# User Manual

## KUKA Integration

Setting up the MIRAI system

Implementing MIRAI skills in KUKA Program

VERSION 21.0.0

micropsi
industries

# Contents

# Introduction

Micropsi Industries' robot control system, MIRAI, allows users to automate complex motion tasks in dynamic process environments by manually training robots. MIRAI is a machine learning-based system that generates robot movements based on sensory input and (re-)acts in real-time based on image data.

Today, most automation solutions are programmed in a script language or 'taught' using a teach pendant and its user interface. MIRAI, on the other hand, can train robots to solve complex hand-eye coordination tasks by 'watching' these being performed through a human operator and 'imitating' the actions seen. Key advantages of MIRAI compared to more 'classical' automation approaches are: first, the ease of use in solving and automating complex motion tasks without previous expert knowledge in automation, machine vision or programming. Second, MIRAI's underlying machine learning technology is inherently able to handle different sources of process variances and dynamics within the task and its environment. With this, the MIRAI controller can solve new classes of tasks and problems in automation and robotics that normally required complex and expensive solutions or were impossible to automate before.

To 'train' the robot, the user scans, either manually or using "recording assistant", the area between the precise target position and the intended starting position(s). These scanned demonstrations are called episodes. For more complex tasks, the user performs and records repeated demonstrations of a task, by guiding the robot from varying starting positions to the target position, in a very precise trajectory. The user manages the camera recording of the respective episodes through our MIRAI Training App for Android tablets. These episodes are then transformed into a vision-based robot motion program by our cloud-based machine learning service, which results in a 'MIRAI skill'. These trained skills ultimately allow the MIRAI controller to imitate the motions and actions it was shown by steering robot movements, dealing with all the dynamics and variances the user trained for in real time (closed loop).

# 1 Getting Started

## 1.1 Safety Precautions

### 1.1.1 Robot Safety

> **CAUTION**
>
> We recommend using Manual Reduced Velocity mode (T1) when working with MIRAI.

To ensure safety, follow these precautions:

- Always keep a safe distance from the robot:
- Keep the KUKA smartPAD with an EMERGENCY STOP button within reach at all times.
- Follow all safety precautions in the KUKA robot safety guidelines.

Manual Reduced Velocity mode (T1) is recommended when using MIRAI. The method for selecting the operation mode (AUTO/T1/T2) depends on your robot and controller model. On the robot controller, use either a key switch or turn-knob. On the KUKA smartPAD, use the Mode Select function. For more details on mode selection, refer to the operator's manual.

After you train a MIRAI skill and add it to the robot program, you can change the operation mode back to T2 or AUTO.

### 1.1.2 The Enabling Switch (Deadman Switch)

> **IMPORTANT**
>
> You must hold the enabling switch on the KUKA smartPAD while using the MIRAI Training App.

When the enabling switch is pressed (in the middle position), the [I] indicator is green. If the switch is released, the indicator turns off, and the KUKA smartPAD will display this message: "RSI cannot set any outputs due to operator protection." Certain features in the MIRAI Training App will be unavailable until the enabling switch is pressed again. To continue using the MIRAI Training App, hold the enabling switch in the middle position.

## 1.2 Supported KUKA models

MIRAI is compatible with all KUKA robot models that meet the following requirements:

- KR C4 or C5 controller
- KUKA.SystemsSoftware (KSS) 8.6 or 8.7.

MIRAI has been tested on the following KUKA robot models so far:

- KR 10 R900-2
- KR 1100-2
- KR 4 R600

## 1.3 MIRAI Components

### 1.3.1 MIRAI Kit

The MIRAI kit includes the following components:

- MIRAI controller: Siemens SIMATIC IPC BX-39A – power supply not included
- Android-based tablet with the MIRAI Training App
- Camera setup:
  - Cameras: USB 3.0 cameras (XIMEA xiQ) or GigE cameras (Baumer VCXG.2-15C.I)
  - Camera lenses (9 mm and 16 mm)
  - Connection cables
  - Ring lights and adapters
  - Camera mount (optional)
- *For USB 3.0 camera setups only:* Ethernet Gigabit switch and cables

### 1.3.2 Additional Components

Some additional components are required to use the MIRAI system, and some components are optional. All additional components must be procured by the customer

**Required components**

- a 24 V power supply to provide power to the MIRAI controller. For other options (230V/110V), contact your account executive
- *For GigE camera setups only:* Siemens SCALANCE XC208G PoE Switch. For other options, contact your account executive.

**Optional components** for using the hand-guiding feature:

- One of the following force/torque sensors:
  - OnRobot HEX-E v2
  - OnRobot HEX-QC
  - ATI Axia80-M20 with adapters
  - Any ATI sensor supporting the Network Force/Torque (NET F/T) system (tested with ATI-9105-Net-Gamma)

To configure the sensors, see: Configuring the OnRobot Compute Box and Configuring the ATI Sensors.

## 1.4 Required KUKA Software Options

- KUKA robot arm and control system: KR C4 or KR C5
- KUKA.SystemSoftware (KSS): Version 8.6 or 8.7
- The following software packages must be purchased and installed on the KUKA robot controller:

| Software Packages | Version | Provided by |
|---|---|---|
| KUKA.Ethernet KRL | Version 3.0–3.2 | KUKA |
| KUKA.RobotSensorInterface (RSI) | Version 4.1 and higher for KR C4<br>Version 5 and higher for KR C5 | KUKA |

- For conveyor functionality only:

| Software Package | Version | Provided by |
|---|---|---|
| KUKA.ConveyorTech | Version 8–8.2 | KUKA |

## 1.5 Compatibility between Micropsi Industries Software Versions

A KOP file (KUKA Option Package file) is required to integrate and manage MIRAI capabilities on the KUKA controller. KOP files. Two MIRAI KOP files available:

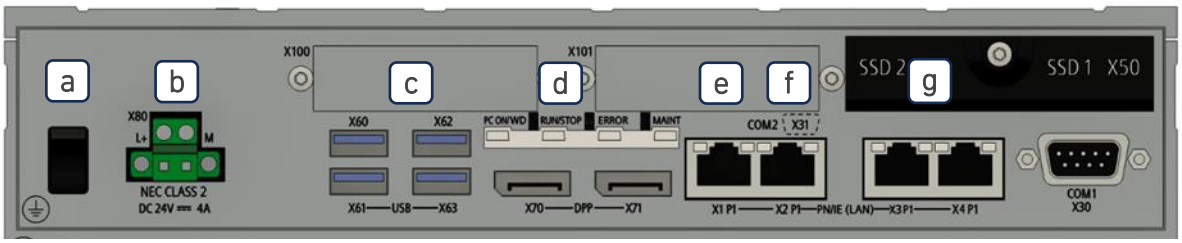
1. **Mirai.kop**: Contains configurations and data to activate MIRAI capabilities
2. **MiraiConveyor.kop**. Identical to Mirai.kop but includes additional conveyor functionality

You will receive the appropriate KOP file based on your need for conveyor functionality. Note that both KOP files cannot be installed simultaneously on the KUKA controller.

Different versions of MIRAI require specific versions of the KOP files for compatibility, as listed below.

| MIRAI Version | Compatible with: Mirai.kop Version | Compatible with: MiraiConveyor.kop Version |
|---|---|---|
| Starting MIRAI 15.0.0 | Version 1.0.3 | - |
| Starting MIRAI 16.0.0 | Version 1.0.4 | - |
| Starting MIRAI 18.0.0 | Version 1.0.5 | - |
| Starting MIRAI 18.1.4 | Version 1.1 | - |
| Starting MIRAI 21.0.0 | Version 2.0 | Version 2.0 |

## 1.6 MIRAI Controller Interface Description




a. Power on/off
b. Power supply (24V, not included)
c. 4 USB 3.2 ports
d. RUN/STOP LED
e. Ethernet port X1P1, configured to 192.168.100.5
f. Ethernet port X2P1, for WAN/internet DHCP
g. Link-aggregated Ethernet ports X3P1 and X4P1, configured to 192.168.99.5

## 1.7 MIRAI Controller Specifications

- Intel Xeon W-11555MLE (1.9/4.4 GHz, 6C/12T)
- 16GB DDR4-3200 SO-DIMM
- SSD 512 GB Eco
- 24 V power supply
- Operating conditions: 0° C to 55° C. Humidity 5% to 80% at 25° C (no condensation)
- Dimensions:
  - Width: 262 mm
  - Height: 139.7 mm
  - Depth: 62.6 mm

# 2 Setting up the MIRAI Controller

The MIRAI controller is a compute and control unit. It uses camera images to calculate robot movements in real time.

To set up the MIRAI controller, first assemble the camera and force/torque sensor stack (Section 2.1). Then connect the MIRAI controller to all system components. Different cameras require specific setups:

- For USB 3.0 cameras, see Section 2.2.
- For GigE cameras, see Section 2.3.

## 2.1 Assembling and Mounting the Camera and Force/Torque Sensor Stack

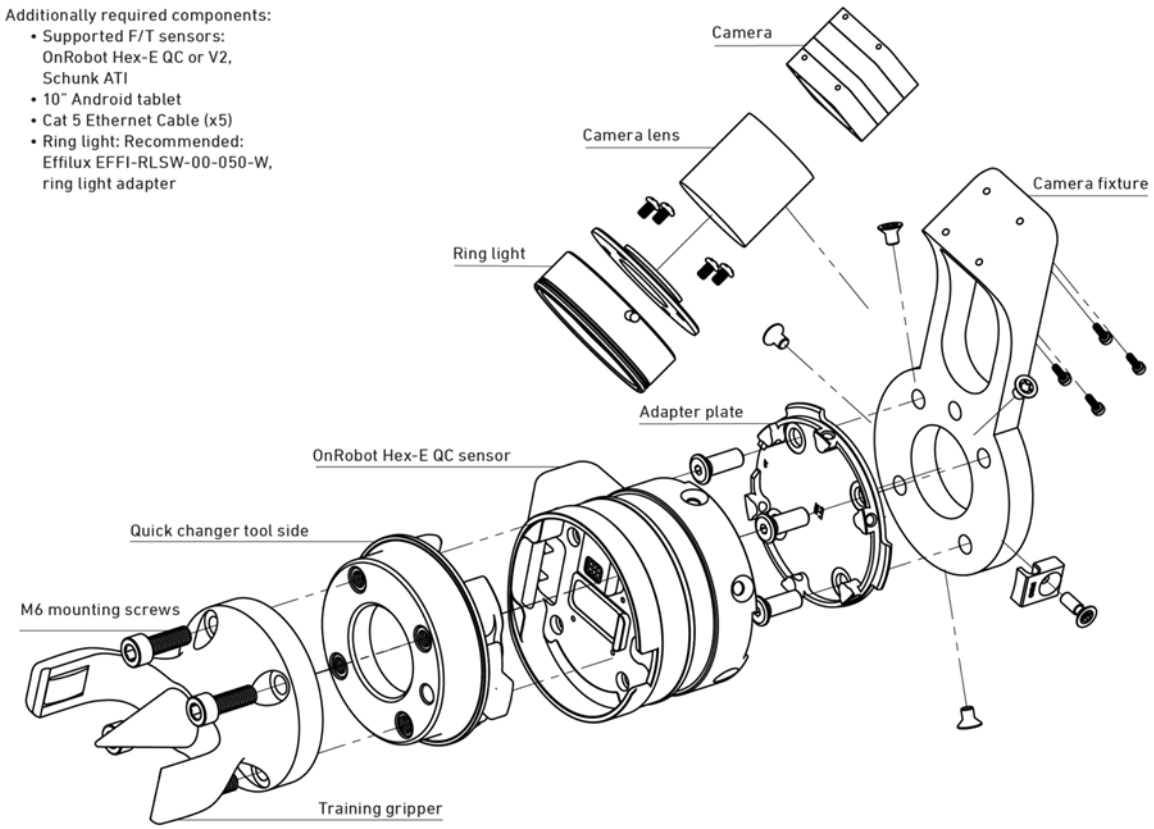



*Figure 1: Diagram of the correct order for assembling the camera and force/torque sensor stack.*

| IMPORTANT: The physical MIRAI setup must be assembled in the order shown above |
|---|

To mount cameras and the force/torque sensor on the robot arm, follow these steps:

1. Attach the camera lens to the camera.
2. Screw the ring light adapter and the ring light on to the lens.
3. Mount the camera on the camera fixture with the provided screws.
4. Mount the assembled camera fixture on the robot tool flange using the adapter plate and the provided screws.
5. Attach the force/torque sensor to the adapter plate with the provided screws.
    - Test if the sensor is mounted correctly by ensuring that the robot arm moves in the direction in you push it in. If it moves in a different direction, you need to remount.

6. Click on the quick changer tool adapter.
7. Attach the training gripper or other end-of-arm tool.
8. Connect the cable with the camera and secure the cable to the robot arm, leaving enough slack to allow the robot arm to move freely.

## 2.2 USB 3.0 Camera setups

### 2.2.1 MIRAI Setup Scheme with USB 3.0 Cameras

This schematic diagram of a MIRAI-controlled robot setup shows the components of a MIRAI-based solution. The diagram shows all supported peripherals and how they connect through various interfaces.

The MIRAI solution includes the elements below:

- **MIRAI robot controller:** generates sensor-based, real-time robot movements based on trained skills
- **MIRAI Training App:** primary user interface for the MIRAI controller; runs on an Android-based tablet and allows users to record training episodes and manage MIRAI skills
- **Micropsi Industries cloud:** calculates MIRAI skills using training episodes

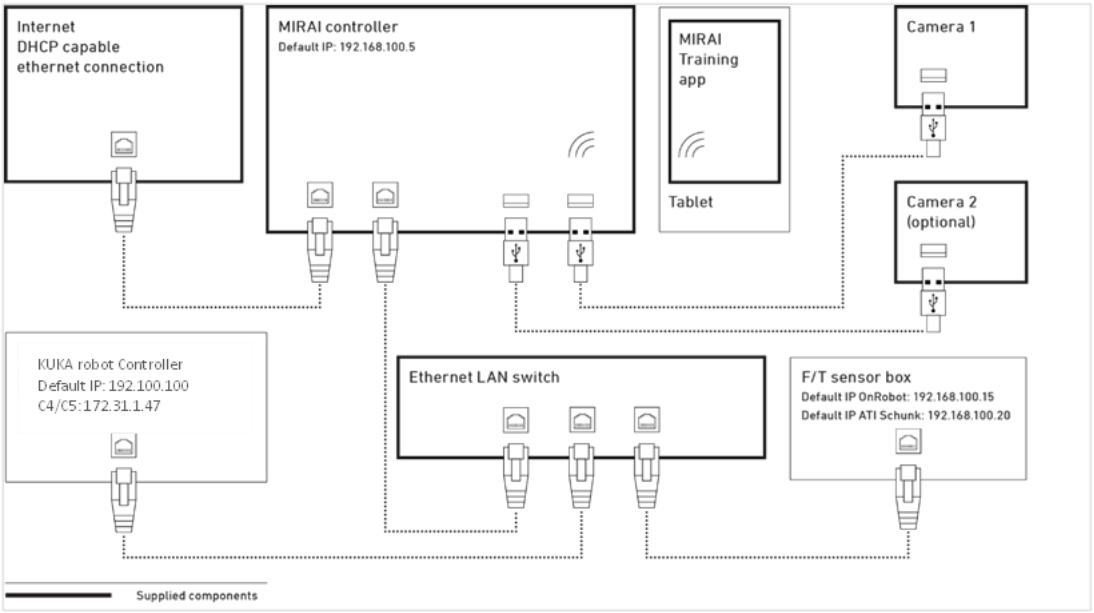



*Figure 2:* Schematic view of a MIRAI-controlled robot setup with USB 3.0 cameras

### 2.2.2 Mounting and Cabling the MIRAI Controller with USB Camera Setup

**NOTE:** Refer to Section 1.6 for a diagram of the MIRAI controller.

To connect all components in the MIRAI-controlled robot setup, follow these steps:

1. Place the MIRAI controller and the Ethernet LAN switch near the KUKA robot controller or robot arm. We recommend securing the MIRAI controller to prevent it from falling or shifting.
2. Connect the KUKA robot controller to the Ethernet LAN switch.

a. For KR C4 controllers:
Connect the Ethernet cable to port X66 on the KUKA controller. This interface can support multiple virtual interfaces. For RSI, create a new virtual interface (such as virtual6, virtual7, etc.) and configure it as UDP with the IP address 192.168.100.100. Leave the default interface, virtual5, for KLI or other customer network interfaces, such as PROFINET.
b. For KR C5 controllers:
Connect the Ethernet cable to the default KPI port XF5. If an additional port, XF6, has been purchased, it can also be used for the Ethernet connection.

3. Connect the force/torque sensor box to the Ethernet LAN switch.
4. Connect the Ethernet LAN switch to Port X1P1 on the MIRAI controller, creating a local area network (LAN) for the robot control environment.
5. Connect the USB 3.0 cameras to the MIRAI controller's USB ports.
6. Connect the Wi-Fi module to a USB port on the MIRAI controller to link it with the tablet.
7. Connect the Ethernet WAN port (X2P1) on the MIRAI controller to a network with WAN/internet access. Ensure that the connected network offers DHCP service, as the IP address for this port will be assigned automatically through DHCP.
    a. NOTE: To use a proxy to connect with the Micropsi cloud, see Connecting to the Micropsi Cloud Using a Proxy Server. For information on the cloud connection and handling of recorded data, see Data FAQ in MIRAI Training User Manual.
8. Connect a 24 V power supply to the MIRAI controller to turn the controller on. Check the LED RUN/STOP indicator for a green light to confirm successful system boot and started runtime.

## 2.3 GigE Camera Setups

### 2.3.1 MIRAI Setup Scheme with GigE Cameras

These schematic diagrams show MIRAI-controlled robot setups for one GigE camera (*top*) and two GigE cameras (*bottom*). The diagrams show all supported peripherals and how they connect through various interfaces.

The MIRAI solution includes the elements below:

- MIRAI robot controller: generates sensor-based, real-time robot movements based on trained skills
- MIRAI Training App: primary user interface for the MIRAI controller; runs on an Android-based tablet and allows users to record training episodes and manage MIRAI skills
- Micropsi Industries cloud: calculates MIRAI skills using training episodes
- The MIRAI software package: integrates MIRAI skills into robot program flows on the Teach Pendant

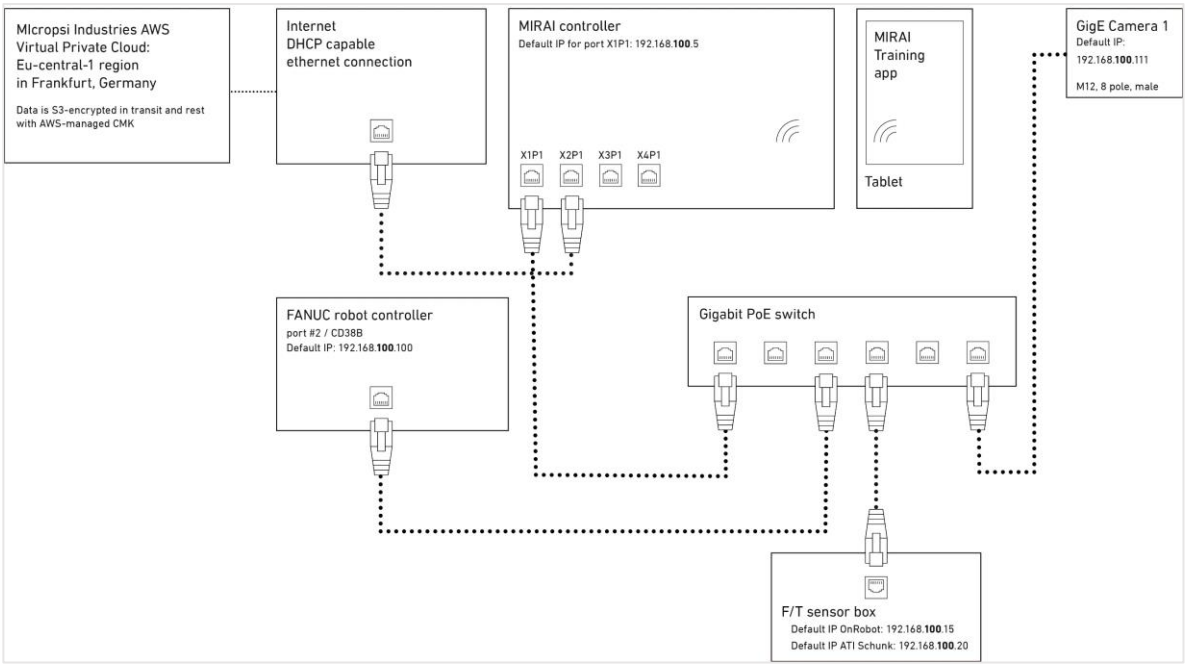



*Figure 3:* Schematic view of a MIRAI-controlled robot setup with one GigE camera

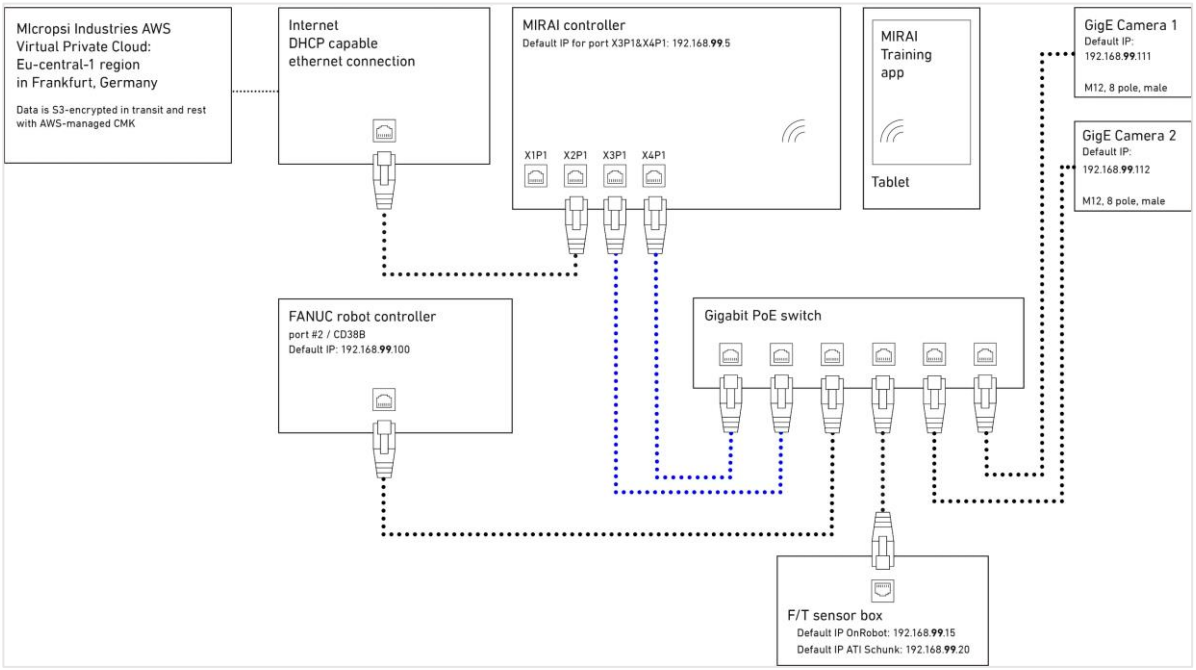



*Figure 4:* Schematic view of a MIRAI-controlled robot setup with two GigE cameras

### 2.3.2 Mounting and Cabling the MIRAI Controller with GigE Cameras:

GigE cameras are part of the MIRAI network and have an Ethernet interface. They have an IP address that needs to be configured to the same subnet as the robot, the force/torque sensor, and the MIRAI controller. The MIRAI controller has preconfigured ports for one or two cameras

Step 5 describes setups using the preconfigured ports on the MIRAI controller. The network can also be configured manually to customize settings, which may use different ports.

| NOTE: Refer to Section 1.6 for a diagram of the MIRAI controller. |
|---|

1. Place the MIRAI controller and the Gigabit Ethernet switch near the KUKA robot controller or robot arm. We recommend securing the MIRAI controller to prevent it from falling or shifting.
2. Connect the KUKA robot controller to the Gigabit Ethernet switch.
    a. For KR C4 controllers:
    Connect the Ethernet cable to port X66 on the KUKA controller. This interface can support multiple virtual interfaces. For RSI, create a new virtual interface (such as virtual6, virtual7, etc.) and configure it as UDP with the IP address 192.168.100.100 (if using custom settings, enter the IP address for your subnet). Leave the default interface, virtual5, for KLI or other customer network interfaces, such as PROFINET.
    b. For KR C5 controllers:
    Connect the Ethernet cable to the default KPI port XF5. If an additional port, XF6, has been purchased, it can also be used for the Ethernet connection.
3. Connect the GigE camera or cameras to the Gigabit Ethernet switch.
4. Connect the force/torque sensor box to the Gigabit Ethernet switch.
5. Connect the Gigabit Ethernet switch to the MIRAI controller using the following ports:
    a. *For one GigE camera*: Connect Ethernet port X1P1 on the MIRAI controller with any port on the Gigabit Ethernet switch.
    b. *For two GigE cameras*: Connect link-aggregated ports X3P1 and X3P4 on the MIRAI controller with two link-aggregated ports on the Gigabit Ethernet switch.
6. Connect port X2P1 on the MIRAI controller to a network with WAN/internet access. Ensure that the connected network offers DHCP service, as the IP address for this port will be assigned automatically through DHCP.
    a. NOTE: To use a proxy to connect with the Micropsi cloud, see Connecting to the Micropsi Cloud Using a Proxy Server. For information on the cloud connection and handling of recorded data, see Data FAQ in MIRAI Training User Manual.
7. Connect the Wi-Fi module to a USB port on the MIRAI controller to link it with the Android tablet.
8. Connect a 24 V power supply to the MIRAI controller to turn the controller on. Check the LED RUN/STOP indicator for a green light to confirm successful system boot and started runtime.

# 3 Installing and Connecting MIRAI Training App

The MIRAI Training App is the primary user interface for the MIRAI solution. It is a mobile application for Android-based tablets that allows users to do the following:

- Create and define new MIRAI skills
- Record training episodes to train new or improve available skills
- Create new skill versions using recorded episodes
- Test newly calculated skills to ensure they perform as expected or continue adding training episodes to further improve the behavior of an available skill
- Set, test, and revise end state parameters that allow MIRAI to trigger a successful skill execution
- View and track execution KPIs of skills while they operate in production mode (in development)
- Accelerate and tune skill execution speed to achieve optimal tact time without compromising on precision
- Manage and delete trained skills

## 3.1 Installing the Latest MIRAI Training App to the Tablet

1. Ensure that the MIRAI controller is set up, powered on and that the Wi-Fi module is connected.
2. Power on your Android tablet and switch on its Wi-Fi network. Go to the Wi-Fi settings of the tablet and choose the network SSID from the MIRAI controller, which should be named "MIRAI-*<ID_number>*." The ID number of the MIRAI controller, which is indicated on the product sticker. Enter the MIRAI password, also printed on the same sticker.
3. Within the Android settings of your device, navigate to **Settings** > **Security**, and activate the option that reads "Allow install of apps from unknown sources" or has similar wording. Depending on your Android version, this menu item might not be present, and instead the system will ask for permission to install the APK file once trying to open it. In this case, grant the request.
4. Start the internet browser of the tablet and go to http://mirai:6543/mint/apk.
5. Click to confirm downloading the MIRAI Training App installation file, mint.apk. After downloading, tap on the file in the Android file browser to install the app. If a security message requests permission to install the APK file, grant the request.
6. Start the MIRAI Training App. It should show the "Skill training" screen with an overview of the trained skills available on the MIRAI controller. When you use the app for the first time, this list will be empty.

# 4 Network Configuration

## 4.1 Networks in the MIRAI setup

There are three networks in the standard MIRAI setup: the robot network, the WAN, and the Wi-Fi network. The MIRAI controller uses different ports for each network, ensuring they remain separated and cannot communicate with each other.

- **Robot Network**: This network includes the MIRAI controller, the robot controller, a force/torque sensor (if used), and the GigE camera or cameras (if used). Each device in this network has a static IP address. The devices are connected through specific ports on the controller, ensuring they operate within an isolated network that cannot communicate with the other networks.
- **Wide Area Network (WAN)**: This network also includes the controller, but it is connected through a different port preconfigured to expect a DHCP-assigned IP address. The WAN typically includes an HTTP/HTTPS proxy that facilitates communication between the controller and the Micropsi cloud, making it reachable via the internet. This network is logically separated from the robot network.
- **Wi-Fi Network**: This network is created by the controller and includes a single member, an Android-based tablet. The Wi-Fi network is isolated from both the robot network and the WAN, ensuring that the tablet can only communicate with the controller through this dedicated wireless connection.

The following sections explain how to set up the **robot network** on the KUKA smartPAD (see Section 4.3) and in the MIRAI Training App (see Section 4.4).

The MIRAI controller has four network ports. The ports have the following settings:

| Port number | Controller name | MIRAI controller | Subnet |
|---|---|---|---|
| Port 1 | X1P1 | 192.168.100.5 | 192.168.100.0/24 |
| Port 2 | X2P1 | set via DHCP | set via DHCP |
| Port 3 | X3P1 | 192.168.99.5 | 192.168.99.0/24 |
| Port 4 | X4P1 | 192.168.99.5 | 192.168.99.0/24 |

Different ports and network configurations are recommended for different camera setups. Choose one of the following configuration options depending on your camera setup:

- **USB 3.0 cameras**: Refer to section 4.4.1.
- **One GigE camera**: Use Port 1 or custom settings (see section 4.4.2)
- **Two GigE cameras**: Use Ports 3 and 4 (see section 4.4.3)

For **custom network settings**, users can manually configure the ports as desired. Refer to section 4.4.4.

## 4.2 Network Configuration on the KUKA smartPAD

NOTE: These steps show the settings for the default configuration for MIRAI controller Port#1 (X1P1) on the 192.168.100.0/24 subnet. The MIRAI controller is assigned to the IP address 192.168.100.5. If you are using a different subnet, the fields you need to adjust will be highlighted in boxes below.

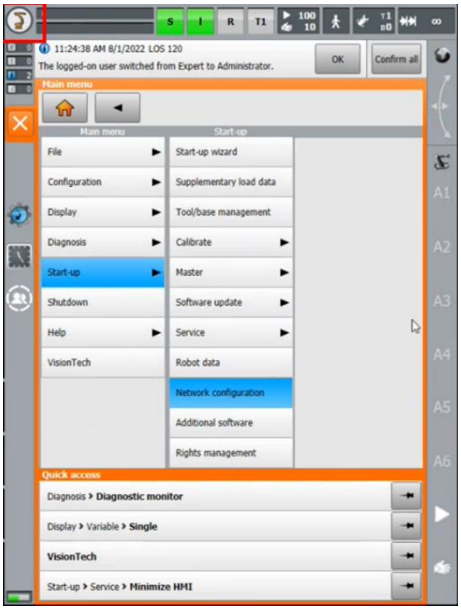
All components in the MIRAI network must be configured to the same subnet.

To enter the robot's IP address on the KUKA smartPad, follow these steps:

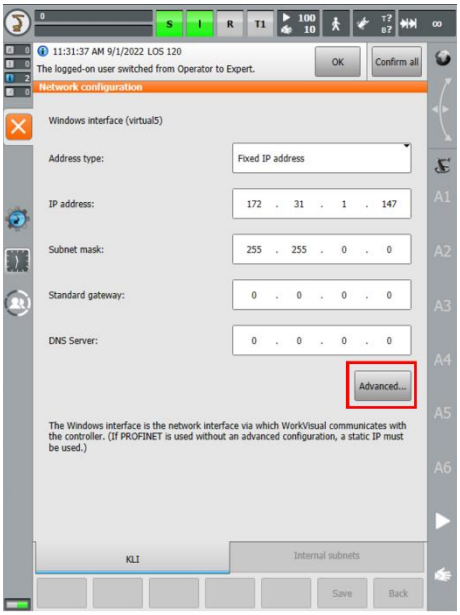
**IMPORTANT:** Make sure you are logged in as an **"Expert"** user group.

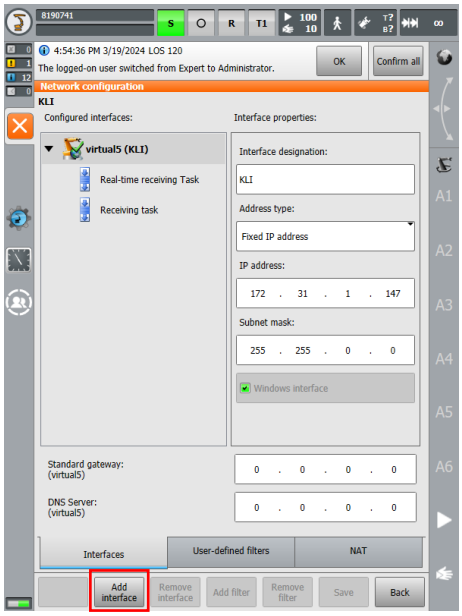
**Step 1**. Tap the top left **main**  icon.

**Step 2.** Tap **Startup** and then **Network Configuration.**




**Step 3**. Select **Advanced** and then select **Add interface** and name it **RSI**.



Step 4. Tap **virtual (RSI).**

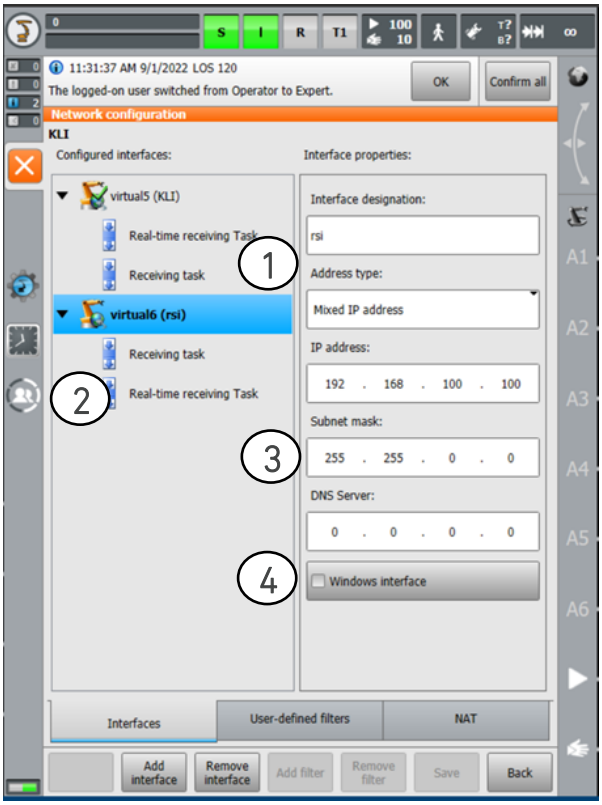
1. In the Interface properties section on the right:
    - Interface designation: enter **RSI.**
    - Address type: Select **Mixed IP address.**

2. Tap on **Real-time receiving Task** and Select **UDP.**

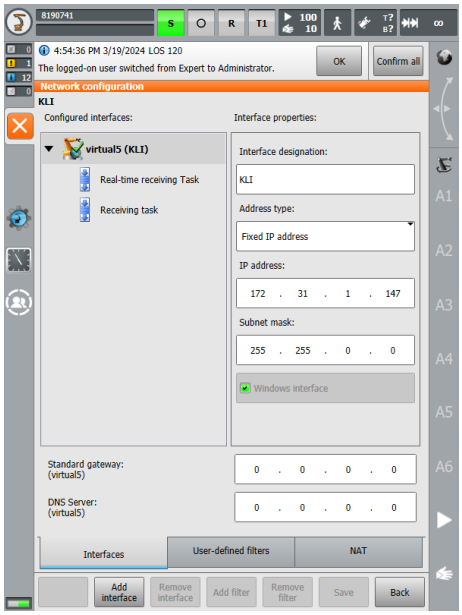
3. **For the standard configuration**, enter the following IP addresses:
    - IP address:**192.168.100.100**
    - Subnet mask: **255.255.0.0**
    - DNS Server: **0.0.0**.0

    **For custom settings**, enter the appropriate subnet.

4. Ensure that the checkbox for the **Windows interface** is not selected.



**NOTE:** The Virtual5 interface on KUKA controllers should be kept for the KUKA Line Interface (KLI) or other customer network interfaces like PROFINET. Both the KR C4 and KR C5 KUKA controllers use the same KLI interface with the IP address 172.31.1.147.




**Step 5**. Configure the force/torque sensor.

- **For the 192.168.100.0/24 subnet** (Port X1P1 on the MIRAI controller)
  - For the OnRobot F/T sensor, use 192.168.100.15. For more information, see Section 7.1.
  - For the ATI F/T sensor, use 192.168.100.20. For more information, see Section 7.2.

- **For the 192.168.99.0/24 subnet** (Ports X3P1 and X4P1 on the MIRAI controller)
  - For the OnRobot F/T sensor, use 192.168.99.15. For more information, see Section 7.1.
  - For the ATI F/T sensor, use 192.168.99.20. For more information, see Section 7.2.

**NOTE**: After changing the IP settings in the KUKA smartPAD the robot controller requires a reboot.

## 4.3 Network Configuration in KUKA.WorkVisual

You must update the IP address in KUKA.WorkVisual if you are using the192.168.99.0/24 subnet or have customized the MIRAI controller IP address to connect to an existing subnet.

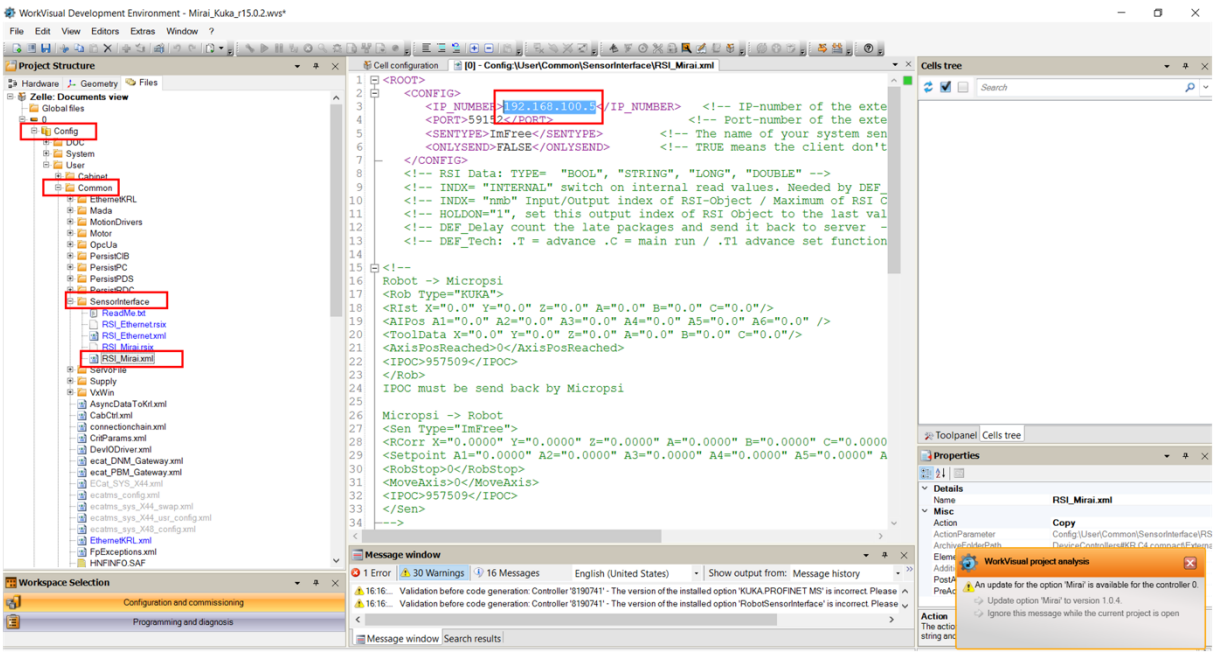
Update the IP address in the following places:

- **KUKA.RobotSensorInterface**: refer to Section 4.3.1
- **KUKA.Ethernet KRL**: refer to Section 4.3.2

If you are using the 192.168.100.0/24 subnet, no changes are necessary in KUKA.WorkVisual. The MIRAI KOP file sets the 192.168.100.0/24 subnet as the default setting.

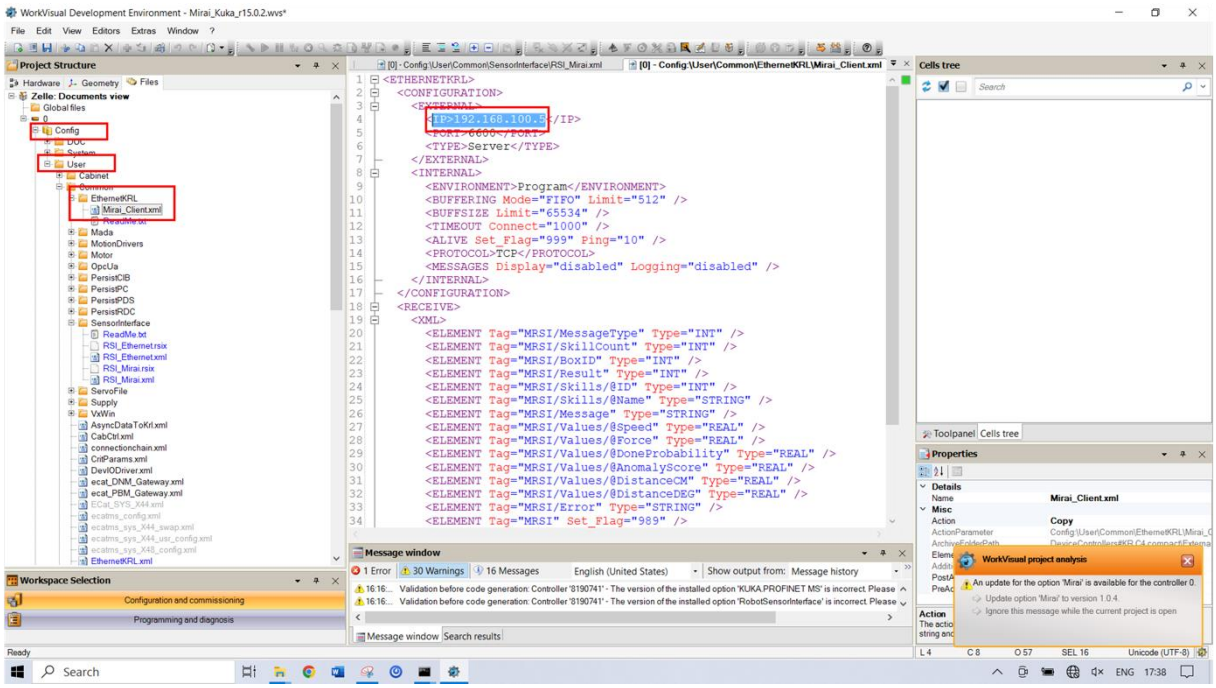
### 4.3.1 Update the IP address in KUKA.RobotSensor Interface

- Open WorkVisual and navigate to **Config** > **User** > **Common** > **SensorInterface**.
- Update the XML file `RSI_Mirai.xml` with the relevant IP address for the MIRAI controller.




### 4.3.2 Update the IP address in KUKA.Ethernet KRL

- Open WorkVisual and navigate to **Config** > **User** > **Common** > **EthernetKRL**.
- Update the XML file `Mirai_Client.xml` with the relevant IP address for the MIRAI controller.



## 4.4 Network Configuration in the MIRAI Training App

The MIRAI Training App includes a Network Configuration screen. This section guides you through setting up the network for your camera system.

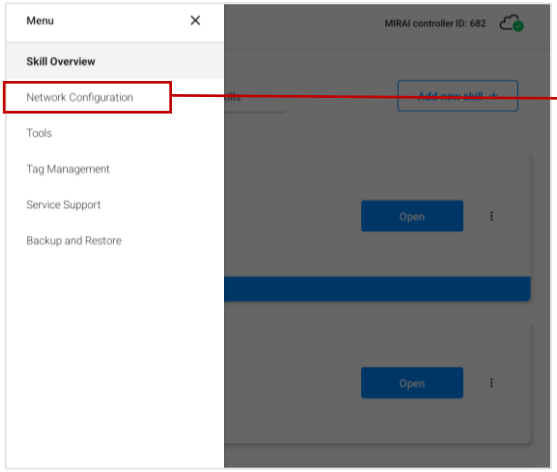
- **USB 3.0 cameras**: Refer to section 4.4.1.
- **One GigE camera**: Use Port 1 or custom settings (see section 4.4.2)
- **Two GigE cameras**: Use Ports 3 and 4 (see section 4.4.3)

For **custom network settings**, users can manually configure the ports as desired. Refer to section 4.4.4.

### 4.4.1 USB 3.0 Cameras

All components in the MIRAI network must be configured to the same subnet. USB 3.0 cameras are not in the MIRAI network and do not require configuration.

To configure the other components in the MIRAI network, use the default settings or enter custom settings:

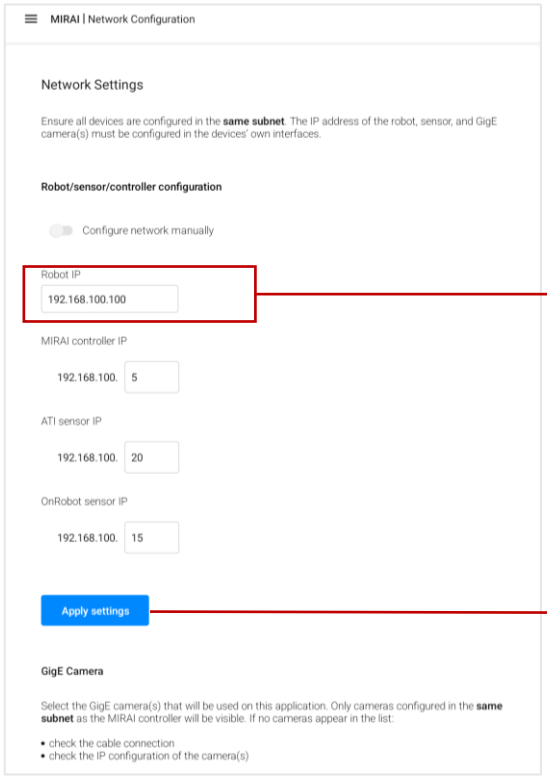



**Step 1.** Enter the main menu (top left corner) and go to **Network Configuration**.

**Default settings**

The app settings for Port 1 (X1P1) are preconfigured to the **192.168.100.0/24 subnet**, as shown below left.

To use these settings, **no changes are necessary in the app**. Ensure that the robot and force/torque sensor are configured correctly on their respective interfaces.




**Custom settings**

**Step 2.** To configure Port 1 (X1P1), enter the IP address of the robot that you are using. The first three IP digit fields will be automatically updated for the MIRAI controller and the relevant force/torque sensor. Assign the remaining fields accordingly.

**Update the IP address in WorkVisual:**

- KUKA.RobotSensorInterface (Section 4.3.1)
- KUKA.Ethernet KRL (Section 4.3.2)

**Step 3.** Tap **Apply Settings**.

An error message will appear for the sensor model you are not using.

- Tap **OK.**

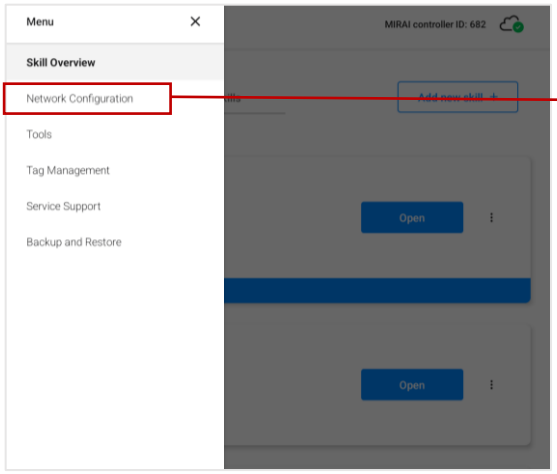
A message to confirm settings will appear.

- Tap **Yes, apply**.

### 4.4.2 One GigE Camera: Using Port 1 (X1P1)

Port 1 is preconfigured to the 192.168.100.0/24 subnet with the MIRAI controller assigned to the IP address 192.168.100.5. If you plan to switch between using one and two GigE cameras, use the 192.168.99.0/24 subnet (see 4.4.3).

To configure the components in the MIRAI network, use the default settings or choose custom settings:



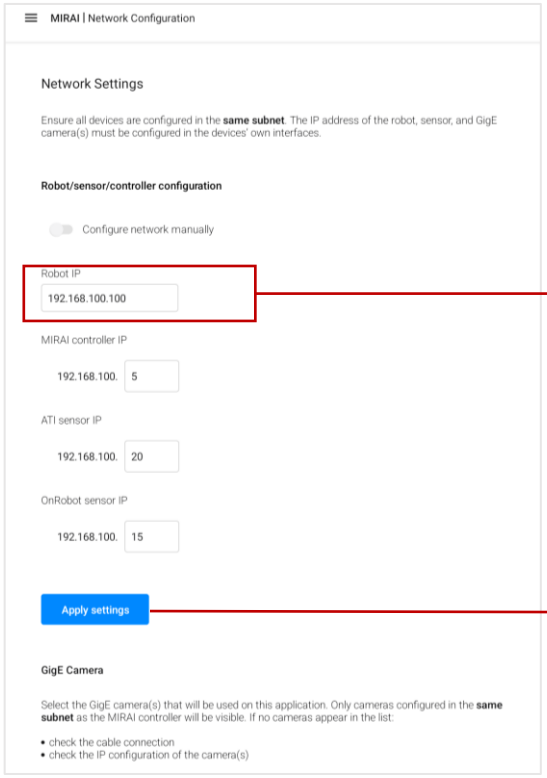

**Step 1.** Enter the main menu (top left corner) and go to **Network Configuration**.

---

Default settings

The app settings for Port 1 (X1P1) are preconfigured to the **192.168.100.0/24 subnet**, as shown below left.

To use these settings, go to Step 5. Ensure that the robot and force/torque sensor are configured correctly on their respective interfaces.




---

Custom settings

**Step 2.** To configure Port 1 (X1P1), enter the IP address of the robot that you are using. The first three IP digit fields will be automatically updated for the MIRAI controller and the relevant force/torque sensor. Assign the remaining fields accordingly.

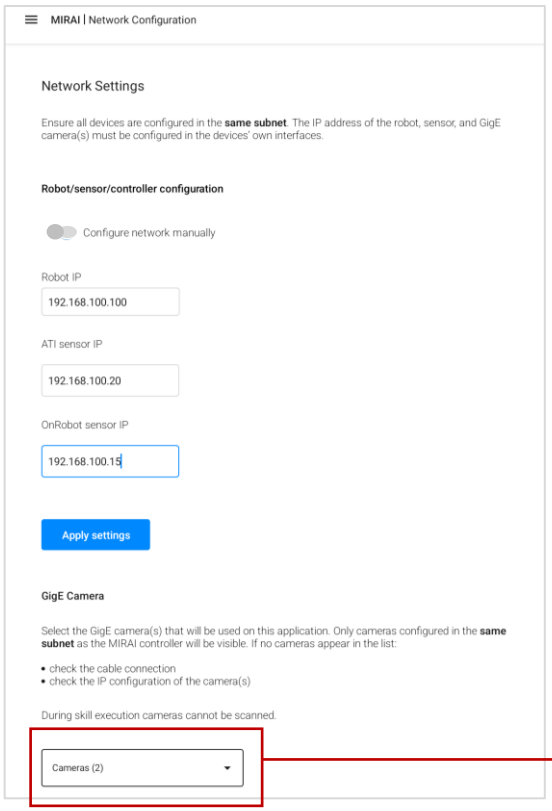
**Step 3.** Tap **Apply Settings**.

An error message will appear for the sensor model you are not using.

- Tap **OK.**

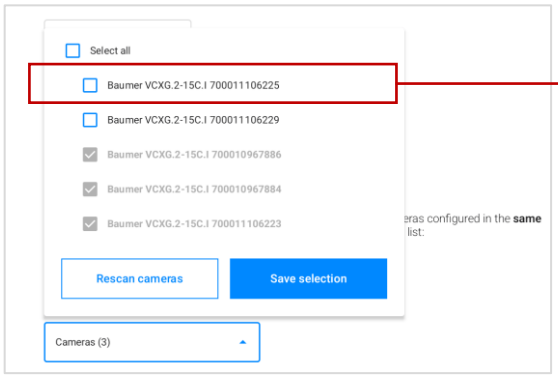
A message to confirm settings will appear.

- Tap **Yes, apply**.

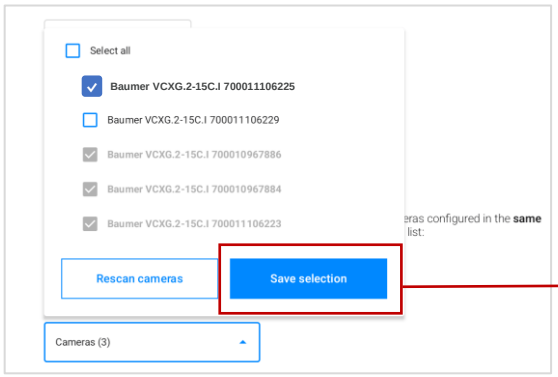



**Step 5.** Ensure that the GigE camera is connected to the Gigabit Ethernet switch using the correct port, and that the IP address is configured to the right subnet. The camera needs power to be visible in the network.

**Step 6.** Go to the **Cameras** drop-down menu.




**Step 7.** Select the configured camera from the **Cameras** drop-down menu.




**Step 8.** Tap **Save selection**. The configured IP address will appear next to the selected camera.

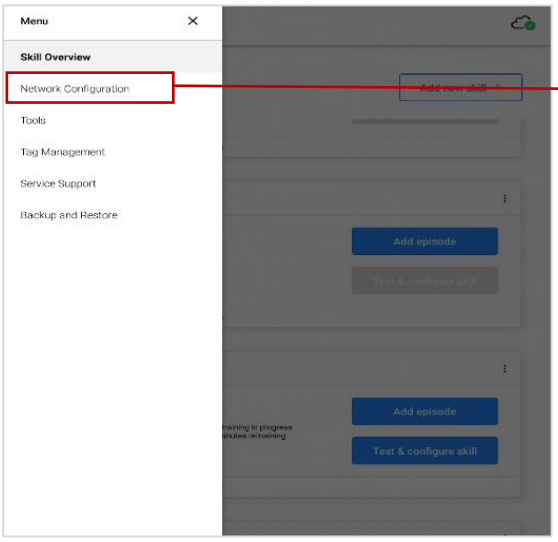
### 4.4.3 Two GigE cameras: Using Preconfigured Ports 3 and 4

Ports 3 and 4 are link aggregated, functioning as a single logical link to provide increased bandwidth. These ports are preconfigured to the 192.168.99.0/24 subnet, with the MIRAI controller assigned to IP address 192.168.99.5. Note that you can also use a single camera on this network.

**You must update the IP address in KUKA.WorkVisual** in the following places:

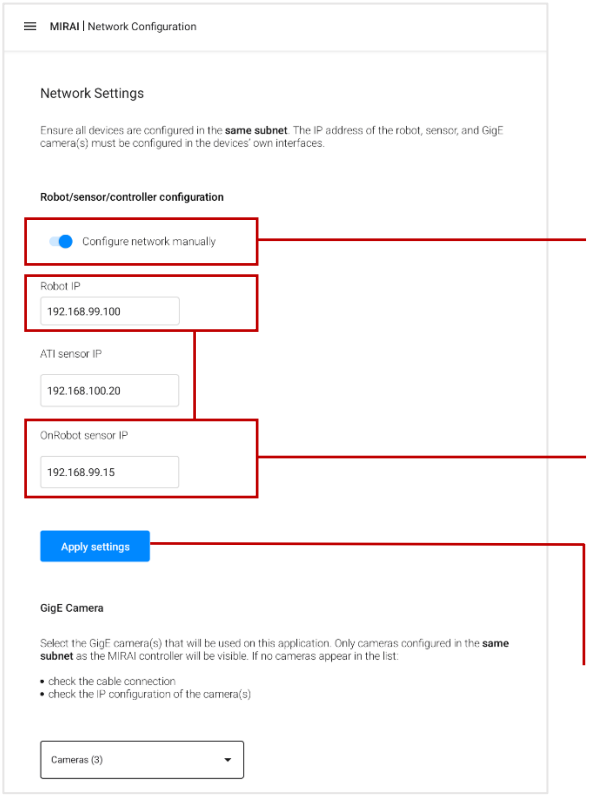
- KUKA.RobotSensorInterface: refer to Section 4.3.1
- KUKA.Ethernet KRL: refer to Section 4.3.2

Then follow these steps to configure the network settings in the MIRAI Training App:




**Step 1.** Enter the main menu (top left corner) and go to **Network Configuration**.

**NOTE:** All devices in the MIRAI network must be configured to the 192.168.99.0/24 subnet. Configure the robot, force/torque sensor, and GigE cameras using their respective interfaces




**Step 2.** Set the toggle button to **Configure network manually** and tap **Continue**.

**Step 3.** Enter the IP address for the robot and for the force/torque sensor you are using to train the skill using the 192.168.99.0/24 subnet

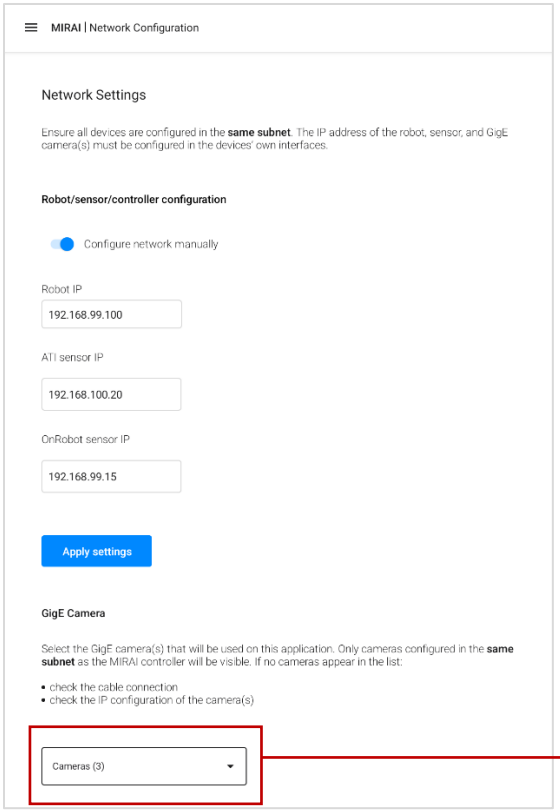
**Step 4.** Tap **Apply Settings**.

An error message will appear for the sensor model you are not using.

- Tap **OK.**

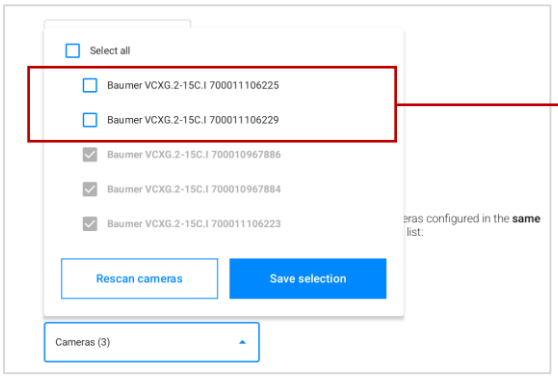
A message to confirm settings will appear.

- Tap **Yes, apply**.

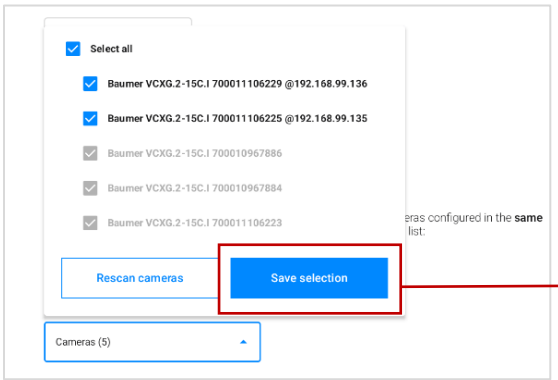



**Step 5.** Ensure that the GigE cameras are connected to the Gigabit Ethernet switch using the correct ports, and that the IP address is configured to the right subnet. The cameras need power to be visible in the network.

**Step 6.** Go to the **Cameras** drop-down menu.




**Step 7.** Select the configured cameras from the **Cameras** drop-down menu.




**Step 8.** Tap **Save selection**. The configured IP addresses will appear next to the selected cameras.

### 4.4.4 Manual Network Configuration

> ⚠ CAUTION
>
> Manual configuration requires Linux command-line skills and a solid understanding of networking principles. Exercise caution, because incorrect configurations can cause network connectivity issues. Users are fully responsible for any changes made to network settings using this method.

This method provides full control and customization of network settings beyond standard configurations.

Set the toggle to **Configure network manually**.

To configure the network setup, access the controller via SSH or a physical terminal, and then use Linux command-line tools.

The username is netadmin. The password is printed on the controller. It is the same password used for the Wi-Fi connection.

Log in to edit **/etc/network/interfaces** and configure each Ethernet interface.

The netadmin user has write access to **/etc/network/interfaces** and to the following sudo commands:

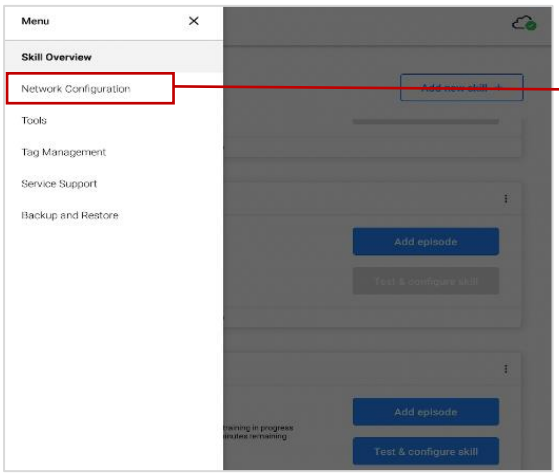
- `sudo ip`
- `sudo ifup`
- `sudo ifdown`
- `sudo reboot`
- `sudo poweroff`
- `sudo systemctl restart micropsi-runtime`
- `gevipconfig`

If you misconfigure the network of the MIRAI controller, you can restore the controller's initial state using the following command:
`mirai-restore-network`

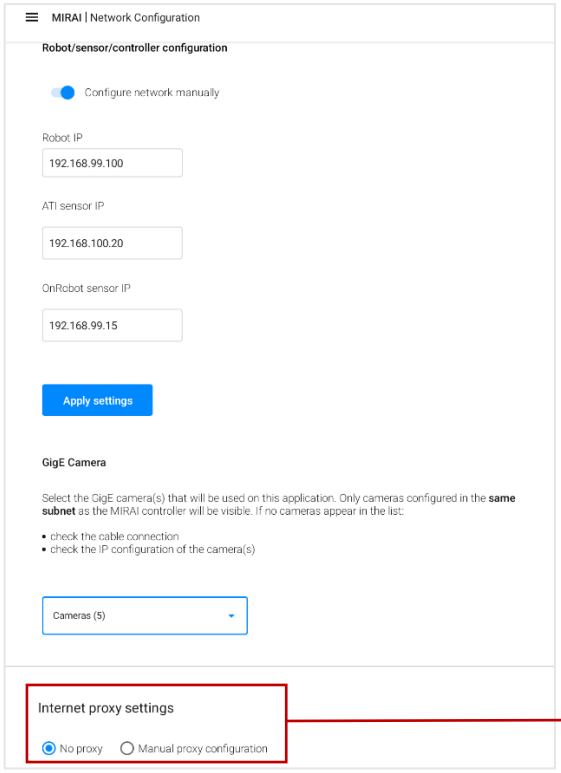
> ⚠ IMPORTANT:
> The MIRAI controller requires a **restart to apply network changes** made by netadmin.

### 4.4.5 Connecting to the Micropsi Cloud Using a Proxy Server

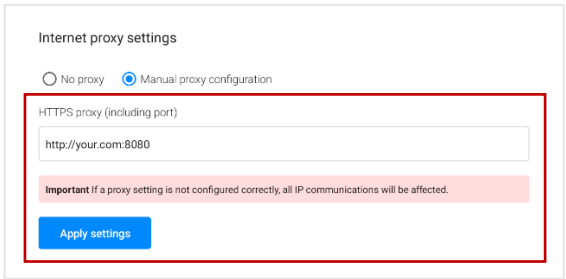
If IT security requires the use of a proxy server to connect the MIRAI controller to the Micropsi cloud, follow these steps:




**Step 1.** Enter the main menu (top left corner) and go to **Network Configuration**.




**Step 2.** Under **Internet proxy settings**, select **Manual proxy configuration.**




**Step 3.** Enter the address and the port in the indicated format. Tap **Apply settings**

---

The proxy must allow connections to these sites:

**For training:**
https://crunch.micropsi-industries.com **(TCP Port 443)**
https://crunch.micropsi.io **(TCP Port 443)**

**For support :**
mirai-vpn.micropsi-industries.com **(UDP Port 1194)**

**For software updates :**
https://apt.tools.micropsi.io **(TCP Port 443)**
http://deb.debian.org **(TCP Port 80)**
http://security.debian.org **(TCP Port 80)**

# 5 Install MIRAI Software on the KUKA SmartPAD

A KOP file (KUKA Option Package file) is required to integrate and manage MIRAI capabilities on the KUKA controller. KOP files. Two MIRAI KOP files available:

1. **Mirai.kop**: Contains configurations and data to activate MIRAI capabilities
2. **MiraiConveyor.kop**. Identical to Mirai.kop but includes additional conveyor functionality

You will receive the appropriate KOP file based on your need for conveyor functionality. Note that both KOP files cannot be installed simultaneously on the KUKA controller.

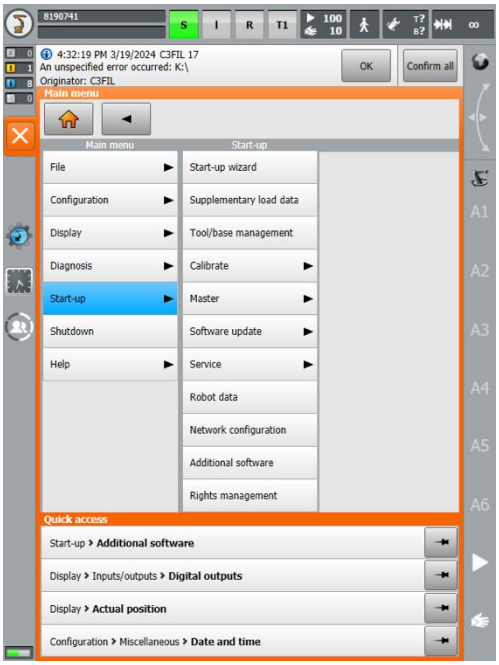
| |
|---|
| **IMPORTANT:** The ConveyorTech Package must be installed **before** installing MiraiConveyor.kop. |

## 5.1 Install the MIRAI KOP file

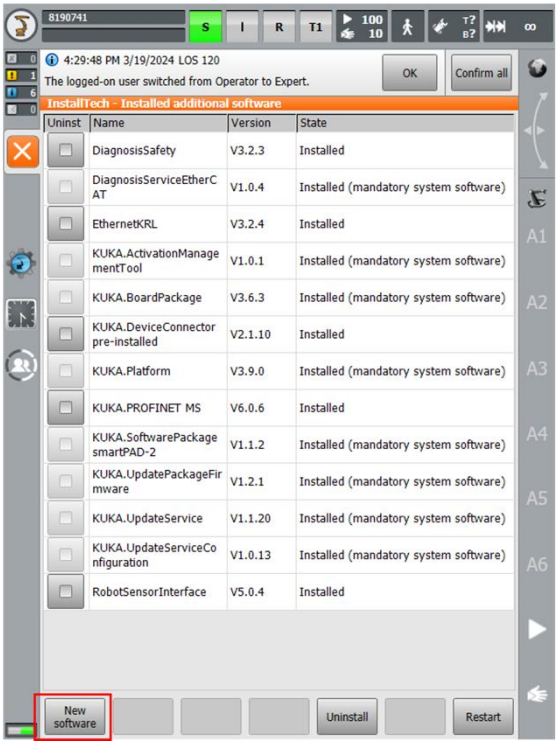
**Step 1**. Insert the USB drive in the KUKA smartPAD USB port.

**Step 2**. Tap  **Menu**, in the top left corner of the smartPAD.

**Step 3**. Tap **Start-up** and then **Additional software.**



Step 4. Tap **New software**, on the bottom bar.

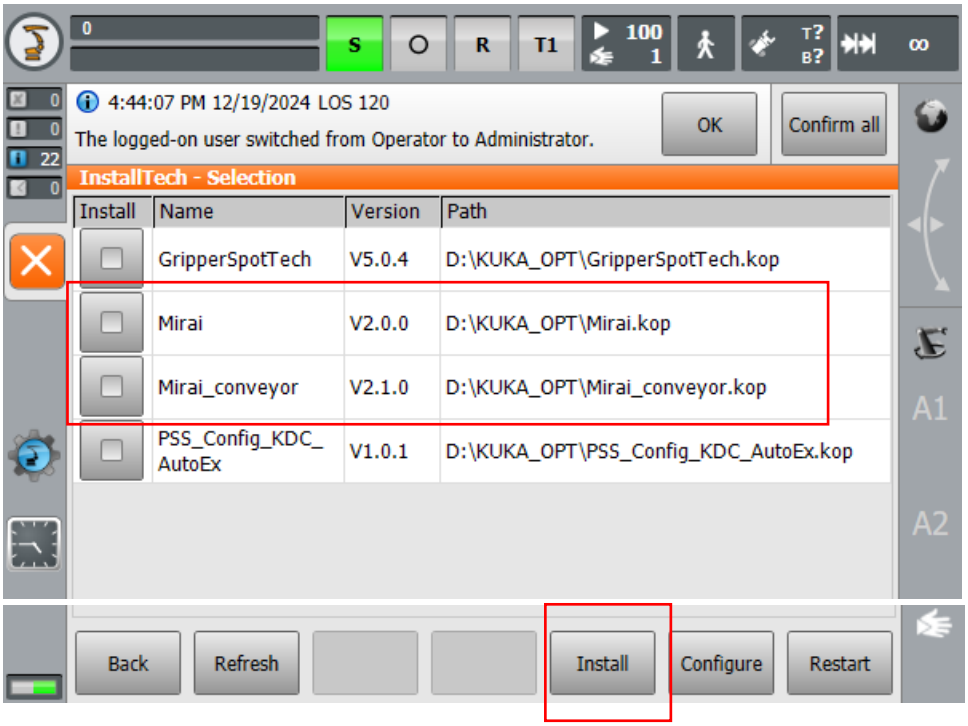



Step 5. Tap **Configure**. Select the installation path and tap **Save** on the bottom bar.

Step 6. Navigate to your USB drive and select the MIRAI KOP file:

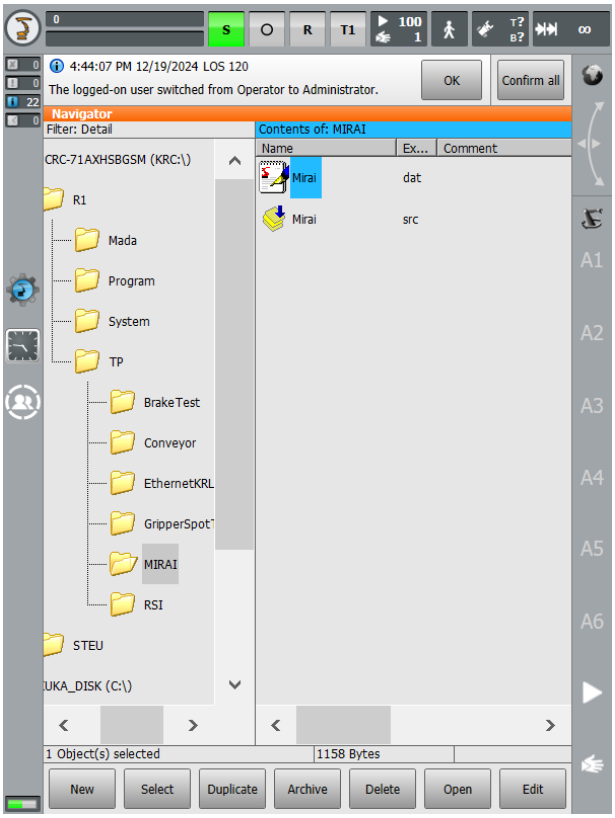
- **Mirai.kop** if you do not need the conveyor function.
- **MiraiConveyor.kop** if you need the conveyor function

Step 7. Tap **Install**, in the bottom-right corner of the smartPAD.



## 5.2 Check communication status

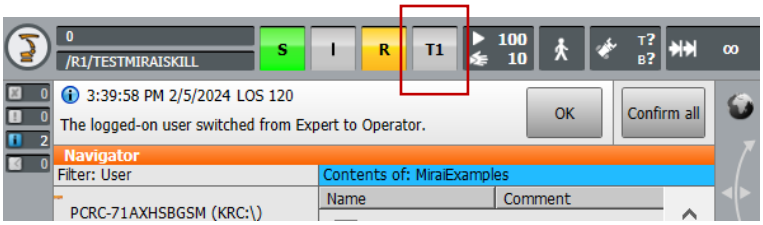
Once the MIRAI KOP file has been successfully executed on the KUKA smartPAD the communication status between MIRAI and the robot can be verified by making sure the MIRAI folder exists with the relevant programs: R1 > TP > MIRAI.




## 5.3 Operating mode for MIRAI training: T1

When training skills using the MIRAI Training App, the use of T1 mode is strongly recommended to ensure user safety during hand-guiding.

To change the operating mode, use the mode selector switch on the KUKA smartPAD, shown below. For older models, turn the black knob at the top of the teach pendant to the right to access the operation mode menu. For more detailed information, refer to the KUKA operator's manual for your robot model.



# 6 Using MIRAI skills with KUKA programs

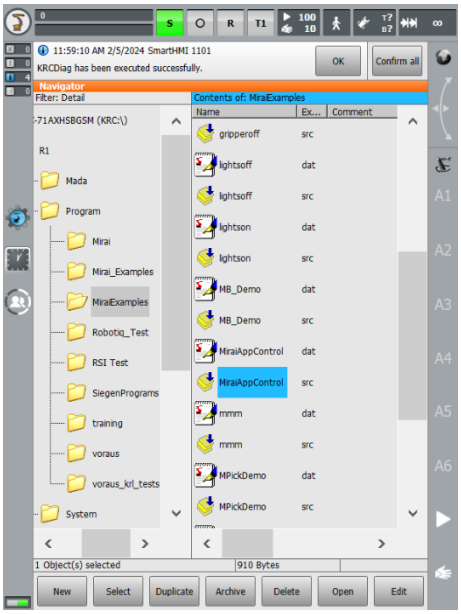
## 6.1 Use the MIRAI Training App with the KUKA smartPAD

Certain features available in the MIRAI Training App will require you to press both the enabling switch and the start key. These features include:

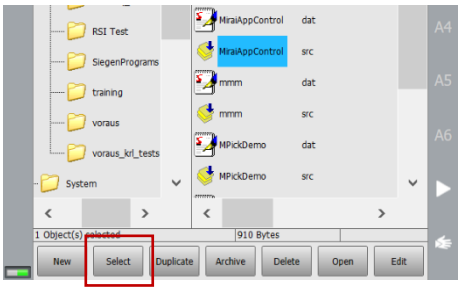
- Creating a new MIRAI skill
- Recording episodes
- Testing and executing a skill
- Certain skill-specific features such as “Reset reference position” and “Camera realignment.”

To use the MIRAI Training App, follow the steps:

**Step 1.** Navigate to the **MiraiExamples** folder and tap the **MiraiAppControl.src** program.




**Step 2**. Tap **Select** on the bottom bar.




The **MiraiAppControl** program will be displayed on the smartPAD. The status keys should show the following colors:

- **S (Submit Interpreter):** Green
- **I/O (Drives Power):** Gray
- **R (Robot Interpreter)**: Yellow

Step 3. Press both the enabling switch and the start key (green button on the KUKA smartPAD). The status keys will now show the following colors:

- S (Submit Interpreter): Green
- I/O (Drives Power): Green
- R (Robot Interpreter): Red

NOTE: When you start the program, you will need to achieve Block Coincidence (BCO). BCO ensures that the robot's current position matches the coordinates of the initial position in the robot program.

Step 4. While pressing the enabling switch, release the start key (green button on the KUKA smartPAD) and then press it again to confirm that the current robot position matches the initial position in the robot program. The status keys will now all be green:

- S (Submit Interpreter): Green
- I/O (Drives Power): Green
- R (Robot Interpreter): Green

Step 5. You can now proceed to use the MIRAI training App, while keeping both the enabling switch and the start key pressed.

| NOTE: Program looping on teach pendant |
|---|
| After all status keys are green, the program on the teach pendant will continue to loop. Looping will stop when MIRAI sends a control signal to the robot via the MIRAI Training App. Control signals are sent from the Skill overview and Skill details screens as follows:<br>Skill overview screen:<br>• Go to Create new Skill > Step 1 [skill type]. Tap Next. The looping will stop when the next screen is displayed, Step 2 [Skill Configuration].<br>Skill details screen:<br>• Looping will stop when you access the following screens:<br>o Go to Record episodes or Test & Configure.<br>o Expand the three-dot menu and go to Camera Setting or Set reference position. |

| NOTE: Changing training position without a force/torque sensor |
|---|
| If you change the robot position during training without a force/torque sensor using the KUKA SmartPad, you need to reset the KUKA program (MiraiAppControl). Ensure the S, I/O, and R indicators are all green, then tap Try again in the MIRAI Training App. |

| Error Message: Stand still: current position and start position STAND_STILL not identical. | |
|---|---|
| Cause: | This message appears if you accidentally release the enabling switch and/or start key. |
| Solution: | 1. Reset the Program by press the R button.<br>2. Repeat Steps 3 and 4. |

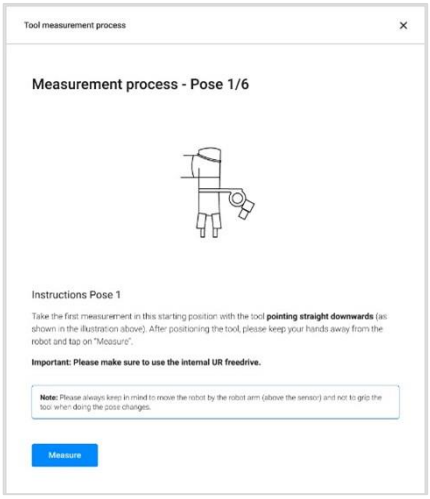
## 6.2 Mass and Center of Gravity Measurement in the MIRAI Training App

On the KUKA smartPAD, open or create the program you would like to add your MIRAI skill to.

**Step 1**. Run **MiraiAppControl** on KUKA smartPAD, with the enabling switch and start key pressed. The status keys will now all be green:

- **S (Submit Interpreter):** Green
- **I/O (Drives Power):** Green
- **Robot Interpreter**): Green

**Step 2**. In the MIRAI Training App, create a new tool to reach the **Measurement process – Pose 1/6** screen. screen. For details, refer to Section 3.2.1 Tool Configuration in the MIRAI Training User Manual.




**Step 3**. Release both the enabling switch and the start key. Do not cancel or reset MiraiAppControl until the CoG measurement process is complete.

**Step 4**. Press and hold only the enabling switch. The status keys will now show the following colors:

- **S (Submit Interpreter):** Green
- **I/O (Drives Power):** Green
- **R (Robot Interpreter)**: Red

**Step 5**. Jog the robot to the position shown in the app for Pose 1/6.

**Step 6**. In the MIRAI Training App, tap **Measure**.

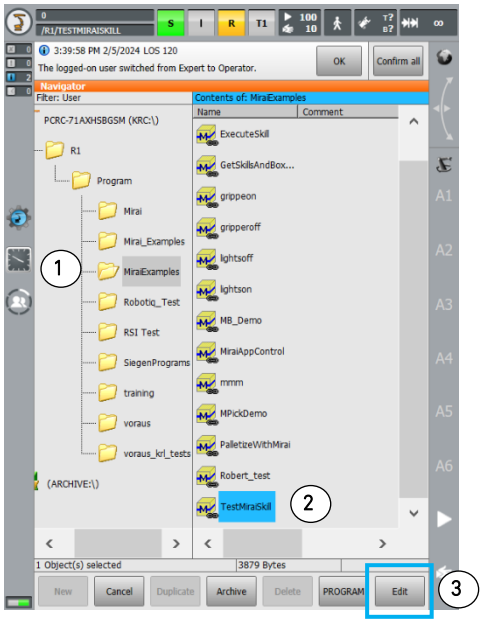
**Step 7**. Repeat steps 5 and 6 until you get all the measurements.

## 6.3 Adding a MIRAI Skill to a KUKA program

On the KUKA smartPAD, open or create the program you would like to add your MIRAI skill to. A template – `TestMiraiSkill` – is provided on the USB drive.

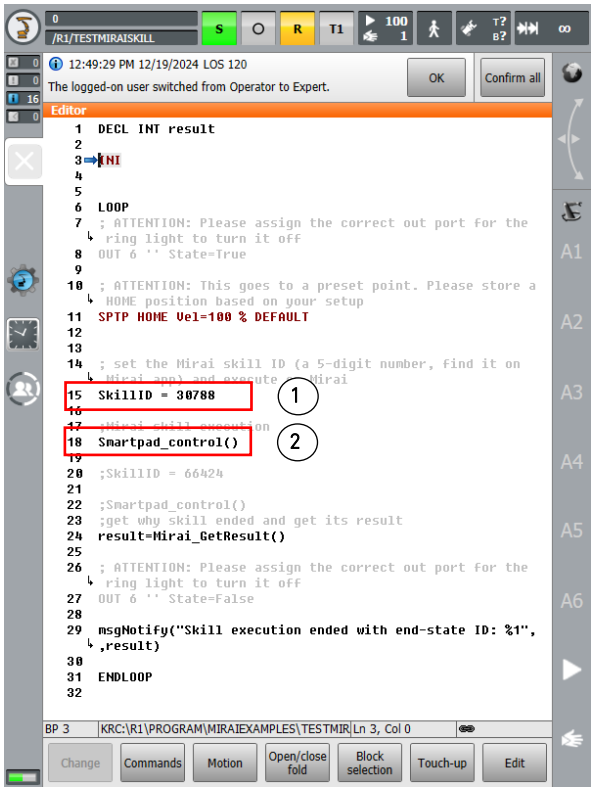
**Follow these steps to open the MIRAI template:**

1. In the directory structure, select the `MiraiExample` folder.
2. In the contents of the folder, select the `TestMiraiSkill.src` file.
3. Tap the **Edit** softkey and select **Open** from the menu.



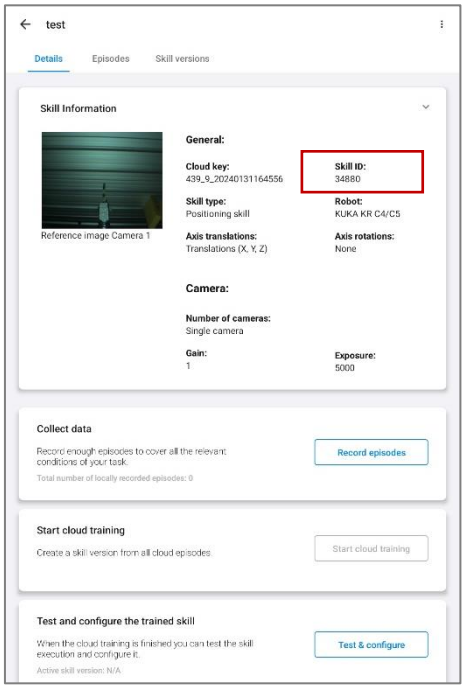

**Copy and paste two lines in the MIRAI template into your main program:**

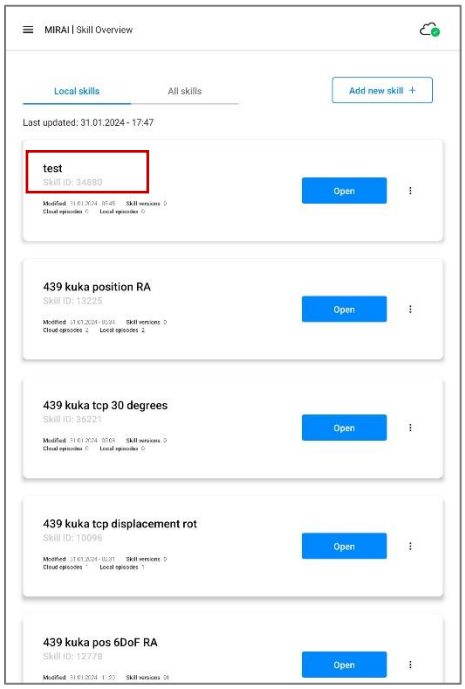
1. `SkillID =` defines the skill to be executed using a 5-digit number in the MIRAI Training App.
2. `Smartpad_control()` is the function that allows MIRAI to execute the skill.



The Skill ID for a MIRAI Skill is displayed in the MIRAI Training App in the following places:

- On the skill card, in the main Skill Overview section of the app.
- In the Skill Information section for each skill.




## 6.4 Adding a MIRAI Conveyor Skill to a KUKA Program

KUKA robots can execute MIRAI skills on conveyors. The robot's actions are adjusted to follow the movement of the conveyor, allowing MIRAI to perform tasks while the conveyor is running. This section explains how to execute MIRAI skills during conveyor tracking.

### 6.4.1 User Requirements

MIRAI conveyor capabilities are integrated on KUKA controllers using the KUKA.ConveyorTech package. Users are responsible for using this package and configuring the conveyor.

**IMPORTANT:** Install the KUKA.ConveyorTech package **before** installing MiraiConveyor.kop. Failure to do so will throw multiple errors and the program will not run.

**User Responsibilities**

- KUKA.ConveyorTech package:
  - Users must have full command over the KUKA.ConveyorTech package. This includes configuring all settings as well as implementing synchronized and asynchronized robot motions on the conveyor.
- Conveyor Base Configuration:
  - Users are responsible for configuring the conveyor base, conveyor operation area, and all necessary integration components.
  - Users define their own conveyor motions, including speed and direction.

**Requirements:**

- **7th axis connection:** The conveyor must be connected as a 7th axis.
- **Speed limitation of the conveyor:** 200 mm/second.

### 6.4.2 Training MIRAI Skills for Conveyor Tracking

In general, you can train skills to be executed on the conveyor just like any other skills. There are two particularities:

1. **Position-based end states cannot be used in conveyor mode** because the robot operates in the conveyor frame, while the coordinates are saved in the robot base frame, leading to a mismatch.
2. **The conveyor must be stationary during skill training.** It is important to train the skills in the same environment that will be present during skill execution. This includes factors such as the appearance of the conveyor belt, lighting conditions, and other relevant elements.

For detailed instructions on training MIRAI skills, refer to the User Manual for MIRAI Training.

**IMPORTANT:** To ensure user safety, MIRAI skills must be trained in T1 mode, even if training is possible in other modes.

### 6.4.3 MIRAI Program to Execute Skills during Conveyor Tracking

A sample conveyor tracking program, `Mirai_conv.src`, is provided on the USB drive with instructions written in the comments. You can use this program as a template or copy and paste the required sections of this program into your own program on the KUKA smartPAD. This section explains which parts you need to add to your program.

**Open the MIRAI template:**

1. In the directory structure, select the `MiraiExample` folder.
2. In the contents of the folder, select the `Mirai_conv.src` file.
3. Tap the **Edit** softkey and select **Open** from the menu.

**Create a CONV_MOV group to execute the skill:**

1. **Set conveyor synchronization points:** For details, refer to the KUKA.ConveyorTech documentation.
   - Set the **conveyor distance** from the optical trigger to the point where the robot will start tracking the workpiece.
   - Set the **home point**: Move the robot to the position where it will start its task, for example, above the workpiece where control will be handed over to MIRAI.
   - Set the **return point**: Move the robot to the position it will return to after completing its task and decoupling from tracking the conveyor.

2. **Identify the skill to be executed and call the function to execute the skill:**
   - `SkillID =` Enter the 5-digit number of the skill listed in the MIRAI Training App.
   - `Mirai_Conveyor()` This function executes the MIRAI skill.

3. **End synchronization with the conveyor and switch off RSI:**
   - ```
     $BASE=BASE_C
     LIN_REL {X 0 }
     Mirai_MoveRobot_RSI_Off()
     ```
   - These commands stop the robot from following the movement of the conveyor after skill execution.
   - **IMPORTANT:** You must use this method to end conveyor synchronization. Alternatives will not work.
     ```

This excerpt of the sample conveyor tracking program, `Mirai_conv.src`, shows the sections of the program described above.


## 6.4.4 Executing Multiple MIRAI Skills during Conveyor Tracking

Multiple skills can be run on specific setups. To execute consecutive skills, your program will need to detach from the conveyor to switch off RSI and then reattach to the conveyor to execute the second skill. The sample program, `Mirai_conv.src`, provides one suggested method for reattachment:

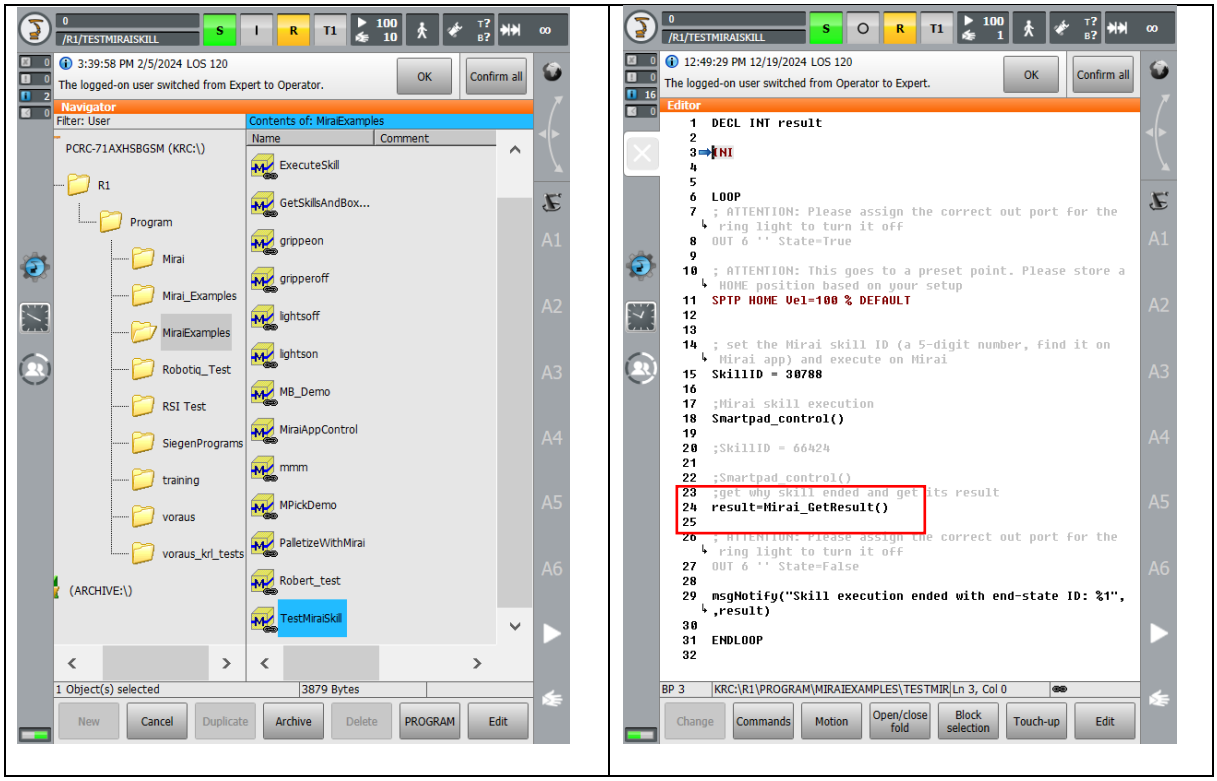
- Connect an output signal (OUT) to the FAST MEASUREMENT system to simulate the synchronization switch.
- Use a second CONV_MOV group to execute a second skill: `CONV_MOV (1,2)`.

Refer to the sample program for instructions.

## 6.5 Returning Values of MIRAI Functions

The return values of the MIRAI functions called during skill execution can be read using the `Mirai_GetResult()` function. This will indicate which end state caused the skill to stop.

Refer to the *User Manual for MIRAI Training* for details on selecting and configuring end states.




The returned value will give information on the result of the skill execution as follows:

- 0 – No result to report. Skill not started or still running
- 1 – Skill execution ended by speed-based endstate
- 2 – Skill execution ended by force-based endstate
- 3 – Skill execution ended by visual endstate
- 4 – Skill execution ended by timeout
- 5 – Skill execution ended by position
- 6 – Skill execution ended by anomaly-based end state
- 7 – Skill execution ended by proximity-based end state
- -1 – an exception occurred during skill execution.

## 6.6 Troubleshooting

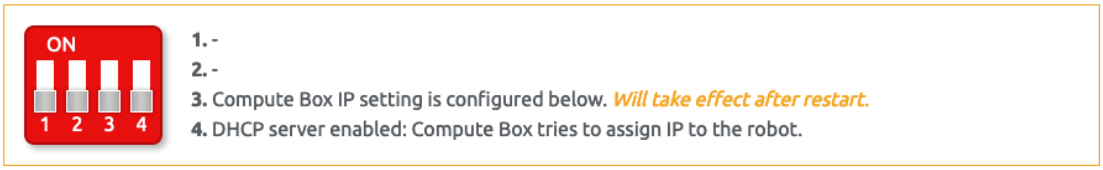
- Communication issues:
  - Recheck the network configuration in the MIRAI Training App and the KUKA smartPAD.
- Program execution issues:
  - If you experience issues with the execution of your program and/or when starting the execution of a new program, we recommend resetting the programs on the KUKA smartPAD.
- Skill execution issues:
  - Refer to "Enabling remote monitoring" in the MIRAI User Guide for the Training App.

# 7 Appendix

## 7.1 Configuring the OnRobot Compute Box

NOTE: The IP addresses given here are for our default configuration for USB cameras or one GigE camera. If you are using two GigE cameras or a manual network configuration, enter the appropriate IP address.

1. Set the DIP Switch 3 to ON to enable static IP.
2. Power on the compute box, and wait 30 seconds.
3. Set your PC's Ethernet network to use DHCP.
4. Connect the Compute Box to your PC via Ethernet. You PC should automatically receive an IP address from the Compute Box in the subnet 192.168.1.0/24.
5. Open your web browser and type: http://192.168.1.1.
6. The administration page will open. Select "Compute Box."
7. Go to the configuration tab.
8. Set the DIP Switch 3 to OFF to disable static IP.
9. Verify that you see the following image:




10. Set the Network Mode to "Static IP."
11. Enter the IP address 192.168.100.15 (or one that matches your network configuration).
12. Select **Save**.
13. Restart the Compute Box by turning it off and on again.

**Steps to verify successful configuration:**

1. Configure your PC's network adapter to a static IP, such as 192.168.100.155, with a Subnet Mask of 255.255.255.0.
2. Disconnect and then reconnect the Ethernet connection to the Compute Box.
3. Open a web browser and enter: http://192.168.100.15 (or the IP address for your subnet)
4. You should see the sensor's configuration page if everything is set up correctly.

## 7.2 Configuring the ATI Sensors

1. Connect the Ethernet cable from the force/torque sensor or Net F/T Box to your computer's ethernet port.
2. Power on the sensor and wait 30 seconds.
3. Open your web browser and navigate to: http://192.168.1.1 to access the ATI Net F/T homepage.
4. Set the IP address to a static IP, such as 192.168.100.20. If you are on a different subnet, enter the appropriate IP address instead.
5. Apply the settings and reboot the sensor.

> IMPORTANT
>
> The ATI force/torque sensors are temperature sensitive, especially the Axia80 series. To reduce output drift, warm up the sensor for one hour before use.

### 7.3 Shut Down the MIRAI controller remotely

For MIRAI controllers with power buttons, use the button to shut down the controller before disconnecting power to ensure the integrity and longevity of the controller. If there is no power button, or if the power button is not accessible due to hardware integration, you can shut down the controller over Ethernet, such as from a PLC.

To shut down the MIRAI controller, use a remote procedure call (RPC) protocol encoded in XML using a cURL -X POST command. An example command for a MIRAI controller with IP address 192.168.100.5 and controller number 6543 would be:

```
curl -X POST http:// 192.168.100.5:6543/skills/xmlrpc -d '<?xml version="1.0"?>
<methodCall><methodName>shutdown_mirai</methodName><params></params>
</methodCall>' -H 'Content-Type:text/xml'
```

# 8 Android Tablets Supporting MIRAI Training Application

To run the MIRAI Training App on Android tablets/mobile devices, follow these recommended minimum configurations:

- **Android 8.1 (Oreo) or higher**: The app may run on older Android versions, but they are not fully tested.
- **Processor**: Quad-core (ARM Cortex-A53 or higher), 1.6GHz. Less powerful processors may result in slower app performance and user interface response.
- **Memory**: 3GB or more recommended.
- **Screen size**: 10" recommended; 8" supported but not extensively used.
- **Screen resolution**: HD (1200 x 800) minimum; FHD (1920 x 1080) recommended.
- **Wi-Fi**: 802.11b/g/n minimum; 802.11a/ac recommended.

If purchasing a new tablet for the MIRAI Training App, consider the following models, which have been extensively tested and used in projects. However, any tablet meeting the above requirements should run the app effectively.

- Lenovo Yoga Tab 3 Plus (10.1" display, 3GB RAM, 32GB eMMC, Wi-Fi, Android 6.0+)
- Lenovo Tab4 10 Plus (10.1" display, 3GB RAM, 16GB eMMC, Wi-Fi, Android 7.0+)
- Lenovo Tab M10 (10.3" display, 4GB RAM, 64GB eMMC, Wi-Fi, Android 9.0+)
- Samsung Galaxy Tab S2 (9.7" display, 3GB RAM, 32GB eMMC, Wi-Fi, Android 6.0+)
- Samsung Galaxy Tab A 10.1 (10.1" display, 2GB RAM, 32GB eMMC, Wi-Fi, Android 7.0+)

# 9 Certificates and Declarations – MIRAI Controller

## 9.1 Declaration of Conformity

CE

spo-comm

EG Konformitätserklärung
Declaration of Conformity

spo-comm GmbH
Beethovenstraße 22
D-91207 Lauf a.d. Pegnitz
Germany

erklärt, dass das Produkt
*declares that the product*

spo-comm Mini-PC **spo-book QUADRO P1000**
*spo-comm Mini-PC **spo-book QUADRO P1000***

mit den Vorschriften folgender Europäischer Richtlinien übereinstimmt:
*complies with the requirements of the European Directives:*

| | |
|---|---|
| **2014/30/EU**<br>*2004/108/EG* | Elektromagnetische Verträglichkeit (EMV)<br>*electromagnetic compatibility* |
| **2011/65/EU**<br>*2011/65/EU* | Beschränkung der Verwendung bestimmter gefährlicher Stoffe in Elektro- und Elektronikgeräten (RoHS)<br>*RoHs* |

**Zutreffende Normen** // *Applicable Standards*

| | |
|---|---|
| **EN60950-1:2006+A11:2009+A1:2010+A12:2011+A2:2013** | **EN50663**<br>**EN301 489-1 V2.2.0**<br>**EN301 489-3 V2.1.1**<br>**EN301 489-17 V3.2.0**<br>**EN 300 328 V2.1.1**<br>**EN 300 328 V2.1.1**<br>**EN 300 440 V2.2.1** |

Diese Erklärung bescheinigt die Übereinstimmung mit den genannten Richtlinien, beinhaltet jedoch keine Zusicherung von Eigenschaften im rechtlichen Sinne.
*This declaration certifies conformance with the above mentioned Directives. Affirmation of attributes in a legal sense is not included.*

Michael Sporrer, Geschäftsführer spo-comm GmbH
Michael Sporrer, CEO spo-comm GmbH

spo-comm

# Konformitätserklärung
# Declaration of Conformity

Die

spo-comm GmbH
Beethovenstraße 22
D-91207 Lauf a.d. Pegnitz
Germany

erklärt, dass das Produkt
*declares that the product*

spo-comm Mini-PC spo-book QUADRO P1000
*spo-comm Mini-PC spo-book* **QUADRO P1000**

mit den Vorschriften folgender Richtlinien übereinstimmt:
*complies with the requirements of the following specifications:*

**FCC Rules and Regulation under 47 CFR Part 15 Subpart B and ANSI C63.4 (2014)**

**FCC Part 15.107**
**FCC Part 15.109**

Diese Erklärung bescheinigt die Übereinstimmung mit den genannten Richtlinien, beinhaltet jedoch keine Zusicherung von Eigenschaften im rechtlichen Sinne.
*This declaration certifies conformance with the above mentioned Directives. Affirmation of attributes in a legal sense is not included.*

Michael Sporrer, Geschäftsführer spo-comm GmbH
Michael Sporrer, CEO spo-comm GmbH

## 9.2 Declaration of Incorporation – Micropsi Industries

# MICROPSI INDUSTRIES

EU Declaration of Incorporation *(in accordance with Machinery Directive 2006/42/EC Annex II, part B)*

| Manufacturer: | Person Authorized to Compile the Technical File: |
|---|---|
| micropsi industries GmbH<br>Möckernstrasse 120,<br>10963 Berlin, GERMANY | Naaimah Saghir<br>VP Product<br>micropsi industries |
| Description and Identification of the Partly Completed Machine(s) | |
| Product and Function: | Vision-based motion control system for industrial robot systems that enables such robot systems to solve automation problems with high variance in position, shape, or background and lighting conditions. The final function is determined by the completed machine (i.e., robot system, robot cell or robot application with intended use). |
| Model: | MIRAI<br>Software version 14.0.0 onwards |
| Incorporation: | The MIRAI vision-based motion control system shall only be put into operation upon being integrated into a final completed machine, which conforms with the provisions of the Machinery Directive and other applicable directives. |
| It is declared that the above product, for what is supplied, fulfils the directives as detailed below:<br>When this partly completed machinery is integrated and becomes a final machinery, the integrator is responsible for determining that the final machinery fulfils all applicable Directives and providing the Declaration of Conformity. | |
| (I) Machinery Directive 2006/42/EC | The following essential health and safety requirements were fulfilled:<br>Annex IV, clause 1.1.2, 1.1.3, 1.1.5, 1.2.2, 1.2.3, 1.5.1, 1.5.2, 1.7.1, 1.7.2, 1.7.4<br>It is declared that the relevant technical documentation was compiled in accordance with Annex VII, Part B<br>The following Harmonized Standard were used (where applicable):<br>EN 60204-1:2019 |
| (II) Low-Voltage Directive 2014/35/EU | The following Harmonized Standard were used (where applicable):<br>EN IEC 60320-1:2023 |
| (III) Radio Equipment Directive 2014/53/EU | The following standards were used:<br>Article 3.1a): EN60950-1:2006+A11:2009+A1:2010 +A12:2011+A2:2013; EN 50663: 2017<br>Article 3.1b): Draft EN 301 489-1 V2.2.0; final draft EN 301 489-3 V2.1.1; Draft EN 301 489-17 V3.2.0<br>Article 3.2): EN 300 328 V2.1.1; ETSI EN 301 893 V2.1.1; ETSI EN 300 440 V2.2.1 |
| (IV) Other directives | All applicable directives and harmonized standards have been followed by the component suppliers and can be provided upon request. |
| Reference to Other Technical Standards and Specifications Used: | |
| EN ISO 10218-2:2011, ISO/TS 15066:2016 | |
| The manufacturer, or his authorized representative, shall transmit relevant information about the partly completed machinery in response to a reasoned request by the national authorities. | |

Berlin, Germany, 15 February 2024

Ronnie Vuine, Chief Product Officer

https://www.micropsi-industries.com/
micropsi industries GmbH Möckernstrasse 120, 10963 Berlin, Germany +49 30 555 71 929
Micropsi Industries USA, Inc, 300 Brannan St. Suite #101, San Francisco, CA 94107, USA +1 718 440 7353