Course Description

Course Duration
The duration is 5 days.

Course Type
This is an instructor led course. Approximately 75% of the time is used for hands-on labs and exercises.

Course Goal
The goal of this course is to teach students the basics in programming industry robots with IRC5 controller.

Student Profile
This training is targeted to robot programmers and service engineers that need to modify robot programs.

Prerequisites and Recommendations
Previous experience working with the robot is helpful, along with understanding of Windows or computer programming.
Recommended previous courses:
IRC5 Operating and/or IRC5 Upgrade from S4Cplus

Course Objectives
Upon completion of this course, students will be able to:
- Practice all areas of safety as it pertains to the robot.
- Properly start-up, operate, and shut down the robot.
- Properly identify and recover from robot errors.
- Perform program storage and retrieval.
- Manual and program control of inputs and outputs.
- Define tools and work objects.
- Edit programmed positions.
- Create a program with subroutine structure.
- Program instructions, such as, output control, decision-making, operator dialog, and clock.
- Name I/O and data with proper names.
- Use Module programming techniques.

Main Topics
- Basics in robot programming
- Structured programming
- Conditional control
- Digital I/O signals
- Communication with operator
- Define and use tools
- Define and use work objects
- Create/define offset
- Event log
- Backup and restore

www.abb.com/abbuniversity
## Course Calendar

### R102 IRC5 Programming Basic

<table>
<thead>
<tr>
<th>Day 1</th>
<th>Day 2</th>
<th>Day 3</th>
</tr>
</thead>
</table>
| • Introduction  
• Safety  
• Getting started  
• Exercise 1A – E Understand MoveInstr, coordinate system etc. | • Create and use tools  
• Exercise 1F – L Add and use tools, understand and use the tool coordinate system, how the tool can be stored in a system module and why.  
• Structured programming Redefine tool (4-p method)  
• Exercise 2A | • Program flow and Conditional control with I/O  
• Exercise 3.1A – D New routines, keep track of things, filter out signals.  
• Conditional control with FlexPendant  
• Exercise 3.2A – C Communication with operator |

<table>
<thead>
<tr>
<th>Day 4</th>
<th>Day 5</th>
</tr>
</thead>
</table>
| • Work object  
• Exercise 4.1A – C Create, use and define/adjust a work object.  
• Offset  
• Exercise 4.2A – C Create/define and use offset | • Final exercise  
• Backup and restore  
• Rev counter update |