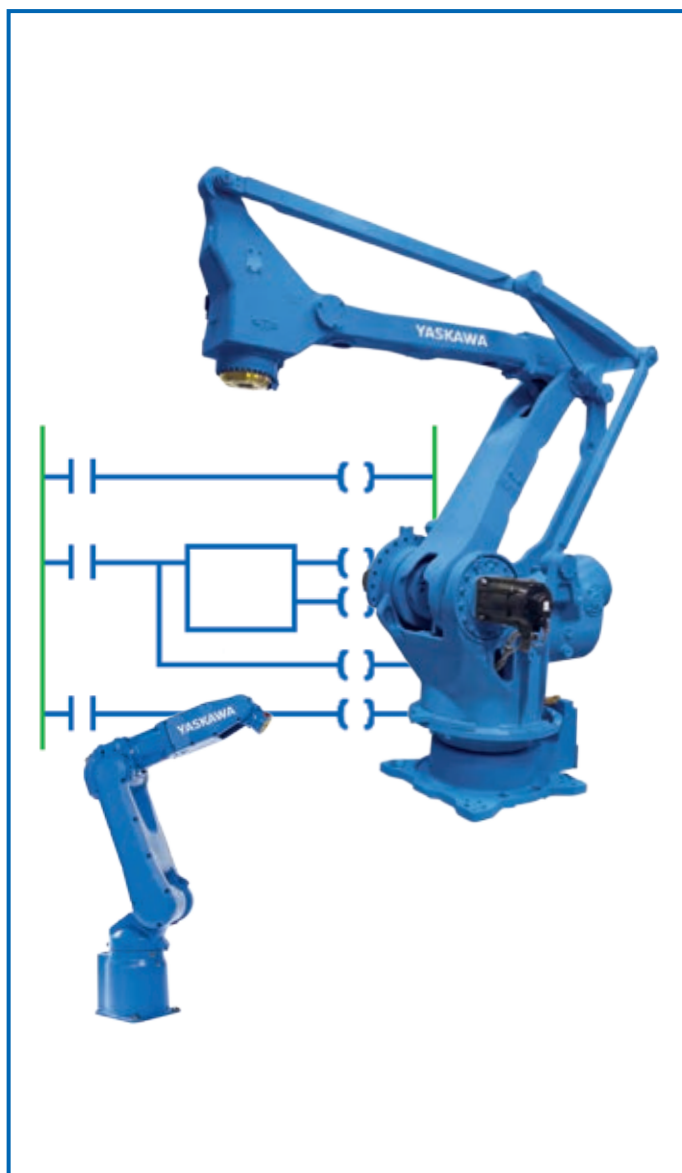


MotoLogix

Controlling software for MOTOMAN robots

MotoLogix is a revolutionary software interface for controlling YASKAWA robots by PLC. Being available for several major PLC brands and fieldbuses it is designed with two primary objectives:

- Enable OEM's to deeply integrate YASKAWA robot systems in their PLC controlled machinery.
- Easy programming/commissioning/teaching/operating of robots in a machine, without need of specialized knowledge.



MotoLogix has two components

1. MotoLogix Runtime – This enables the MotoLogix interface on the YASKAWA DX200 robot controller, using the fieldbus of your choice for communication with the PLC.
2. MotoLogix PLC Library – Comprehensive set of function blocks for writing your robot application logic in the PLC.

Key benefits

- Robot programming carried out in PLC language – unified for the whole system
- Easy to use library of function blocks and sample programs
- Connect all peripheral devices (sensor, camera, conveyor) through PLC
- Robot completely integrated in your PLC and HMI environment
- Test the complete PLC/HMI robot application using virtualization
- Assurance of a YASKAWA path quality and a smooth motion
- All YASKAWA DX200 robots can be controlled
- No Teach pendant nor YASKAWA robotics knowledge is required
- Data stored in the PLC, not in the robot controller
- Control up to 4 robots over one MotoLogix interface

MotoLogix

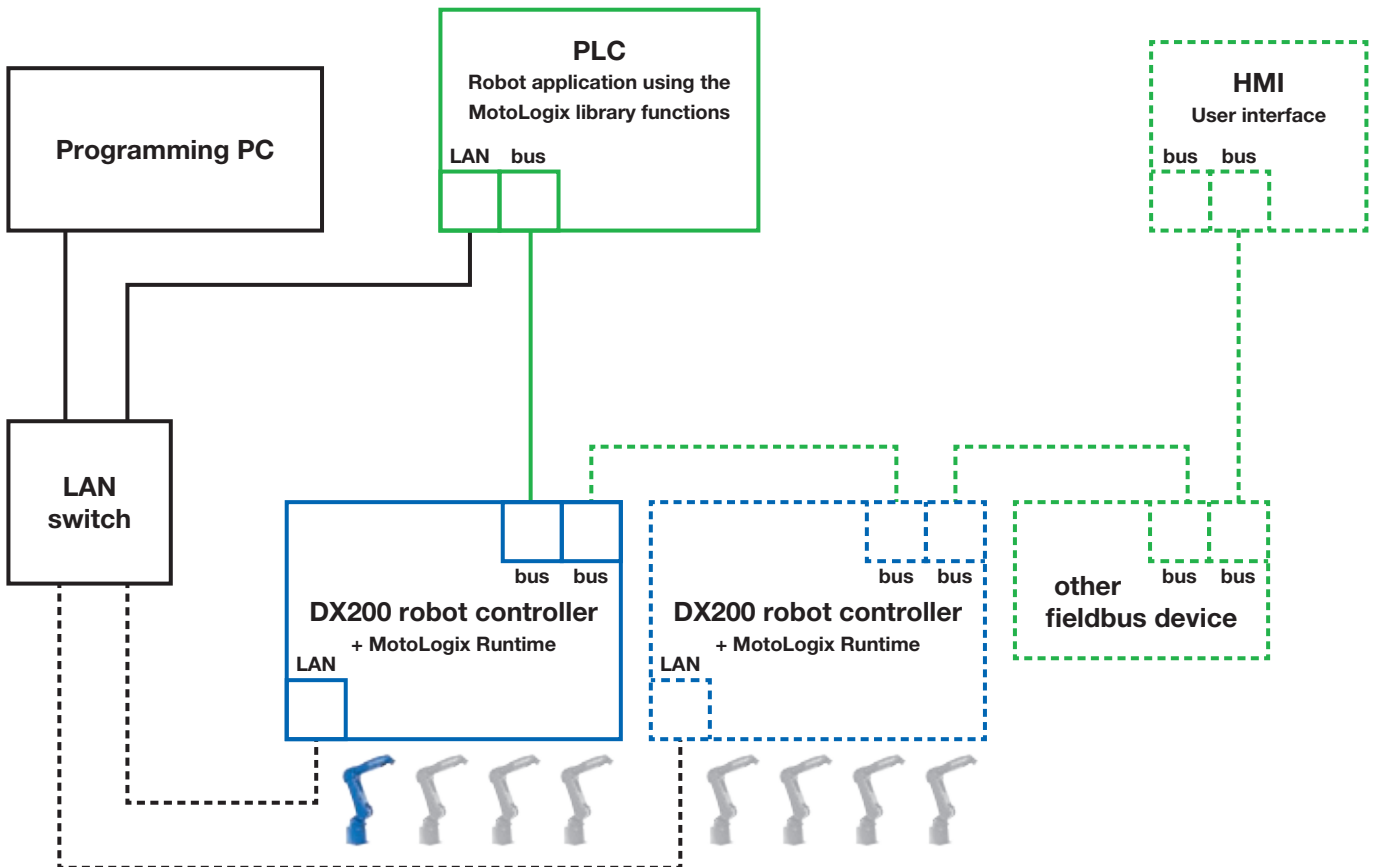
MotoLogix represents a software and hardware interface that enables users to control and program the robot through PLC and offers an innovative approach for a control of all-axis coordinated robot motion, similarly to traditional robot controller.

The difference between PLC controlled robot and conventional robot control is that PLC issues the motion commands for the

robot, while the robot controller performs calculations of motion kinematics. The DX200 robot controller is reduced to the role of a motion controller and the actual program execution and the definition of the motion are carried out by the PLC. This therefore eliminates the need to learn the robot language and allows the programmer to use the PLC language he already knows.

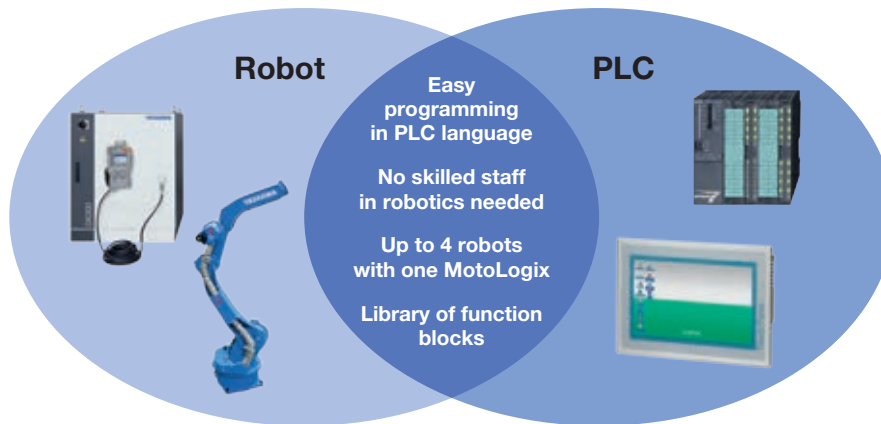
System layout

The picture below shows a typical system layout. The dashed lines are optional devices/connections.



Supported platforms





MotoLogix PLC library

The MotoLogix library offers a comprehensive set of function blocks for a wide range of tasks.

Summary:


- **Motion instructions**
 - Different kinds of moves
 - Jog
 - Conveyor tracking
- **System commands**
 - Enable, Abort, Hold etc
 - Error handling
 - IO handling
- **Robot configuration**
 - Tools, Userframes
 - Interference zones
 - Absolute data (home positions)

The built in documentation for each function block provides fast access to the information needed while programming.

MLxEnable

The MLxEnable instruction is used to enable the servos on all axes/robots and transition the system into Idle state. This must be called before motions can be commanded on the system.

MLxEnable Instruction

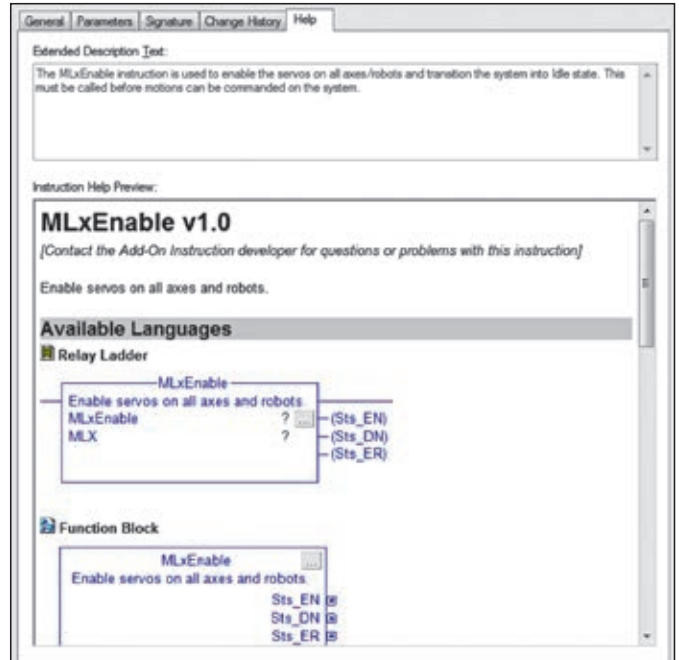


FB_Enable[0]

```

(* usage in ST, simplified syntax *)
//parameters
FB_Enable[0].Enable:= ;
//FB call
FB_Enable[0](MLX:= );
    
```

MLxEnable Instruction			
Name	Data Type	Usage	Description
Sts_EN	BOOL	Output	Enable bit. This bit will stay high as long as the instruction is enabled.
Sts_DN	BOOL	Output	Done bit. This bit will turn high when the instruction has finished.
Sts_ER	BOOL	Output	Error bit. Indicates an error during instruction execution.
MLX	MLxData	InOut	The MLxData Controller Scope tag.



General Parameters Signature Change History Help

Extended Description Text:
The MLxEnable instruction is used to enable the servos on all axes/robots and transition the system into Idle state. This must be called before motions can be commanded on the system.

Instruction Help Preview:

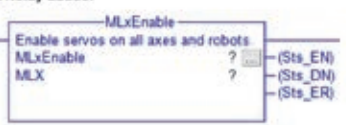
MLxEnable v1.0

[Contact the Add-On Instruction developer for questions or problems with this instruction]


Enable servos on all axes and robots.

Available Languages

Relay Ladder



Function Block



Get off to a quick start using the supplied example programs and documentation.

Examples	Examples for using the MotoLogix library
general	General logic
1_FBcalls	Example 1: Call of each individual FB
2_Jog	Example 2: Jog
3_PosTable	Example 3: Positioning by Table
4_ConveyorTracking	Example 4: Conveyor tracking (with objectqueue)

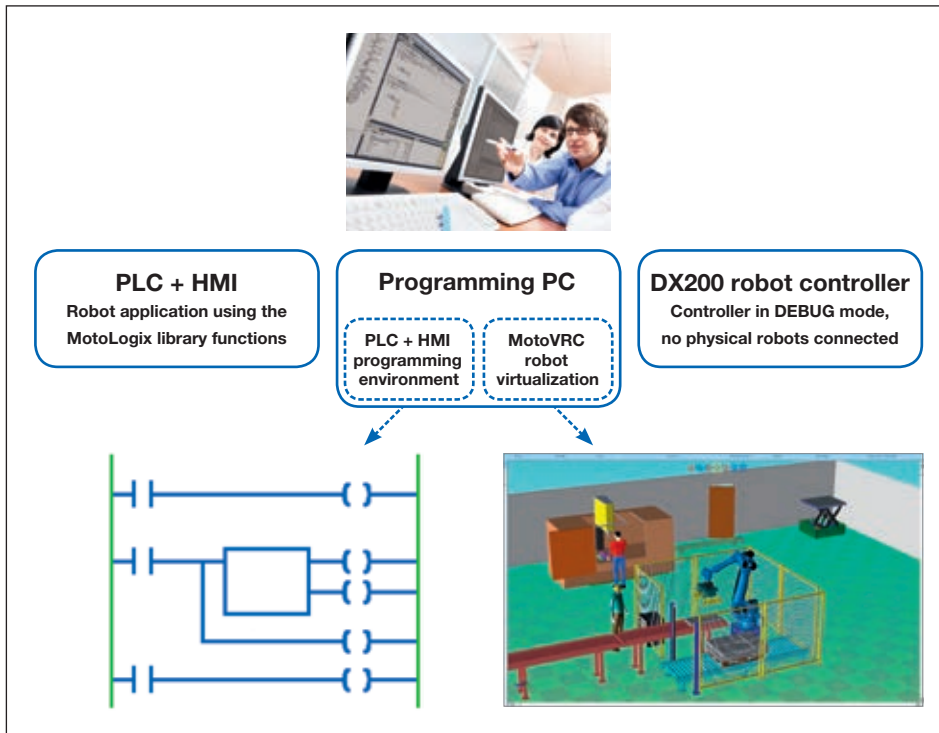
Virtualization

Using the powerful combination of MotoLogix and MotoVRC you can test your entire PLC/HMI robot application without the need of the completely assembled machine.*

* A YASKAWA robot controller is needed.

Benefits:

- Shorter commissioning time on-site
- Discover design- or application issues in an early stage to reduce the risk of endangering the deadline for your project
- Streamline your global manufacturing where engineering and assembly are located at different facilities



MotoLogix specifications	
Supported robots	All DX200 types
Number of robots	Up to 4 robots (or external axes) for each MotoLogix system
Number of MotoLogix systems per PLC	Only limited by PLC and fieldbus capacity
Number of motions, userframes, tools	Only limited by PLC memory*
Number of interference zones	32
Number of conveyors for Conveyor tracking	Only limited by PLC hardware and memory
Robot controller cycle time	4 ms
Data exchange for one MotoLogix system	436 byte consistent data is cyclically exchanged between PLC and each MotoLogix system
Required available PLC memory	> 512 kb (depends on complexity of application)

* If the DX200 is equipped with a Functional Safety Unit (FSU) the amount of tools is limited to 16.



1300 552 333
www.ragroup.com.au



Please request detailed drawings at
 robotics@yaskawa.eu.com –
 MotoLogix, A-09-2015, A-No. 175674