ADDVERB

SYNCRO 5

COBOT

Hardware Manual

Addverb's Advanced Robotics Solutions

material handling, assembly, packaging, machine tending, quality control, and even dispensing and finishing tasks.

Disclaimer

The information contained in this document is confidential, privileged and only for information of the intended recipients and may not be used, published, or redistributed without prior written consent from the Addverb Technologies Ltd.

Nothing in this document constitutes a guaranty, warranty, or license either expressed or implied. Addverb Technologies Ltd. retains the right to make changes to this document at any time without notice. Every precaution has been taken to prepare this document. Nevertheless, Addverb Technologies Ltd. makes no warranty for the use of this document and assumes no responsibility for any errors that can appear in the document, nor does it make a commitment to update the information contained herein.

- Addverb makes no representations or warranties regarding the accuracy, completeness, or reliability of the information contained in this user guide.
- Addverb shall not be held responsible for any damages, direct or indirect, arising from the use or inability to use the Syncro 5 cobot, including any incidental or consequential damages.
- Any modifications or repairs carried out by unauthorized personnel will void the warranty and may result in damage or injury.
- Addverb is not responsible for any third-party software or accessories used in conjunction with the Syncro 5 cobot.

Revision History

Revision No.	Manual Name	Date	Comment
0.0	Syncro 5 cobot User Manual	August, 2025	Original Version

Table of Contents

DISCLAIMER 2

REVISION HISTORY 3

TABLE OF CONTENTS 4

ABOUT THE DOCUMENT 7

- PURPOSE 7
- SCOPE 7
- AUDIENCE 7
- SUPPORT 7
- RESOURCES 8

CHAPTER 1: SAFETY AND COMPLIANCE 9

- 1.1 VALIDITY AND RESPONSIBILITY 9
- 1.2 LIMITATION OF LIABILITY 9
- 1.3 TERMS AND CONDITIONS 9
- 1.4 GENERAL WARNING AND CAUTION MESSAGES 10
- 1.5 GENERAL SAFETY GUIDELINES 11
- 1.6 RISK ASSESSMENT 12
- 1.7 PRE-USE SAFETY CHECKS 12
- 1.8 WARNINGS AND POTENTIAL HAZARDS 13
- 1.9 INTENDED USE 13
- 1.10 MAINTENANCE SAFETY PRECAUTIONS 13

CHAPTER 2: INTRODUCTION TO SYNCRO 5 COBOT 14

- 2.1 OVERVIEW 14
- 2.2 KEY FEATURES OF SYNCRO 5 COBOT 15
  - *2.2.1 Six Degrees of Freedom 15*
  - *2.2.2 Advanced Design and Construction 15*
  - *2.2.3 Intuitive Control System 15*
  - *2.2.4 Collaborative Safety Features 15*
  - *2.2.5 Enhanced Adaptability and Longevity 15*
- 2.3 WHAT'S IN THE BOX 16

CHAPTER 3: HARDWARE REQUIREMENTS 17

- 3.1 PHYSICAL COMPONENT OVERVIEW 17
- 3.2 HARDWARE SPECIFICATIONS 18
- 3.3 ENVIRONMENTAL CONDITION 18
- 3.4 EMERGENCY STOP 19
  - *3.4.1 Using the Emergency Stop 19*
  - *3.4.2 Resetting the Emergency Stop 19*
  - *3.4.3 Recovering from Emergency Conditions 20*
- 3.5 CONTROL BOX DETAILS 20
  - *3.5.1 Precautions and Safety Measures 20*
  - *3.5.2 Control Box LED Indicators 24*

CHAPTER 4: ELECTRICAL REQUIREMENTS 25

- 4.1 ELECTRICAL WARNING AND CAUTION 25
- 4.2 ELECTRICAL CABLES AND CONNECTOR LIST 25
- 4.3 ELECTRICAL SPECIFICATIONS 27
- 4.4 ELECTRICAL CONNECTION ILLUSTRATION 27

CHAPTER 5: INSTALLATION OF SYNCRO 5 COBOT 28

5.1 WARNING AND CAUTION MESSAGES 28
5.2 WORKSPACE AND OPERATING SPACE 29
5.2.1 Mechanical Dimension of Syncro 5 cobot 29
5.2.2 Syncro 5 cobot Mounting Specification 30
5.3 SECURING THE SYNCRO 5 COBOT 30
5.3.1 Load and Torque Criteria 32
5.4 PLACING CONTROL BOX 33
5.5 CONNECTING THE SYNCRO 5 COBOT TO CONTROL BOX 34
5.5.1 Electrical Safety Requirements 34
5.5.2 Electrical Connection Illustration 35
5.5.3 Making the Connection 36

CHAPTER 6: GETTING STARTED 38

6.1 POWERING ON/OFF THE SYNCRO 5 COBOT 38
6.1.1 Preparation before Power Up 38
6.1.2 Powering on the Cobot System 38
6.2 CONNECTING MOUNTING TOOL 38

CHAPTER 7: RECOMMENDED STORAGE CONDITIONS 40

7.1 TEMPERATURE LIMITS 40
7.2 STORAGE BEST PRACTICES 40

CHAPTER 8: PACKING AND TRANSPORTATION 42

8.1 PRECAUTION AND SAFETY MEASURES 42
8.2 TRANSPORTATION DETAILS 42
8.2.1 Transportation Box Specifications 43
8.2.2 Preparation for Transport 43
8.2.3 Handling Instructions 44
8.2.4 Transport Mode 44
8.2.5 Loading and Unloading 44
8.2.6 Environmental Considerations 44

CHAPTER 9: ROBOT CLEANING AND INSPECTION 45

9.1 INSPECTION CHECKLIST 45
9.2 VISUAL INSPECTIONS 47
9.3 CONTROL BOX 48
9.4 ROBOT CLEANING 48
9.4.1 Cobot Cleaning 48
9.4.2 Control Box and Teach Pendant Cleaning 48

CHAPTER 10: DISCARD AND DISPOSAL 49

INSTRUCTIONS 49

10.1 END OF SERVICE LIFE CONSIDERATIONS 49
10.2 DISPOSAL OF ELECTRONIC COMPONENTS 49
10.3 MECHANICAL PARTS AND FRAME 50
10.4 COMPLIANCE WITH LOCAL AND INTERNATIONAL STANDARDS 50
10.5 MATERIAL GROUP DISPOSAL METHODS 50
10.6 ENVIRONMENTAL CONDITION 51

CHAPTER 11: COMMON ISSUES AND RESOLUTIONS 52

CHAPTER 12: FREQUENTLY ASKED QUESTIONS 54

CHAPTER 13: WARRANTY DETAILS 55

13.1 WARRANTY COVERAGE 55

13.2 CUSTOMER OBLIGATIONS 55
13.3 EXCLUSIVES FROM WARRANTY 55
13.4 WARRANTY CLAIM PROCESS 56
13.4.1 Limitations of Liability *56*
13.4.2 Additional Notes *56*

CHAPTER 14: UNDERSTANDING SOFTWARE 58

14.1 COBOT SOFTWARE INSTALLATION 58
14.2 USER SYSTEM SETUP AND COMPILATION 58
14.3 CONFIGURATIONS 58
14.3.1 Payload *58*
14.3.2 Safety type *59*
14.4 EXECUTION 60
14.5 CONTROLLERS OVERVIEW 60
14.6 AVAILABLE CONTROLLERS 61
14.6.1 Recorder Controller *61*
14.6.2 Free Drive Controller *63*
14.6.3 Joint Jogging Controller *63*
14.6.4 Cartesian Jogging Controller *65*
14.6.5 Joint Impedance Controller *66*
14.6.6 Cartesian Impedance Controller *68*
14.6.7 Velocity Controller *70*
14.6.8 PTP Joint Controller *71*
14.6.9 PTP TCP Controller *72*
14.6.10 Effort Controller *74*
14.6.11 Gravity Compensation Effort Controller *75*
14.6.12 Gripper Controller *76*
14.7 SHUTDOWN SERVICE AND ERROR RECOVERY 78
14.8 READ FT SENSOR DATA 79
14.9 TROUBLESHOOTING 80
14.9.1 If the controllers are not working *80*
14.9.2 If gripper is not working *80*
14.9.3 Scripts on the Cobot not working *80*
14.9.4 Unable to connect to the Cobot *80*

About the Document

Purpose

The purpose of this user guide is to provide comprehensive instructions for the safe, efficient, and effective operation of the Syncro 5 cobot. It serves as a reference for understanding the cobot's features, installation, maintenance, and troubleshooting. This guide ensures users can maximize the cobot's potential while adhering to safety protocols and operational guidelines.

Scope

This user guide covers the following key aspects of the Syncro 5 cobot:

- **Setup and Installation:** Step-by-step instructions for proper installation and setup of the cobot in your environment
- **Operating Instructions:** Detailed information on how to operate the cobot, including control modes, task programming, and safety protocols
- **Maintenance and Calibration:** Guidance on maintaining and calibrating the cobot to ensure longevity and precision
- **Troubleshooting:** A list of common issues and solutions to ensure smooth operation and reduce downtime
- **Safety Considerations:** Best practices and safety measures to ensure the safe use of the cobot in collaborative workspaces. This guide does not cover third-party integrations, advanced programming techniques, or modifications to the cobot that fall outside the manufacturer's recommendations

Audience

- **Operators:** Individuals responsible for the day-to-day operation of the cobot
- **Technicians:** Professionals handling installation, programming, and maintenance tasks
- **Supervisors and Managers:** Decision-makers overseeing cobot usage in their workflows
  Prior experience with robotics or automation systems is recommended but not mandatory

Support

For technical assistance or inquiries regarding the Syncro 5 cobot, users can contact Addverb's dedicated support team. **Support services include:**

- Installation and configuration assistance
- Guidance on programming and operational features
- Troubleshooting and maintenance support

Contact information:

- **Email:** support@addverb.com
- **Phone:** +91 9871590101
- **Website:** www.addverb.ai

Resources

This guide includes references to additional resources for users seeking more detailed information or advanced use cases:

- **Documentation:** Supplementary manuals, datasheets, and programming guides are available at www.addverb.com/documentation
- **Training:** Access online or in-person training programs through Addverb's training portal
- **Community:** Join the Addverb user community for tips, best practices, and peer support
- **Safety Standards:** Refer to ISO 12100:2010, Annex I of 2006/42/EC, EN 60204-1:2018, and ISO 10218-1:2011standards for collaborative robots. By utilizing these resources, users can ensure the optimal performance and longevity of their Syncro 5 cobot.

Chapter 1: Safety and Compliance

This section introduces the principles and norms that should be followed while operating the Cobot. Users must read the relevant description in this section carefully.

This section includes safety precautions for protecting the user and preventing any damage to the Cobot:

1.1 Validity and Responsibility

The Syncro 5 cobot safety section outlines operational guidelines but does not include comprehensive instructions on designing, installing, or integrating the robot application. The safety of the overall application, including its peripheral equipment, depends on adherence to the relevant standards and regulations of the deployment country.

Responsibility of Integrators:

Integrators must ensure compliance with applicable regulations and reduce risks associated with the Syncro 5 cobot. **Their responsibilities include:**

- Conducting a risk assessment for the entire robotic system
- Interfacing the Syncro 5 cobot with other machines and implementing additional safeguarding measures as needed
- Configuring safety settings correctly in the software
- Preventing unauthorized modifications to safety measures
- Validating the design, installation, and integration of the robotic application
- Providing clear operational instructions and displaying relevant safety signage at the installation site
- Maintaining documentation, including risk assessments, integration guidelines, and this manual

1.2 Limitation of Liability

The guidance provided in this manual should not be considered a guarantee that the Syncro 5 cobot will eliminate all risks of injury or damage. Even with all safety measures in place, unforeseen risks may occur.

Addverb disclaims any warranty for injuries, property damage, or other incidents resulting from improper use or failure to adhere to these safety instructions.

1.3 Terms and Conditions

Follow these instructions while using the cobot to ensure safe and efficient operation:

- Install the cobot and all electrical equipment exactly as described in this manual
- Press the Emergency Stop switch immediately to halt any unexpected or undesired cobot movements
- Avoid frequently powering the cobot ON and OFF, as the Syncro joint module contains internal brakes

- Maintain a minimum clearance of 1.5 meters around the cobot when operating at speeds greater than 0.5 m/sec to ensure safety

1.4 General Warning and Caution Messages

This section provides an overview of the safety icons and warning symbols used throughout this guide. These icons are designed to highlight critical information, ensuring safe and effective operation of the Syncro 5 cobot.
Icons are visual cues to draw attention to important safety instructions, warnings, and operational guidelines. Familiarize yourself with these icons to mitigate risks and prevent harm.

Icons
 ***WARNING*** *Indicates a hazardous situation that, if not avoided, can result in death or serious injury.*
 ***WARNING: ELECTRICITY*** *Indicates a hazardous situation that, if not avoided, can result in death or serious injury.*
 ***CAUTION*** *Indicates a hazardous situation that, if not avoided, can result in serious injury.*
 ***NOTICE*** *Indicates the risk of damage to equipment and/or information to be noted.*

1.5 General Safety Guidelines

To ensure safe operation of the Syncro 5 cobot, adhere to the following:

CAUTION: INSTALLATION AND SETUP

Before operating the Syncro 5 cobot:

- Ensure the cobot arm and end effector are securely installed
- Provide sufficient operational space to avoid collisions with objects or personnel
- Perform safety configuration during setup to match application requirements

CAUTION: OPERATIONAL SAFETY

Before operating the Syncro 5 cobot:

- Prohibit the use of a damaged cobot
- Ensure all personnel are protected during the cobot's transport, installation, programming, operation, and maintenance
- Inform operators of the cobot's emergency stop locations and procedures

CAUTION: USER SAFETY

Before operating the Syncro 5 cobot:

- Avoid wearing loose clothing, jewellery, or long hair near the cobot
- Maintain safe distances during cobot operation, especially during startup
- Regularly inspect the cobot for loose bolts or signs of wear and tear
- Ensure operators understand the direction of the cobot's movements when using the teach pendant
- Treat the cobot as active even when it appears stopped; it may be waiting for a signal
- Always keep the emergency stop (E-stop) switch easily accessible

1.6 Risk Assessment

A detailed risk assessment must be conducted prior to powering the Syncro 5 cobot for the first time and after any modifications. This includes:

- Evaluating the risks associated with teaching, troubleshooting, and normal operation
- Identifying and implementing safety settings, emergency stops, and safeguarding measures

Safety Configuration Settings:

- **Force and Power Limiting:** Reduces clamping forces and pressures during collisions
- **Speed Limitation:** Limits movement speed to reduce impact forces
- **Orientation Restrictions:** Prevents hazardous positioning of sharp edges or tools toward operators

1.7 Pre-Use Safety Checks

CAUTION: ELECTRICAL HAZARDS

Before operating the Syncro 5 cobot:

- Verify all safety inputs and outputs are correctly connected
- Test emergency-stop functionality and safeguard inputs
- Confirm operational modes change as indicated on the interface
- Test reduced-speed mode to ensure safe manual operation
- Connect 10A MCB in line with the Control box power
- Before powering on the control box make sure the earthing voltage is less than 2V

1.8 Warnings and Potential Hazards

WARNING

Potential Hazards Include:

- Injuries from sharp edges or points on tools, obstacles, or workpieces
- Bruising or fractures due to collisions
- Loose bolts causing unintended movement
- Incorrect safety configurations or unauthorized changes
- Items falling from or flying off the cobot due to poor grip or mechanical failure

1.9 Intended Use

The Syncro 5 cobot is designed for industrial use, such as handling tools, transferring products, or performing collaborative tasks. Ensure the application adheres to low-risk requirements identified in the risk assessment.

PROHIBITED USES

- Direct handling of hazardous materials
- Situations requiring compliance with food-grade or hygiene standards
- Any application outside the cobot's specified operational parameters

1.10 Maintenance Safety Precautions

To ensure the safety of operators and equipment during maintenance, follow these precautions:

- **Power Off Completely**: Always turn off the power and remove the main input cable from the control box to ensure the system is completely powered off before performing any maintenance tasks
- **Electrical Ground Check**: Verify the electrical grounding of the system before turning it back on to prevent electrical hazards
- **No Unauthorized Disassembly**: Avoid disassembling the power supplies or any internal components inside the control box to prevent electrical shocks or system damage
- **Prevent Environmental Hazards**: Ensure no water or dust enters the cobot or control box to maintain operational safety and reliability

These safety precautions must be strictly adhered to during maintenance to avoid injuries or system failures. For detailed Maintenance procedure refer to Maintenance section in this guide.

Chapter 2: Introduction to Syncro 5 cobot

This section introduces the Syncro 5 cobot detailing the necessary features:

2.1 Overview

The Syncro 5 cobot by Addverb revolutionizes collaborative robotics with its advanced design and user-focused capabilities. Featuring six degrees of freedom, this cobot performs complex tasks with exceptional precision and flexibility, making it ideal for a wide range of applications, from assembly to material handling. Equipped with six actuators, it offers a broad range of motion to adapt to diverse operational needs. The cobot integrates seamlessly with human operators, using an intuitive teach pendant and a dedicated control box for simple programming and control. With built-in safety features the Syncro 5 cobot ensures safe and efficient operation in shared workspaces, enhancing productivity while maintaining a strong focus on user safety. Following are some of common use cases of the Syncro 5 cobot:

- Picking and packing in warehouses
- Loading and unloading goods
- Inventory management and tracking
- Assembly tasks on production lines
- Quality control inspections
- Material handling in manufacturing
- Delivering medical supplies in healthcare
- Supporting physiotherapy exercises
- Shelf stocking in retail and warehouses
- Product testing and calibration
- Prototyping in R&D

Figure 1: Syncro cobot

2.2 Key Features of Syncro 5 cobot

Following are a few of the key features that Addverb's Syncro 5 cobot supports:

2.2.1 Six Degrees of Freedom

The Syncro 5 cobot operates with six degrees of freedom, enabling precise and versatile movement. It performs rotational motions along pitch, yaw, and roll axes while executing translational movements along the same axes. This design mimics the complex mobility of a human arm, making the cobot highly adaptable for intricate tasks. It's ability to manipulate objects in various orientations allows seamless operation in dynamic environments and enhances collaboration with human workers.

2.2.2 Advanced Design and Construction

The cobot's robust framework integrates six high-performance actuators, ensuring optimal control and dexterity. These actuators maximize the cobot's reach, allowing it to navigate tight spaces and handle detailed operations effortlessly. The lightweight yet durable construction maintains structural strength while ensuring agility. Its modular architecture supports easy upgrades and modifications, making the Syncro 5 cobot adaptable to evolving operational demands.

2.2.3 Intuitive Control System

The Syncro 5 cobot simplifies operation with a user-friendly control system, combining a teach pendant that the user can purchase if they require and a dedicated control box. The teach pendant provides an intuitive graphical user interface (GUI), enabling operators to program tasks and make real-time adjustments efficiently. The control box synchronizes actuator movements, ensuring precise and coordinated actions for complex tasks. Operators can refer to the Teach Pendant User Manual for detailed guidance on programming and customization.

2.2.4 Collaborative Safety Features

This robot is equipped with torque safety, meaning it automatically stops whenever it detects a force opposing its motion. This ensures that if the robot comes into contact with a human or any unexpected obstacle, it halts immediately to prevent injury or damage.

2.2.5 Enhanced Adaptability and Longevity

The Syncro 5 cobot's modular and flexible design ensures long-term adaptability to diverse applications. Its scalable architecture supports integration into various industrial settings, ensuring consistent productivity and operational efficiency.

2.3 What's in the Box

When you place an order, you receive the following items in a box:

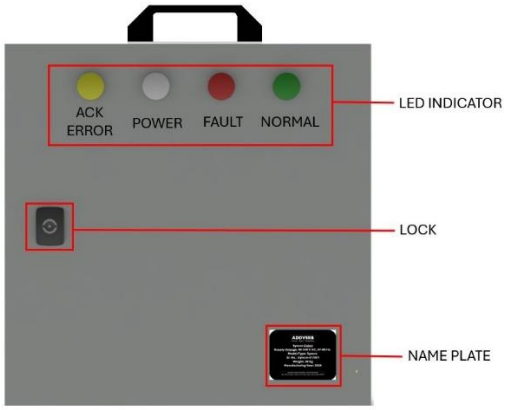
Item List	Quantity
Syncro 5 cobot	1
Control Box	1
Cobot Cable	1
AC Power Cable	1
Emergency Stop kit	1
Teach Pendant (Optional)	1

Chapter 3: Hardware Requirements

This section details the hardware specifications and requirements:

3.1 Physical Component Overview

Refer to this table for the physical component list and its description:

Component	Description
	**Cobot:** The collaborative robot (Cobot) is designed to work safely alongside human operators in various industrial environments. It is a flexible, programmable robotic arm that can perform tasks such as material handling, assembly, or welding.
	**Control Box:** The control box is the central hub for managing the cobot's operations and safety features. It houses the necessary electronics and power systems to control the robot's movements, monitor system status, and communicate with external devices. The control box typically includes input/output (I/O) interfaces for actuators and other connected components.
	**Emergency Stop (E-stop)**: The Emergency Stop (E-stop) is a critical safety feature that instantly halts all robot operations in case of an emergency or unsafe situation. Attached to the control box, the E-stop is a red, mushroom-shaped push-button that, when pressed, immediately disables the Cobot's motion to prevent injury or damage.

3.2 Hardware Specifications

Parameter	Specification
Cobot Model	Syncro
Payload Capacity	Up to 5 kg
Reach	650 mm
Number of Axes	6 Degrees of Freedom
Weight	Up to 30 Kgs
Material	Aerospace-grade aluminum or equivalent durable material
Mounting Options	Floor
End-Effector Compatibility	User-defined
Control Box Dimension	(400L X 400B X 210H) mm

3.3 Environmental Condition

Parameter	Specification
Operating Temperature	0°C to 50°C
Storage Temperature	-10°C to 50°C ***Note:*** *Start when the equipment is at normal temperature*
Humidity	0% - 85% RH (Non-condensing)

NOTE

- Emergency stops are complementary protective measures, not safeguards (ISO 12100). They do not fully prevent injury.
- Perform a risk assessment to determine whether additional emergency stop pushbuttons are required.
- Ensure the emergency stop and actuating device comply with ISO 13850.

3.4 Emergency Stop

The emergency stop (E-stop) is a red push-button located on the Teach Pendant and the control box (see *Figure 2: Emergency stop*). Use the E-stop to immediately stop all robot motion during emergencies or dangerous situations. Pressing the E-stop activates a stop category one, as defined in IEC 60204-1.

Figure 2: Emergency stop connection

3.4.1 Using the Emergency Stop

- Press the E-stop button to halt all robot operations immediately
- Once activated, the E-stop latches into the pressed position and requires manual resetting

3.4.2 Resetting the Emergency Stop

- Before resetting, visually inspect the entire system to identify and resolve the cause of the E-stop activation
- Hold the E-stop button and twist it anticlockwise until the latch disengages. You will feel the button release when it is reset
- Confirm that all issues have been resolved before proceeding
- Once the E-stop is reset, the power is restored and the cobot resumes operation

3.4.3 Recovering from Emergency Conditions

- The E-stop button is equipped with a locking function that engages during activation
- Rotate the E-stop button to release the lock and exit the emergency stop state

3.5 Control Box Details

The control box serves as the central hub for operating the Syncro 5 cobot, providing a safe and efficient control system. It houses critical components, including a primary computer, a secondary computer, two switching power supplies, and a safety PCB to ensure user and cobot safety.

3.5.1 Precautions and Safety Measures

To ensure safety during installation and operation of the cobot and control box, the following precautions must be adhered to:

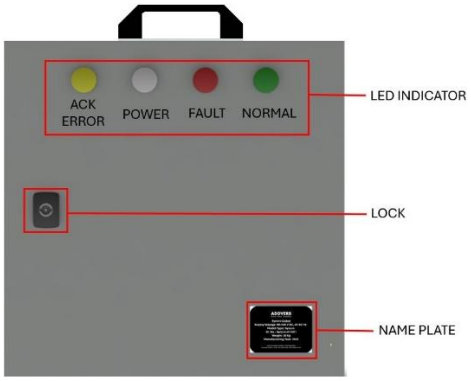


Figure 3: Control box (Front view)

CAUTION: ELECTRICAL HAZARDS

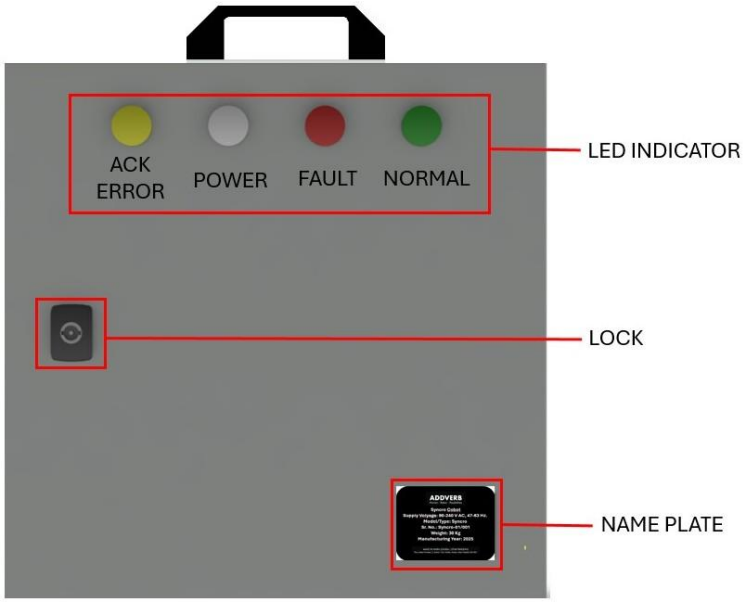
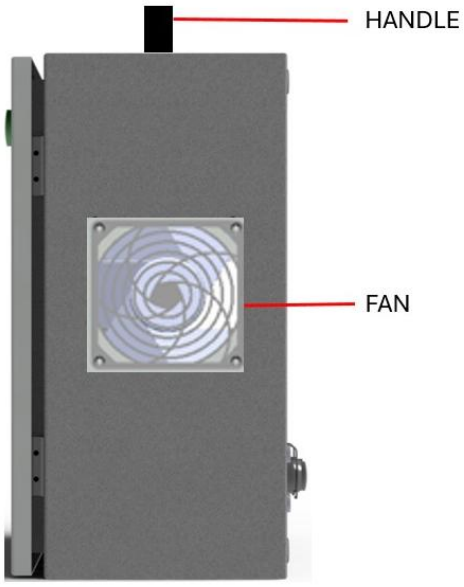
Before operating the Syncro 5 cobot:

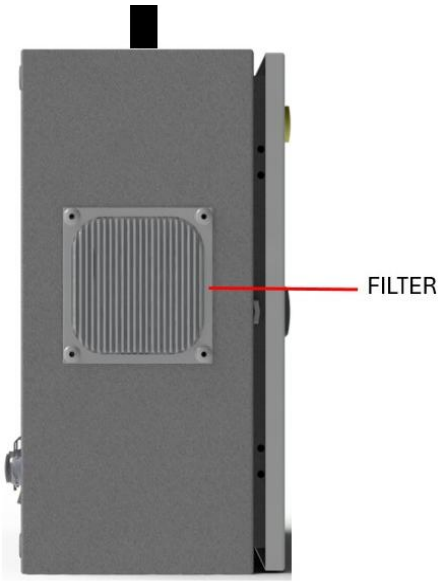
- Always disconnect the control box from the power source before performing maintenance, adjustments, or connections
- Avoid Overheating: Ensure adequate ventilation around the control box. Do not block ventilation slots or place the control box near heat sources
- No Water Exposure: Keep the control box dry at all times. If water enters the box, immediately shut off the power and contact Addverb's support team for assistance
- Authorized Access Only: Only trained personnel should open the control box for troubleshooting or repair. Tampering by unauthorized individuals can result in electrical hazards or void the warranty

CAUTION: CONTROL BOX SAFETY

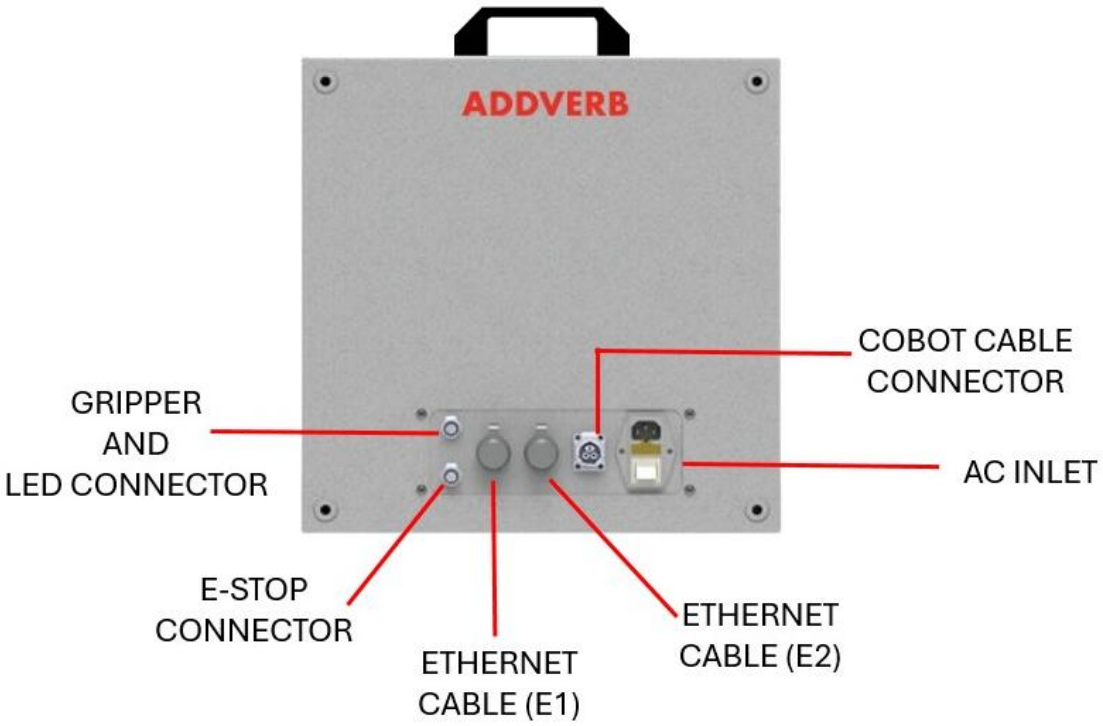
- Secure the control box in a stable and vibration-free location to prevent accidental movement or tipping
- Use only the original cables supplied by Addverb and ensure that all connections to and from the control box are securely fastened and free from tension
- Operate the control box within the specified temperature and humidity ranges outlined in the manual to avoid operational failures
- Connect the control box to a surge protector or UPS to prevent damage from voltage spikes or power fluctuations
- Regularly inspect the control box for signs of wear, loose connections, or damage, and address any issues promptly

The control box is equipped with the following panels for efficient interaction and connectivity:

Electrical Component	Description
	**Front Panel** Houses LED Indicators and lock to open the Control box
	**Right-Side Panel:** Provides fans for cooling the control box



Left Side Panel:
Provides filters in the control box



Back side of the control box:
Features electrical interfaces and power connections

3.5.2 Control Box LED Indicators

The control box features **four LEDs** that provide real-time visual feedback about the cobot's operational status.

LED Color	Indication	Description
White	Power Status	Indicates the control box is powered
Red	E-stop/Error	Indicates that the emergency stop (E-stop) is pressed, or the system is in error
Green	Running Status	Indicates the cobot is running, and the system status is healthy
Yellow	Error Acknowledge	Indicates an error that needs to be acknowledged using the Teach Pendant or a connected computer

Chapter 4: Electrical Requirements

This section details all the required electrical specifications for the Syncro 5 cobot system:

4.1 Electrical Warning and Caution

WARNING: ELECTRICAL HAZARDS

Failure to follow these instructions may result in serious injury, death, or equipment damage due to electrical hazards:

- Ensure that all equipment not rated for water exposure remains dry. If water enters the product, immediately lock out and tag out all power sources before contacting Addverb's support team
- Only use the original cables supplied with the Syncro 5 cobot. Substituting cables or using the cobot in applications where cables are frequently flexed can result in damage or failure
- When installing interface cables to the cobot's I/O ports, remove the bottom metal plate for proper cable management. Use appropriate tools and gland sizes. Ensure that all metal shavings or debris from drilling are thoroughly removed before reattaching the plate

4.2 Electrical Cables and Connector List

Following lists the electrical component cables and connectors that is used for connecting to the cobot:

Cable Types:

- **Power Cable:** Connects the Control Box to the mains power supply
- **Ethernet Cable (E1 and E2):** Provides communication between devices and the Control Box
- **Robot Cable:** Establishes the connection between the robot arm and the Control Box
- **E-Stop Cable:** Connects Emergency Stop Buttons for safety

Electrical Component	Description
	**Cobot cable connector:** This connector serves as the interface between the robot arm and the control box, facilitating the transfer of data and power required for the robot's operation. It ensures secure and reliable connectivity to maintain seamless performance.
	**Power cable:** The power cable delivers electrical energy to the control box and Cobot, ensuring uninterrupted operation. Its robust insulation and secure connections are designed for industrial environments, providing safety and durability.
	**Ethernet cable:** This cable enables high-speed communication between the Cobot and external devices, such as PCs or networked systems. It is essential for data transmission, remote monitoring, and control functions. There are two ethernet cables, E1 and E2. E1 (Interfacing Device Connection): Connects the interfacing device such as your Personal computer or controller to the cobot. E2 (Control Box to Cobot Connection): Connects the Control Box to the Syncro 5 cobot for data communication.
	**E-stop Cable Connector (4 Pin)**: The E-stop cable connector integrates the emergency stop functionality into the system. It provides a secure connection to ensure rapid response in emergency scenarios, halting all operations when the E-stop button is activated.

4.3 Electrical Specifications

Following are the electrical specifications' requirements for operating the Syncro 5 cobot:

Parameter	Specification
Input Voltage	90-240 VAC
Input Current	16 A/115 VAC, 10A/230 VAC
Frequency	47-63 Hz

4.4 Electrical Connection Illustration

Refer to the following illustration to understand the connection of the cobot and its various parts using the cable and connectors mentioned in section Electrical Cables and Connector List.

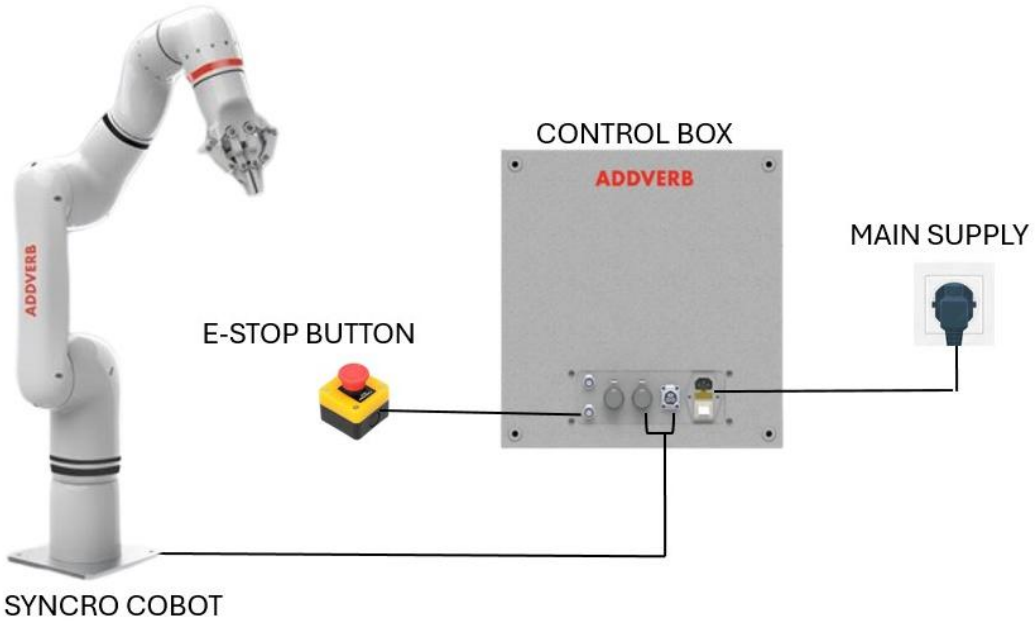


Figure 4: Electrical Connection Illustration

Chapter 5: Installation of Syncro 5 cobot

This section details the assembly instructions and requirements required to install the Syncro 5 cobot:

5.1 Warning and Caution Messages

WARNING: INSTALLATION SAFETY

- Ensure that the main power supply is completely disconnected before starting the installation. Failure to do so could result in serious injury or death
- Use only approved voltage and frequency ratings as specified in the user guide. Incorrect power configurations may lead to electrical fires or equipment damage
- Always use appropriate lifting tools when handling the cobot during installation. Attempting to lift or move the cobot manually could result in serious injury
- Do not install the cobot in areas prone to water exposure unless it has been rated for such environments. Electrical components can malfunction or cause injury if exposed to water

CAUTION: INSTALLATION SAFETY

- Route all cables properly to prevent tripping hazards or damage to connections. Avoid placing cables near sharp edges or areas with frequent movement
- Ensure the cobot is mounted on a flat and stable surface. An improperly secured cobot may result in operational errors or damage
- When securing bolts and connections, do not overtighten, as this may damage components or impair alignment
- Install the cobot in a temperature-controlled environment, as extreme temperatures may impact performance and longevity
- Ensure the installation area is free of excessive dust, debris, or contaminants that could interfere with the cobot's operation or compromise its components

5.2 Workspace and Operating Space

The **workspace** refers to the total area within which the Syncro 5 cobot can physically operate or perform tasks. This includes the reach of the robotic arm, its range of motion, and the defined boundaries that the cobot should not exceed to ensure safe and efficient operations. The **operating space** is a subset of the workspace, focusing on the area actively used for specific tasks. This is where the cobot interacts with objects, tools, or humans during its operations. The operating space is often customized based on the task requirements and safety protocols.

5.2.1 Mechanical Dimension of Syncro 5 cobot

- The workspace of the cobot is a sphere with a radius of **650 mm (2.132 ft)**
- Ensure no obstructions within this range during installation to avoid interference with the cobot's operations

Figure 5: Syncro Cobot Workspace

5.2.2 Syncro 5 cobot Mounting Specification

Parts	Description
Syncro 5 cobot	Mounted with four M8 x 75 mm bolts. For improved installation accuracy, it is advisable to use four Φ9 mm holes for the bolts on the base.
Mounting Tool	Use four M6 thread holes for attaching a tool to the robot. The M6 bolts shall be tightened with 8 Nm, strength class 8.8. For accurate tool repositioning, use a pin in the Ø6 hole provided.
Control Box	The Control Box can be hung on a wall or placed on the ground. You can buy extra brackets for mounting the Control Box.
Teach Pendant (Optional)	The Teach Pendant handheld and is an optional component.

5.3 Securing the Syncro 5 cobot

STEP 1: Place the robot arm on the surface on which it is to be mounted. The surface must be even and clean.

Figure 6: Place Syncro Cobot on the Surface

STEP 2: Attach the Syncro 5 cobot to the base using four M8 x 75mm bolts.

Figure 7: Attach the Syncro Cobot

STEP 3: For accurate installation, make four 9 mm holes on the base, spaced 160 mm apart from each other as shown in Figure 8: Mounting Specification

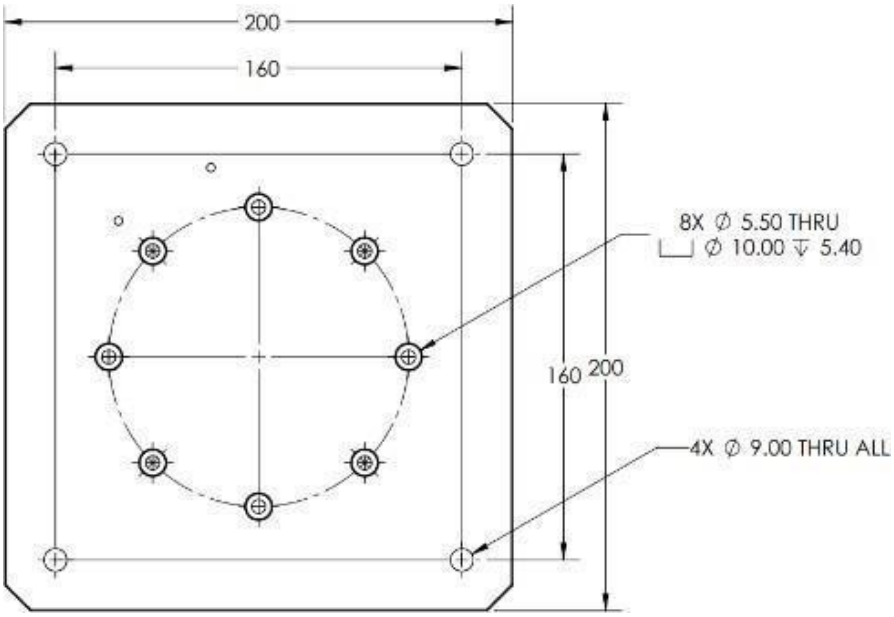


Figure 8: Mounting Specifications

STEP 4: Tighten the four M8 bolts to a torque of 20 Nm.

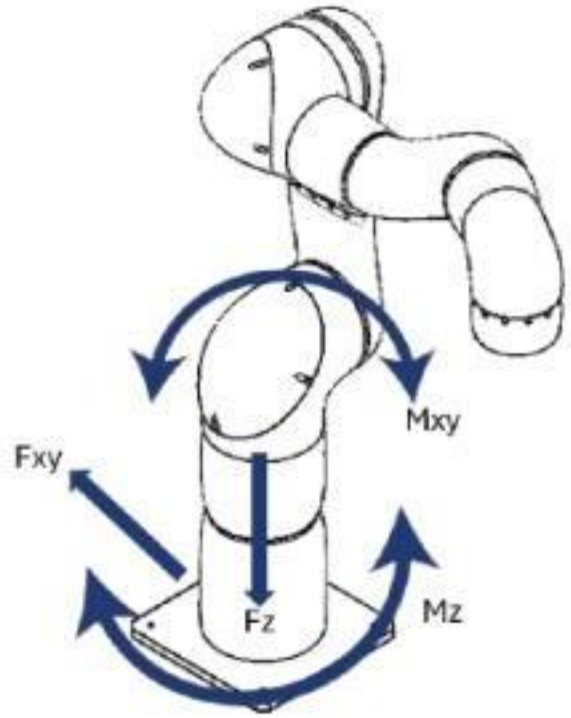
5.3.1 Load and Torque Criteria

Use the following section for adhering to the following torque requirements and accounting for both normal and stopping motion loads to ensure a safe and efficient cobot installation. This section explains the direction of the forces and the amount of the force exerted in those directions when the cobot is moving

Importance of the robot stand

The structure (stand) on which the cobot arm is mounted is a critical component of the installation. It must be robust and free from vibrations caused by external sources to ensure smooth and safe operation. During operation, each robot joint generates torque, which enables movement and stopping of the robot arm. This joint torques transfer loads to the robot stand in the following ways:

- Mz: Torque around the base z-axis
- Fz: Forces along the base z-axis
- Mxy: Tilting torque in any direction of the base xy plane
- Fxy: Force in any direction within the base xy plane



Robot Model	Mz [Nm]	Fz[Nm]	Mxy[Nm]	Fxy [Nm]
Syncro	170	300	162	80

Figure 9: Joint torques loads transferred to the robot stand.

5.4 Placing Control Box

Place the control box on a stable surface, ensuring it meets the required clearance guidelines. Inadequate airflow around the control box can cause overheating and lead to equipment malfunction.

To maintain proper cooling:

- Ensure a minimum clearance of 50 mm on each side of the control box.
- For optimal performance, a clearance of 200 mm is recommended.

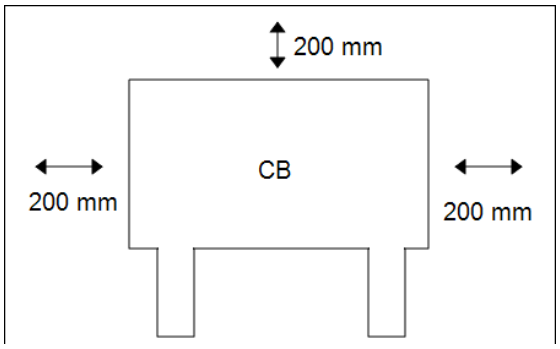


Figure 10: Recommended clearance of space around the control box

5.5 Connecting the Syncro 5 cobot to Control Box

To ensure the proper functioning of the Syncro 5 cobot, all electrical connections must be completed according to the guidelines provided. The connection process involves two primary steps: making the connection to the Control Box and establishing the connection to the Syncro 5 cobot.

5.5.1 Electrical Safety Requirements

WARNING: ELECTRICAL HAZARDS

Failure to follow these instructions may result in serious injury, death, or equipment damage due to electrical hazards:

- Ensure the robot is grounded correctly (electrical connection to ground). Use the unused bolts associated with grounding symbols inside the Control Box to create common grounding of all equipment in the system. The grounding conductor shall have at least the current rating of the highest current in the system
- Ensure the input power to the Control Box is protected with a Residual Current Device (RCD) and a correct fuse.
- Lockout all power for the complete robot installation during service.
- Ensure other equipment shall not supply power to the robot I/O when the robot is locked out.
- Ensure all cables are connected correctly before the Control Box is powered. Always use the original power cord.

5.5.2 Electrical Connection Illustration

Refer to the following illustration to understand the connection of the cobot and its various parts using the cable and connectors mentioned in section Electrical Cables and Connector List.

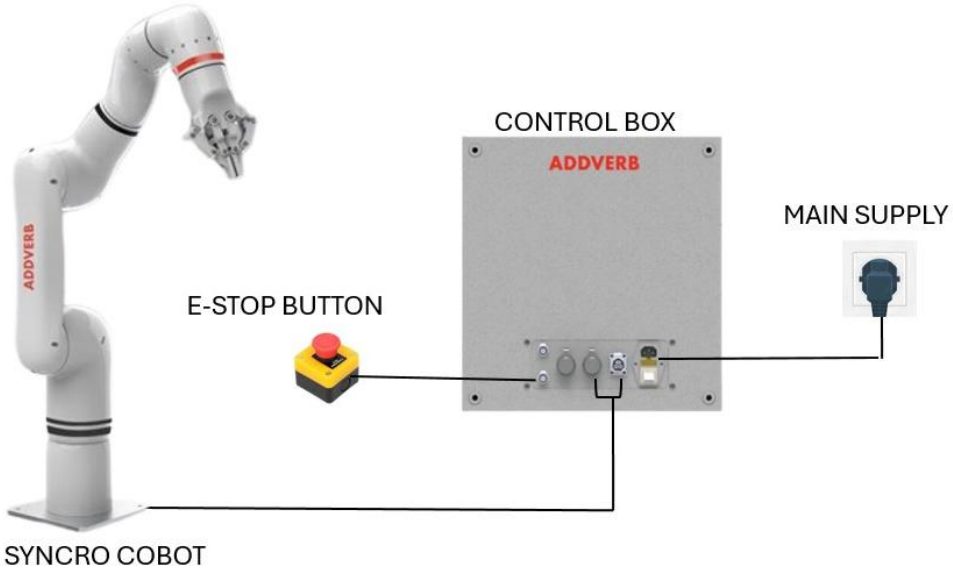
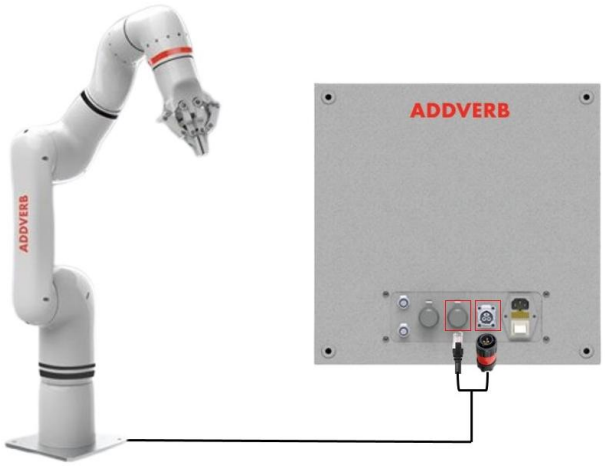
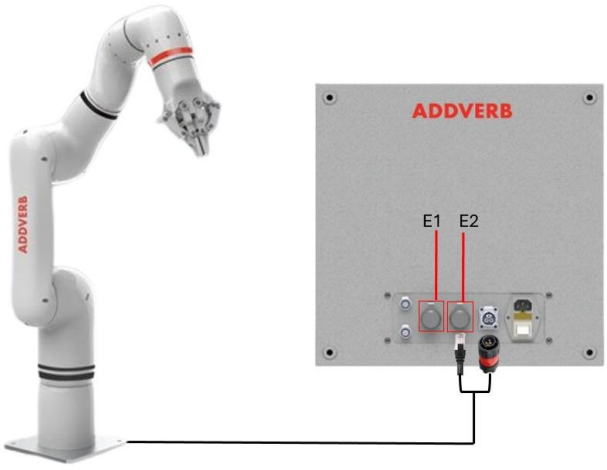


Figure 11: Electrical Connection Illustration

5.5.3 Making the Connection

Steps	Images
Power Connection: Use the provided power cable to connect the Control Box to the mains supply. Install a 10A Miniature Circuit Breaker (MCB) in the power line for additional safety. Ensure the earthing voltage is less than 2V before powering on the Control Box to avoid electrical issues.	
Robot Arm Connection: Establish the connection between the robot arm (Syncro 5 cobot) and the Control Box using the cobot power cable: One end of the robot cable connects to the cobot cable connector on the robot arm. The other end connects to the corresponding port at the bottom of the Control Box. Ensure the connection is secure by twisting the connector twice to lock it in place.	
Ethernet Connections: The Control Box uses two Ethernet ports: **E1 (Interfacing Device Connection):** Connects the interfacing device such as your Personal computer or controller to the cobot. **E2 (Control Box to Cobot Connection):** Connects the Control Box to the Syncro 5 cobot for data communication. Ensure cables are securely inserted into the correct ports. Interchanging Ethernet cables can cause communication failures.	

Emergency Stop (E-Stop) Connections: Connect the Emergency Stop Button to the E-Stop port on the Control Box. For applications requiring additional safety, connect an additional Emergency Stop Button to the ES2 Port.	

Chapter 6: Getting Started

Use this section to get started with the Syncro 5 cobot once the installation of the cobot is completed:

6.1 Powering On/Off the Syncro 5 cobot

6.1.1 Preparation before Power Up

- Check the connection of the cobot and the control box
- Check the connection of the power cable and the control box
- Check and confirm that the power switch of the control box is off when the cobot is unpowered
- Check whether the emergency stop button is released

6.1.2 Powering on the Cobot System

- Connect the Estop connector
- Switch **ON** the AC mains

6.2 Connecting Mounting Tool

WARNING: MOUNTING TOOL

- Always disconnect the power supply to the cobot before installing the mounting tool. Failure to do so may result in unexpected cobot movement, leading to serious injury
- Improperly attached tools may detach during operation, posing a risk of injury or equipment damage. Ensure all connections are tight and secure
- Exceeding the cobot's payload capacity when attaching tools may cause joint failure, leading to operational hazards and potential harm to personnel

NOTE

- Failing to update parameters after mounting an external tool will result in a safety system failure

To mount the tool on the Syncro 5 cobot, follow these steps:

- Attach the tool to the tool mounting flange using M6 bolts, as illustrated in *Figure 12: Mounting Tool Illustration*.
- If an external tool is mounted on the cobot's end, update its respective mass and inertia parameters in the software.

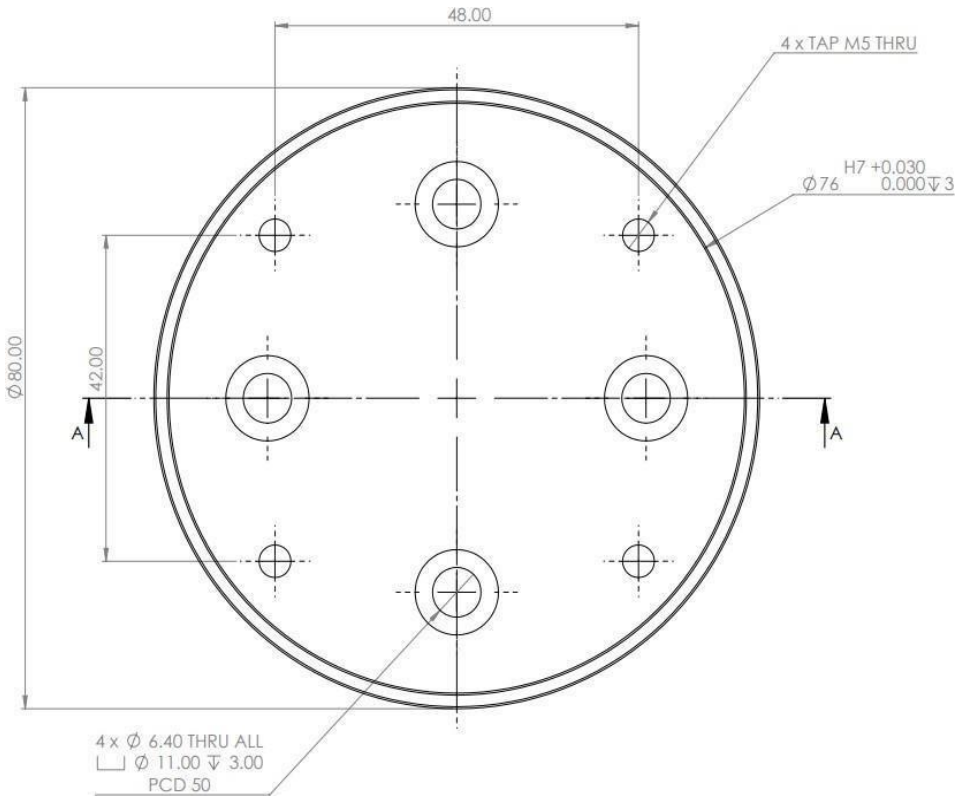


Figure 12: Mounting Tool Illustration

Chapter 7: Recommended Storage Conditions

Refer to this section for the recommended storage condition of the Syncro 5 cobot:

NOTE

- Do not store a robot for more than six months without performing a visual inspection.

7.1 Temperature Limits

Proper temperature management is crucial to ensure the cobot's longevity and operational efficiency.

- **Room Temperature**:
  Store the cobot and control box at normal room temperature to avoid component degradation
- **Avoid Extreme Conditions**:
  Do not expose the cobot to extreme heat or cold, as this can cause material damage and malfunctions
- **Moisture Protection**:
  Prevent condensation, excess moisture, and water exposure to safeguard against corrosion and electrical malfunctions

7.2 Storage Best Practices

- **Clean Environment**:
  Ensure the cobot is stored in a dust-free, clean area, free from dirt, harmful chemicals, or contaminants

- **Protective Covering**:
  Use cases or covers to shield the cobot from dust and particulate matter when idle

- **Stable Surface**:
  Place the cobot on a stable, vibration-free surface to avoid physical damage during storage

- **Secure Positioning**:
  Position the cobot securely to prevent accidental falls, shifts, or mishandling

- **Disconnect Power**:
  Always disconnect the power supply during storage to prevent unnecessary discharge or electrical damage

- **Accessibility**:
  Store the cobot in a location that allows for easy access for visual inspections and periodic maintenance

- **Manuals and Documentation**:
  Keep technical manuals, maintenance logs, and other relevant documentation nearby for quick reference and compliance

- **Routine Inspections**:
  Conduct periodic inspections to monitor the cobot's condition, checking for any signs of wear, damage, or environmental impact

Chapter 8: Packing and Transportation

8.1 Precaution and Safety Measures

CAUTION: PACKING AND TRANSPORTAION

- Incorrect lifting techniques, or using improper lifting equipment, can lead to injury.
- Avoid overloading your back or other body parts when lifting the equipment.
- Use proper lifting equipment.
- All regional and national lifting guidelines shall be followed.
- Make sure to mount the robot according to the instructions in Mechanical Interface.

8.2 Transportation Details

Following materials are used for packing:

- Wooden box: use original box provided by Addverb
- Foam
- Bubble wrap
- Tape
- Top EPE Fitment
- Bottom EPE and EVA Fitment
- 3D VCI Bag with Flap

8.2.1 Transportation Box Specifications

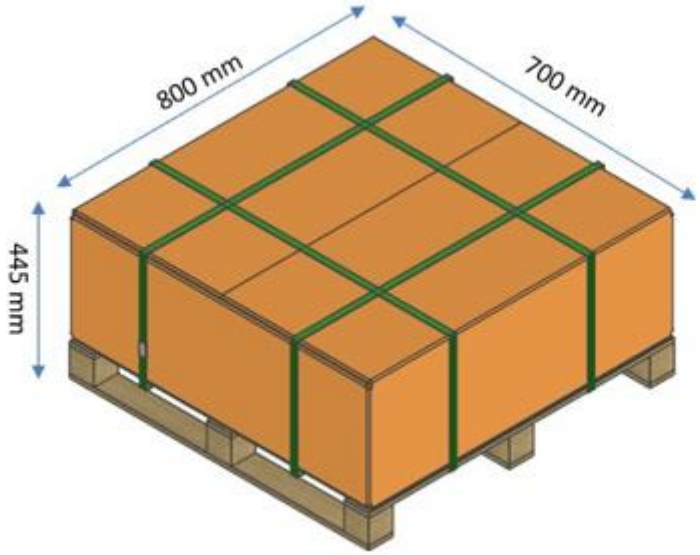


Figure 13: Packing box dimensions

Box Details	
Box OD	(800 X 700 X 445) mm
Gross Weight	45 Kg (Approx.)

8.2.2 Preparation for Transport

- Place the **Bottom EPE** and **EVA Fitment** in the cardbox
- Place the Syncro 5 cobot and control box and secure it with **3D VCI Bag with Flap**
- Ensure the cobot is securely packed within the cardboard box with appropriate cushioning materials to prevent movement and protect against impacts
- Place the **Top EPE Fitment** over the robot parts
- Seal the box with strong packing tape, ensuring all openings are secured
- Refer to *Figure 14: Packing* to understand the packing details

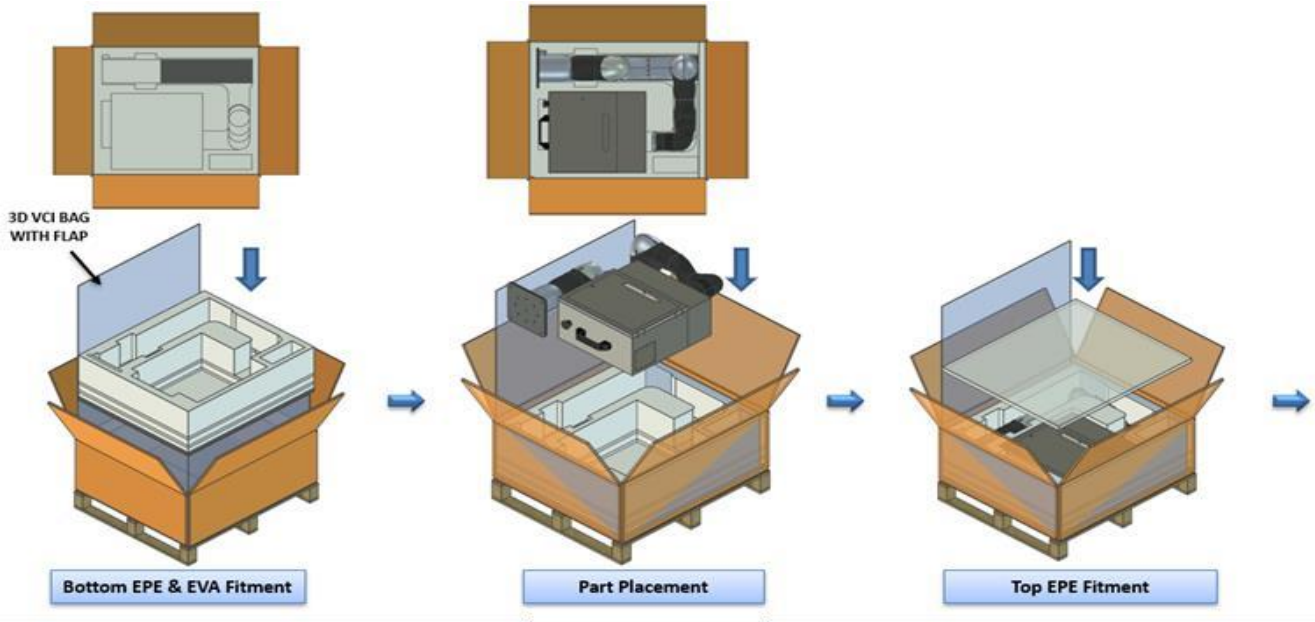


Figure 14: Packing Illustration

8.2.3 Handling Instructions

- **Lift with Care:**
  Due to the weight, two people should lift the box to prevent strain and ensure safety.

- **Avoid Dropping:**
  Handle the package gently to prevent damage to the cobot and its components.

8.2.4 Transport Mode

- **Recommendation:**
  Use a pallet jack or forklift for moving the package, if possible, to minimize physical handling.

- **Transport Vehicle:**
  Ensure the vehicle has a flat loading area and can support the weight of the package without risk of damage.

8.2.5 Loading and Unloading

- Place the box on a stable surface within the vehicle to avoid tipping or sliding during transit.
- Secure the box using straps or ropes to prevent movement during transport.

8.2.6 Environmental Considerations

- Protect the package from extreme temperatures, humidity, and moisture during transport.
- Avoid exposure to direct sunlight for extended periods.

Chapter 9: Robot Cleaning and Inspection

Regular cleaning and inspection are essential to ensure the optimal performance, longevity, and safety of the cobot and its accessories. This section provides guidelines for maintaining a clean cobot environment, including cleaning procedures for the Cobot, Teach Pendant, and Control Box. Proper cleaning practices help prevent issues caused by dirt, dust, or harmful substances, ensuring the robot functions efficiently and safely over time.

9.1 Inspection Checklist

This section details a checklist of inspections recommended by Addverb to be performed at the specified intervals. If any of the referenced parts are found to be in an unacceptable condition during inspection, please address the issues immediately to the Addverb Customer Support. Refer to Figure 15: Inspection Points and the Table 1: Inspection Checklist for the details on the inspection parts.

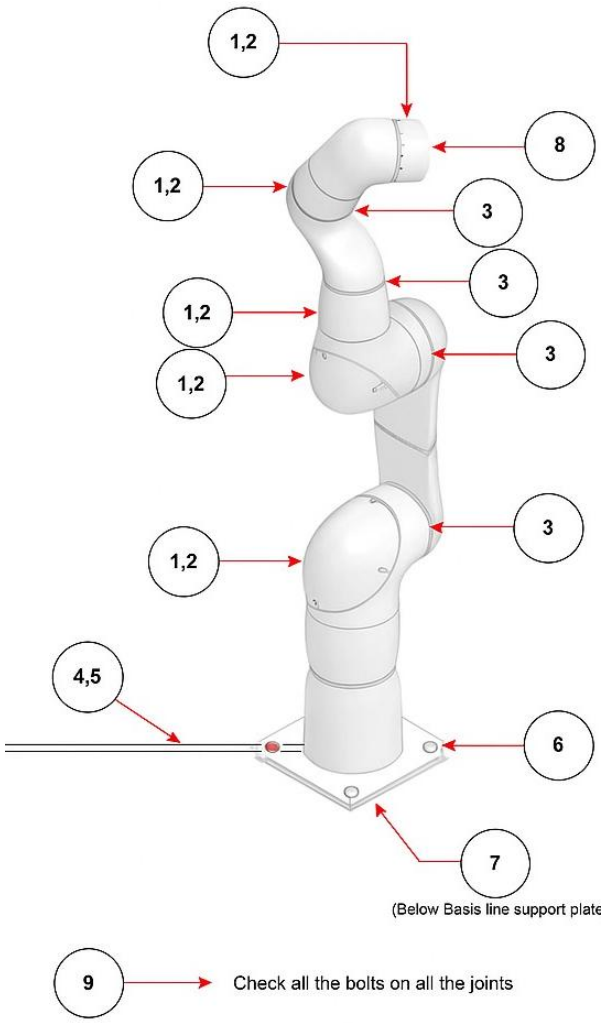


Figure 15: Inspection Points

Table 1: Inspection Checklist

S.No	Items to Check	Type	Monthly	Semi-Annually	Annually
1	Check link covers*	Visual		X	
2	Check the link cover screws	Functional		X	
3	Check the rubber ring	Visual	X		
4	Check Cobot cable	Visual		X	
5	Check Cobot cable connection	Visual		X	
6	Check Cobot mounting bolts*	Functional	X		
7	Check Base-link support bolts*	Functional	X		
8	Check tool mounting bolts*	Functional	X		
9	Check screws/bolts on joints*	Functional		X	

(* Must also be checked after heavy collision)

9.2 Visual Inspections

Perform the following steps during the visual inspection of the cobot to ensure all components are in proper working condition:

- Move the cobot to the zero position, if possible
- Turn off and disconnect the power cable from the control box
- Inspect the cable between the control box and cobot for any damage
- Check that the base mounting bolts are properly tightened
- Verify the tool flange bolts are securely fastened
- Inspect the rubber rings for wear or damage and replace them if necessary
- Examine the link covers for cracks or damage and replace them if required
- Ensure the screws used for the link covers are present and properly tightened

NOTE

- Rubber rings are external parts exposed to the operating environment. Addverb recommends replacing all rubber rings when one or more are found worn or damaged
- For new or spare parts, contact Addverb customer support
- Do not store a robot for more than six months without performing a visual inspection
- If damage is observed on the robot within the warranty period, contact Addverb customer support. Refer to the last page of the user manual for warranty information

9.3 Control Box

Perform the following inspections at the specified intervals:

S.no	Items to Check	Type	Monthly	Semi-Annually	Annually
1	Before powering on the control box for the first time, make sure the earthing voltage is less than 2V.	Functional			X
2	Check safety input and outputs (if connected)	Functional	X		
3	Check the cooling fan and clean the filter	Visual	X		
4	Check terminals in control box	Functional		X	
5	Check main power to control box	Functional			X

Emergency Stop Test:

- Press the Emergency Stop button
- Confirm the robot stops and shuts off power to the joints
- Power on the robot again to resume operations

9.4 Robot Cleaning

Proper cleaning practices help prevent issues caused by dirt, dust, or harmful substances, ensuring the robot functions efficiently and safely over time.

9.4.1 Cobot Cleaning

To remove dust, dirt, or oil from the cobot arm:

- Use a cleaning cloth along with water, isopropyl alcohol, 10% ethanol, or 10% naphtha.
- For robots operating in harsh environments, clean regularly and replace flat rings as needed.
- Do not use bleach or any diluted cleaning solution containing bleach.

9.4.2 Control Box and Teach Pendant Cleaning

- **Teach Pendant:**

Use a standard, mild industrial cleaning agent without thinning agents or abrasive materials to clean the screen.

- **Control Box:**

Remove the outer plastic frame and gently clean the filters on both sides with low-pressure air or replace them if dirty or damaged. Do not use compressed air inside the Control Box as it can damage components.

Chapter 10: Discard and Disposal Instructions

Proper disposal of the cobot and its components is critical to ensuring environmental responsibility, safety, and compliance with local and international waste management regulations. This section provides comprehensive guidelines for safely discarding the cobot, its control box, and associated materials, including electronic components, mechanical parts, and auxiliary agents. Adherence to these instructions will help minimize the environmental impact while maintaining compliance with applicable standards.

By following these discard and disposal instructions, you contribute to sustainable practices that protect the environment and comply with global standards. If you have questions or require additional assistance, contact **Addverb customer support** or consult with a certified waste management provider.

10.1 End of Service Life Considerations

When the cobot has reached the end of its operational life, follow these steps for responsible decommissioning:

- **Safe Decommissioning**:
  Have a trained technician handle the shutdown and disassembly process. This includes:
  - Powering off all systems safely
  - Disconnecting internal connections between components
  - Dismantling cobot parts as detailed in earlier sections of this guide

- **Proper Documentation:**
  Maintain records of the decommissioning process for compliance and traceability

10.2 Disposal of Electronic Components

Electronic components, such as printed circuit boards (PCBs), wires, and actuators, require careful handling due to potential environmental hazards. Follow these guidelines:

- **Classify as E-Waste:**
  Never dispose of electronic parts with general waste. These components should be treated as electronic waste.

- **Contact Certified Facilities:**
  Work with certified e-waste recycling centers to ensure proper disposal.

- **Organizational Policies:**
  Adhere to your organization’s specific guidelines for handling and disposing of electronic waste.

10.3 Mechanical Parts and Frame

Non-electronic components, such as the cobot's frame, joints, and other mechanical elements, should be evaluated for recycling or safe disposal:

- **Recyclable Materials**:
  - **Metals**: Most metallic parts, including the frame and joints, can be recycled.
  - **Plastics**: Non-contaminated plastics may be processed through municipal waste facilities.

- **Contaminated or Non-Recyclable Materials**:
  - Dispose of contaminated materials at authorized hazardous waste facilities following local guidelines.

10.4 Compliance with Local and International Standards

Adhering to disposal regulations ensures responsible waste management. Key standards include:

- **WEEE Directive (Europe):** Governs the disposal of electrical and electronic equipment.
- **EPA Regulations (USA):** Enforces safe handling and disposal of hazardous waste.
- **Other Local Guidelines:** Familiarize yourself with country-specific or regional rules regarding e-waste and hazardous materials.

10.5 Material Group Disposal Methods

The following table outlines recommended disposal methods for different material groups associated with the cobot:

Material Group	Disposal Method
Contaminated materials/auxiliary agents	Hazardous waste
Wood	Municipal waste
Plastic	Municipal waste
Lubrication	Disposal in accordance with the safety data sheets
Metal	Scrap metal collection
Electrical material	E-waste
Electronic waste	E-waste

10.6 Environmental Condition

Taking an eco-friendly approach during disposal is imperative:

- **Reduce Waste:** Evaluate components for repair or reuse before disposal.
- **Recycle:** Separate recyclable materials like metals and plastics.
- **Minimize Harm:** Avoid dumping hazardous waste in general landfill sites to prevent environmental contamination.

Chapter 11: Common Issues and Resolutions

Below is a comprehensive guide to address potential issues that may arise during the operation of the cobot and their corresponding resolutions:
This table can help operators and maintenance teams quickly identify and resolve common issues to ensure smooth cobot operation and maximize productivity. For further assistance, please contact **Addverb Customer Support**.

Issue	Description	Resolution
Safety Concerns	Cobots are designed to work alongside humans, but improper setup or lack of proper safety measures can lead to injuries.	Follow all safety protocols Set speed limits Regularly inspect the cobot's safety features
Programming Errors	Incorrect programming or configuration by the user can result in inaccurate task performance.	Double-check programming instructions Use simulation tools, if available, to test the code before executing it on the cobot
Limited Payload Capacity	Cobots often have lower payload capacities than industrial robots, limiting the tasks they can perform.	Assign tasks within the cobot's specified weight limit Use additional support or different robots for heavier tasks
Communication Failures	Disruptions when interacting with other devices, machines, or systems can affect cobot performance.	Consult with Addverb's support team for proper network configuration
Integration Issues	Integrating cobots with existing systems can be complex.	Use appropriate middleware or integration softwar Seek support from Addverb's integrators for complex systems
Limited Workspace	Cobots may have restricted working radii, making it difficult to reach all parts of the workspace.	Deploy multiple cobots Redesign the workspace for optimal reach
Software Incompatibility	Third-party software might not be compatible with the cobot's control system.	Verify software requirements with Addverb's support team before installation

Overheating	Prolonged operation can lead to overheating of the cobot's actuators.	Provide adequate ventilation or cooling systems Monitor operational temperature
Unstable Power Supply	Fluctuating power supply can cause malfunctions or damage.	Use a UPS or surge protectors Regularly check power cables and connections for wear and tear
Limited Speed and Agility	Cobots are slower and less agile than industrial robots, potentially reducing productivity.	Adjust speed settings to match task requirements Use faster industrial robots for tasks demanding high speed and agility
Human-Robot Interaction Challenges	Unexpected cobot behavior during human interactions can cause confusion.	Train operators on safe collaboration Implement intuitive interfaces and clear communication protocols
Lack of Flexibility for Complex Tasks	Cobots may struggle with tasks requiring high precision or decision-making.	Simplify complex tasks into manageable steps for the cobot

Chapter 12: Frequently Asked Questions

This section addresses the most common questions and concerns related to the Syncro 5 cobot, providing concise and clear answers for users. The FAQs cover a wide range of topics, including operation, programming, maintenance, safety features, and troubleshooting. Whether you're new to using the cobot or looking to enhance its functionality, this section serves as a quick reference guide to help you operate and maintain the cobot effectively.

If you have additional questions that are not addressed here, please refer to the relevant sections of this manual or contact Addverb's support team for assistance.

Q1: What is the maximum load capacity of this Cobot?
A: The maximum load capacity of this cobot is 5 kg. Refer to *Hardware Specifications* for details.

Q2: What kind of sensors does this Cobot use for collision detection?
A: The Syncro 5 cobot does not utilize external sensors for collision detection. It relies on internal encoders and motor currents as reference indicators to identify collisions.

Q3: Can this Cobot be programmed through a computer, or does it require a physical teach pendant to operate?
A: Both options are available:

- If a Teach Pendant is purchased, all control modes are accessible via the dedicated interface provided with the pendant.
- Alternatively, users can run various scripts as described in the user manual. In this case, direct control of the cobot (except for the emergency stop) is not possible, and the cobot performs tasks based on the specified script.

Q4: What type of power supply is required to operate this Cobot?
A: The required Power supply is:

- Input Voltage: 90-240VAC, (47 – 63) Hz
- Input Current: 16A/115VAC, 10A/230VAC

Q5: How does a cobot detect and respond to human presence or contact?
A: The Syncro 5 cobot operates in two modes to respond to contact:

- **Compliant Mode**: If the cobot encounters an object or human, it moves in the direction of the applied force.
- **Rigid Mode**: If contact force exceeds the safety limit (50N ≈ 5 Kgs), the cobot stops in a safe state.

Chapter 13: Warranty Details

Addverb warrants that its products, including the cobot and associated components, are free from defects in design, materials, and workmanship for a period of twelve (12) months from the date of **Provisional Acceptance**. This warranty ensures that Addverb is solely responsible for addressing defects caused by manufacturing faults, design flaws, or material issues, provided the customer adheres to the outlined terms.

13.1 Warranty Coverage

The warranty applies to:

- Defects in the cobot or its components caused solely by faults in manufacturing, design, or materials
- Issues identified and reported within the warranty period

13.2 Customer Obligations

For the warranty to remain valid, the customer must:

- Notify Addverb in writing of any defects promptly, without undue delay
- Report defects within two (2) weeks of their occurrence
- Ensure claims are made no later than the twelve (12)-month warranty period
  Failure to comply with these obligations may result in the nullification of the warranty

13.3 Exclusives from Warranty

The warranty does not cover the following:

- Failures due to inadequate maintenance
- Issues caused by improper or negligent operation of the cobot
- Batteries supplied with the system, regardless of their criticality
- Failures arising from usage conditions that deviate from Addverb's recommended operational guidelines
- Defects reported beyond two (2) weeks from their occurrence
- Defects resulting from customer-provided suggestions, instructions, or material
- Issues arising from unauthorized modifications or replacement of parts by the customer or a third party
- Defects caused by third-party installations, even if installation was initially within Addverb's scope
- Normal wear and tear or gradual deterioration over time under usual operational conditions

13.4 Warranty Claim Process

To initiate a warranty claim:

- **Document the Defect**:
  Provide a detailed description of the issue, including relevant observations and operational history.
- **Notify Addverb**:
  Submit the claim in writing to Addverb within the stipulated two (2)-week period from when the defect is observed.
- **Verification**:
  Allow Addverb representatives to inspect the defective equipment for validation.

13.4.1 Limitations of Liability

Addverb's responsibility under this warranty is limited to defects that are solely attributable to its design, materials, or workmanship. Addverb is not liable for defects arising from external factors, negligence, or improper use beyond its control.

13.4.2 Additional Notes

- Warranty claims must be supported by compliance with the recommended maintenance schedule and operational guidelines provided in this user guide.
- Customers are encouraged to retain all records of maintenance and operational logs to support warranty claims.
- For further information or assistance, please contact *Addverb Customer Support*.

ADDVERB

SYNCRO 5

COBOT

Software Manual

Addverb's Advanced Robotics Solutions

material handling, assembly, packaging, machine tending, quality control, and even dispensing and finishing tasks.

Chapter 14: Understanding Software

14.1 Cobot Software Installation

Detailed installation steps are provided in the *setup.md* file.

14.2 User System Setup and Compilation

It is recommended that the user's computer runs **Ubuntu 22.04 LTS**.

The cobot can be operated either:

a) **Using SDK Directly**, by connecting directly to the cobot, or

b) **Through ROS2**, by running the provided ROS 2 packages

To use ROS2, the user's system must be configured accordingly. The required setup and compilation instructions are provided in the *setup.md* file.

14.3 Configurations

The following cobot parameters can be configured as per requirement.

NOTE: The configuration step must be done before proceeding with the 14.4 Execution section on page 59

14.3.1 Payload

Payload defines the properties of the attachment at the end effector of the cobot. To modify the payload parameters:

```
nano ~/cobot_ws/cobot_ros2/addverb_cobot_control/config/payload_config.yaml
```

Description of parameters:

- **payload_status:** Expects a boolean value **(1 or 0)**. Use **1**, when you have an attachment at the end-effector, **0** otherwise. If this field is set to **0**, further parameters need not be set in the YAML
- **gripper_type**: Expects an integer between 0 to 5, indicating the make of the gripper
  - **0**: No Gripper
  - **1**: Dynamixel Gripper
  - **2**: Robotiq Gripper
  - **3**: DH Gripper
  - **4**: Suction Gripper
  - **5**: Feetech Gripper

- **ft_type**: Expects an integer between 0 to 2, indicating the make of the torque sensor
  - **0**: No Sensor
  - **1**: Robotiq
  - **2**: Robotous
- **ft_rot:** Expects a 3x3 rotation matrix, that relates the orientation of the FT sensor with respect to the base frame
- **mass:** Expects a **double** value, indicating the mass (Kg) of the attachment
- **comx, comy, comz:** Expect a **double** value, representing the Centre of Mass (m) of the cumulative system attached at the end-effector
- **Ixx, Iyy, Izz, Ixy, Ixz, Iyz:** Expect a **double** value, representing the Moment of Inertia ($kg.m^2$) of the cumulative system attached at the end-effector
- **Safety type:** Expects an integer between (0-2). Details of the same are given below in section 14.3.2

14.3.2 Safety type

The Cobot supports **three safety levels**, which define how it responds to external forces and constraints:

- **Safety 0**
  - Baseline safety mode
  - The robot can be physically stopped by hand
  - Human operators can work in the robot's vicinity with minimal risk, as the system ensures compliance and safety stops cobot
- **Safety type 1**
  - Reduced safety compared to Mode 0
  - Human accommodation is not guaranteed
  - This mode allows faster or less restricted operation but should not be used in close human collaboration scenarios
- **Safety type 2**
  - **Least safe when used in conjunction with human** – designed primarily for error recovery
  - Position and joint limits are reset to higher values, bypassing normal safety constraints
  - **Not recommended for regular use** – should only be applied when explicitly required to recover from faults

14.4 Execution

Follow README.MD for a quick start. And for an in-depth understanding of the controllers, or examples refer to the sections below in Available Controllers

NOTE: Follow the (Execution) section given in the README.MD. Controllers will be available only if the controller manager is launched. It is necessary to launch it before using any commands from the following sections

14.5 Controllers Overview

- **Checking Active Controllers:**

To list all controllers and see which ones are active, run:

```
ros2 control list_controllers
```

- **Switching Controllers:** Refer README.MD

14.6 Available Controllers

14.6.1 Recorder Controller

The Recorder Controller enables recording and replaying of cobot joint trajectories.

- Record mode: Captures joint positions at a specified frequency and saves them in a file with a user-defined label at ROS side.
- Replay mode: Loads a recorded trajectory and replays it on the cobot for a user provided number of iterations
- By default, the controller switches between Free Drive mode which allows user to freely move the robot by hand and Replay mode:
  - When the user chooses to record, the cobot switches to Free Drive mode
  - When the user chooses to replay, the cobot switches to Replay mode automatically

EXCECUTION:
Prerequisite:

a) Ensure the controller manager is active, if not launch using cobot_control.launch.py

b) Switch controller and activate recorder_controller. Ensure other controllers are deactivated.

c) Recording Trajectories
The controller exposes a service:

- /recorder_controller/record_mode
- Type: addverb_cobot_msgs/srv/Record
- **Fields:**
  label (string) → name of recording file
  - rate (int) → sampling frequency in Hz
  - enable (bool) → start (true) or stop (false) recording

d) When recording starts, the cobot switches to Free Drive mode, allowing the user to guide it manually.

a) **Example: Start Recording:**

Command:

```
ros2 service call /recorder_controller/record_mode addverb_cobot_msgs/srv/Record "{enable: true,label: "test",rate: 5}"
```

b) Example: Stop Recording:

```
ros2 service call /recorder_controller/record_mode addverb_cobot_msgs/srv/Record "{enable: false,label: "test",rate: 5}"
```

c) Replaying Trajectories:
The controller provides an action server:
/recorder_controller/replay_mode

Type: addverb_cobot_msgs/action/Replay

Goal:
- label (string) → recording to replay
- iterations (int) → number of times to repeat

Feedback:
- Current joint positions
- Iteration count

Result:
"SUCCESSFUL" or "FAILURE"

When replay starts, the robot automatically switches to Replay mode.

d) Example: Replay a Recorded Motion:
ros2 action send_goal /recorder_controller/replay_mode
addverb_cobot_msgs/action/Replay"{label: 'pick_motion', iterations: 2}"

LIMITS:
- Max recording rate: 30 Hz – Rate at which logging is done
- Position tolerance: 0.03 rad – Position tolerance limit for replay mode tracking
- Time tolerance: 0.01 s – Time tolerance limit for replay mode tracking

NOTE: If tolerances are exceeded during replay, the action may fail

PRECAUTIONS:
- Only one recording or replay can run at a time
- Starting a new recording automatically cancels replay, and vice versa
- Avoid recording while external controllers are active this can lead to corrupted trajectories

14.6.2 Free Drive Controller

This controller switches the cobot to free drive mode. In this mode, users can physically move the cobot freely without resistance. The controller does not require any commands from the terminal or scripts.

EXECUTION:
Step 1:
Controller manager must be active. If not, launch it using cobot_control.launch.py

Step 2:
Switch controller to “free_drive_controller”

14.6.3 Joint Jogging Controller

When running:

- Listen for **joint jogging velocity commands** on a dedicated ROS_2 topic
- Map commands to its hardware interfaces
- Continuously update joint velocities until stopped, replaced, or limited by safety checks

DEFAULT BEHAVIOUR:

- **On activation:**

All joint velocities set to 0.0

- **On receiving no commands:**

Velocities remain 0.0

- **On invalid commands:**

Robot logs a warning

INPUT VARIABLE:

- The parameter commands is a string[] that lists hardware command interface names, such as "joint1/velocity". An example value is {linear: ["joint1/velocity", "joint2/velocity", ...]}, representing interfaces for controlling joint velocities
- The parameter is the ROS topic ~/joint_jogging, which uses the message type addverb_cobot_msgs/msg/JointJoggingVelocity. This topic serves as the main command input and includes a jvel_scaling_factor[] array that specifies velocity scaling for each joint. For example, [1.0, 0.0, 0.0, 0.0, 0.0, 0.0] commands the robot to jog joint1 forward

BEHAVIOUR AND PARAMETER EFFECTS EXAMPLES:

- **Changing jvel_scaling_factor[0] from 1.0 → 1.5**

Cobot will jog joint 1 faster in the positive direction

EXCECUTION:
Prerequisite:
Ensure the controller manager is active, if not launch using cobot_control.launch.py

METHOD 1: Run the demo program
Command:
ros2 run examples demo_joint_jogging

Note: This command jogs each joint in loop.

METHOD 2: Send manual input
Command: Send motion
ros2 topic pub /joint_jogging_controller/joint_jogging/command \
addverb_cobot_msgs/msg/JointJoggingVelocity \
'{jvel_scaling_factor: [1.0, 0.0, 0.0, 0.0, 0.0, 0.0]}'

14.6.4 Cartesian Jogging Controller

When running:

- Cobot listens for geometry_msgs/msg/Twist commands on a dedicated ROS 2 topic
- Maps the **linear and angular components**
- Continuously updates velocities until stopped, replaced, or prevented by safety limits

DAFAULT BEHAVIOUR:

- **ON activation:**

All velocities set to 0.0

- **No incoming command**

End-Effector stays stationary

- **Invalid commands**

Ignored, warning logged

INPUT VARIABLES:

- The **commands parameter** is a **string[]** listing hardware command interface names for controlling linear velocities, e.g., {linear: ["linear_x_velocity", "linear_y_velocity", "linear_z_velocity"]}
- The ROS topic ~/cartesian_jogging/command uses geometry_msgs/msg/Twist messages to control robot movement, using only linear.x, linear.y, and linear.z for direction and velocity. For example, {linear: {x: 1.0, y: 0.0, z: 0.0}, angular: {x: 0.0, y: 0.0, z: 0.0}} moves the robot along the X-axis

BEHAVIOUR AND PARAMETER EFFECT EXAMPLES:

Change	Effect
Set linear.x = 1.0	Moves end-effector **forward** along +X axis
Set linear.z = -0.5	Moves end-effector **down** at half speed

EXECUTION:

Prerequisite:

Ensure the controller manager is active, if not launch using cobot_control.launch.py

METHOD 1: Run the demo program

Command:

```
ros2 run examples cartesian_jogging
```

METHOD 2: Send manual input

Command 1: Send motion

Example: Send a simple +Y jog command:

```
ros2 topic pub /cartesian_jogging_controller/cartesian_jogging/command \
  geometry_msgs/msg/Twist \
  '{linear: {x: 0.0, y: 1.0, z: 0.0}, angular: {x: 0.0, y: 0.0, z: 0.0}}'
```

14.6.5 Joint Impedance Controller

Impedance control allows you to command both the target joint positions and the mechanical stiffness and damping of each joint, making the motion compliant or rigid depending on your needs.

When running, the controller will:

- Accepts a goal trajectory that contains joint positions and time to reach each point
- Apply stiffness and damping gains per joint
- Continuously track the trajectory until it completes, or violates constraints
- Validate all incoming data for safety and reject invalid command

DAFAULT BEHAVIOUR:

- **On activation:**

Stiffness and damping set to configured values

- **On invalid command:**

Command ignored
Stiffness and damping reset to defaults if invalid

INPUT VARIABLES:

- The **parameter commands** is a **string[]** that lists hardware command interface names for positions, such as ["joint1/position", "joint2/position", "joint3/position", ...], representing the interfaces used to control joint positions
- The **parameter stiffness** is a **double[]** representing per-joint stiffness gains. For example, [200.0, 200.0, 200.0, 200.0, 200.0, 200.0] sets the stiffness value for each joint
- The **parameter damping** is a **double[]** specifying per-joint damping gains. For example, [5.0, 5.0, 5.0, 5.0, 5.0, 5.0] sets the damping values for each joint.
- The parameter is the ROS action ~/follow_joint_trajectory of type control_msgs/action/FollowJointTrajectory serves as the main command input and includes trajectory.joint_names along with trajectory.points, which contain joint positions and timing information through positions[] and time_from_start.

BEHAVIOR AND PARAMETER EFFECT EXAMPLES:

- **Increasing stiffness[0] from 200.0 → 400.0**
  Joint 1 becomes more rigid, resisting external forces
- **Providing an invalid stiffness vector length**
  Falls back to default gains and logs a warning
- **Setting goal_time_tolerance too small** (Where **goal_time_tolerance** is the extra time after the segment end time allowed to reach the goal)
  Note: Falls back to default tolerance (1.2 s) with a warning

EXECUTION
Prerequisite:
Ensure the controller manager is active, if not launch using cobot_control.launch.py

METHOD 1: Run the demo program:
ros2 run examples demo_joint_impedance

NOTE: This above method defines a ROS 2 demo node that **sends a multiple joint trajectory point** to the joint_impedance_controller.

METHOD 2: Send manual input:

Param call to change the stiffness and damping:
ros2 param set /joint_impedance_controller stiffness "[200.0, 200.0, 200.0, 200.0, 200.0, 200.0]"
ros2 param set /joint_impedance_controller damping "[5.0, 5.0, 5.0, 5.0, 5.0, 5.0]"

Command: Send Motion

```
ros2 action send_goal /joint_impedance_controller/follow_joint_trajectory
control_msgs/action/FollowJointTrajectory "{
 trajectory: {
  joint_names: ['joint1', 'joint2', 'joint3', 'joint4', 'joint5', 'joint6'],
  points: [
   {
    positions: [0.1, -0.2, 0.3, -0.4, 0.5, -0.6],
    time_from_start: {sec: 10, nanosec: 0}
   }
  ]
 },
 goal_time_tolerance: {sec: 1, nanosec: 4}
}"
```

WARNINGS AND ERRORS:

Log Message	Meaning
Stiffness param invalid or missing. Using default	Incorrect stiffness array size
Damping param invalid or missing. Using default	Incorrect damping array size
Goal validation failed: joint_names is empty	Invalid goal
Joint Impedance data validation failed.	Impedance data stiffness and damping matrix are invalid
Incorrect time tolerance provided, using default value	Time tolerance is less than default time tolerance (1.2 seconds)

PRECAUTION

- **Parameter size checks:**
  stiffness and damping arrays must each be length 6 (for 6 Cartesian DOF)
- **Trajectory constraints:**
  - joint_names.size() must match each point's positions.size()
  - The 'time_from_start' values must increase strictly with each point

EXAMPLE: Suppose the goal time for second point occurs before the goal time of first point then the trajectory is incorrect

- **Execution safety:** If validation fails mid-motion, impedance resets to defaults and error logged

14.6.6 Cartesian Impedance Controller

Unlike the Joint Impedance Controller which regulates stiffness and damping per joint, this controller regulates the cobot's end-effector behaviour in Cartesian space

With Cartesian impedance, you can command both:

- A desired end-effector trajectory (pose over time)
- The mechanical stiffness and damping in each Cartesian direction (X, Y, Z, and rotations), making the cobot compliant or rigid in task space

When running, the controller will:

- Accepts a goal trajectory that contains cartesian targets and time to reach each target
- Apply stiffness and damping gains in cartesian space
- Track the trajectory until completion, or constraint violation
- Validate all incoming data for safety

DAFAULT BEHAVIOUR:

- **On activation:**

Stiffness and damping set to configured values (or defaults)

- **On invalid command:**

Command ignored, stiffness and damping reset to defaults if invalid

INPUT VARIABLES:

- The **commands** parameter is a **string[]** that defines hardware interfaces for Cartesian position/orientation control, such as ["end_effector/pose"]
- The **stiffness** parameter is a **double[6] array** that defines Cartesian stiffness gains for the X, Y, Z translational axes and the Rx, Ry, Rz rotational axes. An example is [200.0, 200.0, 200.0, 20.0, 20.0, 20.0]
- The **damping parameter** is a **double[6] array** specifying Cartesian damping gains for the X, Y, Z translational axes and the Rx, Ry, Rz rotational axes. An example value is [5.0, 5.0, 5.0, 1.0, 1.0, 1.0]

- The **Mass_matrix parameter** is a **double[6]** array representing the diagonal mass terms used in Cartesian impedance control. An example is [1, 1, 1, 0.1, 0.1, 0.1]
- The **ft_force parameter** is a **double[6]** array that represents the measured or estimated wrench to be passed to the hardware.. An example is [0, 0, 0, 0, 0, 0]
- The **target_force parameter** is a **double[6]** array that defines the desired wrench to be tracked. An example value is [0, 0, 10, 0, 0, 0]

BEHAVIOR AND PARAMETER EFFECT EXAMPLES:

- **Increasing stiffness[0] from 200.0 → 500.0**
  End-effector resists user interference more strongly along X-axis
- **Sending a trajectory with multiple Cartesian waypoints**
  The controller streams through each pose with the given timing
- **Providing an invalid stiffness vector length**
  Falls back to defaults and logs a warning
- **Setting goal_time_tolerance too small**
  Reverts to default (1.2 s) and logs a warning

EXECUTION:
Prerequisite:
Ensure the controller manager is active, if not launch using cobot_control.launch.py

METHOD 1: Run the demo program:
Command:
ros2 run examples demo_cartesian_impedance

METHOD 2: Send manual input:

Param call to change the stiffness and damping:
ros2 param set /cartesian_impedance_controller stiffness "[200.0, 200.0, 200.0, 200.0, 200.0, 200.0]"
ros2 param set /cartesian_impedance_controller damping "[5.0, 5.0, 5.0, 5.0, 5.0, 5.0]"

Command: Send Motion

```
ros2 action send_goal /cartesian_impedance_controller/follow_joint_trajectory
control_msgs/action/FollowJointTrajectory "{
 trajectory: {
  joint_names: ['joint1', 'joint2', 'joint3', 'joint4', 'joint5', 'joint6'],
  points: [
   {
    positions: [0.1, -0.2, 0.3, -0.4, 0.5, -0.6],
    time_from_start: {sec: 10, nanosec: 0}
   }
  ]
 },
 goal_time_tolerance: {sec: 1, nanosec: 4}
}"
```

14.6.7 Velocity Controller

- This controller allows user to directly command velocity for each joint of the cobot.
- Expects input array of 6x1, where each element in the array represents the velocity command for that particular joint. (Joints index starts from zero). Magnitude is in rad/s

EXECUTION:
Prerequisite:
Ensure the controller manager is active, if not launch using cobot_control.launch.py

METHOD 1: Run the demo program
Command:
ros2 run examples demo_velocity

NOTE:
- The target = [-0.01,0,0,0,0,0] is sent and the cobot's base joint will keep moving at a velocity of 0.01 rad/s in clockwise direction until you stop the robot.

METHOD 2: Send manual input
Command:
```
ros2 topic pub /velocity_controller/commands std_msgs/msg/Float64MultiArray "{
  data: [0.1, 0, 0, 0, 0, 0]
}"
```

NOTE:
- [0.1,0,0,0,0,0] means publishing velocity of 0.1 rad/s to joint_1 , 0 rad/s to joint_2 and so on and so forth, If the user has sent any command, example: [0.1,0,0,0,0,0], the robot will keep moving in that direction till the user explicitly stops the robot
- Stopping the robot can be done by modifying the data members of the array to [0,0,0,0,0,0].
- With positive and negative values meaning motion in counterclockwise direction and clockwise direction respectively
- Default behaviour is to keep publishing zero velocities
- Any command which exceeds the limits will be rejected and warning published on the console

14.6.8 PTP Joint Controller

- This controller moves the cobot to one or more target positions within a specified time. Users can define position and time tolerances to allow for slight deviations and timing flexibility.
- Each target point must have only 6 values in the positions field, any value in velocity, acceleration and effort fields are ignored
- The controller rejects any new goal if a previous goal is already in execution
- You can constantly poll the joint_states as part of the feedback. Once the target position is achieved and target time lapsed, the controller updates the response of the request as detailed in the control_msgs/FollowJointTrajectory

EXECUTION:
Prerequisite:
Ensure the controller manager is active, if not launch using cobot_control.launch.py

METHOD 1: Run the demo program
Command:
ros2 run examples demo_ptp_joint

NOTE:
- This demo script sends a predefined trajectory as goal to the server.

METHOD 2: Send manual input
Command: Send motion

```
ros2 action send_goal /ptp_joint_controller/follow_joint_trajectory
control_msgs/action/FollowJointTrajectory "{
 trajectory: {
  joint_names: ['joint1', 'joint2', 'joint3', 'joint4', 'joint5', 'joint6'],
  points: [
   {
    positions: [0.1, -0.2, 0.3, -0.4, 0.5, -0.6],
    time_from_start: {sec: 5, nanosec: 5}
   },
   {
    positions: [0.2, -0.1, 0.4, -0.3, 0.6, -0.5],
    time_from_start: {sec: 10, nanosec: 15}
   }
  ]
 },
 goal_time_tolerance: {sec: 1, nanosec: 20}
}"
```

NOTE: Notice that each subsequent target position has a goal time greater than previous positions.

14.6.9 PTP TCP Controller

- This controller allows user to give three or more target positions for end-effector of the cobot along with the target time to attain each target configuration
- Goal to this controller can be given by implementing an action client giving an action of type “FollowCartesianTrajectory” which is a custom message type

The components of this message type are explained:

- CartesianTrajectory is an array of CartesianTrajectoryPoint, where the Cartesian Trajectory Point has components:
  - **CartesianPoint**: target configuration
  - **time_from_start**: time to attain given configuration starting from the beginning

- CartesianPoint has components:
  - **geometry_msgs/Point**: target configuration x, y and z values
  - **geometry_msgs/Vector3**: target orientation roll, pitch and yaw values

EXECUTION:
Prerequisite:
Ensure the controller manager is active, if not launch using cobot_control.launch.py

METHOD 1: Run the demo program
Command:
ros2 run examples demo_ptp_tcp

METHOD 2: Send manual input
Note: ptp_tcp needs a minimum of 3 points to execute a trajectory.
Command:

```
ros2 action send_goal /ptp_tcp_controller/follow_cartesian_trajectory
addverb_cobot_msgs/action/FollowCartesianTrajectory '{
 "trajectory": {
  "points": [
   {
    "point": {
     "position": {"x": 0.3, "y": 0.3, "z": 0.3},
     "orientation": {"x": 0.0, "y": 0.0, "z": 0.0}
    },
    "time_from_start": 3.0
   },
   {
    "point": {
     "position": {"x": 0.5, "y": 0.5, "z": 0.5},
```

```
      "orientation": {"x": 0.0, "y": 0.0, "z": 0.0}
     },
     "time_from_start": 7.0
    },
    {
     "point": {
      "position": {"x": 1.0, "y": 1.0, "z": 1.0},
      "orientation": {"x": 0.0, "y": 0.0, "z": 0.0}
     },
     "time_from_start": 10.0
    }
   ]
  }
}'
```

NOTE: Notice that each subsequent target point has a goal time greater than previous points.

PRECAUTIONS:

- Controller will reject a goal if target configuration has lesser than 3 points, a minimum of 3 points is required for multi-point target
- Ensure the coordinates are in workspace of the cobot

14.6.10 Effort Controller

- The Effort Controller expects joint torque commands (in Nm) as input and applies them directly to the robot joints, without adding gravity compensation torques. This provides full control to the user over the applied torques
- Once user has request to switch to effort controller, robot will first go to home configuration before switching to effort control mode
- The controller remains inactive until it receives a valid non-zero effort command from the user. When any joint receives a non-zero torque command, the controller transitions to active mode and begins processing effort commands

EXECUTION:
Prerequisite:
Ensure the controller manager is active, if not launch using cobot_control.launch.py
Launch Controller:
In cobot_control_launch.py, ensure the node for the controller spawner is included:
nodes = [..., effort_controller_spawner]

METHOD 1: Run the Demo Program
Command:
ros2 run example demo_effort

NOTE: Torque Limits:
Absolute Value: 60
Relative Value: 40

If these limits are exceeded, the robot will go in error.

METHOD 2: Send Manual Input
Command 1: Send Motion
ros2 topic pub /effort_controller/commands std_msgs/msg/Float64MultiArray "data: [0.0, -22.0, 22.0, 0.0, 0.0, 0.0, 0.0]"

NOTE: This command activates the controller since a non-zero effort is commanded for one joint.

PRECAUTIONS
• The controller activates only when at least one joint receives a non-zero torque command
• This controller does not include gravity compensation; ensure your commanded torques account for gravity if necessary
• Apply gradual torque changes to prevent abrupt or hazardous robot behavior

14.6.11 Gravity Compensation Effort Controller

- The Effort Controller expects joint torque commands (in Nm) as input and applies them to the robot. It automatically adds gravity compensation torques to these external commands, making the total torque applied to each joint:

  Total Torque = Gravity Compensation Torque + Commanded External Torque

- When zero torque is commanded or the controller is loaded without publishing commands, the robot behaves like freedrive mode

EXECUTION:
Prerequisite:
Ensure the controller manager is active, if not launch using cobot_control.launch.py
Launch Controller:
In cobot_control_launch.py, ensure the node for the controller spawner is included:
nodes = [...,gravity_comp_effort_controller]

using switch controller service switch to gravity_comp_effort_controller.

METHOD 1: Run the Demo Program
Command:
ros2 run example demo_gravity_comp_effort

METHOD 2: Send manual input
Command : Send Motion
ros2 topic pub /gravity_comp_effort_controller/commands std_msgs/Float64MultiArray "data: [0.0, 0.1, 0.0, 0.0, 0.0, 0.0]"

Precautions:

Avoid canceling gravity compensation: Sending external torques that directly oppose gravity compensation can reduce the total torque to near zero, causing the robot to fall under gravity. This may lead to unexpected motion or safety faults.

14.6.12 Gripper Controller

- The Gripper Controller provides a ROS 2 interface for commanding a cobot's gripper position
  - **Command mode:** Accepts user requests for gripper position and grasp force
  - **Feedback:** Returns success or failure of the requested operation
- By default, the controller stays idle until a valid service request is received. Users can command grasp force only for closing the gripper
- The Gripper Controller provides a validated and safe way to control the cobot gripper through ROS 2 services. It supports position and grasp force commands, ensures requests are within limits, and delivers feedback on execution success
- This makes it a reliable component for demonstrations, manipulation tasks, and automation workflows involving cobot grippers

EXECUTION:
Prerequisite:
Ensure the controller manager is active, if not launch using cobot_control.launch.py

METHOD 1: Run the demo program
Command:
ros2 run examples demo_gripper

NOTE:
The controller exposes a ROS 2 service:
- Service Name: /gripper_controller/command
- Type: addverb_cobot_msgs/srv/Gripper

Service Definition
Request:
- *float64 position*
- *float64 grasp_force*

Response:
- *bool success*
- *string message*

METHOD 2: Send manual input
Command 1: (Open Gripper):

ros2 service call /gripper_controller/command addverb_cobot_msgs/srv/Gripper "{position: 1.0, grasp_force: 0.0}"

Response:
success: true
message: "success"

Command 2: (Close Gripper):
ros2 service call /gripper_controller/command addverb_cobot_msgs/srv/Gripper "{position: 0.0, grasp_force: 100.0}"

Response:
success: true
message: “success”

LIMITS

- Position values are normalized over various grippers and are in the range (0 to 1)
- Currently only position value (0 and 1) is supported
- Position value 0 (close) and 1 (open)
- Grasp force limits: 15 Kgf
- Commands are applied sequentially and may take 2 seconds to complete

PRECAUTIONS

- Invalid or unsafe requests are rejected with an error message
- Sending duplicate commands (e.g., close when already closed) does not move hardware
- Incorrect parameter mappings in payload.yaml can lead to wrong execution

14.7 Shutdown Service And Error Recovery

Service call for shutdown:

```
ros2 service call /cobot_services/shutdown_srv std_srvs/srv/Trigger {}
```

Shutdown Service (/cobot_services/shutdown_srv)
This service cleanly shuts down the robot but only succeeds if the cobot is **not in an error state**. If the cobot is in error, shutdown will fail. The server must already be launched on the cobot side, and the ROS 2 control manager must be running on the user's laptop for this service to work. The service is called a blocking call from the user's laptop Command line interface and returns success = true upon successful shutdown.

Service call for error recovery:

```
ros2 service call /cobot_services/error_recovery_srv std_srvs/srv/Trigger {}
```

Error Recovery Service (/cobot_services/error_recovery_srv)
This service resets the cobot from an error state and commands it to move back to the home position. It must be called before shutdown if the cobot is in error. Like shutdown, the server must be running on the cobot and the ROS 2 control manager on the user's laptop. This service is also called from the laptop's command line interface as a blocking call and returns success = true when recovery completes successfully.

Usage Flow:
If the cobot is in error, call the error recovery service first from your laptop's command line interface.
Then call the shutdown service from the laptop's command line interface power off the robot cleanly.

14.8 Read FT Sensor Data

If FT sensor has been provided along with the robot, user can read the data by subscribing to the topic /ft_data.

To read the FT data through command line:
ros2 topic echo /ft_data

Details of the message type:
Data type of the published message is geometry_msgs/msg/Wrench
where geometry_msgs/msg/Wrench consists of

- geometry_msgs/Vector3 force: a 3x1 vector giving force values in X, Y and Z directions (in N)
- geometry_msgs/Vector3 torque: a 3x1 vector giving torque values about X, Y and Z directions (in Nm)

14.9 Troubleshooting

14.9.1 If the controllers are not working

- Check if multiple controllers are active
- Check if the desired controller is active. If not activate it
- Only the gripper_controller can run alongside another controller
- If multiple controllers are active, **deactivate all** and **reactivate** only the required one

14.9.2 If gripper is not working

- Verify that **payload _status** is set to **1**
- Ensure the **gripper_type** parameter is correct
- Confirm that the gripper is **powered** and **properly connected** to the control box

14.9.3 Scripts on the Cobot not working

If the CLI halts after displaying specific messages:

- **"Port Initialized"**: Check if the E-Stop is pressed
- **"heard"**: The script may not be connecting to the controller manager
  - Possible cause: multiple instances of server running
  - Solution: kill duplicate processes or restart the cobot
- **"Setting relay on"** → Press the reset button on the PCB inside the control box, then retry running the script

14.9.4 Unable to connect to the Cobot

- Ensure the user's laptop IP address is static
- Verify that it matches the required network configuration for the cobot

ADDVERB

Plot No. 5, Sector 156, Noida, Uttar Pradesh 201301

+91 9871590101
automate@addverb.com
addverb.ai