it is less than or equal to 0, an error will be reported; (Figure 3-35)

WOVL		?	×
Target:		4	
VEL:	0.5	m/s	
ACC:	0.05	%	
JERK:	0.05	%	
		SL	Ø
	OK Appl	Cancel	

Figure 3-35: MOVL

- (4) MOVA:
 - Function: Circular interpolation motion mode passes through the middle point to the target point;
 - Format: MOVA {LOC1 (target point), LOC2 (intermediate point), VEL [] (velocity), ACC [] (acceleration), JERK [] (acceleration)};
 - Explanation: The unit of VEL is m/s; ACC and JERK both use percentage form, and the input range is a positive number. If it is less than or equal to 0, an error will be reported; (Figure 3-36)

WOVA			?	×
Target:		· •		
Intermediate:		-		
VEL:	0.5	m/s	~	
ACC:	0.05	%	A	
JERK:	0.05	%	AN	De la
			A L	D'
		(3	
	OK Apply	Cancel		

Figure 3-36: MOVA

(5) MOVC:

- Function: Move in a circular interpolation motion along the current point, target point, and intermediate point, forming a circle, and then return to the current position. When moving, it first passes through the target point and then passes through the intermediate point;
- Format: MOVC {LOC1 (target point), LOC2 (intermediate point), VEL [] (velocity), ACC [] (acceleration), JERK [] (acceleration)};
- Explanation: The unit of VEL is m/s; ACC and JERK both use percentage form, and the input range is a positive number. If it is less than or equal to 0, an error will be reported; (Figure 3-37)

MOVC			? ×
Target:		2.	
Intermediate:		(1)	
VEL:	0.5	m/s	\sim
ACC:	0.05	%	AB
JERK:	0.05	%	AKA
		(f	all a
		\bigcirc)
)

Figure 3-37: MOVC

(6) MOVS:

- > Function: Move to the target point through spline interpolation motion;
- Format: MOVS {LOC1 (target point), LOC2 (passing point), VEL [] (velocity), ACC [] (acceleration), JERK [] (acceleration)}; (Figure 3-38)

WOVS						?	×
Target:				•			
VEL:	0.5			m/s			
ACC:	0.05			%			
JERK:	0.05			%	1	b	
		0			J	A Class	I
		ОК	Apply	Cancel			

Figure 3-38: MOVS

Control instructions (Figure 3-39):

- FLYBY # ON :smooth transition on instruction;
- FLYBY # OFF :smooth transition off instruction;
- LOOP :loop start instruction, the number of times the partial loop of the END_LOOP instruction is set after this instruction, which is used in combination with END_LOOP;
- END_LOOP :loop bundle instruction, paired with LOOP;
- WHIL : While loop start instruction, parameter is loop start condition, is a conditional judgment statement, used in conjunction with END_WHILE;
- END_WHILE :WHILE loop content end instruction, used in conjunction with WHILE;
- IF : select structure instruction, parameter is a conditional statement, instruction after IF is an instruction that satisfies the condition and can be executed until ELSE,ELSE_IF, or END_IF is encountered to end the branch content;
- ELSE : another selection branch of IF or ELSE_IF statement, with END_IF as the end of the branch statement;
- ELSE_ IF : used for branch start instructions in multi branch structures, with ELSE as the end of the branch statement;
- END_IF: used for the end of all conditional statements, paired with IF;
- LABEL : constructs a label at the specified position, which can be used together with the GOTO instruction to adjust the instruction process;
- GOTO : jumps to the instruction position marked by LABEL to start execution, generally does not jump to the previous label, which can easily form a dead loop;

ystem	Motion	Control	10
0	(A) ()	50	<i>កំកុ</i> ក្'ុ)≫

Figure 3-39: Control instructions

The IO instructions (Figure 3-40):

- DIN :digital input, reading the signal of the specified port number into the BOOL type variable;
- DOUT : digital output, making different responses to the controller based on the read signal;

System	Motion	Control	10	
--------	--------	---------	----	--

Figure 3-40: IO instructions

3.8 URDF Import and Export

URDF is a format based on the XML specification used to describe the structure of robots, mainly used to define the structure of robots, including the components of robots and the relationships between them. URDF files can be used for robot simulation, visualization, and analysis in ROS (Robot Operating System) to describe the relationships between robot components and their relationships.(Figure 3-41)

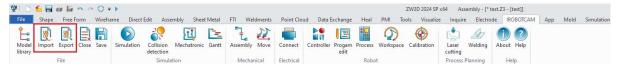


Figure 3-41: Import and Export

Import URDF: Select the path where the file is stored and click OK.(Figure 3-42)



Figure 3-42: Import URDF

Export URDF: In the device bar, click the "+" button, select the robot to export in the assembly relationship bar, select the path to store in the path bar, and click OK.(Figure 3-43)



Figure 3-43: Export URDF

3.9 Workspace

In the IROBOTCAM status bar, click the workspace button.(see Figure 3-44)

File	Shape	Free F	orm	Wirefra	me Direct	Edit Asse	mbly Shee	t Metal	FTI We	eldments	Point Clou	ud Data I	exchange	Heal	PMI Too	ls Visualize	Inquir	e Electroo	de IROBOTCAM	Арр	Mold	Simulati
Model library	Import				Simulation	Collision detection	Mechatronic	Gantt	L. Assembly	Move	Connect	Controller	Progam edit	Process	(A) Workspace	Calibration	Laser cutting	Market Welding	About Help			
		File				Simula	tion		Mecha	anical	Electrical			Robo	t		Process	Planning	Help			

Figure 3-44: Workspace

In the mechatronic bar, click on the collision group, and in the collision group window, create collision groups for the robot's joints. The collision body of each joint of the robot needs to be checked to support sensing. Click on the workspace, select the controller in the workspace parameters, and adjust the sensing range that the detection robot can collide with by modifying the spatial point density. (Figure 3-45)

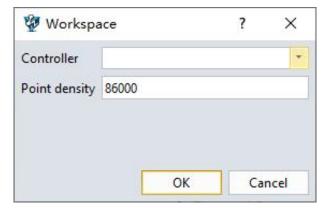
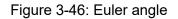


Figure 3-45: Workspace window

3.10 Euler angle

• Euler angle is a set of three independent angular parameters used to uniquely determine the position of a fixed point rotating rigid body.(Figure 3-46)

love				
5	🖌 🗶 🖪			0
	▼ Required			
-0	Select model:			1
2	▼ Optional			
	Position:	* \$	•	Ok
	X axis:	≥ ≤	•	Ok
9	Y axis:	* \$	•	Ok
	Z axis:	*	•	Ok
2	Euler angle			
	RPY:			
	EulerZYX:			
	EulerZYZ:			



3.11 Connect module

Y

• By adding devices, accessing the device server, and controlling internal signals through external signals from the server, communication functions are sequentially achieved. (Figure 3-47)

Opc Ua Sieme	ens Plc Smart	mounds ich	BeckHoff Controlle	
Device				
Device name	Server address	Server port		
V Outer signal				
♥ Outer signal Name	10 type	Data type	Type name	Device n
	IO type	Data type	Type name	Device n
	IO type	Data type	Type name	Device na
	IO type	Data type	Type name	Device n
	IO type	Data type	Type name	Device n
	IO type	Data type	Type name	Device n

Figure 3-47: Communication Module

3.12 Simulation verification

Y

- After programming, simulation verification can be conducted to demonstrate the robot's motion trajectory.
- Switch to real-time mode for mechatronic motion simulation. (Figure 3-48)

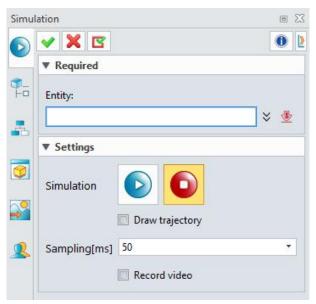
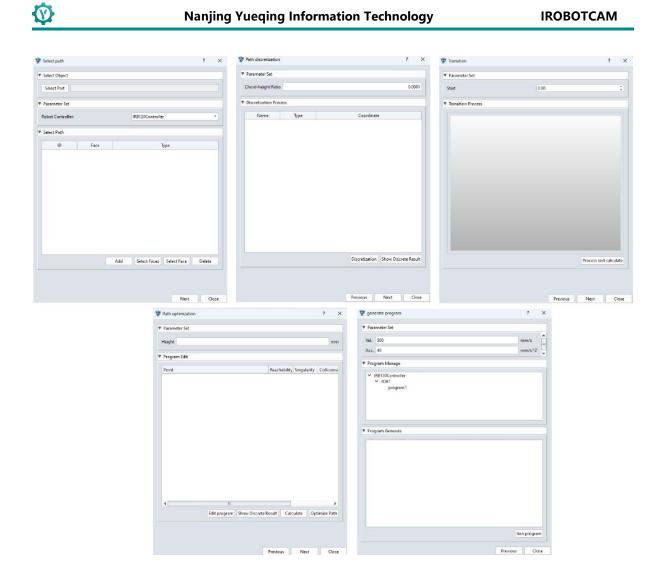


Figure 3-48: Simulation verification

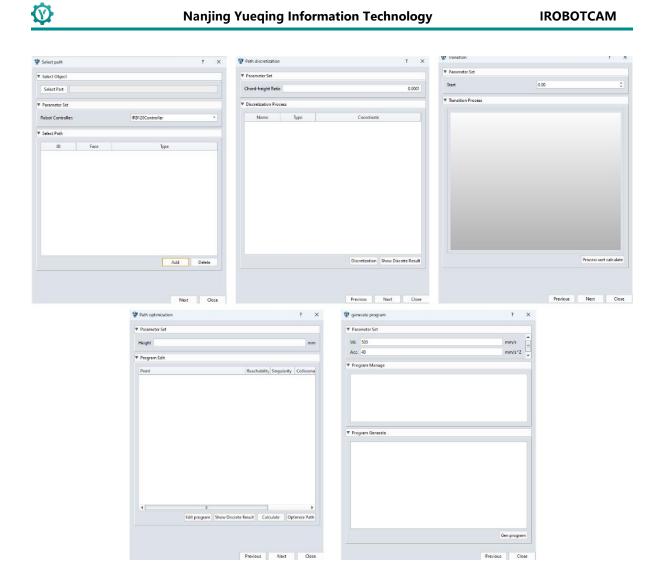
3.13 Laser cutting

- The laser cutting function is a functional module composed of multiple windows by clicking "Next".
- Step sequence: Select path Discretize path Sort transition points Optimize path Generate program.
- Specific operation steps are introduced with reference to application cases.



3.14 Welding

- The welding function is a functional module composed of multiple windows by clicking "Next".
- Step sequence: Select path Discretize path Sort transition points Optimize path Generate program.
- Specific operation steps are introduced with reference to application cases.



3.15 Screen recording function

- Click on the simulation button in the "IROBOTCAM" tab to open the "Simulation Control" interface.
- Check to enable video recording, and then click on the start simulation button on the left side of the "Simulation Control" window to start recording.
- Wait for the machine to run the entire process, and then finally click on the end simulation button on the right side of the "Simulation Control" window to end recording.
- The video recording is complete. (Figure 3-49)

Simu	lation	8 23
	✓ X	0
	▼ Required	
P =	Entity:	
	-	× 🔮
	▼ Settings	
3	Simulation	
	🔲 Draw trajectory	
2	Sampling[ms] 50	*
	🔲 Record video	

Figure 3-49: Screen recording function

Additional Tips:

- Video recording storage location: Open the installation location of ZW3D 2024 software, pull up to see the video recording file, click on the video MP4 file, and you will see the entire process of machine operation(see Figure 3-50).
- If the file for the screen recording function does not exist, you need to run the ZW3D with administrator rights.

名称	修改日期	类型	大小
amrecorder.mp4	2023/5/31 11:09	MP4 文件	3,865 KB
e license.xml	2023/5/31 10:49	Microsoft Edge	3 KB
amrecorder.log	2023/5/31 11:09	文本文档	1,439 KB
license.log	2023/5/31 10:49	文本文档	10 KB
🤄 recorder.dll	2023/5/29 18:52	应用程序扩展	181 KB
kdchart.dll	2022/12/1 16:49	应用程序扩展	1,488 KB
portaudio_x64.dll	2022/11/11 11:25	应用程序扩展	247 KB
gscintilla2_qt5.dll	2022/9/21 10:58	应用程序扩展	1,948 KB
productinfo	2022/7/25 10:41	文件	1 KB
clspecmk.exe	2022/7/20 15:10	应用程序	65 KB
gl-debugger.dll	2022/7/20 15:10	应用程序扩展	1,570 KB
logging.dll	2022/7/20 15:10	应用程序扩展	124 KB
3 z3Thumbnail_x64.dll	2022/7/20 15:10	应用程序扩展	108 KB
Zrc.exe	2022/7/20 15:10	应用程序	19 KB
i) zwoo.ali	2022/7/20 15:10	应用程序扩展	56,139 KB
🖉 zw3d.exe	2022/7/20 15:10	应用程序	493 KB
III ZW3D.lib	2022/7/20 15:10	Object File Library	378 KB
J ZW3D_Base.dll	2022/7/20 15:10	应用程序扩展	194 KB
JZW3D_Blas.dll	2022/7/20 15:10	应用程序扩展	211 KB
ZW3D_cam.dll	2022/7/20 15:10	应用程序扩展	9,478 KB
W3D CamDb.dll	2022/7/20 15:10	应用程序扩展	1,070 KB

Figure 3-50: Video recording storage location

Chapter 4: Notes and Save

IROBOTCAM User Interface: Notes and Storage

1. Shortcut Keys:

 Some shortcut keys are not available within the IROBOTCAM tab. This likely includes common shortcuts like Ctrl+S (save), Ctrl+Z (undo), Ctrl+X (cut), Ctrl+C (copy), and Ctrl+V (paste).

2. Storage:

• Encourage users to explore other methods of saving file data in order to gain a more comprehensive understanding of IROBOTCAM's storage capabilities.

Notes:

- In the IROBOTCAM module, this save button saves the content of the mechatronic.
- This save button does not save the file. (Figure 4-1)

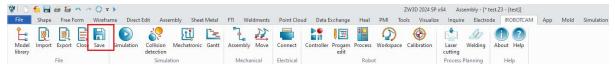


Figure 4-1: Save

Click Save and a save progress bar will pop up. After saving, the output box on the right will display the words "Successfully saved mechatronic data". (Figure 4-2)

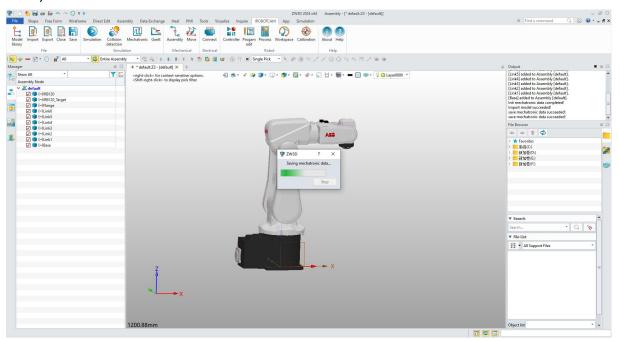


Figure 4-2: Output box

Save File:There are two ways to save: one is to overwrite the original imported file and save it, and the other is to save it as a new file.

• **Overwrite the original file:** In the "IROBOTCAM", click the save button in the upper left corner of the interface to overwrite the original file and save it.(Figure 4-3)

	68	6 6 m /	* 🗘 =	×											Z	N3D 2024 SP	x64
File	Shape	Free Form	Wirefran	ne Direct Ed	it Assem	bly Data	Exchange	Heal	PMI	Tools V	isualize l	nquire	IROBOTCAN	App	Simulation		
Ŀ	ST I				- <u>`</u>										۲	$\bigcirc \bigcirc $?
Model library	Import	Export Close	Save		Collision M letection	Aechatronic	Gantt	Assembly	Move	Connect	Controlle	er Progar edit	n Process	Workspace	Calibration	About H	lelp
		File			Simulatio	n		Mecha	nical	Electrical	l.		Robot	i.		Help	

Figure 4-3: Close window

• Save as a new file: After successfully clicking the 'Save' button, click the 'File' button on the toolbar.(Figure 4-4)

21 D	🖆 🔒	0 G	n 0	<> ₹	•											Z	W3D 2024 S	SP x64
File	Shape	Free F	orm	Wirefram	me Direct	Edit Ass	embly Data	Exchange	Heal	PMI	Tools V	isualize Ir	nquire	IROBOTCAN	App	Simulation		
						-			-									?
Model library	Import	200	Close	Save	Simulation	detection	Mechatronic	Gantt					r Progan edit			Calibration		
		File				Simula	ation		Mecha	nical	Electrical			Robot	t		Hel	р

Figure 4-4: File button

Click Save under the "File Box" and select the "Save As" option in the box on the right. (Figure 4-5) (Whether the save was successful will be displayed as text in the output box on the right.)

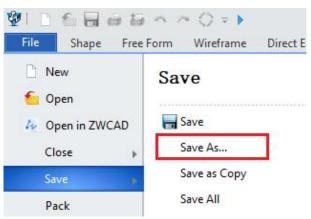


Figure 4-5: Save As

1. Case file storage location:

In the directory where ZW3D 2024 software is installed, click on the following path, and in the example folder and help folder of the plugins\iRobotCAM\example folder, it contains the project case and PDF version of the user manual. (Figure 4-6)

🕘 🖻 前 🚺 排序 ~	≣ 查看 ∽ …		
名称 ^	修改日期	类型	大小
🖁 Collision Detection.Z3	2023/10/25 16:41	ZW3D Document	66,539 KB
🖁 Configure Communication.Z3	2023/12/1 16:25	ZW3D Document	60,689 KB
🗑 Create Robot.Z3	2023/10/10 11:28	ZW3D Document	9,283 KB
🖁 Define Basic Mechatronic Objects.Z3	2023/6/5 10:54	ZW3D Document	205 KB
🖁 Define Conveyor Belts.Z3	2023/11/28 15:26	ZW3D Document	33,853 KB
🖁 Define Motion Array Units.Z3	2023/11/28 15:46	ZW3D Document	66,590 KB
Draw Arc.Z3	2023/10/5 9:34	ZW3D Document	6,575 KB
Robot Grab and Drop.Z3	2023/12/1 11:40	ZW3D Document	60,379 KB
🖏 Turntable Model.Z3	2023/7/3 13:33	ZW3D Document	42,397 KB

Figure 4-6: example

Location of the user manual on the software interface: Enter the "IROBOTCAM" tab and click the help button on the far right to pop up the PDF version of the user manual. (Figure 4-7)

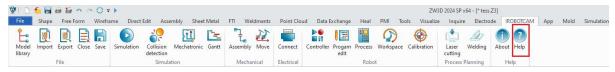


Figure 4-7: Help button

2. Entrance to IROBOTCAM module

In the start interface, next to Quick Start, click on the IROBOTCAM module, which includes the New, Open, and Activate buttons.

New: You can create a new project file.(Figure 4-8)

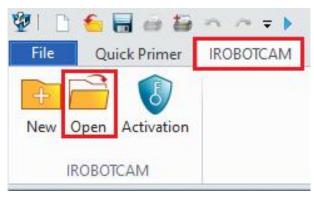


Figure 4-8: Open button

Open: Click the open button to open the corresponding Z3 file. Attention: When creating in the IROBOTCAM module Z3 files must be opened from this window in order to have relevant IROBOTCAM module content.(Figure 4-9)

😵 I 🗋 🗲 🖼 😅 😂 ా ~ ≂ ∓ 🕨	Z	W3D 2024 SP x64	= ∰ ⊠ ← Find a command Q @ •
New Open Activation			
¥ 17#	ogram Files > ZWSDF7 > ZW1D 2024 > RobotCAM v 6	X REFRONCAN*	File Browser (# 32 ************************************
組织 ▼ 新建文件夹		··· • • •	
rev:1320 rev:1320	御安日曜 契型 [▲] 大小 2022/11/28 10-50 文件単 2023/11/28 10-50 文件単	2008 000 2009 000 2009 0	V Sauch Search FileList Proview
文件곱(N):		IFIe(*Z3) ✓ IT开(O) ▼ 際論	
			Output • III 12 [121] successfully cleant.
Select command or entity.		17	

Figure 4-9: Open

Activation: You can see the product user manual of the software package

- Activation authorization: Fill in the activation code entry, click the application button, and the output box will prompt whether activation was successful.
- **Product information:** The CAD kernel status represents the status of ZW3D, which can be divided into three states: inactive, trial, and activated.
- **iRobotCAM status:** It also includes three states: inactive, trial, and activated. When inactive is showing, it indicates that the software trial period of 30 days has ended. The output box has relevant prompts.(Figure 4-10)

W

Activate		
Key ID		Appl
Trial license		
Directory		., Load
Product infomation		
CAD status	Activated	
	Activated	

Figure 4-10: License

Chapter 5: Define Basic Mechatronic Objects

This guide provides step-by-step instructions on defining basic mechatronic objects in IROBOTCAM software.(Figure 5-1)

Note:Compared to before, the software interface has been improved by adding density parameter settings in the material window, and removing mass settings in the rigid body and collision body windows. Users can pay attention to the changes in materials, rigid bodies, and collision body windows during the creative process.

When users create other cases, they may not be updated in a timely manner due to changes in other case images or videos. Users should pay attention to this issue when using the software. thanks

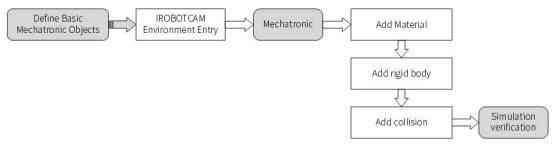


Figure 5-1: flow chart of define basic mechatronic objects

Step 1: Open iRobotCAM project file

 Open iRobotCAM project file named "Define Basic Mechatronic Objects. Z3" to access the IROBOTCAM environment. (Figure 5-2)

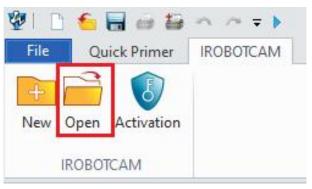


Figure 5-2: IROBOTCAM Environment Entry

• The interface will display three components. (Figure 5-4)

왕! 🗅 🌜 🗑 🗃 🗃 누 ~ 🔿 두 🕨	ZW3D 2024 SP x54 Assembly - [Define Basic Mechatronic Objects:Z3 - [Default]]	- 9 X
File Shape Free Form Wireframe Direct Edit Assem	nbly Sheet Metal FTI Weldments Point Cloud Data Exchange Heal PMI Tools Visualize Inquire Electrode IROBOTCAM App Mold Simulation	A Find a command Q 🛞 😧 * _ 🗗 🗙
Ibrary detection File Simulatic Imr Simulatic Minager 00 32	- 福福 1. 1. 1. 1. 1. 1. 1. 1. 1. 1. 1. 1. 1.	⇒ Output ■ Ⅲ ⊠ □ Define Bair Mechatronic
Image: Second secon	1600mm	[Define Basic Mechanomics Optical.2] Dones Dones And mechanisms of the second s

Figure 5-4: Component Details

Step 2: Create materials

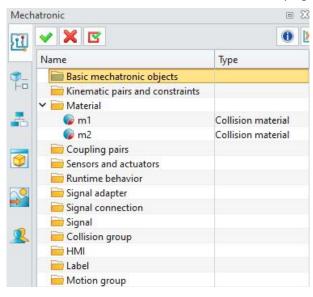
V

• Click on the "IROBOTCAM" to enter the mechatronic interface. (Figure 5-5)

🖗 I 🖻	6 🔒		n 10	0 =	•									ZV	/3D 2024 SP	x64	Assembly	- [Define	Basic Mecha	atronic Objects.Z3 -	[Default]]	
File	Shape	Free F	orm	Wirefram	ne Direct E	dit Asse	embly S	heet Metal	FTI (Weldments	Point Clou	id Data	Exchange	Heal	PMI 1	iools 1	Visualize	Inquire	Electroc	le IROBOTCAM	Арр	Mold	Simulation
Model library		Export			Simulation	Collision detection	Mechatro	onic Gantt		bly Move		Controller			Workspac	e Calibr	ation	Laser cutting	Melding	About Help			
		File				Simula	tion		Med	hanical	Electrical			Robo	t			Process P	lanning	Help			

Figure 5-5: Mechatronic

• Right-click on the material and select "Add Material". (Figure 5-6)



Ø

Figure 5-6: Add Materia	Figure	5-6:	Add	Material
-------------------------	--------	------	-----	----------

• Create a new material named "m1" for the two slopes. (Figure 5-7)

💯 Material			7	Y X
Property				
Frictional coefficient	0.1			
Restitution coefficient	0			
Linear damping	0			=
Angular damping	0			
Density	500			
▼ Name				
m1				
9				
		ОК	Apply	Cancel

Figure 5-7: Material m1

 Create another material named "m2" for the remaining components. (Figure 5-8)

		7	· >	<			
0.2				-			
0							
0							
0							
500				-			
	ОК	Apply	Canc	el			
	0 0 0	0 0 0 500	0.2 0 0 0 500	0.2 0 0 0 0 500			

Figure 5-8: Material m2

Step3: Create Rigid Body

• Right-click on "Basic mechatronic Object", and select "Add Rigid Body". (Figure 5-9)

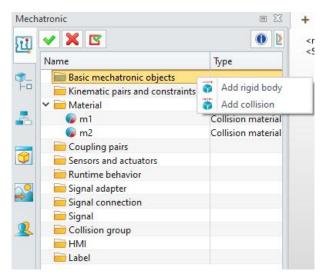


Figure 5-9: Add Rigid Body

- In the "Rigid Body" window, click the "+" button to select one of the slope components.
- Uncheck the property settings and click "OK".
- Repeat above steps to add the other slope as a rigid body. (Figure 5-10)

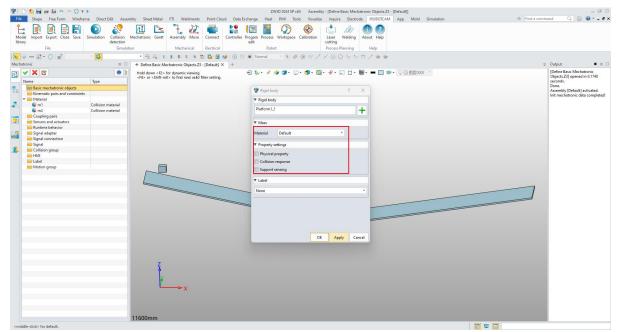


Figure 5-10: Adding a obliquity Rigid Body

- Select the "material" indicated by the arrow to create a rigid body.
- Uncheck the property settings and click "OK".

			ZW3D 2024 SP x64 Assembly - [* Define Basic Methatronic Objects:Z3 - [Default]] Exchange Heal PMI Tools Visualize Inquire Electrode [IROBOTCAM App Mold Simu	ା ଥାଇଥିଲେ 🖉 🖓 🖓 🖕 🖉 🗴
Model Import Export Close Save Si library	imulation Collision detection		rr Progam Process Workspace Calibration edt Robot Help	
	- Simulatio		Robot Process Planning Help □ ■ Normal * Normal * ✓	
Mechatronic	- W		I Normal A Rev Oraci / A O O O O O O V / A A	⇒ Output ■ III 🔀
	0 1	· · · · · · · · · · · · · · · · · · ·		Output In Easic Mechatronic
1 × X 🗉		Hold down <f2> for dynamic viewing. <f8> or <shift-roll> to find next valid filter setting.</shift-roll></f8></f2>		Objects.Z3] opened in 0.1740
Name	Туре			seconds. Done.
Basic mechatronic objects Platform1_1	Rigid body		😵 Rigid body ? X	Assembly [Default] activated.
Platform2_1	Rigid body		▼ Rigid body	Init mechatronic data completed!
Kinematic pairs and constraints				
🗸 🧰 Material			Materia_1 +	
📻 🗣 m1	Collision material			
😟 😡 m2	Collision material		▼ Mass	
Coupling pairs			Material Default	
Sensors and actuators			Iviateliai Delaut	
Signal adapter			▼ Property settings	
🙎 🧰 Signal connection				
🦮 Signal			Physical property	
Collision group			Collision response	
Eabel			Support sensing	
Motion group			▼ Label	
			* Label	
			None ·	
			OK Apply Cancel	
		Z		
		4		
		X		
		——————————————————————————————————————		
		11600mm		
<middle-click> for default.</middle-click>				

Figure 5-11: Add Materia rigid body

Step4: Creat Collision

- Right-click on "Basic mechatronic Object" and select "Add Collision".
- In the "Collision" window, click the "+" button and select one of the slope components.
- Select "Convex Decomposition" for the collision shape.
- Check "Collision Response" for the property setting.
- Select "m1" for the material.
- Repeat above steps to create a collision for the other slope. (Figure 5-12)

Hodel Import Export Close Save File	lation Collision Mechatronic Gantt detection Simulation	FIT Weldments Peint Cloud Data Exchange Heal PM Doots Visuality Atomity Weint Connect Connect Connect Weinty Weinty Connect Conne Connect Conn	Laser Welding Cutting Process Planning Help	
Platform1_1_part Co Platform2_1 Rig Meteria_1 Rig Meteria_1 Rig Meteria_1 Material m1 Co	Pe pe pe dedy discon discon material discon material	b to find next solid litter setting.	+ · ·	Copin C
(middle.click) for default	11600mm			

Figure 5-12: Add obliquity collision body

- Click the "+" button again and select the box to create a collision.
- Select "convex decomposition" for the collision shape.
- Check "physical property" and "collision response" for the collision setting.
- Select "m2" for the collision material. (Figure 5-13)

靈!!	<u>)</u> <mark>6 등</mark> 중 중 ~ ~ ○ ∓)			ZW3D 2024 SP x64 Assembly - [* Define Basic Mechatronic Objects.Z3 - [Default]]	- Ø X
File		e Direct Edit Asser	mbly Sheet Metal FTI Weldments Point Cloud Data E	cchange Heal PMI Tools Visualize Inquire Electrode IROBOTCAM App Mold Simulation 🗠 Find a	ommand Q @ @ • _ # >
Moc libra	del Import Export Close Save	Simulation Collision detection		Image: Process Workspace Calibration Larrer Welding Welding Aboot Help Robot Help Help Help Help Help	
-	+ - Bt · O [117]	. 3	- 1		
Mecha		E XX			⇔ Output ■ 🗷 🗵
	✓ X ⊡	0 2			[Define Basic Mechatronic Objects.Z3] opened in 0.1740
	Name	Туре			seconds. Done
9 -	Basic mechatronic objects				Assembly [Default] activated.
	V i Platform1_1	Rigid body			Init mechatronic data completed!
	Platform1_1_part	Collision			creating collision create collision succeeded!
-	✓ ⑦ Platform2_1	Rigid body			create collision succeeded: creating collision
	Platform2_1_part	Collision			create collision succeeded!
3	Materia_1	Rigid body			
	Kinematic pairs and constraints			😵 Collision ? 🗙	
	Y 🚞 Material				
2	👰 m1	Collision material		V Collision Object	
	🕼 m2	Collision material		Materia_1	
-	Coupling pairs			Materia_1 +	
<u> </u>	Sensors and actuators				
	Runtime behavior			▼ Shape	
	🧰 Signal adapter				
	Signal connection			Collision shape Convex decomposition	
	🧰 Signal				
	Collision group			Material m2 -	
	HMI				
	🚞 Label			V Property settings	
	i Motion group				
				☑ Collision response	
				Chiloine report Collisione report Support sensing	
				* Label	
				None *	
			14.5 Carlos (1997)		
			Z		
			Y		
			<u> </u>	OK Apply Cancel	
			11600mm		
<mid< td=""><td>dle-click> for default.</td><td></td><td></td><td></td><td></td></mid<>	dle-click> for default.				

Figure 5-13: Add Materia collision body

Step 5: View the framework for mechatronic

• At this point, you should have successfully created materials, rigid bodies, and collisions for all components. (Figure 5-14)

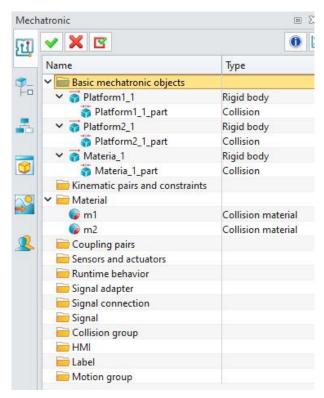


Figure 5-14: The entire mechatronic framework

Step 6: Simulation verification

• Click on the "IROBOTCAM" tab.(Figure 5-15)

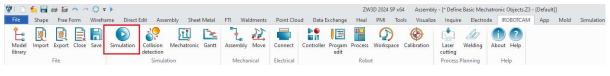


Figure 5-15: Simulation button

Click the simulation button to verify the defined mechatronic objects. (Figure 5-16)

W

望) 🖕 副 値 値 へ へ () テ) ZV00 2014 59 x64 Assembly - ['Define Basic Mechatomic Objects 23 - [Defunkt]] R Supe Tree Form Windrame Direct Edit Assembly Sheet Metal FTI Wedments Point Cloud Data Exchange Heal PMI Tools Vasalice Inquire Bectrade (NODIXLAM) App Mold Simulation	_ 2 ⊠
112 Shape / Free Term Windframe Direct Last Assembly Shape / Mella 191 Windframe Free Term Junit Free Term Simple Simple	🛶 Hito a cominisma 🤍 🔟 🔮 - 🗗 X
1 👿 🌳 — 🗄 + 〇 👔 🗛 🔹 🔹 👘 👘 👘 👘 👘 👘 👘 👘 👘 👘 👘 👘 👘	
Simulation ■ 10 file back Mechanoic Glogical 2: [04:ad] X ● > ● >	Output Output
Samplinging 10	7
Ĩ <u>↓</u> →×	
11600mm <midde-click- done.<="" td="" when=""><td></td></midde-click->	

Figure 5-16: Simulation verification

Chapter 6: Define Conveyor Belts

This document provides a step-by-step guide on how to define a conveyor belt in the IROBOTCAM software.(Figure 6-1)

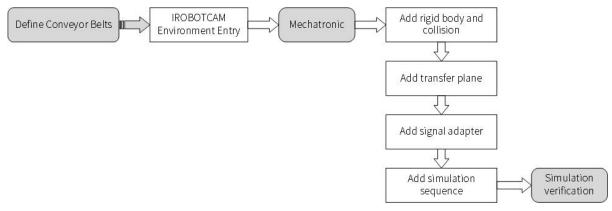


Figure 6-1: Flow chart of define conveyor belts

Step 1: Open iRobotCAM project file

Y

• Open iRobotCAM project file "Define Conveyor Belts.Z3" to access the IROBOTCAM environment. (Figure 6-2)



Figure 6-2: Open iRobotCAM project file

• Wait for the import to complete. A message will be displayed in the output box confirming completion. (Figure 6-3)

Tete Wordsmee Peer fam Wordsmee Peer fam Wordsmee Peer fam Wordsmee Peer fam Wordsmee Peer fam Wordsmee Peer fam Peer Peer fam Peer Pee	왕! 🗅 🌜 🗑 🗃 🗃 🗠 🗠 🔿 두 🕨	ZW3D 2024 SP x64 Assembly - [Define Conveyor Belts.z3 - [Default]]	- P X
	File Shape Free Form Wireframe Direct Edit Asser	mbly Sheet Metal FTI Weldments Point Cloud Data Exchange Heal PMI Tools Visualize Inquire Electrode IROBOTCAM App Mold Simulation	🌣 Find a command 🛛 🖓 🔞 * 🗕 🗗 🗙
Note with y lots Image: Strate with y lots Image: Strate with y lots Image: Strate with y lots Image: Strate with y lots Image: Strate with y lots Image: Strate with y lots Image: Strate with y lots Image: Strate with y lots Image: Strate with y lots Image: Strate with y lots Image: Strate with y lots Image: Strate with y lots Image: Strate with y lots Image: Strate with y lots Image: Strate with y lots Image: Strate with y lots Image: Strate with y lots Image: Strate with y lots Image: Strate with y lots Image: Strate with y lots Image: Strate with y lots Image: Strate with y lots Image: Strate with y lots Image: Strate with y lots Image: Strate with y lots Image: Strate with y lots Image: Strate with y lots Image: Strate with y lots Image: Strate with y lots Image: Strate with y lots Image: Strate with y lots Image: Strate with y lots Image: Strate with y lots Image: Strate with y lots Image: Strate with y lots Image: Strate with y lots Image: Strate with y lots Image: Strate with y lots Image: Strate with y lots Image: Strate with y lots Image: Strate with y lots Image: Strate with y lots Image: Strate with y lots Image: Strate with y lots Image: Strate with y lots Image: Strate with y lots	Model Import Export Close Save Simulation Collision library File Simulation Simulation library - Image: Simulation Simulation library - Image: Simulation Simulation	Machanomic Gartt Assembly More Connect Controller Program Process Workspace Calibration Laser Welding Adval Hep tion Machanical Entrical Robot Process Workspace Calibration Laser Welding Adval Hep * Machanical Entrical Robot Process Proces Process Process </td <td>⊕ Curput 🔹 🗉 52</td>	⊕ Curput 🔹 🗉 52
			Done.
418.56mm			
		⁷ × γ 41856mm	

Figure 6-3: Component Details

Step 2: Create rigid body and collision.

Y

• Go to the "IROBOTCAM" tab and click the "Mechatronic" button. (Figure 6-4)

21 B	6 🔒	a 13	n 0	<> ₹	•										ZW3D 202	4 SP x64 Ass	embly - [[Define Conve	yor Belts.Z3 - [Defa	ult]]		
File	Shape	Free F	orm	Wirefra	me Direct I	Edit Asse	embly Sheet	Metal	FTI We	eldments	Point Clou	d Data I	xchange	Heal	PMI To	ols Visualize	Inquir	e Electroo	le IROBOTCAM	Арр	Mold	Simulation
Model library	Import				Simulation	Collision detection	Mechatronic				Connect				Workspace	Calibration	Laser cutting	(Welding	About Help			
		File				Simula	ation	•	Mecha	anical	Electrical			Robo	t		Process	Planning	Help			

Figure 6-4: Mechatronic

• Under "Basic Mechatronic Objects", right-click and choose "Add Rigid Body". (Figure 6-5)

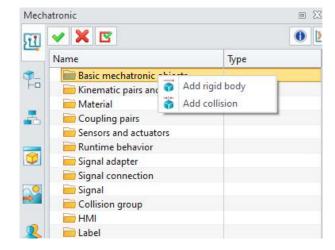


Figure 6-5: Add Rigid Body

- **Part:** Click the "+" button to enter the object selection status.
- Click the left mouse button to select the component , which is the long board, and create a rigid body.
- The material is "default", and the property setting is "unchecked".
- Click the OK button to create the first component as a rigid body. (Figure 6-6)

Model Import Export Clo File	n Windforme Direct Edit Accentify Sheet Meal PT Wodrevets Point Court Data Change Heal PM Took Vosalize Inquire Detroite RODOR-MA App Mold Simulation Sing Size Size Simulation Californic Gart Accentify Move Centered Program Process Workspace California Law Working Control Process Workspace California Calif	
Mechatronic → 🖓 + 🔿	Image: A Define Conveyor Bets:Z3 - [Default] X +	⊊ Output ■ III ⊠
Image: Signal conception Image: Signal c	d constraints f g flogd hody ? X v flogd hody	Diffus Convoyo Rehit.23) spendel in See. 24 Areas by Diffusil) activated. Init mechanisms data completed
<middle-click> for default.</middle-click>		

Figure 6-6: Add Long Board Rigid Body

- Click the "+" button to enter the object selection status, select the component, which is the long board, to create a collision.
- Select "convex decomposition" for the collision shape, "collision response" for the property settings, and "default" for the material. (Figure 6-7)

File Shape Free Form Wirefram Model Import Export Close Save File File File	e Direct Edit A imulation Collisio detection Sim	n Mechatronic Gantt Assembly Mov	i 🖷 🔡	ZW3D 2024 SP +64 J Exchange Heal PMI Tools Visual PIE Progam Process Workspace Calibration edt Robot	1 10 10 1	30TCAM App Mold	Simulation	Find a comm	ං බි යි and Q ම ම • - 8 x
$\mathbf{N} \div = \mathbf{H} \cdot \mathbf{O} \cdot \mathbf{n}^{T}$	- 🗳			🖓 🔳 Normal 📑 🍋 🏘 🛞 🕸 🖉	1/00427/1	* *			
Mechatronic		+ * Define Conveyor Belts.Z3 - (Defa	ult] × +					¥	Output 🔳 🖽 🖾
Image: Second	Pagia body Rajał body Rajał body Rajał body	Collision Object Perturn Support sensing Lead Nene Collision Collision Object Perturn Perturn Perturn Collision Adapt Control Collision Adapt Collision A	r ther setting.						[Define Conveyor Behr 21] operate in 64579 second. Received and a second second second second Received Second Second Second Second Second Second Int mechatronic data completed
<middle-click> for default.</middle-click>		349.305mm							

Figure 6-7: Add Long Board Collision

 Material: With the "+" button, select the material to create it as a rigid body, with a default material, and an property setting of "unchecked". (Figure 6-8)

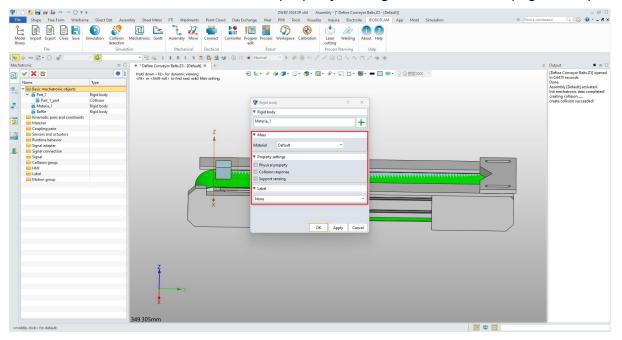


Figure 6-8: Add Rigid Body

- Click the "+" button to enter the object selection status, select the component on the right to create a collision for the material.
- Select "convex decomposition" for the collision shape.
- Select "physical properties", "collision response", "support sensing" for property settings, and "default recognition" for material. (Figure 6-9)

Import Export Close Save Nodel Import Export Close Save File Import Export Close Save	Direct Edit Assembly Sheet	Gantt Assembly Move Connect Controller P	change Heal PMI Tools Visualiz	embly - (* Define Conveyor Belts:Z3 - [Defat Inquire Electrode IROBOTCAM Laser Welding Cutting Process Planning Help		☆ [Find a command Q]	- 0 X
				/ 00 / 11/ 44		⇒ Output	■ m X3
Machatoma Norme Name Name Second	B Compared to the second se	Compose the Ita-2 - (Default) X + + + + + + + + + + + + + + + + + +	Colice Materia V Colice Materia V Colice Colice Materia V Popp V Colice Materia V Popp V Colice Materia V Colice V Colice	in the first interview of the first interview	? X	I (Defec Comp) I (Defec Comp) Dos. Dos. Dos. Dos. Defec Defec cast collector creating collector creating collector creating collector	r Belts.Z3] opened s. itt] activated. data completed!
<middle-click> for default.</middle-click>	349.305r	*					

Figure 6-9: Add Small Wooden Collision

• **Barrier:** Use the "+" button to select the material barrier plate zdk and create it as a rigid body. The default mass is "1", the material is "default", and the property setting is "unchecked". (Figure 6-10)

월 🗋 🌜 🖬 🗃 🗃 🗠 수 🔿 🖛			ZW3D 2024 SP x64 Assembly - [* Define Conveyor Belts.Z3 - [Default]]	- 9 X
File Shape Free Form Wirefram	ne Direct Edit Asser	nbly Sheet Metal FTI Weldments Point Cloud Data E	Exchange Heal PMI Tools Visualize Inquire Electrode IROBOTCAM App Mold Simulation 🗢 Fir	d a command 🛛 🔾 🛞 😧 • 💶 🗗 🗙
Hodel Import Export Close Save File Note: The Save Save Save Save Save Save Save Sav	Simulation Simulation		Image: Program Image: Progra	
Mechatronic	m 23	+ * Define Conveyor Belts.Z3 - [Default] × +		⊊ Output ■ ⊠
Market with a second se	Rigid body Collision Rigid body Collision Rigid body	* Une Converse tetted - (Default) A T Held daven (45 or dynamic version) 4/8 or <5hthroth: to find met wald filter setting.	C Apply Cated	Under Concept and All Second Concept and Concept
		X v		
		349.305mm		
<middle-click> for default.</middle-click>				

Figure 6-10: Add Barrier Plate Rigid Body

- Click the "+" button to enter the object selection status.
- Select the component zdk on the right, which is the barrier, to create a collision.

V

- Select "convex decomposition" for the collision shape,"collision response" for the property settings, and "default" for the material.
- Click OK. (This process needs to be patient for a while) (Figure 6-11)

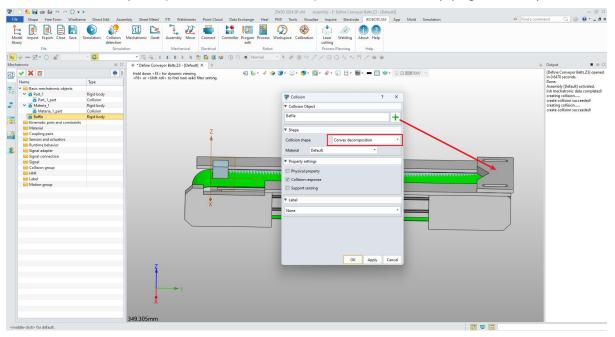


Figure 6-11: Add Barrier Plate Collision Body

Step 3: Basic mechatronic object framework

 All rigid bodies and collisions should now be listed under "Basic Mechatronic Objects." (Figure 6-12)

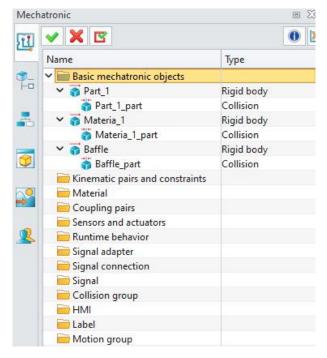


Figure 6-12: Basic mechatronic object framework

Step 4: Add transfer plane

 Right-click "Sensors and actuators" and select "Add transfer plane". (Figure 6-13)

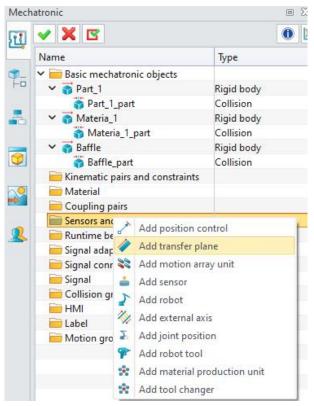


Figure 6-13: Add Transfer plane

- Select "Part" as the collision object.
- Click the "+" button and select CSYS1 as the base coordinate.
- In the "Transfer Surface" window, select "Second Entry" from the "Specify Vector" drop-down list. (conveyor movement along Y-axis).
- Enter the conveyor belt speed as "100".
- Named the transfer plane.
- Click "OK" to generate the conveyor belt object. (Figure 6-14)

Model Import Export Close Save File	detection edit cutting Simulation Mechanical Electrical Robot Process Planning Help	ං බ 22 mmand Q @ • - ඒ x
Mechatronic · ○ · · · · · · · · · · · · · · · · ·	3 ・ 売 売 15 15 15 16 🖄 🖄 🥸 🕼 (1) 🕷 Normal ・ 16 🥙 (1) 🕷 Normal ・ 16 🖉 (1) (1) (1) (1) (1) (1) (1) (1) (1) (1)	⇒ Output ■ III ⊠
Basic mechatronic objects Part,1 Part,2 Part,1 Part,2 Part,2	ended holds for constructions entries entries entries. ended ended holds ended holds ended holds ended holds ended holds ended holds <	portion of conveyer Beta-21 opported in 0.4719 seconds. Does with the second second second second limit mechanism of a second second second contrary collaboration of the second second second contrary collaboration of the second second second contrary collaboration of the second second second second contrary collaboration of the second second second second contrary collaboration of the second second second second second contrary collaboration of the second seco
		S 英 ', \$ 🖩 ¥ o # *
<middle-click> for default.</middle-click>	169.431mm	

Figure 6-14: Transfer plane

Step 5: Add a signal adapter

Click "Signal Adapter" and right-click to select "Add Signal Adapter". (Figure 6-15)

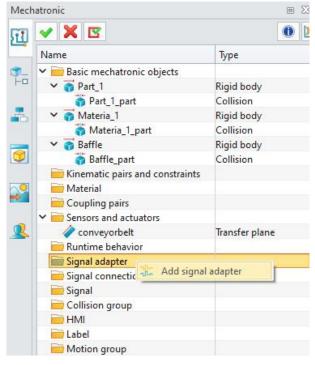


Figure 6-15: Signal Adapter

• Click "Sensors and actuators" then click the "+" button.

• Select "Transmission surface".

V

 Add parameters as shown in the provided figures, including signals, formulas, and names. Click OK to generate signal adapter objects and signal objects. (Figure 6-16)

1 1 1	🗋 🌜 🔚 🎯 😂 🗠 🗠 🛟 🖛		embly Sheet Metal FTI Weld	ments Point Clou	d Data Exchange			Define Conveyor Belts.Z3 - [E re Electrode IROBOTCA		ld Sim	ulation 🗠 Find a command 🔾 🛞 🖗 🕶 🕳 🗗 🗙
- N	lodel Import Export Close Save	D 🔅	Mechatronic Gantt	Signal adapter ▼ Signal	<u> 1</u>				?	×	
				Assign as	Name	Data type	in/Out	Initial value		+	
	$+ - \oplus \cdot \bigcirc \mathfrak{n}^{\vee} $	· · 🗳	· · · · · · · · · · ·	conveyorbelt	conveyorbelt_S		• IN • 0				
Med	hatronic	00 X	+ * Define Conveyor Belts.Z3 -		control on one of the					X	🗢 Output 🔳 💷 🖾
1		0	<right-dick> for context-sensit <shift-right-dick> to display pi</shift-right-dick></right-dick>								[Define Conveyor Belts.Z3] opened in 0.6470 seconds.
	Name	Туре									Done. Assembly [Default] activated.
9-D	✓										Init mechatronic data completed
PO		Rigid body									creating collision
	Part_1_part	Collision									create collision succeeded!
-	✓	Rigid body									creating collision create collision succeeded!
	Materia_1_part	Collision									create collision succeeded: creating collision
	✓ [™] Baffle	Rigid body									create collision succeeded!
		Collision									
	Kinematic pairs and constraints										
2	🖮 Material										
	Coupling pairs										
-	Sensors and actuators										
4	conveyorbelt	Transfer plane									
	Runtime behavior										
	📰 Signal adapter										and the second
	Signal connection			▼ Formula							and a second descent de
	📄 Signal			Assign as		Formula		Note		+	
	Collision group				conveyorbelt_S			1			
	HMI			conveyorbelt_S	conveyorbelt_S					X	
	📄 Label									\sim	
	Motion group										
				Formula							
			Z	▼ Name							
			4								
				conveyorbelt_SD							
				conveyorbeit_so							🔁 A 😗 🍨 🗃 🍟 💩 🗱 🕸
			→ Y								
			×								
								OK	Apply	Cancel	
				-	_	_	_				
			169.431mm								

Figure 6-16: Add Signal Adapter

Step 6: Add runtime behavior

 Click "runtime behavior", right-click and select "Add Simulation Sequence". (Figure 6-17)

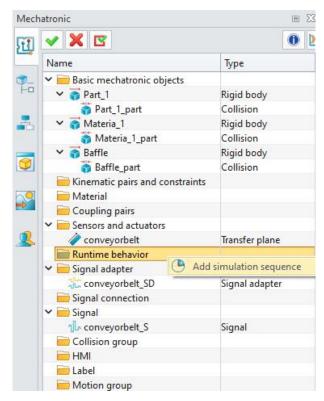


Figure 6-17: Add simulation sequence

- Set the Start time as "0" and the Duration as "9999999".
- Uncheck the "Activation Signal" and "End Signal".
- Click the "+" button and select "Input signal" as the execution signal.
- Set the End Position to "1".
- Click "OK". (Figure 6-18)

월 🗋 🕤 🚰 🗃 🗁 🗠 🔿 = 🕨	ZW3D 2024 SP x64 Assembly - [* Define Conveyor Belts:Z3 - [Default]]	
File Shape Free Form Wireframe Direct Edit Assembly Sheet Metal FTI Weldm	ents Point Cloud Data Exchange Heal PMI Tools Visualize Inquire Electrode IROBOTCAM App Mold Simulation 🛆 Find a com	nand Q @ • _ # ×
Let Import Early Import Early Import Early Assembly Mechanical Assembly Mechanical	edit cutting	
N + - B + O a ² 1 S = 1 + B + B + B + B + B + B + B + B + B +	巻 🗮 🔯 🏘 🕼 ① 円 🖷 Normal 💦 N 🥙 ④ 🖅 / / 〇 〇 〇 〇 〇 〇	
Mechatronic III 🕮 🖾 + * Define Conveyor Belts-Z3 - (De	fault X +	Output 🔳 💷 🖾
III	options. 🗧 🦦 - 🖉 🥩 🐨 - 🗊 - 🕸 - 🔝 - 🔶 - 📰 - 👘 - 🛄 🐲 - 🗘 🖓 🕮 🕬	[Define Conveyor Belts.Z3] opened in 0.6470 seconds.
Name Type		Done. Assembly [Default] activated.
Basic mechatronic objects V Basic mechatronic objects V Basic mechatronic objects		Init mechatronic data completed!
Part_1 Rigid body		creating collision
Part_I_part Consion ✓ ✓ Materia_1 Rigid body		create collision succeeded! creating collision
Materia 1 part Collision		create collision succeeded!
→ Saffle Rigid body		creating collision
Collision	Simulation sequence ? ×	create collision succeeded!
Kinematic pairs and constraints		
Material	▼ Execution time	
Cooping pairs	Start time 0 s	
Y Consors and actuators		
k onveyorbelt Transfer plane	Duration 9999999 s	4
Runtime behavior		
Y 🖻 Signal adapter	▼ Signal configuration	1
Conveyorbelt_SD Signal adapter		
✓ 🔤 Signal	Activation signal	
1 conveyorbelt_S Signal	Activitien condition	
Collision group		
E HM	End signal	-
🚞 Label	Execution signal conveyorbelt, S	
Contract of the second	Execution signal conveyorbelt_S	
	End position 1	
	▼ Name	
	conveyorbelt_EXEC	
	Contegoration	
	OK Apply Cancel	
Z		
1		
		🔁 A 🤫 🍨 🗃 🍟 💩 👪 🅸
*		
×		
169.431mm		

Figure 6-18: Simulation Sequence

Step 7: Simulation Verification

Y

• Click the simulation button Under the "IROBOTCAM". (Figure 6-19)

	6 🔒	ii 1	n 0	0 =	•										ZW3D 2024	SP x64 Ass	embly - [*	Define Conv	reyor Belts.Z3 - [De	fault]]		
File	Shape	Free F	orm	Wirefra	me Direct	Edit Asse	embly Shee	t Metal	FTI We	Idments	Point Clou	ud Data E	xchange	Heal	PMI To	ols Visualize	Inquir	e Electro	de IROBOTCAN	Арр	Mold	Simulation
Model library	Import			Save		Collision detection	Mechatronic				Connect				(A) Workspace	Calibration	Laser cutting	Welding	About Help			
		File				Simula	ation		Mecha	nical	Electrical			Robo	t		Process	Planning	Help			

Figure 6-19: Simulation Button

- Click the "Start Simulation" button in the "Simulation Control" window.
- Observe the material being transported to the right side of the conveyor belt. (Figure 6-20)

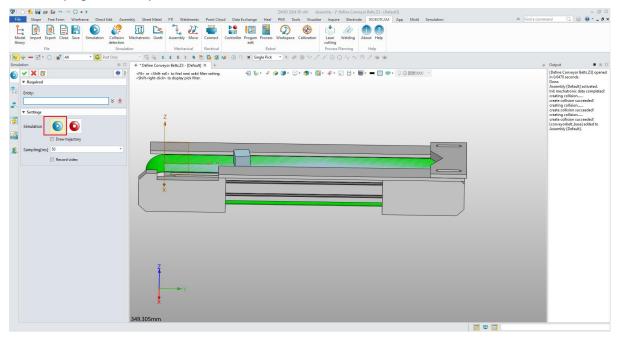


Figure 6-20: Simulation Verification

Chapter 7: Define Motion Array Units

This document outlines the process of defining motion array units within the iRobotCAM environment. This guide is intended for users familiar with the basic functionalities of iRobotCAM and seeks to provide a detailed and professional approach to setting up and simulating conveyor belt systems. (Figure 7-1)

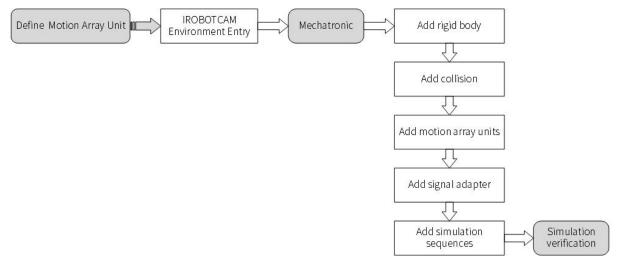


Figure 7-1: Flow chart of define motion array unit

Step 1: Opening and Selecting the Assembly

 Open iRobotCAM project file "Define motion array Unit. Z3" to enter the IROBOTCAM environment. (Figure 7-2)

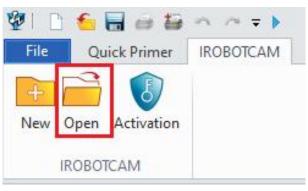


Figure 7-2: Open iRobotCAM project file

• Open the details of the interface components. (Figure7-4)

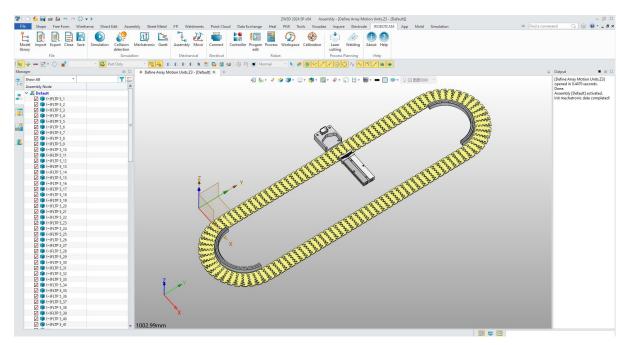


Figure 7-4: Component Details

Step 2: Creating Rigid Body for FLTP5_7

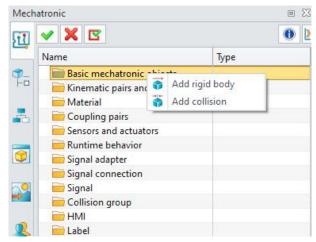
V

• Navigate to the "Mechatronic" tab within the "IROBOTCAM" interface. (Figure 7-5)

File	Shape	Free For	m 1	Wirefram	e Direct I	Edit Asse	mbly Shee	t Metal	FTI We	eldments	Point Clou	d Data Exc	hange Heal	PMI 1	ools Visualiz	e Inquir	e Electroo	de IROBOTCAM	Арр	Mold	Simulatio
Model library	Import		lose		Simulation	Collision detection	Mechatronic	Gantt				Controller P	rogam Process		Calibration	Laser cutting	Market Welding	About Help			
		File				Simula	tion		Mecha	anical	Electrical		Rob	ot		Process	Planning	Help			

Figure 7-5: Mechatronic

 Right-click within the "Basic Mechatronic Objects" panel and select "Add RigidBody" to open the "Add Rigid Body" window. (Figure7-6)





- Click Click the "+" button and choose the component "FLTP5_7" from the available list.
- "Property setting" parameters to suit your specific requirements.
- Notes: Material use "default", property setting to "unchecked".
- Click "OK" to create the rigid body for "FLTP5_7". (Figure 7-7)

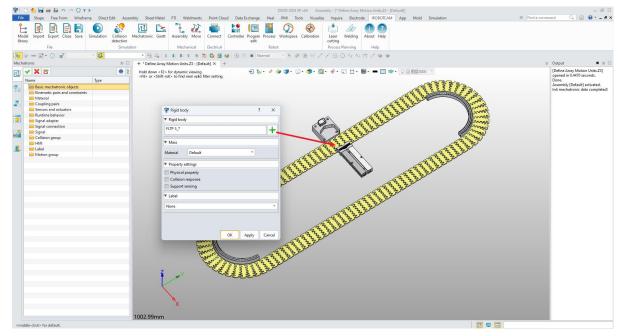


Figure 7-7: Rigid Body Information

Step 3: Creating Collision Body for FLTP5 7

- The Right-click within the "Basic Mechatronic Objects" panel and select "Add Collision" to open the "Add Collision Body" window.
- Click the "+" button and choose the component "FLTP5_7" from the available list.
- Notes: The motion array unit needs to select a conveyor belt component (including rigid body and collision) as the basic model to create the entire conveyor belt. Therefore, the components selected for creating collision bodies here should be consistent with the components selected for creating rigid bodies in step 2.
- Modify the "Mass" "Material" and "Property setting" parameters to suit your specific requirements.
- Notes: Mass could be input 0.1kg. the material use "default", the collision shape needs to be selected as "convex decomposition", check the "Collision response" under the Property settings.
- Click "OK" to create the Collision body for "FLTP5_7". (Figure 7-8)

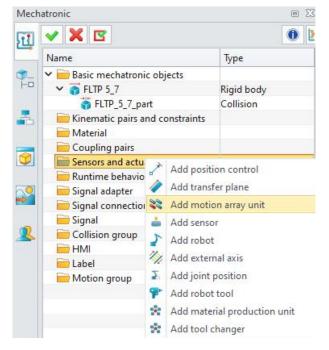
월 🗈 🗧 🗑 🖶 🖨 😂 🗠 🔿 🔿 🖘	ZW3D 2024 SP x64 Assembly - [* Define Array Mation Units.Z3 - [Default]]	- B X
File Shape Free Form Wireframe Direct Edit Assembly	Sheet Metal FTI Weldments Point Cloud Data Exchange Heal PMI Tools Visualize Inquire Electrode IROBOTCAM App Mold Simulation	🌣 Find a command 🛛 🖓 🛞 🖲 🗶 🗙
library detection File Simulation	Image: Contract Contract Program Process Webspace Collision Image: Contract Process Webspace Contract Process Webspace Collision <	
	* Define Array Motion Units:Z3 - [Default] X +	
Image: Control of the second secon	gas-did- for contract-sequence generalization of the sequence	Define Anny Motion Unite:23) operative and a second second second second second Assembly (Markoll Activated. Int mechanismic data completed
	2.99mm	
<middle-click> for default.</middle-click>	12.99mm	

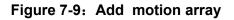
Figure 7-8: Collision Information

 Notes: It should be noted here that the waiting time of the convex decomposition algorithm varies depending on the complexity of the model and the computer performance of the user. The waiting time of this example model is approximately 10 seconds.

Step 4: Adding motion array Units

 Click Right-click within the "Sensors and actuators" panel and choose "Add motion array unit". (Figure 7-9)





- Click the "+" button and select the previously created "FLTP5_7" rigid body as the base element for the motion array unit.
- Click the "+" button and select a CSYS as the base coordinate.
- Define the desired "Speed" (eg, 100) at which the conveyor belt will operate.
- Assign a descriptive "Name" for easy identification. (Figure 7-10)

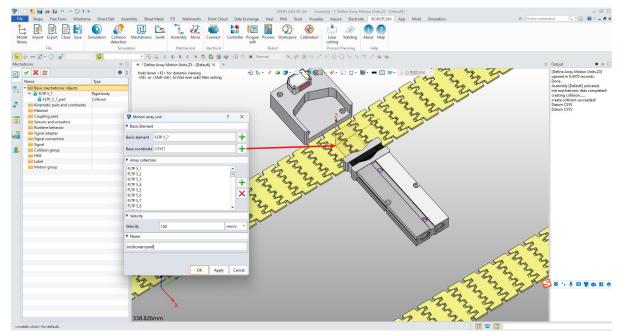


Figure 7-10: Basic Object

- Click Click the "+" button again to initiate the selection of all components comprising the conveyor belt as individual array elements.
- Carefully highlight and select each element within the 3D viewport by pressing the middle mouse button. (Figure 7-11)

SV.

		ZW3D 2024 SP x64	Assembly - [* Define array motion units.Z3 - [C	newo_run_do_oco_do_itteli		
Shape Free Form Wireframe Direct	Edit Assembly Data Exchange Heal PMI	Tools Visualize Inquire IROBOTCAM App	Simulation		Find a command	Q @ @•_
el Import Export Close Save Simulation	Collision Mechatronic Gantt detection	Connect Controller Progam Process Workspa- edit Robot	e Calibration Help			
C			* @ + / / 0 0 N N T /			
			@ @ ~ / / O O ^ ^ ~ ~ /	2.4		
er	Image: Section Units 23 - [Demo_F]				Output Assembly [Demo_Full_QG_GCE	
Assembly Node	Image: second	nts.	3 • ∲• 🖬 번• ■• ■ 🗋 ●•	2 🥥 函葉0000 *	creating collision create collision succeeded! Detum CSYS	o"do"inel activateu.
(-)FLTP 5.85	💇 Motion array unit	7 ×			Datum CSYS	
🖓 📦 (-)FLTP 5_87	V Basic Element				File Browser	
📝 🃦 (+)FLTP 5_88					+ + + 0	
2 C+IFLTP 5_89	Basic element FLTP 5_7	+			> Favorites	
(-)FLTP 5_90 (-)FLTP 5_91					> Windows(C:)	
(-)FLTP 5_92	Base coordinate CSYS1	+			> CATA1(D:)	
-IFLTP 5_93	Array collection					
Image: Control of the second secon	and concession					
Image: Contract of the state						
🔽 🧊 (IFLTP 5_96						
G C-IFLTP 5.97 shift + left m		+				
	ouse button					
CI CI LIGITO E 09						
Select from	FLTP 5_1 to FLTP 5_117 all the	components	(Tr)			
C CIETTS 38 C CIETTS 39 C CIETTS 39 C CIETTS 39 C CIETTS 30 C CIETTS 30 C CIETTS 30 C CIETTS 30 C CIETTS 30	FLTP 5_1 to FLTP 5_117 all the the mouse to the 3D window of	components x on the right and				3 o •, 🌢 📾
C G (-IFLTP 5,98 C G (-IFLTP 5,99 C G (-IFLTP 5,90 C G (-IFLTP 5,100 C G (-IFLTP 5,100 C G (-IFLTP 5,100) C	FLTP 5_1 to FLTP 5_117 all the the mouse to the 3D window o ddle button to confirm the sel	components x on the right and	(°)			3 o •, 🍨 📾
C (-IFLTP 5.88 C (-IFLTP 5.90) C (-IFLTP 5.100 C (-IFLTP 5.100) C (-IFLTP	FLTP 5_1 to FLTP 5_117 all the the mouse to the 3D window of	components x on the right and	(E)	ž l v		3 + • • •
G (-RLP 5, 8) G (-RLP 5, 9) G (-RLP 5, 10)	FLTP 5_1 to FLTP 5_117 all the the mouse to the 3D window of ddle button to confirm the sel	e components X on the right and lection of all components	(E)	1 - v		3 • • • • •
C (-RLP 5 (0) C (-R	FLTP 5_1 to FLTP 5_117 all the the mouse to the 3D window o ddle button to confirm the sel	components x on the right and	18 A	a v		3 • • • •
C (- (-RUP 5, 88) Select from 1 C (- (-RUP 5, 10)	FLTP 5_1 to FLTP 5_117 all the the mouse to the 3D window of ddle button to confirm the sel	e components X on the right and lection of all components	E .	N. V		3 ቀ •, ∳ ₪
2 (-intro 5.88) 2 (-intro 5.88) 2 (-intro 5.10) 2 (-i	FLTP 5_1 to FLTP 5_117 all the the mouse to the 3D window of ddle button to confirm the sel Velocity Velocity	e components X on the right and lection of all components	18	1 ···		3 + •, • =
2 0	FLTP 5_1 to FLTP 5_117 all the the mouse to the 3D window of ddle button to confirm the sel Velocity Velocity	e components X on the right and lection of all components	E.	L.Y	× Sarch	
2 □□□□□□□□□ Select from 1 3 □□□□□□□ Select from 1 4 □□□□□□□ Select from 1 5 □□□□□□ Select from 1 6 □□□□□□ Select from 1 7 □□□□□ Select from 1 7 □□□□ Select from 1	FLTP 5_1 to FLTP 5_117 all the the mouse to the 3D window of ddle button to confirm the sel Velocity Velocity	e components X on the right and lection of all components	(E)	L.Y		
○ □	FLTP 5_1 to FLTP 5_117 all the the mouse to the 3D window of ddle button to confirm the sel Velocity Velocity	e components X on the right and lection of all components	10 F	N. Y	▼ Search Search	
(*********************************	FLTP 5 1 to FLTP 5 117 all the the mouse to the 3D window of ddle button to confirm the sel V Weaky Veraky Veraky	ecomponents on the right and lection of all components	10 M	Ú ···	▼ Search Search	
(a) (110 3	FLTP 5 1 to FLTP 5 117 all the the mouse to the 3D window of ddle button to confirm the sel V Weaky Veraky Veraky	e components X on the right and lection of all components	19 I		▼ Search Search	
C INNE SEE	FLTP 5 1 to FLTP 5 117 all the the mouse to the 3D window of ddle button to confirm the sel V Weaky Veraky Veraky	ecomponents on the right and lection of all components	Ê	× ×	▼ Search Search	
C □ □ □ □ □ □ □ □ □ □ □ □ □ □ □ □ □ □ □	FLTP 5 1 to FLTP 5 117 all the the mouse to the 3D window of ddle button to confirm the sel V Weaky Veraky Veraky	ecomponents on the right and lection of all components	Ë	A CONTRACTOR	▼ Search Search	
C INNE SEE	FLTP 5 1 to FLTP 5 117 all the the mouse to the 3D window of ddle button to confirm the sel V Weaky Veraky Veraky	ecomponents on the right and lection of all components	Ê	×	▼ Search Search	
C INPUT SUB	FLTP 5 1 to FLTP 5 117 all the the mouse to the 3D window of ddle button to confirm the sel V Weaky Veraky Veraky	ecomponents on the right and lection of all components	Ë	×	¥ Search Search > He Ista ¥ Treaseer	·[0] \$
□ □	FLTP 5 1 to FLTP 5 117 all the the mouse to the 3D window of ddle button to confirm the sel V Weaky Vecay	ecomponents on the right and lection of all components	Ë	N N N N N N N N N N N N N N N N N N N	▼ Search Search	• [0] •
C □ □ □ □ □ □ □ □ □ □ □ □ □ □ □ □ □ □ □	FLTP 5 1 to FLTP 5 117 all the the mouse to the 3D window of ddle button to confirm the sel V Weaky Vecay	ecomponents on the right and lection of all components	Ë	×	¥ Search Search > He Ista ¥ Treaseer	• [0] •
○ ○	FLTP 5 1 to FLTP 5 117 all the the mouse to the 3D window of ddle button to confirm the sel V Weaky Vecay	ecomponents on the right and lection of all components	E	W rv	¥ Search Search > He Ista ¥ Treaseer	• [0] •
C □□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□	FLTP 5 1 to FLTP 5 117 all the the mouse to the 3D window of ddle button to confirm the sel V Weaky Vecay	ecomponents on the right and lection of all components	E	×	¥ Search Search > He Ista ¥ Treaseer	• [0] •
○ ○	FLTP 5 1 to FLTP 5 117 all the the mouse to the 3D window of ddle button to confirm the sel V Weaky Vecay	ecomponents on the right and lection of all components	E	A A A A A A A A A A A A A A A A A A A	¥ Search Search > He Ista ¥ Treaseer	• [0] •
C CHURD SAB Select from 1 C CHURD SAB Select from 1 C CHURD SAB Then move f C CHURD SAB press the mi C CHURD SA	FLTP 5 1 to FLTP 5 117 all the the mouse to the 3D window of ddle button to confirm the sel V Weaky Vecay	ecomponents on the right and lection of all components	E	×	¥ Search Search > He Ista ¥ Treaseer	• • • ↓ ■
○ □<	FLTP 5 1 to FLTP 5 117 all the the mouse to the 3D window of ddle button to confirm the sel V Weaky Vecay	ecomponents on the right and lection of all components	E.	×	¥ Search Search > He Ista ¥ Treaseer	• [0] •
C CHURD SAB Select from 1 C CHURD SAB Select from 1 C CHURD SAB Then move f C CHURD SAB press the mi C CHURD SA	FLTP 5 1 to FLTP 5 117 all the the mouse to the 3D window of ddle button to confirm the sel V Weaky Vecay	ecomponents on the right and lection of all components	E	×	¥ Search Search > He Ista ¥ Treaseer	• [0] •

Figure 7-11: Array Elements

 Review and refine the details of each array element as necessary. (Figure 7-12)

🐲 Motion array	unit		?	×
 Basic Element 				
Basic element	FLTP 5_7			+
Base coordinate	CSYS1			+
 Array collection 	n			
FLTP 5_1				1
FLTP 5_2			=	
FLTP 5_3				
FLTP 5_4				-
FLTP 5_5				~
FLTP 5_6				X
FLTP 5_7				
FLTP 5_8			-	
 Velocity 				
Velocity	100		mm/	s *
▼ Name				
motionarryunit				
motionarryunit				
	-			
		OK A	oply C	ancel

Figure 7-12: motion array Unit

• Click "OK" to finalize the creation of the motion array unit.

 Wait for the generation of the circular conveyor belt and all rigid and collision bodies on the conveyor belt under the basic mechatronic objects. (Figure 7-13)

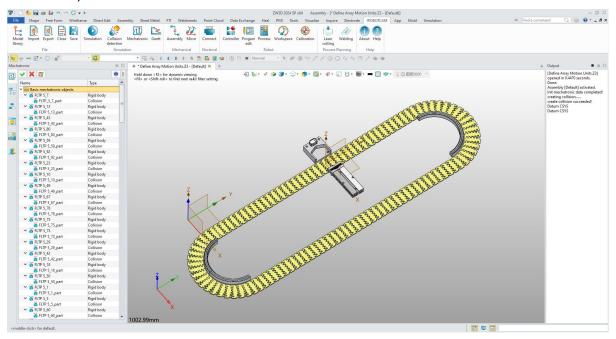


Figure 7-13: Array Elements

Step 5: Add Signal Adapter

V

 Right-click within the "Signal Adapter" panel and choose "Add Signal Adapter". (Figure 7-14)

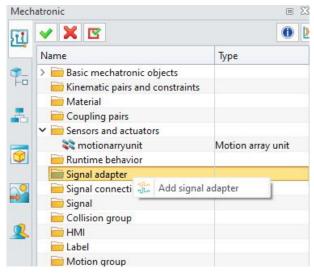


Figure 7-14: Add Signal Adapter

• Click the "+" button within the "Signal" section and select the newly created conveyor belt.

• Specify the desired signals, formulas, and names to define the operational behavior of the conveyor belt. (Figure 7-15)

Image: Second	
Norme Data type in/Out Initial value +	
Kochutoria K	■ m XX
Define Array Mc	ion Units.Z31
Image: State	econds.
Image: Spant Sector	data completed!
Motion group	
▼ formula	
Asign as Formula Note +	
Termula Ter	¥08*
OK Apply Canel	
IUU2.99mm	

Figure 7-15: Signal Adapter Information

• Click "OK" to generate the corresponding signal adapter and signal objects.

Step 6: Add simulation sequences

• Right-click within the "runtime behavior" panel and choose "Add Simulation Sequence" to open the "Simulation Sequence" window. (Figure 7-16)

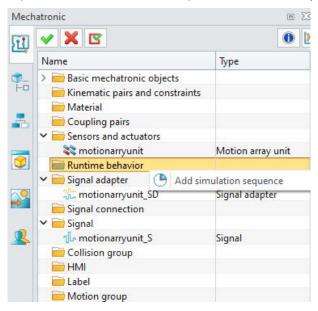


Figure 7-16: Add simulation sequences

- Click Define the "Start time" (eg, 0) at which the simulation should commence.
- Set the "Duration" (eg, 999999) to ensure the motion array unit operates for the desired duration.
- Uncheck the "Activation Signal" and "End Signal" options to initiate the simulation automatically.
- Specify the "Execution signal" to link the simulation sequence to the previously defined motion array unit.
- Set the "End Position" to "1" to ensure the motion array unit completes its full rotation.
- Assign a descriptive "Name" for the simulation sequence. (Figure 7-17)

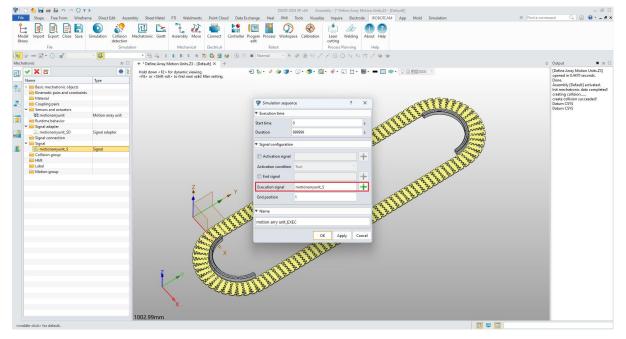


Figure 7-17: T1_EXEC

Step 7: Starting the Simulation

 Click the "Simulation" button located under the "IROBOTCAM" tab. (Figure 7-18)

1 D	6 🔒	iii 12	n 0	0 -	•										ZW3D 2024	SP x64	Assembly - [*	Define Arra	y Motio	on Units.Z3 - [De	fault]]		
File	Shape	Free F	orm	Wirefra	me Direct	Edit Asse	embly Shee	t Metal	FTI W	eldments	Point Clou	d Data E	change	Heal	PMI	Tools Visu	alize Inqu	uire Elect	trode	IROBOTCAM	Арр	Mold	Simulation
Model library	Import	Export				Collision detection	Mechatronic				Connect				Workspac	e Calibratic	n Laser cuttin	Welding	g Ab	out Help			
		File				Simula	ation		Mech	anical	Electrical			Robo	t		Proce	ss Planning		Help			

Figure 7-18: Simulation Button

• Click the "Start Simulation" button within the "Simulation Control" window to initiate the simulated motion of the conveyor belt. (Figure 7-19)

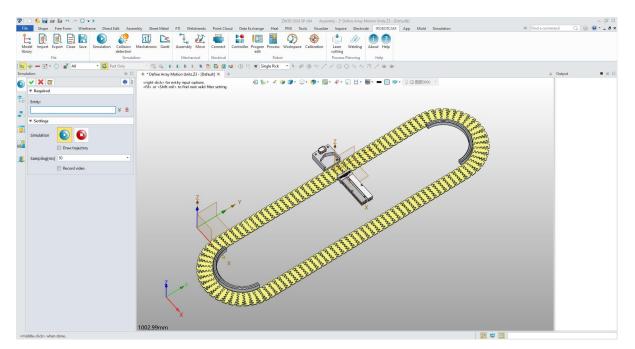


Figure 7-19: Simulation Verification

Conclusion:

Y

By following this detailed and professional guide, you can effectively define and utilize motion array units within the iRobotCAM environment to simulate complex motion array unit systems with accuracy and precision. Adapt and modify the provided parameters and settings to achieve your specific desired behavior and explore the advanced functionalities offered by iRobotCAM for further customization and control.

Chapter 8: Create Robot

This document provides a step-by-step guide on how to create a robot in the IROBOTCAM software.(Figure 8-1)

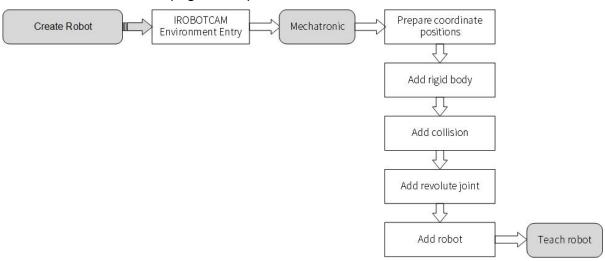


Figure 8-1: Flow chart of Create Robot

Step 1: Open iRobotCAM project file

Y

• Open iRobotCAM project file "Create Robot.Z3". (Figure 8-2)

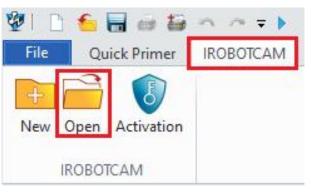


Figure 8-2: Open iRobotCAM project file

• View the details of interface components. (Figure 8-4)

Let Image: Part Control of Contro of Contro of Control of Control of Control of Contr	
Model Impart Exped Case Sam Simulation Callation Mechanica Gata Assembly Move Control Controller Pogan Process Workpace Calibration Laser Help Catterny File Simulation Catterny Move Control Controller Pogan Process Workpace Calibration Laser Help Catterny Help Process Raming Hel	8) opened in 0.2330 ult] activated.
Ranger deckon Mechanical Decisi al Robert outing Imager Imager<	8) opened in 0.2330 ult] activated.
Image:	8) opened in 0.2330 ult] activated.
Manager III + Ceste Robert23 - [Onfund] X + IIII Ceste Robert23 - [Onfund] X + IIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIII	8) opened in 0.2330 ult] activated.
Snew All Image:	8) opened in 0.2330 ult] activated.
	ult] activated.
Image: Second secon	It] activated.
Assembly (Default C S C-Links C S C S C S C S C S C S C S C S C S C S	It] activated.
C 2 C-Date C 2 C-Dat	
	data completed!
92	
19-21	
FA	
Z	
2364.52mm	
2309.52mm	

Figure 8-4: Component Details

Step 2: Prepare coordinate positions

V

• Select the "Datum CSYS" option in the "Shape" tab. (Figure 8-5)

C	9	3	1	7						6		3	\$	9		0	V		000	• •	11	11	11	Y x
ketch	Block						Chamfer						Face		Add	Divide	Simplify	Inlay	Pattern	Mirror	Move	Сору	Scale	Datum
-	-			+	-	*		-	-	-	-		Offset +		Shape +	-		-	Geometry +	Geometry +	-			CSYS +
		Basic S	Shape					Engi	neerin	g Featu	re			Ed	lit Shape			5		Basic Ed	iting			Datum

• Set "Datum CSYS" to "Dynamic". (Figure 8-6)

Datur	m CSYS		Ξ Σ
1	🗸 🗶 🖪		0
3	▼ Required		
	L? L<	₩ Ł 🖌 🏴	2
	Position		👻 🔹 ד
	X axis		🗧 💆 ד
1	Y axis		🗧 💆 ד
	Z axis		🗧 💆 ד
2	▼ Datum Attribu	ites	
	Custom attrik	outes	
<u>&</u>	Color		
	Style		~
	Width		*
	Datum format	Default	•
	▼ Settings		
	Auto orientat	ion	

Figure 8-6: Dynamic

• Hide or uncheck "component Link1". (Figure 8-7)

Man	ager	83
9 -	Show All 🔹	Y 😇
Fo	Assembly Node	
	✓ Z Default	
5	🔽 🌍 (-)Base	
-	✓ Ka (−)Link1	
9	🗹 🇊 (–)Link2	
-	🗹 🟪 (-)Link3	
	🔽 🟪 (-)Link4	
	✓ ¼ (−)Link5	
	🗹 🇊 (–)Linkб	
2		

Figure 8-7: Uncheck Link1

- Create robot coordinate systems:
- ♦ Link1 to Link6 joint rotation axis coordinate system.
- ♦ Base coordinates.
- \diamond Flange coordinates.
- Place and adjust the coordinate systems correctly, except for base and flange coordinates, which need only their Z-axis aligned. (Figure 8-8)

V

0 0 0 0 0 0 0 0 0 0

· Q &

er of curvature

of curvature

⁶ Repeted	ा छि हि
Weiter and de clabe was assessed. Weiter and de clabe was assessed. Weiter and de clabe was assessed. Weiter clase was assessed. We	
Trepton	
Notice Click here to quickly adjust the direction of the coordinate system, Image: Click here to quickly adjust the direction of the coordinate system, Image: Click here to quickly adjust the direction of the coordinate system, Image: Click here to quickly adjust the direction of the coordinate system, Image: Click here to quickly adjust the direction of the coordinate system, Image: Click here to quickly adjust the direction of the coordinate system, Image: Click here to quickly adjust the direction of the coordinate system, Image: Click here to quickly adjust the direction of the coordinate system, Image: Click here to quickly adjust the direction of the coordinate system, Image: Click here to quickly adjust the direction of the coordinate system, Image: Click here to quickly adjust the direction of the coordinate system, Image: Click here to quickly adjust the direction of the coordinate system, Image: Click here to quickly adjust the direction of the coordinate system, Image: Click here to quickly adjust the direction of the coordinate system, Image: Click here to quickly adjust the direction of the coordinate system, Image: Click here to quickly adjust the direction of the coordinate system, Image: Click here to quickly adjust the direction of the coordinate system, Image: Click here to quickly adjust the direction of the coordinate system, Image: Click here to quickly adjust the direction of the coordinate system, Image: Click here to quickly adjust the direction of the coordinate system, Image: Click here to quickly adjust the direction of the coordinate system, Image: Click here to quickly adjust to quickly adjust to quickly adjust to quickly adjust	nbly [R820洪载体-外发20150610] activated. n CSYS
 Item 100 Item 400 Item 400<	rowser (B) 2
Chick here to quickly adjust the direction of the coordinate system, which is relative to the triple axis coordinate system.	+ + + 🗢 🖕
Vers 0.10 0<	Favorites
	Windows(C:)
Cution attributes Color attributes Shyle With Datument V State Auto orientation Field Fi	DAIA1(D:)
S Cafe Syle Won Denne front Ustud Y Setting	
Syle	
Wath Deam fromat: Lefant	
Detum format Defuz	
Defaunt from the Default → V Setting Z Auto interstation V Setting Z Auto interstation V Setting V Se	
▼ Setting:	
V Seed Combined of the last of	
Firsthamer Andrew Andre	
Firsthamer Andrew Andre	
Firstham P First	
Þ field	arch
- X+ X P from	n • Q 🗞
- X+ X P from	
×	
triple axis coordinate system	
2688.95mm	
nddle-dicky to finish.	

Figure 8-8: Coordinate system construction

- Create the base coordinate (CSYS8) using "Critical" and selecting the central triple axis coordinate system.
- Create the flange coordinate (CSYS7) at the desired position and direction on Link6.

(Figure 8-9) B Sweep Loft Filet Chamfer Dreft Hole Rib Thread Lip Stock Face Shell 1 Hold down <F2> for dynamic viewing. <F8> or <Shift-roll> to find next valid filter setting. - 3 0 NUMBER ALSO 20150 5 my 動法CSVS 10 2 Other coordinate system determination methods: The coordinate system can be regarded as a right-hand rule, where the Z-axis represents the thumb, which is the robot's rotation as and the X-axis represents the other four fingers, which are the robot's rotation direction. ▼ Search The XYZ axis of the b - x - X / © × 🔳 2737.7

Figure 8-9: Coordinate system position

Step 3: Create rigid bodies and collisions

V

• Go to the "IROBOTCAM" tab add click the "Mechatronic" button. (Figure 8-10)

1	6 🔒		n 0	0 =	•										ZW	3D 2024 SP x 64	Assemb	ly - [* Create	Robot.Z3 - [Default]]		
File	Shape	Free F	orm	Wirefra	me Direct	Edit Ass	embly Sheet	Metal	FTI V	Veldments	Point Clo	ud Data I	Exchange	Heal	PMI	Tools Visua	lize Inqu	re Electro	de IROBOTCAM	Арр	Mold	Simulation
Model library							Mechatronic											Melding	1 ? About Help			
		File				Simul	ation		Mec	hanical	Electrical			Robo	t		Proces	s Planning	Help			

Figure 8-10: Mechatronic

- In the "Mechatronic" window, click on "Basic Mechatronic Objects".
- Right-click and select "Add Rigid Body" for each of the seven joint component objects. (Figure 8-11)

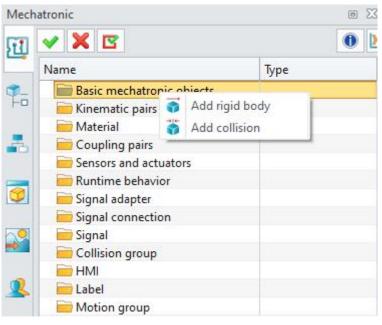


Figure 8-11: Add Rigid Body

• Property settings to 'Unchecked', and material to 'Default'. (Figure 8-12)

Y

		+
	•	
OK	Analy	Cancel
	OK	

Figure 8-12: Rigid body parameters

 After adding rigid bodies to the seven joint objects, the component details will be displayed. (Figure 8-13)

Mech	atronic	8
1	✓ X	0
	Name	Туре
9	Basic mechatronic objects	
FO	👩 Base	Rigid body
_	👩 Link1_1	Rigid body
÷.	👩 Link2	Rigid body
	👩 Link3_1	Rigid body
	👩 Link4_1	Rigid body
0	🕤 Link5_1	Rigid body
	👩 Link6	Rigid body
2	Kinematic pairs and constraints	
	🚞 Material	
_	Coupling pairs	
<u>R</u>	E Sensors and actuators	
	📄 Runtime behavior	
	📄 Signal adapter	
	Signal connection	
	🧮 Signal	
	Collision group	
	HMI 🔁	
	🧰 Label	

Figure 8-13: seven rigid bodies

- Right-click and select "Add collision" for each of the seven component objects.
- Select "convex hull" for the collision shape.

V

• Property settings to "unchecked", and material to "default". (Figure 8-14)

🖗 Collision			?	×
 Collision O 	bject			
Base				+
▼ Shape				
Collision shap	pe	Convex decomposition		•
Material	Default	•		
Property se	ttings			
 Physical p Collision n Support so 	response			
▼ Label				
None				•
		OK Appl		Cancel
		OK Apply		Cancel

Figure 8-14: Collision Parameters

• After adding seven objects to the collisions, the component details will be displayed. (Figure 8-15)

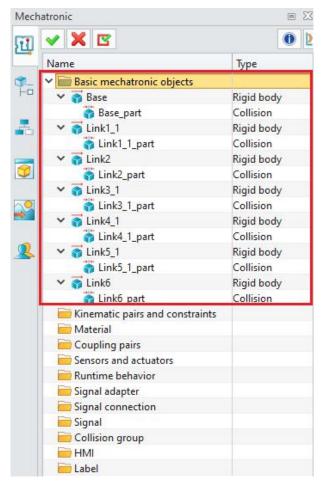


Figure 8-15: Seven Collisions

Step 4: Create joints

V

• In the "Mechatronic" window, click "Kinematic Pairs and Constraints". Rightclick and select "Add Revolute Joint". (Figure 8-16) Y

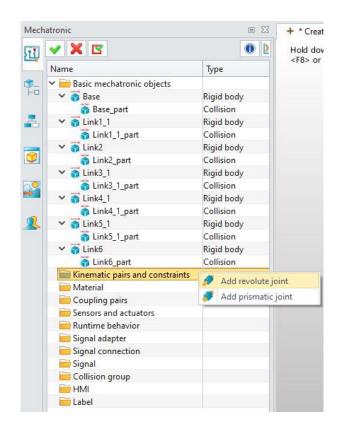


Figure 8-16: Add Revolute Pairs

 Create six revolute joints for each robot joint, specifying parameters for each. (Figure 8-17)

× 🗶 🖾	1	🕴 😰 <right-dick> for context-sensitive options. 😔 🥸 y 🤣 🍞 - 🗇 + 🚳 - 🖗 + 🗔 🖯 + 👼 - 🔲 🧶 -</right-dick>	2 ③ 图层0000 *
Name	Туре	V Revolute joint ? X	
👻 🔤 Basic mechatronic c	bjects	V Connection relation	
Basic mechatronic c Base	Rigid body	delect exect code	
Base_part	Collision Rigid body		
Link1_1_part	Collision	Base_part 🕂	
	Rigid body	Select child node X X	
V 👩 Link2	Collision		
✓ 👩 Link3_1	Rigid body		
Link3_1_part	Collision	×	
- D FUINA	Rigid body		
Link4_1_part	Collision		
V 👩 Link5_1	Rigid body		
G Link5_1_part	Collision		
 Čink6 Čink6_part 	Rigid body Collision	▼ Axis and angle	
Kinematic pairs and		Specify axis vector X: -0.000 Y: 0.000 Z: 244.000	Z
Material	consolaines	Specify axis vector X: -0.000 Y: 0.000 Z: 244.000	
Coupling pairs		Start angle 0	
Sensors and actuato	rs		
Runtime behavior		V Limitations	
📄 Signal adapter		Upper limit 170 - Z	
Signal connection			
Signal		Cover limit -170	
Collision group		▼ Settings	
HMI		▼ Settings X	
Label		Motion type Kinematics •	
		V Name	
		Robot J1	
		OK Apply Cancel	
		× ×	

Figure 8-17: Robot_J1

• Create the six joints of the robot in sequence, with the following parameters:

- ♦ Child node rigid body and parent node collision.
- ♦ Specified axis vector (Datum CSYS1-6).
- ♦ Upper and lower limits of the joint.
- \diamond Start angle ("0").
- ♦ Joint type ("Kinematic").
- ♦ Name (Robot_J1-6).
- Six joint hinge sub interface details. (Figure 8-18)

Kinematic pairs and constraints Robot J1	Revolute joint
🖉 Robot_J2	Revolute joint
🖉 Robot_J3	Revolute joint
🝠 Robot_J4	Revolute joint
🝠 Robot_J5	Revolute joint
🝠 Robot J6	Revolute joint

Figure 8-18: Six Revolute Joints

Step 5: Create robot

 In the "Mechatronic" window, click "Sensors and actuators", right-click and select "Add Robot". (Figure 8-19)

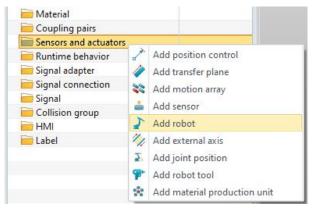


Figure 8-19: Add Robot

- Enter the robot name "GSK".
- Add joints in order with "positive" rotation direction and "0, 0, -90, 0, 0, 0" offset values.
- Choose "GSK_Base" for "Base Coordinate CSYS8" and "GSK_Flange" for "Flange Coordinate CSYS7". (Figure 8-20)

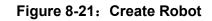
Ś

hatronic	8 23	+ * Create Robot.23 - [RB20装数体-外发20150610] × +			Output	
🗸 🗶 🖪	0 P	Hold down < F2> for dynamic viewing.	1 · 🚳 · 🚸 · 📑 🗄 · 🚍 · 💻 🗐 🧇 · 🖓 🖓 🕾 📰 🕬 ·		Toggle Suppress of history of Toggle Suppress of history of	perations
Name	Type	<f8> or <shift-roll> to find next valid filter setting.</shift-roll></f8>		2	Buppress of history of	perations
	type		😵 Create robot	1	× Suppress of history of	perations
 Basic mechatronic objects Base 	Rigid body	×	Assembly robot Robot pa		suppress of history of	perations
Base part	Collision	4	A	IT AFTIGLIGT	wser	
V G Link1_1	Rigid body		Robot Name GSK		+ + 0	
Link1_1_part	Collision	×	▼ Joint			
v o Link2	Rigid body		* Joint		evorites	
Link2_part	Collision	4	Select Delete		findows(C:)	
V G Link3_1	Rigid body				ATA1(D:)	
a Link3_1_part	Collision		Name Dir	ection Offset value		
Y O Link4_1	Rigid body		1 Robot J1 positiv	e - 0		
Link4 1 part	Collision		stration))) Z			
Y in Link5_1	Rigid body		2 Robot_J2 positiv	e - 0		
Link5 1 part	Collision	0	3 Robot J3 positiv	90		
✓ in Link€	Rigid body		3 KODOQIS POSIDA	e90		
G Link6_part	Collision		4 Robot J4 positiv	e • 0		
Y - Kinematic pairs and constrain	ts					5 × · · • •
Fobot J1	Revolute joint	10-0	5 Robot_J5 positiv	e • 0		D× · Am
S Robot 12	Revolute joint	2	6 Robot J6 positiv			
🛷 Robot_13	Revolute joint					
Robot_14	Revolute joint					
🝠 Robot_J5	Revolute joint	19	^			
S Robet_16	Revolute joint					
Material		000	Joint coupling			
Coupling pairs		7	Coupling coefficient -1		ch	
Sensors and actuators		2				· Q *
Runtime behavior		0	V Base coordinate		-	
Signal adapter					ist	
Signal connection			CSYS8		+	
🧱 Signal		5 m			new	
Collision group			Name GSK_Base			
HMI						
E Label		7	Flange coordinate			
		× ×	CSY56		+	
			Name GSK_Flange			
				OK Apply Ca	incel	

Figure 8-20: Assembly Robot

• Robot parameter settings, select "analytical solution" for the algorithm and confirm other default options. Click OK to create the robot. (Figure 8-21)

Assembly robot	Robot parameter	
Algorithm		
0 Geo 0 N	um o Ana	
Parameters		
	Z_{0} $E = O_{0} \uparrow X_{0}$ X_{0} Z_{0} $\downarrow E$	- Q.
		4
		4 +->
		11
		<u>140</u> .
		-
	00 X0 Y0 00	
a1	190	mm
a2	-192	mm
b	0	mm
c1	585	mm
c2	650	mm
в	750	mm
c4	131.9	mm
Others		
Quick change	Not required 🔻	



• Wait for the created robot to appear in the Mechatronic on the left, and the robot will be successfully created. (Figure 8-22)

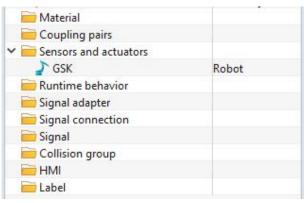


Figure 8-22: Robot creation completed

Step 6: Measure Robot parameters

• Click "Inquire" and select "Distance". (Figure 8-23)



Figure 8-23: Distance Option

- Measure the lengths of a1, a2, b, c1, c2, c3, and c4 according to the robot parameter diagram.
- Enter the corresponding values in the "Add Robot" window.

(Figure 8-24)

Measure Distance Angle Arc Radius Length Area 3D Surt Measure	Info - Info - Continuity Faces - Curvature Continuity Analysis Draft Analysis Analysis Inspect Entities	Curvature Properties Statistics Parts Check Files Inspect Model	Find a command Find a comman
Image: Second	Cit Cancel	Click to select Center of Curvature	tites
efer good.	220.53mm	e2 e3 e4 Others Gainte change Ther	5550000 mm 6600000 mm 7500000 mm 13199944 mm required * OK Apply Cancel

Figure 8-24: Robot parameter measurement

Step 7: Teach robot

• Right-click on the created robot and select "Teach Robot". (Figure 8-25)

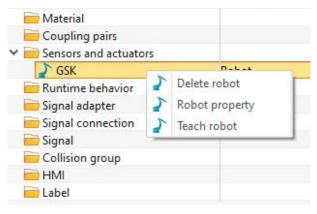


Figure 8-25: Teach Robot

• Click "Drag Teach" and Drag the XYZ axis to check if the robot is moving normally. (Figure 8-26)

W

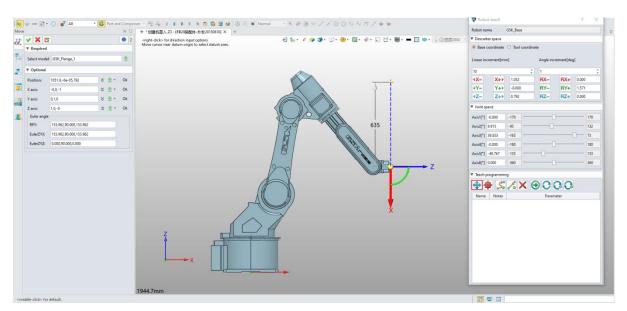


Figure 8-26: Drag Teach

W

Chapter 9: Teach Program Draw Arc

Draw Arc: A Comprehensive and Informative Guide with Advanced Techniques

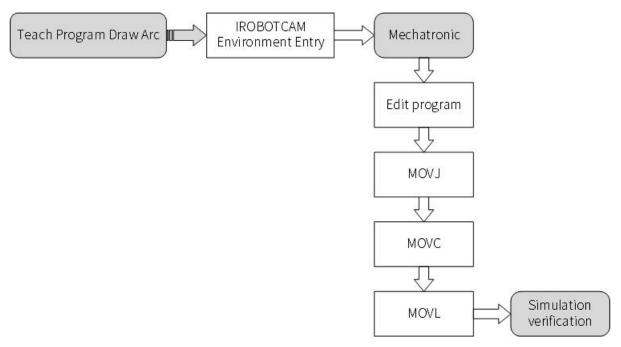


Figure 9-1: Flow chart of teach program-draw arc

Step 1: Opening and Activating the Assembly

• Entering iRobotCAM: Launch the iRobotCAM software by opening the file "Draw Arc.Z3". This will initiate the iRobotCAM environment, providing you with the necessary tools for robot programming and simulation.

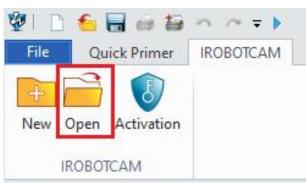


Figure 9-2: Open iRobotCAM project file

• Open interface details.

V

는 네 네 글 등 💿 💸 🖽 🕒 구 22 🗰 🔚 💷 🚇 🕥 🐵 🌒 🕗	
Model Import Esport Close Simulation Calification Meshanization Connect Connect	
v + → 腔 + ○ m = All	
	- Output
	Colption Colption Conservation Conservat
1054.31mm	
elect command or entity.	

Figure 9-4: Component Details

Step 2: Entering Teach Mode and Adding Marker Points

• Enter Mechatronic: Navigate to the "Mechatronic" option within the iRobotCAM interface.(Figure 9-5)

1.110-14		ii ii -														V3D 2024 SP x6		ibly - [Didw	Arc.Z3 - [Default]]			
File	Shape	Free Fo	rm	Wirefran	ne Direct I	Edit Asse	embly Sheet	t Metal	FTI We	Idments	Point Clou	id Data E	xchange	Heal	PMI T	ools Visual	ze Inquir	e Electro	de IROBOTCAM	Арр	Mold	Simulatio
Model library	Import		Close		Simulation	Collision detection	Mechatronic	Gantt	Assembly		Connect	Controller		Process	Workspace	Calibration	Laser cutting	Melding	About Help			
		File				Simula	ation		Mecha	nical	Electrical			Robot	t -		Process	Planning	Help			



• Right Initiating Teach Mode: Right-click on the robot and select "Teach Robot" from the context menu. This will activate the Teach Mode, allowing you to record robot movements and program its trajectory. (Figure 9-6)

	2000 2019 446 Assembly : "Down Arc 22 - Ded Add Stope: Free form: Workname: Detect Eds Assembly : Sheet Maria F1 Widdensets Point Cloud Data Exchange Head PMI Tools Vousie Inquire Electroide Root Adap Moid Simulati Windows: Detect Eds Assembly : Sheet Maria F1 Widdensets Point Cloud Data Exchange Head PMI Tools Vousie Inquire Electroide Root Adap Moid Simulati Windows: Detect Eds Assembly: Sheet Maria F1 Widdensets Point Cloud Data Exchange Head PMI Tools Vousie Inquire Electroide Root Adap Moid Simulati Windows: Detect Eds Assembly: Move Connect Conforter Program Process Workspace Calabration electroide Program Process Workspace Calabration electroide Program Process Workspace Calabration	
		🖉 Robot teach ? ×
<pre>* depined Set result Partie Pa</pre>		 Descartes space
	Move cursor pear datum origin to select datum axes.	Base coordinate Tool coordinate
Priced Price de los 0.000,100,000,1000 Bure do 0.000,000,1000 Bure do 0.000,000,000 Bure do 0.000,000,000,000 Bure do 0.000,000,000,000,000 Bure do 0.000,000,000,000,000,000,00	Select model: [FB_120_flange_1 3	
Patient 10.038 V V 0 Xaiti 10 V 0 Vaiti 10.030,01 V 0 Martin 10.000 0 Martin V 0 Martin 0 0	r Optional	
	Xaxis: -100.00035 🛛 😤 - Ok	
	Yaxis: -0,1.0 🛛 🕉 💁 - Ok	V Joint space
	Z axis: 0.000350,-1 🛛 🛛 🕉 - Ok	
BP1 10000.000.000.000 Buezon 10000.000.000 Buezon 000110000 Buezon 2 V 2	Euler angle	
	RPY: 180.000,0.220-180.000	
	EulerZYX: -180.000,0.02180.000	
Image: Solution of the solutio	Euler2Y2: 0.000,180.000-0.000	
Tech pograming		
		HAURO 400
X Nene Nete Permetr	z	
	×	
		Name Notes Parameter
	839.151mm	

Figure 9-6: Teach Robot

Step 3: Add marker points

Y

- Record Joint Point: Click on "Mark point" in the "Robot Teach" window.
- Select the CSYS1 reference coordinate system.
- Select the suitable item in the pop-up box, the robot will move to that position, and then click the "Record Joint point" button. (Figure 9-7)

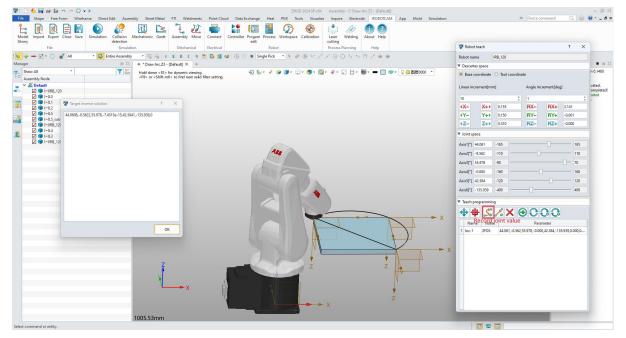


Figure 9-7: Add Marker Points

Y

• Record End Point: Click on "Mark Point", Select "CSYS2" and "CSYS3", and click on the "Record End point" button to add a record. (Figure 9-8)

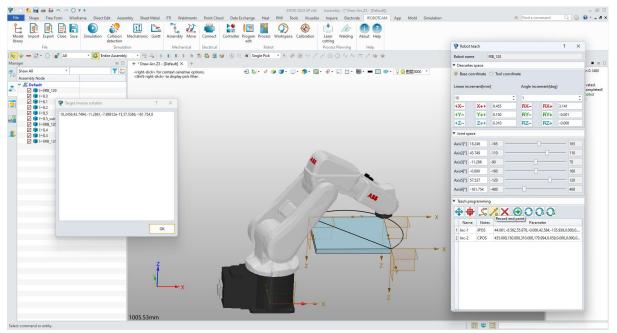


Figure 9-8: Add markings for points 2 and 3

- Arrow Record Point: Click loc-1 in the "Robot Teach" window.
- click the "arrow" to move to the marked point.
- move the robot to the position of point 1 again, click the "Record End point" button to mark, add Record point 4, and finally obtain 4 record point.
- In subsequent program editing, the same point cannot have both MOVJ and MOVL instructions, so add different Record point to the same position to avoid program instruction conflicts. (Figure 9-9)

Y

Rob	ot <mark>teach</mark>				?	×
Robot na	me	IRB_120				
Descar	tes space					
Base	coordinate	O Tool co	ordinate			
Linear in	crement[m	m]	Angle in	crement[deg]	
10			\$ 1			÷
«X –	X++	0.155	RX-	RX+	3.141	
4Y-	Y+>	0.150	RY-	RY+	-0.001	
4Z-	Z+>	0.310	RZ-	RZ+	-0.000	
Joint s	pace					
Axis1["]	44.061	-165 -				165
Axis2[°]	-8.562	-110 -	[}		110
Axis3[*]	55.978	-90 -			-0 [70
Axis4[°]	-0.000	-160 -				160
Axis5[°]	42.584	-120 -				120
Axis6[°]	-135.939	-400 -				400
leach	programmi	-				
♣ +	₽ ~~			33		
Nar	me Not	es	Par	ameter		
1 loc-1		14.0608,	, -8.5622, 55.9780	, -0.0000, 42.	5841, -13	5.93
2 loc-2	18		, 43.7494, -11.286			
3 loc-3		-	9, 43.7494, -11.28			
	13	44.0608	, -8.5622, 55.9780	, -0.0000, 42.	5841, -13	5.93

Figure 9-9: Marking of point 4

• Go Home Point: click Move to Home button to bring the robot to its original position. (Figure 9-10)

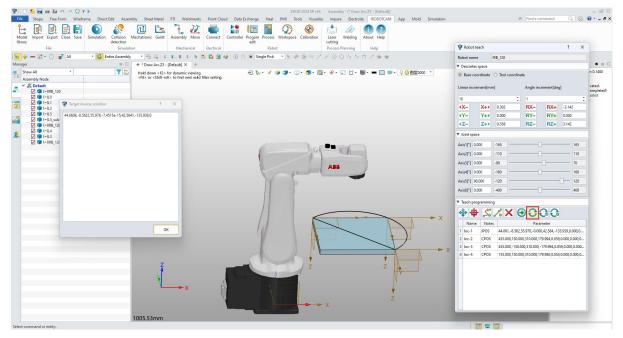


Figure 9-10: Move to Home

Step 4: Program Editing

(Y)

• On the "IROBOTCAM", click the "Program Edit" button. (Figure 9-11)

1 🗅	6 🖬	in 12	n 14	() -)	K										ZWS	3D 2024 SP x64	Assem	oly - [* Draw /	Arc.Z3 - [Default]]			
File	Shape	Free Fo	orm	Wirefram	e Direct	Edit Asse	mbly She	et Metal	FTI V	Veldments	Point Clou	ud Data	Exchange	Heal	PMI To	ols Visualize	Inquir	e Electrod	e IROBOTCAM	Арр	Mold	Simulation
£:	M	SE .				8	E	1	٦.	22			1			۲	1	<u>An</u>				
Model library	Import	Export	Close	Save S	Simulation	Collision detection	Mechatroni	c Gantt	Assemb	ly Move	Connect	Controlle	Progam edit	Process	Workspace	Calibration	Laser cutting	Welding	About Help			
		File				Simula	ition		Mec	hanical	Electrical			Robo	ot		Process	Planning	Help			

Figure 9-11: Program Edit

 In Edit Selected Program: In the "Program Edit" window, right-click on "program1" and select "Edit Program" to open the program parameter interface. (Figure 9-12)

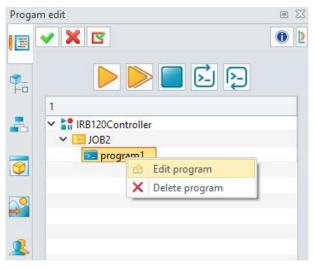


Figure 9-12: Edit Program

• Add MOVJ Command: In the "Program Parameters" window, select the "MOVJ" in the Motion command, add the MOVJ instruction, and set the parameters as shown in the figure. The target point is loc-1, and other parameters remain default. Click OK. (Figure 9-13)

Y

	-dick> for entity input options. 🗧 🐟 • 🖋 🦻 🍞 • 🗇 • 🚳 • 🚳 • 👘 • 🛄 • 🐳 • 🛄 🖯 • 👼 • 🔲	Program parameters Robot IRB_120	C 23 Sifully.
🕨 📄 🖬 🔁		 Contraction Contraction 	
1		▼ Required	
V RB120Controller			
V JOB2	W MOVJ	7 × Name Ty	pe Data
		1 loc-1 JPOS	0.155,0.15,0.31,179.99,0.054326,0,0,0,
	Target: loc-1 *	2 loc-2 CPOS	0.455,0.15,0.31,179.99,0.058922,0,0,0,
	VEL: 200 */s	3 loc-3 CPOS	0.455,-0.15,0.31,-179.99,0.058926,0,0,
	ACC: 100 %	4 loc-4 CP05	0.155,0.15,0.31,179.99,0.054326,0.0,0,
	JERK: 100 %		
		▼ Command	
		System Motion Co	atrol 10
) <mark>(</mark> \$
	G		
		Command	Paramter
	OK Apply Cencel		
	OK Apply Cancel		
			· 0
		•	
	Z Z		
	-		
	j 🛊		
	↓		

Figure 9-13: MOVJ

• Add MOVA Command: Then add the MOVA instruction, with parameter settings as shown in the figure. The target point is "loc-3", the middle point is "loc-2", and other parameters remain default. Click OK. (Figure 9-14)

vedit ✓ X ⊡	0 2	+ * Draw Arc.Z3 - [Draw Arc] × + <ight-dick> for entity input options. <fb> or <5hft-roll> to find next valid filter setting.</fb></ight-dick>	€ \$• * \$ \$. 0 • \$• ₫• 8• . 5	• • • • • • • • • • • • • • • • • • •	Program para	meters	C 23	ssfully.
🔁 📕 🍕								
1 1 1 1 1 1 1 1 1 1 1 1 1 1			VIC 05 NOV Hotmodae 0-2 0 ACC 05 N ACC 05 N ACC 05 N ACC 05 N ACC 05 N		▼ Required 1 loc-1 2 loc-2 3 loc-3 4 loc-4 ♥ Command ♥ Command Command	Type JPOS CPOS CPOS CPOS	Data 0.155,0.15,0.0.1,75.99,0.054326,0.0,0,- 0.455,0.15,0.1,75.99,0.054326,0.0,0,- 0.455,0.15,0.1,75.99,0.054326,0.0,- 0.150,0.15,0.1,75.99,0.054326,0.0,-	• • •
	Ξ	ž x						

Figure 9-14: MOVA

• Add MovL Command: Then add the MOVL instruction, with the parameter settings shown in the figure, the target point being loc-4, and other parameters remaining at default. (Figure 9-15)

edit	0 2	+ * Draw Arc.Z3 - (Draw Arc) × + <pre><right-dick> for entity input options. <fb> or <shift-roll> to find next valid filter setting.</shift-roll></fb></right-dick></pre>	€ &• ≠ ∌ €• ⊙• \$• ₫•	₽• 🖬 번• ■• 🗕 🗋 🗢	- 0 (C) (S)(E0000) -	Program parar	neters	C 8	8.)- ssfuilly.
🕨 🏓 🗖 🔂 🖗						I make a base Second			1
1						▼ Required	1000		
V RB120Controller				and the second se		G C C	C _x		
V JOB2			W MOVL		7 × 1	Name	Type	Data	
			Target: loc-4			1 loc-1	JPOS	0.155,0.15,0.31,179.99,0.054326,0,0,0,	
						2 loc-2	CPOS	0.455,0.15,0.31,179.99,0.058922,0,0,0,	
			VEL: 0.5	m/s		3 loc-3	CPOS	0.455, -0.15, 0.31, -179.99, 0.058926, 0, 0,	
			ACC: 0.05	-* 0		4 loc-4	CPOS	0.155,0.15,0.31,179.99,0.054326,0,0,0,	
			JERK: 0.05	_* //	× ×				
				$ \wedge$		▼ Command			
				G V	0 0	System Mot	ion Control	10	
						C.J.Z	305		
				J			0.0.10	Paramter	
			1		X	Command 1 MOVJ	(loc-1)/el= 200.0	Paramter 00,Acc=100.00,Jerk=100.00)	
			OK Apply	Cancel				=0.50,Acc=0.05,Jerk=0.05)	
			ОК Арру	Cance II-					• Q !
		6							- I'dli
				N.					L
				z					
			Contraction of the local division of the loc						
		7							
		4							
		1							
		× ×							

Figure 9-15: MOVL

Step 5: Simulation Verification

V

• Initiating Simulation: Click on the "Run" button within the "Program Edit" window. This will initiate the simulation process and execute the programmed instructions for the robot. (Figure 9-16)

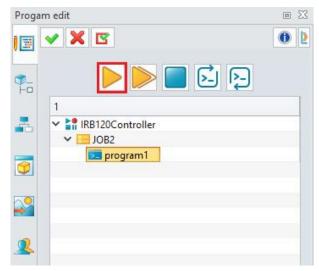


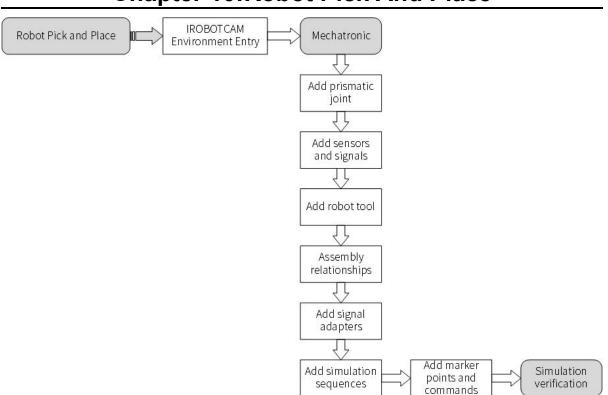
Figure 9-16: Simulation Button

• Analyzing the Motion: Observe the robot's movement during the simulation and compare it to the desired outcome. Pay close attention to the robot's path, speed, and overall behavior. (Figure 9-17)

V

	+ * Draw Arc.23 - [Draw Arc] X +	⊊ Output
	You can set the holdway in Customes Settings. D 🚸 · I 👙 🧊 · ⊘ · 🚳 · 🕼 · I · 🖶 · 📰 🖶 · 💭 🕸 · I · 👹 · 👘 · 🔛 · 👘 · 🔛 · 👘 · 🔛 · I · I · I · I · I · I · I · I · I ·	yupiaceO y retentink by Vo (唐田知道) (周田知道) moved/renamed successfully. Activate an object for editing Done. Assembly [Draw Arc] activated.
		File Browser
RB120Controller		+ + + + \$
JOB2		> traverites > Windows(C:) > DATA1(D:)
	2	▼ Search Search. • Q 1
		► File List
		Preview
	Žx	

Figure 9-17: Simulation Verification



Chapter 10: Robot Pick And Place

Figure 10-1: Flow chart of robot pick and place

Step 1: Open iRobotCAM project file

V

 Open iRobotCAM project file "Robot Pick and Place.Z3" and enter the IROBOTCAM environment. (Figure 10-2)

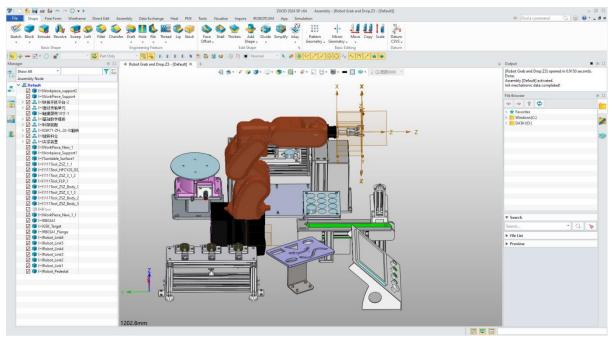


Figure 10-2: Interface details

Step 2: Add Prismatic Joints

V

- With the robot GSK created and its rigid and collision bodies defined, the next step is to add prismatic joints for the robot tool. Here's how:
- ♦ Go to the "IROBOTCAM" tab and click the "Mechatronic" button.
- Right-click the "Kinematic Pairs and Constraints" button and select "Add Prismatic Joint".
- ♦ Define child nodes, select the following rigid bodies:(Figure 10-3)

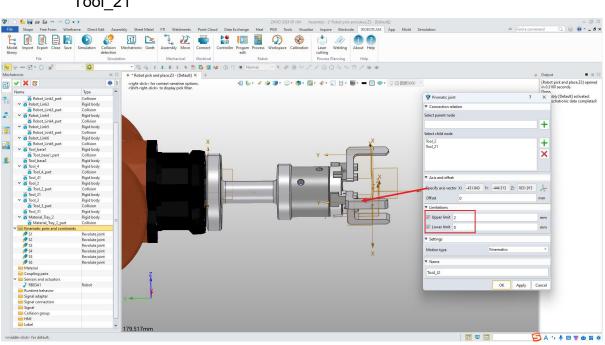


Figure 10-3: W1 Prismatic Joint

- Specify axis vector, click on the "Datum CSYS16" built on the selected rigid bodies to define the axis of movement for the joint.
- Set restrictions, set the "upper limit" to 2 and enter "0" in the "lower limit" field. These values represent the maximum distance the joint can move in the positive and negative directions.
- Choose motion type, select "Kinematics" for this joint.
- Name the joint: Enter a descriptive name for the joint, such as "W1".
- Repeat the above operation steps for the remaining two pairs of claws.
- ☆ "Tool_J3": Child nodes: "Too_3" and "Tool_31", Axis vector: "Datum CSYS18".

Tool_2 Tool_21

- ☆ "Tool_J4": Child nodes: "Tool_4" and "Tool_41", Axis vector: "Datum CSYS13".
- Verify highlighting: After adding the three joints, click on each joint to verify that the corresponding part of the claw is highlighted correctly. (Figure 10-4)

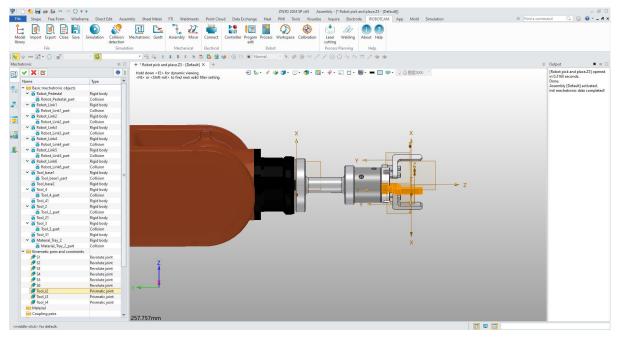


Figure 10-4: Three Prismatic Joints

Step 3: Add sensors

V

- Right-click "Sensors and actuators" button.
- Select "Add Sensors".
- Define sensor dimensions, set the "Length", "Width", and "Height" to 30 for each dimension.
- Specify coordinate system, Click "CSYS8" to refer to the coordinate system of the sensor.
- Name the sensor, enter a descriptive name for the sensor.
- Set the window to default values.
- Click "OK" to add the sensor to the scene. (Figure 10-5)

Shape Free Form Wirefram	 Direct Edit Assembly Si 	Metal FTI Weldments Point Cloud Data Exchange Heal PMI Tools Visualize Inquire Electrode IROBOTCAM App Mold Simulation	🌣 Find a command 🛛 🖓 🔞 • 🗕
	🕑 🔐 🖽	[말 길 값 🛑 🔚 🗏 🖉 🚱 🐵 🖄 🖉 🕐	
I Import Export Close Save	Simulation Collision Mechatro	Gantt Assembly Move Connect Controller Progam Process Workspace Calibration Laser Welding About Help	
Y	detection	edit cutting	
File	Simulation	Mechanical Electrical Robot Process Planning Help	
$r = \boxplus \cdot \bigcirc \mathbf{u}^r $		- 🔒 🗈 🗈 🗉 🐂 🗮 🧕 🍱 🕼 🕕 🛞 円 🔳 Normal 👘 🐂 🦃 🕑 ヤイノ 〇 〇 小 ハ ボノ 🍲 🍉	
ronic		t pick and place.Z3 - (Default) × +	⇒ Output ■
X	<f8></f8>	n < R> for dynamic viewing.	[Robot pick and place.Z3] op in 0.3160 seconds.
lame	Туре 🔶		Done.
Y 👩 Robot Link4	Rigid body		Assembly [Default] activated
Robot_Link4_part	Collision	X	Init mechatronic data comp
Y Robot Link5	Rigid body	•	
Robot Link5 part	Collision		
Y G Robot_Link6	Rigid body		
Robot_Link6_part	Collision	🕸 Sensor ? X	
✓ nool base1	Rigid body	V Type	
Tool_base1_part	Collision	· · · · · · ·	
		Distance sensor	
Tool_base2	Rigid body		
V o Tool_4	Rigid body		
o Tool_4_part	Collision	▼ Property X	
o Tool_41	Rigid body		
🗸 👩 Tool_2	Rigid body	Conson shape scraynt me	
Tool_2_part	Collision		
o Tool_21	Rigid body	Specify coordinate X: -405.342 Y: -444.809 Z: 1831.499	
V G Tool_3	Rigid body		
Tool 3 part	Collision	Length 30 mm	
Tool 31	Rigid body		
 Material_Tray_2 	Rigid body	Height 30 mm	
Material Tray 2 part	Collision	Width 30 mm	
Kinematic pairs and constraints			-> Z
S1	Revolute joint		
2	Revolute joint	V Name	1
3	Revolute joint =		
2 53 2 54		sensor	
	Revolute joint		
Ø \$5	Revolute joint	Bind signal	2
A 56	Revolute joint		
🝠 Tool_J2	Prismatic joint		
ool_J3	Prismatic joint	OK Apply Cancel	
🝠 Tool_J4	Prismatic joint	OK Apply Cancel	
🚞 Material			
Coupling pairs		Z IIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIII	
Sensors and actuators			
RB03A1	Robot		
Runtime behavior			1
Signal adapter	100		
Signal connection	Y		
Signal			
Collision group			
HMI			
E Label	72.66		

Figure 10-5: Impact Sensor

Step 4: Add signals

V

- Right-click "Signal" button.
- select "Add Signal".
- Define signal type, in the "IO type" field, select "Out" to define the signal as an output signal.
- Name the signal, such as "Tool_out".
- Set Port, enter -1 in the "port" field.
- Click "OK". (Figure 10-6)

🐲 Signal	?	×
Settings		
🗹 Connect runtime para	ameters	
Parameter name	Angle	•
IO type	Out	Ŧ
Data type	BOOL	
Dimension	Angle	
Unit	•	
Initial value	false	-
🗸 Name		
Signal name	Tool_out	
Controller	RB03A1Controller	
Port	-1	
	OK Apply	Cancel

Figure 10-6: Add out signal

Step 5: Add robot tool

- Right-click "Sensors and actuators" button.
- Select "Add Robot Tool".
- Select motion pairs, Click on the + button and select the "three prismatic pairs" you created previously.
- Set speed, In the "Speed" field, enter "100" to define the speed of the tool movement.
- Select Rigid Bodies, select the rigid body representing the base part of the robot tool.
- Assign Sensor: In the "Sensing Point" field, select the sensor you created previously.
- Define Base, In the "Base coordinate" field, select the "CSYS9" coordinate system.
- Select the tool center and select the "CSYS8" coordinate system.
- Assign Signal, click the + button next to "Signal" and select the "Tool_out" signal you created previously.
- Name the tool: Enter a descriptive name for the robot tool, such as "Tool".
- Click the "OK" button.(Figure 10-7)

🗋 🌜 🔜 🎯 🖆 🥎 🦳 🏷 🖘 🚺								024 SP x64 A								nd a command	9.0	
	The Direct Lont As:	-	et metal PII	weightents		-	PMI	ioois visuaid	te inquire	Electrode	IROBOICAM	App IV	iola simula	stion		no a commano	G 100	
el Import Export Close Save	Simulation Collision	Mechatroni	c Gantt Asse	mbly Move C	Connect Controller		s Workspac	e Calibration		Velding Ab	iout Help							
File	detection	n ulation		Aechanical E	lectrical	edit Ro	bot		cutting Process Pla	inning	Help							
	· 3	• 20	4.0.0	E. E. N. 🛤	0 0 0 0 D	Normal	· h.	· · · /										
tronic	100 2		😵 Robot tool				×									⇒ Ou	tput	
✓ X ⊑	0	right-	V Required					🗇 - 🍵 - 🕅	• ⊕ • .	1 H = 10-	- 🔳 🧕	- 00B	度0000 ~				bot pick and place	e.Z3] ope
Name	Туре	<shift-< td=""><td></td><td>(=</td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td>in (Do</td><td>0.3160 seconds.</td><td></td></shift-<>		(=												in (Do	0.3160 seconds.	
Robot Link4 part	Collision			Tool_base1			+									Ass	sembly [Default] ad	ctivated.
			Rigid body	Tool_base2													mechatronic data	
✓ nobot_Link5	Rigid body Collision						×											
Robot_Link5_part																		
✓ nobot_Link6	Rigid body		Base coordinate	CSVS9			+											
Robot Link6 part	Collision																	
✓ [™] Tool_base1	Rigid body		Tool center	CSYS8			+											
Tool_base1_part	Collision		ioor center	C3130			T											
Tool_base2	Rigid body		▼ Optional															
✓ [™] Tool_4	Rigid body																	
Tool_4_part	Collision			Tool_J2			+				() () () () () () () () () ()				X			
Tool_41	Rigid body			Tool_J3											Vá			
✓ [™] Tool_2	Rigid body			Tool_14			×			- 4					*			
Tool_2_part	Collision		Kinematic pair				^							E				
Tool_21	Rigid body	1									h			V S				
✓ nol_3	Rigid body																	
Tool_3_part	Collision																	
Tool_31	Rigid body																	
✓ Material Tray 2	Rigid body		Sensing point	sensor			+							•				
Material_Tray_2_part	Collision						- H.								1			
Kinematic pairs and constraints			Signal	Tool out			+							-1-		•		
Ø \$1	Revolute joint		Jugitar	looi_out			T							Y -=				
22	Revolute joint		▼ Constrains					1										
Ø 53	Revolute joint		Constraints					1										
Ø 54	Revolute joint												4	γ 🥌				
Ø \$5	Revolute joint		Velocity	100		mm/s												
Ø 56	Revolute joint						_	1		-	-							
Tool J2	Prismatic joint		Max jerk	20		mm/s	· • 1	1										
Tool J3	Prismatic joint																	
Tool J4	Prismatic joint		Max force												*			
Material	Prismatic joint		Positive force			N												
			r comme force			IN		6 m							Λ.			
Coupling pairs			Negative force			N												
Sensors and actuators							-											
RB03A1	Robot																	
a sensor	Sensor		V Name															
Runtime behavior							_											
🚞 Signal adapter		V-	Tool															
Signal connection			10				_											
' 🚃 Signal																		
1. Tool_out	Signal																	
Collision group					OF													
🖮 HMI		-			ОК	Apply Ca	ncel											
		215.10	8mm															

Figure 10-7: Robot Tool

Step 6: Establish assembly relationships

- Click the "Assembly" button in the "IROBOTCAM" tab.
- Click the "Add" button.

Y

- Select child device, select "Tool" robot tool you created in the previous step.
- Install on robot, in the "Parent Device" list, select the "RB03A1" robot.
- Click "OK" button the confirm. (Figure 10-8)

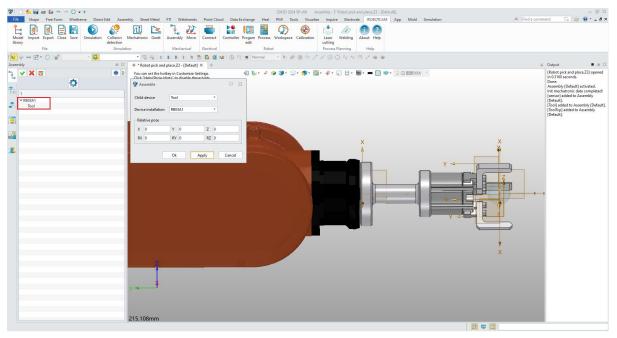


Figure 10-8: Assembly relationships

Step 7: Add signal adapters (2)

Y

- Right-click "Signal Adapter" button.
- Select "Add Signal Adapter". (Figure 10-9)

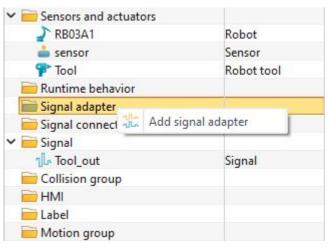


Figure 10-9: Add signal adapters

- Select Robot, click the + button and select the "RB03A1" robot.
- Name and Set Data Type, enter a descriptive name for the signal adapter, such as "RB03A1_S". In the "Data Type" field, select "Bool" to indicate that the signal transmits Boolean values.
- In the formula bar, fill in "RB03A1_S" and name the signal "RB03A1_SD".(Figure 10-10)

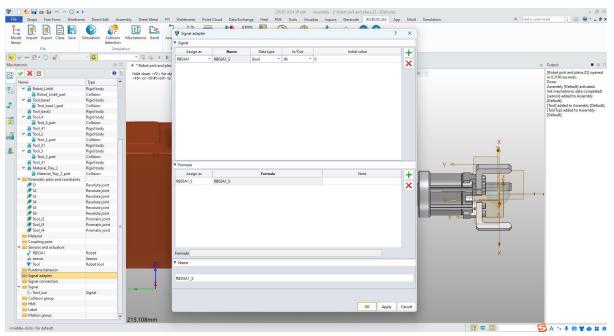


Figure 10-10: Signal Adapter RB03A1

• Repeat the above operations for the robot tools, assigning appropriate names and data types (such as "Tool_S" and "Float"). (Figure 10-11)

Assign as Name Data type In/Out Initial value Tool Tool_S Float IN 0							
Formula Assign as Formula Note	Assign as	Name	Data type	In/Out	Initi	al value	-
Formula Assign as Formula Note fool_S Tool_S	Tool	▼ Tool_S	Float *	IN -	0		
Assign as Formula Note iool_S Tool_S Iool_S							
Assign as Formula Note ool_S Tool_S							
Assign as Formula Note bol_S Tool_S Image: Second se							
Assign as Formula Note ol_S Tool_S Image: Second Sec							
Assign as Formula Note oL_S TooL_S mula							
Assign as Formula Note ol_S Tool_S mula							
Assign as Formula Note oL_S TooL_S mula							
Assign as Formula Note ol_S Tool_S mula							
Assign as Formula Note ol_S Tool_S mula							
Assign as Formula Note ol_S Tool_S mula							
Assign as Formula Note ol_S Tool_S mula							
ol_S Tool_S mula ame	ormula						1
imula lame							
Jame	Assign as		Formula			Note	
lame		Tool_S	Formula			Note	
ame		Tool_S	Formula			Note	
ame		Tool_S	Formula			Note	
ame		Tool_S	Formula			Note	
ame		Tool_S	Formula			Note	
ame		Tool_S	Formula			Note	
lame		Tool_S	Formula			Note]
lame		Tool_S	Formula			Note	
Vame		Tool_S	Formula			Note)
	ol_S	Τσοι_S	Formula			Note	
vol_SD	rmula	Τσοι_S	Formula			Note	
ol_SD	rol_S	Τσοι_S	Formula			Note	
	mula	Τσοί_ς	Formula			Note	
	mula	Tool_S	Formula			Note	
	mula	Tool_S	Formula			Note	

Figure 10-11: Signal adapter tool

Step 8: Add signals

Y

- Right-click on "Signal".
- Select "Add signal" button.(Figure 10-12)

Signal Signal	-11
ျှင် Tool_o ျှင် Add signal	Signal
ျ <mark>မ်း Tool_S</mark>	Signal
Collision group	-

Figure 10-12: Add Signal

• Set input type, in the "IO Type" field, select "In" to define the signal as an input signal.

• Name the Signal, such as "Tool_opt".

Y

- Set Port, In the "Port" field, enter 0 to specify the port on which the signal will be received.
- Click "OK" button to confirm. (Figure 10-13)

🧐 Signal			?	×
 Settings 				
🖉 Connect runtime para	meters			
Parameter name	An	gle		•
IO type		In		-
Data type		BOOL		
Dimension		Angle		÷
Unit		٠		•
Initial value		false		•
▼ Name				
Signal name	Тос	ol_opt		
Controller	RB	03A1Controller	8	•
Port	0			
		OK App		Cancel
		ОК Арр	iy i	Jancer

Figure 10-13: Signal

Step 9: Add simulation sequences (3)

- Right-click "Runtime behavior" button.
- Select "Add simulation sequence". (Figure 10-14)

RB03A1	Robot
📥 sennor	Sensor
🚏 Tool	Robot tool
Runtime beha	
Signal adapter	imulation sequence
Signal adapter Signal connection Signal connection Signal Collision group	imulation sequence

Figure 10-14: Add simulation sequences

• Define RB03A1_EXEC Sequence:(Figure 10-15)

- ♦ Start Time: 1
- ♦ Duration: 999999 (essentially constant execution)
- ♦ Activation/End Signals: Unchecked (no specific activation or end triggers)
- Execution Signal: RB03A1_S (controls the robot's overall movement)
- End Position: 1 (presumably defines the final position of the robot)
 Name: RB03A1_EXEC.

🖉 Simulation sequ	ence		1	? X
Execution time				
Start time	1			s
Duration	99999			5
 Signal configuration 	n			
Activation signal				-
Activation condition	True			*
End signal				-
Execution signal	RB03A	1_S		-+
End position	1			
▼ Name				
RB03A1_EXEC				
		ОК	Apply	Cance

Figure 10-15: GSKRobot_EXEC

- Define Tool_catch_EXEC Sequence:(Figure 3-12)
- ♦ Start Time: 0
- ♦ Duration: 5
- ♦ Activation Signal: Checked (requires activation)
- ♦ Activation Condition: Tool_opt = True
- Execution Signal: Tool_S (controls the robot tool's movement)
- End Position: 1 (presumably defines the final position of the tool during grasping)
- ♦ Name: Tool_catch_EXEC

👰 Simulation sequ	ence		7	? ×
Execution time				
Start time	0			s
Duration	5			s
 Signal configuration 	n			
Activation signal	Tool_opt	t		-+
Activation condition	True			•
🔲 End signal				+
Execution signal	Tool_S			+
End position	1			mm
Name				
Tool_catch_EXEC				
		ОК	Apply	Cancel

Figure 10-1: gongju_catch_EXEC

- Define Tool_pull_EXEC Sequence:(Figure 3-12)
- ♦ Start Time: 0
- ♦ Duration: 5
- ♦ Activation Signal: Checked (requires activation)
- ♦ Activation Condition: Tool_opt = False
- Execution Signal: Tool_S (controls the robot tool's movement)
- End Position: 0 (presumably defines the final position of the tool after releasing)
- ♦ Name: Tool_pull_EXEC

💇 Simulation seq	?	\times	
Execution time			
Start time	0		5
Duration	5		s
 Signal configurati 	on		
Activation signa	I Tool_opt		+
Activation conditio	n False	τ.	
🔲 End signal			+
Execution signal	Tool_S		+
End position	0		mm
Name			
Tool_pull_EXEC			
	ок	Apply (ance

Figure 10-1: gongju_pull_EXEC

- Three simulation sequences define the overall behavior of the robot and robotic tool. They specify the timing, activation conditions and execution signals of different operations:
- ♦ RB03A1_EXEC: This sequence controls the main movement of the robot throughout the simulation.
- Tool_catch_EXEC: When signal "Tool_opt" becomes True, this sequence triggers the robot tool to move towards the object and catch it.
- Tool_pull_EXEC: This sequence triggers the robot tool to move away from the object and drop it when the signal "Tool_opt" becomes False

Step 10: Add rigid body and collison

- Add a rigid body to the material.(Figure 3-12)
- ♦ Click on "Basic Mechatronic Objects" in the IROBOTCAM interface.
- ♦ Right-click and select "Add Rigid Body".
- \diamond Select the desired rigid body (Figure 3-12).
- ♦ Choose the material as "default".
- \diamond Leave the property settings unchecked.
- \diamond Click "OK" to confirm and add the rigid body.

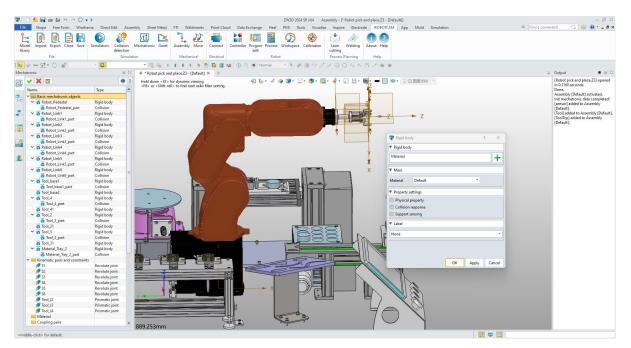


Figure 10-1: Add rigid body

- Add Collision to "Material" (Figure 3-12)
- ♦ Right-click on the "Basic Mechatronic Object" again.
- ♦ Select the rigid body you just added.
- ♦ Choose "Convex decomposition" as the collision shape.
- ♦ Check all three property settings:
 - > Physical properties

Y

- Collision response
- Support induction
- \diamond Set the material to default.
- ♦ Click "OK" to confirm and add the collision object.

	hiect			
Collision O	oject			-
Material				+
▼ Shape				
Collision sha	pe	Convex decor	mposition	•
Material	Default			
Property set	ettings			
Physical p	property			
Collision				
Support s				
▼ Label				
				•
None				
None				
None				

Figure 10-1: Add Collison

- Add Rigid Body to "Material_tray":(Figure 3-12)
- ♦ Repeat steps similar to adding the rigid body to "Material_tray".
- \diamond Select the rigid body representing the support table.
- ♦ Properties are set to default.
- \diamond Click "OK" to confirm and add the rigid body.

😨 Rigid bo	dy			? ×
Rigid bod	ly			
Material_Tr	ay			-
▼ Mass				
Material	Default		•	
Property	settings			
Collision	property response			
 Support Label 	sensing			
None				•
		ОК	Apply	Cancel

Figure 10-1: Add Rigid Body

- Add Collision to "Workpiece Support1":(Figure 3-12)
- ♦ Repeat steps similar to adding the collision object to "WorkPieceNew_1".
- \diamond Select the rigid body representing the support table.
- ♦ Choose "Convex decomposition" as the collision shape.
- ♦ Properties are set to default.

Y

- ♦ Set the material to "default".
- ♦ Click "OK" to confirm and add the collision object.

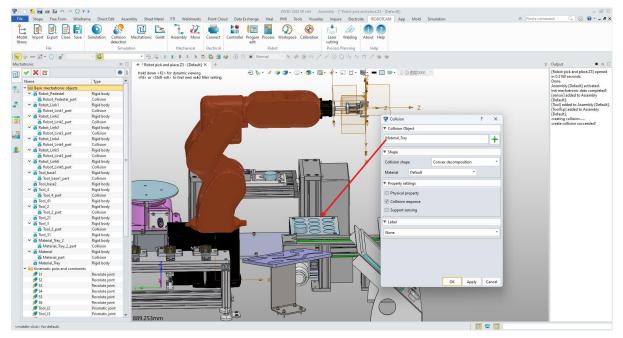


Figure 10-1: Add Collision

Step 11: Prepare coordinate positions

- Click on the "DatumCSYS" option in the "Shape" column.
- This will allow you to create a coordinate system directly on the material itself.
- Ensure the "Z-axis" direction points downwards.
- Select the "Dynamic" option to enable real-time updates of the coordinate system during simulation.
- Right-click in an empty space and choose "Center of Curvature".

(Figure 3-12)

V

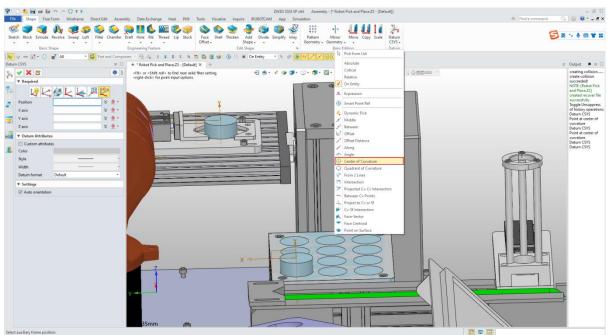


Figure 10-1: Coordinate position

Step 12: Add marker points

- Hide Unnecessary Coordinate Systems:
- ♦ Select the CSYS8 and CSYS18 coordinate systems on the robot tool.
- \diamond Hold down the Shift key and selected all other displayed coordinate systems.
- Uncheck all selected coordinate systems, effectively hiding them from the scene.

(Figure 3-12)

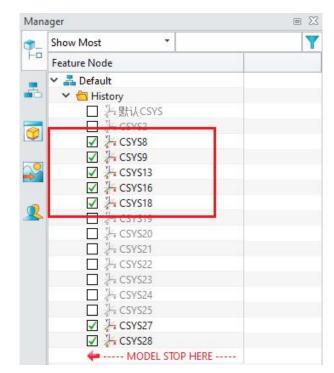


Figure 10-1: Hide the Coordinate Systems

- Teach Robot Movement:
- ♦ Navigate to the "Sensors and actuators" section under the "RB03A1" tab.
- \diamond Right-click and selected "Teach Robot" to begin adding marker points.
 - (Figure 3-12)

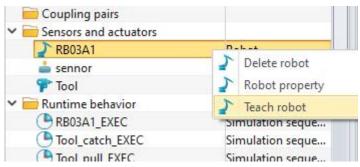


Figure 10-1:

- Adding Marker Points:
- \diamond Set the fifth joint of the robot arm to 90 degrees.
- Click "Record Joint Point" to mark this position as the first point. (Figure 3-12)

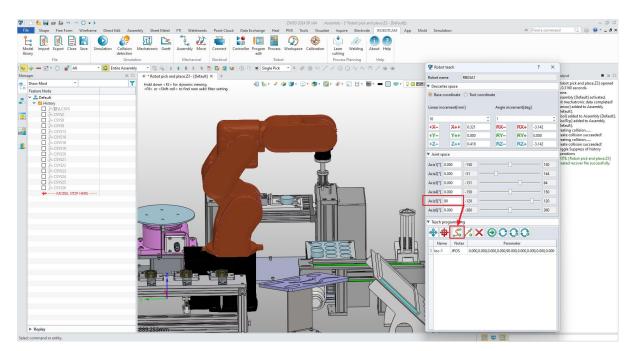


Figure 10-1: The First Point

- Click the "Mark Point" button and selected the coordinate system established on the rigid body.
- Utilize "Drag Teach" and adjusted the robot's gripper position using the XYZ axes.
- Click "Record End Point" to mark this as the second point.

(Figure 3-12)

V

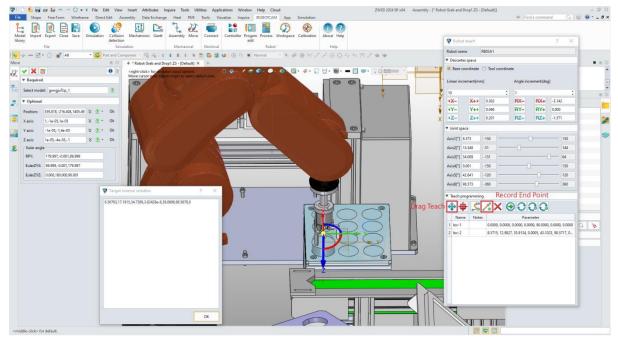


Figure 10-1: The Second Point

- Repeat the "Drag Teach" and "Record End Point" process to mark additional points for the robot's movements, including upwards positions and a point where the gripper separates from another rigid body.
- Finally, click "Move to Home" to return the robot to its initial position. (Figure 3-12)

4	Rob	ot te	ach				?	×			
Ro	bot na	me		RB03A1							
,	Descar	tes sp	ace								
0	Base	coord	dinate		inate						
Lir	near in	crem	ent[m	n]	Angle inc	rement[deg	1]				
10	10				\$ 1	1					
4	X- X++		X+>	0.530	RX-	RX+	0.000				
4	(Y- Y+)		Y+Þ	0.000	RY-	RY- RY+ 1.5					
4	Z-		Z+⊧	0.627	RZ-	RZ+	0.000				
	oint s	pace									
Ax	is1[°]	0.00	0	-150]		150			
Axis2[°]		0.00	0	-51	-0			144			
Ax	is3[°]	0.00	0	-131				64			
Ax	is4[°]	0.00	0	-150	150						
Ax	is5[°]	0.00	0	-120							
Ax	is6[°]	0.00	0	-360		-		360			
	feach (progr	ammir	-							
4	₽ 	₽	S	<u> %</u> 🗙	$\Theta \mathbf{C}$						
	Nar	me	Note			meter					
1	loc-1			0.0000, 0.00	000, 0.0000, 0.0	000, 90.0000	, 0.0000	, 0.0000			
2	loc-2			8.5715, 12.9827, 33.9134, 0.0005, 43.1023, 98.5717, 0							
3	loc-3			8.5728, 16.5933, 34.6565, 0.0006, 38.7480, 98.5729, 0							
4 loc-4		12		8.5727, 3.92	8.5727, 3.9233, 29.4452, 0.0005, 56.6294, 98.5731, 0						
1.65	5 loc-5			61.4811, 30	.0593, -9.6852,	-0.0004, 69.	6247, 15	1.481			
5	6 loc-6		61.4811, 31	61.4811, 31.2714, -7.7833, -0.0004, 66.5107, 151.481							
33	loc-6	,	7 loc-7		61.4811, 28.6646, -12.9885, -0.0004, 74.3227, 151.48						

Figure 10-1: Mark Completed

Step 13: Add commands

Y

• Navigate to "IROBOTCAM" tab, click the "Program Edit" button. (Figure 3-12)

A I D	6 🖬	er 12	n 10	Q =	•										ZW3D 202	4 SP x64 /	Assembly - ['	" Robot pick a	and place.Z3 - [Defa	ult]]		
File	Shape	Free Fo	orm	Wirefram	me Direct	Edit Asse	mbly She	et Metal	FTI We	ldments	Point Clou	id Data	Exchange	Heal	PMI To	ols Visuali	ze Inquir	e Electrod	le IROBOTCAM	Арр	Mold	Simulation
Model library		Export				Collision detection	Mechatroni		Assembly				Progam edit		(And the second	Calibration	Laser cutting	Welding	About Help			
		File				Simula	tion		Mecha	nical	Electrical			Robo	t		Process	Planning	Help			

Figure 10-1: Program Edit

- Add JOB and Program:
- ♦ In the left window, right-click "RB03A1Controller" and select "Add JOB".

Click on the newly added "JOB1" and right-click to choose "Add Program".
 (Figure 3-12)

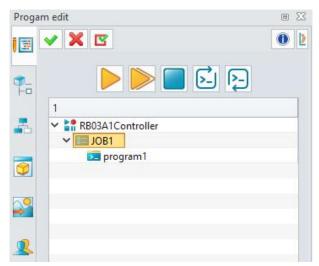


Figure 10-1: Edit Program

- Edit program parameters:
- ☆ Right-click on "program1" to edit it, and a window for program parameters appeared
- ♦ Add all command positions and names used in the program.

(Figure 3-12)

V

Y

V	Required			
•				
	Name	Туре	Data	-
1	loc-1	JPOS	-0.40534, -0.23541, 1.6229, 180, 0, 90,	
2	loc-2	CPOS	-0.35982, -0.21641, 1.4075, 180, -0.00	=
3	loc-3	CPOS	-0.35981,-0.2164,1.3885,180,-0.000	
4	loc-4	JPOS	-0.35981,-0.2164,1.4665,180,-0.000	-
5	loc-5	CPOS	-0.025339, -0.12089, 1.493, 180, -0.00	-
(System M	otion Control	IO Paramter	
•				
•				
•				
•				
•				
•				
•				
•				
•				

Figure 10-1: Program Parameters

- Define Wait time and DOUT values:
- \diamond Set the value of WAIT (ms) to "1000".
- Define the false value of DOUT as "0" and the true value as "1".
 (Figure 3-12)

V Command MOVJ MO Stem Mattern	VL Control IO	
▼ Command WA	IT	
System Mon	Control IO	
0000) 😐	
Command	Paramter	
▼ Command		
System Motion	Control IO	
S 着 🗟 🖉		
Command	Paramter	

Figure 10-1: Command instruction

- Adjust MOVL parameters:
- ♦ Modify the parameter values under the "MOVL" command.
 - (Figure 3-12)

V

Target:	loc-2		
VEL:	0.2	m/s	
ACC:	0.02	%	
JERK:	0.02	%	
		g	V

Figure 10-1: MOVL

- Add commands and modify MOVJ parameters:
- ♦ Added commands according to the provided style.
- Specifically, change the MOVJ command parameter in the seventh line to "100, 50, 50".

(Figure 3-12)

100	bot RB03A1			Ø
V I	Required			
G	- G G	C×		
	Name	Туре	Data	
3	loc-3	CPOS	-0.35981,-0.2164,1.3885,180,-0.000	
4	loc-4	JPOS	-0.35981, -0.2164, 1.4665, 180, -0.000	Γ
5	loc-5	CPOS	-0.025339, -0.12089, 1.493, 180, -0.00	
6	loc-6	CPOS	-0.025339,-0.12089,1.476,180,-0.00	-
7	loc-7	JPOS	-0.025339, -0.12089, 1.518, 180, -0.00	
• (Command			
S	stem Moti	on Control	10	
>	° ৮° ৫৩	? \ 2 ?	<u>,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,</u>	
	Command		Paramter	
1	Command MOVJ	(loc-1,Vel=100.00	Paramter D,Acc=50.00,Jerk=50.00)	
1		(loc-1,Vel=100.00		
2	NOVJ	1000		
2	MOVJ WAIT	1000	0,Acc=50.00,Jerk=50.00)	
2 3	MOVJ WAIT MOVL	1000 (loc-2,Vel=0.20,A	0,Acc=50.00,Jerk=50.00)	
2 3 4	MOVJ WAIT MOVL DOUT	1000 (loc-2,Vel=0.20,A DO[0] 1 1000	0,Acc=50.00,Jerk=50.00)	
2 3 4 5	MOVJ WAIT MOVL DOUT WAIT	1000 (loc-2,Vel=0.20,A DO[0] 1 1000 (loc-3,Vel=0.20,A	0,Acc=50.00,Jerk=50.00) Acc=0.02,Jerk=0.02)	
2 3 4 5 6	MOVJ WAIT MOVL DOUT WAIT MOVL	1000 (loc-2,Vel=0.20,A DO[0] 1 1000 (loc-3,Vel=0.20,A (loc-4,Vel=100.00	0,Acc=50.00,Jerk=50.00) Acc=0.02,Jerk=0.02) Acc=0.02,Jerk=0.02)	
2 3 4 5 6 7	MOVJ WAIT MOVL DOUT WAIT MOVL MOVJ	1000 (loc-2,Vel=0.20,A DO[0] 1 1000 (loc-3,Vel=0.20,A (loc-4,Vel=100.00	0,Acc=50.00,Jerk=50.00) Acc=0.02,Jerk=0.02) Acc=0.02,Jerk=0.02) Acc=0.02,Jerk=0.02) 0,Acc=50.00,Jerk=50.00)	
2 3 4 5 6 7 8	MOVJ WAIT MOVL DOUT WAIT MOVL MOVJ MOVL	1000 (loc-2,Vel=0.20,A DO[0] 1 1000 (loc-3,Vel=0.20,A (loc-4,Vel=100.00 (loc-5,Vel=0.20,A DO[0] 0	0,Acc=50.00,Jerk=50.00) Acc=0.02,Jerk=0.02) Acc=0.02,Jerk=0.02) Acc=0.02,Jerk=0.02) 0,Acc=50.00,Jerk=50.00)	

Figure 10-1: Command

• These steps ensure that the robot follows specific commands and performs the desired pick-and-place operation based on the marked points and defined parameters.

Step 14: Simulation Verification

- Make the interface clean
- Hide all coordinate systems by unchecking them to improve the interface clarity.
- ♦ Additionally, you hid the component "ReferenceMateial" to better visualize the robot's gripper and its interaction with the object.

(Figure 3-12)

✓ ∠ Default	
🖌 🇊 (–)Material_Tray_2	
🖌 🇊 (–)Material	
🖌 🇊 (–)Material_Tray	
🗹 🇊 (–)Tool_2	
🖌 🍞 (–)Tool_base2	
🖌 🇊 (–)Tool_4	
🖌 🇊 (–)Tool_base1	
🗹 🧊 (–)Tool_21	
🖌 🇊 (–)Tool_3	
🖌 🇊 (–)Tool_31	
🔽 🗊 (-)Tool_41	-
🔽 🇊 (–)ReferenceMaterial	
🔽 🍞 (-)RB03A1	-
🖌 🇊 (–)GSK_Target	
🔽 🧊 (–)RB03A1_Flange	

Figure 10-1: Hide

- Start simulation:
- ♦ Click the "Simulation" button in the "IROBOTCAM" tab.

(Figure 3-12)

10	💯 🗋 🗲 开 🥁 🏭 🗠 🗢 🔿 🗢 🕽 🖛 🖉																					
File	Shape	Free F	orm	Wirefra	me Direct	Edit Asse	mbly Shee	t Metal	FTI V	Veldments	Point Clou	ud Data E	Exchange	Heal	PMI T	ools Visuali	e Inquir	e Electro	de IROBOTCAM	Арр	Mold	Simulation
Model library						Collision detection					Connect				Workspace	Calibration	Laser cutting	Welding	About Help			
		File				Simula	tion		Mec	hanical	Electrical			Robo	t		Process	Planning	Help			

Figure 10-1: Simulation Button

• Record robot motion

- Optionally checked the "Record video" checkbox to capture the robot's movement during simulation.
- Run simulation and stop recording
- ♦ Click the "Start" button in the "Simulation Control" to run the programmed robot actions.
- After observing the entire process and confirming successful capture (if video recording was enabled), you clicked the "Stop" button.

(Figure 3-12)

Y

nulation	≝• () • *	1	1	Part Only
	12			0
▼ Requ	iired			
Entity:				
				> 🔮
▼ Setti	ngs			
Simula	tion	0		
	🔲 Drav	w <mark>trajector</mark> y		
Sampl	ing[ms] 50			*
	Rec	ord video		

Figure 10-1: Start

- Verify successful pick and place:
- Confirmed that the robot successfully grasped and released the object based on the simulation results.

(Figure 3-12)

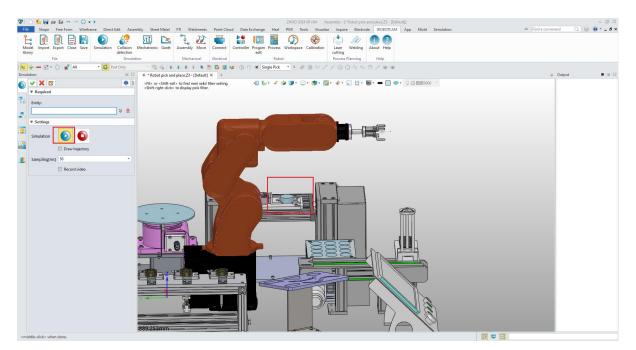


Figure 10-1: Simulation verification

This step ensures that the robot's movements and actions match the programmed commands and successfully execute the desired pick-and-place operation. Recording the simulation allows you to analyze the process and verify its accuracy and efficiency.

Here are some additional thoughts on each error type:

• Speed, acceleration, and jerk parameters:

1.Symptoms: The robot may move too quickly or slowly, or it may experience jerky movements.

2.Causes: The values for speed, acceleration, and jerk may be outside the acceptable range for the robot's capabilities.

Solutions:

1.Check the robot's specifications for the recommended values for these parameters.

2.Adjust the values in the program instructions until the robot moves smoothly and accurately.

3.Use the "Record joint point" and "Record end point" functions during teaching to ensure realistic speeds and accelerations.

• Same point with two different commands:

1.Symptoms: The robot may not move to the desired location, or it may move in an unexpected way.

2.Causes: The program may include two different instructions for the same point in the robot's path.

• Solutions:

1.Carefully review the program instructions to identify any duplicate commands.

2.Remove the duplicate command or choose the desired command for that point.

3.Use the "History management" bar to view and edit the teaching points and instructions.

4.Here are some additional tips for avoiding errors:

5.Test the program in stages: Start by testing the program with a small number of instructions and then gradually add more complexity. This will help you identify any errors more quickly.

6.Use the simulation visualization: Watch the robot's movements in the simulation window to identify any unexpected behavior.

7.Consult the IROBOTCAM documentation: The documentation provides detailed information on the software's features and functions, as well as troubleshooting tips.

By following these tips and carefully reviewing your program, you can avoid these common errors and ensure that your robot pick-and-place simulation runs smoothly and accurately.

Chapter 11: Collision Detection

Step-by-Step Guide on Collision Detection Simulation in iRobotCAM.(Figure 11-1)

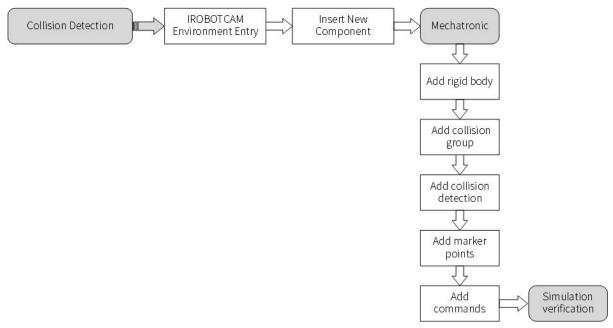


Figure 11-1: Flow chart of collision detection

Step 1: Open iRobotCAM project file

Y

• Open iRobotCAM project file "Collision Detection. Z3" to enter the IROBOTCAM environment. (Figure 11-2)

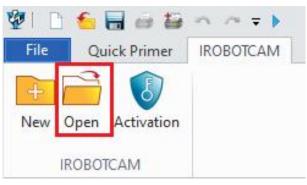


Figure 11-2: Open iRobotCAM project file

• The component details will show after enter it.(Figure 11-4)

월 🗅 🌜 🖬 🗃 🗃 🗛 주 주 🗘 🖘	ZW3D 2024 SP x64 Assembly - [Collision Detection.Z3 - [Default]]	. g II
File Shape Free Form Wireframe Direct Edit Assembly Si	heet Metal FTI Weldments Point Cloud Data Exchange Heal PMI Tools Visualize Inquire Electrode IROBOTCAM App Mold Simulation	🌣 Find a command 🛛 🖓 🔞 🔹 🕳 🗗 🗙
Import Export Close Save Initiation File Simulation Simulation	nic Gant Assembly Move Controller Progam Process Workspace Calibration Unity Process Planning Help	
🙀 🕂 🗕 📰 🔹 🚫 💕 🔹 😻 Part Only 🔹	12 🖡 🗈 🗉 🗉 🕷 🖄 🕼 🕼 0. P. 🖲 Normal 👘 🔌 🖉 🞯 🖅 🖊 🔿 🗛 🖉 🖉	
Manager III III + Col	lision Detection.Z3 - [Default] × +	🗢 Output 🔳 💷 🖾
Seven Al		[Collison Detection.23] opened in 0.1030 second: Done. Assembly [Dutault] activated. Init mechatronic data completed]
✓ (~)(unb) ✓ (~)(unb)		
	Z	
	x x	
4772.	34mm	

Figure 11-4: Component Details

Step 2: Insert a new component

- Right click on "KR210_R2700_extra" to insert a new component.
- Name the part "COLLISION".
- Uncheck the "Automatic Activation" checkbox.
- Click the "Create" button to complete the creation of the new component. (Figures11-6)

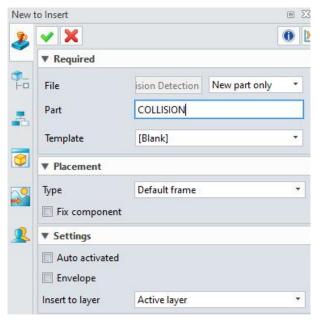


Figure 11-6: Insert New Component

- Right click to delete the coordinates under the constraint.Deleting the coordinates will also delete the constraints.(Figure 11-7)
- This creates the component named "COLLISION" that will be used for collision detection.

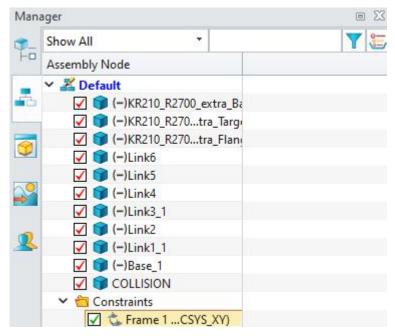


Figure 11-7: Delete Constraint

Note:

- The provided figures (11-2 to 11-7) are helpful for visualizing the described steps.
- Please refer to the software's documentation for confirmation.

Step 3: Insert and Adjust block

- Insert block.
- ♦ Find the "block" option in the "Shape" column and select it.
- ♦ Build the desired hexahedron shape within the interface, as shown in the provided figure.
- ♦ Click the "Checkmark" button to finish creating the hexahedron.(Figure 11-8)

Sketch Bickt Extude Revolve Sweep Loft Fillet Chamfe	Construction Offset Support Geometry Construction Construction Support TH 1	A Field a command Q 💿 😧 • = 🗗
Replay	<pre>**Celling deadling 2: [CULUIDS] * * " " " " " " " " " " " " " " " " " "</pre>	Output Output

Figure 11-8: Insert Hexahedron

- Adjust Hexahedron Position:
- Double-click on the "KR210_R2700_extra" assembly to activate it as a whole. (Figure 11-9)

Man	ager	0 23
¢_	Show All 🔹	75
+0	Assembly Node	
-	✓ [™] Default	
6	✓ (¬)KR210_R2700_extra_Ba	
-	🔽 🌍 (-)KR210_R270tra_Targi	
	🗹 🌍 (-)KR210_R270tra_Flan	
	🗸 🌍 (–)Link6	
	🗹 🇊 (–)Link5	
	🗹 🌍 (–)Link4	
	🗹 🇊 (-)Link3_1	
1	🗹 🌍 (–)Link2	
	🗹 🌍 (-)Link1_1	
	🗹 🌍 (-)Base_1	
	🗹 🌍 (–)COLLISION	

Figure 11-9: Activate Assembly

- ♦ Click on "Move" in the "Assembly" column.
- ♦ Select the hexahedron you created.
- ♦ Adjust the XYZ axes to move the hexahedron to the desired position within the assembly. (Figure 11-10)

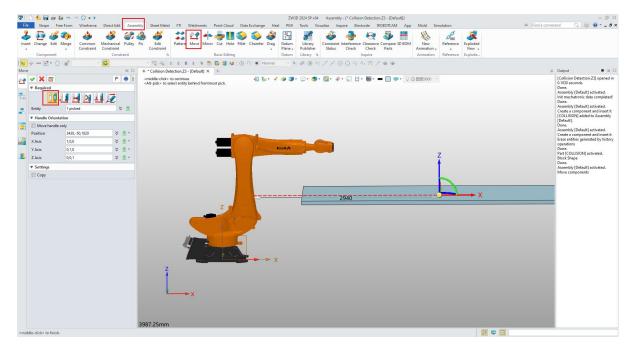


Figure 11-10: Adjust the hexahedron

Step 4: Add Rigid Body and collision

Y

- In the "IROBOTCAM" column, click on "Mechatronic".
- Right-click on "Basic Mechatronic Objects" and select "Add Rigid Body and Collision".
- Click the + button and choose the hexahedron you created from the list.
- Choose the ""Default" material.
- In the collision window, Under "Property Settings", select ""Support Sensing". (Figure 11-11)
- Click "OK" to finalize the creation of the rigid body and collision.

D 😉 🖬 🗃 🗃 🗠 🗠 🗘 Ŧ			ZW3D 2024 SP x64 Assembly - [* Collision Detection.Z3 - [Default]]	- 3
	ime Direct Edit Assen	nbly Sheet Metal FTI Weldments Point Cloud Dat	a Exchange Heal PMI Tools Visualize Inquire Electrode IROBOTCAM App Mold Simulati	tion 🗢 Find a command 🔍 🛞 👰 • 💶 e
Model Import Export Close Save File	Simulation Collision detection Simulati		Image: Notest and the sector of the secto	
	- 3	· · · · · · · · · · · · · · · · · · ·		
chatronic				⊊ Output ■ m
✓ X E	0 1	Hold down <f2> for dynamic viewing.</f2>	은 않고 🖉 🤣 🗊 - 🗇 - 🍘 - 🔞 - 😨 🗄 - 🗐 - 💻 🔲 🧶 - 🖓 🖓 🖄 👘	[Collision Detection.Z3] opened
Name	Туре	<f8> or <shift-roll> to find next valid filter setting.</shift-roll></f8>		0.1030 seconds. Done.
	type			Assembly [Default] activated.
 Basic mechatronic objects Base_1 	Rigid body		🖞 Collision 7 X	Init mechatronic data complete
Base_1_part	Collision		V Collision Object	Done.
V i Link1_1	Rigid body		* Conson object	Assembly [Default] activated. Create a component and insert
ink1 1 part	Collision		COLLISION_1	[COLLISION] added to Assembly
V i Link2	Rigid body			[Default].
Link2 part	Collision		▼ Shape	Done.
✓ S Link3 1	Rigid body		* snape	Assembly [Default] activated. Create a component and insert
Link3_1_part	Collision		Collision shape Convex decomposition	Erase entities generated by hist
V S Link4	Rigid body	· · · · · · · · · · · · · · · · · · ·		operations
Link4 part	Collision		Material Default *	Done.
Y n Link5	Rigid body			Part [COLLISION] activated.
Link5_part	Collision		V Property settings	Block Shape Done.
✓ a Link6	Rigid body			Assembly [Default] activated.
Link6 part	Collision		Physical property	Move components
COLLISION 1	Rigid body		Collision response	
V Kinematic pairs and constraints				
Joint1	Revolute joint		Support sensing	
Joint2	Revolute joint			
Joint3	Revolute joint		V Label	
Joint4	Revolute joint	7	New Y	
Joint5	Revolute joint		None	
Joint6	Revolute joint	- Contraction -		
Material		Contraction of the second s		
Coupling pairs				
Sensors and actuators		de la companya de la		
KR210_R2700_extra	Robot			
Runtime behavior				
🚞 Signal adapter				
Signal connection			OK Apply Cancel	
🧰 Signal				
Collision group		Z		
🚞 HMI		4		
🚞 Label				
i Motion group				
		X		
		3987.25mm		
hiddle-click> for default.				

Figure 11-11: Add rigid body and collision

Step 5: Add collision group

- Right-click on "Collision Group" to add a new group.
- Select multiple components representing the robot's rigid bodies.
- Click the "Add" button to add them to the box on the right.
- Alternatively, hold the "Shift" key and click to quickly add all robot rigid body components.
- Name the newly created collision group "Robot". (Figure 11-12)

SV.

2 Collision group		?	×
Basic object	Collision group object		
Base_1_part Link1_1_part Link2_part Link3_1_part Link4_part Link5_part Link6_part COLLISION_1_part	Base_1_part Link1_1_part Link2_part Link3_1_part Link4_part Link5_part Link6_part		
▼ Name robot			
	ОК А	pply	Cancel

Figure 11-12: Robot collision group

- Repeat the above steps to create a new collision group for the hexahedral rigid body.
- Name this group "object". (Figure 11-13)

💯 Collision group		?	\times
Basic object	Collision group object		
Base_1_part Link1_1_part Link2_part Link3_1_part Link4_part Link5_part Link6_part COLLISION_1_part	COLLISION_1_part		
▼ Name			
object			
	ОК	Apply	Cancel

Figure 11-13: Object collision group

Step 6: Add collision detection

• In the ""IROBOTCAM" column, click on the "Collision Detection" button. (Figure 11-14)

21 B	6 🔒	÷ 5	n 0	0 =	•										ZW3D 20	24 SP x64 As	sembly - [* Collision D	etection.Z3 - [Defau	lt]]		
File	Shape	Free F	orm	Wirefra	me Direct	Edit Asse	embly Shee	t Metal	FTI We	eldments	Point Clo	ud Data B	Exchange	Heal	PMI To	ols Visualize	Inquir	Electro	le IROBOTCAM	Арр	Mold	Simulatio
Ŀ	ST I	DI T	E.		\triangleright	<u></u>									\bigcirc		١	D	1?			
Model library	Import	Export	Close	Save	Simulation	Collision detection	Mechatronic	Gantt	Assembly	Move	Connect	Controller	Progam edit	Process	Workspace	Calibration	Laser cutting	Welding	About Help			
		File				Simula	ation		Mecha	anical	Electrical			Robo	t		Process	Planning	Help			

Figure 11-14: Collision Detection

- Select the content shown in the figure under the output boxes of "Target object" and "Obstacle" respectively.
- These should correspond to the previously created "object" and "robot" groups.
- Click "Add Collision" to create two collision pairs.
- Click the "checkmark" to complete the creation. (Figure 11-15)

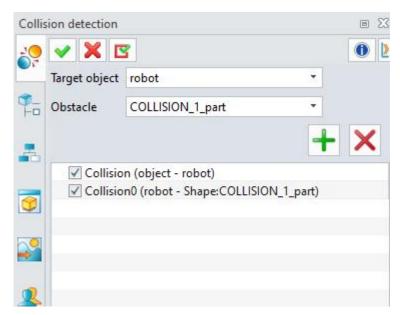


Figure 11-15: Collision detection

Step 7: Add marker points and commands

• Under "Sensors and actuators", right-click on "Robot" and select "Teach Robot". (Figure 11-16)

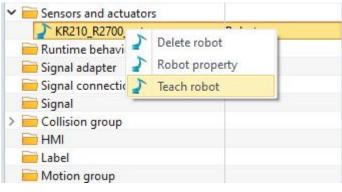


Figure 11-16: Teach Robot

- Click on "Drag Teach", then click anywhere on the robot.
- Drag the X-axis downwards until you reach the bottom of the hexahedron.
- Click to "Record End point" at this location. (Figure 11-17)

SV.

rary detection File Simulat	Image: Section of the section of	Robot teach ? X Robot teach ? X Robot teach ? X	
K C Required Select model: KR210,R2700,extra,Flange,1 Cyptional	◆ * Calina Dectaina 2 - Ufutudi X ◆ More corar new functionaringh to subsch dama ase. - cright-dicks for direction input options	Base coordinate Tool coordinate Linear increment[mm] Angle increment[deg] 10 1 1	[Collision Detection.23] opened 0.1030 seconds. Done. Assembly [Default] activated. Init mechatronic data complete Done. Assembly [Default] activated. Create a component and insert (COLLISION) added to Assembl
Papelier 1764.9580_0.004.43.997 S ▲ 2 ▲ 0. X mits -0.4.1 S ▲ - 0. Y mits -0.4.1 S ▲ - 0. Z mits 10.4.0 S ▲ - 0. Z mits 10.4.0 S ▲ - 0. Externage -0. S ▲ - 0. Externage -0. S ▲ - 0. Externage -0.0.0.0.0. S ▲ - 0. Externage -106.70.0.0.0.0.0.0. -0.0.		4Y- Y++ 0.000 RY- RY+ 1.571 4Z- Z++ 0.044 RZ- RZ+ 0.000 V Joint space Assit17 0.000 -185	[Coccision] added to exemult [Default]. Done. Assembly [Default] activated. Create a component and insert Generations Dene. Part (COLUSION] activated. Biock Shape Done. Assembly [Default] activated. Move components creating collision
Issuerce UMA(MUMA, UMA		Anistig 45.607 125 Anistig 1.000 350 300 V Each programming 300 300 V Each programming 300 300 Image: Notes Farameter 1 10c-1 1 10c-1 1764-596,0.009,443.994,0.000,0000,0.000	
	X Z		

Figure 11-17: First marking point

- Close drag teach window and add a record point above the hexahedron. (Figure 11-18)
- This ensures the robot collides with the hexahedron when the command is executed.

Stape Free Form Windfame Direct Edit Assembly Sheet Me a Import Impo	s 🤁 🕰 🖷 🔚 🔳	Workspace Calibration	E Inquire Electrode IROBOTCAM						la command 🔍 🍥 🖗 * 🗕
			1004NA/**	💇 Robot	teach			? ×	
	Detection.Z3 - [Default] × +	10 MP (0 1. 7	/ 00 9 9 11 /	Robot nan	me KR210_R2700_	extra_Base			⊊ Output ■
V X E Move cursor	near datum origin to select datum axes. 🗧 🦦 🗸	👙 🌒 - 🗇 - 🍵 - 🔯	•	- Descarte	is space				[Collision Detection.Z3] opene
▼ Required <right-dick></right-dick>	for direction input options			Base o	oordinate 🔿 Tool coor	dinate			0.1030 seconds. Done.
Select model: KR210_R2700_extra_Flange_1				Linear inc	rement[mm]	Angle inc	crement[de	g]	Assembly [Default] activated. Init mechatronic data complet Done.
▼ Optional				10		0 1		5	Assembly [Default] activated. Create a component and inser
Position: 1765.001.0.005.1803.997 😵 🖲 - Ok				•X-	X+> 1.765	RX-	RX+	0.000	[COLLISION] added to Assemb
				4Y-	Y+> 0.000	RY-	RY+	1.571	[Default]. Done.
				<z−< td=""><td>Z++ 1.804</td><td>RZ-</td><td>RZ+</td><td>-0.000</td><td>Assembly [Default] activated. Create a component and inse</td></z−<>	Z++ 1.804	RZ-	RZ+	-0.000	Assembly [Default] activated. Create a component and inse
Yaxis: -0,1,-0 🛛 💥 💇 - Ok		-20		▼ Joint spa	ace				Erase entities generated by his operations
Z axis: 1,0,-0 😵 👲 * Ok		the N		Axis1[*]	-0.000 -185		1	185	Done. Part [COLLISION] activated.
Euler angle RPY: -157.314.90.000157.314	KOK		→ Z	Axis2[*]				4,99999	Block Shape Done.
			·	Axis3[*]				155	Assembly [Default] activated. Move components
EulerZYX: -157.314,90.000,-157.314								350	creating collision
EulerZYZ: 0.000,90.000,-0.000	Z			Axis4[*]					create collision succeeded!
				Axis5[*]				125	
		- ^		Axis6[*]	-0.001 -350			350	
				▼ Teach pr	rogramming				
	lane.				₽ <i>\$\$</i> /\$ X	00	00		
	4			Nam			imeter	· · · · · · · · · · · · · · · · · · ·	
		► X		1 loc-1		.009,443.994.0.0		000 0.000 0.0	
				2 loc-2				0.000,0.000,0	
	Z Ó X								
5725.06mi	n								

Figure 11-18: Second marker

• In the "IROBOTCAM" column, click on "Program Edit". (Figure 11-19)

1 D	🐉 🗋 🖕 🗃 🥔 🖆 🗢 🔿 🗇 🛟 🔹 🕨																					
File	Shape	Free F	orm	Wirefram	me Direct	Edit Asse	embly She	et Metal	FTI W	/eldments	Point Clos	ud Data	Exchange	Heal	PMI To	ols Visualize	e Inquir	e Electro	de IROBOTCAM	App	Mold	Simulation
£																						
library	import	Export	Close	Save	Simulation	detection	Wechatroni	c Gantt	Assembl	y move	Connect	Controlle	edit	rocess	workspace	Calibration	Laser cutting	Welding	About Help			
		File				Simula	ation		Mech	nanical	Electrical			Robo	t		Process	Planning	Help			

Figure 11-19: Program edit button

• Right-click on "Controller" and select ""Add JOB".

V

- Right-click on "JOB1" and select "Add Program".
- Right-click on "program1" and select "Edit Program".
- Use the "MOVL" command and specify "loc-1" and "loc-2" as the motion targets. (Figure 11-20)

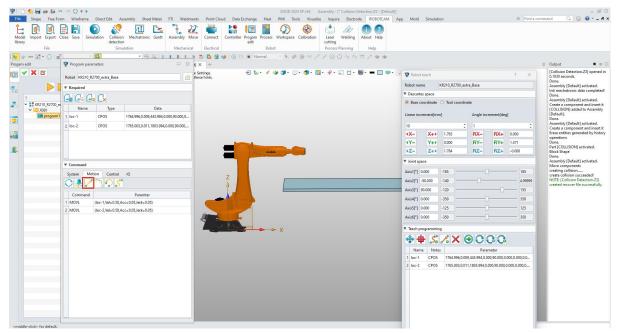


Figure 11-20: Add command

Step 8: Simulation Verification

M I D 6 🖂 🖂 🗠

• In the "IROBOTCAM" column, click on "Program Edit". (Figure 11-21)

File	Shape	Free F	orm	Wirefram	me Direct	Edit Asse	mbly Shee	t Metal	FTI We	ldments	Point Clos	ud Data	Exchange	Heal	PMI To	ols Visualize	Inquir	e Electroo	le IROBOTCAM	App	Mold	Simulatio
Model library							Mechatronic									Calibration	Laser cutting	Melding	About Help			
		File				Simula	tion		Mecha	nical	Electrical	1	1	Robo	t		Process	Planning	Help			

Figure 11-21: Program edit

• In the "Program Edit" window, click the "Run" button. (Figure 11-22)

V

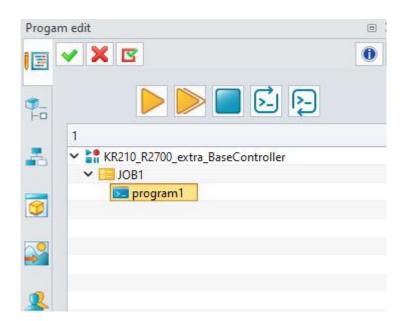


Figure 11-22: Run button

- If the program doesn't start running immediately, try clicking "Run" once, waiting a few seconds, and then clicking "Stop".
- Refresh the interface and then click "Run" again. This sometimes triggers the program to start running.
- Once the program is running, you should see a highlighted status indicator, confirming successful collision detection. (Figure 11-23)

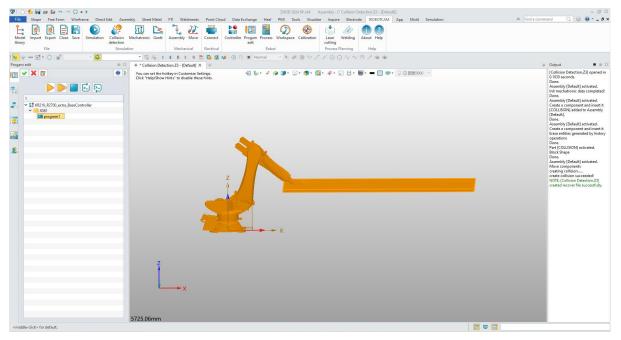


Figure 11-23: Simulation verification

Chapter 12: Configure Communication

Connecting iRobotCAM Model to External Controller.(Figure 12-1)

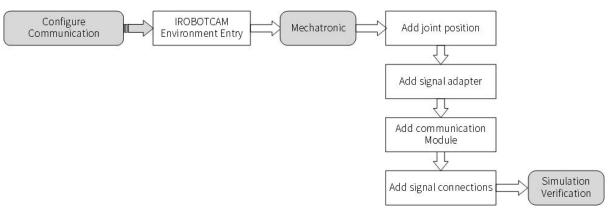


Figure 12-1: Flow chart of configure communication

Step 1: Open iRobotCAM project file

V

• Open iRobotCAM project file "Configure Communication.Z3" to enter the iRobotCAM environment. (Figure 12-1)



Figure 12-2: Open iRobotCAM project file

• Wait for the model to import. Once complete, there is a message will appear in the red box below the output box on the right. (Figure 12-3)

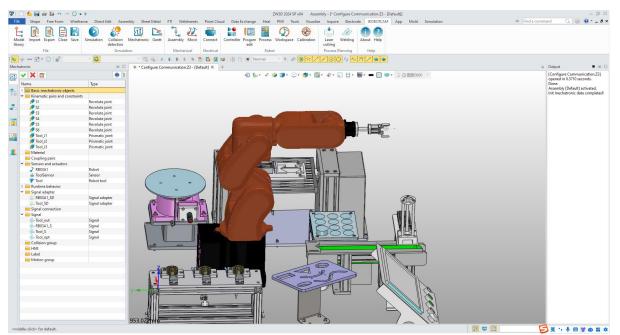


Figure 12-3: Component details

Step 2: Add Joint Positions

W

- In the "IROBOTCAM" column, click the "Mechatronic" button, then "Runtime behavior".
- Right-click on "Robot_EXEC" and delete the simulation sequence. (Figure 12-4)

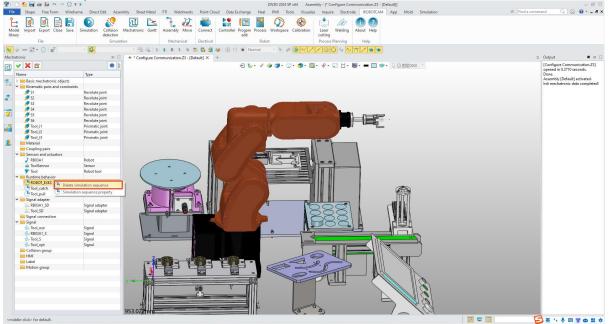
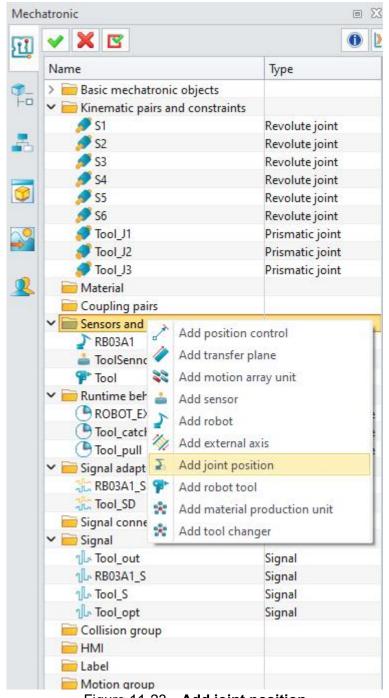


Figure 12-4: Delete the simulation sequence

 Click on "Sensors and actuators" and right-click to add joint positions. (Figure 12-5)



- Figure 11-23: Add joint position
- For each joint "S1" to "S6", follow these steps:
- \diamond Click the "+" button.
- \diamond Select the joint (eg, "S1").
- \diamond Name it "S1Pos" (replace "1" with the appropriate joint number).
- ♦ Set the motion type to "Specify position".
- ♦ Click OK. (Figure 12-6)

🖗 Joint positi	on			? ×
🔻 Kinematic p	oair			
S1				+
▼ Name				
S1POS				
Motion type:	Specify po	sition 🔻		
		ОК	Apply	Cancel

Figure 12-6: Joint position parameters

• After completing steps for all joints, the interface should resemble.

(Figure 12-7)

RB03A1	Robot
📥 ToolSennor	Sensor
🏲 Tool	Robot tool
🚡 S1POS	Joint position
S2POS	Joint position
S3POS	Joint position
S4POS	Joint position
S5POS	Joint position
S6POS	Joint position

Figure 12-7: Joint position completed

Step 3: Add Signal Adapter

- Click on "Signal Adapter".
- Right-click and select "Add Signal Adapter".
- Follow these steps for each joint: (Figure 12-8)
- \diamond Click the "+" button.
- ♦ Select the joint position (eg: "S1Pos").
- Double-click the second box and enter the name "S1Pos_S" (replace "1" with the appropriate joint number).
- ♦ Select "Float" for the data type in the third box.
- \diamond Click the "+" button on the right in the second column.
- \diamond Fill in "S1Pos_S, S1Pos_S * 0.017444" (the value source 3.14/180).
- Name the adapter "S1Pos_SD" (replace "1" with the appropriate joint number).

W

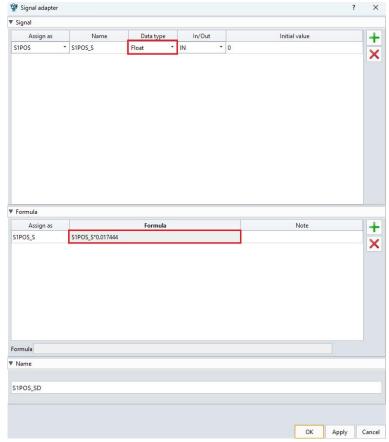


Figure 12-8: Signal adapter S1Pos_SD

• The final result should look similar to .(Figure 12-9)

🔁 Signal adapter	
號 RB03A1_SD	Signal adapter
Tool_SD	Signal adapter
🏦 S1POS_SD	Signal adapter
🚲 S2POS_SD	Signal adapter
🚲 S3POS_SD	Signal adapter
S4POS_SD	Signal adapter
抗 S5POS_SD	Signal adapter
🚠 S6POS_SD	Signal adapter
Signal connection	
Signal	
1 Tool_out	Signal
1 RB03A1_S	Signal
1 Tool_S	Signal
1 Tool_opt	Signal
1 S1POS_S	Signal
1 S2POS_S	Signal
1 S3POS_S	Signal
1 S4POS_S	Signal
1 S5POS_S	Signal
1 S6POS_S	Signal

Figure 12-9: Signal adapter completed

Step 4: Add Communication Module

Y

• Click on the "Connect" button in the iRobotCAM column. (Figure 12-10)

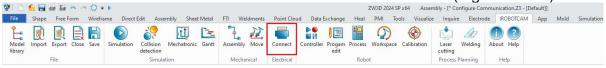


Figure 12-10: Connect button

• Under "Modbus Tcp", right-click the blank box under "Devices" and select "Add Device". (Figure 12-11)

Device				
Device name	Server address	Server port		
		 Add dev Delete d Device p 		
 Outer signal 	IO type	Data type	Type name	Device na
Name				

Figure 12-11: Outer signal

- In the pop-up window: (Figure 12-12)Here is an example setting.
- ♦ Name the device "ModbusTcp1".
- ♦ Set the server address to "192.168.1.10".
- ♦ Set the server port to "502".
- ♦ Set the mode to "1".
- ♦ Click OK.

Y

Device Name	Server address	Server port	Mode	
ModbusTcp	192.168.1.10	502	1	
Outer signal				
	IO type	Data type	Offset	ls joint
 Outer signal Name 	IO type	Data type	Offset	ls joint
	IO type	Data type	Offset	ls joint
	IO type	Data type	Offset	ls joint
	IO type	Data type	Offset	ls joint

Figure 12-12: Outer signal device parameters

Device				
Device Name	Server address	Server port	Mode	
ModbusTcp	192.168.1.10	502	1	
Ø Outer signal				
🗸 Outer signal Name	IO type	Data type	Offset	ls joint
	IO type	🕒 Ad	d signal	Is joint
	IO type	Ad De		Is joint

Figure 12-13: Add signal

- In the pop-up window: (Figure 12-14)
- \diamond Name the signal "J1".
- ♦ Select "Double" for the signal type.
- ♦ Select "OUT" for the IO type.
- ♦ Select "Yes" for the joint.
- \diamond Set the offset address to "65536".
- \diamond The remaining window settings are filled in according to the actual device.
- ♦ Click Add.

Y

Device Na	me	Server a	ddress	Server port	Mode						
ModbusTc		192.168.1		502	1						
	💯 Mo	odbus tcp	signal			?	×				
	Name	:	J1								
	Signal	type:	Doubl	e			•				
	IO typ	e:	OUT -								
Outer si	ls join	t:	Yes 🔹								
Name	Offset	:	l I					oint			
	Lengt						_				
	Index:										
	Regist	er type:	Input	Register	ОК	Car	ncel				
		a l			OR	Cur	icei				

Figure 12-14: Add joint signal

 Repeat step 5 for the remaining joints, naming them "J2" to "J6", selecting "Double" for the signal type, "OUT" for the IO type, "Yes" for the joint, and offset addresses incrementally increasing by 4 (eg: 65540, 65544). (Figure 12-15)

Opc Ua Sieme	ens Plc Smart	Modbus Tcp	BeckHoff Cont	roller
Device	- 14	1.2		
Device Name	Server addr	Server port	Mode	
ModbusTcp1	192.168.1.10	502	1	
 Outer signal 	1.2			
Name	IO type	Data type	Offset	Is joint
	IO type OUT	Data type Double	Offset 65536	Yes
Name J1	and the second s	and Annal	a Articles	
Name J1 J2	OUT	Double	65536	Yes
Name	OUT OUT	Double Double	65536 65540	Yes Yes
Name J1 J2 J3	OUT OUT OUT	Double Double Double	65536 65540 65544	Yes Yes

Figure 12-15: Joint signal completed

- Add communication for grabbing and releasing: (Figure 12-16)

SV.

- ♦ Name the signal "Grab".
 ♦ Select "Bool" for the signal type.
- ♦ Select "OUT" for the IO type.
- ♦ Select "No" for the joint.
- \diamond Set the offset address bit to "65560".
- ♦ The remaining window settings are filled in according to the actual device.

 \odot

		1		1				_
Device Na	me	Server a	ddress	Server port	Mode			
ModbusTc	p	192.168.1	.10	502	1			
	🖗 м	lodbus tcp	signal			?	×	
	Nam	e:	clamp	c.				8
	Signa	il type:	Bool				•	
	10 typ	pe:	OUT	Ψ.				
Outer si	ls joir	nt:	No				•	
Name	Offse	t:						oint
	Leng	th:						
	Index	:						
	Regis	ter type:	Input	Register				
	A	dd			ОК	Ca	ncel	

Figure 12-16: Pick and place outer signals

• Click the "Save" button to save the external signal connection information to the computer for future use. (Figure 12-17)

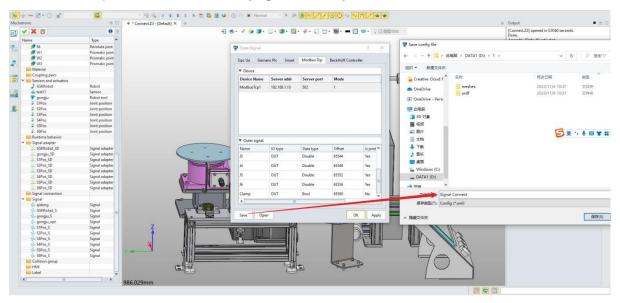


Figure 12-17: Save signal

Connecting iRobotCAM Model to External Controller (Continued)

Step 5: Add Signal Connections

Y

• Under "Mechatronic", locate "Signal Connection". (Figure 12-18)

>	Sensors and actuators	
>	Runtime behavior	
>	🔁 Signal adapter	
>	Signal connection	
	Collision group	_
-	HMI	
6	Label	

Figure 12-18: Add signal connection

- Right-click and select "Add Signal Connection".
- In the pop-up box, name it "SignalConnect1" and click "OK". (Figure 12-19)

💯 Please enter name of sig	nal connection	?	×
SignalConnect1			
	OK	Can	col

Figure 12-19: Signal connect name

- Within the "Signal Connection" box:
- Click the internal signal "S1Pos_S" in the left column. (Figure 12-20, point 1)
- ♦ Click the external signal "J1" in the right column. (Figure 12-20, point ②)
- ♦ Click the "Connect" button.

ignal connection	name					Sign	alConnect1		
roject name					C)efaultProject			
evice name					1	AodbusTcp1			
Signal									
Name	Adapter name	IO type	Data type			Name	IO type	Data type	-
S1Pos_S 1	S1Pos_SD	IN	Float			J1 (2)	OUT	Double	
S2Pos_S	S2Pos_SD	IN	Float			J2	OUT	Double	
S3Pos_S	S3Pos_SD	IN	Float		0	J3	OUT	Double	3
S4Pos_S	S4Pos_SD	IN	Float		3	J4	OUT	Double	
S5Pos_S	S5Pos_SD	IN	Float		~	J5	OUT	Double	
S6Pos_S	S6Pos_SD	IN	Float	Ļ		J6	OUT	Double	
Signal connect									10
-	Direction								
Inner signal S1Pos S	<-	Outer sign				of each in	a matically a mi		
			Ine	en	ect	of sequ	entially es	tablishing sigi	hais
S2Pos_S	<-	J2							
S3Pos_S	<-	J3							
S4Pos_S	<-	J4							
S5Pos_S	<-	J5							
	<-	J6							

Figure 12-20: Connection method

- Repeat steps for all remaining pairs of internal and external signals, connecting them according to the order shown in Figure 12-20 (1) -> (2) -> (3).
- This step allows the robot to simulate grabbing and dropping by controlling its internal signals through external signals.

Step 6: Simulation Verification

 In the "IRO 	BOTCAM" column, cl	ick the "Simul	ation" button. (Figure 12-	-21)
🕸 🗋 🌜 🔚 🗃 😂 🗠 🔿 🤝	•		ZW3D 2024 SP x64 Asser	nbly - [* Define Bas
File Shape Free Form Wirefr	a <u>me Direct</u> Edit Assembly Data Exchange	e Heal PMI Tools Vi	sualize Inquire IROBOTCAM App Simulation	
🗄 🖻 🖻 🗎	💿 💸 🗵 🖻	ે. 🖄 🛑	計 🖪 📔 🕥 🛞	1?
Model Import Export Close Save library	detection	Assembly Move Connect	Controller Progam Process Workspace Calibration edit	About Help
File	Simulation	Mechanical Electrical	Robot	Help

Figure 12-21: Simulation verification

- In the "Simulation Control" window, click the "Start Simulation" button on the far left side. (Figure 12-22, point ①)
- You will see the device server you added, identified by ②, along with its ③ external signals. (Figure 12-22)

	Wilstion		Connect.Z3 - (Default) or <shift-roll> to fir ft-right-click> to disp</shift-roll>	nd next valid filter se lay pick filter.	etting. 🕣 🧐	. 2 9 3	• @• • •	. H · ■· - □ ♥· 0 © EEC000 ·	Output [Connect.Z3] opened in 0.9560 second Done. Assembly [Default] activated. Init mechatronic data completed!	∎ (8) ds.
Setting Image: Secure Ris Secure Ris Secure Mode To Restrict Controller Simulation Image: Secure Ris	C Entity:		OuterSignal				7. 5			
• Senting: • Description: • multion: • Description: • Description: • Description: <td></td> <td> ¥ 👲</td> <td>-</td> <td></td> <td></td> <td>112 112 12</td> <td>100</td> <td></td> <td></td> <td></td>		¥ 👲	-			112 112 12	100			
Simulation Image: Severable Size radio Size regret Mode Severable Nodes Severable Size radio Size regret Mode Image: Severable Nodes Severable Nodes Image: Nodes Image: Nodes Image: Nodes Image: Nodes Image: Nodes Image: Nodes				mens Pic Smart	Modbus icp	BeckHoff Con	troller			
Simulation Simulation <td>(1)</td> <td></td> <td>V Device</td> <td></td> <td></td> <td></td> <td></td> <td></td> <td></td> <td></td>	(1)		V Device							
□ Dues togectory © Dues togectory © Outer signed ○	Simulation		(2))}		
Samplighting '9 •			ModbusTcp1	192.168.1.10	502	1				6
Improvinging * • Outer signal • Oute										
■ Record videe ● Pacerd videe	Sampling(ms) 50									
Image: Description Image: Distribution of the type of	Record video								> DATA1(D:)	
Image: Description Image: Distribution of the type of								Mar al	G	· ·
Name Offset Outrie Outrie <td></td> <td></td> <td>(3) • Outer signa</td> <td>el</td> <td></td> <td></td> <td></td> <td></td> <td>-</td> <td></td>			(3) • Outer signa	el					-	
12 00T 0udite 65540 Vis 1 13 00T 0udite 65544 Vis 1 14 00T 0udite 65584 Vis 1 14 00T 0udite 65584 Vis 1 1 0uthe 65584 Vis 1 1 Enclose 1 1 0uthe 65584 Vis Vis 1 <td< td=""><td></td><td></td><td>Name</td><td>IO type</td><td>Data type</td><td>Offset</td><td>ls joint 📤</td><td></td><td></td><td></td></td<>			Name	IO type	Data type	Offset	ls joint 📤			
13 OUT Double 65544 Yes 14 OUT Double 65548 Yes 15 OUT Double 65522 Yes 1 Image: Strate in the			л	OUT	Double	65536	Yes			
13 001 0xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx			12	OUT	Double	65540	Yes			
H OUT Double 65549 Yes 5 OUT Double 65522 Yes ><			В	OUT	Double	65544	Yes			
A OUT Double 6552 Vrs			14	OUT	Double	65548	Yes		Seatch	· Q 🏷
Protein			15				1000		▶ File List	
			1			03332	2.0		Preview	
			Save O	lpen			OK Apply			
			+	100						
			1		1	18				
			*		-					
		Yne	N			_	-			
				141	_			TAN TAN		
				C HP			C.			
							6			

Figure 12-22: Simulation verification

- The external signals added through ③ control the internal signals of ④ sequentially.
- Finally, the internal signals control the (5) joint positions and the programmed simulation sequence, successfully establishing an external communication function. (Figure 12-23)

SV.

XC	0 1	+ * Connect.Z3 - De				1.1		a 1991 A	[] 년 - 동 - = 🛛 🗢		[Connect.Z3] opened in 0.9	160 recents
	0 5	<right-dick> for con <shift-right-dick> to</shift-right-dick></right-dick>	text-sensitive optio	15.	-U @ .	· .	3. Q. I	1. M	M G. G G -	 A C) #3610000 	Done.	Sou seconos.
sme	Type 🔺	Signal conne							2	×	Assembly [Default] activate	
🝠 W2	Prismatic joint	· signal conne	4000							n	Init mechatronic data com	pleted!
🝠 W3	Prismatic joint	Signal connection					£1.0	malConnect1				
Material			Indine									
Coupling pairs		Project name					DefaultProject					
Sensors and actuators		Device name				1	ModbusTcp1			•		
GSKRobot	Robet						in dealing i					
📥 test11	Sensor (5)	▼ Signal										
😤 oonoiu	Robot tool	Name	Adapter name	IO type	Data type		Name	10 type	Data type			
S1Pos	Joint position					-					File Browser	
3 S2Pos	Joint position	qidong	null	OUT	Bool		31	OUT	Double		+ + + + 🗘	
S3Pos	Joint position	GSKRobot S	GSKRobot SD	IN	Bool		12	OUT	Double		> 📌 Favorites	
3 S4Pos	Joint position	OSKN000CS	OSKNOBOL SD	-	DOOI	- 22	12	001	Double		> Windows(C:)	
SSPos	Joint position	gongju_S	gongju_SD	104	Float	B	13	OUT	Double	=	> DATA1(D:)	
S6Pos	Joint position					07		in the second se			2 Deletion	
Runtime behavior		gongju_opt	null	IN	Bool		,4	OUT	Double			
gongju_catch_EXEC	Simulation sec	S1Pos_S	S1Pos SD	IN .	Float		15	OUT	Double			
🕒 gongju_pull_EXEC	Simulation sec	Siros_3	No. of Const.		P Iores		~					
Signal adapter		S2Pos_S	S2Pos_SD	IN	Float		36	OUT	Double			
SKRobot_SD	Signal adapter	N				¥				*		
🚛 gongju_SD	Signal adapter	Signal connec								577F		
S1Pos_SD	Signal adapter	aignal contrec	uon	3								
S2Pos_SD	Signal adapter	Inner signal	Direction	Outer sig	nal					A 1000		
S3Pos_SD	Signal adapter	gongju_opt		Clamp								
J. S4Pos_SD	Signal adapter	gong/u_opt		Clamp							▼ Search	
SSPos_SD	Signal adapter	S1Pos_S	<	11							Search	· Q 1
SSPos_SD	Signal adapter									- RI D DEAD		
Signal connection		S2Pos_S	<-	12							▶ File List	
SignalConnect1	Signal connect	S3Pos_S	11.0	J3							Preview	
Signal	Signal			1								
GSKRobot S	Signal	S4Pos_S	<-	34								
gongiu_S	Signal	SSPos_S		15								
u gongiu_opt	Signal	32P05_5		15								
S1Pos S	Signal		- 10			-						
L S2Pos S	Signal								OK Appl			
L S3Pos_S	Signal											
SAPos S	Signal	A A	S III	_			_					
1 SSPos_S	Signal	1.86	11	-	-	-	-				2	
S6Pos_S	Signal		IUU		-				- On			
Collision group			CALEP.					4	33		1	
HM			Call of						4			

Figure 12-23: Simulation verification

- These steps guide you through connecting your iRobotCAM model to an external controller.
- By following these instructions, you can enable external communication and control your robot's movements through external signals.

Chapter 13: Turntable Model

 This document presents a comprehensive guide for constructing a functional turntable model in IROBOTCAM, encompassing various elements and functionalities.(Figure 13-1)

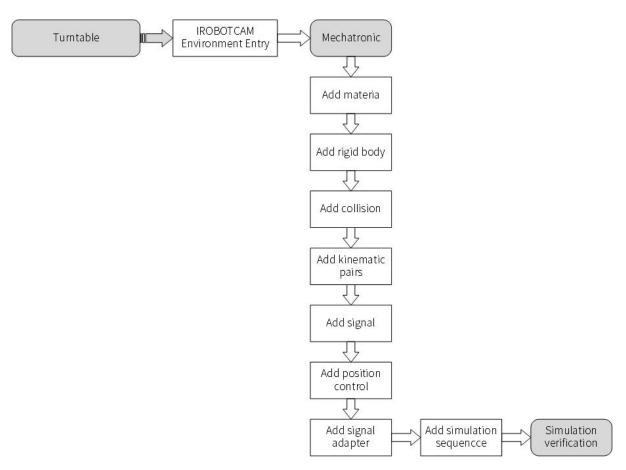


Figure 13-1: Flow chart of collision detection

Prerequisites

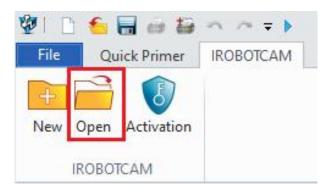
- Familiarity with IROBOTCAM interface and basic functionalities.
- Understanding of mechatronic concepts like rigid bodies, collisions, joints, signals, and simulation sequences.

Procedure:

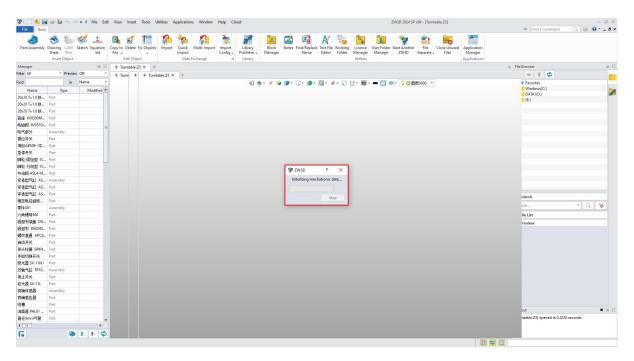
Step1: Model Import and Initialization

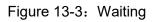
- Open iRobotCAM project file "Turntable.Z3" in IROBOTCAM.(Figure 13-2)
- Wait for the model import to complete.(Figure 13-3)
- Access the component details interface.(Figure 13-4)

Y









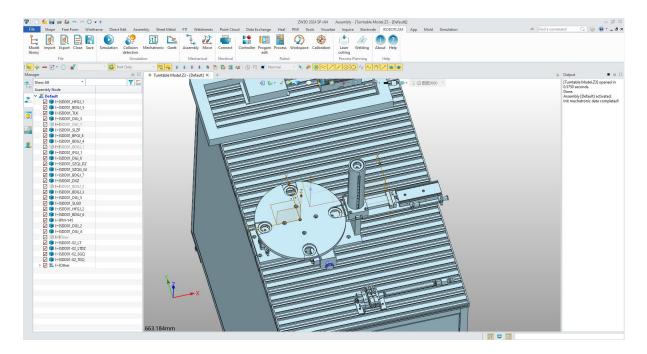


Figure 13-4: Component details

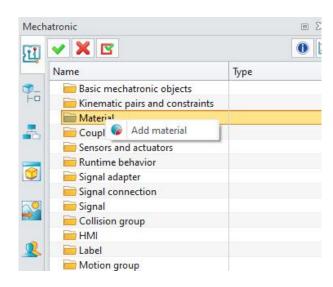
Step2: Material Definition

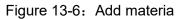
Y

- In the "IROBOTCAM" tab, click the "Mechatronic" button.(Figure 13-5)
- Open the "Materials" window.(Figure 13-6)
- Define two materials: "Material1" and "Material2".(Figure 13-7)
- Assign specific values for friction coefficient, restitution coefficient, linear and angular damping for each material.(Figure 13-7)

1 B	6 🔒	ii 1i	n 10	0 =	•										ZW3D 2	2024 SP x64	Assembly	- [Turntable	Model.Z3 - [Default]]		
File	Shape	Free F	orm	Wirefra	me Direct	Edit Ass	embly Sheet	t Metai	FTI We	dments	Point Clo	ud Data	Exchange	Heal	PMI To	ols Visualize	Inquir	e Electro	de IROBOTCAM	Арр	Mold	Simulation
Model library	Import	_		E Save	Simulation	Collision detection	Mechatronic	Gantt				Controller			Workspace	Calibration	Laser cutting	Welding	About Help			
		File				Simula	ation		Mecha	inical	Electrical			Robo	t		Process	Planning	Help			

Figure 13-5: Mechatronic button





👰 Material			? ×	🐲 Material				? ×
Property				▼ Property				
Frictional coefficient	1		<u> </u>	Frictional coefficient	0.01			- F
Restitution coefficient	0.5			Restitution coefficient	0.05			
Linear damping	0		=	Linear damping	0			
Angular damping	0			Angular damping	0			
Density	500		•	Density	500			
▼ Name				▼ Name				
m1				m2				
		_			1		1	
	OK	Apply	Cancel			OK	Apply	Cance

Figure 13-7: Materia1 and Materia2

Step3: Rigid Body Creation

- Click the "Mechatronic" button.(Figure 13-8)
- Use the "+" button to create eight rigid bodies corresponding to the main turntable components.(Figure 13-9)
- Specify their masses, material assignments, and property settings.(Figure 13-10)
- Create two additional rigid bodies ("SDD01DGJ_2_1" and "SDD01DGJ3_1").(Figure 13-11)
- Assign "Material2", enable "Physical Property" and "Collision Response" for these rigid bodies.(Figure 13-12)

1 D	6 🔒	ii 12	n	<> -	•										Z	W3D 2024 SP x	64 Assen	bly - [Turnta	able,Z3 - [Default]]			
File	Shape	Free F	orm	Wirefra	me Direct	Edit Asse	mbly Sheet	Metal	FTI We	ldments	Point Clo	ud Data i	Exchange	Heal	PMI	Tools Visua	lize Inqui	re Electro	de IROBOTCAM	Арр	Mold	Simulation
Model library		Export			Simulation	Collision detection	Mechatronic	Gantt	L. Assembly			Controller			Workspa	ce Calibration	Laser cutting	Melding	About Help			
		File				Simula	tion		Mecha	nical	Electrical			Robo	t		Proces	Planning	Help			

 \odot

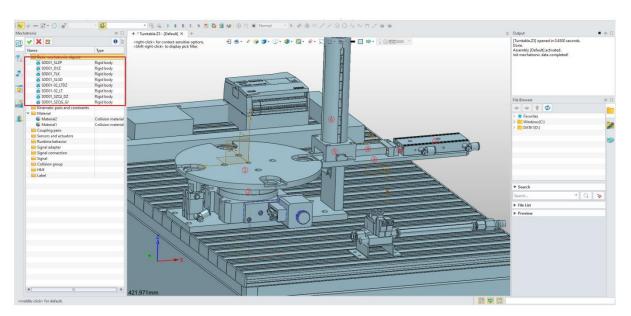


Figure 13-8: Mechatronic button

Figure 13-9: Add rigid body

Ny INSEE dia mampina mampina	·		?	×
Rigid body	SDD01_SLZP Mass laterial Default Property settings Physical property Collision response Support sensing Label			
SDD01_SLZP				+
Mass				
Material	Default			
Property se	ttings			
Collision r	esponse			
▼ Label				
None				

Figure 13-10: Rigid body message

Basic mechatronic objects	
SDD01_SLZP	Rigid body
SDD01_DXZ	Rigid body
SDD01_TLK	Rigid body
SDD01_SLGD	Rigid body
SDD01-02_LTDZ	Rigid body
SDD01-02_LT	Rigid body
SDD01_SZQJ_DZ	Rigid body
SDD01_SZQG_GJ	Rigid body
SDD01_DGJ_2_1	Rigid body
SDD01_DGJ_3_1	Rigid body

Figure 13-11: Two additional rigid body

🖉 Rigid body	?	🗙 🦉 Rigid I	oody			? ×
▼ Rigid body		▼ Rigid bo	ody			
SDD01_DGJ_2_1	-	SDD01_D	GJ_3_1			+
▼ Mass		▼ Mass				
Material M2	•	Material	M2		•	
Property settings		▼ Propert	y settings			
Physical property Collision response Support sensing		Collisio	al property on response rt sensing			
▼ Label		▼ Label				
None		+ None				•
						1
ОК	Apply Cano	cel		ОК	Apply	Cancel

Figure 13-12: Rigid body message

Step4: Collision Definition

- Click the "Mechatronic" button.(Figure 13-13)
- Open the "Basic Mechatronic Objects" window and select "Add Collision".(Figure 13-14)
- Create six collisions for the turntable components.(Figure 13-15)
- Choose appropriate collision shapes like "convex decomposition" or "convex hull".(Figure 13-16)
- Assign mass, material, and property settings for each collision.(Figure 13-16)

1 D	6	e 12	n 10	○ =	•										ZV	V3D 2024 S	SP x64	Assemb	ly - [* Turnta	ible.Z3 - [Default]]			
File	Shape	Free F	orm	Wirefram	me Direct	Edit Asser	mbly Sheet	Metal	FTI W	eldments	Point Clou	ud Data E	xchange	Heal	PMI	Tools V	isualize	Inquire	Electroc	le IROBOTCAM	Арр	Mold	Simulation
Model library					Simulation	Collision detection	Mechatronic				Connect	Controller			Workspace	e Calibra		Laser cutting	Market Welding	About Help			
		File				Simulat	ion		Mech	anical	Electrical			Robo	t			Process I	lanning	Help			

Figure 13-13: Mechatronic

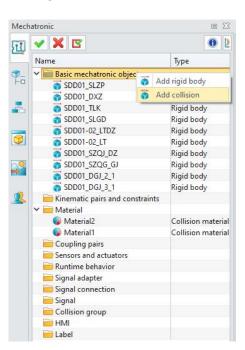


Figure 13-14: Add collision

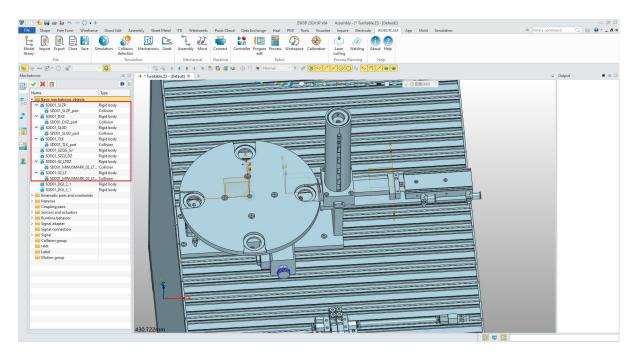


Figure 13-15: Six collision



	? ×	🔮 Collision ? 🗙	🔮 Collision 🤅 X
Collision Object		V Collision Object	▼ Collision Object
SDD01_SLZP	+	SDD01_DXZ	SDD01_TLK
▼ Shape		▼ Shape	▼ Shape
Collision shape	Convex decomposition *	Collision shape Convex hull	Collision shape Convex decomposition •
Material M1	•	Material Default -	Material Default -
Property settings		▼ Property settings	▼ Property settings
Physical property		Physical property	Physical property
Collision response		Collision response	Collision response
Support sensing		Support sensing	Support sensing
▼ Label		▼ Label	▼ Label
None		None	None
😨 Collision	? ×	🕲 Collision ? 🗙	
 Collision Object 		Collision Object	2 Collision2 ×▼ Collision Object
 Collision Object SDD01_SLGD 	+		
		Collision Object	▼ Collision Object
SDD01_SLGD		Collision Object SDD01-02_LTDZ	Collision Object SDD01-02_LT
SDD01_SLGD	+	Collision Object SDD01-02_LTDZ Shape	▼ Collision Object SDD01-02_LT ▼ Shape
SDD01_SLGD V Shape Collision shape	Convex decomposition	Collision Object SDD01-02_LTDZ Shape Collision shape Convex decomposition	✓ Collision Object SDD01-02_LT ✓ Shape Collision shape Convex decomposition *
SDD01_SLGD Shape Collision shape Material Default	Convex decomposition	Collision Object SDD01-02_LTDZ Shape Collision shape Convex decomposition Material Default *	Collision Object SDD01-02_LT Shape Collision shape Convex decomposition Material Default
SDD01_SLGD Shape Collision shape Material Property settings	Convex decomposition	Collision Object SDD01-02_LTDZ Shape Collision shape Convex decomposition Material Default Property settings	Collision Object SDD01-02_LT ✓ Shape Collision shape Convex decomposition Material Default ✓ Property settings
SDD01_SLGD Shape Collision shape Material Perperty settings Physical property	Convex decomposition	Collision Object SDD01-02_LTDZ Shape Collision shape Convex decomposition Material Default Property settings Physical property	Collision Object SDD01-02_LT Shape Collision shape Convex decomposition Material Default ▼ Property settings Physical property
SDD01_SLGD ▼ Shape Collision shape Material Default ▼ Property settings ■ Physical property ♥ Collision response ■ Support sensing	Convex decomposition	Collision Object SDD01-02_LTDZ Shape Collision shape Convex decomposition Material Default Property settings Physical property Collision response	Collision Object SDD01-02_LT Shape Collision shape Convex decomposition Material Default Vroperty settings Physical property Collision response
SDD01_SLGD Shape Collision shape Material Property settings Physical property Collision response	Convex decomposition	Collision Object SDD01-02_LTDZ Shape Collision shape Convex decomposition Material Default Property settings Physical property Collision response Support sensing	Collision Object SDD01-02_LT Shape Collision shape Convex decomposition Material Default Property settings Physical property Collision response Support sensing

Figure 13-16: Collison message

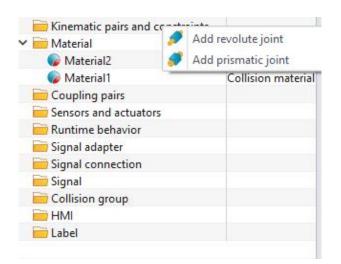
Step5: Kinematic Pair Definition

- Click the "Mechatronic" button.(Figure 13-17)
- Open the "Kinematic pairs and constraints" window and select "Add Revolute and Prismatic joints".(Figure 13-18)
- Create a revolute joint for the turntable base and specify the axis, start angle, limits, and motion type.(Figure 13-19)
- Create a prismatic joint (Cylinder_J1) for the cylinder and define the axis, offset, limits, and motion type.(Figure 13-20)

1 D	6 🔒		n //	0 =	•										Z	W3D 20	24 SP x64	Assembl	y - [* Turntat	le.Z3 - [Default]]			
File	Shape	Free F	orm	Wirefra	me Direct	Edit Ass	embly Sheet	Metal	FTI We	Idments	Point Clor	d Data I	Exchange	Heal	PMI	Tools	Visualize	Inquire	Electrode	IROBOTCAM	Арр	Mold	Simulation
Model					Simulation	Collision					Connect							Laser		About Help			
library	and barre	- Parts				detection							edit					cutting					
		File				Simul	ation		Mecha	nical	Electrical			Robo	ot			Process P	lanning	Help			

Figure 13-17: Mechatronic button

Y





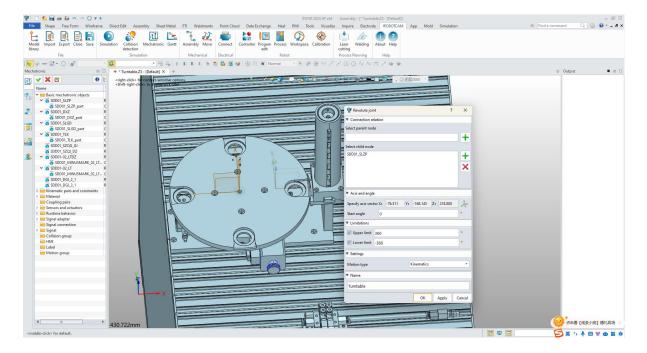


Figure 13-19: Revolute joint

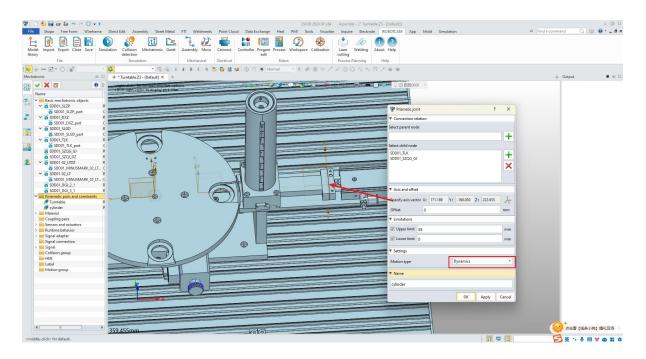


Figure 13-20: Prismatic joint

Step6: Output Signal Definition

Y

- Click the "Mechatronic" button.(Figure 13-21)
- Open the "Signal" window and select "Add Signal".(Figure 13-22)
- Create two output signals: "Turntable_Out" and "Cylinder_Out".(Figure 13-23)
- Specify "Out" as the IO type, assign names, and set port values.(Figure 13-23)

1 D	6 🗟		n 0	<> ₹	•										ZW	/3D 2024 SP >	64 Asser	mbly - [* Turn	table.Z3 - [Default]			
File	Shape	Free F	orm	Wirefran	ne Direct	Edit Asse	embly She	et Metal	FTI W	/eldments	Point Clo	ud Data	Exchange	Heal	PMI T	fools Visu	alize Inqu	uire Electro	de IROBOTCAN	App	Mold	Simulation
						Collision detection	Mechatroni				Connect					e Calibratio	n Laser cuttin	Welding	About Help			
		File				Simula	ation		Mech	nanical	Electrical			Robo	t		Proce	ss Planning	Help			

Figure 13-21: Mechatronic

🎯 Material2	Collision materia
🌍 Material1	Collision materia
🧰 Coupling pairs	
🧮 Sensors and actuators	
🧮 Runtime behavior	
🧰 Signal adapter	
🧰 Signal connection	
Signal	-
Collision grd	
📄 HMI	
🔤 Label	

Figure 13-22: Add signal

🐲 Signal		? ×	🐲 Signal		?)	×
Settings			▼ Settings			
Connect runtime param	ieters		Connect runtime para	meters		
Parameter name	Angle	•	Parameter name	Angle		٠
IO type	Out		IO type	Out		٠
Data type	BOOL	•	Data type	BOOL		
Dimension	Angle		Dimension	Angle		z
Unit	•	•	Unit	•		•
Initial value	false	•	Initial value	false		٠
▼ Name			▼ Name			
Signal name	Turntable_Out		Signal name	Cylinder_Out		
Controller		•	Controller			٠
Port	-1		Port	-1		
			1			
	OK Apply	y Cancel		OK Appl	y Cano	cel

Figure 13-23: Output signal

Step7: Position Control Implementation

- Click the "Mechatronic" button.(Figure 13-24)
- Open the "Sensors and actuators" window and select "Add Position Control".(Figure 13-25)
- Define position control for both "Turntable" and "Cylinder". (Figure 13-26/13-27)
- Associate the control with the corresponding kinematic pairs and rigid bodies.(Figure 13-26/13-27)
- Specify coordinate systems, signals, axis types, target positions, speeds, and acceleration limits for each control.(Figure 13-26/13-27)

10	6 🗟		n 0	0 =	•										ZV	V3D 2024 SP x	4 Assem	bly - [* Turnta	ble.Z3 - [Default]]			
File	Shape	Free	Form	Wirefra	me Direct	Edit Asse	mbly Sheet	Metal	FTI W	eldments	Point Clou	d Data	Exchange	Heal	PMI	Tools Visua	lize Inqui	re Electroo	le IROBOTCAM	Арр	Mold	Simulation
Model library	Import			Rave	Simulation	Collision detection	Mechatronic				Connect				Workspace	e Calibration	Laser cutting		About Help			
		File				Simula	tion		Mech	anical	Electrical			Robo	ot		Process	Planning	Help			

Figure 13-24: Mechatronic button

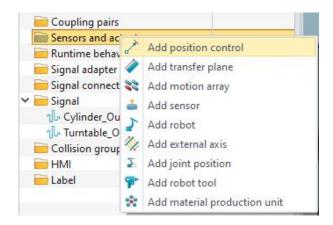


Figure 13-25: Add position control

S

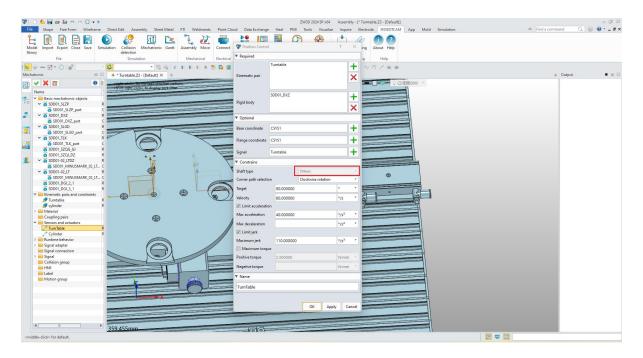


Figure 13-26: Turntable message

Image: State State Image: State State Stat	ipp Mold Simulat	tion	☆ Find	a command Q	- 9 2 • • • •
library detection edit cutting	V Position Control		2 X		
File Simulation Mechanical Bestrical Robot Process Planning Help Image: Process Planning Image: Planning </td <td>▼ Required</td> <td></td> <td></td> <td></td> <td></td>	▼ Required				
		cylinder		⇒ Output	■ m ⊠
VIX IS 0 2 Add source to the page data source)		cymac.	+ ×	• output	- 66 64
Image: Second		SDD01_SZQJ_DZ SDD01_SLGD	+ ×		
	▼ Optional				
	Base coordinate	CSVS2	+		
	Flange coordinate		+		
	Signal Constrains	cylinder	+		
	Shaft type	Air cylinder			
Sourd Juby - Kinematic pais and constraints Minematic pais and constraints Minematic pais and constraints Minematic pais and constraints	Target	0.000000	mm •		
Ginder P	Velocity	150	mm/s *		
> Material	Limit acceleration	n			
Coupling pairs Sensors and actuators	Max acceleration	39.999999	mm/s² *		
TumTable P Cylinder P	Max deceleration		mm/s² *		
	Limit jerk				
Kuntine behavior Signal adapter	Maximum jerk	10.000000	mm/s* *		
Signal connection	Maximum force				
Collision group	Positive force	50.000000	N *		
HM HA	Negative force		N -		
Metion group					
	▼ Name				
	Cylinder				
		OK	Apply Cancel		
1 359.455mm - Kolo				点击著 【线条小	海]婚礼现场 ×

Figure 13-27: Cylinder

Step 8: Signal Adapter Creation

- Click the "Mechatronic" button.(Figure 13-28)
- Open the "Signal Adapter" window and select "Add Signal Adapter".(Figure 13-29)
- Create two signal adapters: "Turntable_SD" and "cylinder_SD".(Figure 13-30)
- Assign the corresponding "Turntable" and "Cylinder" control signals.(Figure 13-30)

()

• Define data types, input/output direction, initial values, and formulas for each adapter.(Figure 13-30)

1 D	6	ii 12	n //	0 =	×										Z	W3D 20	24 SP x64	Assemb	ly - [* Turnta	ble.Z3 - [Default]]			
File	Shape	Free F	orm	Wirefra	me Direct	Edit Ass	embly She	eet Metal	FTI W	eldments	Point Clo	ud Data I	Exchange	Heal	PMI	Tools	Visualize	Inquire	Electrod	e IROBOTCAM	App	Mold	Simulation
							Mechatron													About Help			
library	impon	Export	Close	Jave	Simulation	detection		ic pariti	Assembly	Nove	connect	Controller	edit	Process	workspa	ice cai	libration	cutting	weiding	About Help			
		File				Simul	ation		Mech	anical	Electrical			Robo	t			Process I	lanning	Help			

Figure 13-28: Mechatronic button

Material	
Coupling pairs	
Sensors and actuators	
Runtime behavior	
Signal ada 😳 🕹 Signal adapt	
Signal cor de Add signal adapt	er
Signal	
Collision group	
HMI	
Label	
Motion group	



😵 Signal adapte	ir.					7 ×	😨 Signal adapter						7 ×
▼ Signal							▼ Signal						
Assign as	Name	Data type	in/Out		Initial value		Assign as	Name	Data type	In/O	ut	Initial velue	+
Turntable	Turntable_S	Float	IN	0		+ ×	cylinder	cylinder_S	Float	IN	0		
													X
▼ Formula Assign as Turntable_S	Turntable_S	Formula			Note	+	▼ Formula Assign as cylinder_5	cylinder_5	Formula			Note	+ ×
						×							×
Formula							Formula						
▼ Name							▼ Name						
Turntable_SD							cylinder_SD						
					ОК	Apply Cancel						ОК	Apply Cancel

Figure 13-30: Two signal adapter

Step 9: Input Signal Definition

- Click the "Mechatronic" button.(Figure 13-31)
- Open the "Signal" window and select "Add Signal".(Figure 13-32)

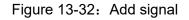
V

- Create input signals for both "Turntable" and "Cylinder".(Figure 13-33)
- Specify "In" as the IO type, assign names, and set port values.(Figure 13-34)

🕸 I 🗅	6 🔒	ii 1i	n 0	0 =	×										ZV	W3D 2024 SP x6	Assem	oly - [* Turnt	able.Z3 - [Default]]			
File	Shape	Free F	orm	Wirefra	me Direct	Edit Ass	embly She	et Metal	FTI W	eldments	Point Clou	id Data	Exchange	Heal	PMI	Tools Visua	ze Inquir	e Electro	de IROBOTCAM	Арр	Mold	Simulation
																			1?			
Model library	Import	Export	Close	Save	Simulation	Collision detection		c Gantt	Assembly	y Move	Connect	Controller	Progam edit	Process	Workspac	ce Calibration	Laser cutting	Welding	About Help			
		File				Simul	ation		Mech	anical	Electrical			Robo	t		Process	Planning	Help			

Figure 13-31: Mechatronic button

🔁 Signal adapter	
抗 Turntable_SD	Signal adapter
👫 cylinder_SD	Signal adapter
🚞 Signal connection	
🚰 Signal	
1 Cylinder_Out 1 Add signa	l și
ျမှ Turntable_Out	Signal
ျပီ Turntable_S	Signal
ျမာ cylinder_S	Signal
🧰 Collision group	
🚔 HMI	
📄 Label	



Cylinder_Out	Signal
Turntable_Out	Signal
ျ <mark>ြာ Turntable_S</mark>	Signal
ղ <mark>և cylinder_</mark> S	Signal
լի Turntable_In1	Signal
1 Turntable_In2	Signal
1 Turntable_In3	Signal
1 Cylinder_In1	Signal
1 Cylinder_In2	Signal
1 Cylinder_In3	Signal
Collision group	
HMI	
Label	

Figure 13-33: Six signal

		1	? ×
ters			
Ang	gle		•
[In		
	BO	OL	Ŧ
	Ang	gle	
	•		•
		false	-
Turr	ntable_	In1	
			-
-1			
	Ang	Angle In BO Ang •	ters Angle In BOOL Angle • false

Figure 13-34: Signal message

Step10: Simulation Sequence Design

- Click the "Mechatronic" button.(Figure 13-35)
- Open the "Runtime behavior" window and select "Add Simulation Sequence".
- Create three simulation sequences: (Figure 13-36) (Figure 13-37)
- ☆ "Turntable_EXEC1": activates the turntable to reach 90 degrees within 5 seconds based on specific input and output signals.
- ☆ "Cylinder_EXEC1": moves the cylinder to reach 58 units within 5 seconds based on the turntable's position.
- ♦ "Cylinder_EXEC2": activates the cylinder to reach 58 units within 5 seconds based on another input signal.

2	10	6 🔒		n 0	0=	×										ZV	V3D 2024 S	SP x64	Assembly	y - [* Turntab	le.Z3 - [Default]]			
	File	Shape	Free	Form	Wirefra	me Direct	Edit Ass	embly She	et Metal	FTI V	Veldments	Point Clou	ud Data I	Exchange	Heal	PMI	Tools Vi	isualize	Inquire	Electrode	IROBOTCAM	Арр	Mold	Simulation
	Model					Simulation	Collision detection	Mechatron				Connect					e Calibra	ation			1 ? About Help			
1	library		File				Simu			Mech	hanical	Electrical		eait	Robo	ot			Process P	lanning	Help			

Figure 13-35: Mechatronic button

Turntable	Position control
🖍 cylinder	Position control
Runtime beh	10 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
📄 Signal adapte 🎦 Add s	imulation sequence
抗 Turntable_SD	Signal adapter
號 cylinder_SD	Signal adapter

😨 Simulation seque	nce ?	×	👰 Simulation seque	ince	? ×	👹 Simulation seque	ence	? ×
Execution time			Execution time			Execution time		
Start time	0	s	Start time	0	s	Start time	0	s
Duration	5	s	Duration	5	s	Duration	5	s
Signal configuration			 Signal configuration 			 Signal configuration 		
Activation signal	Turntable_In1	+	Activation signal		+	Activation signal	Cylinder_In1	+
Activation condition	True •		Activation condition	True		Activation condition	True	
📝 End signal	Cylinder_In1	+	🗹 End signal	Turntable_In1	+	🗹 End signal	Turntable_In2	+
Execution signal	Turntable_S	+	Execution signal	cylinder_S	+	Execution signal	cylinder_S	+
End position	90]•	End position	58		End position	58	
▼ Name			▼ Name			▼ Name		
Turntable_EXEC1			Cylinder_EXEC1			Cylinder_EXEC2		
	OK Apply	Cancel		OK Ap	ply Cancel	-	ОК	Apply Cancel

Figure 13-36: Add simulation sequencce

Figure 13-37: simulation sequencce message

Step11: Simulation and Verification

- Click the "Simulation" button.(Figure 13-38)
- Click the "Start" button to initiate the turntable operation. (Figure 13-39)



Figure 13-38: Simulation button

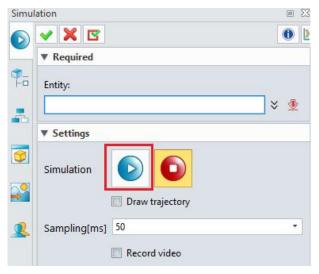


Figure 13-39: Start button

Chapter 14: Laser Cutting

Introduction

- Laser cutting is a technology that uses a laser beam to cut materials. It has the advantages of fast cutting speed, high precision, smooth cutting edge and small heat affected zone. It is widely used in sheet metal processing, electronic components, mechanical manufacturing, automobile manufacturing, aerospace and other fields.
- The software provides laser cutting simulation function, users can simulate the cutting process, view the cutting effect.

This guide provides step-by-step instructions for laser cutting in IROBOTCAM software.

Step 1: Open the iRobotCAM project file

 Open the iRobotCAM project file named "Laser Cutting.Z3" to enter the IROBOTCAM environment.

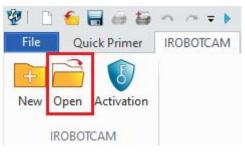


Figure 14-1: Open button

• The interface after opening is shown in the figure.

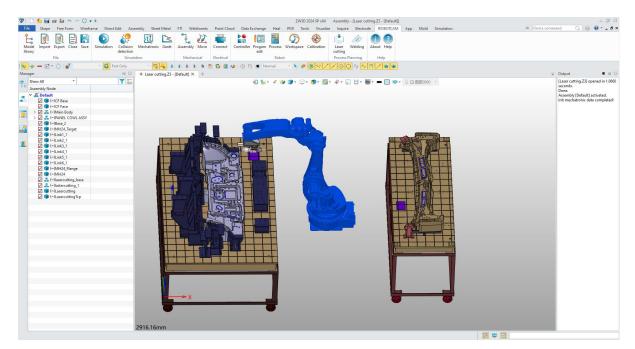


Figure 14-2: Interface details

Step 2: Add collision detection

• Click "IROBOTCAM" to enter the mechatronic interface.

1 B	6	iii 12	n //	0 =	•										ZW3	D 2024 SP x64	Assemb	y - [Laser cut	ting.Z3 - [Default]]			
File	Shape	Free Fo	orm	Wirefran	ne Direct	Edit Asse	embly Sheet	t Metal	FTI W	eldments	Point Clou	d Data	Exchange	Heal	PMI To	ools Visualize	Inquir	e Electroc	le IROBOTCAM	Арр	Mold	Simulation
Ŀ	80	SF.	E		\bigcirc	\$	1	2	٦.	11			13			•	١	ß				
Model library	Import	Export	Close	Save	Simulation	Collision detection	Mechatronic	Gantt	Assembly	/ Move	Connect	Controller	Progam edit	Process	Workspace	Calibration	Laser cutting	Welding	About Help			
		File				Simula	ation		Mech	anical	Electrical			Robo	ot		Process	Planning	Help			

Figure 14-3: mechatronic button

- In the collision group window, select the collision body of the workpiece to create a collision group.
- The support induction checkbox needs to be selected in the collision body window of the workpiece.

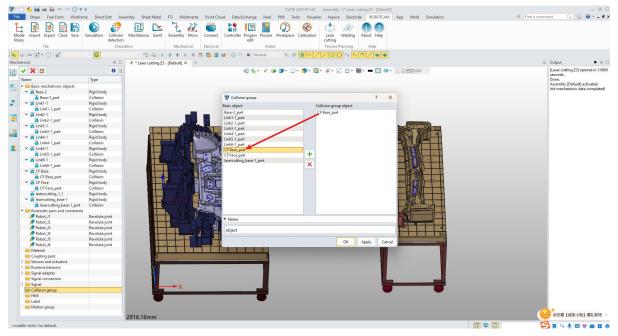


Figure 14-4: Workpiece collision group

- In the collision group window, select the collision body of the robot and the collision body of the robot tool to create a collision group.
- The support induction checkbox needs to be selected in the collision body window of the workpiece and the tool.

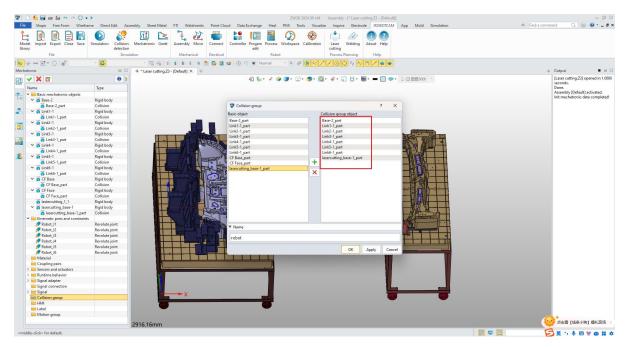


Figure 14-5: Robot and robot tool collision group

- Click the collision detection button.
- Select "robot" as the target object. Select "object" as the obstacle.
- Click the "+" button to add.

Y

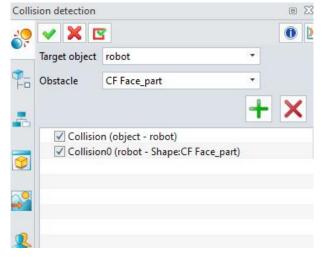


Figure 14-6: Collision detection

Step 3: Add curve list

- Double-click to activate the component to be cut.
- Click the wireframe button to enter the curve list window.
- Select the edge to be cut to create a curve list.

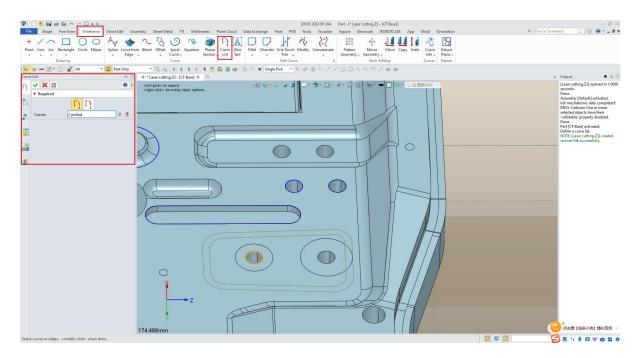


Figure 14-7: Curve list

Step 4: Laser cutting - Select path

• Click the laser cutting button.

Y

21 D	6 🔒	e 18	n 10	<> ₹	•										ZW3	0 2024 SP x64	Assemi	oly - [* Laser	cutting.Z	Z3 - [Default]]			
File	Shape	Free F	orm	Wirefram	ne Direct	Edit Asse	mbly She	et Metal	FTI V	Veldments	Point Clor	ud Data	Exchange	Heal	PMI T	ools Visu	lize Inqu	ire Electr	rode I	ROBOTCAM	Арр	Mold	Simulatio
£:	ST I	-	L=X			<u> </u>		E							0	•		D	1	?			
Model library	Import	Export	Close	Save	Simulation	Collision detection	Mechatron	ic Gantt	Assemb	ly Move	Connect	Controlle	er Progam edit	Process	Workspace	e Calibratio	Laser cutting	Welding	Abou	rt Help			
		File				Simula	tion		Mec	hanical	Electrical			Robo	ot		Proce	s Planning	F	Help			

Figure 14-8: Laser cutting

- In the "Select Path" window, click the Select Part button to select the workpiece to be cut.
- Drop down to select the robot controller.
- In the Select Path bar, first add all the curve lists of the edges to be cut by clicking the Add button. (Note that clicking the Add button once can only add one curve list)

Y

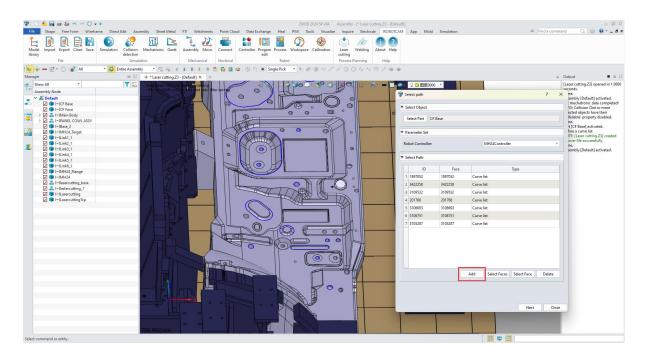


Figure 14-9: Select path - Add curve list

- Add all the faces where the cutting edge curve list is located.
- After adding all the curve lists, click all the faces where the curve lists are located, and select all of them by pressing the CTRL key.
- Then click the Select Multiple Faces button. After adding, if the output box does not prompt an error, the addition is successful.

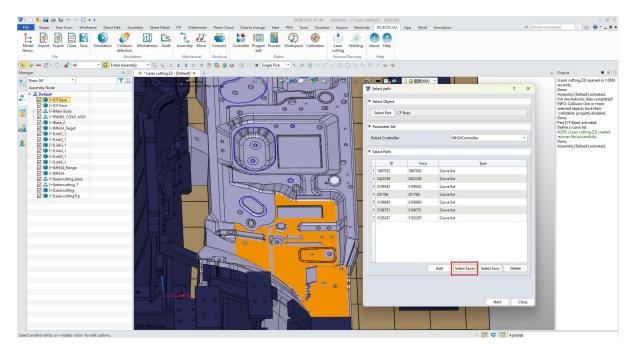


Figure 14-10: Select multiple faces

Step 5: Path Discretization

• Click "Next".

V

- The default value for the chord height ratio.
- Click the Discretization button to generate discrete points in the Discretization window.
- Click Show Discretization Result to view the coordinate system direction of the discrete points.

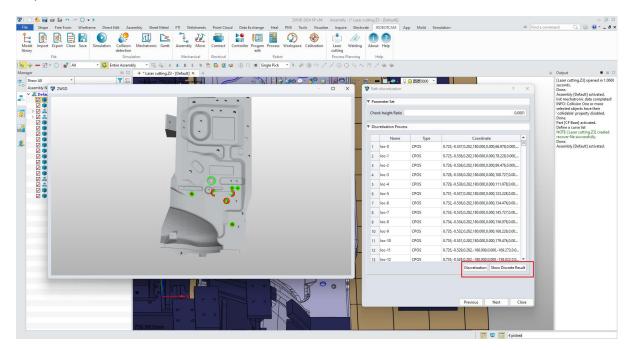


Figure 14-11: Path discretization

Step 6: Transition Point Sorting

- Click "Next".
- Click the "process sort calculate" button.
- View the laser cutting order in Process sorting calculation chart.

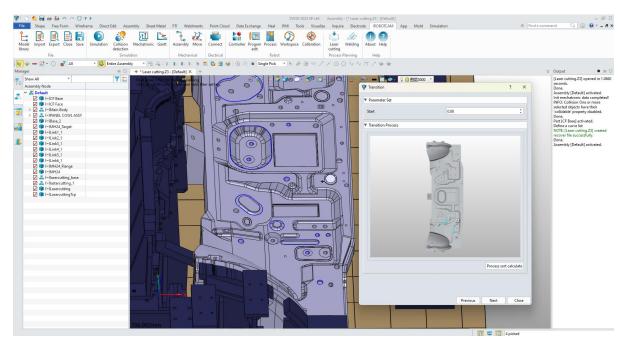


Figure 14-12: Transition point sorting

Step 7: Path Optimization

- Click "Next".
- In the Raise Gun Height bar, enter the raise gun height value.
- Click the "edit program" button to view the coordinate position of the raise gun height after discretization.
- Click the Calculate button to automatically list the three detection tables of Reachable, Singular and Collision.
- Finally, click the Optimize Path button. In the Optimize Path window, click the "Change fill data" and "Path optimize" buttons in sequence.
- In the Optimize Path window, click Modify Fill Data. Red represents the collision detection.

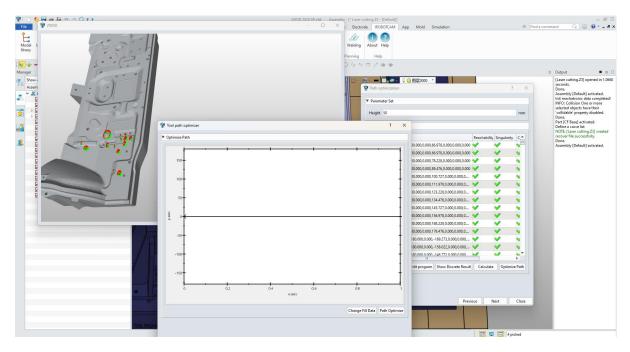
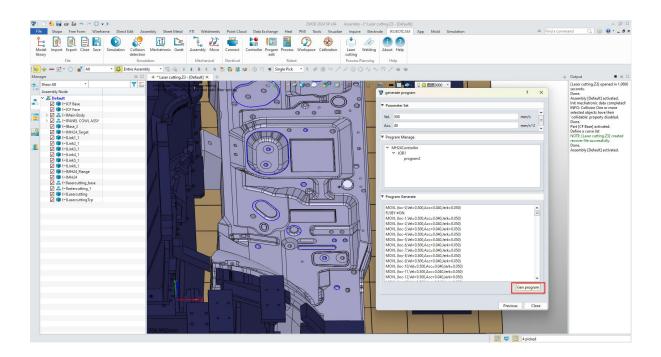


Figure 14-13: Path optimization

Step 8: Generate program

- Click "Next".
- The default values for speed and acceleration are set in the Generate Program window.
- Under the Program Management bar, add JOB and Program programs under Program Editing.
- In the Program Generation bar, click the "Generate program" button.



Step 9: Simulation verification

Y

- If the program has added a simulation sequence, you can click the simulation button to verify it. If not, you can click Run in the program editing window to view the laser cutting effect.
- During laser cutting, if a collision is detected, the target object will be displayed in a highlighted manner.

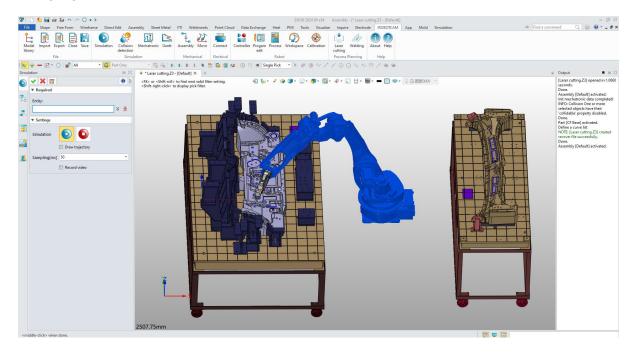


Figure 14-14: Simulation verification

The software provides two cutting workpieces in the scene, users can try the laser cutting function by themselves.

Chapter 15: Welding

Introduction

Welding is a forming method that uses heating or pressurizing, or both, to cause two separate objects to join together by interatomic bonding. Welding is an essential process in manufacturing and is widely used in mechanical manufacturing, construction, ships, aerospace, electronics and other fields.

The software provides welding simulation function, users can simulate the welding process, view the welding effect.

This guide provides step-by-step instructions for welding in IROBOTCAM software.

Step 1: Open the iRobotCAM project file

Open the iRobotCAM project file named "Welding.Z3" to enter the IROBOTCAM environment.

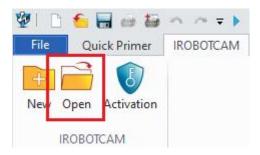


Figure 15-1: Open button

• The interface after opening is shown in the figure.

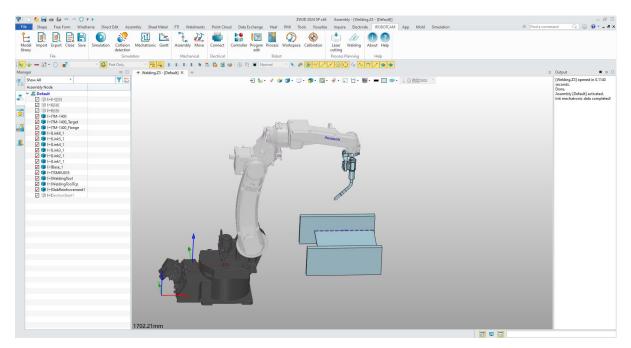


Figure 15-2: Interface details

\mathbf{Q}

Step 2: Add collision detection

• Click "IROBOTCAM" to enter the mechatronic interface.

1 D	6 🔒	iii 12	n 0	0 =	•										ž	ZW3D 2024	SP x64	Assem	bly - [Weldi	ng.Z3 - [Default]]			
File	Shape	Free F	orm	Wirefram	me Direct	Edit Ass	embly Sh	eet Metal	FTI W	eldments	Point Clo	ud Data	Exchange	Heal	PMI	Tools Vi	sualize	Inquire	Electro	de IROBOTCAM	Арр	Mold	Simulation
Model library	Import				Simulation	Collision detection	Mechatron				Connect					-	tion		Melding	About Help			
instatiy	File					Simul			Mech	anical	Electrical		cuit	Robo	ot			Process F	lanning	Help			

Figure 15-3: mechatronic

• In the collision group window, select the collision body of the workpiece to create a collision group. The support induction checkbox needs to be selected in the collision body window of the workpiece.

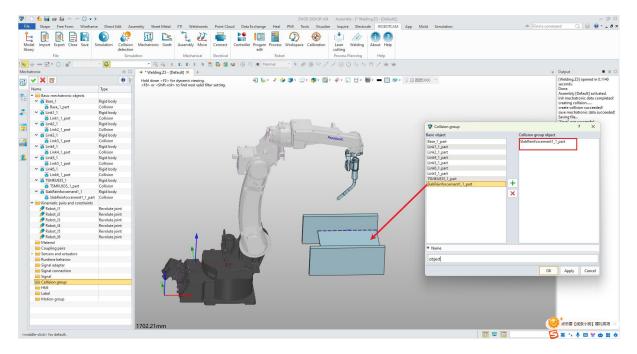


Figure 15-4: Workpiece collision group

• In the collision group window, select the collision body of the robot and the collision body of the robot tool to create a collision group. The support induction checkbox needs to be selected in the collision body window of the workpiece and the tool.

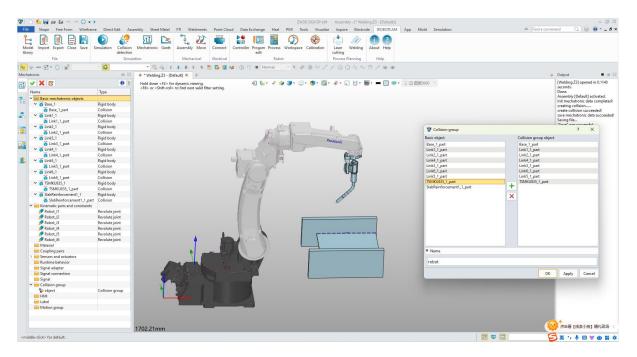


Figure 15-5: Robot and robot tool collision group

- Click the collision detection button.
- Select "robot" as the target object. Select "object" as the obstacle.
- Click the "+" button to add.

Y

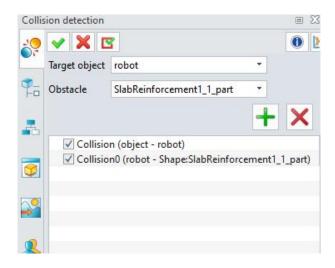


Figure 15-6: Collision detection

Step 3: Add weldments

- Click the weld button.
- Click Fillet Welding. In the Fillet Weld bar, select two faces respectively and enter the weld leg width.
- The weld function can be used under a specific ZW3D authorization code.

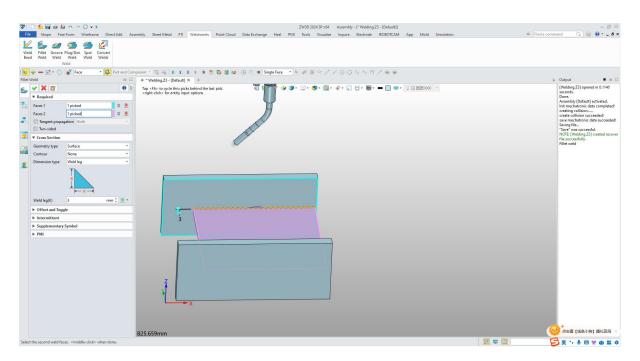


Figure 15-7: Add weld

Step 4: Welding - Select path

• Click the welding button.

Y

10	🗏 🗋 🌜 🟭 🧀 🖉 🔿 🗠 🔇 🖘 Markov ZW3D 2024 SP x64															x64 As	Assembly - [* Welding.Z3 - [Default]]						
File	Shape	Free F	orm	Wirefran	ne Direct	Edit Ass	embly She	et Metal	FTI W	eldments	Point Clou	ud Data E	xchange	Heal	PMI	Tools Vis	ualize Ir	nquire	Electrode	IROBOTCAM	Арр	Mold	Simulation
Ŀ	B	SE .	E)		\bigcirc	\$	1	2	٦ <u>-</u>	11			🗐		0	۲	E	5	ß	1?			
Model library	Import	Export	Close	Save	Simulation	Collision detection	Mechatroni	c Gantt	Assembly	/ Move	Connect	Controller	Progam edit	Process	Workspac	e Calibrat	on La: cut	ser ting	Welding	About Help			
	File				Simulation				Mechanical		Electrical	Robot					Pro	cess P	lanning	Help			

Figure 15-8: Welding

- In the "Select Path" window, click the Select Part button to select the workpiece to be welded.
- Drop down to select the robot controller.
- In the Select Path bar, first add the generated weld face by clicking the Add button.

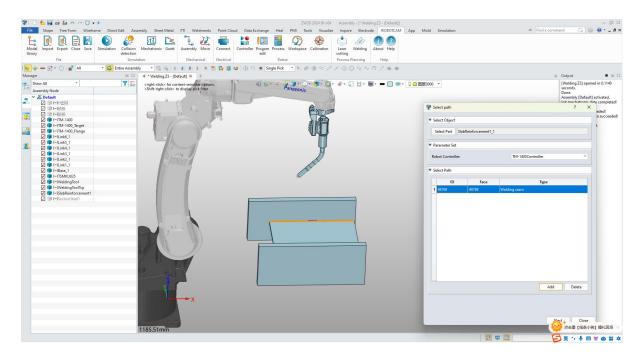


Figure 15-9: Select path

Step 5: Path Discretization

• Click "Next".

Ś

- The default value for the chord height ratio.
- Click the Discretization button to generate discrete points in the Discretization window.
- Click Show Discretization Result to view the coordinate system direction of the discrete points.

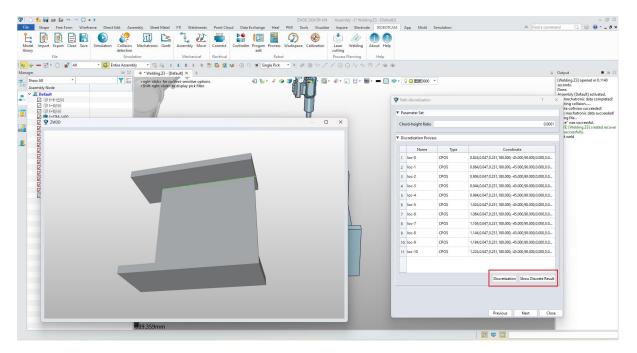


Figure 15-10: Path discretization

Step 6: Transition Point Sorting

- Click "Next".
- Click the "process sort calculate" button.
- View the welding order in the process sort calculate chart.

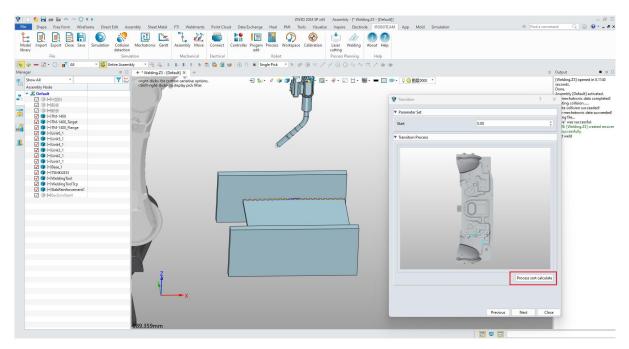


Figure 15-11:process sort calculate

Step 7: Path Optimization

- Click "Next".
- In the Raise Gun Height bar, enter the raise gun height value.
- Click the "edit program" button to view the coordinate position of the raise gun height after discretization.
- Click the Calculate button to automatically list the three detection tables of Reachable, Singular and Collision.
- Finally, click the Optimize Path button. In the Optimize Path window, click the "Change fill data" and "Path optimize" buttons in sequence. (It takes about 1 minute to load the modified filling data)
- In the Optimize Path window, click Modify Fill Data. Red represents the collision detection.

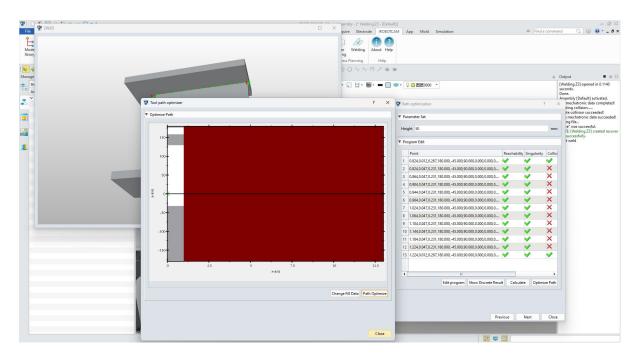


Figure 15-12: Path optimization

Step 8: Generate program

• Click "Next".

V

- The default values for speed and acceleration are set in the Generate Program window.
- Under the Program Management bar, add JOB and Program programs under Program Editing.
- In the Program Generation bar, click the "generate program" button.
- Click the Close button.

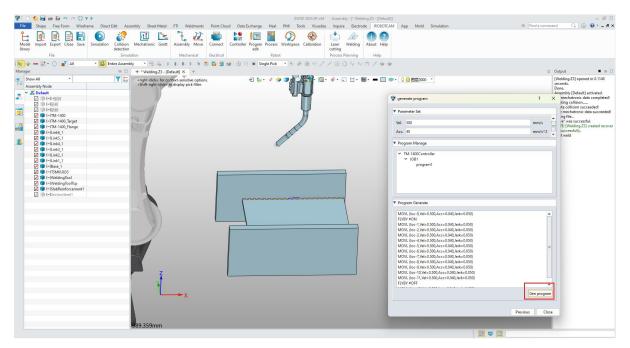


Figure 15-13: Generate program

Step 9: Simulation

Ś

- If the program adds a simulation sequence, you can click the simulation button to verify. If not available, click on run in the program editing window to view the effect of laser cutting.
- During the welding process, if a collision is detected, the target object will be highlighted.

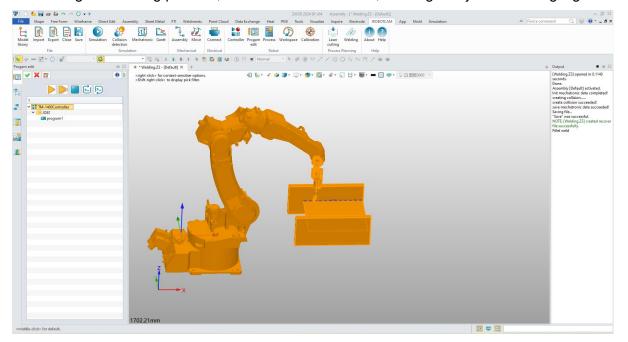


Figure 15-14: Simulation