



**BITS Pilani**  
Pilani Campus

# Introduction to ABB 1410 Industrial Manipulator

Prof. BK Rout  
Mr. SK Das  
Mr. Chetan Jalendra

# WILL DISCUSS...

---

- ☐ System overview
- ☐ Descriptions of the FlexPendant and the controller.
- ☐ Safety instructions and warnings.
- ☐ Descriptions of connections and step-by-step instructions to the most common tasks
- ☐ Navigating and handling the FlexPendant.
- ☐ Procedures for jogging.
- ☐ Procedures for programming and testing, including descriptions of some concepts for programming.

# ABB 1410



# System Overview

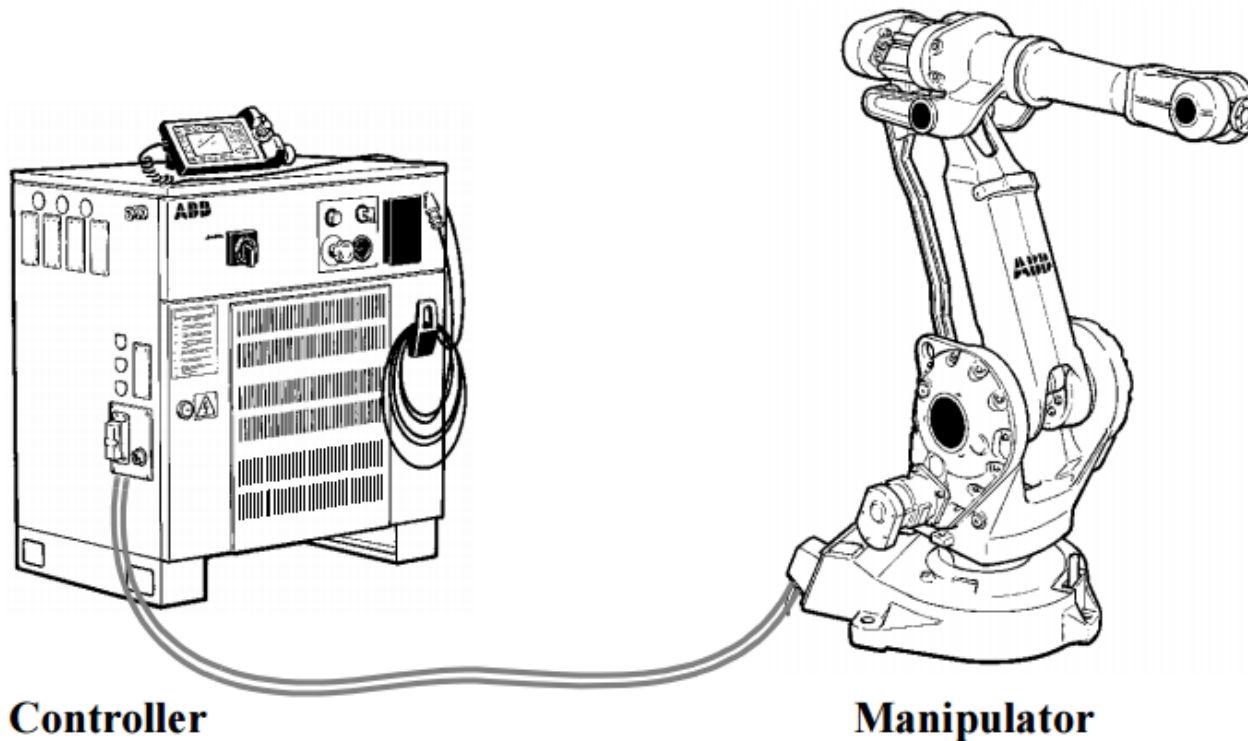
---

## IRB 1410:-

- **Reliable - Stiff and robust design** :The IRB 1410 design translates into low noise levels, long service intervals and long economic life.
- **Accurate - Consistent parts quality** : Superior levels of control and path-following accuracy (+ 0.05 mm) provide excellent work quality.
- **Strong - Maximized utilization** : The robot has a large working area and long reach (max 1.44 m). The handling capacity is **5kg load** on the upper arm for process equipment.
- **Fast-Short cycle times** :The robust design together with the fast and accurate IRC5 controller enables short cycle times.

# System Overview

- A robot is made up of two principal parts: CONTROLLER and MANIPULATOR.
- We can communicate with the robot using a teach pendant and/or an operator's panel located on the controller.



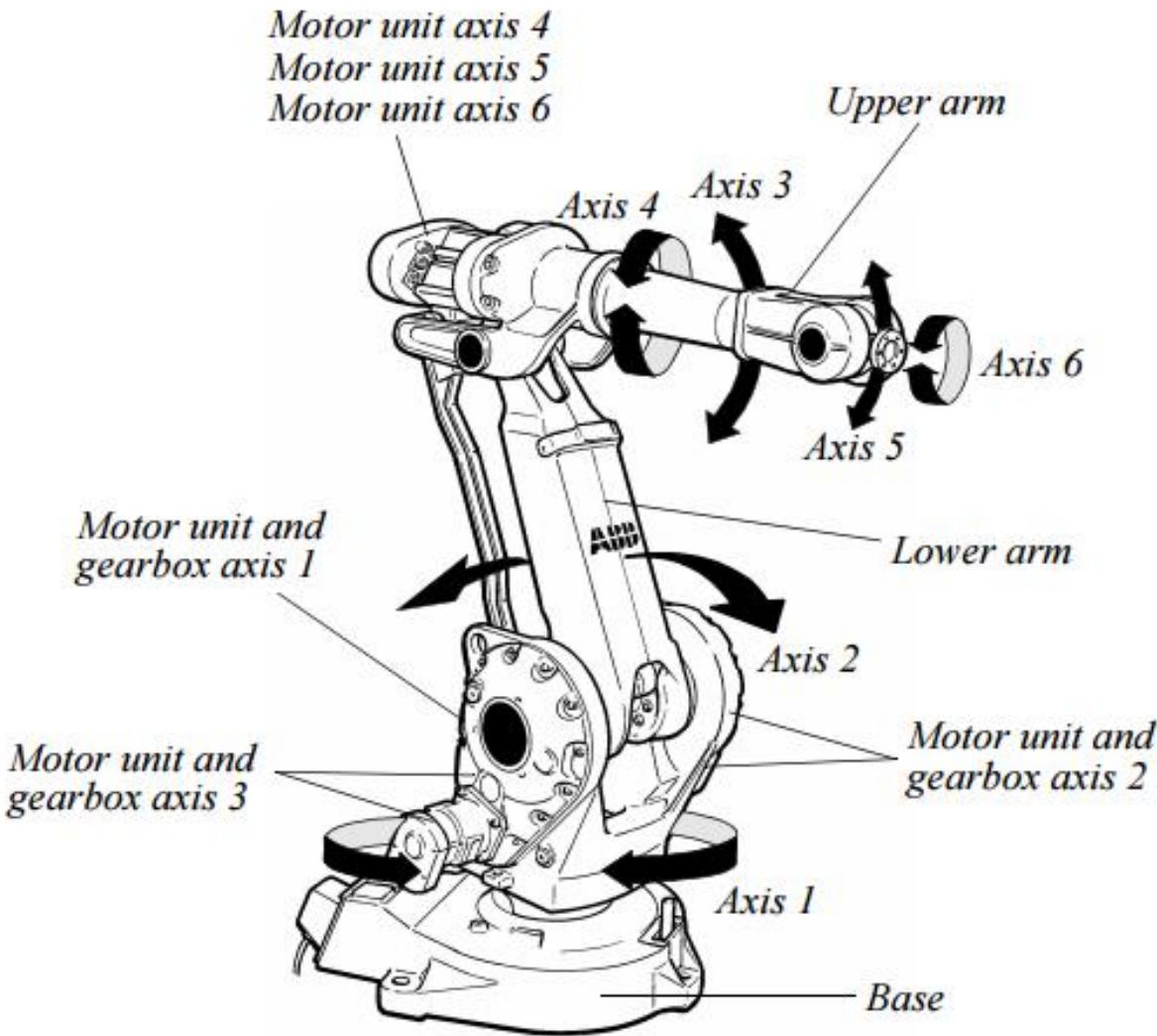
# Robot Information

## Robot Specifications

Axes:	6
Payload:	5kg
H-Reach:	1444mm
Repeatability:	±0.05mm
Robot Mass:	225kg
Structure:	

## Robot Motion Range

Axis 1	±170°
Axis 2	±70°
Axis 3	+70° - 65°
Axis 4	±150°
Axis 5	±115°
Axis 6	±300°



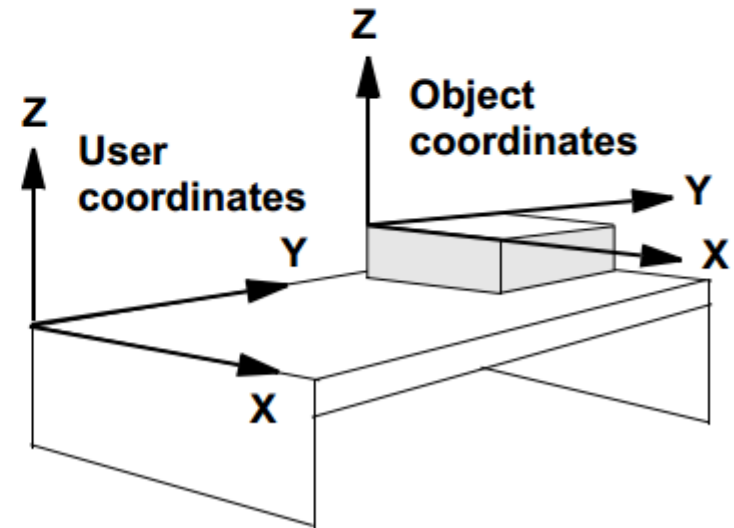
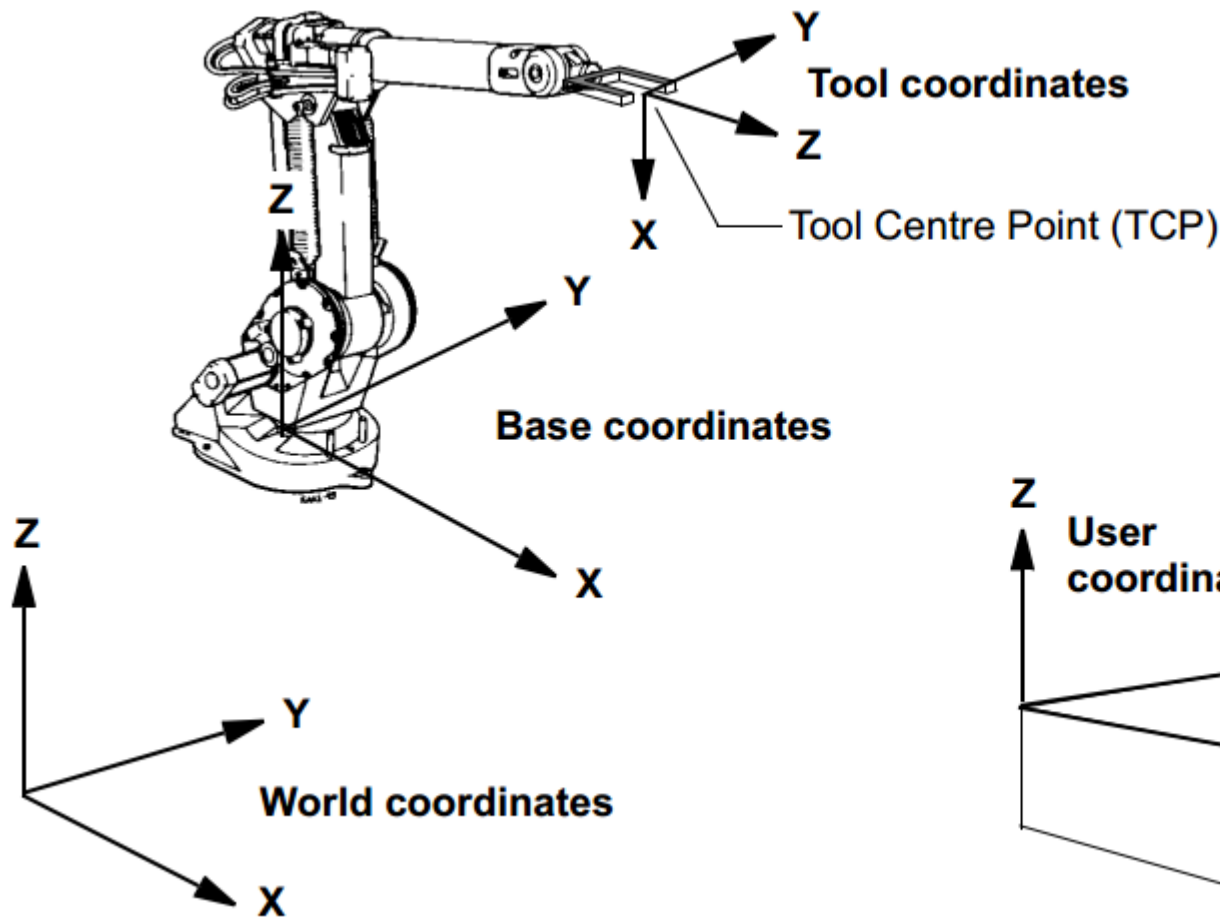
# Manipulator Load

Maximum load in relation to the base coordinate system

## Floor Mounted

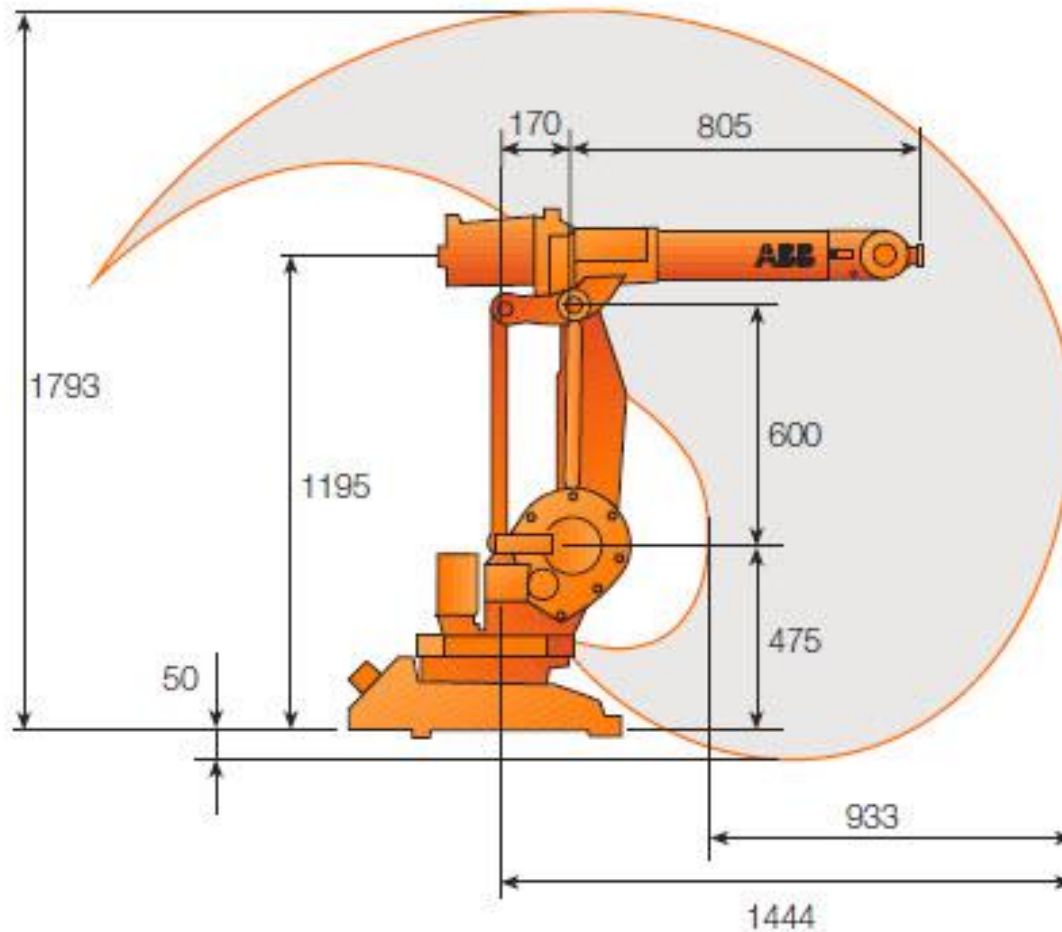
Force	Endurance load (in operation)	Max. load (emergency stop)
Force xy	$\pm 1500$ N	$\pm 2000$ N
Force z	$+2800 \pm 500$ N	$+2800 \pm 700$ N
Torque xy	$\pm 1800$ N	$\pm 2000$ N
Torque z	$\pm 400$ N	$\pm 500$ N

# Coordinate systems

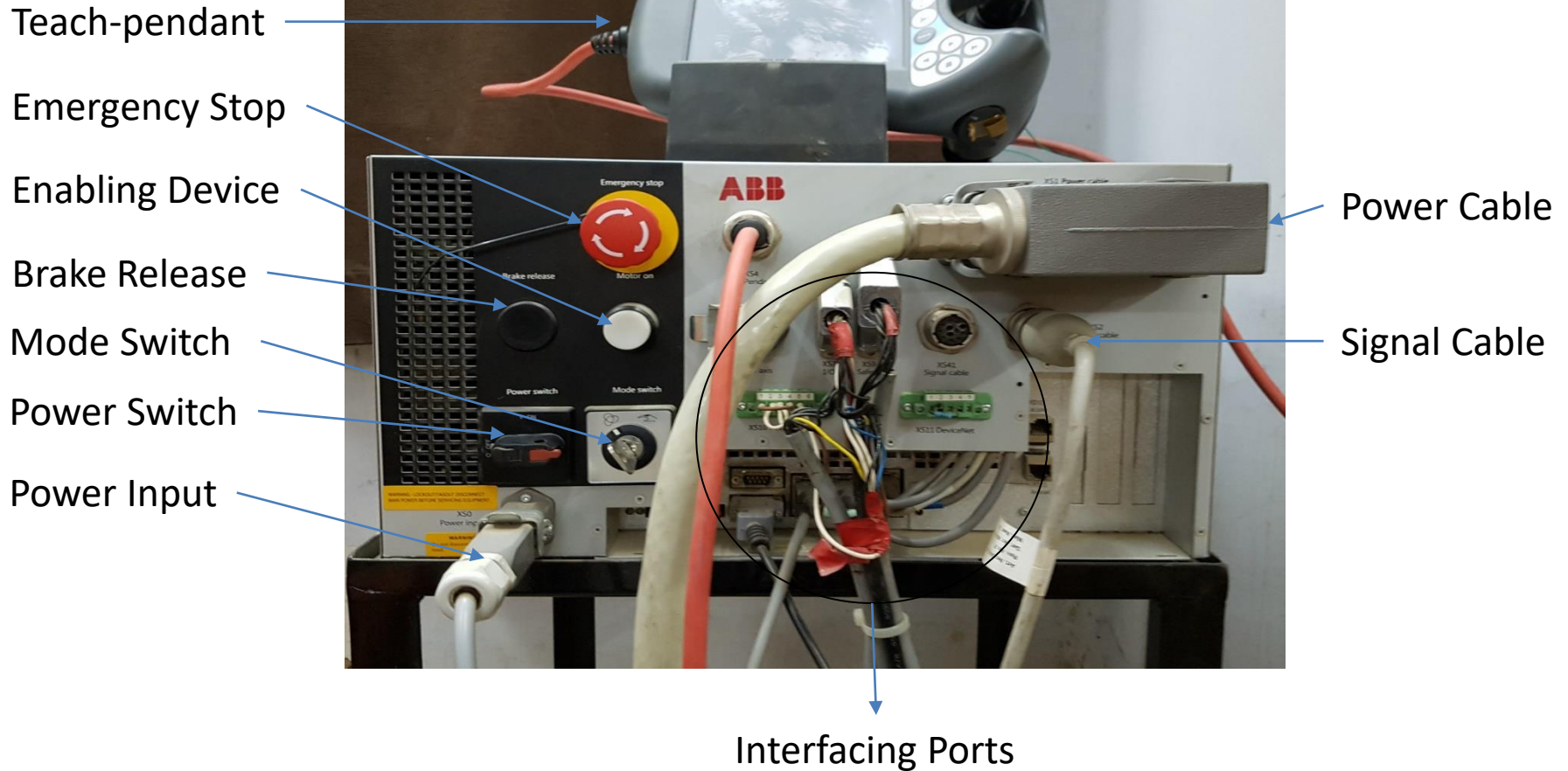




# Workspace



# The controller



# System Overview

Teach-pendant



Pneumatic power supply



# What is an IRC5 controller ?

The IRC5 controller contains all functions needed to move and control the robot.

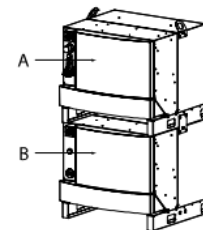
The base variant of the IRC5 controller, can consist of a single cabinet or be divided into two separate modules-

**A) Control module** : The control module **contains all the control electronics** such as main computer, I/O boards, and flash memory.

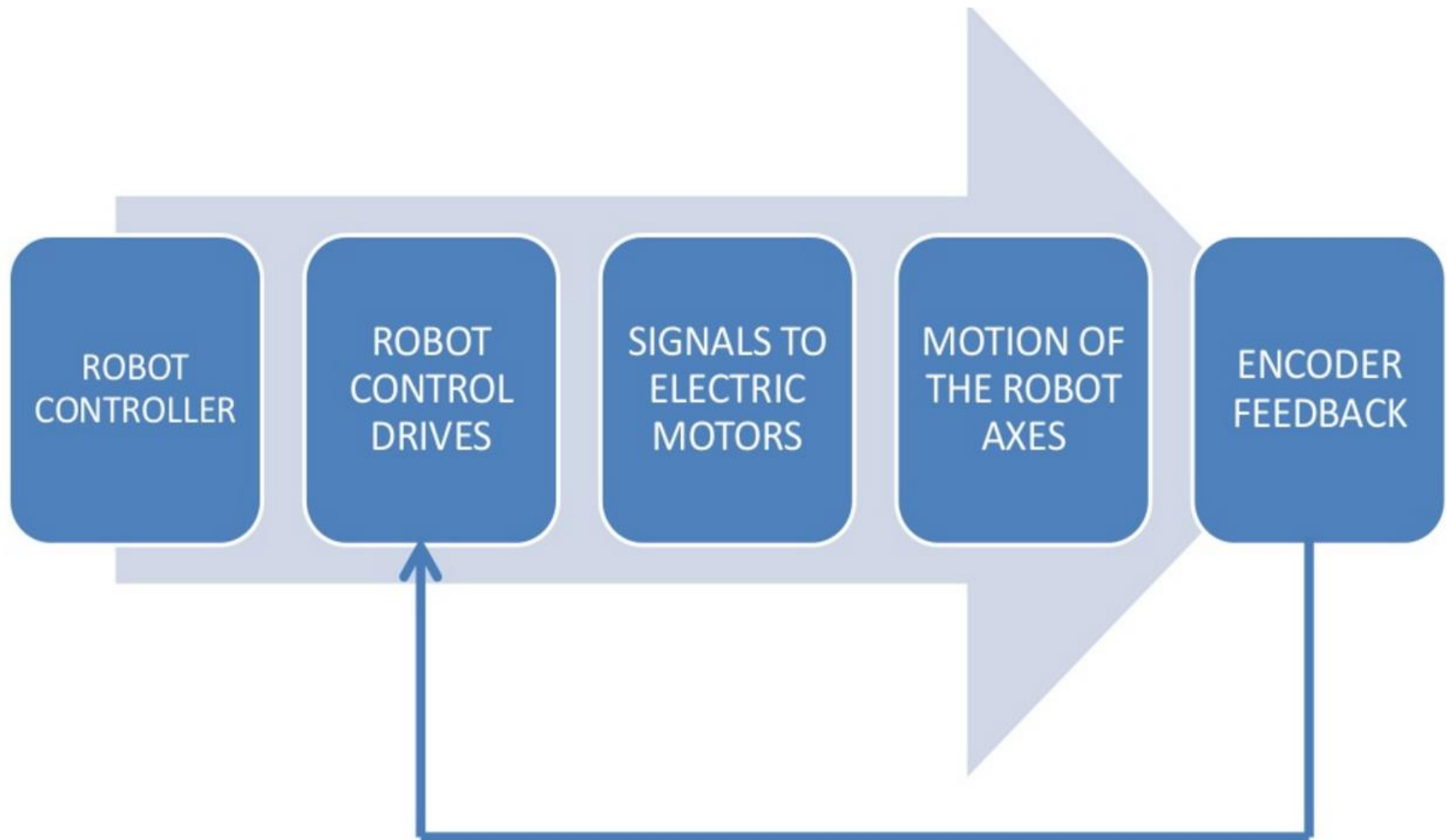
The control module runs all software necessary for operating the robot .

**B) Drive module** : The drive module **contains all the power electronics** supplying the robot motors. An IRC5 drive module may contain **nine drive** units and handle six internal axes depending on the robot model.

In a single cabinet, the control and drive module are integrated into one single module.

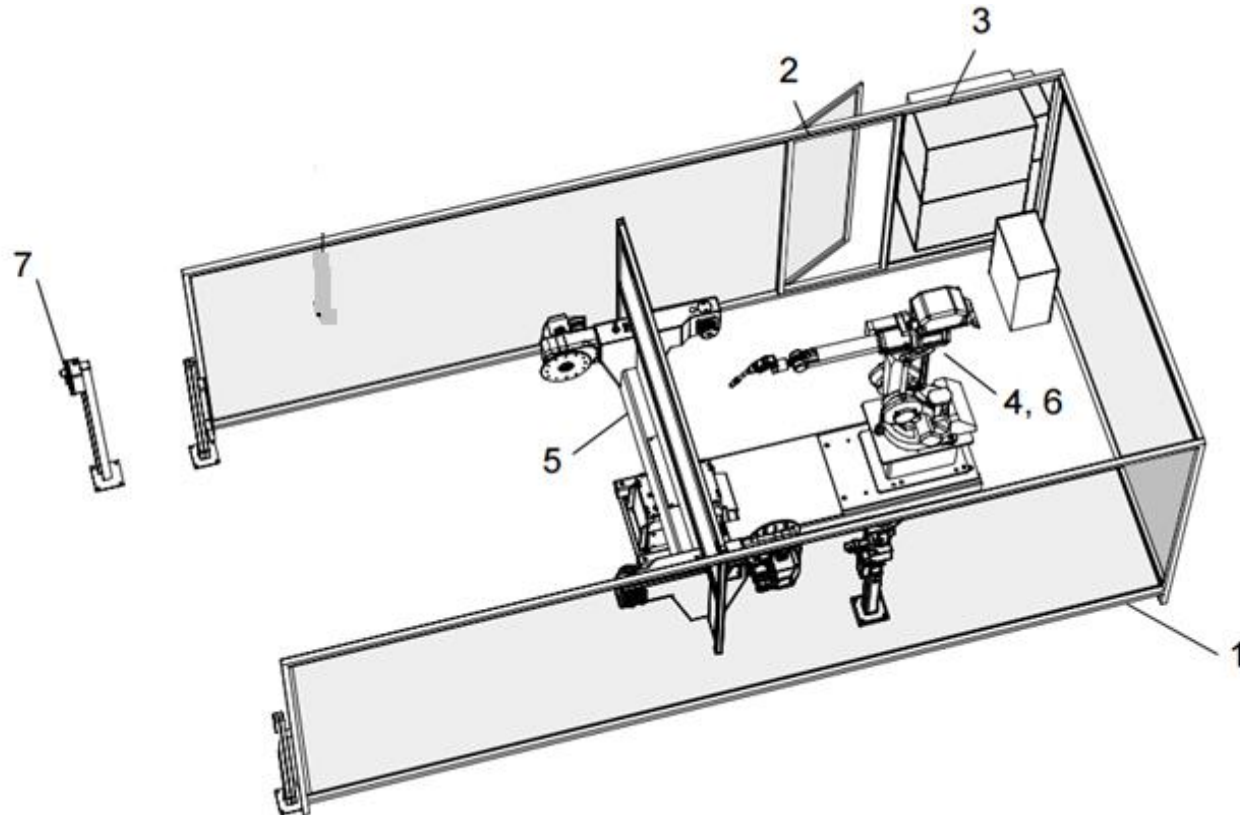


# Process of the Robot Controller



# System Block Diagram: Station Layout

- 1- Protective barrier
- 2- Gate interlock
- 3- Control equipment
- 4- Robot
- 5- Positioner
- 6- Wire feeder system
- 7- Operator panel

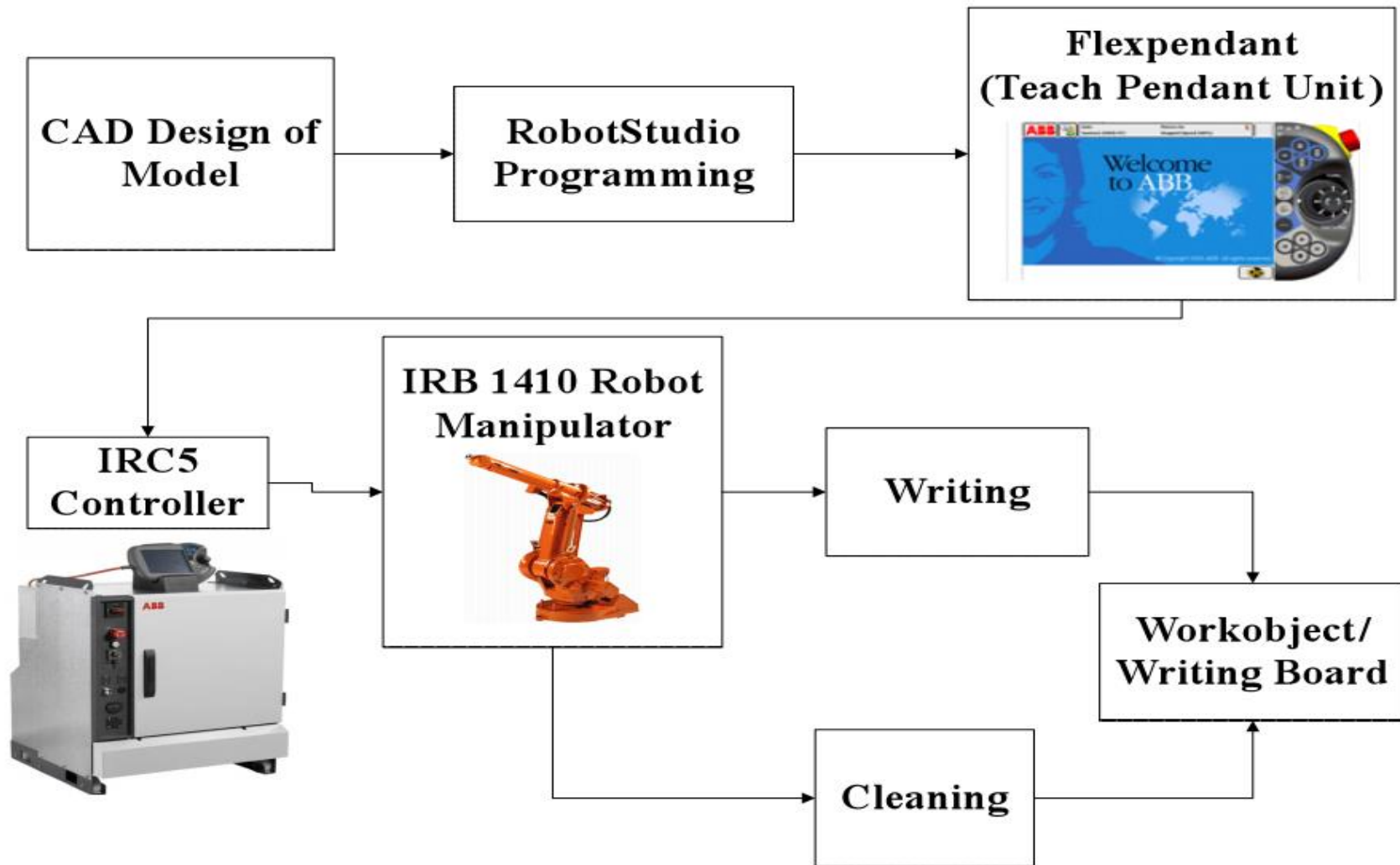




# Lab layout :



# System Block Diagram:





# Safety Functions...

---

## **Emergency stop** – IEC 204-1 :

A condition which overrides all other robot controls, removes drive power from robot axis actuators, stops all moving parts and removes power from other dangerous functions controlled by the robot.

## **Enabling device** – ISO 11161 :

A manually operated device which, when continuously activated in one position only, allows hazardous functions but does not initiate them. In any other position, hazardous functions can be stopped safely.

## **Safety stop** – ISO 10218 (EN 775) :

When a safety stop circuit is provided, each robot must be delivered with the necessary connections for the safeguards and interlocks associated with this circuit. It is necessary to reset the power to the machine actuators before any robot motion can be initiated.

# Safety symbols on manipulator labels...

DANGER



CAUTION



WARNING



ELECTROSTATIC DISCHARGE



ELECTRICAL SHOCK



# Safety symbols on manipulator labels...



Crush-Risk of crush injuries.



Heat-Risk of heat that can cause burns.



Lifting of robot



Mechanical stop



No mechanical stop

# Welcome to FlexPendant

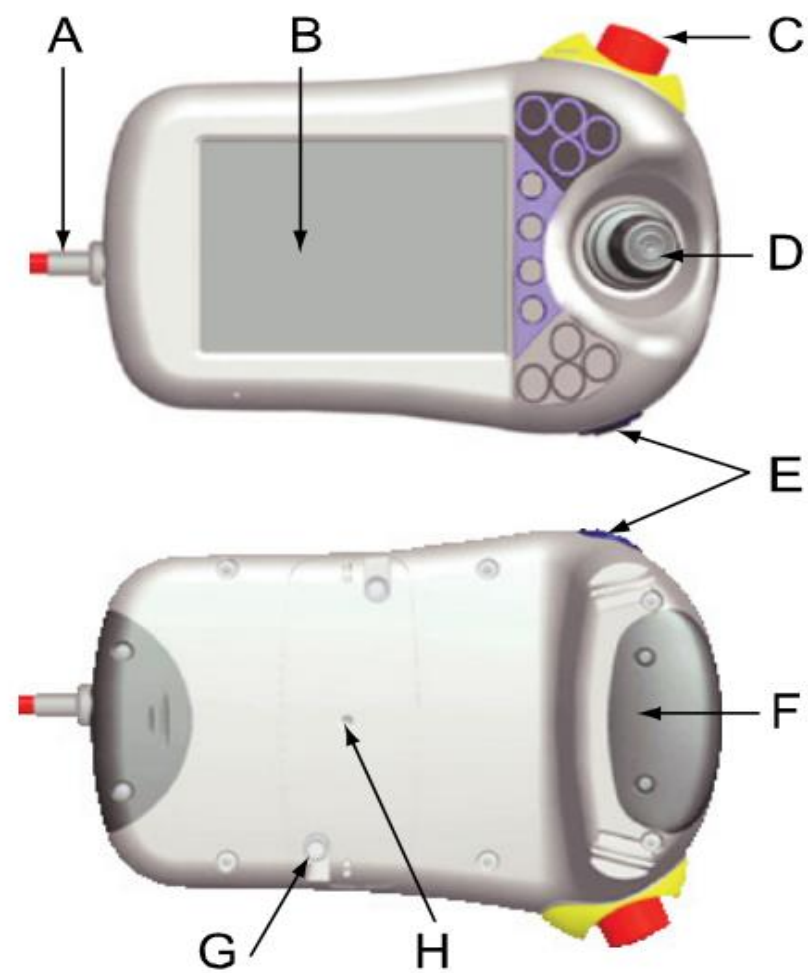
A basic IRC5 robot system consists of

- a robot controller,
- the FlexPendant,
- Robot-Studio, and one or several robots or other mechanical units.

The **FlexPendant** (called TPU or **teach pendant unit**) is a hand-held operator unit used to perform many of the tasks involved when operating a robot system: running programs, jogging the manipulator, modifying robot programs etc.



# Main parts of the FlexPendant...



A	Connector
B	Touch screen
C	Emergency stop button
D	Joystick
E	USB port
F	Enabling device
G	Stylus pen
H	Reset button

# FlexPendant

**Joystick :** Use the joystick to move the manipulator. This is called jogging the robot.

There are several settings for how the joystick will move the manipulator.

**USB port :** Connect a USB memory to the USB port to read or save files.

The USB memory is displayed as drive/USB.

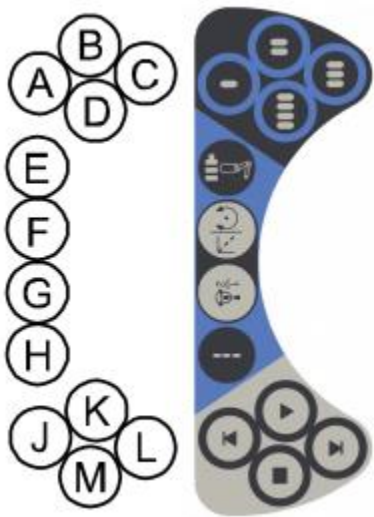
**Stylus pen :**The stylus pen included with the FlexPendant is located on the back.

Pull the small handle to release the pen.

**Reset button :** The reset button resets the FlexPendant, not the system on the controller.

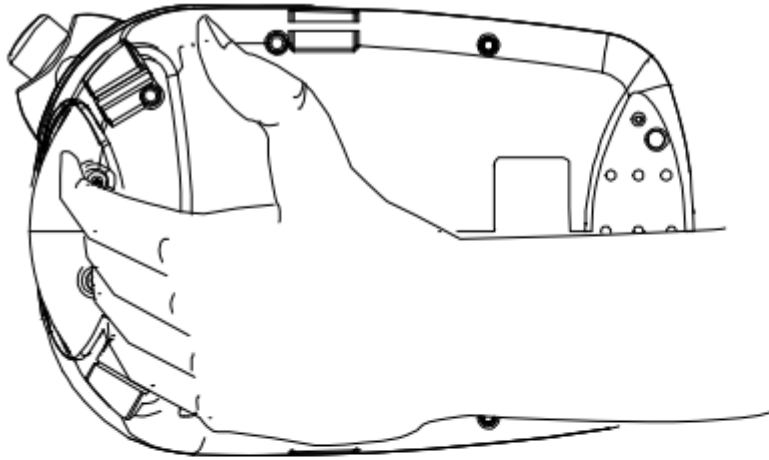
**Hard buttons :** There are dedicated hardware buttons on the FlexPendant.

You can assign your own functions to four of the buttons.

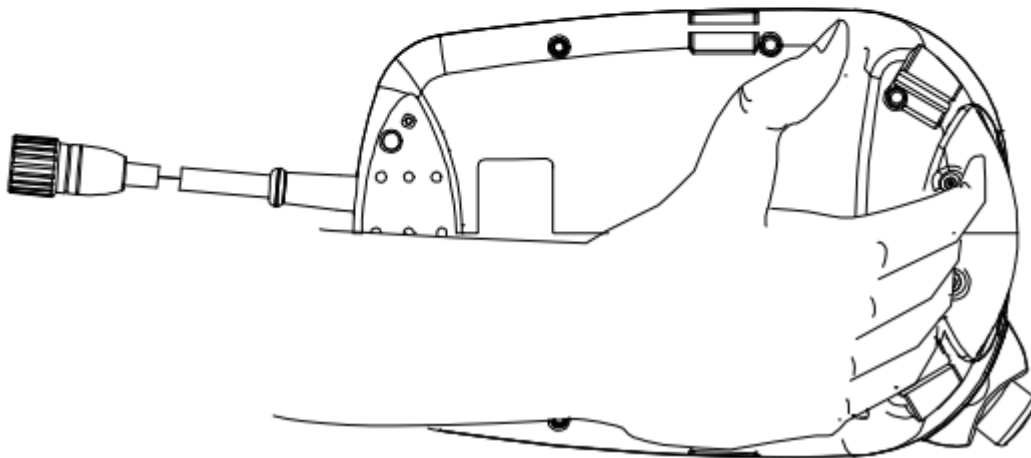


A - D	Programmable keys, 1 - 4. How to define their respective function is detailed in section Programmable keys,.
E	Select mechanical unit.
F	Toggle motion mode, reorient or linear.
G	Toggle motion mode, axis 1-3 or axis 4-6.
H	Toggle increments.
J	<b>Step BACKWARD</b> button. Executes one instruction backward as button is pressed.
K	<b>START</b> button. Starts program execution.
L	<b>Step FORWARD</b> button. Executes one instruction forward as button is pressed.
M	<b>STOP</b> button. Stops program execution.

# How to hold the FlexPendant ?



**The FlexPendant is typically operated while being held in the hand.**



# Important Elements Of The Flexpendant Touch Screen...



A	ABB menu
B	Operator window
C	Status bar
D	Close button
E	Task bar
F	Quickset menu



# Elements Of The Flex-pendant Touch Screen



- Operator window** : The operator window displays messages from robot programs.  
This usually happens when the program needs some kind of operator response in order to continue.
- Status bar** : The status bar displays important information about system status, such as operating mode, motors on/off, program state and so on.
- Close button** : Tapping the close button closes the presently active view or application.
- Task bar** : You can open several views from the ABB menu, but only work with one at a time. The task bar displays all open views and is used to switch between these.
- Quickset menu** : The quickset menu provides settings for jogging and program execution.

# ABB Menu

---

The following items can be selected from the ABB menu:

- HotEdit
- Jogging
- Program Editor
- Backup and Restore
- Control Panel
- Flex Pendant Explorer
- Inputs and Outputs
- Production Window
- Program Data
- Calibration
- Event Log
- System Info

# ABB MENU



The screenshot displays the ABB robot control interface. At the top left is the ABB logo. To its right is a status bar with a yellow warning icon and the text "Error Not Acknowledged" and "50026 Close to singularity". Below this is the "Event Log - Event Message" section. It shows "Event Message 50026" with a timestamp of "2013-08-30 05:19:35". A red square icon with a white 'X' is next to the message. The text "Close to singularity" is displayed. Below this, the "Description" section contains the text: "Task: -", "Robot too close to singularity.", "Program Ref. -", and "(Internal code: 17)". The "Actions" section contains the text: "Modify path away from the singularity or change to joint interpolation." At the bottom of the event log section are two buttons: "Show Log" and "Acknowledge". Below the event log is a "Jogging" button with a joystick icon. To the right of the interface is a large joystick with various buttons. A yellow highlight is on the joystick, with a green "Enable" button and a grey "Hold To Run" button. At the bottom right is a "ROB\_1" button with a robot icon.

**ABB** Error Not Acknowledged  
50026 Close to singularity

Event Log - Event Message

Event Message 50026 2013-08-30 05:19:35

Close to singularity

Description  
Task: -  
Robot too close to singularity.  
Program Ref. -  
(Internal code: 17)

Actions  
Modify path away from the singularity or change to joint interpolation.

Show Log Acknowledge

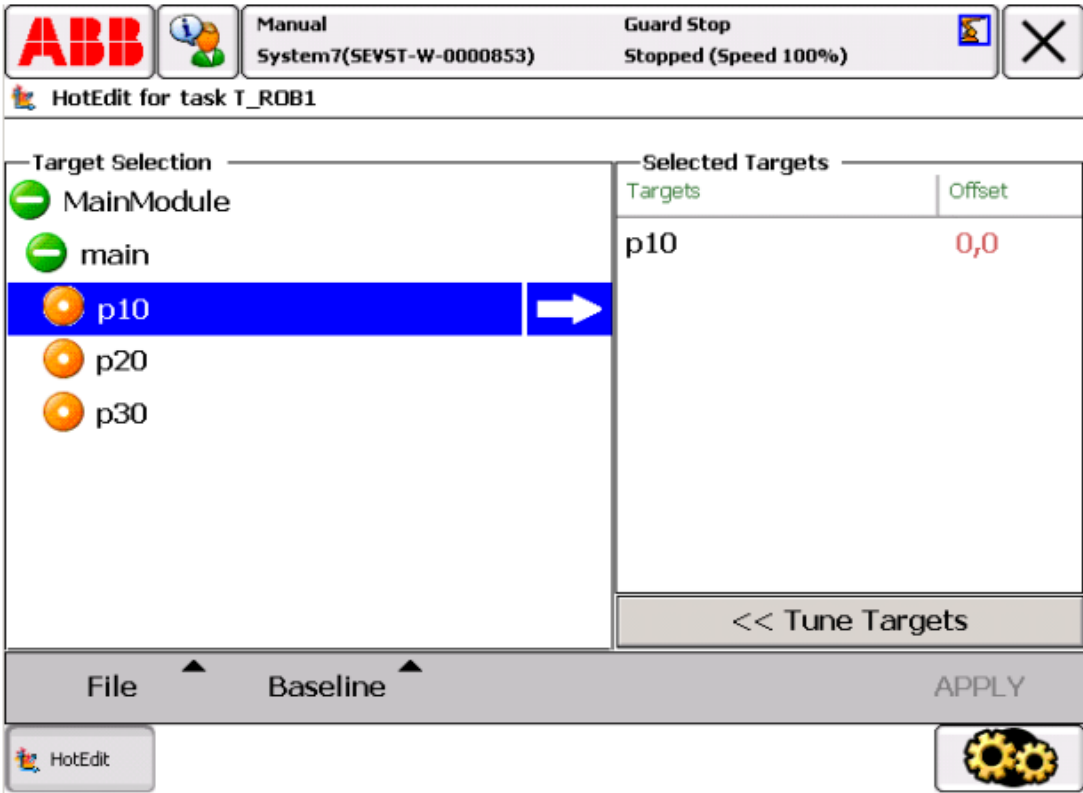
Jogging

ROB\_1

# HotEdit

HotEdit is a function for editing programmed positions. This can be done in all operating modes, even while the program is running.

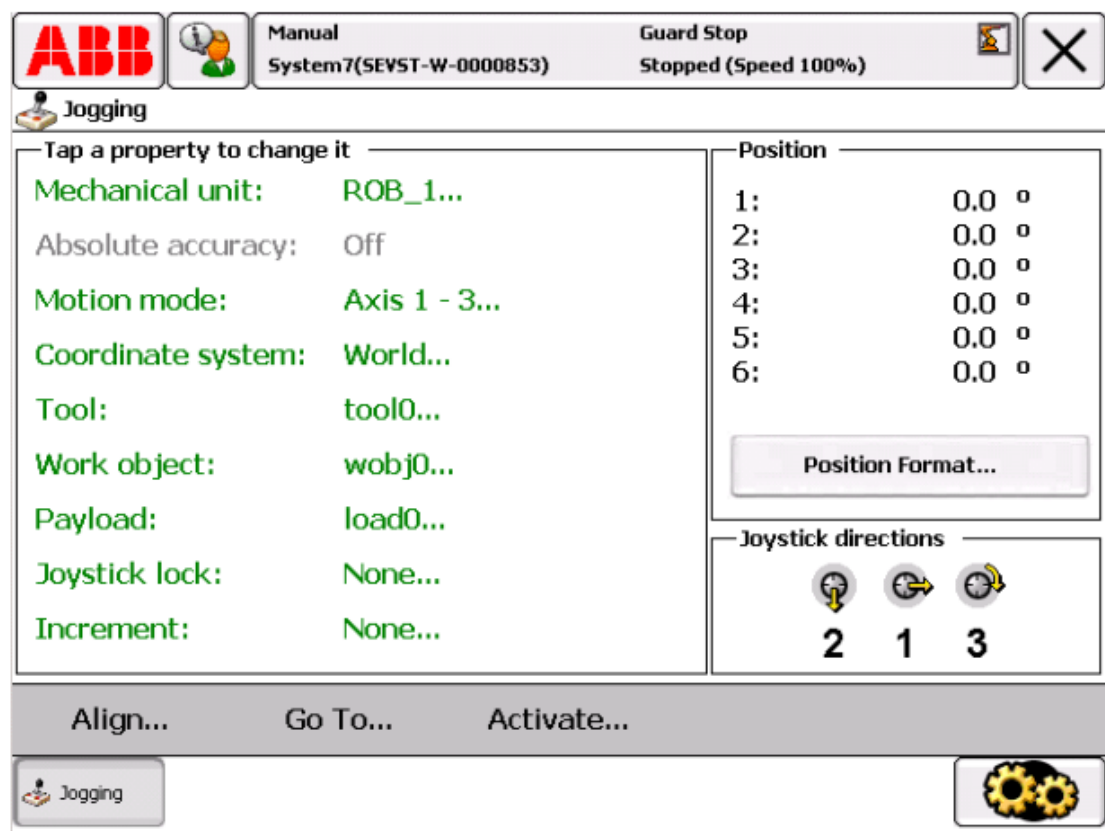
HotEdit can only be used for positions of the type rob-target.



# Jogging

The Jogging functions are found in the Jogging window.

The most commonly used properties are also available under the Quickset menu.

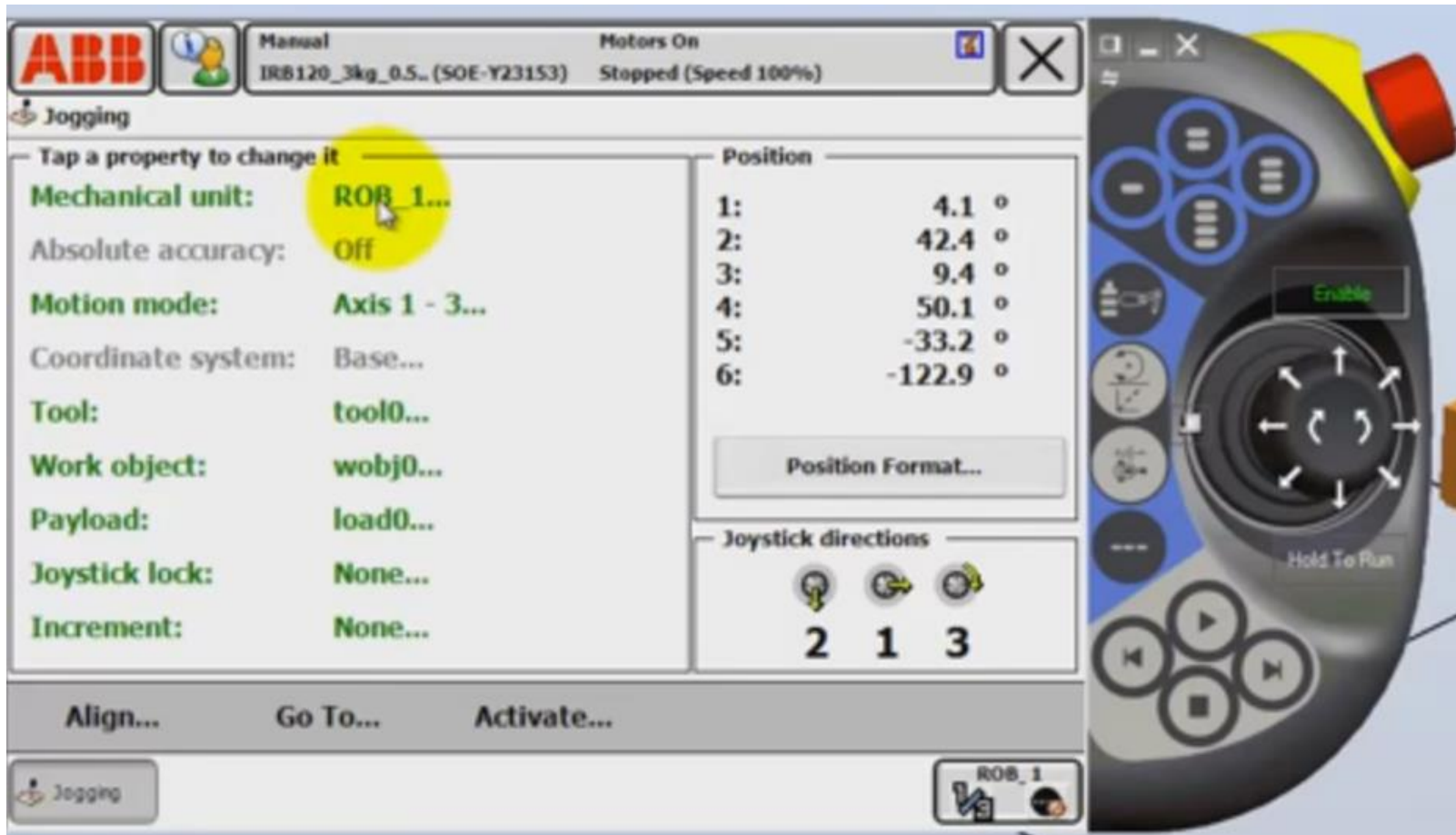


# Jogging





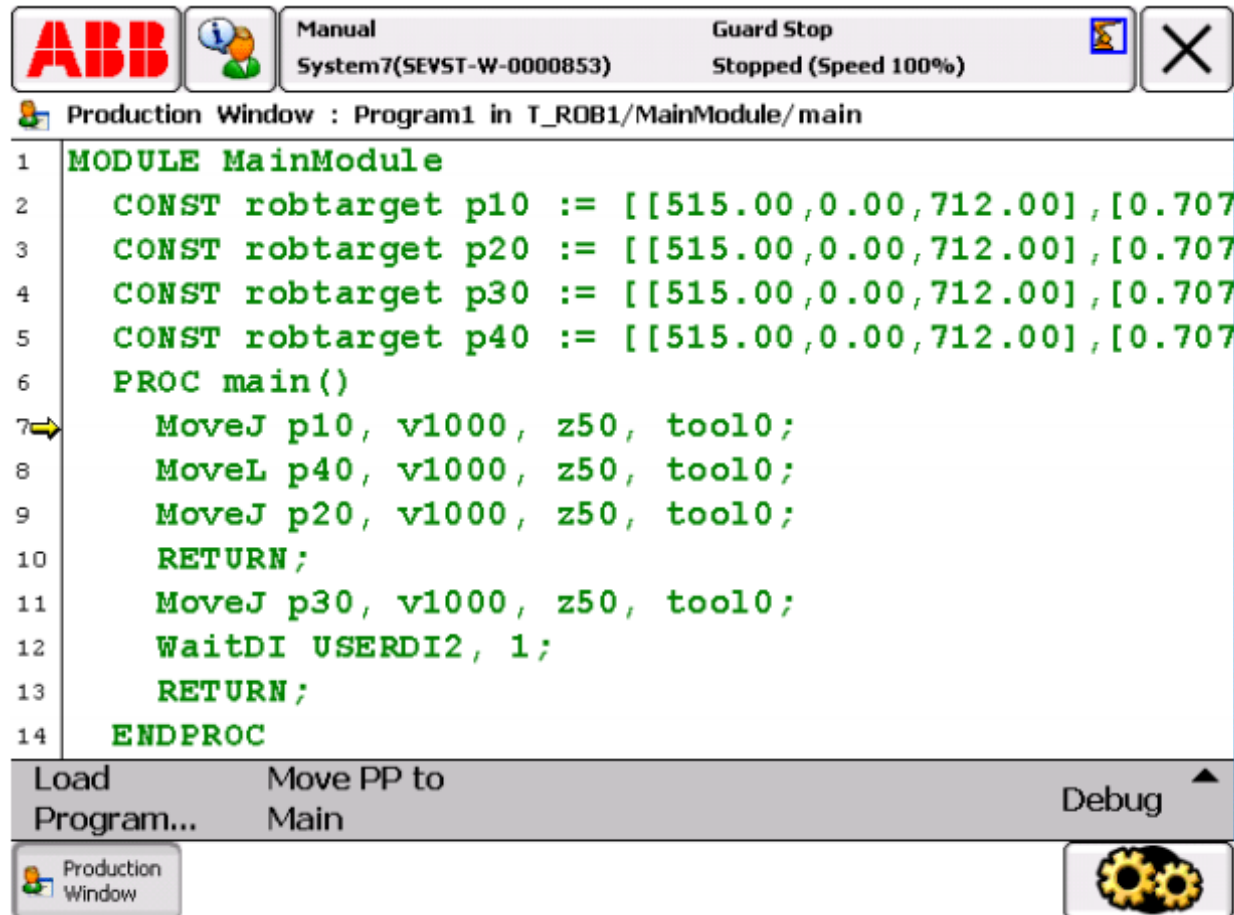
# Jogging





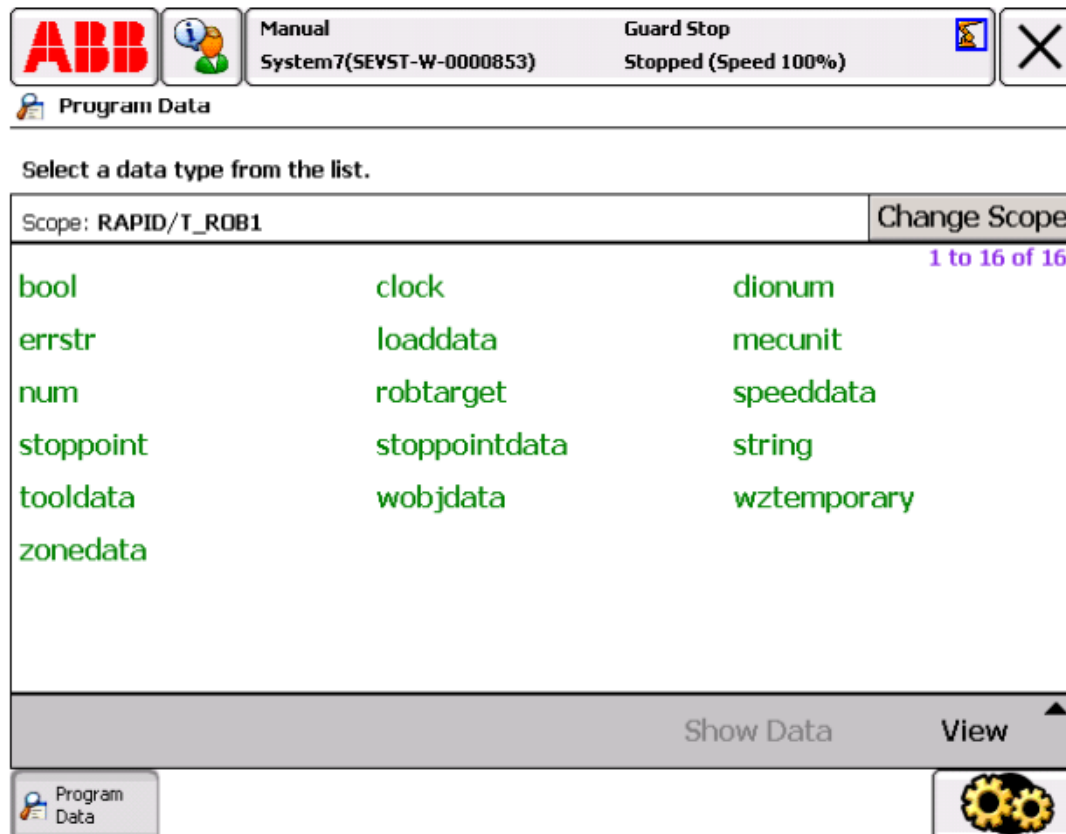
# Production Window

The Production window is used to view the program code while the program is running.



# Program data

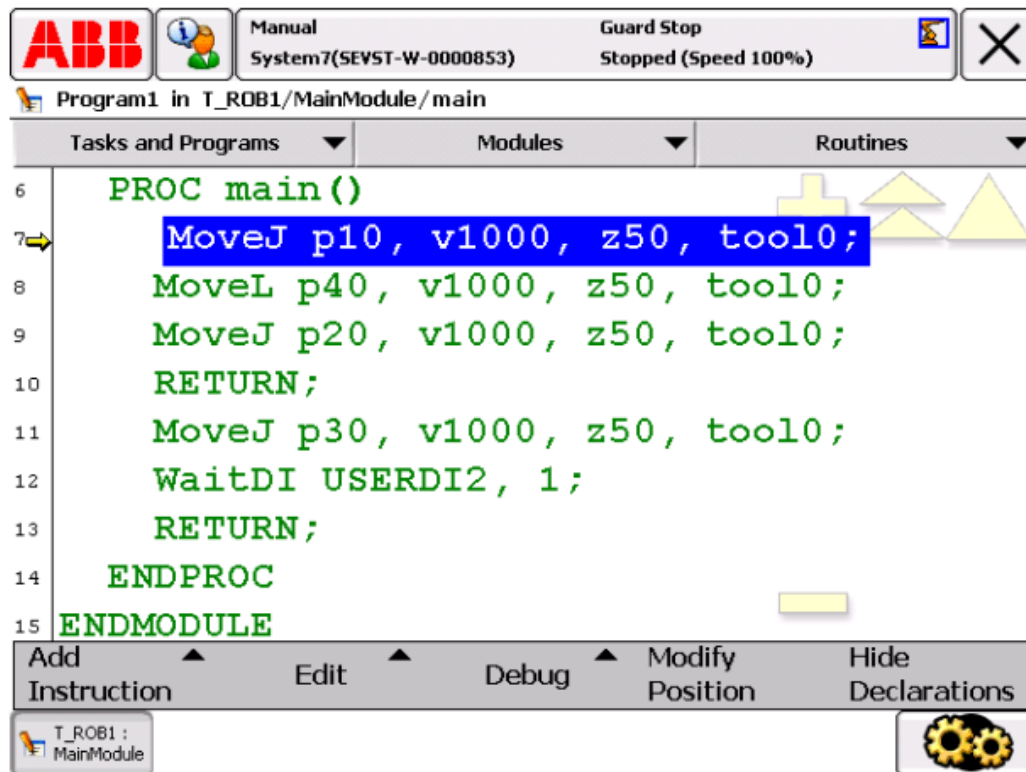
The Program data view contains functions for viewing and working with data types and instances.



# Program Editor

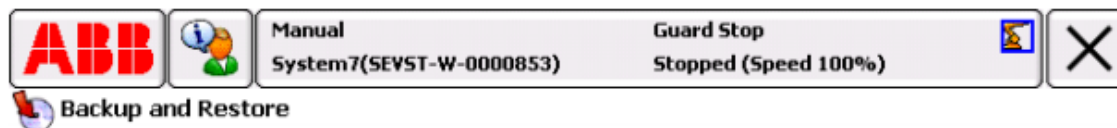
The Program editor is where you create or modify programs.

You can open more than one window of the Program editor, which can be useful when working with multitasking programs for instance.



# Backup And Restore

The Backup and restore menu is used for performing backups and restoring the system.



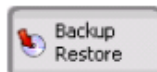
Choose to backup the current system or to restore an old system by selecting the corresponding icon.



Backup Current System...

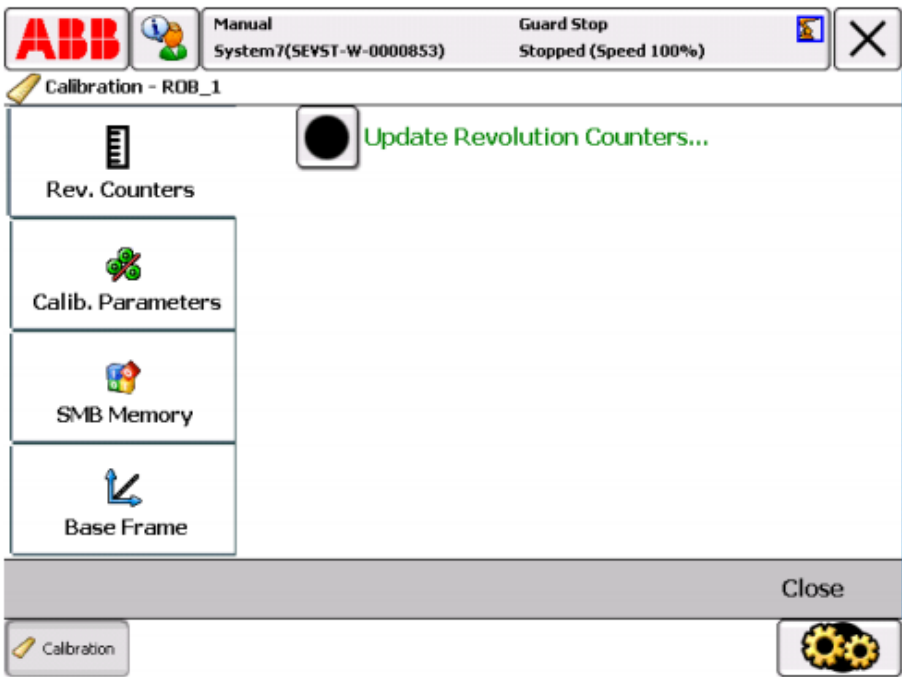
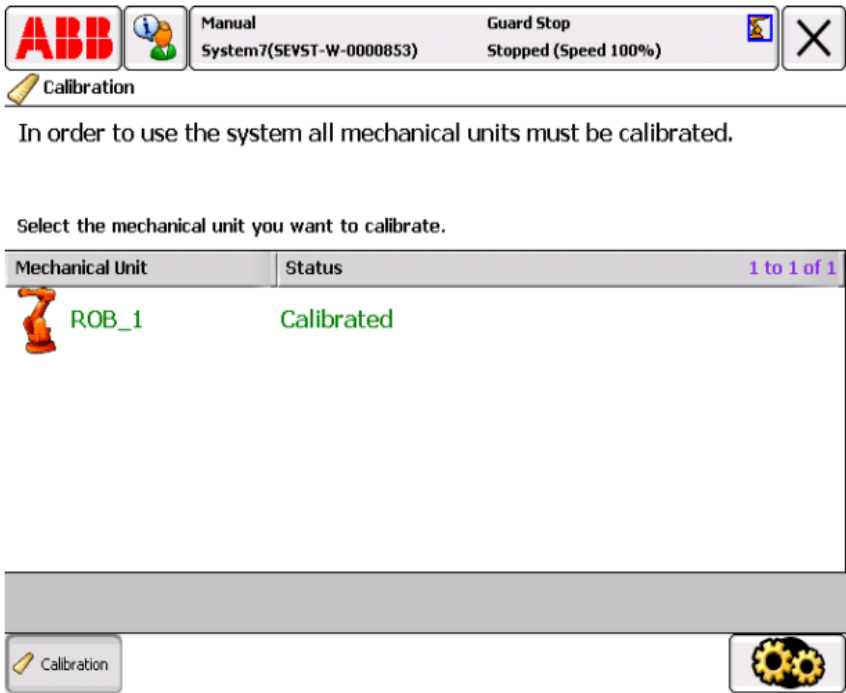


Restore System...



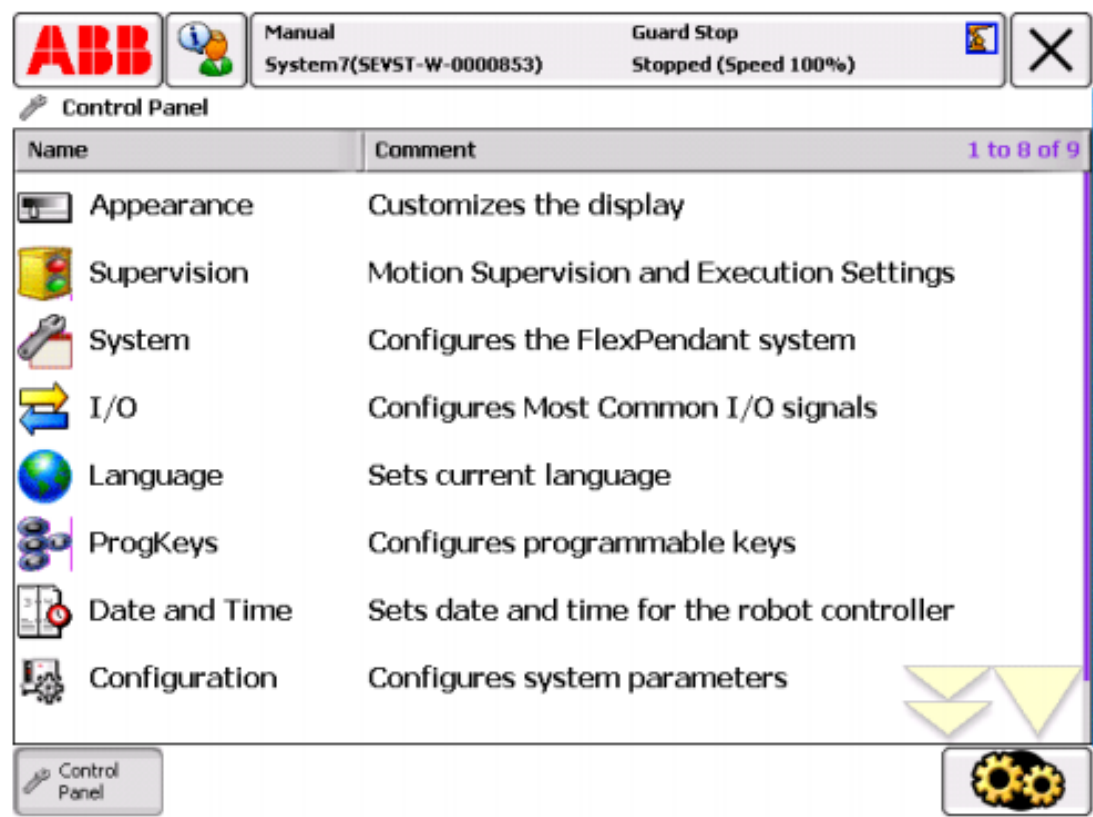
# Calibration

The Calibration menu is used to calibrate mechanical units in the robot system. Calibration can be performed using the options Pendulum Calibration or Levelmeter Calibration



# Control Panel

The Control panel contains functions for customizing the robot system and the FlexPendant



# Event Log Menu

Manual  
System7(SEYST-W-0000853)

Guard Stop  
Stopped (Speed 100%)

Event Log - Common

Tap a message to open it.

Code	Title	Date & Time
10002	Program pointer has been reset	2005-05-05 08:17:59
10040	Program loaded	2005-05-05 08:16:58
10041	Program deleted	2005-05-05 08:16:57
10012	Safety guard stop state	2005-05-05 07:52:53
10129	Program stopped	2005-05-05 07:52:51
10002	Program pointer has been reset	2005-05-05 07:52:51
10015	Manual mode selected	2005-05-05 07:52:50
10010	Motors OFF state	2005-05-05 07:52:50
10150	Program started	2005-05-05 07:52:50

Save All Logs As...

Delete

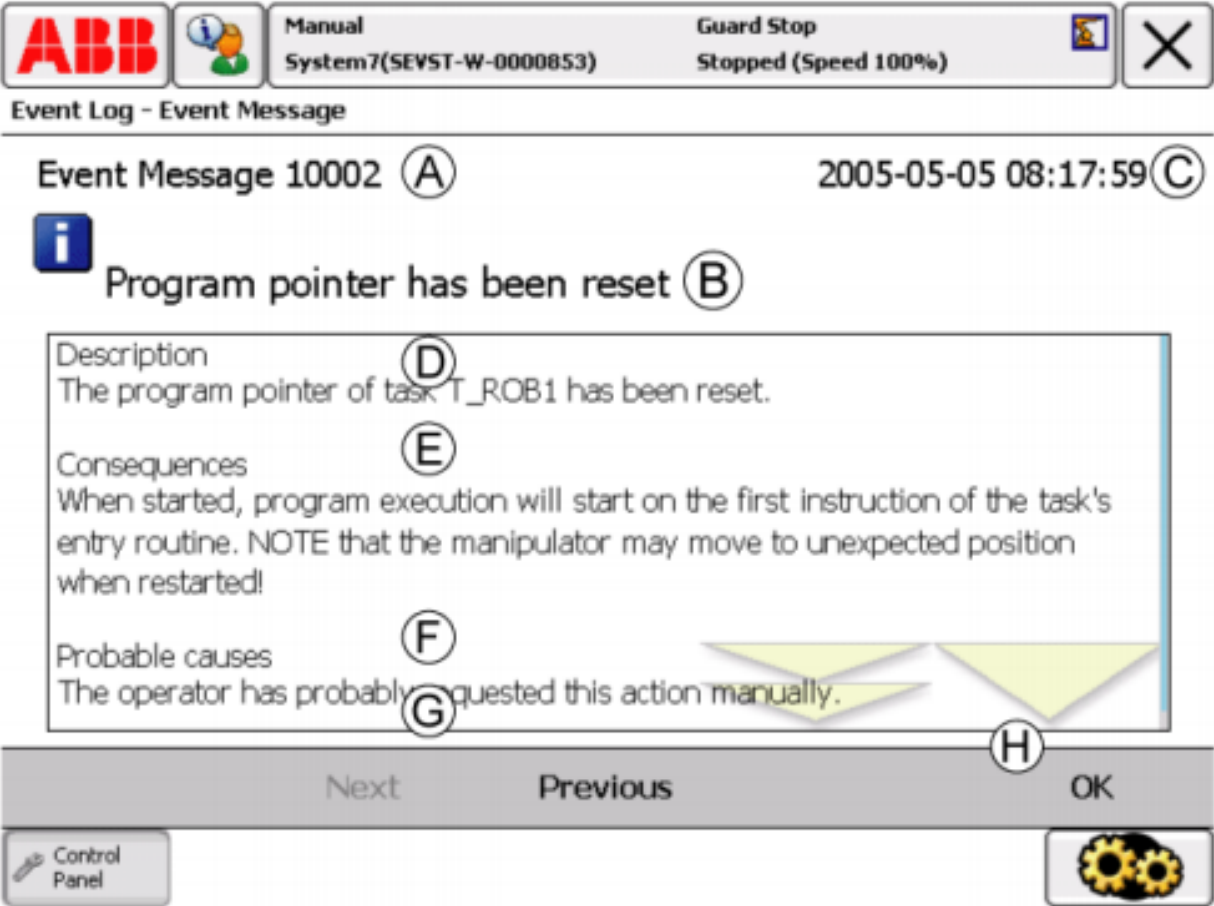
Update

View

Control Panel

# Event Log Menu

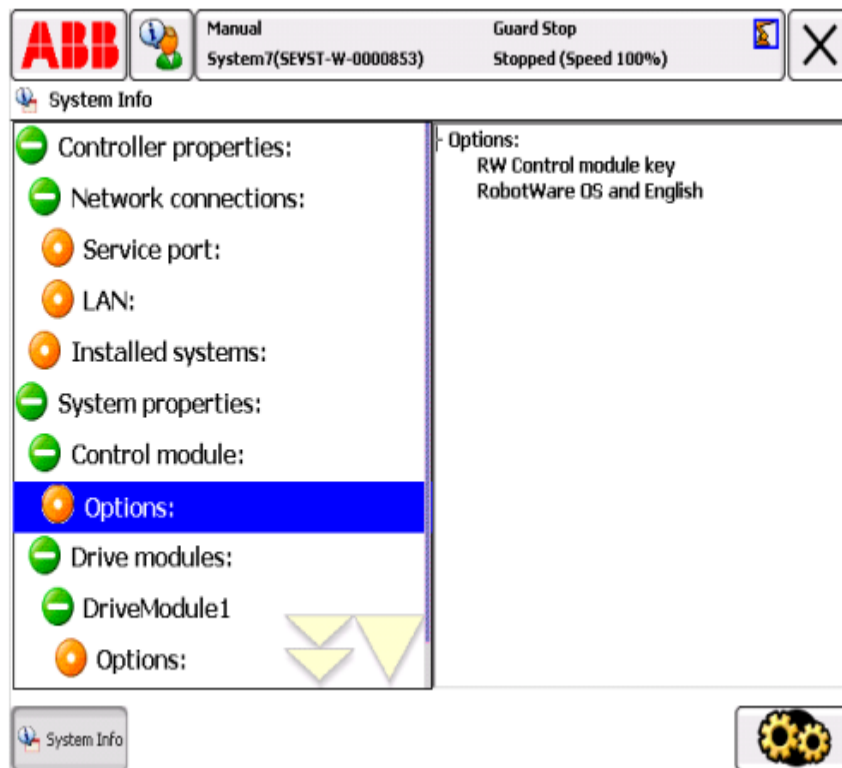
- A- Event number. All errors are listed by numbers.
- B- Event title. Briefly states what has happened.





# System info:

System info displays all settings, properties and program versions, valid for the controller and the installed system



**Controller properties-** Name of the controller.

**Network connections-** Settings for the service port and LAN.

**Installed systems-** Information on the installed systems.

**System properties-** information about the loaded system.

**Control module** - Name and key for the control module.

**Options** - All installed RobotWare options and languages.

**Drive module-** Lists all drive modules.

# Mode Selection Using The Operating Mode Selector



One **automatic** and two **manual** modes are available:



Manual mode: **< 250 mm/s** –

max. speed is 250mm/s 100% - full speed

- ✓ must be selected whenever anyone enters the robot's safeguarded space.



Automatic mode: The robot can be operated via a remote control device

- ✓ the operating mode selector is switched to , and all safety arrangements, such as doors, gates, light curtains, light beams and sensitive mats, etc., are active.

# Motion Control

---

## **TrueMove :**

Very accurate path and speed, based on advanced dynamic modeling. Speed independent path. Flexible and intuitive way to specify corner zones

## **QuickMove :**

By use of the dynamic model, the robot always and automatically optimizes its performance for the shortest possible cycle time. No need for manual tuning! This is achieved without compromising the path accuracy.

## **Coordinate Systems :**

A very powerful concept of multiple coordinate systems that facilitates jogging, program adjustment, copying between robots, off-line programming, sensor based applications, external axes co-ordination etc.

## **Singularity handling :**

The robot can pass through singular points in a controlled way, i.e. points where two axes coincide.

## **Motion Supervision :**

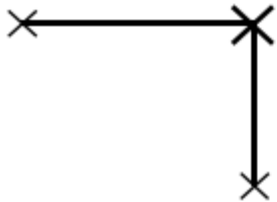
The behavior of the motion system is continuously monitored as regards position and speed level to detect abnormal conditions and quickly stop the robot if something is not OK.

# Movements

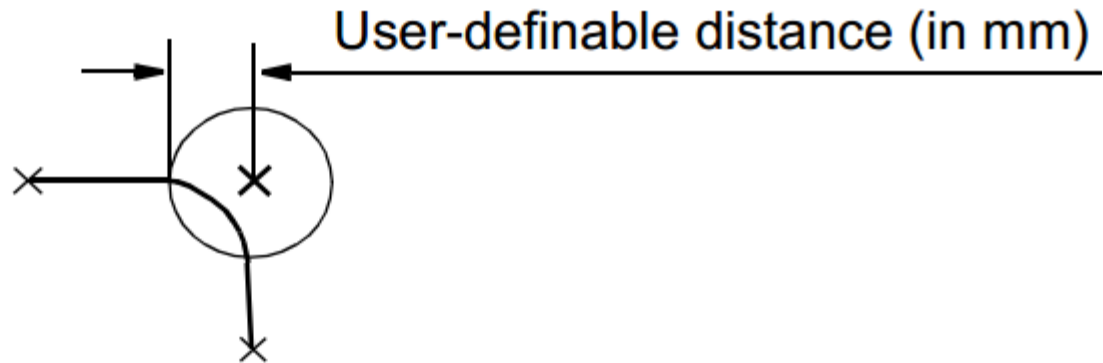
The exact position can be defined

1. a **stop point**, i.e. the robot reaches the programmed position
2. a **fly-by point**, i.e. the robot passes close to the programmed position.

## Stop point



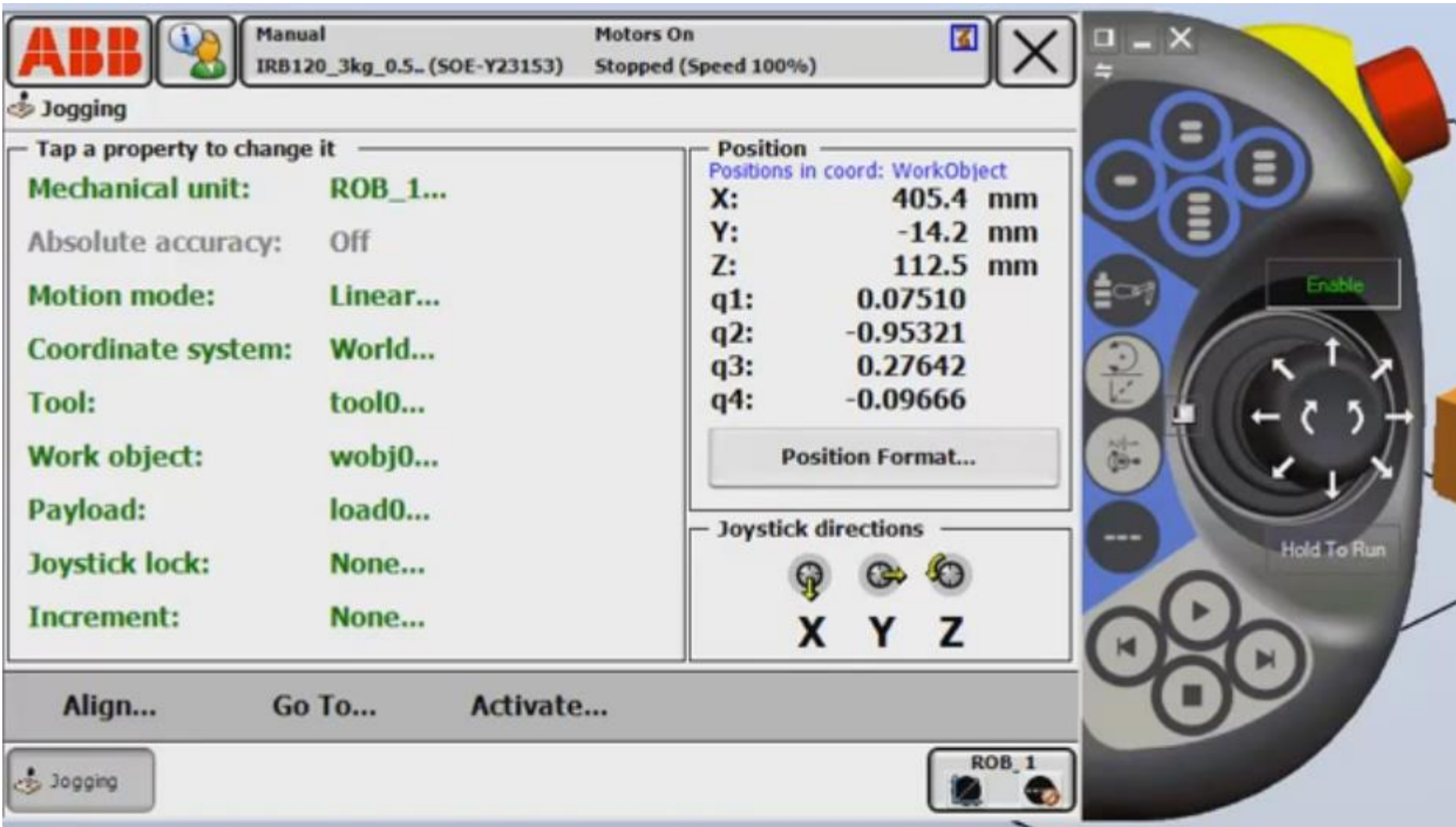
## Fly-by point



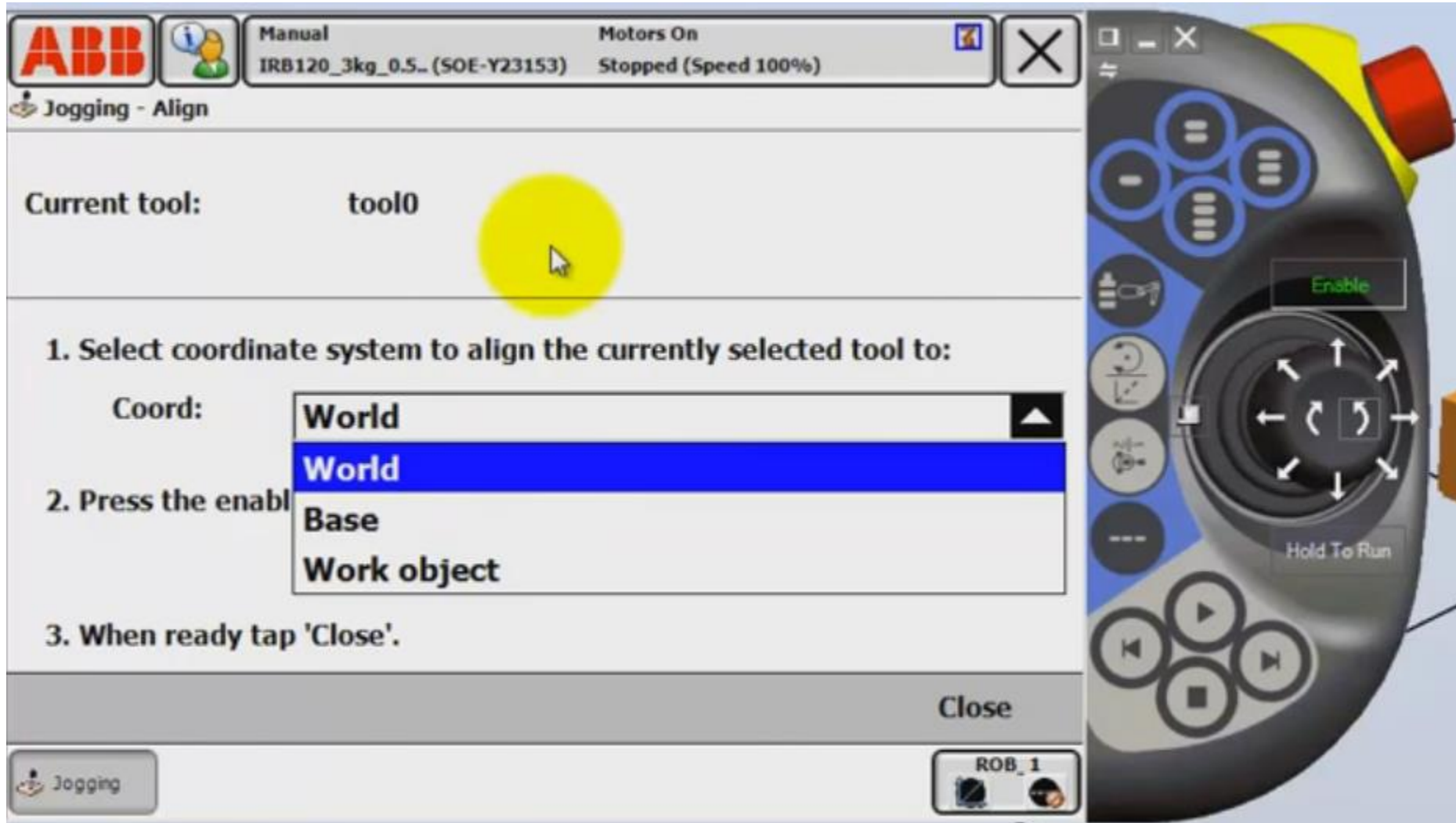
The velocity may be specified in the following units:

- **mm/s**
- **seconds** (time it takes to reach the next programmed position)
- **degrees/s** (for reorientation of the tool or for a rotation of an external axis)

# Coordinate System



# Coordinate System



# Defining The Work Object Coordinate System



Action	
1	On the ABB menu, tap <b>Jogging</b>
2	Tap <b>Work object</b> to display the list of available work objects.
3	Tap the work object you want to define, then tap <b>Edit</b> .
4	In the menu, tap <b>Define.....</b>
5	Select method from the <b>User method</b> and/or the <b>Object method</b> menu. See <a href="#">How to define the user frame on page 180</a> and <a href="#">How to define the object frame on page 181</a>

ABB Manual MySystem (IN-L-ITX1007121) Motors On Stopped (2 of 2) (Speed 100%)

Program Data - wobjdata - Define

**Work Object Frame Definition**

Work object: wobj1 Active tool: tool0

Select a method for each frame, modify the positions and tap OK.

User No Change Object No Change

Point	Status
-------	--------

Positions Modify Position OK Cancel

Jogging ROB\_1

en0400000893



# Instructions

File

Edit

View

IPL1

IPL2

Program Instr

EXERCISE/main

1 (4)

MoveL \*, v300, fine, tool0;

MoveL \*

,

v300

,

fine

,

tool0

MoveL \*, v300, fine, tool0;

MoveL \*, v300, fine, tool0;

Copy

Paste

OptArg...

ModPos

Test->

The name of the instruction which moves the robot linearly.

Hides the values of the instruction's position.

Determines the velocity of the robot.

Determines the precision of the robot's position.

Specifies which tool is active.



# Motion Instructions

---

**MoveL** \*, v500, z20/fine, tool0;

- ✓ TCP of the selected tool moves in a straight line from the initial position of the robot to the robot target.

**MoveJ** \*, v1000, z20/fine, cutter;

- ✓ TCP doesn't follow a straight line between initial position of robot and the robot target.

**MoveC** \*,\*, v1000, z100, welder;

- ✓ The TCP of the selected tool moves in a circular arc joining the initial TCP position to the two robot targets.

# BaseWare OS

---

The properties of BaseWare OS can be split up in five main areas:

- ✓ The Rapid Language and Environment;
- ✓ Exception handling;
- ✓ Motion Control;
- ✓ Safety;
- ✓ the I/O System.

**The Rapid Language and Environment :** It contains the following concepts:

- Hierarchical and modular program structure to support structured programming and reuse.
- Routines can be Functions or Procedures.
- Local or global data and routines.
- Data typing, including structured and array data types.
- User defined names (shop floor language) on variables, routines and I/O.
- Extensive program flow control. - Arithmetic and logical expressions.
- Interrupt handling.

# Programming :

---

Users do not need to remember the format of instructions, since they are prompted in plain English. “**See and pick**” is used instead of “**remember and type**”.

The programming environment can be easily customized using the **teach pendant**.

Programs, parts of programs and any modifications can be tested immediately without having to translate the program.

The program is **stored as a normal PC text file**, which means that it can be edited using a standard PC.

A robot position can easily be changed either by:

- jogging the robot with the joystick to a new position and then pressing the “ModPos” key (this registers the new position)

or by

- entering or modifying numeric values.

# MOTION CONTROL



# MOTION CONTROL



# MOTION CONTROL

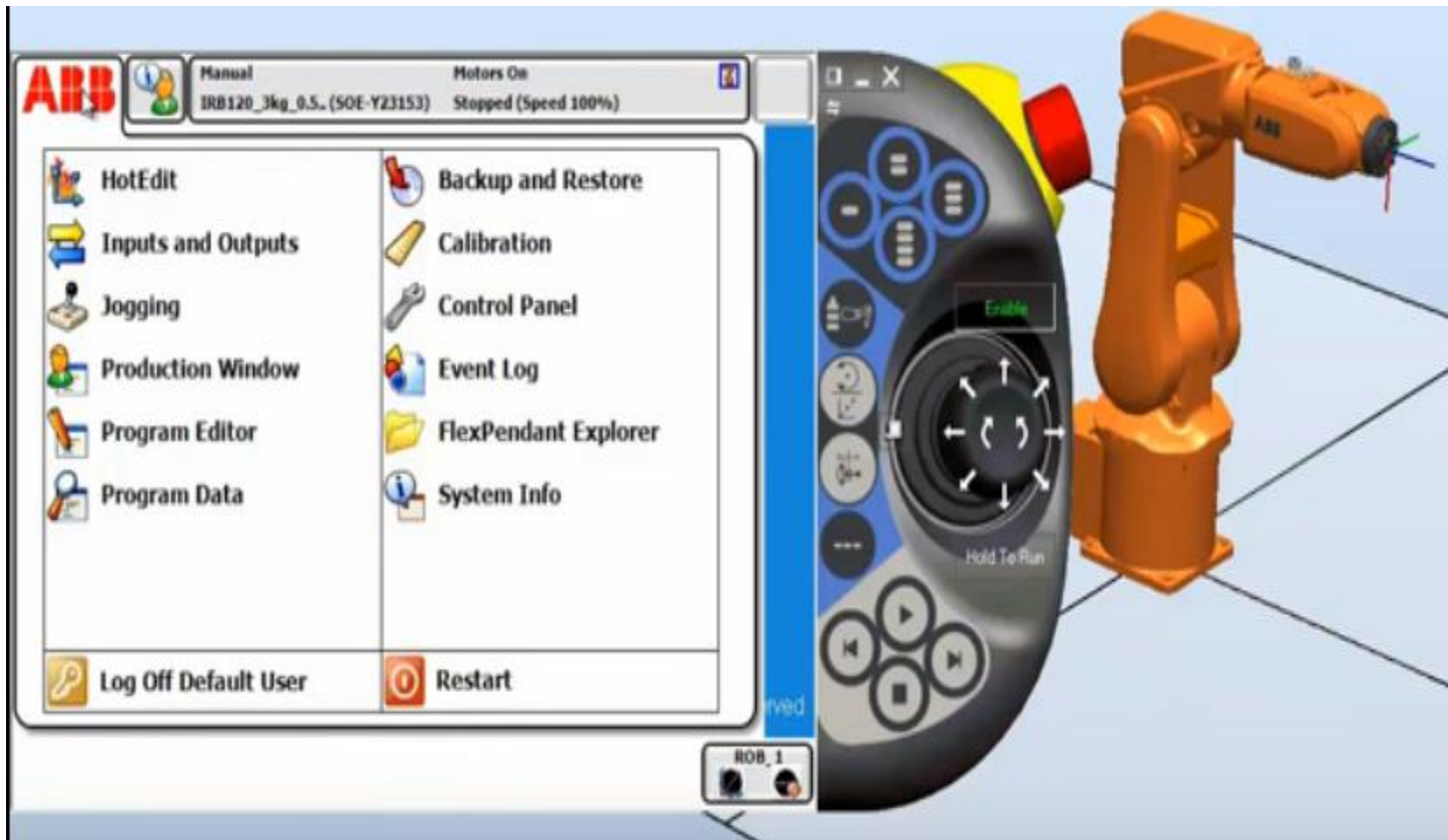




# MOTION CONTROL

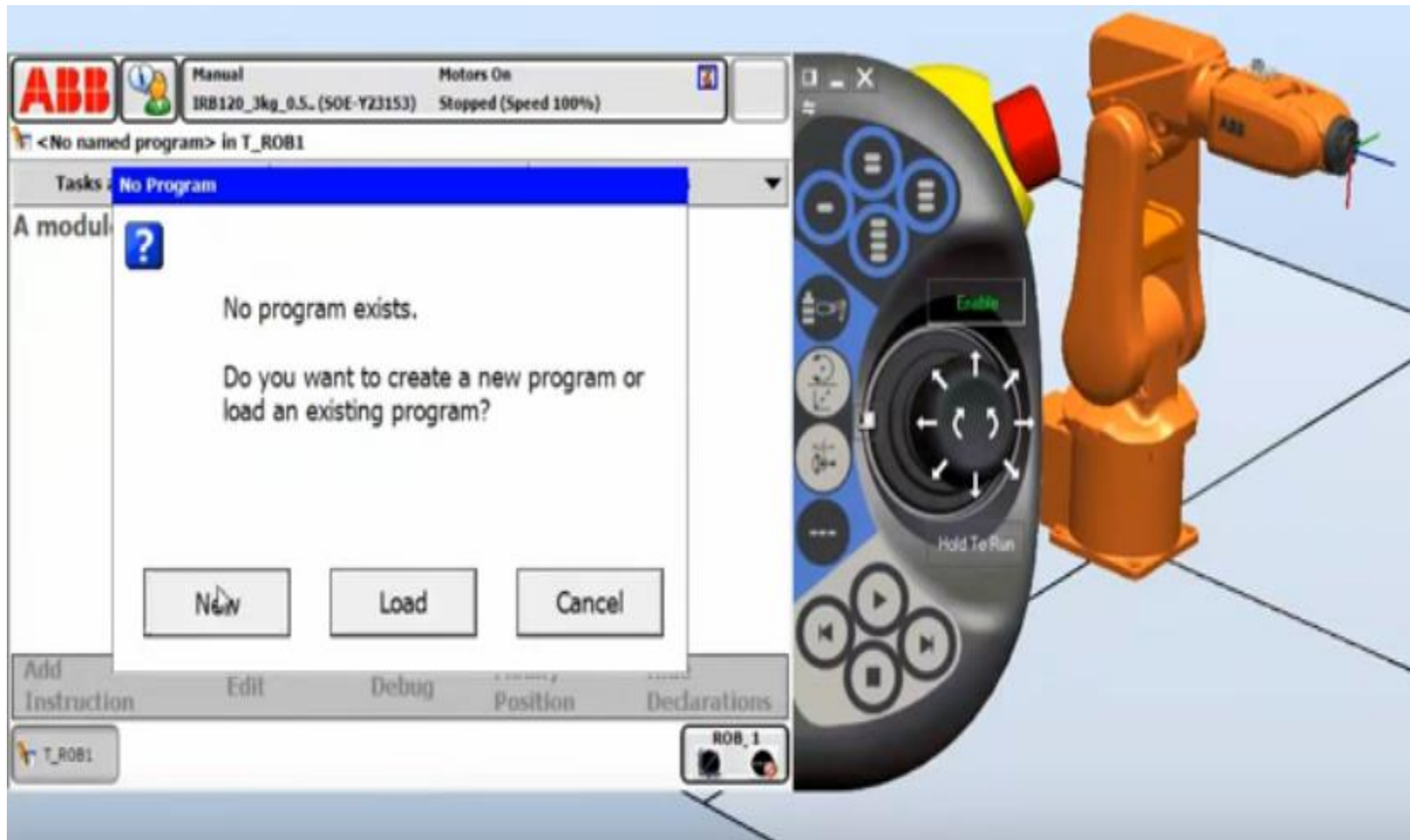


# MOTION CONTROL

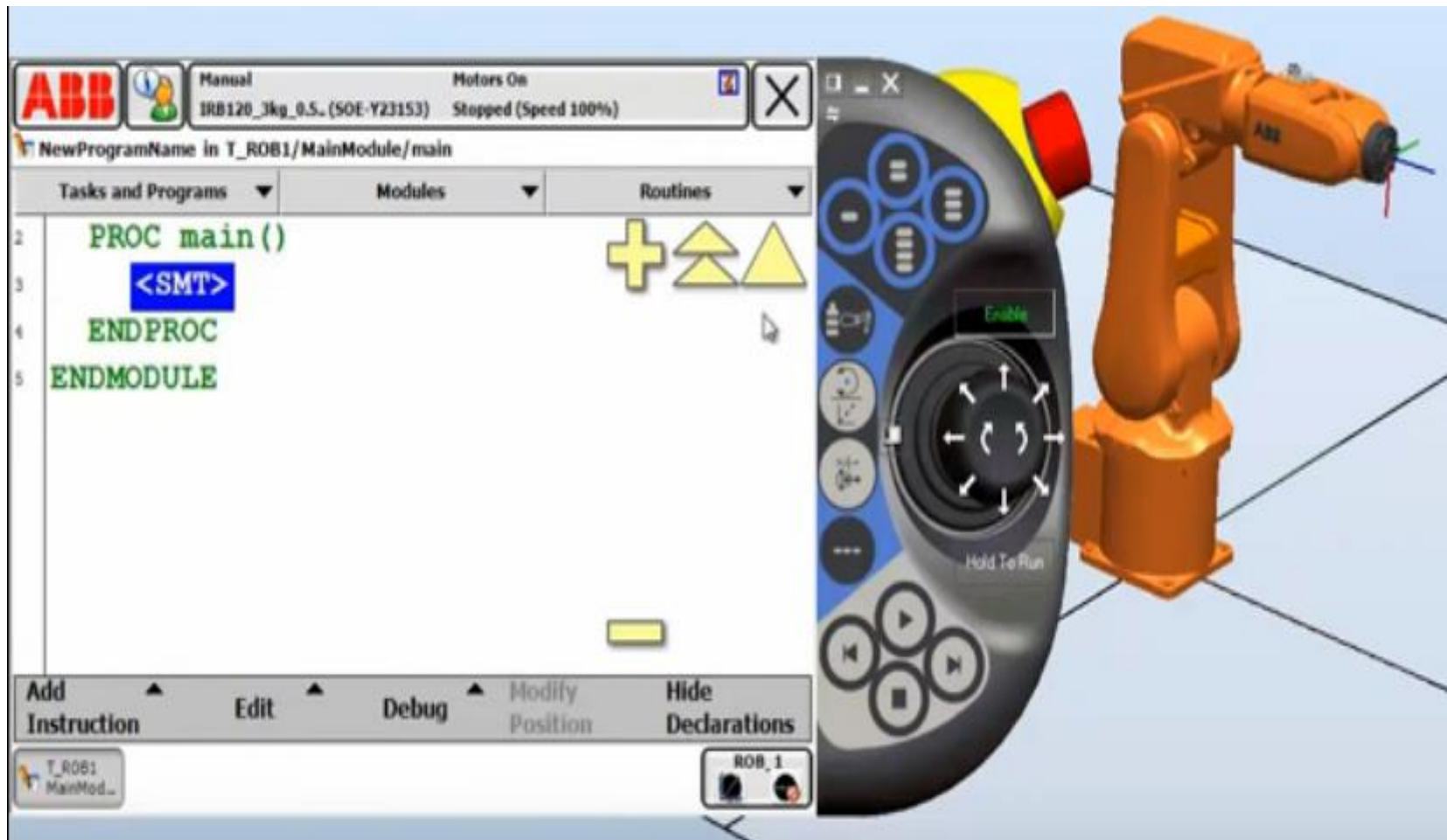




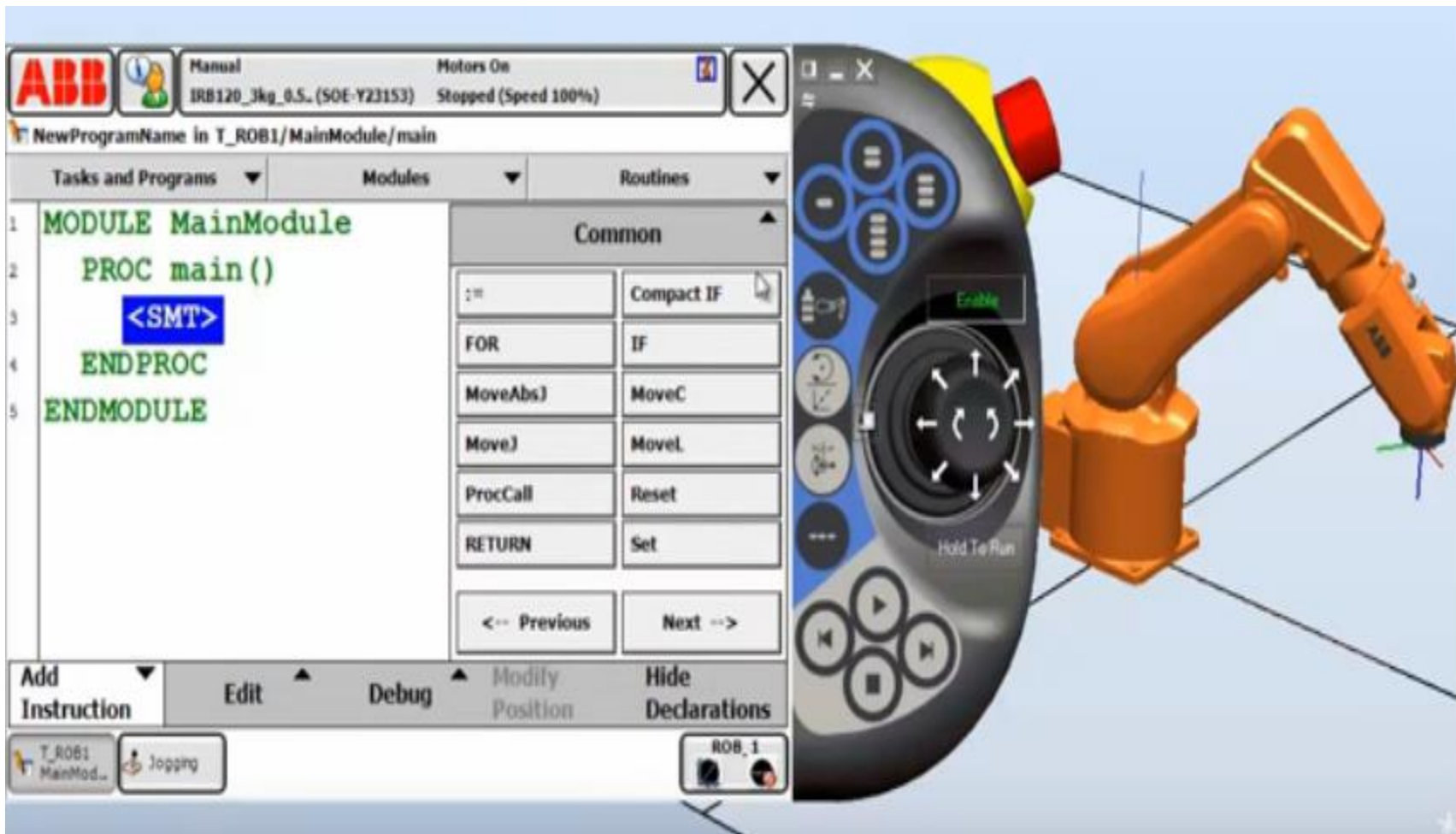
# MOTION CONTROL



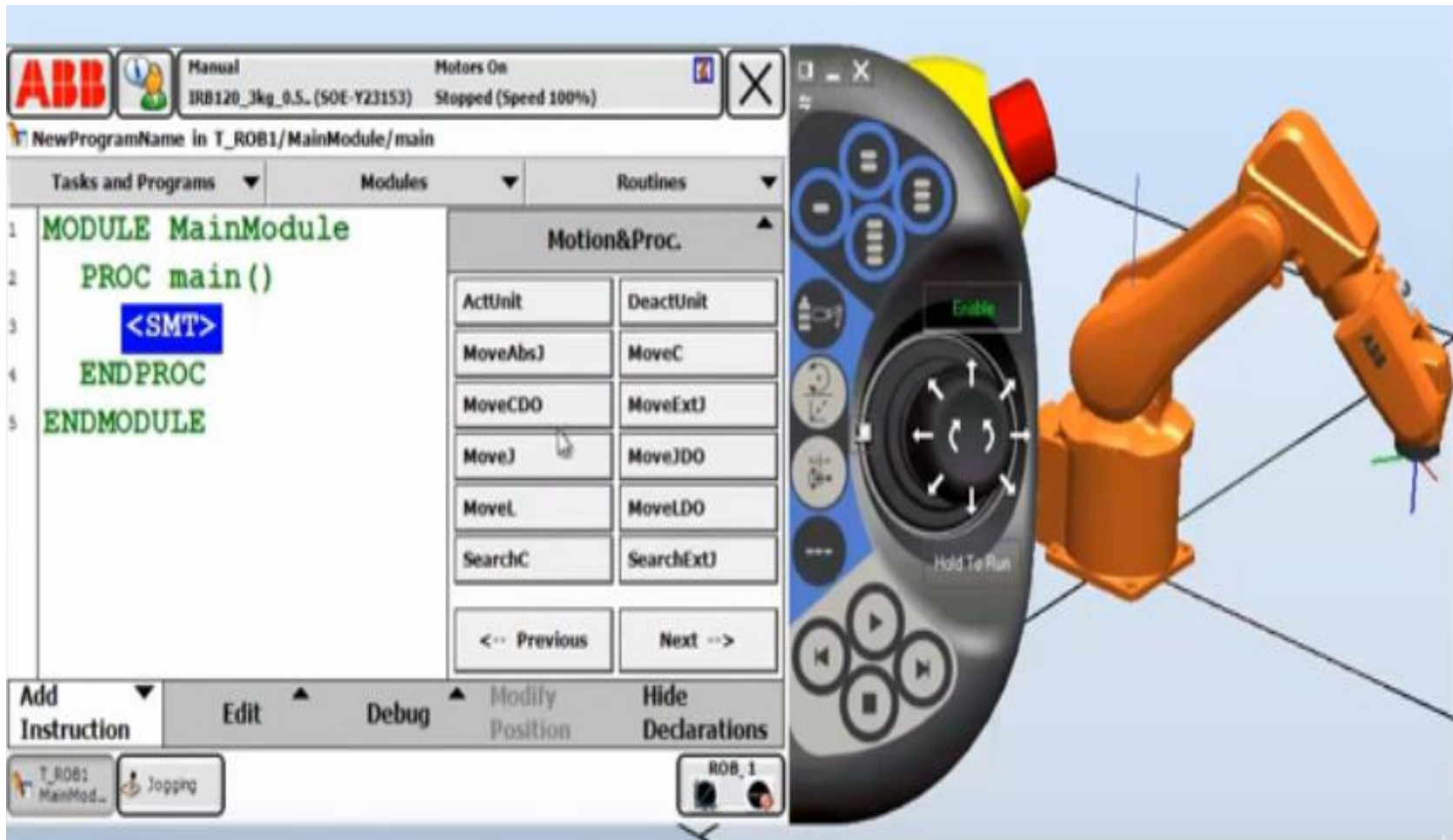
# MOTION CONTROL



# MOTION CONTROL

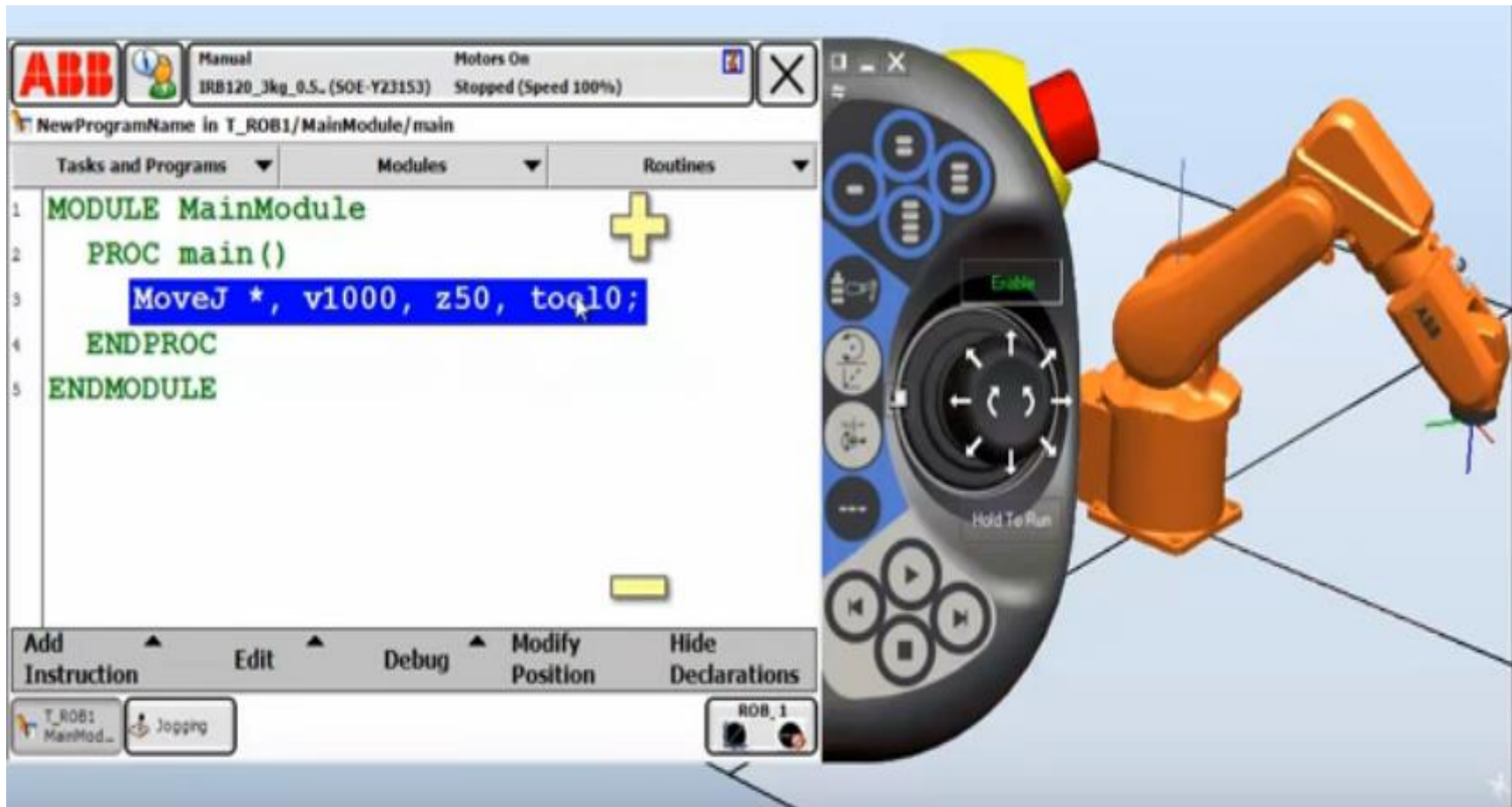


# MOTION CONTROL





# MOTION CONTROL



# MOTION CONTROL

