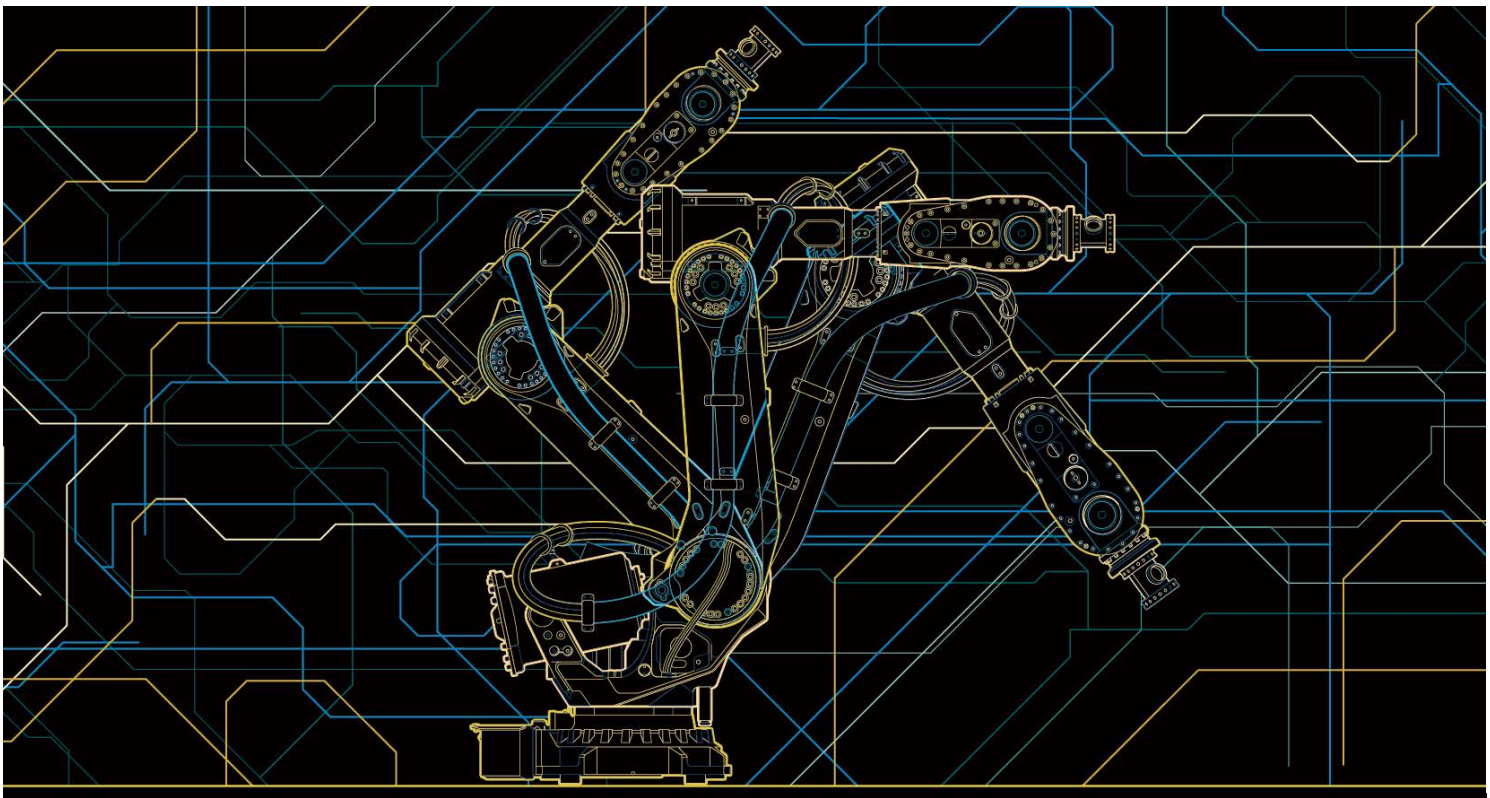


Product specification
Controller software IRC5

Power and productivity
for a better world™

Trace back information:
Workspace R15-1 version a3
Checked in 2015-03-25
Skribenta version 4.1.349

Product specification
Controller software IRC5
RobotWare 6.01

Document ID: 3HAC050945-001
Revision: A

ABB AB
Robotics Products
Se-721 68 Västerås
Sweden

Table of contents

Overview of this specification 9

1 Introduction to RobotWare 11

2 Option restructuring 13

3 RobotWare-OS 15

3.1 Multiple Axis Positioner 15
3.2 Fixed Position Events 16
3.3 File and Serial Channel Handling 18
3.4 Advanced RAPID 20
3.5 Auto acknowledge input 23
3.6 Logical Cross Connections 24
3.7 Analog Signal Interrupt 25
3.8 Electronically Linked Motors 26
3.9 Service Information System 27

4 General RobotWare 29

4.1 RobotWare Add-In prepared [988-1] 29

5 Motion Performance 31

5.1 Advanced robot motion [687-1] 31
5.2 Advanced Shape Tuning 32
5.3 WristMove 34
5.4 Absolute Accuracy, floor mounted [603-1] 36
5.5 Absolute Accuracy, inverted [603-2] 39
5.6 Motion Process Mode 40

6 Motion Coordination 41

6.1 MultiMove Coordinated [604-1] 41
6.2 MultiMove Independent [604-2] 44
6.3 Conveyor Tracking [606-1] 46
6.4 Indexing Conveyor Control [606-2] 48
6.5 Sensor Synchronization [607-1] 50
6.6 Analog Synchronization [607-2] 51

7 Motion Events 53

7.1 World Zones [608-1] 53

8 Motion Functions 55

8.1 Independent Axis [610-1] 55
8.2 Path Recovery [611-1] 57
8.3 Path Offset [612-1] 58
8.4 SoftMove [885-1] 59

9 Motion Supervision 61

9.1 Collision Detection [613-1] 61

10 Communication 63

10.1 FTP Client [614-1] 63
10.2 NFS Client 65
10.3 PC Interface [616-1] 66
10.4 FlexPendant Interface [617-1] 67
10.5 Field bus Command Interface [618-1] 68

10.6 RobotStudio App Connect [688-1] 69
10.7 Socket Messaging 71

11 Engineering Tools **73**

11.1 Multitasking [623-1] 73
11.2 Continuous Application Platform [624-1] 75
11.3 Optical Tracking [813-1] 76
11.4 Discrete Application Platform [625-1] 77
11.5 Sensor Interface [628-1] 78
11.6 Robot Reference Interface 80
11.7 Externally Guided Motion [689-1] 82
11.8 MultiFunction [824-1] 84
11.9 Production Screen [637-1] 85
11.10 RAPID Message Queue 86
11.11 Production Manager [812-1] 88

12 Servo Motor Control **89**

12.1 Servo Tool Control 89
12.2 Servo Tool Change [630-1] 91

13 Vision **93**

13.1 Integrated Vision 93

14 Application options **95**

14.1 MultiProcess [634-1] 95
14.2 Arc [633-4] 96
14.3 Production Manager for Arc 99
14.4 Arc options 103
14.4.1 RobotWare Power Source 104
14.4.1.1 Standard I/O Welder [650-7] 105
14.4.1.2 Fronius TPS 4000/5000 [650-9] 106
14.4.1.3 AristoMig integrated [650-10] 108
14.4.1.4 ABB RPC S Integrated [650-11] 109
14.4.1.5 Fronius TPS Integrated [650-12] 110
14.4.1.6 Lincoln ArcLink [650-13] 111
14.4.2 Additional Arc System 114
14.4.2.1 Two additional [651-2] 114
14.4.3 BullsEye 115
14.4.3.1 BullsEye [652-1] 116
14.4.4 Torch Service Equipment 117
14.4.4.1 ABB - TC 2013 [653-1] 118
14.4.4.2 Binzel - TC97 [653-2] 119
14.4.5 SmarTac - I/O version [657-1] 120
14.4.6 Production Monitoring [659-1] 122
14.4.7 Navigator [814-1] 127
14.4.8 Optical Tracking Arc [660-1] 130
14.4.9 WeldGuide [815-2] 131
14.5 Spot 6 [635-6] 135
14.6 Bosch Interface [832-1] 138
14.7 Dispense [641-1] 139
14.8 Prepared for PickMaster 141
14.8.1 PickMaster 3 [642-1] 141
14.8.2 PickMaster 5 [642-2] 143
14.9 Force Control Base [661-2] 145
14.10 Machining FC GUI [877-1] 152
14.11 RobotWare Cutting [951-1] 154
14.12 RobotWare Machine Tending [1167-1] 157

Index **161**

This page is intentionally left blank

Overview of this specification

About this product specification

It describes all RobotWare (that is controller software) options for the IRC5 controller

Usage

Product specifications are used to find data and performance about the product, for example to decide which product to buy. How to handle the product is described in the product manual.

Users

It is intended for:

- Personnel dealing with ordering of ABB robots
- Personnel seeking to obtain an overview of RobotWare functionality
- Sales and marketing personnel

References

Reference	Document ID
Product specification - Controller IRC5	*3HAC047400-001*
Product specification - Integrated Vision	*3HAC046868-001*
Application manual - Controller software IRC5	*3HAC050798-001*
Application manual - Continuous Application Platform	*3HAC050990-001*
Application manual - Discrete application platform	*3HAC050994-001*
Application manual - Additional axes and stand alone controller	*3HAC051016-001*
Product specification - Robot user documentation, IRC5 with RobotWare 6	*3HAC052355-001*
Application manual - RobotWare Add-Ins	*3HAC051193-001*

Revisions

Revision	Description
-	First revision
A	Released with RobotWare 6.01. • Added the option *RobotWare Add-In prepared [988-1] on page 29*. • The functionality of RAPID Message Queue is corrected, see *RAPID Message Queue on page 86*. • The option Miller AutoAxcess [650-4] is removed. • The previous option Servo Tool Control is now included in the option Spot 6, see *Servo Tool Control on page 89*, • Minor corrections/update

This page is intentionally left blank

1 Introduction to RobotWare

Software products

RobotWare is a family of software products from ABB Robotics. The products are designed to make you more productive and lower your cost of owning and operating a robot. ABB Robotics has invested many years into the development of these products and they represent knowledge and experience based on several thousands of robot installations.

Product classes

Within the RobotWare family, there are different classes of products:

Product classes	Description
RobotWare-OS	This is the operating system of the robot. RobotWare-OS provides all the necessary features for fundamental robot programming and operation. It is an inherent part of the robot, but can be provided separately for upgrading purposes. For a description of RobotWare-OS, see *Product specification - Controller IRC5*.
RobotWare options	These products are options that run on top of RobotWare-OS. They are intended for robot users that need additional functionality for motion control, communication, system engineering, or applications.
Process application options	These are extensive packages for specific process application like spot welding, arc welding, and dispensing. They are primarily designed to improve the process result and to simplify installation and programming of the application.
RobotWare Add-ins	A RobotWare Add-in is a self-contained package that extends the functionality of the robot system. Some software products from ABB Robotics are delivered as Add-ins. For example track motion IRBT, positioner IRBP, and stand alone controller. The purpose of RobotWare Add-ins is also that a robot program developer outside of ABB can create options for the ABB robot systems, and sell the options to their customers. For more information on creating RobotWare Add-ins, contact your local ABB Robotics representative at www.abb.com/contacts.

Process application options

For IRC5, the former *ProcessWare* options are included in the RobotWare options. These are extensive packages for specific process application like welding, dispensing and painting. They are primarily designed to improve the process result and to simplify installation and programming of application.

Option groups

For IRC5, the RobotWare options have been gathered in groups, depending on the customer benefit. The goal is to make it easier to understand the customer value of the options. However, all options are purchased individually. The groups are as follows:

Option groups	Description
Motion performance	Options that optimize the performance of your robot.

Continues on next page

Continued

Option groups	Description
Motion coordination	Options that make your robot coordinated with external equipment or other robots.
Motion Events	Options that supervise the position of the robot.
Motion functions	Options that control the path of the robot.
Motion Supervision	Options that supervise the movement of the robot.
Communication	Options that make the robot communicate with other equipment. (External PCs etc.)
Engineering tools	Options for the advanced robot integrator.
Servo motor control	Options that make the robot controller operate external motors, independent of the robot.

2 Option restructuring

General

A change in the option structure has been implemented, in order to achieve a simplified product offer. This means that a number of options have been removed from the specification form. The corresponding functionality has been either included in the basic robot product or merged with other options.

In this product specification, all options are still described as before. A comment has been added in the beginning of each chapter affected by the change, to indicate that options have been included in the base product or merged with another option.

This page is intentionally left blank

3 RobotWare-OS

3.1 Multiple Axis Positioner

General

Note

This functionality is included in RobotWare - OS.

The option *Multiple Axis Positioner* enables coordination of robot motion with multiple axis manipulators or robot carriers (gantries).

Note

Note that simultaneous coordination with several single axis manipulators, for example track motion and work piece manipulators, does not require the option *Multiple Axis Positioner*.

Features

- Coordinated movement of robot and multiple axis manipulator

Application

This option shall be used for all types of multiple axis manipulators for example positioners for arc welding.

The kinematic model of the positioner enables the coordinated movement of robot and manipulator together, meaning correct TCP movement relative to the work piece, also when the work piece or the robot (for robot carrier Application) is moved around, during program execution or jogging.

Performance

The performance of the coordinated robot movement with a moving work object on a multiple axis manipulator is the same as for a fix work object, if the manipulator is correctly calibrated.

Requirements

Note

A configuration file describing the kinematics of the manipulator is necessary and is normally supplied by the manipulator supplier.

RAPID instructions

There are no RAPID instructions included in this option.

3.2 Fixed Position Events

General

Note

This functionality is included in RobotWare - OS.

The option *Fixed Position Events* is used to issue certain events depending on the current robot position. The events can be used to control or check the status of surrounding equipment.

Features

- Change the value of an I/O signal, when the TCP is at a certain time and/or distance before or after a programmed position.
- Generate an interrupt, when the TCP is at a certain time and/or distance before or after a programmed position.
- Check the value of an I/O signal, when the TCP is at a certain time and/or distance before or after a programmed position.
- Make a procedure call, when the TCP is at a certain position on the path or in the middle of a corner zone.

Application

Application	Description
Handling press work	To provide a safe communication system between the robot and the press and to reduce cycle time. At the instant when the robot leaves a press, an output is set and restarts the press action. This function is also useful for other process equipment. The start/stop will always occur when the robot is at the exact position, irrespective of the robot speed.
Check status of process equipment	For example a robot which is used for extracting parts from a die casting machine. Before entering, the robot can check if the gate is open (check an I/O signal) or check a number of logical conditions and take care of the complete press start (make procedure call).

Performance

The event issued with *Fixed Position Event* will always occur when the robot is at the exact position, irrespective of the robot speed.

Requirements

There are no software or hardware requirements for this option.

RAPID instructions

RAPID instructions included in this option:

Instruction	Instruction
TriggIO	Definition of trigg conditions for an output
TriggEquip	Definition of trigg conditions for process equipment with compensation for equipment delay

Continues on next page

Instruction	Instruction
TriggCheckIO	Definition of trigg conditions for check of signal value
TriggInt	Definition of trigg conditions for an interrupt
TriggL	Position fix output/interrupt during linear movement
TriggC	Position fix output/interrupt during circular movement
TriggJ	Position fix output/interrupt during joint movement
MoveLSync	Procedure call in the middle of the path or in corner zone, during linear movement
MoveCSync	Procedure call in the middle of the path or in corner zone, during circular movement
MoveJSync	Procedure call in the middle of the path or in corner zone, during joint movement

3.3 File and Serial Channel Handling

General

Note

This functionality is included in RobotWare - OS.

File and Serial Channel Handling is an option, which allows the robot system to communicate with external units.

Features

- Transferring information via serial channels.
- Read part numbers from a bar code reader
- Print out production statistics on a printer during production
- Transfer data between the robot and a PC
- Data transfer via files
- Write/read production data on a USB memory stick or other mass storage memory from RAPID program

Application

Application	Description
Transferring information via serial channels	Bar code readers can be used to trace a product with its corresponding production information, for every work object throughout a production line. Bar code readers can also be used to make the robot perform the proper actions, corresponding to the work object, in lines which handle different types of products, for example in packing and pelletizing Application. This is the same as controlling the robot production from a file. This file may have been created in a PC, stored on a USB memory stick, and read by the robot at a later time.
Data transfer via files	Storing production statistics on a USB memory stick or other mass storage memory. This information can then be processed by an ordinary PC.

Performance

Data/Information	Description
Transferring information via serial channels	The transfer is controlled entirely from the robot's work program. To control the transfer from a PC, use the option *PC Interface*.
Data transfer via files	Data in the form of text strings (characters), numerical values or binary information can be read/written.

Requirements

This option includes software functionality only. Serial channels (RS232 or RS 485 serial channel), bar code readers etc. need to be purchased separately, from ABB or external provider.

Continues on next page

RAPID instructions

RAPID instructions included in this option:

Instruction	Description
Open/Close	Open/Close a file/serial channel
Write	(Write/WriteBin/WriteStrBin/WriteAnyBin)Write to a character- or string-based/binary serial channel or file.
Read	(Read/ReadNumReadStr/ReadBin/Read-StrBin/ReadAnyBin)Read a string/number/binary value from a serial channel or file.
FSSize	Get the size of a file system
MakeDir	Create a new directory
RemoveDir	Delete a directory
OpenDir	Open a directory to read the underlying files or subordinates
CloseDir	Close a directory
ReadDir	Read next object in a directory, file or subdirectory
IsFile	Check the type of a file
FileSize	Get the size of a file
CopyFile	Copy a file, from RAPID
RenameFile	Rename a file from RAPID
RemoveFile	Delete a file
Rewind	Start reading from the beginning of a file
ClearIOBuff	Clear the input buffer of a serial channel
ReadRawBytes	Read data from raw byte
WriteRawBytes	Write data raw byte to a device
ClearRawBytes	Clear all contents of a rawbytes variable
CopyRawBytes	Copy raw byte data
PackRawBytes	Pack data from variables into rawbytes
UnpackRawBytes	Unpack data from rawbytes into variables
RawBytesLen	Returns the amount of data in a ‘container’ of type raw byte (bytes)

Data types

Datatype	Description
rawbytes	A general data ‘container’, for communication with I/O devices

3.4 Advanced RAPID

General

> **Note**
>
> This functionality is included in RobotWare-OS.

The option *Advanced RAPID* is directed towards advanced RAPID programmers. The package includes a detailed reference manual on the RAPID language kernel and a number of instruction and function groups useful for application development, as listed below.

The groups are:

- Bit Functions
- Data Search Functions
- RAPID Support Functions
- Power Failure Functions
- Advanced Trigg Functions

Technical reference manual - RAPID kernel

The manual describes the RAPID language syntax and semantics in detail concerning the kernel, that is all general language elements which are not used to control robot or other equipment. In addition to this the manual includes descriptions on:

- Built-in Routines
- Built-in Data Objects
- Built-in Objects
- Intertask Objects
- Text Files
- Storage allocation for RAPID objects

Bit Functions

Bit functions is a package for handling, that is setting, reading and clearing, individual bits in a byte. The instructions/functions are:

Instructions/Functions	Description
byte	Data type for a byte data
BitSet	Set a specified bit in a byte
BitClear	Clear a specified bit in a byte
BitCheck	Check if a specified bit in a byte is set
BitAnd	Logical bitwise AND operation on byte
BitOr	Logical bitwise OR operation on byte
BitXOr	Logical bitwise XOR operation on byte
BitNeg	Logical bitwise NEGATION operation on byte

Continues on next page

Instructions/Functions	Description
BitLSh	Logical bitwise LEFT SHIFT operation on byte
BitRSh	Logical bitwise RIGHT SHIFT operation on byte

Data Search Functions

With these functions it is possible to search all data in a RAPID program, where the name or the data type is given as a text string. This might be useful for instance in the following examples:

- A common need is to check if a data with a certain name is declared in the system, and in such case what is its value, for example a robtarget.
- Another need is to list all variables of a certain datatype, which are declared in the system, and write their values on the screen, for example all weld data.

The following instructions/functions are included in the data search functions:

Instructions/Functions	Description
SetDataSearch	Define the search criteria
GetNextSym	Search next data and get its name as a string
GetDataVal	Get the value of a data, specified with a string for the name
SetDataVal	Set the value of a data, specified with a string for the name
SetAllDataVal	Set the value of all searched data

RAPID Support Functions

This package includes a number of miscellaneous instructions etc., which are used in application development.

Instruction	Description
AliasIO	Instruction used to define a signal of any type with an alias (alternative) name. The instruction can be used to make generic modules work together with site specific I/O, without changing the program code.
ArgName	Function used inside a routine to get the name of a data object, which is referenced as argument in the call of the routine. The name is given as a string. The function can also be used to convert the identifier of a data into a string.
BookErrNo	Instruction used to book a new RAPID system error number. This should be used to avoid error number conflicts if different generic modules are combined in a system.
ErrLog	Write a system error message.
ErrRaise	Write a system error message and RAISE the error to the calling routine.
TextTabGet	Function used to get the text table number of a user defined text table during runtime.
TextGet	Function used to get a text string from the system text tables (installed at cold start).
TextTabInstall	Instruction used to install a text table in the system.
TextTabFreeToUse	Function to test whether the text table name (text resource string) is free to use.

Continues on next page

Instruction	Description
SetSysData	Instruction which will activates the specified system data (tool or workobject). With this instruction it is possible to change the current active tool or workobject.
IsStopStateEvent	Function which will return information about the movement of the Program Pointer (PP).
ReadCfgData	Read system configuration data.
WriteCfgData	Write system configuration data.
WarmStart	Restart the system.

Power Failure Functions

The package is used to get I/O signal values before power failure and to reset them at power on. The following instructions are included and are normally used in the power on event routine:

Instruction	Description
PFRestart	Check if path has been interrupted

Advanced Trigg Functions

Instruction	Description
TriggSpeed	Instruction to define conditions and actions for control of an analog output signal with an output value proportional to the actual TCP speed.Note that this instruction must be used in combination with a TriggL/C/J instruction.
StepBwdPath	Instruction used to move backward on its path in a RESTART event routine.
TriggStopProc	Generation of restart data at program stop or emergency stop.
IPers	An interrupt when changing a persistent.
IError	An interrupt at event (error) generation.
GetTrapData	Used in a trap routine to obtain all information about the interrupt that caused the trap routine to be executed.
ReadErrData	Used in a trap routine to obtain numeric information (domain, type and number) about an error, a state change, or a warning, that caused the trap routine to be executed.

3.5 Auto acknowledge input

General

Note

This functionality is included in RobotWare-OS.

Auto Acknowledge Input is a system input which will acknowledge the dialog presented on the FlexPendant when switching from operator mode manual to auto with the key switch on the robot controller.

WARNING

Note that using such an input will be contrary to the regulations in the safety standard ISO 10218-1 chapter 5.3.5 Single point of control with following text:

"The robot control system shall be designed and constructed so that when the robot is placed under local pendant control or other teaching device control, initiation of robot motion or change of local control selection from any other source shall be prevented."

Thus it is absolutely necessary to use other means of safety to maintain the requirements of the standard and the machinery directive and also to make a risk assessment of the completed cell. Such additional arrangements and risk assessment is the responsibility of the system integrator and the system must not be put into service until these actions have been completed

Features

An optional system input can be created, which will acknowledge the dialog presented at the FlexPendant when switching from operator mode manual to auto. The option must be activated in Installation Manager and then the system input defined in the configuration file for I/O.

Limitations

The system parameter cannot be defined using the FlexPendant or RobotStudio, only with a text string in the I/O configuration file.

3.6 Logical Cross Connections

General

Note

This functionality is included in RobotWare-OS.

The option *Logical Cross Connections* can be used to check or control process equipment, which is external to the robot. The functionality can be compared to the one of a simple PLC.

Features

- Boolean values (true/fault) based on the logical conditions: AND, OR, NOT

Application

Any application where logical conditions are used for digital signals.

Application	Description
Program execution	To be interrupted when both inputs 3 and 4 are ‘high’.
Register is to be incremented	When input 5 is set, but only when output 5=1 and input 3 = 0.

Performance

Maximum of 100 cross connections can be configured.

Requirements

There are no software or hardware requirements for this option.

RAPID instructions

There are no RAPID instructions included in this option.

3.7 Analog Signal Interrupt

General

Note

This functionality is included in RobotWare-OS.

The option *Analog Signal Interrupt* can be used to generate a program interrupt, when a supervised analog signal reaches a predefined limit.

The interrupt can be used to give an error message for example ‘temperature above limit’, or make the robot wait for a door to be opened.

Features

- Supervision of analog signals

Application

Supervision of external equipment, such as temperature sensors and equipment doors. In the later case, the Analog Signal Function can be used to minimize cycle time of the cell, since the robot can enter an area, which is enclosed by a door, at an optimal moment.

Performance

Analog Signal Interrupt requires less computer capacity than handshaking methods.

Requirements

There are no software or hardware requirements for this option.

RAPID instructions

RAPID instructions included in this option:

Instruction	Description
ISignalAI	Interrupt from analog input signal
ISignalAO	Interrupt from analog output signal

3.8 Electronically Linked Motors

General

Note

This functionality is included in RobotWare-OS.

Electronically Linked Motors is used to make master/slave configurations of motors, which are defined as external axes. The main application is to replace mechanical driving shafts of gantry machines, but the option can be used to control any other set of motors as well. In addition to plain position following, also a torque/slave function is available, where the combined torque will be distributed between master and slaves.

Features

- Up to 4 master motors
- Up to 11 motors total (masters and followers)
- Jogging and calibration routines
- Replacement of mechanical driving shafts
- Arm/Motor position available on the TPU
- Possibility to activate/deactivate link during process
- Automatic calibration at startup

Application

Gantry machines: to replace mechanical driving shafts.

Performance

Performance	Description
When jogging	the electronically linked motors will follow the master motor
Calibration	running follower motors independent of the master - is performed through a RAPID calibration program, to ensure high personnel safety
At startup	a routine will automatically set the master- and follower motors at the start position, through a safe maneuver

Requirements

There are no software or hardware requirements for this option.

RAPID instructions

There are no RAPID instructions included in this option.

3.9 Service Information System

General

Note

This functionality is included in RobotWare-OS.

Service Information System is a service routine, which gives an alarm on the FlexPendant when the robot needs service.

Service alarms intervals exist for gearbox supervision as default. There is also a possibility to set user specific service intervals for calendar and operational time as well as for gearbox supervision.

The status of the service values can be checked on the FlexPendant when the robot is in manual mode.

Service information system furthermore includes a duty time counter function. This function is available for all robot types.

Note

There is also a hardware duty time counter available as option.

Service Information System is a free option and comes with all robots (does not have to be ordered).

Features

- Duty time. Sum of time, when the controller has been in the state ‘Motors on’.
- Calendar time. Elapsed time from latest service.
- Gearbox supervision. Calculated from advanced algorithms.
- Service interval alarms (Default and User set). The default alarms indicates when service should be performed. The user can also make alarms appear at a given interval before the default alarms.
- WebWare Support.

Application

All robot installations, with high quality demands. The *Service Information System* function gives possibility to predict the maintenance production stops.

Performance

- FlexPendant. Alarms are given when a service interval is passed. (Both when robot in production and manual mode.) Possibility to check the status of the service values. (Manual mode, only).
- WebWare. The option provides support for building WebWare interfaces, with possibility to check Service Information status via a computer network (LAN).

Continues on next page

3.9 Service Information System
Continued

Requirements

Service Information System does not require any additional software or hardware.

RAPID instructions

There are no RAPID instructions included in this option.

4 General RobotWare

4.1 RobotWare Add-In prepared [988-1]

General

The option *RobotWare Add-In prepared* makes it possible to run licensed Add-Ins from 3rd party developers on the IRC5 controller.

Features

Add-Ins allow to create installable supplemental software packages that extend the capabilities offered by RobotWare, making ABB's robot controllers even smarter and even more user-friendly. Creating RobotWare Add-Ins is also the recommended way for 3rd party developers to add new features into RobotWare.

An Add-In can include a number of RAPID modules, system modules, or program modules which hold the basic code for the Add-In. The Add-In also includes some files for loading and configuration at start up. The Add-In may also include *.xml* files with event log messages in different languages.

An Add-In can also consist of more advanced coding, such as *C#* code, for FlexPendant applications. This manual will cover the first case, with coding done in RAPID only. For more advanced coding, use RobotStudio SDK applications.

> **Note**
>
> The RobotWare option *RobotWare Add-In prepared* is only needed for licensed Add-Ins. It is not needed for open Add-Ins or Add-Ins delivered together with RobotWare, for example track motion and positioners.

For more information, see *Application manual - RobotWare Add-Ins*.

Application

Add-Ins can be used for any application, equipment, or functionality that extends the capabilities offered by RobotWare.

Performance

There is no specific performance data available for this option.

Requirements

Unlicensed, open, Add-Ins

What you need from ABB to package and run your own open Add-In is:

- RobotWare Add-In Packaging tool

Licensed Add-Ins

What you need from ABB to package and run your own licensed Add-In is:

- RobotWare Add-In Packaging tool
- a licence certificate for the RobotWare Add-In Packaging tool for your Add-In name

Continues on next page

4.1 RobotWare Add-In prepared [988-1]
Continued

- RobotWare option *RobotWare Add-In prepared*

To license the Add-In, you will also need:

- License Generator
- a publisher certificate.
- a licensing certificate for the License Generator

RAPID instructions

There are no RAPID instructions included in this option.

5 Motion Performance

5.1 Advanced robot motion [687-1]

About Advanced robot motion

The option *Advanced robot motion* gives you access to:

- *Advanced Shape Tuning*, see *Advanced Shape Tuning on page 32*.
- *Wrist Move*, see *WristMove on page 34*.
- Changing *Motion Process Mode* from RAPID, see *Motion Process Mode on page 40*.

5.2 Advanced Shape Tuning

General

Note

This functionality is included in the option *Advanced robot motion*, see *Advanced robot motion [687-1] on page 31*.

Advanced Shape Tuning offers the possibility to compensate for frictional effects that might appear at low speed cutting robot motion (10-100 mm/s). This is especially useful when cutting advanced shapes, for example, small circles, or other similar applications when path accuracy is crucial. Friction tuning can be used to improve path accuracy of the robot in cutting applications.

The option gives the user access to tuning parameters and the possibility to change the tuning parameters for each axis during program execution with RAPID commands in the robot program. The option also includes RAPID instructions for automatic fine tuning of friction level for each specific shape. The software automatically repeats the movement until the best friction level for each axis has been found. After the tuning has been completed each robot axis has a unique set of tuning values for each shape. The tuning is performed by the user and for each specific shape.

Features

- Very accurate path performance for advanced motion at low speed, e.g. shape cutting
- Automatic tuning of the friction level using RAPID
- Access to tuning parameters
- Tuning axis by axis
- Change tuning from RAPID program

Application

Friction effects typically arise when cutting small, advanced shapes like for example circles. The major source of friction effects comes changing direction of axis movement. The effects appear in the form of up to 0.5 mm path deviations.

Typical applications are cutting of small shapes, such as holes, slots, rectangles. Other applications may be high precision gluing or dispense of small geometries.

Performance

Using *Advanced Shape Tuning*, typically a 0.5 mm path deviation can be reduced to about 0.1 mm. This however, requires careful tuning of the friction level (see *Operating manual - IRC5 with FlexPendant* for tuning procedure, and the instruction `TuneServo` described in *Technical reference manual - RAPID Instructions, Functions and Data types*). Note that even with careful tuning, there is no guarantee that "perfect" paths will always be generated.

Continues on next page

Requirements

There are no hardware or software requirements for this option.

RAPID instructions

RAPID instructions included in this option:

Instruction	Description
FricIdInit	Instruction to start the friction level identification
FricIdEvaluate	Function that will return the friction level that produced the best results
FricIdSetFricLevels	Instruction to set friction levels

Change of tuning from RAPID is done with standard parameters.

Instruction	Description
TuneServo	*Technical reference manual - RAPID Instructions, Functions and Data types*

Limitations

- For the IRB66X0 and 7600 families of robots, no significant effects can be expected by applying *Advanced Shape Tuning*.
- For a MultiMove system, friction tuning can only be done for one robot at a time.
- The movement sequence for which friction tuning is done must begin and end with a fine point.
- The tuning process requires about 15 iterations of the movement per axis.
- The movement sequence between `FricIdInit` and `FricIdEvaluate` cannot be longer than 4 seconds.

5.3 WristMove

General

> **Note**
>
> This functionality is included in the option *Advanced robot motion*, see *Advanced robot motion [687-1] on page 31*.

WristMove is an interpolation method that only uses two axes to perform the movement. It is favorable to use in applications where one need to improve the accuracy for small shapes, for example in cutting applications. For shapes like small holes, the friction effects from the main axes (axes 1-3) of the robot may cause path deviations. *WristMove* is a method to limit the axes movement to only use two wrist axes, and thereby minimizing the friction effects on the path. In addition, a movement with *WristMove* interpolation is faster than corresponding movement without *WristMove* as less robot weight needs to be moved. The user can define which axis pair to be used for the specific movement.

Features

- Interpolation method to only use a pair of two axes. Allowable combinations, Axis5/Axis6, Axis4/Axis5 or Axis4/Axis6
- Support for any shape consisting of circular arc and straight lines, e.g. holes, slots, rectangles, etc.
- Lead in - Lead out -> any shape
- Activate interpolation mode prior to shape generation
- Used together with RAPID instruction `CirPathMode` and movement instructions for circular arcs, that is, `MoveC`, `TrigC`, `CapC`, etc

Application

WristMove is an option that can be used in cutting applications, like laser cutting, water jet cutting, routing, etc, to improve the accuracy for small shapes. The solution is a flexible, easy-to-use software feature that can be applied in any application where the robot needs to perform small shape movements.

The option can help to reduce path deviations up to or above 50% in favorable circumstances

Performance

WristMove is especially useful in cutting small holes, or other similar movements, with radius up to 25 mm. For these kinds of movements one can expect an accuracy of about ±0,1 mm when using *WristMove* at normal cutting speeds. This accuracy is the radial deviation between the actual movement and the programmed circle. This however, requires careful usage of the *WristMove* option (see *Operating manual - IRC5 with FlexPendant* for details, and the instruction `CirPathMode` described in *Technical reference manual - RAPID Instructions, Functions and Data types*). Note that even with careful usage, there is no guarantee that "perfect" paths will always be generated.

Continues on next page

WristMove can potentially improve cycle time as a movement with *WristMove* is faster than a corresponding movement without *WristMove* interpolation. The reason is that less robot weight needs to be moved in order to achieve the movement.

Requirements

There are no specific hardware or software requirements for this option.

RAPID instructions

There are no RAPID instructions included in this option.

Change of interpolation mode is done by setting parameters in RAPID instruction `CirPathMode`.

Limitations

- *WristMove* cannot be used if the work object is moving
- *WristMove* cannot be used if the robot is mounted on a track that is moving
- Can only use movement instructions for circular arcs, that is, `MoveC`, `TrigC`, `CapC` etc
- When cutting holes, or other shapes, the edges will be conical depending on the robot movement and the distance between tool and workobject
- The tool's height above the surface and the distance to the cutting point will vary during the cut due to the movement of only two axes

5.4 Absolute Accuracy, floor mounted [603-1]

General

Absolute Accuracy (AbsAcc) is a calibration concept which ensures a TCP absolute accuracy of better than ±1 mm in the entire working range (see, however, limitation for "bending backwards" robots below). The user is supplied with robot calibration data (compensation parameters saved on the manipulator SMB) and a certificate that shows the performance ("birth certificate"). The difference between an ideal robot and a real robot can typically be up to 10 mm, resulting from mechanical tolerances and deflection in the robot structure. The *Absolute Accuracy* option is integrated in the controller algorithms for compensation of this difference, and does not require external position recalculation.

Features

- Compensation of mechanical tolerances
- Compensation of deflection due to load (tool, object and equipment on arm)

Application

Any application where *Absolute Accuracy* is needed to facilitate:

- Exchangeability of robots
- Off-line programming with non or minimum touch-up
- On-line programming with accurate movement and reorientation of tool
- Accurate cell alignment for MultiMove coordinated motion
- Programming with accurate offset movement in relation to eg. vision system or offset programming
- Re-use of programs between Applications

Performance

Once the *Absolute Accuracy* parameters are loaded and activated, the robot can be used.

Typical production data regarding calibration are:

Robot	Positioning accuracy [mm]		
	Average	Max	% Within 1 mm
IRB 140 IRB 1400	0.35 0.35	0.75 0.75	100 100
IRB 1600	0.35	0.65	100
IRB 2400 - L IRB 2400 - 10/16	0.45 0.30	0.80 0.70	100 100
IRB 2600	0.40	0.80	100
IRB 4400	0.30	0.75	100
IRB 4600-20/2.50 IRB 4600-40/2.55 IRB 4600-45/2.05 IRB 4600-60/2.05	0.40 0.40 0.40 0.50	0.80 1.00 0.80 1.00	100 98 100 98

Continues on next page

Robot	Positioning accuracy [mm]		
	Average	Max	% Within 1 mm
IRB 6620-150/2.2	0.50	0.95	100
IRB 6640 (All variants)	0.50	1.20	97
IRB 6650S-125/3.50 IRB 6650S-200/3.00	0.50	1.20	97
IRB 6660 (All variants)	0.50	1.14	98
IRB 6700 (All variants)	n.a.	n.a.	n.a.
IRB 7600 (All variants)	0.55	1.20	95

Absolute Accuracy is active in	Absolute Accuracy is inactive in
Motion function based on `robtarget` (`MoveJ`, `MoveL`, `MoveC` and ModPos)	Motion function based on a `jointtarget` (`MoveAbsJ`). Independent joint
Reorientation jogging	Joint based jogging
Linear jogging (no online compensation as the user defines the physical location, but absolute coordinates are determined for the active pose and shown in the jogging window)	External axes
Tool definition (4, 5, 6 point tool definition, room fixed TCP, stationary tool)	Track motion
Workobject definition	Any other feature not listed in “Absolute Accuracy is active in”

For joint based motions, switching to the jogging window and selecting a cartesian jog mode (Linear, Reorient) will show the correct absolute coordinates. Similarly creation of a robtarget at a point taught by joint based motion will be absolutely accurate.

Parameters

The AbsAcc parameters are stored in the robot (on the Serial Measurement Board) and are handled automatically at first start up of the robot system. Absolute Accuracy must be activated in order to be in effect.

Supported robot types

All 6-axes robots are supported.

Absolute Accuracy supports both floor mounted and inverted installations. Since calibration needs to be done in the mounting position in which the robot will be used, a special option 603-2 should be selected for inverted robots. See *Absolute Accuracy, inverted [603-2] on page 39*.

Limitations

For bending backwards robots (serial link robots), for example IRB 6640, only the forward positions are absolutely accurate.

RAPID instructions

There are no RAPID instructions included in this option.

Continues on next page

5.4 Absolute Accuracy, floor mounted [603-1]
Continued

MultiMove

If the main robot (that is the robot equipped with the main CPU) in a MultiMove system is equipped with the *Absolute Accuracy* option, it opens up Absolute Accuracy capability for all the robots in the system. However, each robot of course still needs to be calibrated individually.

The main robot (if equipped with the *Absolute Accuracy* option) will always be shipped with the correct calibration data and a “birth certificate”. For additional robots this is also the case, as long as the *Absolute Accuracy* option is specified for the robot in question.

Note

Note that this is the only RobotWare option that is relevant for an additional robot.

Note

It is possible to mix robots with and without the option *Absolute Accuracy* arbitrarily in a MultiMove system.

5.5 Absolute Accuracy, inverted [603-2]

General

Absolute Accuracy (AbsAcc) supports also inverted robots, that is hanging robots. However, since this will require that calibration shall be done with the robot in the inverted position, this option must be selected instead of option 603-1.

All features, performance and parameters as described for option 603-1 are valid also for this option, see *Absolute Accuracy, floor mounted [603-1] on page 36*.

Supported robot types

IRB 140, IRB 1600, IRB 2400, IRB 2600, IRB 4600, and IRB 6620 can be mounted in an inverted position. However, currently this option [603-2] is only available for IRB 140 and IRB 1600, that is, with calibration done before delivery. For other robots, please contact ABB Robotics.

Limitations

For bending backwards robots (serial link robots), for example IRB 6640, only the forward positions are absolutely accurate.

5.6 Motion Process Mode

General

> **Note**
>
> This functionality is included in the option *Advanced robot motion*, see *Advanced robot motion [687-1] on page 31*.

The purpose of *Motion Process Mode* is to simplify application specific tuning, i.e. to optimize the performance of the robot for a specific application.

For most applications the default mode is the best choice.

Available motion process modes

A motion process mode consists of a specific set of tuning parameters for a robot. Each tuning parameter set, that is each mode, optimizes the robot tuning for a specific class of applications.

There following modes are predefined:

- *Optimal cycle time mode* – this is the default mode and gives the same tuning as the standard robot tuning in RobotWare releases prior to 6.0. This mode gives the shortest possible cycle time and is normally the default mode.
- *Low speed accuracy mode* – this mode is recommended for applications where path accuracy is important, and for process speeds up to approximately 500 mm/s. The cycle time will be increased compared to *Optimal cycle time mode*.
- *Low speed stiff mode* - this mode is recommended for contact applications where maximum servo stiffness is important. Could also be used in some low speed applications, where a minimum of path vibrations is desired. The cycle time will be increased compared to *Low speed accuracy mode*.

Selection of mode

The default mode is automatically selected and can be changed by changing the system parameter *Use Motion Process Mode* for type *Robot*.

Changing the *Motion Process Mode* from RAPID is only possible if the option *Advanced Robot Motion* is installed. The mode can only be changed when the robot is standing still, otherwise a fine point is enforced.

The following example shows a typical use of the RAPID instruction `MotionProcessModeSet`.

```
MotionProcessModeSet OPTIMAL_CYCLE_TIME_MODE;
! Do cycle-time critical movement
MoveL *, vmax, ...;
...

MotionProcessModeSet LOW_SPEED_ACCURACY_MODE;
! Do cutting with high accuracy
MoveL *, v150, ...;
...
```

6 Motion Coordination

6.1 MultiMove Coordinated [604-1]

General

The option *MultiMove Coordinated* makes a robot system a MultiMove system with coordinated robots functionality.

A MultiMove system is a system where a common controller controls up to four robots, each equipped with its own drive module. MultiMove exists in two different modes - Independent and Coordinated.

With the *MultiMove Coordinated* option, a MultiMove system is able to work together on a common work piece and coordinated in a common workobject. *MultiMove Coordinated* also includes all *MultiMove Independent* functionality.

Features

- Up to four robots simultaneously coordinated with a common work object
- Up to six simultaneous motion tasks, handing for example four robots, one positioner and one single additional axis
- The work object can be in motion during processing. This motion can be executed by an additional axis, a multi-axis positioner or by one or several of the robots in the MultiMove group
- Any of the robots in a MultiMove group can work independently while others work coordinated. Which robots are coordinated and which are independent can change dynamically during the cycle
- Coordination is active both in automatic and manual mode. In the latter case, this means that robots can maintain their position and orientation in relation to the work object when this is moved by the joystick
- Calibration features to define coordinate systems between robots or positioners
- Synchronizing of movements in different tasks. This means that the different movements will be executed synchronized and in the same time
- MultiMove user interface on FlexPendant
- RobotWare - Multitasking
- RobotWare - Multiple Axis Positioner

Application

- Multi robot processing on a work piece mounted on a positioner.
- Processing by one or multiple robots on a work piece handled by another robot. (Flex positioner)
- Moving of heavy or flimsy objects by multiple robots.

Continues on next page

6.1 MultiMove Coordinated [604-1]
Continued

Performance

The motion performance of robots in a MultiMove group is equivalent to that of a single robot system, in terms of speed and acceleration. The total path accuracy when one robot is moving the work object and another is doing processing on the same, will consist of a superimposition of the two robot's accuracy. This means an error, which is less than or equal to the sum of those of the individual robots.

Regarding absolute accuracy, see *Requirements on page 42*.

The main CPU has reserved power for dealing with MultiMove, including "normal" RAPID processing. For very demanding RAPID processing, there may be a slight impact on cycle time, compared to a single robot system.

Requirements

- For communication with additional drive units, the hardware option 710-1 is required. This option is obviously not needed if MultiMove is used within a single drive module.
- This option is relevant and required only for systems, when coordination between robots and/or manipulators, controlled from different tasks, is needed. Each RAPID task can control one robot and up to six external axes (positioners without TCP).
- Coordination between robots, controlled from different tasks but working in a common moveable workobject, is only possible within synchronized movement sequences (see instructions below). When working in a stationary workobject, no synchronization is needed, and the option ‘MultiMove Independent’ can be used.
- The accuracy of coordinated motion is obviously depending on the position accuracy of each robot. To achieve the best possible coordination between robots, it is highly recommended to use *Absolute Accuracy* [603-1] on the robots involved.

Limitations

A MultiMove system is to be regarded as “one machine”, in the sense that all robots involved are always in the same state, since there is only one common safety system. MultiMove can thus not be applied for robots in different cells.

When a robot is moved by an additional axis, e.g. a track, the track and the robot must be controlled by the same task. This implies that the same additional axis cannot move multiple robots. There are two exceptions where such a set-up is, after all, possible:

- The robots are working independently and the robots in other tasks than the one controlling the additional axis are not dependent on "knowing" their position in the world coordinate system.
- All robots moved by the same additional axis are working synchronized all the time (i.e. using `SyncMoveOn` and attaching an `identno` to each `Move` instruction). The only possible exception (outside a `SyncMoveOn/SyncMoveOff` sequence) is using `MoveAbsJ`.

A maximum of two IRB 360 can be used in a MultiMove system.

Continues on next page

This option is not available with IRC5C Compact controller.

RAPID instructions

RAPID instructions included in this option:

Instruction	Description
SyncMoveOn	Activation of synchronized movement sequence for two or more robots and manipulators
SyncMoveOff	Deactivation of synchronized movement sequence
SyncMoveUndo	Deactivation of synchronized movement sequence from any place in the RAPID program

6.2 MultiMove Independent [604-2]

General

The option *MultiMove Independent* makes a robot system a MultiMove system with independent robots functionality.

A MultiMove system is a system where a common controller controls up to four robots, each equipped with its own drive module. MultiMove system exists in two different modes - Independent and Coordinated.

With *MultiMove Independent*, the robots run independently of each other, i.e. controlled by separate RAPID tasks. It is also possible to run positioners independently (controlled by separate RAPID tasks.)

Features

- Up to four robots in a MultiMove System
- Up to six simultaneous motion tasks, handing for example four robots, one positioner and one single additional axis
- The robots in the MultiMove system work independently of each other
- MultiMove user interface on FlexPendant
- RobotWare - Multitasking
- RobotWare - Multiple Axis Positioner

Application

Multi robot processing where each robot is working independently, controlled by separate RAPID tasks.

Performance

The motion performance of robots in a MultiMove system is equivalent to that of a single robot system, in terms of speed and acceleration. The main CPU has reserved power for dealing with MultiMove, including "normal" RAPID processing. For very demanding RAPID processing, there may be a slight impact on cycle time, compared to a single robot system.

Requirements

- For communication with additional drive units, the hardware option 710-1 is required. This option is obviously not needed if MultiMove is used within a single drive module.

Limitations

A MultiMove system is to be regarded as “one machine”, in the sense that all robots involved are always in the same state, since there is only one common safety system. MultiMove can thus not be applied for robots in different cells.

When a robot is moved by an additional axis, e.g. a track, the track and the robot must be controlled by the same task. This implies that the same additional axis

Continues on next page

cannot move multiple robots. There are two exceptions where such a set-up is, after all, possible:

- The robots are working independently and the robots in other tasks than the one controlling the additional axis are not dependent on "knowing" their position in the world coordinate system.
- All robots moved by the same additional axis are working synchronized all the time (i.e. using `SyncMoveOn` and attaching an `identno` to each `Move` instruction). The only possible exception (outside a `SyncMoveOn/SyncMoveOff` sequence) is using `MoveAbsJ`.

A maximum of two IRB 360 can be used in a MultiMove system.

This option is not available with IRC5C Compact controller.

RAPID instructions

There are no RAPID instructions included in this option.

6.3 Conveyor Tracking [606-1]

General

The option *Conveyor Tracking* (also known as line tracking) contains functionality which makes the robot follow a work object on a moving conveyor. While tracking the conveyor, the programmed TCP speed, relative to the work object, will be maintained even when the conveyor speed is changing slowly.

Features

- Linear and circular conveyors
- Up to 4 conveyors simultaneously. Switch between tracking the one or the other
- Up to 254 objects can be organized in an object queue that can be manipulated by RAPID instructions
- Possibility to define a start window in which an object must be before tracking can start
- A maximum tracking distance may be specified
- If the robot is mounted on a parallel track motion, the system can be configured such that the track will follow the conveyor and maintain the relative position to the conveyor.
- Tracking of a conveyor can be activated ‘’on the fly’’, that is it is not necessary to stop in a fine point.

Application

Any application where conveyors are used for example painting, arc welding, picking and other applications.

Performance

At 150 mm/s constant conveyor speed, the TCP will stay within 2 mm of the path as seen with no conveyor motion. When the robot is stationary relative to the conveyor, the TCP will remain 0.7 mm of the intended position.

Note

Make sure that the robot is within its dynamic limits with the added conveyor motion and that the conveyor is accurately calibrated.

Requirements

The following hardware components are required for measuring the conveyor position:

- DeviceNet
- Encoder card DSQC 377B

See the product specification for the respective robot.

Continues on next page

RAPID instructions

RAPID instructions included in this option:

Instruction	Description
WaitWObj	Connects to a work object in the start window
DropWObj	Disconnects from the current object

6.4 Indexing Conveyor Control [606-2]

General

The option *Indexing Conveyor Control* offers functionality which makes the robot follow a work object on a fast moving indexing conveyor. An indexing conveyor has a number of equally sized partitions or pockets, where the handled objects are placed. To receive an object in such a pocket requires that the conveyor is momentarily stopped to get the object, and then very quickly moved to receive a new object in the next pocket etc. This kind of fast stopping and moving in specific distances is called *indexing*. While the conveyor is indexing, the robot will be tracking the objects on the conveyor, and the position, relative to the work object, will be maintained even when the conveyor speed is changing during indexing.

Features

- Linear conveyors
- Up to two IRB 360 in a MultiMove setup, working with separate indexing conveyors
- Up to 2 indexing conveyors and up to 2 regular conveyors may be handled simultaneously. Switch between tracking the one or the other.
- Indexing conveyors are controlled by the robot system, i.e. no encoder board is needed.
- Synchronizing of the indexing conveyor with the incoming objects through digital input.
- Up to 100 objects can be organized in an object queue that can be manipulated by RAPID instructions.
- Possibility to define a start window in which an object must be before tracking can start.
- A maximum tracking distance may be specified.
- Tracking of a conveyor can be activated "on the fly", that is it is not necessary to stop in a fine point.
- Can work independently or in combination with PickMaster 3.

Application

Any application where high speed indexing conveyors are used, for example small parts handling and other applications. Normally, in a "box loading" application, an in feeder will place the objects on the indexing conveyor with high rate frequency. After that the robot is picking the objects from the indexing conveyor and placing them on an out feeder conveyor.

Performance

In a typical case with box widths of 50 mm, 2 kg robot payload, product feeding frequency of 7.5 Hz and with accelerations/decelerations up to 35 m/s2, the TCP will stay within 2 mm of the path as seen with no conveyor motion.

(Make sure that the robot is within its dynamic limits with the added conveyor motion and that the conveyor is accurately calibrated.)

Continues on next page

Requirements

Since the indexing conveyor is controlled by the robot system, there is no need for an encoder board. However, if also handling parts on a regular conveyor, hardware components for measuring the conveyor position are needed like DeviceNet option and encoder board DSQC 377B. See the product specification for the respective robot.

For the synchronization of the indexing conveyor with the incoming objects, a synchronization pulse shall be provided at a digital input minimum 200 ms in advance (depending on robot pay load).

To ensure accuracy, the conveyor transmission must follow specific rules (see *Application manual - Conveyor tracking*).

RAPID instructions

RAPID instructions included in this option:

Instruction	Description
WaitWObj	Connects to a work object in the start window
DropWObj	Disconnects from the current object
IndCnvInit	Initiation of conveyor
IndCnvEnable	Enabling indexing mode
IndCnvDisable	Disabling indexing mode
IndCnvReset	Resetting indexing mode back to normal mode
IndCnvAddObject	Adding an object to object queue

Limitations

Indexing conveyor control is only available for IRB 360.

Indexing conveyor should be equipped with an ABB motor package or similar, see *Application manual - Additional axes and stand alone controller*.

6.5 Sensor Synchronization [607-1]

General

Sensor Synchronization adjusts the robot speed to an external moving device (for example a press or conveyor) with the help of a sensor. The option can also be used for synchronizing two robots with each other.

This *Sensor Synchronization* option simplifies programming and improves productivity of any loading /unloading application, since it provides automatic sensor status check and speed adjustment.

The robot TCP speed will be adjusted in correlation to the sensor output, so that the robot will reach the programmed robtargets at the same time as the external device reaches the programmed position. The synchronization is started/stopped with a RAPID instruction combined with movement instructions (fine points or corner zones).

Features

- Synchronize robot with sensor: Up to 4 sensors/robot
- Synchronize 2 or more robots
- “On-the-fly” activation
- Valid for any type of movement
- RAPID access to sensor and queue data
- Object queue (see *Conveyor Tracking [606-1] on page 46*)

Application

- Synchronization of robots and press equipment
- “Side robot” or “Top robot“ in paint application

Performance

The TCP will stay within +/- 50ms delay of the taught sensor position, for linear sensors and constant sensor speed.

Requirements

Please refer to the Product specification for your controller:

- Synchronize robot to server: DeviceNet option encoder card DSQC 377B
- Synchronize robot to robot: DeviceNet option

RAPID instructions

RAPID instructions included in this option:

Instruction	Instruction
SyncToSensor	Start/stop synchronization
WaitSensor	Connect to an object in the start window
DropSensor	Disconnect from current object

6.6 Analog Synchronization [607-2]

General

Analog Synchronization adjusts the robot speed to an external moving device (for example a machine) with the help of an analog linear sensor attached to the moving device. This *Analog Synchronization* option simplifies programming and improves productivity of any loading/unloading application, since it provides automatic sensor status check and speed adjustment.

The robot TCP speed will be adjusted in correlation to the sensor output, so that the robot will reach the programmed robtargets at the same time as the external device reaches the programmed position. The synchronization is started/stopped with a RAPID instruction combined with movement instructions (fine points or corner zones).

Features

- Synchronized move in
- Supervised move out
- “On-the-fly” activation
- Valid for any type of movement

Application

When *Analog Synchronization* is used, the work flow of robot and machines overlap to save time. With *Analog Synchronization*, at machine opening, as soon as there is enough space for the robot between the machine platens, the sensor triggers the robot to start to move into the machine, synchronized with the moving platen of the machine. Time is saved by the robot’s early entering.

At machine closing, as soon as the part has been gripped, the robot starts to move out of the machine. When the robot has reached the machine closing point on its synchronized path, the machine starts to close. The machine closes safely while the robot moving out. Time is also saved by the machine’s early closing. Applicable to injection moulding, die-cast or other machine tending or material handling.

Performance

Note

See *Sensor Synchronization [607-1] on page 50*.

Requirements

See the product specification for the robot controller:

- Analog linear sensor with analogue signal between 0 and 10 V (for example Balluf)
- Fieldbus coupler to connect the sensor to the IRC5
- Analog I/O card (3rd party DeviceNet solution)
- Optional: *Electronic position switches*

Continues on next page

6.6 Analog Synchronization [607-2]
Continued

RAPID instructions

Note

See *Sensor Synchronization [607-1] on page 50*.

Limitations

- 607-1 and 607-2 cannot be combined.

7 Motion Events

7.1 World Zones [608-1]

General

The *World Zones* option is used to define in which area in space the TCP is operating, or the current joint configuration.

Features

- Set input/output signal, when TCP or joint within or outside zone
- Stop robot when reaching a zone border
- Cubical, Cylindrical, Spherical and Joint zones
- Set I/O when the robot is in ‘home’ position and configuration
- Automatic activation at start-up or activated/deactivated from RAPID program
- Active in automatic and manual mode
- For a MultiMove system, each robot will have its own world zones, independent of each other

Application

Application	Description
Home position	When the robot is started from a PLC, the PLC will check that the robot is inside the volume of the home configuration. In this way other equipment may move safely in the cell.
Protection of equipment	A zone may enclose other cell equipment, and thus prevent the robot from moving into that area.
Robots working in the same area	Handshaking between robots ensures that only one robot at a time is working within a zone. This functionality also ensures efficiency in these operations, since robots can be put waiting for another robot to finish its work within the zone and immediately enter the zone, when the first is finished.

Performance

For safety reasons, this software function shall not be used for protection of personnel. Use hardware protection equipment.

Requirements

There are no software or hardware requirements for this option.

RAPID instructions

RAPID instructions included in this option:

Instruction	Description
WZBoxDef	Define a cubical world zone
WZCylDef	Define a cylindrical world zone
WZLimSup	Activate world zone limit supervision
WZSpDef	Define a spherical world zone

Continues on next page

7.1 World Zones [608-1]
Continued

Instruction	Description
WZDOSet	Activate world zone digital output
WZDisable	Deactivate world zone supervision
WZEnable	Activate world zone supervision
WZFree	Erase world zone supervision
WZHomeJointDef	Define a global zone in joint coordinates
WZLimJointDef	Define a global zone in joint coordinates, for limitation of work area

8 Motion Functions

8.1 Independent Axis [610-1]

General

The *Independent Axis* option is used to make an external axis (linear or rotating) run independently of the other axes in the robot system.

The option also includes the function *Axis Reset*, which can reset the axis position counter from RAPID. *Axis Reset* is useful for repeated maneuvers, where mechanical reset of the axis (mechanically turning back the axis) would mean loss of cycle time in the process.

Features

- Movement of an axis, independent of the robot motion
- Independent movements, programmed with absolute or relative positions
- Continuous rotational/linear movement of an axis
- Speed regulation of the independent axis
- Reset of Axis position counter (axes 4, 6 and additional (rotating) axes)

Application

Application	Description
Welding: Independent Axis	A robot is working with different stations (additional axes). First, a work object located at station 1 is welded. When this operation is completed, station 1 is moved to a position where it is easy to change the work object and at the same time the robot welds the work object at station 2. Station 1 is moved independently of the robot's movement, which simplifies operator work and reduces the cycle time.
Plasma spraying: Independent Axis	A robot is spraying an object, which is rotated continuously by an additional axis in front of the robot. The speed of the additional axis can be changed during the process for optimal results.
Polishing: Axis Reset	When polishing, a large work area is sometimes needed on the robot axis 6 in order to be able to carry out final polishing without stopping. Assume that the axis has rotated 3 turns, for example. It can now be reset using this function, without having to physically rotate it back again. Obviously this will reduce cycle times.

Performance

The movements will be made with the same performance as additional axes without *Independent Axis*.

Requirements

There are no software or hardware requirements for this option.

Limitations

If an axis has a gear ratio which is not an integer number, fine calibration is required after resetting the revolution counter on the serial measurement board, if a precise axis position is needed in the application.

Continues on next page

Internal and customer cabling and equipment may limit the ability to use independent axis functionality on axis 4 and 6. The option is not possible to use together with track motion, IRBT and option SafeMove.

For information about what capability a specific robot's axis 4/6 has, please contact ABB.

RAPID instructions

RAPID instructions included in this option:

Instructions	Description
IndCMove	Running an axis continuously
IndDMove	Running an axis independently a specified distance
IndRMove	Running an axis Independently to a position within one revolution, without taking into consideration the number of turns the axis had rotated earlier
IndAMove	Running an axis Independently to an absolute position
IndInpos	Checking whether or not an independent axis has reached the programmed position
IndSpeed	Checking whether or not an independent axis has reached the programmed speed
IndReset	Change an axis to dependent mode and/or reset the working area

8.2 Path Recovery [611-1]

General

The *Path Recovery* option is used to store all system data, when an interrupt occurs (fault message or other) and restore them after necessary actions have been taken.

Features

- Store path data (all current system information)
- Restore path data (all system information, as was before interrupt/fault)

Application

Application	Description
Service of welding guns	When an error message occurs, the position/path data can be stored and the robot moves automatically to a service area. After service, the robot moves back to the exact same position, including all system data and continues welding.

Performance

There is no specific performance data available for this option.

Requirements

There are no software or hardware requirements for this option.

RAPID instructions

RAPID instructions included in this option:

Instruction	Description
StorePath	Stores the path when an interrupt occurs
RestorePath	Restores the path after an interrupt
PathRecStart	Start the path recorder
PathRecStop	Stop the path recorder
PathRecMoveBwd	Move path recorder backwards
PathRecMoveFwd	Move path recorder forward

8.3 Path Offset [612-1]

General

Path Offset (path corrections) changes the robot path according to the input from a sensor. The robot can thus follow/track a contour, such as an edge or a weld.

The path corrections will take effect immediately when receiving data from the sensor, also during movement between two positions. The path corrections are entered from the RAPID program. *Path Offset* can be made in the path coordinate system.

Features

- Track a robot path at a user set offset
- Read current path offset
- Change path offset ‘in action’

Application

Application	Description
Path offset	Mainly used in ArcWelding, to track a work object at a certain distance. A sensor is used to define the robot input for path correction during motion. The input can be defined via an analog input, a serial channel or similar. Multitasking or interrupts are used to read this information at specific intervals. Based on the input value, the path can then be adjusted

Performance

Minimum offset: 0.1mm.

Requirements

There are no software or hardware requirements for this option.

RAPID instructions

RAPID instructions included in this option:

Instruction	Description
CorrCon	Activating path correction
CorrDisCon	Deactivating path correction
CorrRead	Read current path correction
CorrWrite	Changing path correction
CorrClear	Removes all correction generators

8.4 SoftMove [885-1]

General

SoftMove is a cartesian soft servo option that allows the robot to be compliant or floating in order to adjust to external forces or variations in work objects. *SoftMove* can lower the stiffness of the robot in one or several pre-defined cartesian direction(s) (in relation to either the tool or the work object) while keeping the original behavior in the other directions. The behavior of the softness is controlled by two parameters *Stiffness* and *Damping*. With *SoftMove*, the robot is compliant in the specified direction(s) only which facilitates high accuracy and reliability. The option reduces robot programming time and enables effective interaction between robot and machine, which reduces cycle time.

Features

SoftMove is used to set up softness in one of the following directions:

- one of the Cartesian directions (x, y or z)
- one of the Cartesian planes (xy, xz or yz)
- all directions (xyz)
- the plane xy and rotational around the z axis

Applications

Machine tending of different machines, for example die casting machines, injection moulding machines, machine tools, etc. Extraction of parts from machine

- Insertion of parts into the machine – robot holds or pushes
- Extraction of parts from machine
- Placing/picking a work object in a tool
- Placing a molded or cast part in a fixture
- Tool exchanging on peripheral machines
- Absorbing of shocks and vibrations

Assembly functions

- Framing - a robot holds and presses a part towards the rest of the car body while another process attaches the part to the body
- Simple assembly functions not requiring searching or fitting

Welding

- Hold-and-Weld
- Hotplate welding

Press tending

- Follow movement of press

Polishing/Grinding

- Simple polishing and grinding applications with low process forces not requiring process feedback

Continues on next page

8.4 SoftMove [885-1]
Continued

RAPID instructions

See *Application manual - SoftMove*.

Limitations

Collision Detection is deactivated when *SoftMove* is activated.

Activation and deactivation of *SoftMove* can only be done in fine points.

SoftMove does not work for 4-axis robots, for example IRB 260, IRB 360, and IRB 660.

SoftMove does not work together with:

- *MultiMove Coordinated*
- Tracking functionality such as, `CorrWrite`, *Conveyor Tracking*, seam tracking and *WeldGuide*.
- Force control options

`CSSAct` does not activate motion control. A movement instruction is required to activate the motion control.

9 Motion Supervision

9.1 Collision Detection [613-1]

General

Collision Detection is a software option, which reduces collision impact forces on the robot. In this way, the robot and external equipment can be protected from severe damage.

Features

- Protection of robot and equipment
- Protection from collisions from any direction
- Robot movement stops and the robot moves back, along the programmed path
- Can reduce the collision force to 30%

Application

Abnormal torque levels on any robot axis (for additional axes, only positioners listed below are covered) are detected and will cause the robot to stop quickly and thereafter back off to relieve forces between the robot and environment.

Performance

The sensitivity (with default tuning) is comparable to the mechanical alternative (mechanical clutch) and it is in most cases much better. In addition, *Collision Detection* has the advantages of no added stick-out and weight, no need for connection to the e-stop circuit, no wear, the automatic backing off after collision and, finally, the adjustable tuning.

Tuning is normally not required, but the sensitivity can be changed from RAPID or manually. *Collision Detection* can also be switched off completely for part of a program. This may be necessary when strong process forces are acting on the robot.

Requirements

This option is available for all robot types and for positioners:

- IRBP-L
- IRBP-K
- IRBP-R
- IRBP-A

RAPID instructions

RAPID instruction included in this option:

Instruction	Description
MotionSup	Changing the sensitivity of the collision detection or activating/deactivating the function.

This page is intentionally left blank

10 Communication

10.1 FTP Client [614-1]

General

Note

In the specification form, this option is called *FTP and NFS client*, since both functions are merged into a common option.

The option *FTP client* (File Transfer Protocol) makes it possible to read information on a remote hard disk, for example a PC, directly from the controller.

Features

- Read information on remote hard disk
- Backup/restore from Flex Pendant
- Load and save RAPID programs
- Issue saving/reading of data from RAPID program

Application

- A robot might be using different programs for different products and the programs might be loaded on a PC. When a new part is to be produced, and thus a new program is to be loaded, the program can be read directly from the PC. Reading a program can be done by a manual command from the flex pendant or with a RAPID instruction in a program.
- Several robots might be connected to a PC via Ethernet. The control program and the user programs for all the robots are stored on the PC. A software update or a program backup can easily be executed from the PC.

Performance

There is no specific performance data available for this option.

Requirements

Access rights (file attributes) read/write/delete are not explicitly checked by the FTP option prior a read/write/delete operation. It is the server's responsibility to check if a client has access rights to a file or not. The FTP option tries to access files/folders, but if it does not have access rights to perform an operation, the server should return an access violation message.

The following servers are supported:

- vsftd 1.2.2 (ReadHat Linux 9.0)
- MS IIS, (Windows XP SP2)
- Serv-U, 6.4 (Windows XP SP2)
- ftp (HP-UX version 10.20)
- Distinct ftp server, version 3.0 (Windows XP SP2)

Continues on next page

10.1 FTP Client [614-1]
Continued

Note

There is an FTP server included in RobotWare-OS, to be used when an external FTP client is available.

RAPID instructions

There are no RAPID instructions included in this option.

10.2 NFS Client

General

Note

This option is merged into option 614-1 FTP and NFS client.

The option *NFS client* (Network File System) makes it possible to read information on a remote hard disk, for example a PC, directly from the controller.

Features

- Read information on remote hard disk
- Backup/restore from Flex Pendant
- Load and save RAPID programs
- Issue saving/reading of data from RAPID program

Application

- A robot might be using different programs for different products and the programs might be loaded on a PC: When a new part is to be produced, and thus a new program is to be loaded, the program can be read directly from the PC.Reading a program can be done by a manual command from the flex pendant or with a RAPID instruction in a program.
- Several robots might be connected to a PC via Ethernet. The control program and the user programs for all the robots are stored on the PC: A software update or a program backup can easily be executed from the PC.

Performance

There is no specific performance data available for this option.

Requirements

Access rights (file attributes) read/write/delete are not explicitly checked by the FTP option prior a read/write/delete operation. It is the server's responsibility to check if a client has access rights to a file or not. The FTP option tries to access files/folders, but if it does not have access rights to perform an operation, the server should return an access violation message.

The following servers are supported:

- OmniNFS version 4.3 (Windows XP SP2)
- Distinct, version 3.0 (Windows XP SP2)
- nsf (HP-UX, version 10.20)

RAPID instructions

There are no RAPID instructions included in this option.

10.3 PC Interface [616-1]

General

Note

This option includes also *Socket Messaging* (see *Socket Messaging on page 71*).

PC Interface provides the communication interface between the robot controller and network connected PCs.

Features

- Manual robot backup to a networking PC (included in RobotStudio)
- OPC Server interface for SCADA integration (delivered with the RobotWare DVD)
- Communication interface for use with ABB Industrial Software Products; WebWare and "PC-SDK".
- *RAPID Message Queue* (see description in *RAPID Message Queue on page 86*).

Application

Application	Description
RobotStudio	RobotStudio - offers manual tools for commissioning of robots, including activities like update/booting of the systems software, system parameter update, RAPID programming and recording of the robot log file. With the PC Interface option RobotStudio can connect to a controller over a LAN. If the PC Interface option is missing, RobotStudio only connects through the local service port.
RC5 OPC Server	IRC5 OPC Server is an OPC interface to the IRC5 controller (according to the OPC standard).
WebWare Server	WebWareTM Server - offers a 24 hours a day solution for automated backup and version control of robot programs as well as local and remote access to production reports and diagnostic information using a standard Internet browser.
PC - SoftWare Developer Kit	PC-SDK, which is included in RobotStudio, allows system integrators and end-users to program their own operator interfaces for the IRC5 robot controller. These custom applications are either added to the general functionality of the FlexPendant using FP-SDK or run as independent PC applications using PC-SDK. The PC Interface option is required to use a custom PC-based application with a robot.

Requirements

Application needs to be developed with the software product: PC-SDK.

RAPID instructions

RAPID instruction included in this option:

Instruction	Description
SCWrite	Sends a message to the network PC

10.4 FlexPendant Interface [617-1]

General

FlexPendant Interface provides the possibility to download and run user-developed operator interfaces on the FlexPendant.

Features

- Download and execute operator interfaces on the FlexPendant
- Application specific operator interface visible as a new entry in the ABB-menu of the FlexPendant
- *RAPID Message Queue* (see *Multitasking [623-1] on page 73*)

Application

Application	Description
FlexPendant - Software Developer Kit	FP-SDK, which is included in RobotStudio, allows system integrators and end-users to program their own operator interfaces for the IRC5 robot controller. These custom applications are either added to the general functionality of the FlexPendant using FP-SDK or run as independent PC applications using PC-SDK. The FlexPendant Interface option is required to execute a FlexPendant application on the FlexPendant.

FP-SDK is entirely integrated with Visual Studio 2005. The visual design support of Visual Studio 2005 is used to build the graphical user interface of the FlexPendant application. The functionality targeting the robot controller is developed in the .NET programming languages C# or Visual Basic, which are available in VS 2005. A FlexPendant application can be deployed to a customer as an additional option or added to the HOME directory by the Installation Manager wizard.

Performance

There is no specific performance data available for this option.

Requirements

Application needs to be developed with the software product: "FP-SDK".

RAPID instructions

There are no RAPID instructions included in this option.

10.5 Field bus Command Interface [618-1]

General

> **Note**
>
> This functionality is included in RobotWare-OS.

Field bus Command Interface is an option, which is used when transferring commands or messages other than I/O signals from the controller to/from units connected via the physical DeviceNet.

Features

- Open a DeviceNet unit
- Read from a DeviceNet unit
- Write to a DeviceNet unit
- Close a DeviceNet unit

Application

The option will be used for example when an intelligent control unit for external equipment is connected to the robot controller via the DeviceNet bus. An example of this is the integrated ARCITEC power source for arc welding.

After such a unit is configured as a device on DeviceNet, it can be handled like other communication devices for example using the RAPID commands `Open\Bin`, `Close`, `ReadRawBytes`, `WriteRawBytes`.

Performance

The *Fieldbuses Command Interface* can handle data blocks of a size of 1024 bytes.

Requirements

This option requires the DeviceNet option and the option *File and Serial Channel Handling*. See *File and Serial Channel Handling on page 18*.

RAPID instructions

RAPID instructions included in this option:

For open and close, the standard `Open` and `Close` instructions shall be used.

Instruction	Description
PackDN Header	Pack the header of a DeviceNet message

10.6 RobotStudio App Connect [688-1]

General

RobotStudio App Connect provides the possibility to use standard commercially available tablets for commissioning at the shop floor as an alternative to the Flexpendant.

Features

Enables the possibility to interact with the robot controller using RobotStudio Online apps.

Application

RobotStudio Online is a suite of apps for Windows tablets to interact with the IRC5 controller.

The following RobotStudio Online apps are available for free download at the Windows Store:

RobotStudio Online Manage

RobotStudio Online Manage is a tool to manage ABB industrial robots with IRC5 controllers on a network. The network can be partitioned into groups. Status information will be shown either for an individual robot controller or for a group.

Features:

- Show status for one or several robot controllers
- Show event logs with color coded event log messages
- Show the controller data and properties
- Save diagnostics data
- Take a backup
- Locate robot controllers
- Input/Output signal view
- Launch another RobotStudio Online app

RobotStudio Online Adjust

RobotStudio Online Adjust is a tool for calibration and definition of frames for an ABB industrial robot with an IRC5 controller.

Features:

- Base frame definition
- Tool, workobject and payload definition
- Fine calibration
- Revolution counter update
- Execute service routines

RobotStudio Online Jog

RobotStudio Online Jog is a tool for manual positioning (jogging) of the ABB industrial robot with an IRC5 robot controller. The robot can be moved using different settings and directions.

Continues on next page

10.6 RobotStudio App Connect [688-1]
Continued

Features:

- Jog individual axis
- Jog a tool linearly
- Reorient a tool
- Change settings
- Jog supervision
- Align tool to workobject

RobotStudio Online Tune

RobotStudio Online Tune is a tool for shop floor editing of RAPID programs.

Features:

- Start and step through a RAPID program
- Cut, copy and paste RAPID
- Comment and uncomment RAPID lines
- Manipulating program pointer
- Load and unload of RAPID programs and modules

Performance

There is no specific performance data available for this option.

Requirements

- The controller needs to be connected to a wireless network
- To move the robot in manual mode (that is jogging or program execution) the options [976-1] T10 or [983-1] Jokab are required as mandated by safety regulations.

RAPID instructions

There are no RAPID instructions included in this option

10.7 Socket Messaging

General

Note

This option is merged with *616-1 PC Interface*.

The purpose of *Socket Messaging* is to allow a RAPID program to exchange TCP/IP messages over a network, with a C/C++ program on another computer or a RAPID program on another robot controller.

Socket Messaging sends and receives messages over the permanent Ethernet channel of the IRC5 (which can simultaneously be used for other network traffic, for example communication with RobotStudio or WebWare Server).

Socket Messaging is a standard supported by for example UNIX and Microsoft Windows.

Features

- Creating and closing of sockets
- Setting up of a communication session
- Sending and receiving data

Application

Sockets can be used for any kind of network communication between computers/controllers. Typical examples are:

- Two robot controllers exchanging interlocking information
- Communication between a robot controller and peripheral devices like sensors, bar code readers or process controllers
- Intertask communication within the same controller

Performance

There is no specific performance data available for this option.

Requirements

There are no hardware or software requirements for this option.

RAPID instructions

RAPID instructions included in this option:

Instruction	Description
SocketCreate	Create a new socket and assign it to a `socketdev` variable
SocketClose	Close a socket and release all resources
SocketBind	Bind the socket to a specified port number
SocketListen	Make the computer act as a server and accept incoming connections
SocketConnect	Make a connection request to a remote computer

Continues on next page

Instruction	Description
SocketAccept	Accept an incoming connection request
SocketSend	Send data via a socket connection to a remote computer
SocketReceive	Receive data and store it
SocketGetStatus	Returns the current state of a `socketdev` variable

11 Engineering Tools

11.1 Multitasking [623-1]

General

The *Multitasking* option gives the possibility of executing up to 20 programs (tasks) in parallel, including the main program. *Multitasking* can be used to control peripheral equipment or other processes concurrently with robot motion.

Features

- Automatic start at power on
- START/STOP commands for task execution
- Tasks are programmed using standard RAPID instructions
- Priorities can be set between tasks
- All input and output signals and the file system are accessible for each task
- *RAPID Message Queue*, see *RAPID Message Queue on page 86*.

Application

Application	Description
Supervision	A task can be used to continuously monitor certain signals even when the main program has stopped, thus taking over the job traditionally allocated to a PLC.
Operator dialogue	An operator dialogue might be required at the same time as the robot is performing, for example welding. By putting this operator dialogue into a parallel task, the operator can specify input data for the next work cycle without having to stop the robot.
Control of external equipment	The robot can control a piece of external equipment in parallel with the normal program execution.

Performance

It is possible to configure if the task shall react on START/STOP requests or if it shall start automatically. In the later case it will not stop at emergency stops, which can be useful for some applications.

Note

The response time of *Multitasking* does not match that of a PLC. *Multitasking* is primary intended for less demanding tasks.

The normal response time is 5-120 ms. The longer time is for cases when heavy calculation of movement is performed.

Requirements

There are no software or hardware requirements for this option.

Continues on next page

RAPID instructions

RAPID instructions included in this option:

Instruction	Description
WaitSyncTask	Synchronize several program tasks at a special point in each program.

11.2 Continuous Application Platform [624-1]

General

Continuous Application Platform (CAP) is a software platform for time critical application, where a continuous process, for example arc welding, must be synchronized with the TCP movement of the robot.

Target users are advanced application software engineers and system integrators, for example for arc welding, laser welding, or laser cutting. The main advantages are achieved in the following areas:

- Development time
- Program execution time
- Similar look and feel for application
- Stable software kernel (RobotWare)

Features

- Special RAPID instructions and data types
- A single instruction for motion and process control
- Supports encapsulation of the process and motion in RAPID wrappers provided to the end-user
- Flying start/Flying end support

Application

Creation of advanced application software with a continuous process to be synchronized with robot movement, for example arc welding, laser cutting, or laser welding.

Performance

Part of the RobotWare kernel and RAPID instructions: CAP is designed to support fast and quality secured continuous application demands. The application developer defines the degree of ease-of-use by hiding process complexity from the end-user.

Requirements

There are no software or hardware requirements for this option.

RAPID instructions

Application manual - Continuous Application Platform

11.3 Optical Tracking [813-1]

General

The option *Optical Tracking* is identical to the option *Optical Tracking Arc [660-1]*, except for that it is integrated with the CAP movement instructions instead of with the arc welding movements.

This means that *Optical Tracking* can be used together with customized process instructions, which are using the CAP option [624-1].

For further description, see *Optical Tracking Arc [660-1] on page 130*.

11.4 Discrete Application Platform [625-1]

General

Discrete Application Platform (DAP) is a software platform for time critical application, where certain actions shall be performed at specific robot positions.

Target users are advanced application software engineers and system integrators, for example for spot welding, drilling, measuring, quality control. The main advantages are achieved in the following areas:

- Development time
- Program execution time
- RAPID program memory needed
- Similar look and feel between application
- Tested kernel software

Features

- Specialized RAPID instructions and data types
- A single instruction for motion and process execution
- Combination of fine point positioning with execution of up to 4 parallel processes
- Specialized process for monitoring of external process devices, like spot welding controllers
- Supports encapsulation of the process and motion, in shell-routines provided to the end-user

Application

Creation of software for advanced application with a discrete behavior, such as spot welding, drilling, measuring, quality control.

Performance

The DAP platform is designed to have an internal kernel, administrating the fast and quality secured process sequence skeleton. The kernel calls RAPID routines, which are prepared by the application developer to fulfill the specific tasks. The application developer regulates the degree of flexibility of the end-user.

Requirements

In order to use DAP, the option *MultiTasking* needs to be installed.

RAPID instructions

Application manual - Discrete application platform

11.5 Sensor Interface [628-1]

General

The *Sensor Interface* option can be used to integrate sensor equipment for adaptive control, like path correction or process tuning. For communication between the sensor and the robot controller two different communication links are available: serial link (RS 232) and Ethernet.

If the RS232 link is used, the sensors have to be adapted to the ABB standard application protocol, LTAPP. Sensors that are adapted to LTAPP are available from companies like Servo Robot, Meta and Scout.

The Ethernet link is available for optical sensors from ServoRobot, which use the application protocol Robo-Com Light, defined by ServoRobot.

The communication link makes it easy to exchange data between the robot controller and the sensor system, using predefined numbers for different data like x,y,z offset values, gap between sheets, time stamp etc.

If the Ethernet link, available for optical sensors from ServoRobot, is installed and configured, then the application protocol Robo-Com Light, defined by ServoRobot, will be used (see *Optical Tracking Arc [660-1] on page 130*).

Features

- Interrupt routines, based on sensor data changes
- Read/write sensor data from/to sensors using RAPID functions
- Store/retrieve sensor data as a block to/from a mass memory
- Seam tracking functionality, when combined with option *Path Offset [612-1]*
- Easy calibration with dedicated FlexPendant interface

Application

Sensor Interface can be used in any application to read/control a sensor during execution, and to react on changes in certain data, like path offset or process supervisory data, thus making adaptive seam tracking and process control possible.

Performance

There is no specific performance data available for this option.

Requirements

External sensors, communicating with the robot controller via serial link, using application protocol LTAPP, or Ethernet, using Robo-Com Light protocol.

For calibration, hardware option calibration plate [1250-1] is required.

RAPID instructions

RAPID instructions included in this option:

Instruction	Description
IVarValue	Used to order and enable an interrupt when the value of a variable accessed via the serial sensor interface has been changed

Continues on next page

Instruction	Description
ReadBlock	Used to read a block of data from a device connected to the serial sensor interface
ReadVar	Used to read a variable from a device connected to the serial sensor interface
WriteBlock	Used to write a block of data to a device connected to the serial sensor interface
WriteVar	Used to write a variable to a device connected to the serial sensor interface

11.6 Robot Reference Interface

General

Note

Robot Reference Interface is included in the RobotWare option *Externally Guided Motion*, see *Externally Guided Motion [689-1] on page 82*.

Robot Reference Interface (RRI) is an option which enables high performance data exchange between the robot controller and an external device using Ethernet link with Internet Protocol (IP). The supported data exchange provides the possibility to periodically send planned and actual robot position data as well as the exchange of other RAPID variables.

Features

The message contents are represented in XML format and are configurable using appropriate configuration files.

Data distributed from controller to external device

- Planned robot position (Cartesian TCP or Joint values with time stamp)
- Actual robot position (feedback from drives)
- Additional application data, which can be any RAPID data or operation mode status
- Data obtained from external device to controller
- Application data

Application

This option can be used for a number of different applications where accurate information about planned and actual robot position are essential. Some examples:

- High precision measuring devices, mounted on the robot, where the relevance of the collected data is based on a real time knowledge of the actual position of the robot.
- Remote laser welding, where a laser equipment with moveable laser head is held by and moved by the robot. Such an equipment can then superimpose a complex pattern over the robot movements, based on accurate robot position information.

Performance

The cyclic communication channel can be executed in the high-priority network environment of the IRC5 Controller which ensures a stable data exchange up to 250Hz.

Requirements

External devices communicating with the controller via Ethernet link, using TCP/IP.

Specific application protocol (CDP – Cyclic Data Protocol) used.

Continues on next page

Limitations

- No support for external axis positions
- No support for MultiMove
- No support for *Absolute Accuracy*

RAPID instructions

Instruction	Description
SiConnect	Establishes connection to the specified device (sensor sens)
SiSetCyclic	Initializes sending of any data at the specified rate Installs hooks in the Motion domain (sensor sens, anytype data, num rate)
SiGetCyclic	Initializes receiving of any type data at the specified rate (sensor sens, anytype data, num rate)
SiClose	Close connection to the specified device. Removes installed hooks in Motion domain. (sensor sens)

11.7 Externally Guided Motion [689-1]

General

Externally Guided Motion (EGM) is designed for advanced users and provides a low level interface to the robot controller, by by-passing the path planning that can be used when high responsiveness to robot movements are needed. EGM can be used to read positions from and write positions to the motion system at a high rate, every 4 ms with a control lag of 10–20 ms depending on the robot type. The references can either be specified using joint values or a pose. The pose can be defined in any work object for robots.

Features

The RobotWare option *Externally Guided Motion* gives you access to:

- Instructions to set up, activate, and reset EGM.
- Instructions to initiate EGM movements and to stop them.
- A function to retrieve the current EGM state.
- System parameters to configure EGM and set default values.
- Support of MultiMove and *Absolute Accuracy*.

Application

The purpose of *Externally Guided Motion* (EGM) is to use external devices to generate position data for one or several robots. The robots will be moved to that given position.

Some examples of applications are:

- Place an object (e.g. car door or window) at a location (e.g. car body) that was given by an external sensor.
- Bin picking. Pick objects from a bin using an external sensor to identify the object and its position.

Performance

EGM can be used to read positions from and write positions to the motion system at a high rate, every 4 ms with a control lag of 10–20 ms depending on the robot type.

Requirements

External devices communicating with the controller via Analog Signals, Group signals or an Ethernet link (UDP). For the Ethernet link, the application protocol (UdpUc – UDP User Communication) is used.

Limitations

- EGM can only be used in RAPID tasks with a robot, i.e. it is not possible to use it in a task that contains only additional axis, i.e. in robtargets there are values in the pose portion of the data.
- An EGM movement has to start in a fine point.

Continues on next page

- For each robot a maximum of 4 external devices may be used to define position data. For example one external device can give the x-position, a second the y- and z-position, and a third the orientation. It is then important that each data component has one and only one source. This is not checked by EGM.
- It is not possible to perform linear movements using EGM, the actual path of the robot will depend on the robot configuration, the start position and the generated position data.

RAPID instructions

For information about the included RAPID instructions, functions, and data types see *Application manual - Controller software IRC5*.

11.8 MultiFunction [824-1]

General

MultiFunction provides a discount package of a number of popular RobotWare options.

Features

- 608-1 World Zones
- 611-1 Path Recovery
- 617-1 FlexPendant Interface
- 623-1 Multitasking

Requirements

There are no software or hardware requirements for this option.

RAPID instructions

See the respective options listed above.

11.9 Production Screen [637-1]

General

Production Screen is a user-friendly HMI that upgrades your FlexPendant to a graphical user-interface. The software is based on widgets or graphical elements that are used to execute features of your choosing. You can create widgets or choose them from a wide selection of downloads in RobotApps.

With *Production Screen* you can use several widgets at once just by toggling between them. In the portal's working area, you can add widgets that control applications like arc welding, spot welding and dispensing. The variety of widget content is endless, including features for control, monitoring and notification.

Features

- Application-independent FlexPendant portal, launches apps and displays widgets.
- Window handling, new navigation feature.
- Operator information available in an easy, accessible way.
- Seamless integration of Spot, Arc, Dispense, Production Manager, and other applications.
- Possible to customize the appearance on-screen.
- Possible to create widgets using ScreenMaker or Visual Studio.

Application

Production Screen can be used in all applications to upgrade the FlexPendant to a graphical user-interface.

Performance

There is no specific performance data available for this option.

Requirements

- A FlexPendant of type SxTPU3.
- ScreenMaker in RobotStudio.

Limitations

Production Screen is only used to launch applications. A separate tool like Screen Maker is required to create the applications.

11.10 RAPID Message Queue

General

RAPID Message Queue is included in the following options:

- *Multitasking*
- *PC Interface*

The purpose of *RAPID Message Queue* is to communicate with another RAPID task or an external client. Some examples of application:

- Sending data between two RAPID tasks executing on the same controller.
- Sending data between a RAPID task and a PC application executing on a PC connected to the robot controller via Ethernet.

Features

- Configuration of communication channels and message queues using DIPC (Distributed Inter-Process Communication) techniques for distributed program executing
- Sending and receiving messages with specific RAPID commands for synchronous and asynchronous communication
- Interrupt controlled message handling

Applications

RAPID Message Queue can be used for any kind of application software running on a PC. It will facilitate communication based on specific RAPID instructions for easy programming and fast response based on interrupt hand-ling. Typical examples are:

- Fast and easy communication with peripheral devices like sensors, bar code readers and process controllers
- Efficient inter task communication within the same controller

Performance

No performance data available.

Requirements

No specific requirements for this option.

RAPID instructions

RAPID instructions included in this option:

Instruction	Description
RMQFindSlot	Find the identity number of a RAPID task or an external client
RMQSendMessage	Send data to the queue of a RAPID task or an external client
RMQGetMessage	Get the first message from the RAPID Message Queue
RMQSendWait	Send a message and wait for the answer
RMQGetMsgHeader	Extract header information from a message

Continues on next page

Instruction	Description
RMQGetMsgData	Extract data from a message
IRMQMessage	Order and enable interrupts for a specific data type
RMQReadWait	Wait for the first message from the RAPID Message Queue and return this
RMQResetQueue	Empty the RAPID Message Queue

Function	Description
RMQGetSlotName	Get the name of a RMQ client from the identity number of the client

11.11 Production Manager [812-1]

General

Production Manager delivers extended functionality for cell/production control. *Production Manager* is a process independent middle-layer software running on the IRC5 controller. *Production Manager* is working between the operating system of the robot and the end user application – that is true production ready software. *Production Manager* is a software package, written in RAPID, used for production control. *Production Manager* can be used in autonomous cells or in a PLC controlled production line.

Production Manager has a highly modular structure that allows partners/line builders/customers to plug in applications. *Production Manager* provides no real value to the end-user without an application built on top of it. However, it provides a tremendous value to systems integrators, line builders and application developers, especially to secure compatibility with other applications and to avoid development of redundant functionality. *Production Manager* also decreases time to market for new features to be developed and provides the following benefits and features.

For more details on *Production Manager* functionality see *Features on page 96*, where it is described how *Production Manager* can be used for Arc applications.

12 Servo Motor Control

12.1 Servo Tool Control

General

Servo Tool Control can be used to control a servo tool, for example in a spot weld application. *Servo Tool Control* makes it possible to close the tool to a specific plate thickness and force, and maintain the force during the process until the tool is requested to be opened.

Target users are advanced system integrators who want to develop customer specific application software, such as spot welding packages.

Servo Tool Control is included in the RobotWare option *Spot*, see *Spot 6 [635-6] on page 135*.

Features

- Position control (gap)
- Force control
- Dynamic and kinematic model (tool configured as external axis)
- Quick Start code package

Application

Application	Description
Spot welding with servo guns	The option provides advanced control functionality for servo guns. Communication with Weld timers and other process control functionality needs to be implemented outside this option. For a total spot welding package, see the option *Spot 6 [635-6] on page 135*.

Performance

The tool is configured as an external axis, which ensures optimal performance, regarding path following and speed. (Dynamic and kinematic model)

The option *Servo Tool Change* can be added to the system in order to allow a switch between two or more servo tools which will then utilize the same drive unit and measurement board.

Requirements

A specific servo tool parameter file must be installed in the controller for each servo tool. The parameter file is optimized for each system, regarding system behavior and motion/process performance.

For information on drive module & measurement board see *Application manual - Additional axes and stand alone controller*.

Limitations

- *Servo Tool Control* can only be applied to one robot in a MultiMove configuration.

Continues on next page

RAPID instructions

RAPID instructions included in this option:

Instruction	Description
STClose	Close a servo tool with a predefined force an thickness
STOpen	Open a servo tool
STCalib	Calibrate a servo tool
STTune	Tune motion parameters for a servo tool
STTuneReset	Reset tuned motion parameters
STIsClosed	Test if a servo tool is closed
STIsOpen	Test if a servo tool is open
STCalcTorque	Calculate the motor torque for a servo tool
STCalcForce	Calculate the programmable force for a servo tool

12.2 Servo Tool Change [630-1]

General

Servo Tool Change enables an on-line change of tools (external axes), for a certain drive- and measurement system. The control is switched between the axes by switching the motor cables from one servomotor to another. The switch is performed on-line during production.

Main advantages:

- Flexibility in the production process. One robot handles several tools.
- Minimized equipment. A single drive-measurement system shared by many tools.

Features

- On-line change of tools
- Up to 8 different tools

Application

Application	Description
Servo gun changing	Robot held servo guns, designed for different reach and weld forces, equipped with different brands and sizes of servo motors, may be held and operated by a robot, switching from one servo gun to another.
Servo Tool Change	Can be used as an independent option, or as an addition to the RobotWare Spot Servo options.

Performance

When switching tools, the following steps are performed (switching from Axis 1 to Axis 2):

- Axis 1 is deactivated using the RAPID instruction `DeactUnit`
- Axis 1 is disconnected from the motor cables
- Axis 2 is connected to the motor cables
- Axis 2 is activated using RAPID instruction `ActUnit`
- After activation, axis 2 is ready to run

The motor position at the moment of deactivation of one axis is saved and restored next time the axis is activated.

Note

The motor position must not change more than half a motor revolution, when the axis is disconnected. In RobotWare Spot Servo, there is a calibration routine, which handles larger position changes.

Requirements

Servo Tool Change requires a mechanical wrist interface, a tool changer.

Continues on next page

12.2 Servo Tool Change [630-1]
Continued

A MOC service parameter, *Disconnect deactive* (type *Measurement channel*), must be defined as *YES* for each tool (external axis) used with this function.

Limitations

Servo Tool Change can only be applied to one robot in a MultiMove configuration.

RAPID instructions

There are no RAPID instructions included in this option.

13 Vision

13.1 Integrated Vision

General

For information about the option *Integrated Vision*, see *Product specification - Integrated Vision*.

This page is intentionally left blank

14 Application options

14.1 MultiProcess [634-1]

General

MultiProcess is used for applying *RobotWare Arc* or *RobotWare Dispense* to multiple robots in a MultiMove system.

Features

MultiProcess enables process capabilities on any number of robots. The setup of which robot(s) that should have capabilities is done in the *Additional MultiMove selections* part of the specification form. It can also be carried out at the final installation in RobotStudio.

Requirements

Option 633-1 Arc or 641-1 Dispense must be specified, as well as any of the options:

- 604-1 MultiMove Coordinated or
- 604-2 MultiMove Independent

14.2 Arc [633-4]

General

RobotWare Arc comprises a large number of dedicated arc welding functions, which make the robot well suited for arc welding.

RobotWare Arc is a simple yet powerful option since both the positioning of the robot and the process control and monitoring are handled in one and the same instruction. I/O signals, timing sequences and weld error actions can be easily configured to meet the requirements of a specific installation. *RobotWare Arc* also includes a tool with functionality for cell/production control called *Production Manager*.

Features

- Adaptation to different equipment. The robot can handle different types of weld controllers and other welding equipment. Normally communication with the welding controller uses parallel signals but a serial interface is also available.
- Advanced process control. Voltage, wire feed rate, and other process data can be controlled individually for each weld or part of a weld. The process data can be changed at the start and finish of a welding process in such a way that the best process result is achieved.
- Testing the program. When testing a program, welding, weaving or weld guiding can all be blocked. This provides a way of testing the robot program without having the welding equipment connected.
- Automatic weld retry. A function that can be configured to order one or more automatic weld retries after a process fault.
- Weaving[I]. The robot can implement a number of different weaving patterns up to 10 Hz depending on robot type. These can be used to fill the weld properly and in the best possible way. Weaving movement can also be ordered at the start of the weld in order to facilitate the initial striking of the arc.
- Wire burn back and rollback. These are functions used to prevent the welding wire sticking to the work object.
- Fine adjustment during program execution. The welding speed, wire feed rate, voltage and weaving can all be adjusted whilst welding is in progress. This makes trimming of the process much easier because the result can be seen immediately on the current weld. This can be done in both manual and automatic mode.
- RobotWare Arc can be applied to several robots in a MultiMove system (requires option *MultiProcess*).
- *Production Manager*, a tool for cell/production control, see *Production Manager for Arc on page 99*.
- Seam Displacement allows the operator to shift seams in relation to a reference frame. The displacement are applied via FlexPendant operators screens without stopping production.

Continues on next page

- Weld error recovery. During robot production process errors sometimes arise causing the robot to stop. The Weld Error Recovery feature provides several different solutions for process error recovery, which allows operators through dialogs to automatically move the robot out of the error location to a service position. The dialogs, normally presented on the FlexPendant, may also be acknowledged from remote sources like PLC or host computer. After the process error is corrected the robot automatically returns back to the error location and continues production. This will help minimizing production downtime.
- Configurable units. Possibility to choose your own set of units for wire feed, length and velocity.
- *Weld Data Monitoring* is used to supervise the welding parameters and to synchronize the collected data to the actual weld, and welded part.

I Only available for IRB 140, IRB 1400, IRB 1600, IRB 2400, and IRB 2600.

Interface signals digital outputs

The following process signals are, if installed, handled automatically by RobotWare Arc. The robot can also support dedicated signals for workpiece manipulators and sensors.

Process signals/Digital outputs	Description
Power on/off	Turns weld on or off
Gas on/off	Turns gas on or off
Wire feed on/off	Turns wire feed on or off
Wire feed direction	Feeds wire forward/backward
Weld error	Weld error
Error information	Digital outputs for error identification
Weld program number	Parallel port for selection of program number, or 3-bit pulse port for selection of program number, or Serial CAN/Devicenet communication

Interface signals digital inputs

Process signals/Digital inputs	Description
Arc OK	Arc established; starts weld motion
Voltage OK	Weld voltage supervision
Current OK	Weld current supervision
Water OK	Water supply supervision
Gas OK	Gas supply supervision
Wire feed OK	Wire supply supervision
Manual wire feed	Manual command for wire feed
Weld inhibit	Blocks the welding process
Weave inhibit	Blocks the weaving process
Stop process	Stops/inhibits execution of arc welding instructions

Continues on next page

14.2 Arc [633-4]
Continued

Process signals/Digital inputs	Description
Wire stick error	Wire stick supervision
Supervision inhibit	Program execution without supervision
Torch collision	Torch collision supervision

Interface signals group outputs

Group outputs	Description
Schedule Port	Weld schedule sent to power source
Mode Port	Weld mode sent to power source

Interface signals analog outputs

Analog outputs	Description
Voltage	Weld voltage
Wire feed	Velocity of wire feed
Current	Weld current

Interface signals analog inputs

Analog inputs (cont.)	Description (cont.)
Voltage	Weld voltage measurement for monitoring and supervision
Current	Weld current measurement for monitoring and supervision

RAPID instructions

RAPID instructions included in this option:

Instruction	Description
ArcLStart	Arc welding start with linear movement
ArcL	Arc welding with linear movement
ArcLEnd	Arc welding end with linear movement
ArcCStart	Arc welding start with circular movement
ArcC	Arc welding with circular movement
ArcCEnd	Arc welding end with circular movement
ArcKill	Aborts the process and is intended to be used in error handler
ArcRefresh	Updates the weld references to new value

14.3 Production Manager for Arc

General

Production Manager delivers extended functionality for cell/production control to the RobotWare Arc option. To fully benefit from these extended features, RobotWare Arc has been adapted to this platform of functionality. This means that RobotWare Arc, from now on, is dependent of *Production Manager*, thus *Production Manager* will always be included when ordering RobotWare Arc.

Production Manager is, in fact, a process independent middle-layer software running on the IRC5 controller, but at present time only RobotWare Arc has been adapted to this platform.

Production Manager is working between the operating system of the robot and the end user application – that is true production ready software.

Production Manager is a software package, written in RAPID, used for production control.

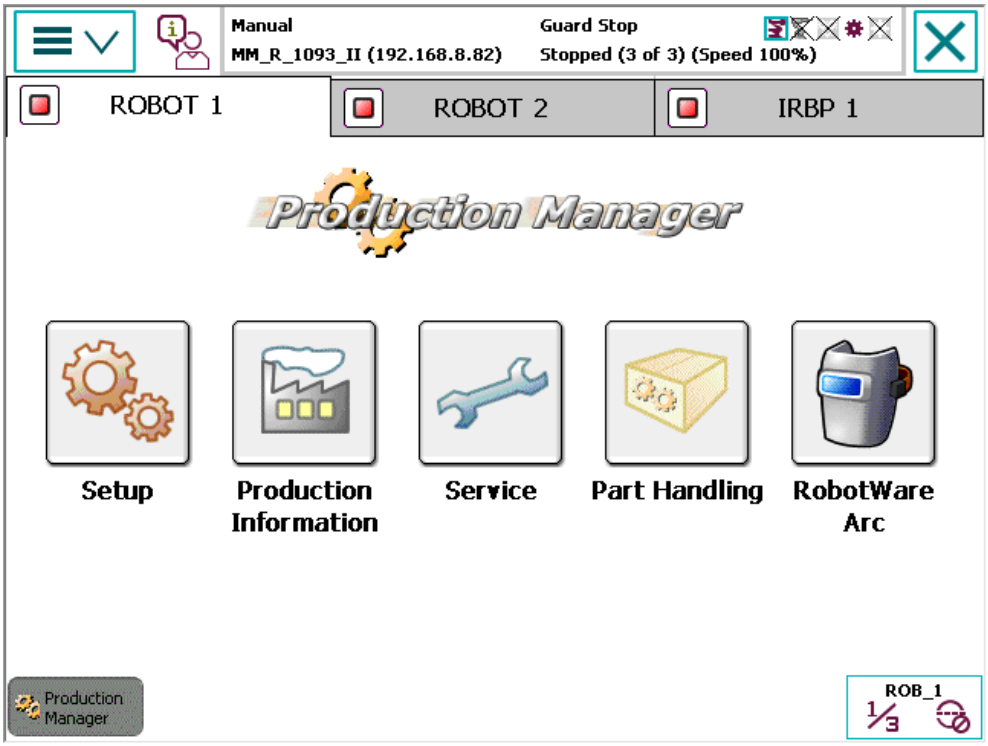
Production Manager can be used in autonomous cells or in a PLC controlled production line

Production Manager has a highly modular structure that allows partners/line builders/customers to plug in applications. *Production Manager* provides no real value to the end-user without an application built on top of it. However, it provides a tremendous value to systems integrators, line builders and application developers, especially to secure compatibility with other applications and to avoid development of redundant functionality. *Production Manager* also decreases time to market for new features to be developed and provides the following benefits and features.

Continues on next page

Features

- Production Manager includes a FlexPendant user interface for running setup and service routines, managing part handling (Select, Test, Create, Edit), displaying production information, links to application interfaces etc.

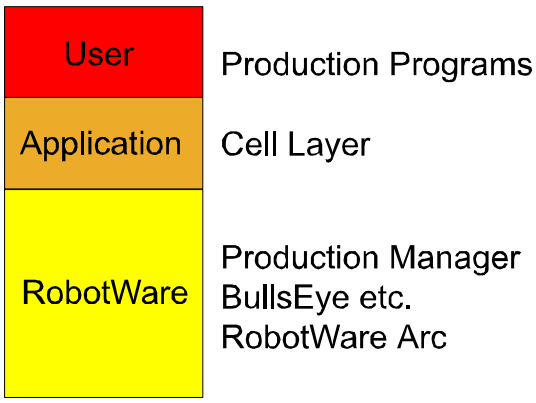


xx1000000338

- The possibility to automatically add entries to the *Production Manager* menu system (such as custom setup/service routines) enables a seamless plug-in behavior for add-on utilities.
- Part Handling. A *part* is a user defined set of RAPID code that performs some kind of task/production/etc. The GUI finds and lists the different parts (*Part* data instances) in the system. When selected, the part in question is "activated" for execution. Each motion task has its own *Part Go* signal. Setting this signal triggers the execution of the selected *Part*.
- Utilization of strong concepts provided by *Production Manager*, such as part handling and production cycle enables logging of production statistics. This creates a perfect situation for monitoring and traceability of production.
- One of the strengths of *Production Manager* is the ability to handle MultiMove synchronization and interlock issues. This is typically one of the main challenges for the system integrators when integrating a line or cell with multiple robots.
- OpReady & PLC interface.
- UAS Support
- One of the most important technical innovations that Production Manager provides is a three-tier architecture for application development, which

Continues on next page

enables application developers to separate application RAPID code from the end-user's RAPID code.



xx1000000339

Tier one: RobotWare

Standard utilities (*BullsEye*, *Torch Cleaner*, etc.) and process control software (*RobotWare Arc*) are all implemented as part of RobotWare. Platform resources for application development (*Production Manager*, RobotWare pptions) are also implemented as part of RobotWare.

Tier two: Application code

The application code that defines a complete system is developed in RAPID based on *Production Manager*, and is implemented as a middle-layer software between RobotWare and the user code. Application code is called "Cell Layer".

A typical cell layer includes:

- Call to execution engine from main procedure (`ExecEngine;`)
- Configuration of *Production Manager* on FlexPendant
- Definition of robot positions, such as home and service
- Supervision of home and service positions
- Definition of workstations for station interchange
- I/O signals for Operator Ready signals and/or PLC interface Definition of safety signals, such as light beams and curtains
- Supervision of safety signals
- Definition of service and setup routines (menu data)
- Miscellaneous user defined routines before and after part execution

Cell layer code is booted into the system and is built-in, which creates a transparent behavior with several benefits:

- User domain is clean and uncluttered
- Additional Option is revision controlled
- Additional Option code not included in system backup
- Possible to upgrade Additional Option without having the Restore function revert to old version

Continues on next page

Tier three: User code

The user code consists mainly of motion and process programs. Each program typically performs a processing task (arc welding, spot welding, laser cutting, painting etc…) for production of a part. These programs are called "Part Programs".

RAPID instructions

	Instructions	Description
ExecEngine		Run production loop
GetNextPart	`(\num gapTaskNo, num station, VAR partdata retData\VAR string instanceName)`	Get active part for station in task.
SetNextPart	`(\num gapTaskNo,num station, PERS partdata newData)`	Activate part for station in task.
UpdateNextPart	`(\num station, partdata updData)`	Update active part for station in task.
bool AtSafe	`(\num taskNumber)`	Task at safe (manipulator).
bool AtService	`(\num taskNumber)`	Task at service (manipulator).
num AtState	`(\num taskNumber)`	Task state. `GAP_STATE_UNKN:=0;` Unknown state/not running `GAP_STATE_IDLE:=1;` Executing but idle `GAP_STATE_SETUP:=2;` Executing setup routine `GAP_STATE_PART:=3;` Executing part `GAP_STATE_SERV:=4;` Executing service routine
num AtStation	`(\num taskNumber)`	Task at station.
num NextStation	`(\num taskNumber)`	Next station for task.
RunMenu	`(\num gapTaskNo,VAR menudata menu)`	Run menu from RAPID in task.
GapSetupStop	`(\switch NoRegain)`	Stop if tasks state is `GAP_STATE_SETUP`.

14.4 Arc options

General

RobotWare Arc Options is a collection of software applications, designed to simplify the use of different hardware options to improve productivity and lower the cost of owning and operating an ABB arc welding system. Some of the software options need to be purchased together with specific hardware option available in separate IRC5 Application product prices list, and other software option can be purchased individually. This document will describe these software Arc options in terms of basic functionality and if it is connected to a specific hardware.

The software is distributed with RobotWare and is activated when building a system using the Installation Manager.

Continues on next page

14.4.1 RobotWare Power Source

General

RobotWare Arc options is a collection of software applications, designed to simplify the use of different hardware options to improve productivity and lower the cost of owning and operating an ABB arc welding system. Some of the software options need to be purchased together with specific hardware option available in separate IRC5 application product prices list, and other software option can be purchased individually. This document will describe these software Arc options in terms of basic functionality and if it is connected to a specific hardware.

The software is distributed with RobotWare and is activated when building a system using the Installation Manager.

Continues on next page

14.4.1.1 Standard I/O Welder [650-7]

General

The *Standard I/O Welder* user interface is a standard arc programming tool used for operating and managing one or more I/O-based power sources. The communication is performed through a robot I/O based interface, which allows the operator to program the main process parameters from the robot FlexPendant. The operator can through this software set required wire feed speed, voltage level and welding speed for each specific weld in the robot program. Following main tasks can be handled from the FlexPendant, *Standard I/O Welder* process user interface:

- Set required wire feed speed, voltage level and absolute welding speed.

14.4.1.2 Fronius TPS 4000/5000 [650-9]

General

The *Fronius TPS 4000/5000* user interface is a standard arc programming tool used for operating and managing one or more TPS/TP-type of power sources.

The Fronius TPS and TS welding machines are totally digitized, microprocessor-controlled inverter power sources. An interactive power-source manager is coupled with a digital signal processor, and together they control and regulate the entire welding process.

The communication is performed through a robot DeviceNet interface, which allows the operator to program the main process parameters from the robot FlexPendant.

The operator can through this software can for example call for pre-programmed jobs (called schedules in robot controller), which prior have been made in on the Fronius power source operator panel.

This software option is using the RobotWare standard seam and weld data types to supply welding process information to the power supply.

The arc option *Fronius TPS 4000/5000* interface provides complete access to all Fronius modes.

The arc option *Fronius TPS 4000/5000* gives the user access to following Fronius modes:

1 Job mode
2 Job mode with correction
3 Standard program mode
4 Pulsed program mode
5 Manual

Job mode

All weld parameters, including start and end data, are stored in the Fronius power source.

The user can in this mode select a pre-programmed *job* in the Fronius power source, and use this job as the weld data in the FlexPendant.

Job mode with correction

All weld parameters, including start and end data, are stored in the Fronius power source. Same functionality as Job Mode, but the user can also get access to wire feed speed and voltage in the “weld data” to "trim" the synergic values specified in the Job.

Standard program mode

Standard synergic welding, were the user specifies wire feed speed and Fronius power source will select an appropriate voltage. Voltage may be trimmed using the voltage field. This mode gives RW - Arc the control over start and end data.

Continues on next page

Pulsed program mode

Pulsed synergic welding, were the user specifies wire feed speed and Fronius power source will select an appropriate voltage. Voltage may be trimmed using the voltage field. This mode gives RW - Arc the control over start and end data. Depending on how the system is configured, appropriate fields will be masked or exposed in weld data and seam data so that the user only sees the fields valid for the mode that is selected.

Arc errors display

The Fronius interface displays in the error number provided by Fronius in the arc errors displayed on the FlexPendant.

Robot software requirement

The RobotWare option *Arc* is required for the *Fronius TPS 4000/5000* option.

Minimum Fronius power source hard- and software requirements

- Fronius Welding Power Source (TPS/TS 4000 or 5000)
- Fronius Wire feed Systems (VR1500)
- Fronius Software option, Jobexplorer
- Fronius Interface (Bus Systems and standard discrete)

Optional

Fronius Remote Control Units (RCU 4000 and 5000)

Note

The above Fronius equipment and software options are not included in RW Fronius TPS 4000/5000 option [650-9].

For more information see separate Application manual-Fronius TPS 4000/5000 IRC5 Interface.

14.4.1.3 AristoMig integrated [650-10]

General

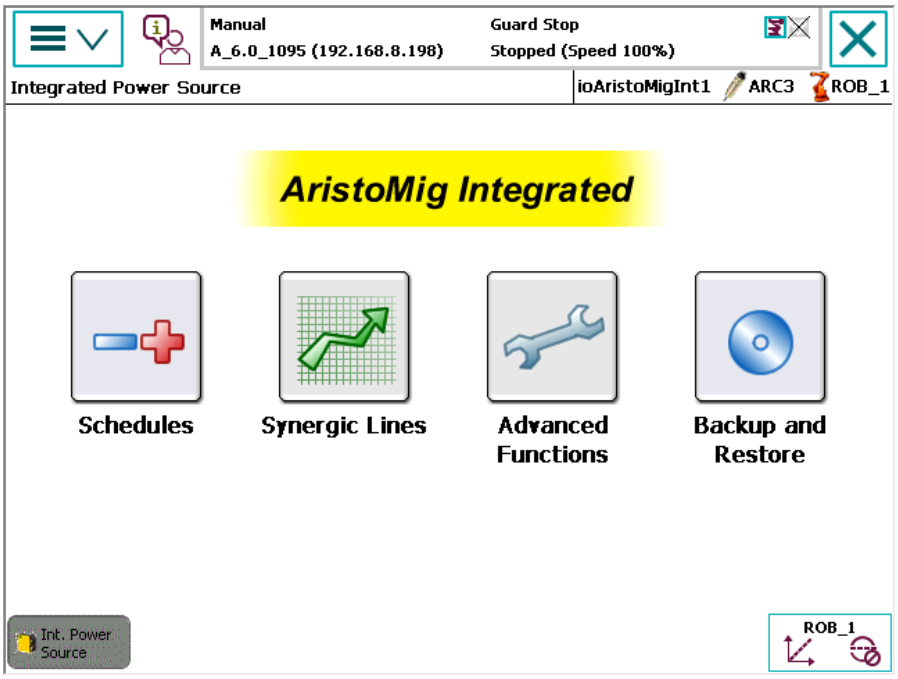
The option *AristoMig integrated* is a user interface for programming one or more Esab AristoMig with a W8 ABB interface.

Features

The *AristoMig integrated* option has the following functionality:

- Create and edit schedules.
- Create user defined synergic lines.
- Reading service information and executing service functions.
- Backup and restore of SID files.

Application



xx1000000337

14.4.1.4 ABB RPC S Integrated [650-11]

General

The option *ABB RPC S Integrated* is a software option for managing the RPC S 400 Power source.

Features

The *ABB RPC S Integrated* option has the following functionality:

- Create and edit schedules.
- Monitoring of current and voltage.
- Backup and restore of schedules in the RPC S 400 Power source.

Requirements

The following RobotWare options are required for the *ABB RPC S Integrated* option:

- *Arc*
- *Standard I/O welder*

14.4.1.5 Fronius TPS Integrated [650-12]

General

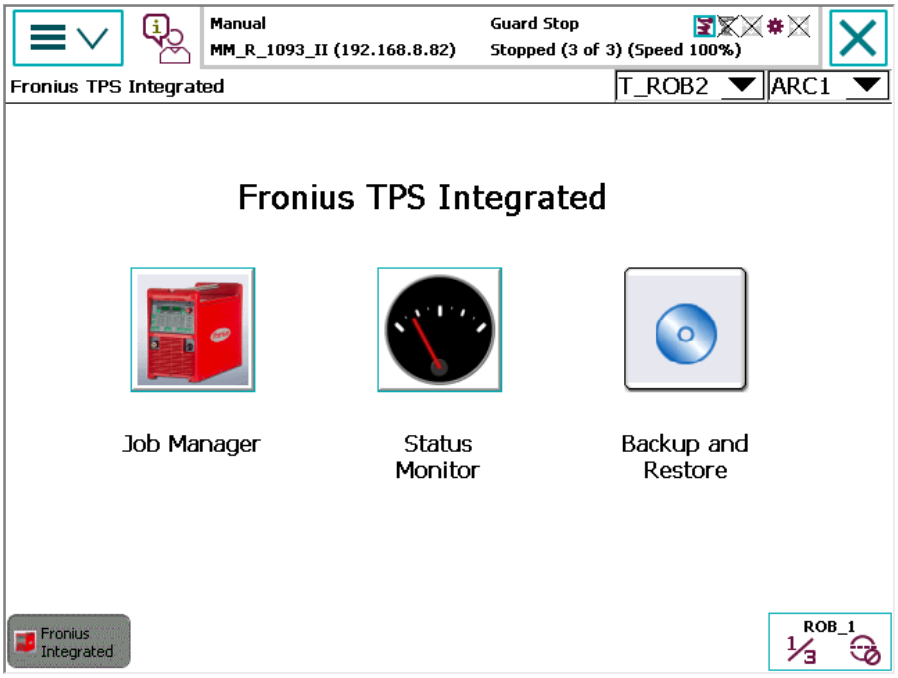
The option *Fronius TPS Integrated* is a programming and administrative interface for Fronius power sources on the FlexPendant.

Features

The *Fronius TPS Integrated* option has the following functionality:

- Administrate power source jobs.
- Monitor power source runtime information.
- Backup and restore of jobs defined in the power source.

Application



xx1400002647

Requirements

The following RobotWare options are required for the *Fronius TPS Integrated* option:

- *EtherNet/IP Scanner/Adapter*

The power source has to meet the following requirements:

- Fronius power source TPS 4000i/5000i with EtherNet communication support
- EtherNet/IP communication interface with software version 1.07.9 or later.
- PowerMAG-OS version 4.3018 or later.

Limitations

The option *Fronius TPS Integrated* cannot be used together with the Fronius external programming device RCU5000i.

14.4.1.6 Lincoln ArcLink [650-13]

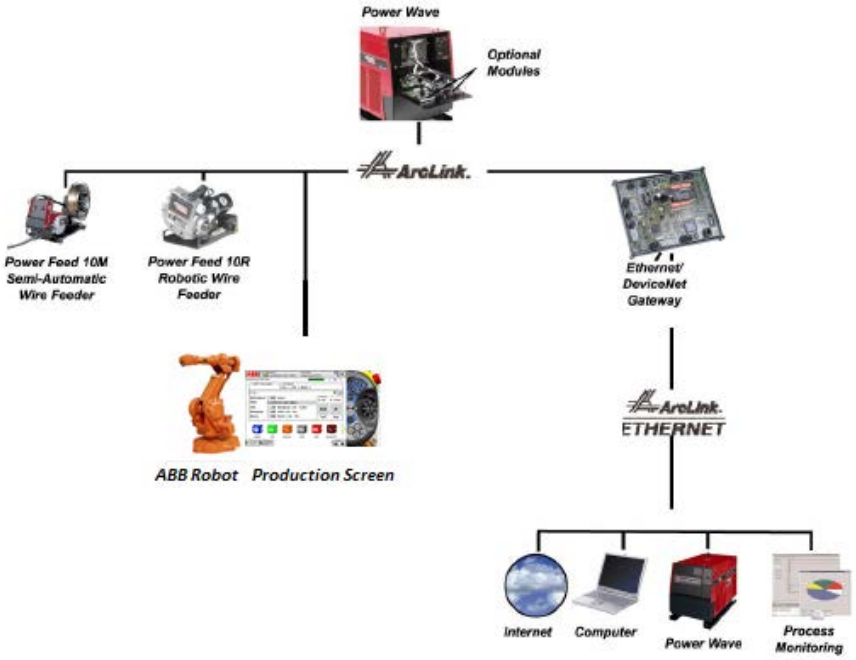
General

Lincoln ArcLink is digital communications system used in Lincoln Electric Power Wave welding systems.

ArcLink is specially designed for the arc welding environment and is the best method to communicate to cell components such as semiautomatic wire feeders or robots. ArcLink is simply the way each piece "talks" to each other in a Power Wave system. The Power Wave welding systems also provide "gateways" to allow other digital networks such as Ethernet to connect to the ArcLink system.

ArcLink integrates all welding components for seamless, time-critical data transfer. The strength of ArcLink lies in the ability to communicate with each system component in a pre-defined welding language. ArcLink brings modularity to welding systems and provides a single, intelligent connection between all modules.

ArcLink also allows for networking capabilities through gateways (such as Ethernet) to allow simultaneous monitoring of multiple welding cells, which establishes a means for developing supervisory or monitoring tools.

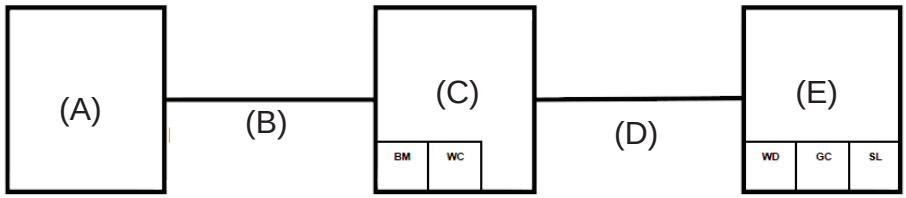


xx1400002223

Continues on next page

14.4.1.6 Lincoln ArcLink [650-13]
Continued

An ArcLink bus can consist of a CAN and Ethernet bus together; some nodes on CAN and others on Ethernet as in the figure below. ArcLink/XT adds Ethernet capability to the ArcLink CAN bus.



xx1400000529

Pos	Description
A	Robot
B	Ethernet - UPD (ArcLink/XT)
C	Power Source
D	CAN (ArcLink)
E	Wire feeder

Working principle

ArcLink uses peer-to-peer, event-driven messaging, which broadcasts on an “as-needed” basis.

Peer-to-peer messaging frees space on the data highway, reducing message traffic. Because ArcLink logic is based on a Controller Area Network (CANIIB) standard; it inherits the benefits of priority-based, deterministic messaging and noise immunity. This means that critical messages are guaranteed to reach their destination with minimum delay, even in the presence of electrical noise.

Features

File transfer

Weld files can be saved, shared, or modified.

Data acquisition

Weld data is collected and stored for Production Monitoring purposes.

Advanced Diagnostics

ArcLink provides the capability to monitor component failures system wide.

Ease of Installation

ArcLink uses one control cable for communications, component power, and electrode voltage sensing connections. ArcLink is used in the control cable connection between the weld power source and the wire feeder. It provides flexibility of physical configuration so that any component may be connected to any other component in the system.

ArcLink/XT uses one Ethernet cable for communication between the robot controller and weld power source.

Continues on next page

Power Wave

The Power Wave welding systems are extensively used in Robotic Arc Welding applications. It is provided with a number of PC tools, ranging from diagnostics to advanced production monitoring. In addition to this, Power Wave uses ArcLink, a digital communications system which was specially designed for arc welding environment. The Power Wave welding systems also provide “gateways” to allow other digital networks such as DeviceNet and Ethernet to connect to the ArcLink system. The unit is not supplied by ABB Robotics.

xx1400000549

Requirements

The following RobotWare options are required for the *Lincoln ArcLink* option:

- *Arc*
- *PC Interface*, (necessary for *Socket Messaging*)
- *Production Manager*
- *Production Screen*

14.4.2 Additional Arc System

14.4.2.1 Two additional [651-2]

General

The *Additional Arc Systems* software option is designed to prepare a robot system to control two or three arc systems. An example of that could be to control a MIG/MAG process equipment together with a TIG process equipment through one robot. In order to prepare the software communication to control two or three arc systems with the same robot, ABB has developed software options, in which one or two additional process equipment can be defined.

14.4.3 BullsEye

General

BullsEye™ provides completely automated Tool Center Point (TCP) definition for welding robots controlled by the IRC5 robot controller. By checking and updating the TCP alignment at regular intervals the robot will always operate with an accurate TCP.

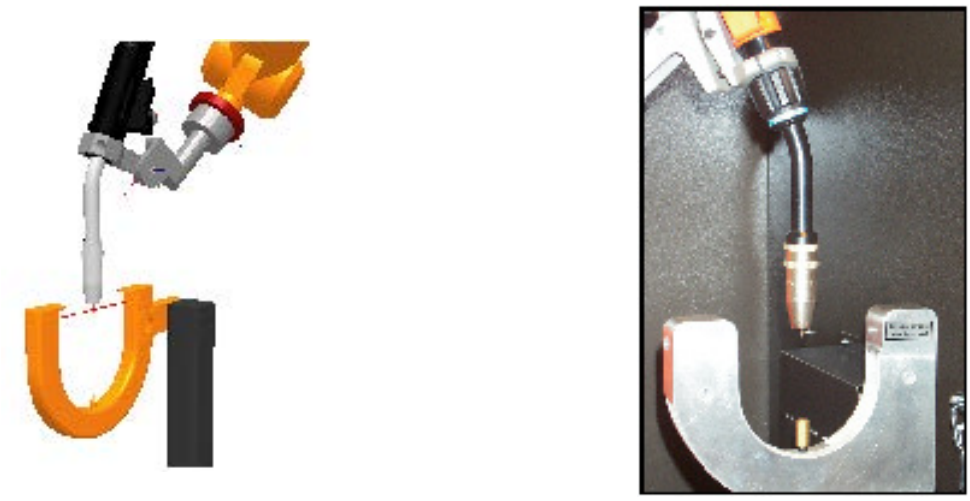
The robot moves the welding wire across an optical beam in several different orientations and the TCP is calculated through triangulation. At designed intervals, the robot can be programmed to zero in on BullsEye, do a quick check in about 10 seconds, and go back to work. If the TCP is misaligned BullsEye automatically recalculates TCP and torch angle.

Continues on next page

14.4.3.1 BullsEye [652-1]

General

The BullsEye software option will give the user the possibility to define and program a standalone BullsEye hardware device, see below pictures of different available hardware. The BullsEye arc option includes software that is loaded into all arc welding motion tasks, which must be involved when programming a TCP (Tool Center Point) check activity in a robot system.



xx1000000343

Pos	Description
A	North American version
B	European version

Above standalone hardware versions are available in separate Application product price lists.

The following main tasks can be handled from the BullsEye software options:

- Definition of tool data.
- Definition and programming of QuickCheck™ procedures.
- Definition and programming of full scan sequence and update the tool data.
- Programming of extended Electrode Extension (Wire Stick-out) on existing tool data.
- Instruction to view the deviation in tool data that has been checked over time.

14.4.4 Torch Service Equipment

General

The *Torch Service Equipment* software option provides automatic cleaning of the welding torch. It automatically clears and reduces buildup of spatter, which ensures continuous and reliable operation of the robot cell. The *Torch Service Equipment* option includes software that is loaded into all arc welding motion tasks.

> **Note**
>
> The *Torch Service Equipment* software loads unique equipment drivers, but the RAPID instruction interface is the same regardless of hardware selected.

Torch Service Equipment software maximizes arc on time and minimizes down time since it can be programmed to clean automatically between weld or station interchanges. This reduces the need to manually disassemble and clear the nozzle. *Torch Service Equipment* software makes it possible to perform torch cleaning during positioner interchange, which reduces cycle time. Once *Torch Service Equipment* is programmed the operator can decide how often the cleaning process needs to take place to prolong the contact tip and gas nozzle life.

Continues on next page

14.4.4.1 ABB - TC 2013 [653-1]

General

This software option will give the user the possibility to define torch cleaning and anti spatter procedures in a robot system for the hardware option, *ABB - TC 2013*. The hardware option is available in separate application product price list.

The following main tasks can be handled from the *ABB - TC 2013* software options:

- Procedure for automatic cleaning of gas nozzle interior wall and the exterior of the contact tip and face of the gas diffuser for quick removal of weld spatter.
- Procedure for anti spatter spray program to provide longer interval between torch cleaning.
- Procedure for define automatic wire cutting for consistent wire electrode extension (wire stick-out).

14.4.4.2 Binzel - TC97 [653-2]

General

This software option will give the user the possibility to define torch cleaning and anti spatter procedures in a robot system for the hardware option, *Binzel - TC97*. The hardware option is available in separate application product price list.

xx1000000345

The following main tasks can be handled from the Binzel - TC97 software options:

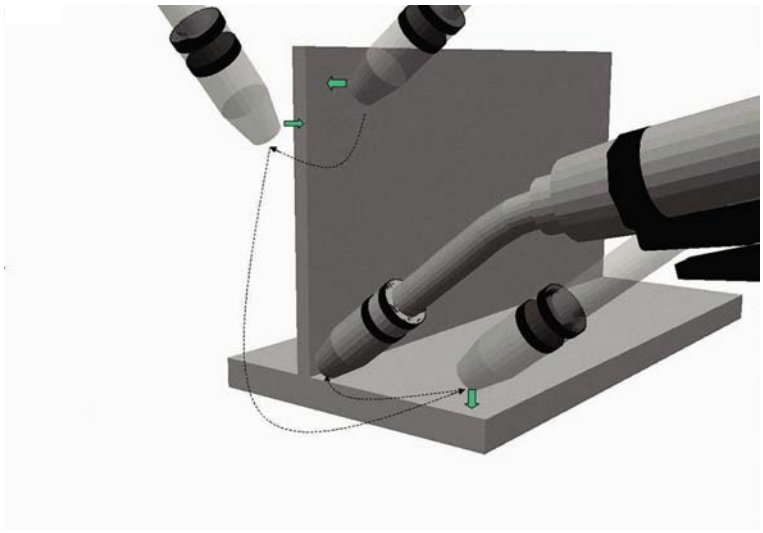
- Procedure for automatic cleaning of gas nozzle interior wall and the exterior of the contact tip and face of the gas diffuser for quick removal of weld spatter.
- Procedure for anti spatter spray program to provide longer interval between torch cleaning.

14.4.5 SmarTac - I/O version [657-1]

General

This software option will give the user the possibility to define search procedures in a robot system for the American hardware option. The hardware option is available in separate Application product price list.

The SmarTac™ software option is designed to control an electrical tactile sensor for locating weld joint positions. It uses the standard gas nozzle on the torch as a sensor. Several search instructions are included, which enable you to shift welds based on search results in run-time.



xx1000000347

Features

With SmarTac a part feature may be “searched” using part of the torch. Typically the welding wire or the gas cup is used as the sensing portion of the torch. Searches are programmed into a weld sequence. Each search consists of two robtargets; one for the start location and one for the expected location of the part feature. While searching the torch feature (gas cup or wire) is energized with about 38VDC. When the torch feature makes contact with the part (at ground potential) an input is set in the robot controller. When the input is detected, robot location is stored and motion stops.The Search instructions included in the SmarTac software are designed to return “offset” information. In other words, the result of a search is the distance between where the original search location was programmed and where the robot has now found the part.Using SmarTac effectively can dramatically reduce fixture costs. It can also help account for part variability that cannot otherwise be controlled.

Continues on next page

Hardware

The main component is an electronic sensor board, which detects contact with the part feature to be located. The SmarTac board can be ordered as an add-on unit and is installed in the robot cabinet. Different hardware is used on the European and North American markets, but the functionality is identical.

Note

Note! The hardware options are not included in below specified software option.

14.4.6 Production Monitoring [659-1]

General

The *Production Monitoring* option enables logging of production information in a WebWare Server database. The database is stored on a WebWare Server and is displayed on a regular web page through the WebWare Client. Access to this information will help users improve part quality and production throughput.

Features

The *Production Monitoring* option provides detailed, automated data collection for every weld seam on each part produced. This feature has driven standardization of process measurement to a new level, by proactive manage welding production rather than just react to problems.

Key features include:

- Online reporting of production data and faults
- Logging of production data for part traceability
- Real-time reporting via standard web browser
- Fixed cost deployment without added software development expense

The *Production Monitoring* option provides a systematic way to capture the production data needed to achieve continuous improvement goals.

Typical applications include:

- Cycle time analysis
- Tracking production output
- Quantify lost production time
- Monitor equipment utilization rates
- Archiving of production data for traceability
- Data capture to document acceptance criteria

WebWare tables in Production Monitor

Production Monitor promotes WebWare Server tables that contain statistical results. For example, RobotWare Arc will provide a table of *Cycle Results* that combines information from `CycleEvents` data with `SeamResults` data to create the *Cycle Result* table.

Event Tables:

- Event tables provide specific, time-stamped information that occur in the system. These include start and end events for cycles, service routine calls and setup routine calls. These tables are characterized by having records for start events and end events. To determine the duration of an event from start to end requires calculations based on the data stored in the database.

Result Tables:

- Result tables contain data that is calculated on the fly within RAPID. These tables provide useful information about production without making SQL queries from multiple tables. These tables are typically written after an event has finished and usually include a Duration column that provides the time

Continues on next page

that elapsed during the event. Often the columns included in these tables will contain fields that are accumulated over the duration of an action.

Event Table

Every component that writes to tables in the WebWare database includes a column for an Event ID. This number is incremented by the underlying Execution Engine each time the engine executes an event. The Events include the execution of a part cycle, a service routine or a setup routine. When the underlying structure reacts to a command, it assigns an index value to that event and makes that variable available to all components. The component includes that Event ID in its database table so that SQL queries may be made based on that key. For example, when the underlying structure reacts to a command, a start event is generated and logged in the event table.

The event table contains the records for all responses from requests to the Execution Engine.

These include part cycle calls, service routine calls and setup calls. Records will be added for all start and end events. An EventType field will be used to differentiate between part cycle calls, service routine calls and setup routine calls.

Column Name	Data Type	Description
EventID	Long Integer	A number supplied by the Execution Engine. May be used as a key
EventType	Integer	Enumeration for Cycle, Service and Setup types
Condition	Integer	Enumeration for Start or End
UserID	string	User that is logged-in
RobotID	string	Task name
ControllerID	string	Controller Name
Time	DateTime	Time stamp

Cycle Events table

The *CycleEvents* table contains the records for all part cycles executed by the Execution Engine. A CycleID field will be used as a serial number for parts. This number may be provided to the underlying structure from an external device, or by simple increments in the Execution Engine.

Column Name	Data Type	Description
EventID	Long Integer	A number supplied by the Execution Engine. May be used as a key
CycleID	Long Integer	A number supplied by the Execution Engine. May be used as a key
Condition	Integer	Enumeration for Start or End
Part	string	Path to, and name of, part procedure
PartDescription	string	From `partdata` description

Continues on next page

14.4.6 Production Monitoring [659-1]
Continued

Column Name	Data Type	Description
PartCount	Integer	Count at time of event. A 'Start' condition will show the part count before running the part. An 'End' condition should show an incremented PartCount.
Station	Integer	From `partdata` station
UserID	string	User that is logged-in
RobotID	string	Task name
ControllerID	string	Controller Name
Time	DateTime	Time stamp

Seam Results Table

The SeamResults table contains a record for each weld seam that is finished. It provides information about the seam from RobotWare Arc. The value in the CycleID field will match the value in the CycleEvents table. This number may be provided to the underlying structure from an external device, or by simple increments in the Execution Engine. The value will be '0' if the seam was not executed within the context of a cycle. Cycle information from the Execution Engine will be included in the table, if those values are available. Otherwise the fields will be set to '0' or "-" depending on the data type.

Column Name	Data Type	Description
EventID	Long Integer	A number supplied by the Execution Engine. May be used as a key
CycleID	Long Integer	A number supplied by the Execution Engine. May be used as a key
Part	string	Path to, and name of, part procedure
PartDescription	string	From partdata description
Station	Integer	From partdata station
SeamName	string	Name supplied in Arc instruction
ArcStartDuration	number	How long it takes to ignite arc
NominalArcStart	number	Nominal length of time to ignite arc
SeamLen	number	Length of actual weld completed for the seam
NominalSeamLen	number	Saved nominal length of seam
Duration	number	Time in seconds to complete seam
NominalDuration	number	Saved nominal time in seconds to complete part
ArcStarts	Integer	Number of arc starts for the seam - ideally 1
Stops	Integer	Number of stops during welding for any reason
Completed	boolean	True if all welds finished to completion
UserID	string	User that is logged in
RobotID	string	Task name
ControllerID	string	Controller Name

Continues on next page

Column Name	Data Type	Description
Time	DateTime	Time stamp

Cycle Results Table

The *CycleResults* table contains a record for each part cycle made by the Execution Engine. It provides more information about the cycle based on information provided by RobotWare Arc. The value in the CycleID field will match the value in the CycleEvents table. This number may be provided to the underlying structure from an external device, or by simple increments in the Execution Engine. The underlying structure provides procedure hooks that allow Process Applications like Arc to be informed about cycle events. This table is written by RobotWare Arc. Other process applications may have similar, but slightly different, CycleResults tables.

Column Name	Data Type	Description
EventID	Long Integer	A number supplied by the Execution Engine
CycleID	Long Integer	A number supplied by the Execution Engine
Part	string	Path to, and name of, part procedure
PartDescription	string	From partdata description
PartCount	Integer	Count at time of event. A ‘Start’ condition will show the part count before running the part. An ‘End’ condition should show an incremented PartCount.
Station	Integer	From partdata station
WeldLen	number	Accumulated weld lengths for all welds in part
NominalWeldLen	number	Saved accumulated weld lengths for all welds in part
Duration	number	Time in seconds to complete part
NominalDuration	number	Saved time in seconds to complete part
Welds	Integer	Number of welds completed during the part cycle
NominalWelds	Integer	Saved number of welds completed during the part cycle
ArcStarts	Integer	Number of arc starts during the cycle
WeldStops	Integer	Number of stops during welding for any reason
Completed	boolean	True if all welds finished to completion
AccumArcTime	number	Total arc time for the robot
AccumArcStarts	number	Total number of arc starts for the robot
UserID	string	User that is logged in
RobotID	string	Task name
ControllerID	string	Controller Name
Time	DateTime	Time stamp

Continues on next page

14.4.6 Production Monitoring [659-1]
Continued

Requirements

Robot controller requirements	PC requirements
PC Interface option	WebWare Server 4.5 or higher. For complete PC requirements, see WebWare Server Administrator's Guide.

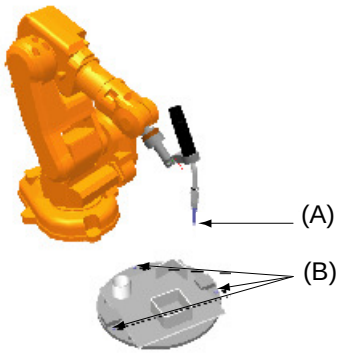
14.4.7 Navigator [814-1]

General

Navigator is a product designed to accurately perform calibration of external axes and fixtures. It consists of two main features, *Frame Definition* and *Coordinate Measurement*.

Frame Definition (WorkObject calibration)

Tooling balls are typically mounted on the fixture and the measuring probe is mounted on the robot. The probe TCP is defined using the BullsEye TCP calibration device.



xx1000000349

Pos	Description
A	Probe
B	Tooling Balls

The robot is then used to measure a number of reference points (tooling balls). Letting the robot locate tooling balls placed on the fixture performs the frame definition. The procedure can be totally automated and user independent. Thus, the manual step of pointing out reference positions for frame definition is replaced by automated search methods.

Coordinate Measurement (Robot CMM)

The main purposes for coordinate measurement with Navigator is to calibrate a series of fixtures, but can also be used for measuring part variations:

- A: Fixture line up. A master fixture is created and manually mechanically adjusted. The robot is then used to measure a number of reference points (surfaces). The points measured on the master fixture are called nominal points. Then the master fixture is removed and a clone fixture is mounted in the same robot cell. The robot is then ordered to measure the same reference points on the new not mechanically adjusted fixture. The referenced points are compared to the nominal points and a report created describing how the clone fixture should be mechanically adjusted to be an exact copy of the nominal. This procedure can be repeated for unlimited number of clone fixtures.

Continues on next page

14.4.7 Navigator [814-1]
Continued

- B: Production batch monitoring. Instead of measuring fixture points the nominal points are created on a master part. Subsequent parts/batches are thereafter checked periodically for deviation from the master part. The result is logged, which creates a possibility to automatically stop production.

Hardware

The *Navigator* hardware for frame definition consists of the measurement probe (for example gas nozzle equipped with a tooling ball) and mounting holes on the fixture for the tooling balls. This hardware is sold separately.

Expected performance

The repeatability accuracy of localization of a single sphere is in the same magnitude as the robots repeatability accuracy, which is estimated to 0.05 [mm]. The expected repeatability accuracy of calibration of the base frame for an external rotational axis is maximum 0.16 [mm] and an average of 0.09 [mm]. The expected maximum position deviation of an entire cell calibration including probe TCP calibration, external axis calibration and workobject/fixture calibration is 0.24 [mm] and the average position deviation is 0.13 [mm]. These numbers are based on empirical tests and should only serve as an estimate of expected performance. ABB cannot be held responsible for deviations from these values. For best performance the robot should be equipped with BullsEye TCP calibration and SmarTac tactile sensor.

Requirements

A tactile sensor (preferably SmarTac).

RAPID instructions

RAPID instructions included in this option:

Instruction	Description
SearchSpL	`SearchSpL` is an instruction for tactile measuring the position of the center of a sphere (for example tooling ball). The robot moves with linear motion to the starting point for the search pattern. From the starting point the robot moves in a downward spiral pattern until the probe touches the tooling ball. The pattern is repeated several times and the center of the tooling ball is calculated.
SearchSpJ	`SearchSpJ` is an instruction for tactile measuring the position of the center of a sphere (for example tooling ball). The robot moves with joint motion to the starting point for the search pattern. From the starting point the robot moves in a downward spiral pattern until the probe touches the tooling ball. The pattern is repeated several times and the center of the tooling ball is calculated.

Continues on next page

Instruction	Description
Measure_1D	`Measure_1D` is an instruction used for tactile measuring a point perpendicular to its plane. The robot will make a linear movement to the position `ApprPoint`. The search equipment will be activated and motion will start towards the point to measure, `NominalPoint`. The robot will continue past the search point for a maximum total search distance twice the distance between `ApprPoint` and `NominalPoint`. Once the feature is sensed, motion stops, and the distance between `NominalPoint` and search position is logged and displayed on the FlexPendant. If the argument `\UpdateNominal` is used the `NominalPoint` will get the value of the search.

14.4.8 Optical Tracking Arc [660-1]

General

The option *Optical Tracking Arc* is a software option that allows integration of external optical sensors, which can be used for real time seam tracking. For communication between the sensor and the robot controller two different communication links are available: serial link (RS 232) and Ethernet.

If the RS232 link is used, the sensors have to be adapted to the ABB standard application protocol, LTAPP. Sensors that are adapted to LTAPP are available from companies like Servo Robot, Meta, and Scout.

The Ethernet link is available for optical sensors from ServoRobot, which use the application protocol Robo-Com Light, defined by ServoRobot.

Features

Key features include:

- Easy calibration with dedicated FlexPendant interface and calibration plate, art. No. 3HAC032252-001. Contact ABB Robotics.
- Contour tracking integrated with arc welding movements.
- Adaptive process control during path motion and tracking.
- TCP/IP enabled to facilitate plug and play capability, using the Ethernet link.
- Separate sensors can be installed for each robot in a MultiMove system (if using serial link, the number of serial ports may be limited).
- Includes all features of the option Sensor Interface [628-1].

Performance

Designed to work for normal arc welding applications. Real performance, i.e. path following accuracy, is however depending on current speed, curvature, look ahead, sampling frequency etc.

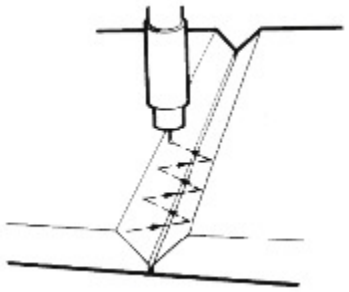
Requirements

For calibration, hardware option calibration plate [1250-1] is required.

14.4.9 WeldGuide [815-2]

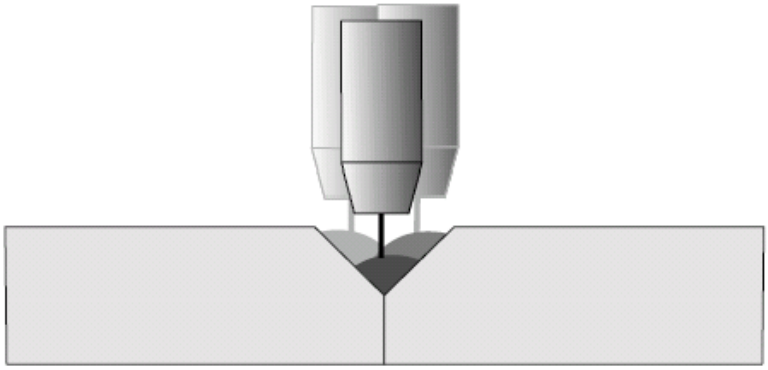
General

The WeldGuide, is a “Thru-Arc” joint tracking RobotWare arc option to be used together with the optional AWC (Advanced Weld Control) unit which is an optional hardware integrated into the robot controller. It is designed to track welding joint variations due to cast components or other pre-process problems and will in an automatic mode monitor track the weld joint during welding.



xx1000000350

Sometimes multiple weld passes are required due to the required weld size and thickness of the material being joined. WeldGuide makes this easy by tracking the first pass and storing the actual tracked path so it can offset for subsequent passes.



xx1000000355

Features

The main feature is the “Thru-Arc” tracking capability which divides in following specific software features.

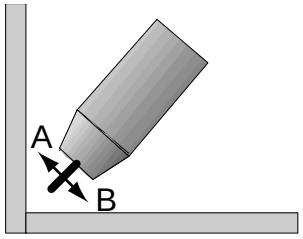
The first weld pass is recorded by making a weld with normal arc instructions. The interval between stored path points is dictated by the weave length. Multi-pass welding can be used in conjunction with seam tracking. The replayed path can be offset in either the plus or minus “Y” and “Z” seam coordinates and rotated plus or minus “X” and “Y” in seam coordinates. Replayed paths can also be executed in the forward or reverse direction. The start and end path points can be lengthened

Continues on next page

or shortened by a specified distance in millimeters. If the path is lengthened, the new end point is projected outward by using the last two points that were stored in the path. Lengthening and shortening the path allows for the weld to be tied into previous welds or the parent material itself.

Center line Tracking

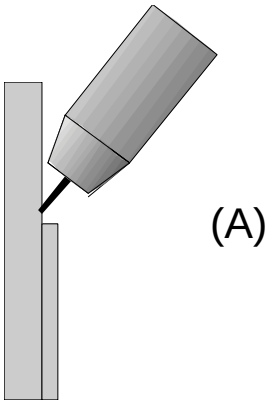
The WeldGuide software is controlling the measurement of current and voltage (impedance) at the respective end position of the robot weave width as the torch weaves across the weld joint. These measurements are in real time analyzed by the WeldGuide software which then will result in robot path adjustments to ensure that the arc stays in the weld joint along the whole weld path. See Figure below.



xx1000000351

Single-Side Tracking

The WeldGuide software can also control tracking from a single side, tracking data samples of just one wall, to follow the robot path during welding. In addition the amount of penetration in the side plate can be programmed. See Figure below.



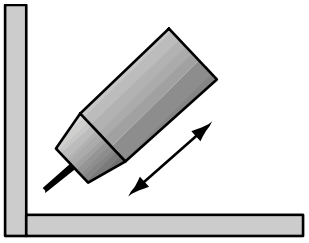
xx1000000352

Pos	Description
A	WeldGuide takes data samples of just one wall.

Continues on next page

Torch-To-Work Tracking

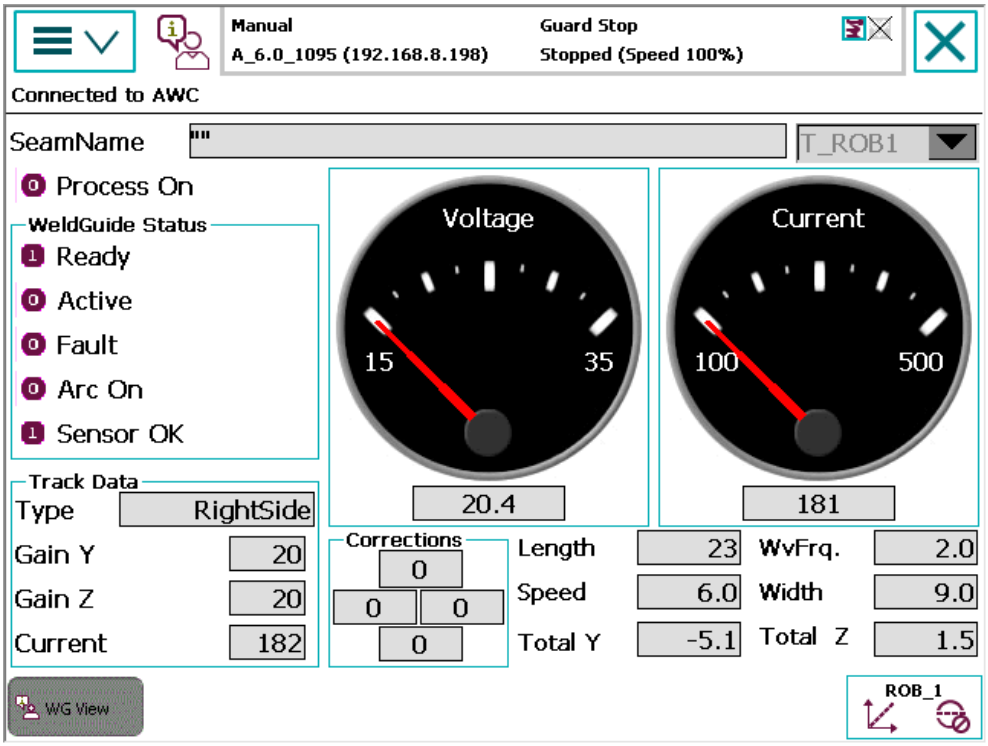
In Torch to Work mode using the WeldGuide software the same contact tip to work length is maintained. The contact tip to work distance is specified as voltage and current settings in the weld data. Weaving is not required for this type of tracking. But because the correction calculations are synchronized with the weave pattern, a weave with almost zero width is required. See Figure below.



xx1000000353

Operator user interface

The WeldGuide feature is easily activated through the operator FlexPendant unit, which is an easy to use graphical user interface. Through the FlexPendant the operator can access weld data, such as wire feed speed, voltage, travel speed. With this software option the operator will also get access to specific WeldGuide track data parameters, which is easily selected and adjusted from the display.



xx1000000354

Continues on next page

14.4.9 WeldGuide [815-2]
Continued

Requirements

The option WeldGuide for RobotWare Arc communicates within the robot controller through the RS232 serial port. The robot hardware option *[714-1] RS232 to 422 converter* is needed as the hardware AWC unit communicates through an RS484 serial connection. WeldGuide parameters are factory downloaded using the provided parameters disk, and the AWC Companion software installed on a laptop PC, following the hardware option delivery. Parameters can be transferred between the PC and the WeldGuide software through the serial port.

Note

Note: The RW Arc Option“ WeldGuide” does not include the AWC hardware option.

For more information see separate product application price list.

Limitations

The following limitations apply for WeldGuide (815-1 or 815-2) in a MultiMove system:

- If more than one robot is to be equipped with WeldGuide, multiple serial channels are needed. This is available on request only (based on the previous main computer).
- Multiple WeldGuides can not be used when the workpiece is held by a mechanical unit (robot or positioner). This limitation applies also when the workpiece is stationary during welding.

RAPID instructions

RAPID instructions included in this option:

For more detailed information see *Operating manual - Seam tracking with Weldguide III and MultiPass*.

14.5 Spot 6 [635-6]

General

The *RobotWare Spot* option is a general and flexible software platform for creation of customized and easy to use function packages for different types of spot welding systems and process equipment.

There are three different **main** Spot configurations supporting spot welding, two for servo guns and one for pneumatic guns. All Spot configurations provides dedicated spot welding instructions for fast and accurate positioning combined with gun manipulation, process start and supervision of the different gun equipment. The included instructions can be used for sequential welding with one or several gun equipment or for welding with up to four guns at the same time.

The Spot configurations are also general and can be customized extensively. They have a default “ready to use” functionality directly after installation and can be if needed, customized via process configuration.

Features

The Spot option package contains the following features:

- Fast and accurate positioning using the unique QuickMove and TrueMove concept
- Default “ready to use” functionality directly after installation
- Gun pre-closing, i.e having the gun closing synchronized with weld position
- Gun equalizing, that is having the gun “floating” around the weld position
- Reverse execution with gun control
- Constant tip force during welding for servo guns
- Multiple gun forces during welding for servo guns
- Detecting of missing or improper plates for servo guns
- Gun calibration functions and tip wear counter for each used servo gun
- Fast switch between two servo guns with a tool changer (This feature requires the Servo Tool Change option)
- Weld error recovery with automatic rewelding
- Quick start after a weld
- Manual actions for welding and gun control
- Several simulation possibilities for test purposes
- Wide customizing possibilities
- A dedicated Spot GUI on the FlexPendant
- Integrated weld timer fault management (for options 782-X Bosch Weld Timer)
- Predefined Spot configurations, pneumatic, servo and servo equalizing configuration

Continues on next page

SpotWare Servo Equalizing Features

For the Spot Servo Equalizing configuration the gun equalizing features are embedded in the Spot Servo process software. Spot Servo Equalizing allows you to solve spot welding gun equalizing without mechanical equalizing hardware on the weld gun and thus provides an opportunity to reduce investment cost and improve productivity. Removed hardware on the gun also reduces weight, and in addition, no gravitational influence means easier optimizing when programming the robot path.Furthermore, with the Spot Servo Equalizing software instead of mechanical equalizing, less spare parts are required and lower maintenance costs are achieved. Spot Servo Equalizing is valid for:

- Robot mounted C- and X-type guns
- Pedestal C- and X-type guns
- Tool changing between guns with different equalizing data
- Weld position touch up - Simplifies adjustment of the programmed weld positions.
- Release of the fixed gun arm - Automatic gun arm release from the welded sheets when the robot is moving among weld points.
- Gun arm deflection compensation - The robot program compensates for the gun arm deflection automatically.
- Tip wear measurement and compensation - The robot program automatically compensates the weld gun TCP (Tool Center Point) for tip wear without using external sensors.
- The equalizing type (mechanical or software) is determined by data set in the gundata for each used gun.
- SoftMove Equalizing - The robot will be set to a "soft state" in the tool z-direction during the gun closing.

Application

The RobotWare Spot option can be used in any spotwelding process. The robot’s movement and the spotwelding process are controlled from the instructions SpotL/SpotJ and SpotML/SpotMJ.

The spot welding process is specified by:

- Spot and equipment specific data types
- Spot and equipment specific process configuration
- The I/O configuration
- RAPID routines and global data for customizing purposes
- User modules

RAPID instructions

Included RAPID instructions:

Instruction	Description
SpotL	Control the motion, gun closure/opening and the welding process.Move the TCP along a linear path and perform a spot welding at the end position with up to four robots at the same time.

Continues on next page

Instruction	Description
SpotJ	Control the motion, gun closure/opening and the welding process.Move the TCP along a non-linear path and perform a spot welding at the end position with up to four robots at the same time.
SpotML	Control the motion, gun closure/opening and 1 - 4 welding processes.Move the TCP along a linear path and perform spot welding with 1 - 4 gun equipment at the end position.
SpotMJ	Control the motion, gun closure/opening and 1 - 4 welding processes.Move the TCP along a non-linear path and perform spot welding with 1 - 4 gun equipment at the end position.
IndGunMove	Set the servo gun in independent mode and thereafter move the gun to a specific independent position.
IndGunMoveReset	Reset the independent mode for servo gun.
SetForce	Close the gun a predefined time then open the gun.
CalibL	Calibrate a servo gun during linear movement to the programmed position.
CalibJ	Calibrate a servo gun during non-linear movement to the programmed position.
Calibrate	Calibrate a servo gun in current position without movement.
OpenHighLift	Open a pneumatic gun to the high lift position (large gap).
CloseHighLift	Close a pneumatic gun to the work stroke position (small gap).
MeasureWearL	Measures the tip wear and recalculates the TCP. Only Spot Servo Equalizing.
ReCalcTCP	Calculates the tip wear and recalculates the TCP. Only Spot Servo Equalizing.

Limitations

- Spot Servo Equalizing can only be applied during sequent welding (not with several guns simultaneously, SpotML).
- RobotWare Spot can be applied to only one robot in a MultiMove system, on request, not standard.
- The SoftMove function in Spot Equalizing has been validated on the IRB 6640/6700 robot types; other robot types are not yet validated. The SoftMove performance may depend on the specific robot type.

14.6 Bosch Interface [832-1]

General

The Bosch interface option is used to simplify programming and monitoring of the Bosch weld timer. The option includes a robot FlexPendant interface that could be used for setting or editing weld parameters, monitor weld status and give easy access information if weld error occurs.

The option is integrated as an add on functionality to the RobotWare Spot option:

- Option 635-6. Spot Welding

Features

The features within the option are designed to support needs during normal production conditions. Some examples of useful functions are given below:

- Features to set or edit the weld parameter
- Information about weld errors plain text and also proposal of recovery steps
- Pre-warning of electrode wear
- Information of last performed weld

Weld timers

The option is prepared to be used in combinations with the following weld timers:

Weld timer	Description
Bosch PST 6100.630	Option 782-1, Bosch Devicenet AC
Bosch PSI 6100.630	Option 782-7, Bosch Devicenet MFDC
Bosch PSI 61C0.751	Option 782-11, Bosch Profinet MFDC

The option includes both software as well as necessary additional cabling between the weld timer and the robot controller.

Limitations

This software does not replace the need for the PC software BOS5000/6000 wich is needed during setup, and to backup parameters etc.

Advanced setup and configuration can not be performed from this software, this software is a complement to the BOS5000/BOS6000 PC software.

Before using the Bosch interface option some necessary setup is needed. This is done by using a Bosch specific PC software (BOS5000/BOS6000). The PC software will also be required if extra ordinary weld faults (hardware faults) occur.

14.7 Dispense [641-1]

General

The *RobotWare Dispense* option provides support for different types of dispensing processes such as gluing and sealing. The option provides fast and accurate positioning combined with a flexible process control. Communication with the dispensing equipment is carried out by means of digital and analog outputs. *RobotWare Dispense* is an option that can be extensively customized.

Features

- On/Off gun support
- Proportional gun support
- Handles up to 4 gun equipment in the same program
- Fast and accurate positioning and process control
- Dispensing in wet or dry mode
- Restart an interrupted dispensing sequence
- Speed proportional AO
- Different anticipated times for the DO and AO
- Equipment delay compensation for the TCP speed proportional analog signals
- Global or local flow rate correction factors
- Automatic reduction of the robot acceleration/deceleration during dispensing
- Wide opportunities of customizing the functionality to adapt to different types of dispensing equipment
- A dedicated Dispense GUI on the FlexPendant
- RobotWare Dispense can also be used in MultiMove system, where up to four dispensing robots can be controlled from the same IRC5 system.
- Dispense robots can be combined with other process application robots in one MultiMove system.

Application

RobotWare Dispense can be used in any gluing or sealing process.

The robot's movement and the dispensing process are controlled from the instructions DispL and DispC.

A dispensing process is specified by:

- Bead specific dispensing data
- Equipment specific dispensing data
- Equipment specific process configuration
- RAPID routines and global data for customizing purposes
- System Module DPUSER
- The I/O configuration

Continues on next page

14.7 Dispense [641-1]
Continued

RAPID instructions

RAPID instructions included in this option:

Instruction	Description
DispL	Move the TCP along a linear path and perform dispensing with the given data
DispC	Move the TCP along a circular path and perform dispensing with the given data

14.8 Prepared for PickMaster

14.8.1 PickMaster 3 [642-1]

General

Prepared for PickMaster/PickMaster 3 supports the PickMaster 3 application software residing on a PC. *Prepared for PickMaster* features together with the *PickMaster 3* application configurable integration of robots, vision and conveyors. The option contains all necessary functions and further options to interface PickMaster 3 and to run enhanced conveyor tracking. PickMaster 3 is approved for running on one IRC5 controller with multiple robots. For further information see *Product Specification - PickMaster 3*.

Feature

- Dedicated data types and instructions for efficient data transfer between PC and controller
- Built-in enhanced conveyor tracking capability with capability to control up to six conveyors. Maximum approved conveyor speed is 1400 mm/s.
- Enables run-time cell control by PickMaster with fully autonomous application startup
- Supports minimal parameterized RAPID code downloaded from PickMaster at startup
- Boundary checks and start/stop control on conveyors
- High performance camera trigger synchronization to fixed work areas and/or conveyor frames. Maximum eight work areas in total.
- Type and quality selection
- Advanced functions for mixing and sorting
- Approved for running two MultiMove robots on one controller.

Includes RobotWare options

- Advanced Rapid [626-1]
- Fixed Position Events [609-1]
- Tracking functionality included
- PC-Interface [616-1]

Application

Aimed for packaging applications, typically random high-speed picking and packing on the fly. Further, PickMaster 3 is the ideal standard vision integration for all robot types in material handling and machine tending applications.

Performance

See *Product Specification - PickMaster 3*.

Continues on next page

14.8.1 PickMaster 3 [642-1]
Continued

Requirements

Each robot operated by PickMaster requires the option *Prepared for PickMaster*.

- At least one *Digital 24 VDC I/O [716-1]* or *AD Combi I/O [717-1]*
- One encoder interface unit [726-1] per tracking process
- Not together with *Conveyor Tracking [606-1]*

PickMaster 3 user manual

The PickMaster user manual describes the application as well as related RAPID data and instructions in detail.

RAPID data types	RAPID Instructions
itmsrc	AckItmTgt
itmsrcdat	CreateInstDat
itmtgt	FlushItmSrc
selectiondata	FreeAllItmSrc
sortdata	FreeItmSrc
	GetAllItmSrc
	GetItmTgt
	ItmSrcByName
	NewItmSrc
	NextItmTgtType
	QstartItmSrc
	QstopItmSrc
	ResetItmSrc

14.8.2 PickMaster 5 [642-2]

General

Prepared for PickMaster/PickMaster 5 works with the offline PickMaster 5 application software residing on a PC. It features a flexible ease of use application software for palletizing. *Prepared for PickMaster/PickMaster 5* operates alone in runtime independently on the controller as a self-contained single cell application. The application is operated by the FlexPendant PickMaster 5 application. The option contains all necessary functions and further options to interface PickMaster 5 and to run the application. PickMaster 5 is approved for running on one IRC5 controller with one single robot. For further information see *Product specification - PickMaster 5*.

Features

PickMaster 5 offers advanced automated pelletizing functions for multi-station cells and mixed pallet loads. It is configured and commissioned through a PickMaster 5 PC-application residing on a standard laptop computer. Dedicated data types and instructions for efficient template based RAPID execution and application information exchange. Open code for user enhancements. Dedicated FlexPendant application for launching and controlling PickMaster 5 projects.

- Fully autonomous operation without online PC-connection
- Dedicated pelletizing production panel on the FlexPendant.
- Parametrized fly-in and –out trajectories
- Stack search functions
- Safety height calculations
- Built-in automatic application sequencer logics
- I/O inter connectivity to station controls production run orders, and equipment.
- Integrated control of generic multiple zone grippers addressing both vacuum and mechanical grippers.
- Approved for running one robot per controller Applicable on both 4 and 6 axes robots

Includes RobotWare options

- Advanced Rapid [626-1]
- Fixed Position Events [609-1]
- FlexPendant Interface [117-1]
- PC-Interface [616-1]

Application

Aimed for pelletizing applications.

Performance

See *Product specification - PickMaster 5*.

Continues on next page

14.8.2 PickMaster 5 [642-2]
Continued

Requirements

Each robot operated by PickMaster 5 requires the option *Prepared for PickMaster/PickMaster 5*.

- To run a palletizing application, at least one I/O device is needed to exchange signal information between the robot controller and the PLC.
- Not together with MultiMove [604-1] or [604-2]

PickMaster 5 user manual

The PickMaster 5 user manual describes the application as well as related RAPID data, functions and instructions in detail.

RAPID data types	RAPID Instruction	RAPID Functions
pm_accdata	PmAckTarget	PmCalcIntermid
pm_acktype	PmCalcArmConf	PmGetEvent
pm_actiondata	PmErrorLog	PmGetTarget
pm_actionhandle	PmGetOperation	PmGetAction
pm_actiontype	PmGetWaByWobj	PmGetWaHeight
pm_eventdata	PmStartProj	
pm_eventtype	PmWaitProjStart	
pm_movetype		
pm_offsetdata		
pm_operationdata		
pm_ophandle		
pm_singareatype		
pm_targetdata		
pm_targethandle		
pm_wadescr		

14.9 Force Control Base [661-2]

General

Force Control Base is an option that contains the RAPID instructions required to run force control. The option contains what was previously the option *Assembly FC* and the RAPID layer of the option *Machining FC*. The *Machining FC* GUI is still a separate option, *Machining FC GUI [877-1]*.

Assembly FC

Assembly FC (Force Control) is a set of functions, which will highly facilitate the use of robots for assembly tasks. Assembly is a very demanding application where parts with very small tolerances shall fit together, e.g shafts into gear wheels. Since tolerances are in the size of robot repeatability, this normally requires high accuracy fixtures and robots with very high precision and still the risk for position failure, where parts are stuck or damaged, is very high. In many cases also parts are not fixed but can have random positions, and thus traditional position controlled robots are not possible to use.

With *Assembly FC* and a 6-DOF Force/Torque sensor the robot is equipped with "tactile" sense and thus can handle the parts like a human worker, that is search along a predefined pattern and try pushing until parts slip into position with only small contact forces used. If measured force gets larger than desired the robot would move back so as to decrease the force. This will save both installation cost and programming time but also reduce the process cycle time.

Other areas where *Assembly FC* is useful are:

- Product testing - apply the same amount of force to a product repetitive
- Part mating
- Automated fastening

Assembly FC includes functions to use different predefined assembly patterns, designed for different tasks, but also possibility to combine some of them to create new patterns for more complex assembly tasks.

Machining FC

Machining FC is a set of functions, which will highly facilitate the use of robots for machining applications. *Machining FC* includes solutions for using force control in different machining applications where the robot needs to be sensitive to process forces. *Machining FC* offers two kinds of process control, optimized for different kinds of machining, namely *FC Pressure* and *FC Speed Change*.

FC Pressure functionality

With the FC Pressure functionality the robot will always keep a constant force against a surface, resulting in that the robot will follow the curvature of the surface.

The FC Pressure functionality is for example advantageous to use in machining processes like grinding and polishing where the robot needs to obtain a defined pressure (force) between the tool and part for proper result. With FC Pressure functionality, the programmed path roughly needs to follow the surface as the robot

Continues on next page

position will be adjusted to achieve the defined contact force between tool and part. This will save programming time as well as improve the product quality.

FC Speed Change functionality

FC Speed Change will allow programming at the maximum process speed and automatically slow down the robot when the machining forces are too high.

The FC Speed Change functionality can for example be favorable to use in a machining process like deburring, where the robot needs to slow down when hitting excessive burr, or even retract for sequential machining. The robot is position controlled, that is it will follow the programmed path, in this functionality. With the FC Speed Change functionality the robot will vary the processing speed. This will save programming time but also reduce the process cycle time.

Functional description

Normally robots are position controlled, which means that the robot is forced to move to a certain ordered position. If the robot is prevented to reach the target, the servo will increase power until max torque or collision is detected.

With *Assembly FC* it will be possible to let the robot search along a surface while keeping a predefined contact force. Once the mounting hole is found, the part will “fall” into place and the robot will push the part in to the opening until correct position is reached.

A typical assembly task will for example be the mounting of axis and gear wheels in a clutch. Such a task will include several different movements to find the correct location and insert the axis. For any assembly task it will be possible to tailor the best optimum search pattern/movement and choose between a number of criteria for accepting/ending the operation.

With *Force Control for Machining* it will be possible to either let the robot maintain a constant force perpendicular to the path while following the surface curvature (FC Pressure) or let the robot slow down on the pre-defined path to reduce the force acting on the robot (FC Speed Change).

About FC Pressure

The purpose of FC Pressure is to make the robot sensitive to contact forces perpendicular to the move direction. The robot can “feel” its surroundings and follow the surface of the processed part to obtain a certain pressure against an object. This means that the robot will change its position in order to apply a constant force/pressure on a surface, even if the exact position of the surface is not known. Since pressure is obtained by moving the robot path, this function is more suited for polishing, grinding and cleaning, where a surface should be made even and smooth. The material that is removed and the changes of the surface topology / dimensions depends on the process parameters like tooling, applied pressure, robot speed etc.

See below some examples from foundry and metal fabrication where FC Pressure is useful:

- Grinding of faucets
- Polishing of kitchen sinks

Continues on next page

- Deflashing and cleaning of castings
- Deburring of castings

About FC Speed Change

In processes where path accuracy is important and where the finished result shall comply with specific dimensions, FC Speed Change is recommended. This function will be useful combined with force sensor or other input indicating excessive forces, which can negatively affect the finished result. When speed change is active and if machining forces exceed a certain value, then the path speed will automatically be reduced, thus decreasing forces, minimizing changed dimensions due to deflections of the robot arm and most probably avoid damaging the part/tool due to stress and heat. This will guarantee path accuracy even if much material shall be removed. Some examples where FC Speed Change may be used:

- (Wheel) Grinding unevenly distributed material on casted surfaces
- Milling along the edge of a work piece
- Deburring along contour of a work piece
- Deflashing unevenly distributed burr along a part line on castings
- Deburring of castings

Assembly FC features

- Sensor calibration and load identification, which will calibrate the force sensor to remove offsets such as the gravity force. RAPID instructions: `FCCalib, FCLoadId`
- Activation and deactivation of Force Control. When activating Force Control for instance the following can be set up.
- Force control coordinate system
- Damping, that is how the robot speed depends on the contact force. RAPID instructions: `FCAct, FCDeact, FCRefStart`
- Definition of reference values (desired force, torque and/or search pattern). When activating a force/torque reference, the robot will move in order to achieve the specified reference level. When activating a search reference, the robot will attempt to move according to the specified pattern, like circular, spiral or linear movement. RAPID instructions: `FCRefForce, FCRefLine, FCRefCircle, FCRefSpiral, FCRefSpringForce` etc.
- End conditions (how long to apply the force, torque and search pattern). The reference force, torque and movement is used to search for a fit. The end conditions are used to determine when this search has been successful. All conditions have a time-out that allows execution to continue even if the end condition is not met. RAPID instructions: `FCCondPos, FCCondForce, FCCondTime, FCCondWaitWhile` etc.
- Supervision, that is Force Control data can be supervised and limits set that must be met. Any data outside supervision conditions will result in an emergency stop. This can be used as a safety measurement by limiting the work area, robot speed, etc. RAPID instructions: `FCSupvForce, FCSupvPos` etc.

Continues on next page

- Functions providing feedback from the process, which will give a snapshot of some process data like contact forces, actual load, speed, or if the condition was met or if timed out. RAPID functions: `FCGetForce, FCGetProcessData, FCIsForceMode`
- Data types supporting the instructions and functions
- Up to four robots can be controlled independently in a MultiMove cell.

Machining FC feature

- Instructions for programming FC Pressure start, movements and stop.
- Instructions for activating/deactivating FC Speed Change.
- Instruction to setup gravity compensation and sensor offset calibration.
- Instructions for defining reference values (desired force, speed change parameters or movement).
- Instructions for defining recover function for Speed Change
- Instructions for supervision.
- Functions returning data about load, detected forces or process status.
- Data types supporting the instructions and functions

Example of setting up an assembly application

The following steps will illustrate how to set up a new assembly task:

1. Calibrate force sensor
2. Calibrate work piece weight and center of gravity
3. Done once for one type of work piece by executing a RAPID program
4. Determine start point to activate force control, that is where contact is possible
5. Determine nominal contact for magnitude and direction period during which assembly take place
6. Driving force to make two parts fit together

   No motion command necessary during this period
7. Determine the search pattern magnitude, frequency and direction
8. Determine supervision criteria

   Optional, may be used to prevent robot to move into fragile areas
9. Determine end point to deactivate force control

   Typical point for the completion of assembly
10. Activate Force Control and references and wait until criteria are fulfilled

    Robot will move until assembly end condition or time out is reached

Product

The option *Force Control Base* will include following:

- Advanced software for the force control
- A specific set of RAPID instructions for assembly
- A specific set of RAPID instructions for machining
- Manual

Continues on next page

The option *Force Control Base* requires the option *738-1, Prep. for Force Control*, adapted for force control.

The following is not included but must be purchased and installed separately:

- Force sensor to be mounted on mounting plate
- Cabling between robot and controller
- Interface board for the sensor, to be mounted in the controller on the axis computer

Force sensor

Force Control Base requires either a 6 DOF (Degree Of Freedom) full force/torque sensor or a 1 DOF Force sensor. The 6 DOF sensor offers more flexibility in force control directions than the 1 DOF sensor, as 1 DOF sensor only controls the force in one direction. With 1 DOF sensor it is normally very important that the tool is correctly orientated.

The choice of sensor input depends on the required flexibility of the application.

The sensor may be mounted on the robot (between the toolflange and the tool) or stationary in a room fixed position. It is not necessary to mount sensor and tool on the same place.

ABB Force sensor

ABB provides a 6 DOF force sensor that is fully integrated into the robot hardware and software.

The force sensor is included in a *Force Control Package*, which contains all required hardware and software to run the robot in force control mode. The specification of the *Force Control Package* can be found in *Product specification - Controller IRC5*.

ATI Force sensor

The ABB alternative 6 DOF sensors are ATI Force/Torque sensors of model Delta, Theta and Omega. See http://ati-ia.com/ for more information.

The following items need to be ordered from ATI in order to complete the Force Control installation:

ATI provides the following Sensor product range adapted to ABB Force Control.

Items to complete the Force Control installation	Description
Sensor including adapter plate for ABB robot	See details of part number below
Connection cable	See details of part number below
ABB data disk	9030-05-1005

ATI Sensor Type, including adapter plate for ABB tool flange

Robot	Proposed sensor type
IRB 120	Mini45

Continues on next page

14.9 Force Control Base [661-2]
Continued

Robot	Proposed sensor type
IRB 140 IRB 2400L IRB 1600 IRB 2400-10/-16 IRB 2600	Delta
IRB 4400 IRB 4600	Theta Omega160
IRB 66XX	Omega160, Omega190, Omega250
IRB 7600	Omega250

These sensors and adapter plates are only recommendations. For correct selection depending on robot type, ATI should be consulted.

ATI IP-environment protection

Sensors are available in the following protection classes: IP60, Dust protection, IP65, Wet spray protection, IP65V, with Viton seals for applications with exposure to solvents and aggressive oils and IP68 for underwater protection (10m).

ATI Sensor Part number

The part number shall be:

Part number	Description
9105-TIF-Type-IPxx	Where: Type is equal to sensor type according to table in *ATI Sensor Type, including adapter plate for ABB tool flange on page 149*. IPxx is chosen according to *ATI IP-environment protection on page 150*.

ATI Cable Part number

The part number shall be:

Part number	Description
9105-C-Lx-AM-yy	Where: Lx has to be chosen L for IP60, IP65 and IP68, LV for IP65V yy will be chosen according cable length in meters that is required (from measuring board to sensor). Standard lengths available are 12, 17, 20, 25, 27, 32, 35 and 40 m (robot floor cable + 5 meter for small robots and 10 meters for large robots). Other lengths are available on request.

Limitations

- Force Control can not be activated when the robot is running in *MultiMove Coordinated* mode.
- *Force Control Base* is not available for IRB 260, IRB 360, IRB 460, IRB 660 and IRB 760.
- *Force Control Base* requires IRC5 controller (all variants).
- The total load, that is the sum of gravitational forces and external contact forces, must not exceed limits as specified in the load diagrams for a specific robot.

Continues on next page

- Force Control shall not and can not be activated simultaneously with the following functions:
- - *Sensor* or *Analog synchronization*
  - *Sensor interface*
  - *Conveyor tracking*
  - *Independent axes*
  - *SoftMove*
  - *Path Offset*
  - CAP with *Optical tracking*
  - *Arc*
  - *PickMaster*

14.10 Machining FC GUI [877-1]

Machining FC GUI

Machining FC GUI is a set of instructions, which will highly facilitate the use of robots for machining applications. *Machining FC GUI* includes solutions for using force control in different machining applications where the robot needs to be sensitive to process forces. *Machining FC* provides a Graphical User Interface (GUI) supporting a new solution for solving difficulties in programming complex 3D paths through lead-through programming.

Lead-through programming

Lead-through programming: allow easy, quick and accurate creation of complex robot paths. It is supported by an icon based Graphical User Interface where the user first leads the robot by hand and then the robot automatically generates the path.

The *Machining FC GUI* requires a sensor input that measures the process forces that will adjust the behavior of the robot.

- The GUI requires a 6DOF sensor mounted on the robot.

About the Lead-through programming

Main components in the lead-through programming tool are:

- Lead-through
- Automatic Path Learning
- Testing of the path running the actual process
- Export of the final result to include in the original robot program

These features use online active force control technique, which means that the programming becomes less time consuming and more effective. The programming technique offered by this option is by teaching certain geometric targets by lead-through dragging and collision free jogging of the robot. Then the robot goes into a self-learning mode where the robot automatically follows the previous taught geometric positions exerting a pressure perpendicular to the work object and at the same time recording the path.

Machining FC GUI features

- Graphical User Interface on the FlexPendant for easy, quick and accurate programming

Examples of setting up a machining function

The following steps will illustrate how to set up a new FC Pressure task:

1. Identify the load and calibrate the system
2. Move to a point close to contact
3. Set up desired force and start movement towards the surface
4. Move linear or circular performing the process with contact.
5. Leave surface and deactivate force control.

Continues on next page

All of the above steps are supported in the GUI for lead-through programming through icon based programming wizard.

The following steps will illustrate how to set up a new FC Speed Change task:

1 Configure the parameters for speed change controller
2 Identify the load and calibrate the sensor (if 6DOF sensor is used)
3 Define recover function
4 Activate FC Speed Change
5 Execute the machining task; move along the path performing the process.
6 Deactivate FC Speed Change.

All of the above steps are supported in the GUI for lead-through programming through icon based programming wizard.

Product

The *Machining FC GUI* software option includes following:

- Advanced software for the force control
- An icon based Graphical User Interface supporting lead-through programming
- Manual

The option *Machining FC GUI* requires the option *Force Control Base*.

Limitations

- Force Control can not be activated when the robot is running in *MultiMove Coordinated* mode.
- The total load, that is the sum of gravitational forces and external contact forces, must not exceed limits as specified in the load diagrams for a specific robot.
- Force Control shall not and can not be activated simultaneously with the following functions:
  - *Sensor* or *Analog synchronization*
  - *Sensor interface*
  - *Conveyor tracking*
  - *Independent axes*
  - *SoftMove*
  - *Path Offset*
  - CAP with *Optical tracking*
  - *Arc*
  - *PickMaster*

14.11 RobotWare Cutting [951-1]

General

RobotWare Cutting is a software product primarily developed for laser cutting and other similar cutting methods requiring advanced robot motion performance. The software provides an interactive user interface on the Flexpendant that helps program, configure and operate an ABB Robot in cutting applications.

The *RobotWare Cutting* software contains motion performance tools, such as automatic friction tuning, iterative learning control, and *WristMove*, which can substantially improve the path accuracy in laser cutting applications. The software also contains dedicated user interfaces that will help program and configure the motion performance tool set. To help the programmer the software incorporate simplified shape generation through a library of pre-defined shapes and cut instructions. These have automatic optimization and interaction to the other cutting components (friction tuning, iterative learning control, equipment classes, etc). To facilitate integration of external equipment such as cutting heads, laser sources, etc the software include pre-defined communication interfaces, equipment class templates, between the robot controller and the equipment. The pre-defined cut instructions and external equipment can be configured through user screens on the Flexpendant, which helps the programmer to program, install and configure a robotic cutting application. In addition the software supports standard ABB Robot motion features for example *TrueMove*, *QuickMove* and BullsEye.

ABB also offer a cutting add-in to the RobotStudio PC tool. The cutting add-in, called *RobotStudio Cutting PowerPac* currently includes feature based 2D cut instruction generation, wizard to generate customized 2D cut shape based on CAD models, free form path generation from the edges, simulation and pre-defined virtual signals for the cutting equipment.

Features

User interface

- Intuitive user interface for online programming
- Predefined 2D shape generation Instructions
- Auto shape tuning integrated with cut instructions
- Standard cutting and laser equipment interfaces
- HMI - Quick argument checking and editing

Customizable operator interface made by Screen maker

- General laser equipment operator interface
- Cell logic (support by Product Manger)

Shape generation

- Library of pre-defined shapes and cut instructions (Circle, Slot, Rectangle, Hexagon, CAD)
- Free-Form Cut Instructions (Linear, Circular)
- Automatic optimization and interaction to other cutting components

Continues on next page

Equipment interfaces

- Standard Laser Equipment functions controlled from dedicated equipment classes
- General cutting process interface for customized cutting equipment

Advanced shape tuning - automatic friction compensation

- Automatic tuning for optimum motion performance for a specific shape at a specific location

ILC - Iterative learning control

- Shape perimeter adjustment & compensation

Speed modulation

- Laser power can be modulated by TCP speed

Additional features (require additional robot options)

- WristMove - Movement method that only use two robot axes to move the tool
- BullsEye - Allows the operator to automatically define an accurate TCP

Application

The software is primarily targeting laser cutting applications. From a robotics perspective, laser cutting is typically characterized by high motion performance requirements, complex paths and shapes, and advanced equipment integration.

RobotWare options

RW Cutting includes the following RobotWare options:

- 602-1 Advanced Shape Tuning
- 617-1 FlexPendant Interface
- 611-1 Path recovery

Additional options that may be required for special use cases:

- 623-1 Multitasking
- 616-1 PC interface
- 812-1 Production manager

Limitations

- *RobotWare Cutting* is not available as MultiMove system.
- *RobotWare Cutting* is available for IRC5 and all 6-axes ABB robots.
- The *RobotWare Cutting* option will not improve the motion performance for large robots, IRB 66XX / IRB 76XX.
- As a general rule of thumb, typically no improvement of motion performance can be achieved for circle diameters smaller than 3 mm or cutting speed above 400 mm/s. The unique conditions (robot, tools, material, fixtures, etc) in the cutting application will determine the overall performance.
- The software is not designed for cutting applications with high contact forces.
- Available languages are English and Chinese. Operator interface can be customized to any language with RobotStudio ScreenMaker.

Continues on next page

- *RobotWare Cutting* requires an additional option license that is delivered with the controller.
- *RobotWare Cutting* does not work together with option 633-1 Arc.

14.12 RobotWare Machine Tending [1167-1]

General

RobotWare Machine Tending is a software product for programming, installation and operation of ABB robots in machine tending and material handling applications. The software can make it easier to manage the robot and the peripheral equipment in handling applications, both for the operator as well as for the programmer. It provides a RAPID framework and graphical user interface for programming and configuration.

The RAPID framework is modular and flexible as it is fully integrated with ABB's standard programming language. A set of RAPID data types, instructions and functions are provided in the *RobotWare Machine Tending* facilitating integration to application programs.

The software also provides an easy to understand and customizable graphical operator interface that facilitate trouble-free operation. The graphical user interface gives the users an overview of operating states of the stations and production processes. *RobotWare Machine Tending* has functions for production monitoring and control, hot edit, safe home run, production statistics, control over gripper actuation, selection of production cycles, user authorization, error handling, etc.

Features

Graphical user interface

- Visualization of the operating states and production processes
- Operation screen with information and status of stations and production cycles, and robot movements
- Customizable graphical user interface
- Gripper actuation and gripper monitoring
- Cycle handling and program control – Definition and selection of production cycles
- Execution of setup and service routines
- Advanced Hot Edit – modification of robot positions
- Automatic and safe HomeRun
- Error handling and messaging tasks
- Production statistics and other information available in the program
- Event log
- Signal status
- Status and error indication in colors

RAPID program framework

- Modular functions based on standard RAPID syntax - Data types, instructions and functions covering the following
  - Software engine that controls the program and necessary data to display graphic elements on the user interface
  - Instructions and functions signal handling, and parametrization and instructions to execute movement routines

Continues on next page

  - Data types, instructions and functions, for actuating and controlling the robot grippers
  - Association of events with routines
  - Declaration of signals and variables
  - Automatically return the robot to home from any position
- The RAPID framework provides several features such as
  - Different operating modes
  - Pre-condition check of part handling in production
  - Move robot to pre-defined service positions with help of digital input signals

Process configuration

- Configuration and parameterization of the machine tending process
  - Influence the appearance of the graphical user interface
  - Parameters with which the signals can be set for the remote operation of the RWMT-functions as well as other functional values
  - Parameters for signals and movements to positions, e.g. home position, safe position, and up to three service positions
  - Definition of the external interface, which is necessary to execute a pre-selected program or service request
  - Modify behavior of automatic and safe home run
- Configure the graphical user interface to start customized FlexPendant applications
- Pre-configured configuration files for Euromap and SPI

Performance

RobotWare Machine Tending has a modularized machine tending program structure, which essentially consists of three components:

- graphical user interface
- RAPID data types, instructions and functions
- process configuration

The user interface gives the operator and programmer an overview of the handling cell with all its stations such as the processing machines, conveyors, control stations, etc and it also contains control functions. A library of RAPID data types, instructions and functions supports creating the robot programs and designing the details of the user interface. *RobotWare Machine Tending* functionality can easily be modified through the process configuration, for example the graphical user interface appearance or the digital signal assignments that controls the most common functionality.

This gives a very powerful and flexible programming tool. Users can utilize advanced programming possibilities to create and modify stations, grippers and configuring program execution. The architecture of *RobotWare Machine Tending* makes it possible to integrate the software into existing robot automation cells, because the software only adds of additional data, commands and functions in the robot program.

Continues on next page

RobotWare Machine Tending works best together with the PC tool *RobotStudio Machine Tending PowerPac*. With the *RobotStudio Machine Tending PowerPac* users can program, simulate, and configure machine tending programs in an offline environment. The *Machine Tending PowerPac* is seamlessly integrated with the *RobotWare Machine Tending* software, which facilitates easy deployment, installation and operation.

Requirements

- RobotWare 5.15 and later
- Software option *617-1 FlexPendant interface* (included)
- Software option *608-1 World Zones* (included)

Optional:

- Software option *623-1 Multitasking*
  - If interface signals or RAPID variables needs to be set to a specific value when operation mode changes, e.g. when using Euromap interface for injection molding machines

Limitations

- *RobotWare Machine Tending* cannot be used with *MultiMove Coordinated*
- *RobotWare Machine Tending* is available on all six-axis robots, that is, not on ABB 4-axis robots
- *RobotWare Machine Tending* requires an additional option license that is delivered with the controller
- Application manual available in English. Additional languages will follow.
- Operating manual available in multiple languages.
- Default operator interfaces on the FlexPendant is available in English, but it is possible to customize to any language by the user.

This page is intentionally left blank

Index

M
Motion Process Mode, 40

Contact us

ABB AB
Discrete Automation and Motion
Robotics
S-721 68 VÄSTERÅS, Sweden
Telephone +46 (0) 21 344 400

ABB AS, Robotics
Discrete Automation and Motion
Nordlysvegen 7, N-4340 BRYNE, Norway
Box 265, N-4349 BRYNE, Norway
Telephone: +47 51489000

ABB Engineering (Shanghai) Ltd.
5 Lane 369, ChuangYe Road
KangQiao Town, PuDong District
SHANGHAI 201319, China
Telephone: +86 21 6105 6666

www.abb.com/robotics

3HAC050945-001, Rev A, en