



## Product specification

### Controller Software IRC5

RobotWare 5.06





# Product Specification

Controller Software IRC5

RobotWare 5.06

3HAC 022349-001

Revision 2

The information in this manual is subject to change without notice and should not be construed as a commitment by ABB. ABB assumes no responsibility for any errors that may appear in this manual.

Except as may be expressly stated anywhere in this manual, nothing herein shall be construed as any kind of guarantee or warranty by ABB for losses, damages to persons or property, fitness for a specific purpose or the like.

In no event shall ABB be liable for incidental or consequential damages arising from use of this manual and products described herein.

This manual and parts thereof must not be reproduced or copied without ABB's written permission, and contents thereof must not be imparted to a third party nor be used for any unauthorized purpose. Contravention will be prosecuted.

Additional copies of this manual may be obtained from ABB at its then current charge.

©Copyright 2004 ABB All right reserved.

ABB Automation Technologies AB  
Robotics  
SE-721 68 Västerås  
Sweden

<b>1 Introduction</b>	<b>7</b>
<b>2 IRC5 vs. S4Cplus</b>	<b>9</b>
<b>3 Installation Options</b>	<b>13</b>
3.1 RW 5 not installed [600-1] . . . . .	13
3.2 RW 5 Boot package [601-1] . . . . .	14
<b>4 Motion Performance</b>	<b>15</b>
4.1 Advanced Shape Tuning [602-1] . . . . .	15
4.2 Absolute Accuracy [603-1] . . . . .	16
<b>5 Motion Coordination</b>	<b>19</b>
5.1 MultiMove Coordinated [604-1] . . . . .	19
5.2 MultiMove Independent [604-2] . . . . .	21
5.3 Multiple Axis Positioner [605-1] . . . . .	22
5.4 Conveyor Tracking [606-1] . . . . .	23
5.5 Sensor Synchronization [607-1] . . . . .	24
<b>6 Motion Events</b>	<b>25</b>
6.1 World Zones [608-1] . . . . .	25
6.2 Fixed Position Events [609-1] . . . . .	27
<b>7 Motion Functions</b>	<b>29</b>
7.1 Independent Axis [610-1] . . . . .	29
7.2 Path Recovery [611-1] . . . . .	31
7.3 Path Offset [612-1] . . . . .	32
<b>8 Motion Supervision</b>	<b>33</b>
8.1 Collision Detection [613-1] . . . . .	33
<b>9 Communication</b>	<b>35</b>
9.1 FTP Client [614-1] . . . . .	35
9.2 NFS client [615-1] . . . . .	36
9.3 PC Interface [616-1] . . . . .	37
9.4 FlexPendant Interface [617-1] . . . . .	39
9.5 Fieldbus Command Interface [618-1] . . . . .	40
9.6 Socket Messaging [619-1] . . . . .	41
9.7 File and Serial Channel Handling [620-1] . . . . .	42
<b>10 Engineering Tools</b>	<b>45</b>
10.1 Multitasking [623-1] . . . . .	45
10.2 Continuous Application Platform [624-1] . . . . .	47
10.3 Discrete Application Platform [625-1] . . . . .	48
10.4 Advanced RAPID [626-1] . . . . .	49
10.5 Sensor Interface [628-1] . . . . .	52

# Table of Contents

---

<b>11 I/O Control</b>	<b>53</b>
11.1 Logical Cross Connections [621-1] .....	53
11.2 Analog Signal Interrupt [622-1] .....	54
<b>12 Servo Motor Control</b>	<b>55</b>
12.1 Servo Tool Control [629-1] .....	55
12.2 Servo Tool Change [630-1] .....	57
12.3 Electronically Linked Motors [631-1] .....	59
<b>13 Diagnostic Tools</b>	<b>61</b>
13.1 Service Information System [632-1] .....	61
<b>14 Application Options</b>	<b>63</b>
14.1 Arc [633-1] .....	63
14.2 Software Arc Options .....	66
14.2.1 Introduction .....	66
14.3 Power Sources .....	67
14.3.1 Introduction .....	67
14.3.2 Arcitec [650-1] .....	67
14.3.3 MigRob [650-2] .....	68
14.3.4 Fronius TPS [650-3] .....	70
14.3.5 Miller AutoAxxess [650-4] .....	70
14.3.6 Miller DeltaWeld [650-5] .....	71
14.3.7 RPB [650-6] .....	71
14.3.8 Standard I/O Welder [650-7] .....	71
14.3.9 Simulated Welder [650-8] .....	71
14.4 Additional Arc Systems .....	72
14.4.1 Introduction .....	72
14.4.2 One additional [651-1] .....	72
14.4.3 Two additional [651-2] .....	72
14.5 BullsEye .....	73
14.5.1 Introduction .....	73
14.5.2 BullsEye [652-1] .....	73
14.6 Torch Cleaner .....	75
14.6.1 Introduction .....	75
14.6.2 ABB - TC96 [653-1] .....	75
14.6.3 Binzel - TC97 [653-2] .....	76
14.6.4 Tweco - TC2000 [653-3] .....	76
14.6.5 Torch Clean [654-1] .....	77
14.6.6 Anitspatter [655-1] .....	77
14.6.7 Wire Cutter [656-1] .....	77
14.7 Torch Service Center .....	78
14.7.1 Introduction .....	78
14.7.2 Torch Service Center [674-1] .....	78
14.8 SmarTac .....	79
14.8.1 Introduction .....	79
14.8.2 SmarTac - North American [657-1] .....	80

14.8.3 SmarTac - European [657-2] ..... 80

**14.9 Seam displacement ..... 81**

14.9.1 Introduction ..... 81

**14.10 Production Monitoring ..... 83**

14.10.1 Introduction ..... 83

**14.11 Optical Tracking. .... 88**

14.11.1 Introduction ..... 88

**14.12 MultiProcess [634-1] ..... 89**

**14.13 Spot [635-1] ..... 90**

**14.14 Spot Multiple Guns [635-2] ..... 92**

**14.15 Spot Servo [635-3] ..... 94**

**14.16 Spot Servo Multiple Guns [635-4]. .... 97**

**14.17 Spot Servo Equalizing [635-5] ..... 99**

14.17.1 Introduction ..... 99

**14.18 Dispense [641-1] ..... 100**

**14.19 Prepared for PickMaster [642-1]. .... 102**

**14.20 RobotWare Plastics Mould [643-1]. .... 104**

**Table of Contents**

---



# 1 Introduction

## SoftWare Products

RobotWare is a family of software products from ABB Automation Technologies AB, Robotics. The products are designed to make you more productive and lower your cost of owning and operating a robot.

ABB Automation Technologies AB, 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 two classes of products:

Product classes	Description
RobotWare – OS	This is the operating system of the robot and constitutes the foundation of the robot family. 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 the Product Specification for IRC5 with Flex-Pendant.
RobotWare options	These products are options that run on top of RobotWare – OS. They are intended for robot users that need additional functionality for example run multitasking for surveillance of equipment or transfer information from file to robot or communicate with a PC or perform advanced motion tasks, etc.

## 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.



## 2 IRC5 vs. S4Cplus

### General

In RobotWare 5.0 (IRC5), the RobotWare options have been restructured, as compared to S4Cplus.

The previous option packages have been split up in individual functions so that it is now possible to choose exactly the functionality required.

Furthermore, a number of functions have been moved to RobotWare - OS.

Below you find a conversion table between IRC5 and S4Cplus.

S4Cplus	IRC5 RobotWare	No.	Group
-	Fieldbus Command Interface	618-1	Communication
Absolute Accuracy	Absolute Accuracy	603-1	Motion Performance
Advanced Functions	File and Serial Channel Handling	620-1	Communication
Advanced Functions	Logical Cross Connections	621-1	I/O Control
Advanced Functions	Analog Signal Interrupt	622-1	I/O Control
Advanced Functions	World Zones	608-1	Motion Events
Advanced Functions	Fixed position Events	609-1	Motion Events
Advanced Functions	Path Recovery	611-1	Motion Functions
Advanced Functions	Advanced RAPID	626-1	Engineering Tools
Advanced Motion	Multiple Axis Positioner	605-1	Motion Coordination
Advanced Motion	Independent Axis	610-1	Motion Functions
Advanced Motion	Path Offset	612-1	Motion Functions
Advanced Motion	Advanced Shape Tuning	602-1	Motion Performance
ArcWare	Arc	633-1	Application options
ArcWare Plus	Arc Advanced Tracking (Not in RW 5.06)	633-2	Application options
Collision Detection	Collision Detection	613-1	Motion Supervision
Continuous Application Platform	Continuous Application Platform	624-1	Engineering Tools
Conveyor Tracking	Conveyor Tracking	606-1	Motion Coordination
Developer's Functions	Advanced RAPID	626-1	Engineering Tools
Discrete Application Platform	Discrete Application Platform	625-1	Engineering Tools
DispenseWare	Dispense	641-1	Application options
Electronically Linked Motors	Electronically Linked Motors	631-1	Servo Motor Control
Ethernet Services/FTP	FTP client	614-1	Communication
Ethernet Services/NFS	NFS client	615-1	Communication
FactoryWare Interface	PC Interface	616-1	Communication

S4Cplus	IRC5 RobotWare	No.	Group
Multitasking	Multitasking	623-1	Engineering Tools
PalletWare	Pallet	640-1	Application options
PickWare	Prepared for PickMaster	642-1	Application options
ScreenViewer	FlexPendant Interface	617-1	Communication
Sensor Interface	Sensor Interface	628-1	Engineering Tools
Sensor Synchronization	Sensor Synchronization	607-1	Motion Coordination
Service Information System	Service Information System	632-1	Diagnostic Tools
Servo Tool Change	Servo Tool Change	630-1	Servo Motor Control
Servo Tool Control	Servo Tool Control	629-1	Servo Motor Control
SpotWare	Spot	635-1	Application options
SpotWare Plus	Spot Multiple Guns	635-2	Application options
SpotWare Servo	Spot Servo	635-3	Application options
SpotWare Servo Plus	Spot Servo Multiple Guns	635-4	Application options
Load ID	Standard	202-1	RobotWare - OS
I/O Plus	Standard	193-1	RobotWare - OS

### Future options

The following options are planned for later releases than RW 5.06.

Option	Description
[633-3]	Arc Advanced Tracking
[640-1]	Pallet

### 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.
Motion Coordination	Options that make your robot coordinated with external equipment or other robots.
Motion Events	Options that issues an event, depending on the motion of the robot.
Motion Functions	Options that change the functionality of your robot. (E.g. makes a robot axis move independently of the others.)
Communication	Options that make the robot communicate with other equipment. (External PCs etc.)
I/O Control	Options that make the robot communicate with I/O.

Option Groups	Description
Servo Motor Control	Options that make the robot controller operate external motors, independent of the robot.
Diagnostic Tools	Options to supervise your robot system.
Engineering Tools	Options for the advanced robot integrator.
Application options	Option packages dedicated to certain applications, such as arc welding. (Former "ProcessWare" options.)



## 3 Installation Options

### 3.1 RW 5 not installed [600-1]

---

**General**

This option is to be used when RobotWare – OS is not to be installed on the controller at delivery.

---

**RobotWare-OS**

In cases when this option is not chosen, the latest RobotWare – OS version will be installed in the controller in the production.

---

**Features**

- The customer installs RobotWare – OS from CD

### 3.2 RW 5 Boot package [601-1]

---

**General**

This option is to be used when the customer wants a separate CD with RobotWare – OS. (Always the latest version.)

---

**Features**

- Separate CD with RobotWare – OS
- RobotStudio Online
- Boot cable - for connection between laptop and Service port on the controller
- Getting started manual



## 4 Motion Performance

### 4.1 Advanced Shape Tuning [602-1]

---

<b>General</b>	<p>Advanced Shape Tuning offers the possibility to compensate for frictional effects that might appear at low speed cutting (10-100 mm/s).</p> <p>The option gives the user access to tuning parameters and the possibility to change the tuning parameters during program execution with RAPID commands in the robot program. The tuning is performed by the user and for each specific shape.</p>
<b>Features</b>	<ul style="list-style-type: none"><li>• Very accurate path performance for advanced shape cutting</li><li>• Access to tuning parameters</li><li>• Tuning axis by axis</li><li>• Change tuning from RAPID program</li></ul>
<b>Application</b>	<p>Friction effects typically arise when cutting advanced shapes, e.g. circles, in small format. The effects appear in the form of approximately 0.5 mm path deviations.</p>
<b>Performance</b>	<p>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 the FlexPendant Operating Manual for tuning procedure and instruction TuneServo in RAPID Reference manual). Note that even with careful tuning, there is no guarantee that "perfect" paths will always be generated.</p> <p>For the IRB 66X0 and 7600 families of robots, no significant effects can be expected by applying Advanced Shape Tuning.</p>
<b>Requirements</b>	<p>There are no hardware or software requirements for this option.</p>
<b>RAPID instructions</b>	<p>There are no RAPID instructions included in this option.</p> <p>Change of tuning from RAPID is done with standard parameters.</p>

---

### 4.2 Absolute Accuracy [603-1]

---

#### General

Absolute Accuracy (AbsAcc) is a calibration concept, which ensures a TCP absolute accuracy of better than  $\pm 1$  mm in the entire working range (AbsAcc is limited to only forward positions for bending backward robots, eg IRB 6600.). The user is supplied with robot calibration data (compensation parameter file) and a certificate that shows the performance (Birth Certificate). The difference between an ideal robot and a real robot can typically be up to 8 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 need external equipment or calculation.

---

#### 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
- Exchangeability of robots

---

#### Performance

Once the Absolute Accuracy parameter file is 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	0.35	0.75	100
IRB 1400	0.35	0.75	100
IRB 2400 - L	0.45	0.80	100
IRB 2400 - 10/16	0.30	0.70	100
IRB 4400	0.30	0.75	100
IRB 6600	0.50	1.20	95
IRB 7600	0.56	1.20	91

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.

**Requirements**

Each Absolute Accuracy robot is shipped with a compensation parameter file that is unique to that robot. This file must be loaded into the controller and subsequently activated in order to use Absolute Accuracy. Absolute Accuracy functionality may also be deactivated.

**Supported robot types**

All M2004 robot types, except IRB 340 and IRB 940 are supported. However, only floor mounted robot variants are supported.

**RAPID instructions**

There are no RAPID instructions included in this option.



## 5 Motion Coordination

### 5.1 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 five simultaneous motion tasks, handling e.g. four robots and one positioner.
- 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 FlexPendant interface.
- RobotWare - Multitasking.
- RobotWare - Multiple Axis Positioner.

#### Application

Application	Description
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. (Flexpositioner)
Moving	Of heavy or flimsy objects by multiple robots

#### 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.

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.

---

### 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
MoveExtJ	Move external axes in joint coordinates

## 5.2 MultiMove Independent [604-2]

<b>General</b>	<p>The option MultiMove Independent makes a robot system a MultiMove system with independent robots functionality.</p> <p>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.</p> <p>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.)</p>
<b>Features</b>	<ul style="list-style-type: none"> <li>• Up to four robots in a MultiMove System</li> <li>• Up to five simultaneous motion tasks, handling e.g. four robots and one positioner</li> <li>• The robots in the MultiMove System work independently of each other</li> <li>• MultiMove FlexPendant interface</li> <li>• RobotWare - Multitasking</li> <li>• RobotWare - Multiple Axis Positioner</li> </ul>
<b>Application</b>	<p>Multi robot processing where each robot is working independently, controlled by separate RAPID tasks.</p>
<b>Performance</b>	<p>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.</p>
<b>Requirements</b>	<ul style="list-style-type: none"> <li>• 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.</li> </ul>
<b>Limitations</b>	<p>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.</p>
<b>RAPID instructions</b>	<p>RAPID instructions included in this option:</p>

Instruction	Description
MoveExtJ	Move external axes in joint coordinates

### 5.3 Multiple Axis Positioner [605-1]

---

#### General

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



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

---

#### Features

- Coordinated movement of robot and multiple axis manipulator
- Calibration possibilities for multiple axis manipulators

---

#### Application

This option shall be used for all types of multiple axis manipulators e.g. 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!** There is a built-in general method for defining the geometry for a manipulator comprising two rotating axes (see User's Guide Calibration). For other types of manipulators/robot carriers, comprising up to six linear and/or rotating axes, a special configuration file is needed. Please contact you nearest local ABB office.

---

#### RAPID instructions

There are no RAPID instructions included in this option.



## 5.4 Conveyor Tracking [606-1]

### General

Conveyor Tracking (also called line tracking) is the function 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", i.e. it is not necessary to stop in a fine point.

### Application

Any application where conveyors are used e.g. Painting, ArcWelding, 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.



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

### Requirements

Hardware components for measuring the conveyor position: DeviceNet option and encoder card QSQC 377Q. Please refer to the Product Specification for your robot.

### 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

### 5.5 Sensor Synchronization [607-1]

#### General

Sensor Synchronization adjusts the robot speed to an external moving device (e.g. 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

- Sync. robot with sensor: Up to 4 sensors/robot
- Sync. 2 or more robots
- “On-the-fly” activation
- Valid for any type of movement
- RAPID access to sensor and queue data
- Object queue (See option Conveyor Tracking)

#### 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:

- Sync. robot to server: DeviceNet option encoder card QSQC 377A
- Sync. robot to robot: DeviceNet option

#### RAPID instructions

RAPID instructions included in this option:

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

## 6 Motion Events

### 6.1 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

## 6 Motion Events

---

<b>Instruction</b>	<b>Description</b>
WZSpDef	Define a spherical world zone
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

## 6.2 Fixed Position Events [609-1]

### General

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, see 'Application' below.

### 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 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	E.g. 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 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	Description
TriggIO	Definition of trigg conditions for an output
TriggEquip	Definition of trigg conditions for process equipment with compensation for equipment delay
TriggCheckIO	Definition of trigg conditions for check of signal value
TriggInt	Definition of trigg conditions for an interrupt

## 6 Motion Events

---

<b>Instruction</b>	<b>Description</b>
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

## 7 Motion Functions

### 7.1 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.

## 7 Motion Functions

---

---

**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.

Internal and customer cabling and equipment may limit the ability to use independent axis functionality on axis 4 and 6.

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

---

**RAPID instructions**

RAPID instructions included in this option:

Instruction	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



## 7.2 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 continuous 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/error

### 7.3 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 Motion Supervision

### 8.1 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 the following robot families: IRB 140, IRB 1400, IRB 2400, IRB 4400, IRB 6400 (not 640) IRB 7600 and for positioners IRBP-L, IRBP-K, IRBP-R and 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.



## 9 Communication

### 9.1 FTP Client [614-1]

---

#### General

The option FTP client (File Transfer Protocol) makes it possible to read information on a remote hard disk, e.g. 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

The following servers are supported:

- Serv-U, version 4.1
- HP-UX, version 11.11
- Linux Suse version 7.0
- MS IIS, Windows 2000 Server 5.0 SP 1
- Distinct, version 3.0



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.

### 9.2 NFS client [615-1]

---

<b>General</b>	The option NFS client (Network File System) makes it possible to read information on a remote hard disk, e.g. a PC, directly from the controller.
<b>Features</b>	<ul style="list-style-type: none"><li>• Read information on remote hard disk</li><li>• Backup/restore from Flex Pendant</li><li>• Load and save RAPID programs</li><li>• Issue saving/reading of data from RAPID program</li></ul>
<b>Application</b>	<ul style="list-style-type: none"><li>• 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.</li><li>• 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.</li></ul>
<b>Performance</b>	There is no specific performance data available for this option.
<b>Requirements</b>	The following servers are supported: <ul style="list-style-type: none"><li>• Distinct, version 3.0</li><li>• Omni, version 4.12</li><li>• HP-UX, version 11.11</li></ul>
<b>RAPID instructions</b>	There are no RAPID instructions included in this option.

---

## 9.3 PC Interface [616-1]

**General** PC Interface provides the communication interface for the robot to a network PC (Ethernet).

- Features**
- Manual robot backup to a networked PC (included in RobotStudio Online)
  - OPC Server interface for SCADA integration (delivered with the RobotWare CD)
  - Communication interface for use with ABB Industrial Robot Software

### Application

Application	Description
RobotStudio Online	<ul style="list-style-type: none"> <li>-Remote update of programs, configurations or systems are possible over the LAN network.</li> <li>-WebWare Server</li> <li>-Backup of current robot programs and configurations are automatically made each night to a network server.</li> <li>-Production metrics are tracked for the purpose of reporting on output, cycle time variations, utilization rates, quality data, etc.</li> <li>-Status information about the robot and peripheral equipment is displayed for the operator on the PC screen.</li> </ul>
PC SDK	<ul style="list-style-type: none"> <li>-Production is controlled by an operator from a remote superior computer.</li> <li>-Status information about the robot and peripheral equipment is displayed for the operator on the PC screen.</li> <li>-Program execution and fixture clamps can be controlled from the PC.</li> </ul>

### ABB Industrial Robot Software

ABB offers value-added PC software products that are network enabled through this option:

- RobotStudio Online - 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. Note that RobotStudio Online can connect to a controller even if PC Interface is missing, in that case, however, only through the local service port.
- WebWare™ 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 SDK - is a state-of-the-art Software Developers Kit for designing graphical PC operator stations that communicate with ABB robot controllers and other production equipment. PC SDK includes a collection of ActiveX controls that are standard to Microsoft visual programming environments. The custom PC-based control Application built with PC SDK can be connected with this option. PC SDK is a subset of the product Robot SDK.

### RAPID instructions

---

RAPID instruction included in this option:

Instruction	Description
SCWrite	Sends a message to the network PC



## 9.4 FlexPendant Interface [617-1]

### General

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

### Features

- Download and execute application interfaces on the FlexPendant
- Application visible as new entries in the ABB-menu of the FlexPendant

### Application

Application	Description
FlexPendant Interface	An option which makes it possible to download and run Application developed with FlexPendant SDK (Software Development Kit) on the FlexPendant.
FlexPendant SDK	Contains all the needed software necessary to develop an application that will be contained in an external option to the IRC5 controller and hosted by the FlexPendant standard software.
Microsoft Visual Studio	The NET environment is used for all FlexPendant application development and the SDK works within this environment.
SDK	Templates (Solution and projects for VS.NET) Assemblies (UI controls, components and Controller API controls) Add-in for VS.NET Online Help Compliance tool (verifies that application follows FlexPendant limitations and deploys during development.)

The final results from developing an application with VS.NET and the FlexPendant SDK is a set of files that can be included in an External Option for the controller. External Options are added to systems created by RobotStudio-Online and downloaded to the controller or run as a Virtual Controller. During development of an application, the FlexPendant SDK's compliance tool is used for deployment.

### Performance

There is no specific performance data available for this option.

### Requirements

Application need to be developed with the software tool: FlexPendant SDK.

### RAPID instructions

There are no RAPID instructions included in this option.

### 9.5 Fieldbus Command Interface [618-1]

---

**General** Fieldbus Command Interface is an option, which is used when transferring commands or messages other than IO-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 e.g. 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 e.g. using the RAPID commands Open\Bin, Close, ReadRawBytes, WriteRawBytes.

---

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

---

**Requirements** This option requires the DeviceNet option and the option RobotWare – File and Serial Channel Handling.

---

**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

## 9.6 Socket Messaging [619-1]

### General

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, e.g. communication with RobotStudio-Online or WebWare Server).

Socket Messaging is a standard supported by e.g. 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/controller. 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

### Performance

There is no specific performance data available for this option.

### Requirements

There are no hardware or software requirements for this option.

### RAPID instructions

The following instructions are included in Socket Messaging:

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
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

### 9.7 File and Serial Channel Handling [620-1]

#### General

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, e.g. in packing and palletizing 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.

**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/ReadStrBin/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 rawbyte
WriteRawBytes	Write data rawbyte to a device
ClearRawBytes	Clear all contents of a rawbytes variable
CopyRawBytes	Copy rawbyte 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 rawbyte (bytes)

**Datatypes**

Datatype	Description
Rawbytes	A general data 'container', for communication with I/O devices



# 10 Engineering Tools

## 10.1 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

### 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, e.g. 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.

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.

---

**RAPID  
instructions**

RAPID instructions included in this option:

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



---

## 10.2 Continuous Application Platform [624-1]

**General**

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

Target users are advanced application software engineers and system integrators, e.g. for arc welding, laser welding, 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

**Application**

Creation of advanced application software with a continuous process to be synchronized with robot movement, e.g. arc welding, laser cutting, 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**

See RAPID Continuous Application Platform User's Guide.

### 10.3 Discrete Application Platform [625-1]

---

<b>General</b>	<p>Discrete Application Platform (DAP) is a software platform for time critical Application, where certain actions shall be performed at specific robot positions.</p> <p>Target users are advanced application software engineers and system integrators, e.g. for spot welding, drilling, measuring, quality control. The main advantages are achieved in the following areas:</p> <ul style="list-style-type: none"><li>• Development time</li><li>• Program execution time</li><li>• RAPID- program memory needed</li><li>• Similar 'look and feel' between Application</li><li>• Tested kernel software</li></ul>
<b>Features</b>	<ul style="list-style-type: none"><li>• Specialized RAPID instructions and datatypes</li><li>• A single instruction for motion and process execution</li><li>• Combination of finepoint positioning with execution of up to 4 parallel processes</li><li>• Specialized process for monitoring of external process devices, like spot welding controllers</li><li>• Supports encapsulation of the process and motion, in shell-routines provided to the end-user</li><li>• Includes RobotWare options: World zones, Fixed position Events, Path Recovery, File and Serial Channel Handling, Logical Cross Connections, Analog Signal Interrupt</li></ul>
<b>Application</b>	<p>Creation of software for advanced Application with a discrete behavior, such as spot welding, drilling, measuring, quality control.</p>
<b>Performance</b>	<p>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.</p>
<b>Requirements</b>	<p>In order to use DAP, the option MultiTasking needs to be installed.</p>
<b>RAPID instructions</b>	<p>See RAPID Discrete Application Platform manual.</p>

---

## 10.4 Advanced RAPID [626-1]

### General

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

### RAPID Kernel Reference Manual

The manual describes the RAPID language syntax and semantics in detail concerning the kernel, i.e. 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, i.e. 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
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, e.g. 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, e.g. 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.
SetSysData	nstruction which will activates the specified system data (tool or workobject). With this instruction it is possible to change the current active tool or workobject.

Instruction	Description
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.

## 10.5 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. The option includes a driver for serial communication with the sensor system using a specific link protocol (RTP1) and a specific application protocol (LTAPP).

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.

**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

**Application**

Sensor Synchronization 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 links.

**Limitations**

Sensor Interface can only be applied to one robot in a MultiMove configuration.

**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
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 I/O Control

## 11.1 Logical Cross Connections [621-1]

**General** 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 200 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.

### 11.2 Analog Signal Interrupt [622-1]

---

**General**

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 e.g. '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



# 12 Servo Motor Control

## 12.1 Servo Tool Control [629-1]

### General

The Servo Tool Control is a general and flexible software platform for controlling an integrated servo tool from IRC5. For additional features, like control of external processes, or control of several ServoGuns in parallel, please refer to the option RobotWare SpotServo.

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

As a “quick-start”, the option includes an example code package. This package can be used as a base for application development.

### 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, please refer to the option RobotWare - Servo.

---

**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.

In order to use Servo Tool Control, the options Advanced RAPID and Path Recovery both need to be installed.

For information on Drive Module & Measurement board see User's Guide External Axes.

---

**Limitations**

- Servo Tool Control can only be applied to one robot in a MultiMove configuration.
- Servo Tool Control cannot be combined with any of the spotwelding options [635-1, 635-2, 635-3, 635-4].

---

**RAPID instructions** RAPID instructions included in this option:

Instruction	Description
STClose	Close a Servo Tool with a predefined force and 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 or Servo Tool Control 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.



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.

## 12 Servo Motor Control

---

---

<b>Requirements</b>	Servo Tool Change requires a mechanical wrist interface, a Tool Changer. A MOC service parameter, Disconnect deactivate = YES (Types: Measurement channel), must be set for each tool (external axis) used with this function.
<b>Limitations</b>	Sensor Interface can only be applied to one robot in a MultiMove configuration.
<b>RAPID instructions</b>	There are no RAPID instructions included in this option.

## 12.3 Electronically Linked Motors [631-1]

### General

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.

### 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.



---

# 13 Diagnostic Tools

## 13.1 Service Information System [632-1]

---

### General

Service Information System is a service routine, which gives an alarm on the FlexPendant when the robot needs service. This function is available for IRB 6600/6650 and IRB 7600.

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.

Remark: 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).

## 13 Diagnostic Tools

---

---

**Requirements**

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

---

**RAPID  
instructions**

There are no RAPID instructions included in this option.



---

# 14 Application Options

## 14.1 Arc [633-1]

---

### 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.

---

### 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<sup>1</sup>**  
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 burnback 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).**

---

1. Only available for IRB 140, IRB 1400 and IRB 2400.

## 14 Application Options

---

### 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
Wirestick error	Wirestick 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  
(cont.)**

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 Application Options

### 14.2.1 Introduction

## 14.2 Software Arc Options

### 14.2.1 Introduction

#### 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 system builder. All arc options are sub-components listed under RobotWare Arc as seen in the picture below.

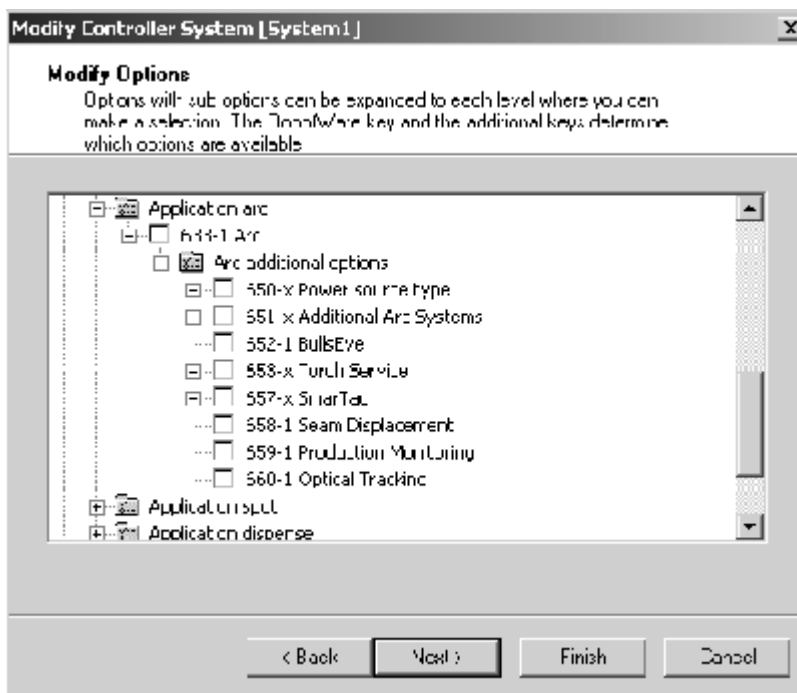


Figure 1

## 14.3 Power Sources

### 14.3.1 Introduction

#### General

In order to control and communicate with a certain type of power source supply, a software interface is needed. ABB have developed a number of specific software interfaces for different power source supplies, providing the user an easy to use and workable Arc Welding Robot process interface. The available software interfaces are designed to manage different levels of integration depending on selected power source supply.

The integration level on the power source options, described below, covers between a standard I/O based user interface and a High-Level user interface based on Device Net communication, where all process parameters are controlled from the FlexPendant.

### 14.3.2 Arcitec [650-1]

#### General

The Arcitec IRC5 user interface is a high-level programming tool used for operating and managing one or more LRC-type ABB power sources.

The communication is performed through a robot Device Net interface, which allows the operator to program all process parameter from the Robot FlexPendant.

The operator can through this software access and control all process parameters on an easy to use and intuitive graphic touch screen, see below example of screen view.

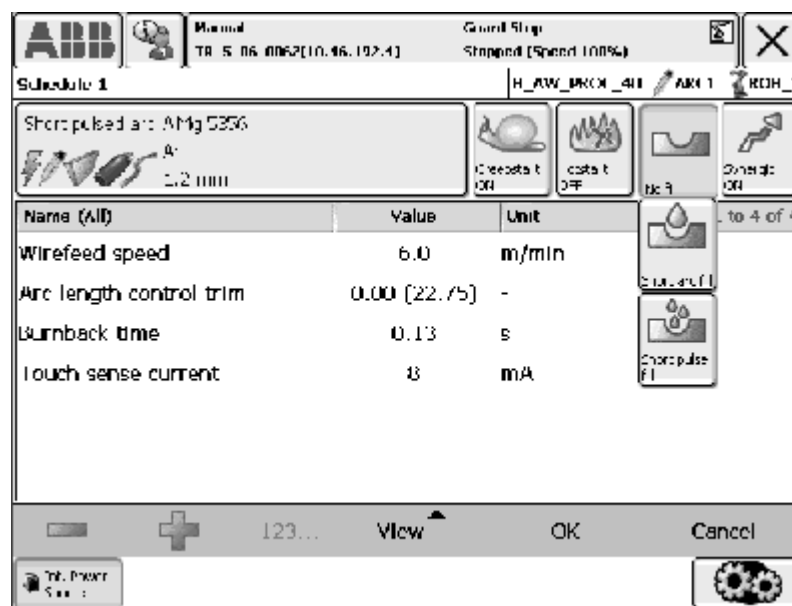


Figure 2

## 14 Application Options

### 14.3.3 MigRob [650-2]

Following main tasks can be handled from the Arcitec IRC5 process user interface:

- Changing data in the schedule memory.
- Creating and removing individual synergic lines.
- Programming of advanced functions, including service functions in the power source and welding setting for the robot program.
- Backing up and restoring of all process data in the schedule memory.

### 14.3.3 MigRob [650-2]

#### General

The MigRob 500 user interface is a high-level programming tool used for operating and managing one or more MigRob 500 ABB power sources.

The communication is performed through a robot Device Net interface, which allows the operator to program all process parameters from the Robot FlexPendant.

The operator can through this software access and control all process parameters on an easy to use and intuitive graphic touch screen, see below example of screen view.

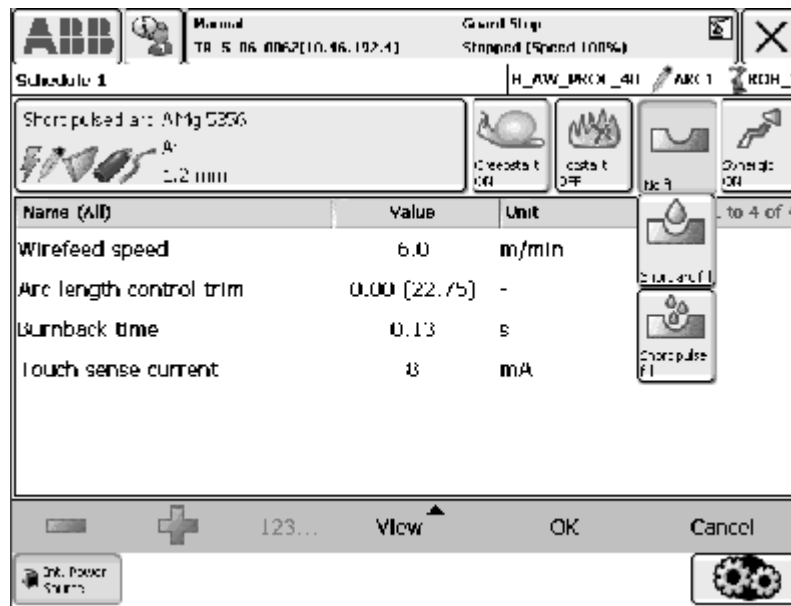


Figure 3

Following main tasks can be handled from the MigRob 500 process user interface:

- Changing data in the schedule memory.
- Creating and removing individual synergic lines.
- Programming of advanced functions, including service functions in the power source and welding settings including Super Pulse parameters for the robot program, see below example of screen view.
- Backing up and restoring of all process data in the schedule memory.

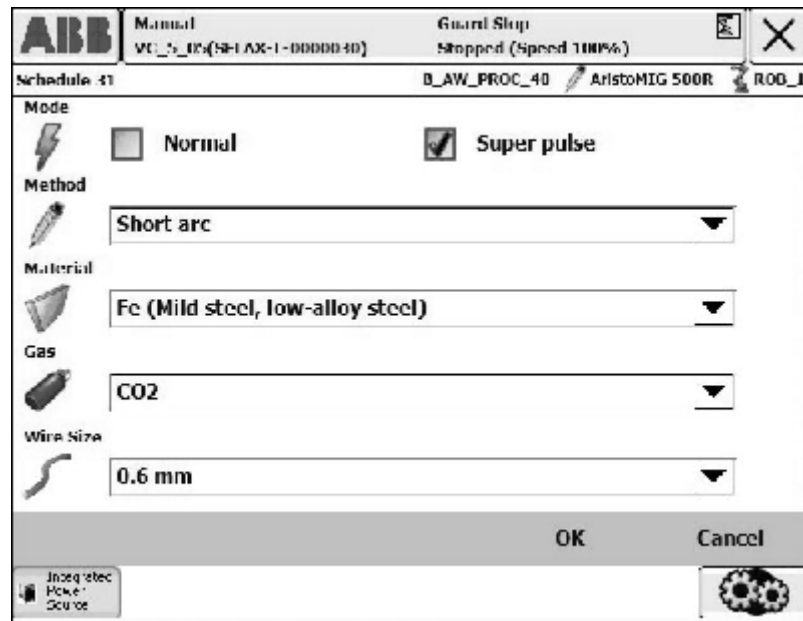


Figure 4

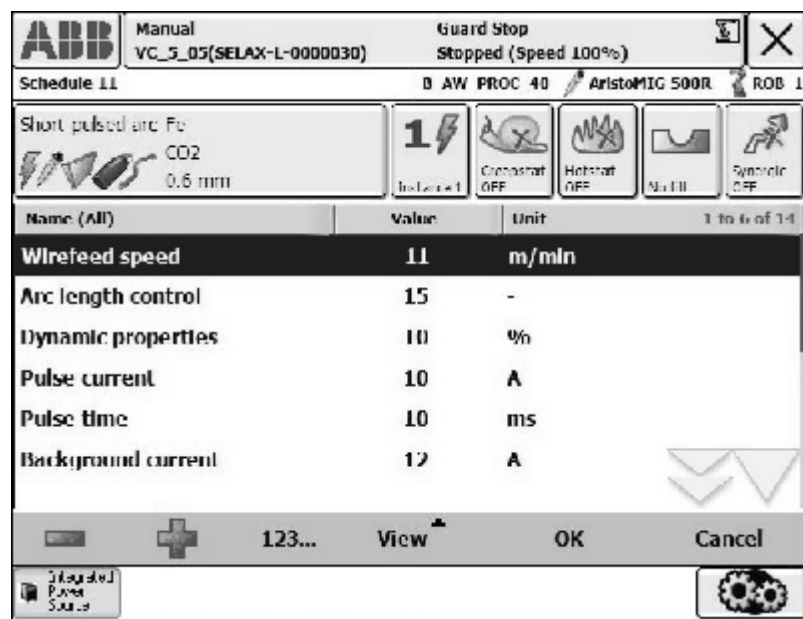


Figure 5

## 14 Application Options

---

### 14.3.4 Fronius TPS [650-3]

#### 14.3.4 Fronius TPS [650-3]

---

##### General

The Fronius TPS user interface is a standard Arc programming tool used for operating and managing one or more TPS-type of power source.

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

The operator can through this software call for pre-programmed jobs (schedules), which have been made in the separate Fronius power source operator panel. Each pre-programmed job (schedule) can then through the FlexPendant be adjusted by changing the power  $\pm$  (wire feed in syneric mode) and individual voltage trim  $\pm$ .

If individual process parameters need to be adjusted within a job (schedule) except for the power and the voltage trim, the operator must then change the original pre-programmed schedules on the Fronius power source panel, or program a completely new schedule.

Following main tasks can be handled from the FlexPendant, Fronius TPS process user interface:

- Call for and use pre-programmed jobs (schedules) made in the Fronius TPS power source.
- Adjust power  $\pm$ , voltage trim  $\pm$  and absolute welding speed.

#### 14.3.5 Miller AutoAxxess [650-4]

---

##### General

The Miller AutoAxxess user interface is a standard Arc programming tool used for operating and managing one or more AutoAxxess-type of power sources.

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

The operator can through this software call for pre-programmed schedules, which have been made in the separate Miller power source operator panel. Each pre-programmed schedule can then through the FlexPendant be adjusted by changing the the wire feed speed in syneric mode  $\pm$  and individual voltage trim  $\pm$ .

If individual process parameters need to be adjusted within the schedule except for the wire feed speed and the voltage trim, the operator must then change the original pre-programmed schedule on the Miller power source, operator panel or program a new schedule.

Following main tasks can be handled from the FlexPendant, Miller AutoAxxess process user interface:

- Call for and use pre-programmed schedules made in the Miller AutoAxxess power source.
- Adjust wire feed speed in synergic mode  $\pm$ , voltage trim  $\pm$  and absolute welding speed.



#### 14.3.6 Miller DeltaWeld [650-5]

##### General

The Miller DeltaWeld user interface is a standard Arc programming tool used for operating and managing one or more DeltaWeld-type 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, Miller AutoAccess process user interface:

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

#### 14.3.7 RPB [650-6]

##### General

The RPB user interface is a standard Arc programming tool used for operating and managing one or more RPB-type of ABB 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, RPC process user interface:

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

#### 14.3.8 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.

#### 14.3.9 Simulated Welder [650-8]

##### General

The Simulated Welder user interface is a standard Arc programming tool used for simulating and managing a virtual power source supply.

## 14.4 Additional Arc Systems

### 14.4.1 Introduction

---

**General**

The Additional Arc System 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 equipments can be defined.

### 14.4.2 One additional [651-1]

---

**General**

This software option will give the user the possibility to define features in one robot system to control two different process equipments.

### 14.4.3 Two additional [651-2]

---

**General**

This software option will give the user the possibility to define features in one robot system to control three different process equipments.

## 14.5 BullsEye

### 14.5.1 Introduction

#### General

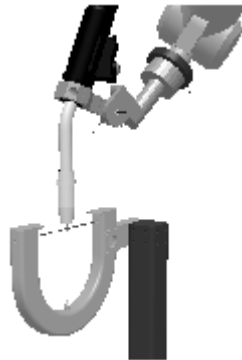
BullsEye™ provides completely automated Tool Center Point (TCP) definition for welding robots controlled by the IRC5 robot controller. By checking and updating the TCP alignment at regular intervals the robot will always operate with an accurate TCP.

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.

### 14.5.2 BullsEye [652-1]

#### General

This 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.



(A)



(B)

Figure 6 North American version and European version.

Pos	Description
A	North American version
B	European version

## 14 Application Options

---

### 14.5.2 BullsEye [652-1]

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

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

- Definition of tool data.
- Definition and programming of QuikCheck™ 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.

## 14.6 Torch Cleaner

### 14.6.1 Introduction

#### General

The Torch Cleaner 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 Cleaner option includes software that is loaded into all arc welding motion tasks.



Note that the Torch Cleaner software loads unique equipment drivers, but the RAPID instruction interface is the same regardless of hardware selected

Torch Cleaning 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 Cleaning software makes it possible to perform torch cleaning during positioner interchange, which reduces cycle time.

Once Torch Cleaning is programmed the operator can decide how often the cleaning process needs to take place to prolong the contact tip and gas nozzle life.

### 14.6.2 ABB - TC96 [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 - TC96. The hardware option is available in separate Application product price list.



Figure 7 ABB - TC96 Hardware.

The following main tasks can be handled from the ABB - TC96 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 cleanings.

## 14 Application Options

---

### 14.6.3 Binzel - TC97 [653-2]

#### 14.6.3 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.



Figure 8 Binzel - TC97 Hardware.

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 cleanings.

#### 14.6.4 Tweco - TC2000 [653-3]

---

##### 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, Tweco - TC2000. The hardware option is available in separate Application product price list.



Figure 9 Tweco - TC2000 Hardware.

The following main tasks can be handled from the Tweco - TC2000 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 cleanings.

### 14.6.5 Torch Clean [654-1]

---

**General** This software option will give the user the possibility to define only the torch clean procedure in a robot system.

### 14.6.6 Anitspatter [655-1]

---

**General** This software option will give the user the possibility to define only the anti spatter procedure in a robot system.

### 14.6.7 Wire Cutter [656-1]

---

**General** This software option will give the user the possibility to define only the wire cutting procedure in a robot system.giving the user the possibility to define a procedure automatic wire cutting for consistent wire electrode extenion (wire stick.out).

## 14.7 Torch Service Center

### 14.7.1 Introduction

---

**General**

The Torch Service Center software option provides a combined solution for all four service procedures.

It will include the following options:

- BullsEye [652-1]
- ABB-TC96 [653-1]
- Touch Clean [654-1]
- Anti Spatter [655-1]
- Wire Cutter [674-1]

### 14.7.2 Torch Service Center [674-1]

---

**General**

This software option will give the user the possibility to define and program the following four service procedures:

- Torch Cleaning operation
- Anit Spatter operation
- Wire Cutting operation
- BullsEye operation



Figure 10 Torch Service Center - Hardware.



## 14.8 SmarTac

### 14.8.1 Introduction

#### General

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.

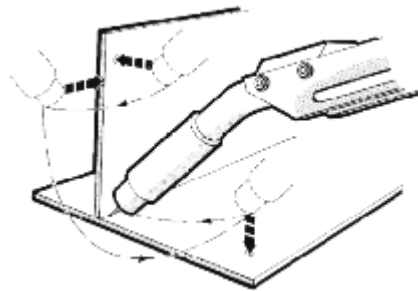


Figure 11

#### 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 fixturing costs. It can also help account for part variability that cannot otherwise be controlled.

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

## 14 Application Options

---

### 14.8.2 SmarTac - North American [657-1]

#### 14.8.2 SmarTac - North American [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.

#### 14.8.3 SmarTac - European [657-2]

---

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

The SmarTac package includes software that is loaded into all arc welding motion tasks, when the SmarTac option is selected. Process configuration parameters are used to connect real I/O signals and to modify the default settings.

RobotWare 5.06 introduces a new fully configurable I/O mapping feature not available in previous SmarTac versions. SmarTac I/O connections are now configured in the process configuration database (PROC). Actual I/O assignments to real I/O boards are not made by the SmarTac installation. The user or system designer must add these definitions to the EIO configuration database.

#### **RAPID instructions**

RAPID instructions included in this option:

<b>Instruction</b>	<b>Description</b>
Search_1D	One-dimensional search
Search_Part	Search for feature presence
Search_Groove	Find groove width and location
PDispAdd	Add program displacements
PoseAdd	Adds the translation portions of pose data
OFrameChange	Create a new, shifted object frame

## 14.9 Seam displacement

### 14.9.1 Introduction

#### General

The Seam Displacement option allows the operator to shift seams in relation to a reference frame. The displacements are applied via FlexPendant operator screens without stopping production.

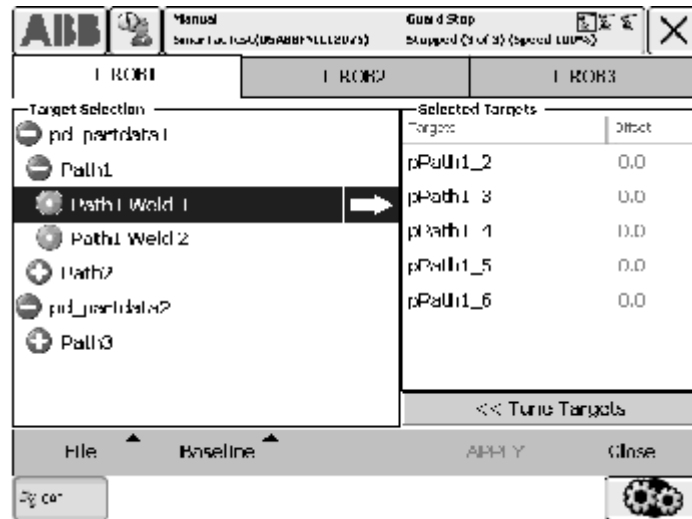


Figure 12

It is possible to shift an entire seam or targets within a seam individually. The operator can enter offsets at any point in time, whether the robot is welding or not. The applied changes will take effect in the next production cycle. This lets the operator visually inspect a part, apply seam offsets where needed, and the changes will take effect when the next part is welded. This functionality will help users to control part quality.

The functions available in Seam Displacement may be restricted by the user authorization system, UAS.

#### Features



##### Target selection:

- Select welds from the tree view and add them to the right-hand section by tapping the arrow.

Note that if a weld is used in more than one routine, it will appear in all places and any changes made to the offset will be the same for everywhere it is used.

##### Selected targets:

- Lists all selected targets within a weld and their current offset. Select the trash bin to remove the target from the selection.

##### File:

- It is possible to load selections of often used targets using the File menu. If your system uses UAS, this may be the only way to select targets for editing.

## 14 Application Options

---

### 14.9.1 Introduction

#### Baseline:

- Apply or reject the changes made to offset values:
- Restore to original and discard all changes to the currently selected target positions.
- Restore entire program and original to discard all changes to target positions (also applies to changes made in the program editor).
- Commit to current to apply all current changes to the selected target positions.
- Commit entire program to current to apply all changes to target positions (also applies to changes made in the program editor).

#### Tune targets:

- Select Tune Targets to display a keyboard for editing the offset values. The offset value is the length of the vector calculated from the x, y and z values changed in the Tune targets menu.

---

## 14.10 Production Monitoring

### 14.10.1 Introduction

---

**General**

The production monitoring option enables logging of production information in a WebWare 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 proactively 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
- Microsoft SQL database for data warehouse
- 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 productio data for traceability
- Data capture to document acceptance criteria

---

**WebWare tables in Production Monitor**

Production Monitor promotes WebWare tables that contain statistical results. For example, RobotWare Arc will provide a table of Cyle Results that combines information from the GAP-driven CycleEvents data with Arc-driven 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 send events for GAP 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

## 14 Application Options

### 14.10.1 Introduction

tables. These tables are typically written after an event has finished and usually include a Duration column that provides the time that elapsed during the event. Often the columns included in these tables will contain fields that are accumulated over the duration of an action.

**GAP Events Table** Every component that writes to tables in the WebWare database includes a column for the GAP Event ID. This number is incremented by the GAP Execution Engine each time the engine executes an event. GAP Events include the execution of a part cycle, a service routine or a setup routine.

When GAP 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 GAP reacts to a command, a start event is generated and logged in the GAPEvents table.

The GAPEvents table contains the records for all responses from requests to the GAP 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 GAP 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 <sup>1</sup>	string	User that is logged-in.
RobotID	string	Task name
ControllerID	string	Controller Name
Time	DateTime	Time stamp

1. UserID columns will be empty on 5.06. This information will be provided with 5.07.

### Cycle Events table

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

Column Name	Data Type	Description
EventID	Long Integer	A number supplied by GAP Execution Engine. May be used as a key.
CycleID	Long Integer	A number supplied by GAP Execution Engine. May be used as a key.

Column Name	Data Type	Description
Condition	Integer	Enumeration for Start or End
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
UserID <sup>1</sup>	string	User that is logged-in
RobotID	string	Task name
ControllerID	string	Controller Name
Time	DateTime	Time stamp

1. UserID columns will be empty on 5.06. This information will be provided with 5.07.

**Seam Results Table**

The SeamResults table contains a record for each weld seam that is finished. It provides information about the seam from RobotWare Arv. The value in the CycleID field will match the value in the CycleEvents table. This number may be provided to GAP from an external device, or by simple increments in the GAP Execution Engine. The value will be '0' if the seam was not executed within the context of a GAP cycle. Cycle information from the GAP 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 GAP Execution Engine. May be used as a key.
CycleID	Long Integer	A number supplied by GAP 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.

## 14 Application Options

### 14.10.1 Introduction

Column Name	Data Type	Description
Stops	Integer	Number of stops during welding for any reason.
Completed	boolean	True if all welds finished to completion.
UserID <sup>1</sup>	string	User that is logged in.
RobotID	string	Task name
ControllerID	string	Controller Name
Time	DateTime	Time stamp

1. UserID columns will be empty on 5.06. This information will be provided with 5.07.

#### Cycle Results Table

The CycleResults table contains a record for each part cycle made by the GAP 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 GAP from an external device, or by simple increments in the GAP Execution Engine.

GAP 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 GAP Execution Engine.
CycleID	Long Integer	A number supplied by GAP 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.



Column Name	Data Type	Description
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 <sup>1</sup>	string	User that is logged in.
RobotID	string	Task name
ControllerID	string	Controller Name
Time	DateTime	Time stamp

1. UserID columns will be empty on 5.06. This information will be provided with 5.07.

### Requirements

Robot Controller Requirements	Ethernet hardware
PC Interface	
Server PC Requirements	WebWare Server software
CPU 1 GHz +	
RAM 256 MB +	
Windows 2000/XP Professional or Server	
Client PC Requirements	WebWare Client software
Windows 98/NT/2000/XP	
Internet Explorer 6.0 or higher	

## 14.11 Optical Tracking

### 14.11.1 Introduction

---

**General**

The Optical Tracking option is a software allowing integration of external optical sensors, like for example, Serco Robot/M-Spot 90-, Scout-, Oxford- Laser sensor etc. This software includes a standard interface protocol with the same functionality for all optical sensors that by the supplier have been adapted to the ABB standard protocol. This standard protocol is after installation available in the robot system on the RS232 serial link.

---

**Features**

Key features include:

- Contour tracking in normal movements (path corrections).
- Adaptive process control during path motion and tracking.

---

**Limitations**

The Optical Tracking sensor interface can only be applied to one robot in a Multi-Move/MultiArc configuration.

## 14.12 MultiProcess [634-1]

---

<b>General</b>	MultiProcess is used for applying RobotWare-Arc to multiple robots in a MultiMove system.
<b>Features</b>	MultiProcess enables arc welding on any number of robots. The setup of which robot(s) that should have arc welding capabilities is done in the “Additional MultiMove selections” part of the Specification Form. It can also be carried out at the final installation in the RobotStudio-Online System Builder.
<b>Requirements</b>	Option 633-1 Arc must be specified, as well as any of the options 604-1 MultiMove Coordinated or 604-2 MultiMove Independent.
<b>Limitations</b>	In RW 5.05, MultiProcess does not apply to other process options than RW-Arc.

---

## 14 Application Options

---

### 14.11.1 Introduction

## 14.13 Spot [635-1]

---

<b>General</b>	<p>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 spotweld systems and process equipments.</p>
<b>Pneumatic guns</b>	<p>The RobotWare Spot option is used for sequential welding with one or several pneumatic gun equipments. For welding with several pneumatic guns at the same time use the option RobotWare Spot Multiple Guns.</p>
<b>Spotweld instructions</b>	<p>The RobotWare Spot option provides dedicated spotweld instructions for fast and accurate positioning combined with gun manipulation, process start and supervision of the weld equipment.</p> <p>Communication with the welding equipment is normally carried out by means of digital inputs and outputs but a serial interface is also available for some type of weld timers.</p>
<b>“Ready to use” functionality</b>	<p>The RobotWare Spot options are general and can be customized extensively . They have a default “ready to use” functionality directly after installation, but it is intended that some configuration data, RAPID data and RAPID routines have to be changed during the customizing.</p>
<b>Features</b>	<p>Some examples of useful functions are given below:</p> <ul style="list-style-type: none"><li>• Fast and accurate positioning using the unique QuickMove and TrueMove concept</li><li>• Gun pre-closing</li><li>• Quick start after a weld</li><li>• Handling of an on/off gun with two strokes</li><li>• Dual/single gun</li><li>• Manual actions for welding and gun control</li><li>• Simulation possibilities for test purposes</li><li>• Spot counters</li><li>• User-defined supervision and error recovery. Weld error recovery with automatic rewelding</li><li>• User-defined continuous supervision of the weld equipment, such as weld current signal and water cooling start <u>Note</u>: This feature requires the MultiTasking option.</li><li>• Wide customizing possibilities</li><li>• In a single robot system, both RobotWare Spot and RobotWare Dispense can be used simultaneously.</li><li>• A dedicated Spot GUI on the FlexPendant</li><li>• RobotWare Spot can also be used in a MultiMove system</li></ul>

---

**Application**

The RobotWare Spot functions will be controlled by separate internal program processes, which will run in parallel. For instance the robot movements, the continuous supervision and the spot welding will be handled in different independent processes. This means that if for instance the program execution and thus the robot movement is stopped, then the welding and supervision will continue until they come to a well defined process stop. For example, the welding process will carry on and finish the weld and open the gun, although the program has been stopped during the weld phase.

For well defined points in the welding sequence and movements, calls to user routines offer adaptations to the plant environment. A number of predefined parameters are also available to shape the behavior of the RobotWare Spot instructions.

**Performance**

Both the robot movement and the control of the spot weld equipment are embedded in the basic spot weld instructions SpotL and SpotJ.

The spot welding process is specified by:

- Spotdata: spot weld process data
- Gundata: spot weld equipment data

The system modules SWUSRC and SWUSRF: RAPID routines and global data for changing of process and test behavior.

System parameters: the I/O Signal configuration.

**Limitations**

- RobotWare Spot can be applied to only one robot in a MultiMove system.
- RobotWare Spot cannot be combined with other application options for other robots in a MultiMove system.

**RAPID instructions**

RAPID instructions included in this option:

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
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

## 14.14 Spot Multiple Guns [635-2]

---

<b>General</b>	The RobotWare Spot Multiple Guns package provides support for sequential welding with one or several pneumatic on/off gun equipments, as the RobotWare Spot package, but also welding and full individual monitoring of up to four separate gun equipments at the same time.
<b>Features</b>	<ul style="list-style-type: none"><li>• RobotWare Spot, see previous chapter</li><li>• Possibility to weld with up to four guns at the same time</li></ul>
<b>Application</b>	Both the robot movement and control of up to four spot weld processes are embedded in the basic spot weld instructions for multiple welding, SpotML and SpotMJ.
<b>Performance</b>	<p>As in RobotWare – Spot, the spotweld functions will be controlled by separate internal program processes, which will run in parallel. For instance the robot movements, the continuous supervision and each spotweld process will be handled in different independent program processes. This means that if for instance the program execution and thus the robot movements is stopped, then the weld processes and supervision will continue until they come to a well defined process stop. For example, the welding processes will carry on and finish the welds and open the guns, although the program was stopped during the weld phase.</p> <p>For well defined points in the welding sequence, calls to user routines offer adaptations to the plant environment. A number of predefined parameters are also available to shape the behavior of the RobotWare – Spot instruction.</p> <p>The opening and closing of the guns are always executed by RAPID routines. These gun routines may be changed from the simple on/off default functionality to a more complex gun control and they may contain additional gun supervision.</p> <p>RobotWare – Spot is based on the DAP (Discrete Application Platform).</p>

---

**System modules** The system modules SWUSRF and SWUSRC: RAPID routines and global data for customizing purposes and data for changing of process and test behavior.  
System parameters: the I/O Signal configuration.

---

**RAPID instructions** RAPID instructions included in this option.  
RobotWare Spot MultiMove Guns package contains the same instructions as RobotWare Spot and some additional instructions, as listed below:

Instruction	Description
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 equipments 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 equipments at the end position

## 14 Application Options

---

### 14.11.1 Introduction

## 14.15 Spot Servo [635-3]

---

<b>General</b>	The RobotWare Spot option is a general and flexible software platform for the creation of customized and easy to use function packages for different types of spotweld systems and process equipments.
<b>Sequential welding</b>	The RobotWare Spot Servo option is used for sequential welding with one or several servo gun equipment. For welding with two or up to four servo guns at the same time, use the RobotWare – Spot Servo Multiple Guns option.
<b>Spotweld instructions</b>	The RobotWare Spot Servo option provides dedicated spotweld instructions for fast and accurate positioning combined with gun manipulation, process start and supervision of the different gun equipments. Communication with the welding equipment is carried out by means of digital inputs and outputs.
<b>“Ready to use” functionality</b>	It should be noted that the RobotWare – Spot options are general and can be extensively customized. They have a default “ready to use” functionality directly after installation but it is intended that some configuration data, RAPID data and RAPID routines must be changed during the customizing.
<b>Features</b>	<ul style="list-style-type: none"><li>• Fast and accurate positioning using the unique QuickMove and TrueMove concept</li><li>• Gun pre-closing, i.e having the gun closing synchronized with weld position</li><li>• Gun equalizing, i.e. having the gun “floating” around the weld position</li><li>• Constant tip force during welding</li><li>• Manual actions for welding and gun control</li><li>• Several simulation possibilities for test purposes</li><li>• Reverse execution with gun control</li><li>• Weld error recovery with automatic rewelding</li><li>• User-defined supervision and error recovery</li><li>• User-defined autonomous supervision, such as weld current signal and water cooling start</li><li>• Wide customizing possibilities</li><li>• Default “ready to use” functionality directly after installation</li><li>• Detecting of missing or improper plates</li><li>• Gun calibration functions</li><li>• Spot counters and tip wear data for each used gun</li><li>• Fast switch between two servo guns with a tool changer <u>Note:</u> This feature requires the Servo Tool Change option.</li><li>• In a single robot system, both RobotWare Spot and RobotWare Dispense can be used simultaneously.</li><li>• A dedicated Spot Servo GUI on the FlexPendant</li></ul>

---



- RobotWare Spot Servo can also be used in a MultiMove system.

### Application

Both the robot movement and the control of the spot weld equipment are embedded in the basic spot weld instructions SpotL and SpotJ.

The spot welding process is specified by:

System Parameters	Description
Spotdata	spot weld process data
Gundata	spot weld equipment data

- The system modules SWDEFINE and SWDEFUSR: RAPID routines and global data for customizing purposes e.g. adaptations for specific process equipment.
- The system module SWUSER: RAPID routines and global data for changing of process and test behavior.
- System parameters: the I/O Signal configuration and the Manipulator configuration.

### Performance

The RobotWare Spot Servo functions will be controlled by separate internal program processes, which will run in parallel. For instance the robot movements, the continuous supervision and the spotwelding will be handled in different independent processes. This means that if, for instance, the program execution and thus the robot movement is stopped, then the welding and supervision will continue until they come to a well defined process stop. For example, the welding process will carry on and finish the weld and open the gun, although the program stopped during the weld phase.

For well defined points in the welding sequence and movements, calls to user routines offer adaptations to the plant environment. A number of predefined parameters are also available to shape the behavior of the RobotWare Spot instructions.

### Limitations

- RobotWare Spot Servo can be applied to only one robot in a MultiMove system.
- RobotWare Spot Servo cannot be combined with other application options for other robots in a MultiMove system.

### RAPID instructions

RAPID instructions included in this option:

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.
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.

## 14 Application Options

---

### 14.11.1 Introduction

<b>Instruction</b>	<b>Description</b>
SetForce	Close the gun a predefined time then open the gun.
CalibL	Calibrate the gun during linear movement to the programmed position.
CalibJ	Calibrate the gun during non-linear movement to the programmed position.
Calibrate	Calibrate the gun in current position without movement.
STTune	Tune motion parameters for the servo gun.
STTuneReset	Reset tuned motion parameters for the servo gun.

## 14.16 Spot Servo Multiple Guns [635-4]

### General

The RobotWare Spot Servo Multiple Guns package provides support for sequential welding with one or several servo gun equipment, such as the RobotWare Spot Servo package, but also welding with up to four servo guns at the same time.

### Features

- RobotWare – Spot Servo, see previous chapter
- Possibility to weld with two servo guns at the same time

### Application

Both the robot movement and the control of the spot weld equipments are embedded in the basic spot weld instructions. SpotL and SpotJ are used for sequential welding. With SpotML or SpotMJ it is possible to weld with several guns simultaneously.

Each spot welding process is specified by:

System Parameters	Description
Spotdata	spot weld process data
Gundata	spot weld equipment data

- The system modules SWDEFINE and SWDEFUSR: RAPID routines and global data for customizing purposes e.g. adaptations for a specific process equipment.
- The system module SWUSER: RAPID routines and global data for changing of process and test behaviour.
- System parameters: the I/O Signal configuration and the Manipulator configuration.

### Performance

As in RobotWare Spot Servo Multiple Guns the spotwelding functions will be controlled by separate internal program processes, which will run in parallel. For instance the robot movements, the continuous supervision and the spotwelding will be handled in different independent processes. This means that if, for instance, the program execution and thus the robot movements is stopped, then the weld processes and supervision will continue until they come to a well defined process stop. For example, the welding processes will carry on and finish the weld and open the guns, although the program was stopped during the weld phase.

## 14 Application Options

---

### 14.11.1 Introduction

#### **RAPID instructions**

---

RAPID instructions included in this option:

The RobotWare Spot Servo Multiple Guns package contains the same instructions as RobotWare Spot Servo and some additional instructions, as listed below:

<b>Instruction</b>	<b>Description</b>
SpotML	Control the motion, gun closure/opening and 1 - 2 welding processes. Move the TCP along a linear path and perform spot welding with 1 - 2 gun equipment at the end position
SpotMJ	Control the motion, gun closure/opening and 1 - 2 welding processes. Move the TCP along a non-linear path and perform spot welding with 1 - 2 gun equipment at the end position

## 14.17 Spot Servo Equalizing [635-5]

### 14.17.1 Introduction

#### General

This option, Spot Servo Equalizing, is identical to the option 635-3, Spot Servo, but with the addition of the gun equalizing feature 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

#### SpotWare Servo Equalizing Features

- 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.

## 14.18 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
- 4 guns 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
- In a single robot system, both RobotWare Dispense and RobotWare Spot/Spot Servo can be used.
- A dedicated Dispense GUI on the FlexPendant
- RobotWare Dispense can also be used in 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 gluing process is specified by:

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

### Limitations

- RobotWare Dispense can be applied to only one robot in a MultiMove system.
- RobotWare Dispense cannot be combined with other ProcessWare for other robots in a MultiMove system.

### **RAPID instructions**

RAPID instructions included in this option:

<b>Instruction</b>	<b>Description</b>
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

## 14.19 Prepared for PickMaster [642-1]

---

### General

Prepared for PickMaster supports the PickMaster application software residing on a PC.

Prepared for PickMaster features together with the PickMaster application configurable integration of robots, vision and conveyors.

The option contains all necessary functions and further options to interface PickMaster and to run enhanced conveyor tracking.

PickMaster is approved for running on one IRC5 controller with multiple robots.

For further information see the Product Specification for PickMaster 3.0.

---

### Features

- 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 [609-1]
  - Fixed Position Events [642-1]
-



---

<b>Application</b>	Aimed for packaging applications, typically high-speed picking. Further, PickMaster is the ideal standard vision integration for all robot types.
<b>Performance</b>	See PickMaster 3.0 Product Specification.
<b>Requirements</b>	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.
<b>PickMaster 3.0 User's Guide</b>	The PickMaster User's Guide describes the application as well as related RAPID data and instructions in detail.  RAPID Data: <ul style="list-style-type: none"><li>• itmsrc</li><li>• itmsrcdat</li><li>• itmtgt</li><li>• selectiondata</li><li>• sortdata</li></ul> RAPID instruction: <ul style="list-style-type: none"><li>• AcklTmTgt</li><li>• CreateInstDat</li><li>• FlushTmSrc</li><li>• FreeAllTmSrc</li><li>• FreeTmSrc</li><li>• GetAllTmSrc</li><li>• GetlTmTgt</li><li>• ItmSrcByName</li><li>• NewlTmSrc</li><li>• NextlTmTgtType</li><li>• QstartlTmSrc</li><li>• QstoplTmSrc</li><li>• ResetlTmSrc</li></ul>

---

## 14.20 RobotWare Plastics Mould [643-1]

---

### General

RobotWare Plastics Mould provides easy programming and operation of ABB robots used for machine tending, especially for plastics injection moulding applications. RobotWare Plastics Mould reduces learning curve, installation and programming time and improves error recovery.

### Features

- Pre-configured eio.cfg for Euromap 12/67 and SPI
- Graphical user interface for programmer and operator
- Possible to define typical machine tending robot cycles
- Programmer wizard to guide user through program creation (program can also be created and edited in traditional way)
- User Authorisation System (levels: operator, machine setter, programmer)
- Automatic and safe home run
- Production window that displays robot movement in real time by the active station indication
- Production statistics
- Event log
- Signal status
- Status and error indication in colors

### Performance

RobotWare Plastics Mould is both a way to modularise machine tending program structure and hide the RAPID layer from the user. A RobotWare Plastics Mould program is built up by a number of stations (representing a machine or a post-processing application) and tool modules from the station library. The station modules contain the machine tending process knowledge. These station modules are combined as building blocks to form all necessary robot cycles (production, warm-up, quality check etc.) RobotWare Plastics Mould functions as a state machine that depending on the station priority and status optimizes the robot cycle. The inexperienced robot operator never needs to see the RAPID structure that is hidden by the graphical user interface from which the robot is run and programmed.

### Limitations

- RobotWare Plastics Mould is not available as MultiMove system
- RobotWare Plastics Mould is available for IRC5 and IRB 140, 1400, 1600, 2400, 4400, 6600 (including different variants)
- Available languages: Chinese, English, French, German, Spanish, Italian, Swedish

**A**

Advanced RAPID, 49  
Arc Advanced Tracking, 10  
ARCITEC, 40  
Axis Reset, 29

**C**

calibration concept, 16

**F**

File Transfer Protocol, 35

**H**

Home position, 25

**L**

line tracking, 23

**M**

MultiMove system, 19, 21

**N**

Network File System, 36

**O**

OPC Server, 37

**P**

Pallet, 10  
path corrections, 32  
PC SDK, 37

**R**

RobotStudio Online, 37  
RobotWare – OS, 7

**S**

SCADA, 37

**U**

upgrading, 7

**W**

WebWare Server, 37







ABB Automation Technologies AB  
Robotics  
S-721 68 VÄSTERÅS  
SWEDEN  
Telephone: +46 (0) 21 344000  
Telefax: +46 (0) 21 132592

3HAC 022349-001, Revision 2, en