

This document can serve as an example of adding an EthernetIP module to an Allen-Bradley IO list and configuring a Generic Ethernet Module along with the associated PLC tags. It shows how the Generic Module Input and Output memory size relates to the tags which get created.

The below configuration is from a UR demo application, UR.ACD, downloaded from their website. This is what an application should look like if it is setup to use EthernetIP between AB and UR.

The image displays three screenshots from the Studio 5000 software interface:

- Top Left:** The Controller Organizer tree view. The 'ETHERNET-MODULE UR' module is highlighted at the bottom. A red arrow points from this module to the first dialog box.
- Top Right:** The 'Module Properties Report: Local (ETHERNET-MODULE 1.001)' dialog box, 'General' tab. The 'Name' field is set to 'UR' and is circled in red. Other fields include 'Type: ETHERNET-MODULE Generic Ethernet Module', 'Vendor: Rockwell Automation/Allen-Bradley', 'Parent: Local', 'Description', 'Comm Format: Data - SINT', and 'Address / Host Name' (IP Address: 10.1.1.237). Connection parameters are shown on the right: Input (100, 480 (8-bit)), Output (112, 224 (8-bit)), and Configuration (1, 0 (8-bit)).
- Middle Right:** The same dialog box, 'Connection' tab. It shows 'Requested Packet Interval (RPI): 2.0 ms (1.0 - 3200.0 ms)', 'Inhibit Module' (unchecked), 'Major Fault On Controller if Connection Fails While in Run Mode' (unchecked), and 'Use Unicast Connection over EtherNet/IP' (checked).
- Bottom Left:** The 'Data Types' list under 'User-Defined'. It includes various UR-specific types like UR_BitRegisters, UR_Coordinate, UR_FloatRegisters, UR_IntRegisters, UR_O2T_Assembly_Combined, UR_O2T_Outputs, UR_O2T_Robot, UR_T2O_Assembly_Combined, UR_T2O_IO, UR_T2O_Joints, UR_T2O_Robot, UR_T2O_Safety, UR_T2O_TCP, and UR_T2O_Tool.
- Bottom Right:** The 'Module Properties Report: Local (ETHERNET-MODULE 1.001)' dialog box, 'Module Info' tab. It shows fields for Identification (Vendor, Product Type, Product Code, Revision, Serial Number, Product Name) and Status (Major Fault, Minor Fault, Internal State, Configured, Owned, Module Identity). Buttons for 'Refresh' and 'Reset Module' are at the bottom.

Annotations:

- Red text below the top left screenshot: "This module can be imported from Ethernet-ModuleUR.L5X. I exported this from the UR.ACD application."
- Red text below the bottom left screenshot: "This user defined type can be imported from the file UR_DataTypes.L5X"

Controller Organizer

Controller Tags - Test(controller)

Scope: Test Show: All Tags

Name	Data Type	Description
Local:1:C	AB:Embedded_DiscretelO:C:0	
Local:1:I	AB:Embedded_DiscretelO:I:0	
Local:1:O	AB:Embedded_DiscretelO:O:0	
UR:C	AB:ETHERNET_MODULE:C:0	
UR:C.Data	SINT[400]	
UR:I	AB:ETHERNET_MODULE_SINT 480Bytes:I:0	
UR:I.Data	SINT[480]	
UR:O	AB:ETHERNET_MODULE_SINT 224Bytes:O:0	
UR:O.Data	SINT[224]	
UR_In	UR_O2T_Assembly_Combined	
UR_In.Robot	UR_O2T_Robot	Universal Robot - O2T sub-structure ...
UR_In.Outputs	UR_O2T_Outputs	Universal Robot - O2T sub-structure ...
JR_In.Bit	UR_BitRegisters	Universal Robot - sub-structure "Bit ...
UR_In.Int	UR_IntRegisters	Universal Robot - sub-structure "Int ...
UR_In.Float	UR_FloatRegisters	Universal Robot - sub-structure "Floa...

Module Properties Report: Local (ETHERNET-MODULE 1.001)

General Connection Module Info

Type: ETHERNET-MODULE Generic Ethernet Module
 Vendor: Rockwell Automation/Allen-Bradley
 Parent: Local
 Name: JR
 Description:

Connection Parameters

Assembly Instance: 100
 Input: 480 (8-bit)
 Output: 112
 224 (8-bit)
 Configuration: 1 0 (8-bit)

Address / Host Name
 IP Address: 10 . 1 . 1 . 237

Controller Organizer

Program Parameters and Local Tags - MainProgram

Scope: MainProgram Show: All Tags

Name	Value	Force Mask	Style	Data Type
URI	{...}	{...}	{...}	UR_T2O_Assembly_Combin
URI.Robot	{...}	{...}	{...}	UR_T2O_Robot
URI.Safety	{...}	{...}	{...}	UR_T2O_Safety
URI.IO	{...}	{...}	{...}	UR_T2O_IO
URI.Tool	{...}	{...}	{...}	UR_T2O_Tool
URI.Joints	{...}	{...}	{...}	UR_T2O_Joints
URI.TCP	{...}	{...}	{...}	UR_T2O_TCP
URI.Bit	{...}	{...}	{...}	UR_BitRegisters
URI.Int	{...}	{...}	{...}	UR_IntRegisters
URI.Float	{...}	{...}	{...}	UR_FloatRegisters
URO	{...}	{...}	{...}	UR_O2T_Assembly_Combin
URO.Robot	{...}	{...}	{...}	UR_O2T_Robot
URO.Outputs	{...}	{...}	{...}	UR_O2T_Outputs
URO.Bit	{...}	{...}	{...}	UR_BitRegisters
URO.Int	{...}	{...}	{...}	UR_IntRegisters
URO.Float	{...}	{...}	{...}	UR_FloatRegisters

User-Defined

- UR_BitRegisters
- UR_Coordinate
- UR_FloatRegisters
- UR_IntRegisters
- UR_O2T_Assembly_Combined
- UR_O2T_Outputs
- UR_O2T_Robot
- UR_T2O_Assembly_Combined
- UR_T2O_IO
- UR_T2O_Joints
- UR_T2O_Robot
- UR_T2O_Safety
- UR_T2O_TCP
- UR_T2O_Tool

Any given UR system has a built in set of IO available to other systems such as a PLC over EthernetIP. Look up complete list. If more is needed it can likely be achieved using Modbus TCP.

Note: **UR only supports Ethernet/IP Class 1 implicit or "I/O" messaging, not Ethernet/IP Class 3 explicit or "client/server" messaging.**

EthernetIP connection between AB PLC and UR robot.

Name	Usage	Value	Force Mask	Style	Data Type
URI	Local	{...}	{...}		UR_T20_Assembly_...
URO	Local	{...}	{...}		UR_O2T_Assembly_...

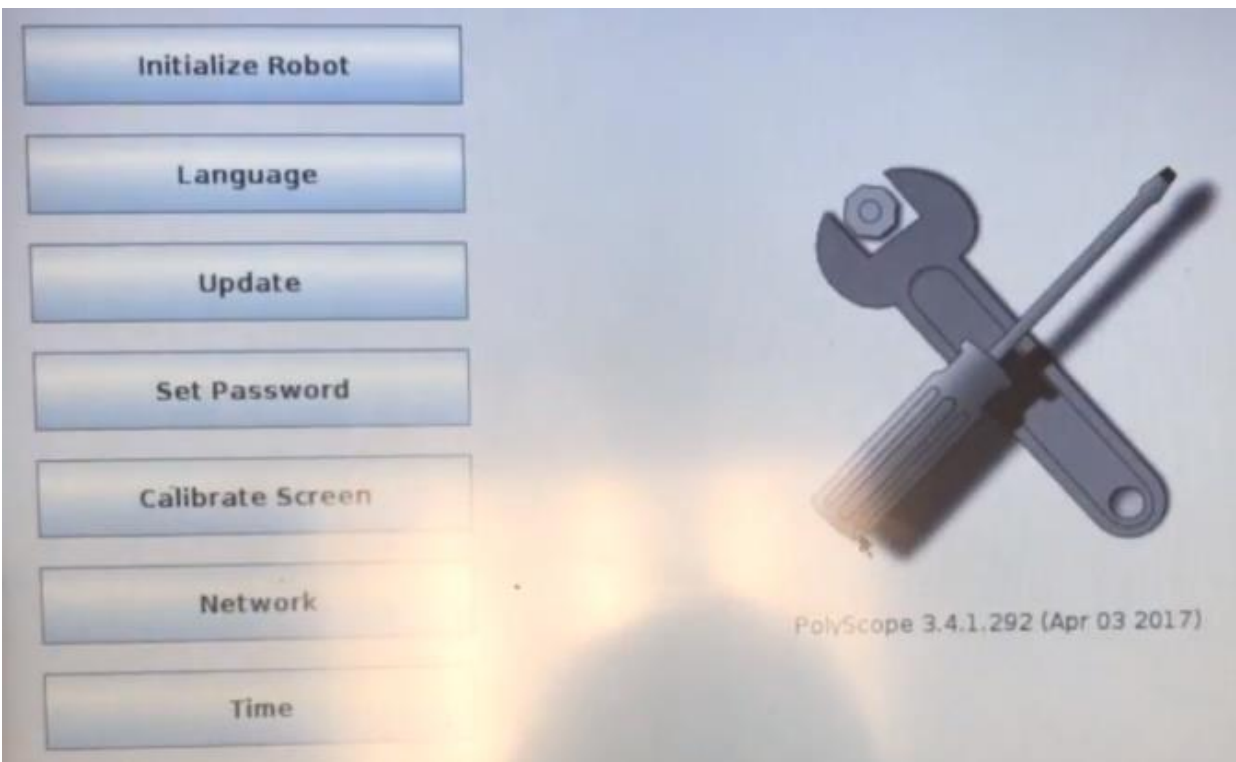
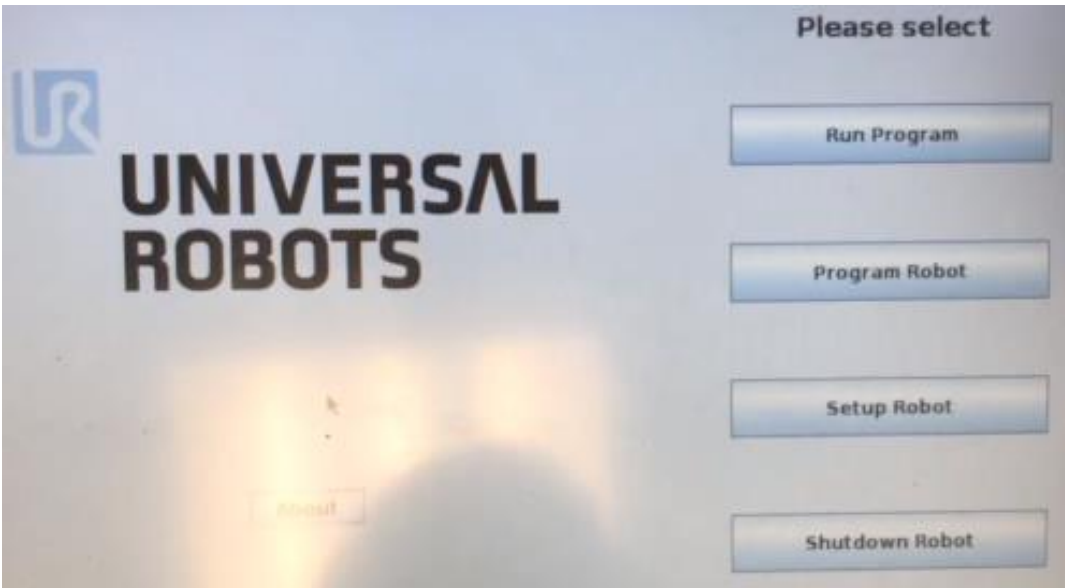
The result of adding the network connection is that IO tags are added to the Parameters and Local Tags group under Main Program or Main Tasks. Slightly different than most where the tags would appear under Controller Tags.

							Data Types	
Name	INPUTS	Usage	Value	Force Mask	Style	Data Type	Description	
URI		Local	{...}	{...}		UR_T20_Assembly_...		
URI.Robot			{...}	{...}		UR_T20_Robot	Universal Robot - T20 sub-structure "...	
URI.Safety			{...}	{...}		UR_T20_Safety	Universal Robot - T20 sub-structure "...	
URI.IO			{...}	{...}		UR_T20_IO	Universal Robot - T20 sub-structure "...	
URI.Tool			{...}	{...}		UR_T20_Tool	Universal Robot - T20 sub-structure "...	
URI.Joints			{...}	{...}		UR_T20_Joints	Universal Robot - T20 sub-structure "...	
URI.TCP			{...}	{...}		UR_T20_TCP	Universal Robot - T20 sub-structure "...	
URI.Bit			{...}	{...}		UR_BitRegisters	Universal Robot - sub-structure "Bit R...	
URI.Int			{...}	{...}		UR_IntRegisters	Universal Robot - sub-structure "Int R...	
URI.Float			{...}	{...}		UR_FloatRegisters	Universal Robot - sub-structure "Float...	
URO		Local	{...}	{...}		UR_O2T_Assembly_...		

Scope: **MainProgram** Show: **All Tags** v. Enter Name Filter

							Data Types	
Name	Usage	Value	Force Mask	Style	Data Type	Description		
URI	Local	{...}	{...}		UR_T2O_Assembly_...			
URO	Local	{...}	{...}		UR_O2T_Assembly_...			
URO.Robot		{...}	{...}		UR_O2T_Robot	Universal Robot - O2T sub-structure "...		
URO.Outputs		{...}	{...}		UR_O2T_Outputs	Universal Robot - O2T sub-structure "...		
URO.Bit		{...}	{...}		UR_BitRegisters	Universal Robot - sub-structure "Bit R...		
URO.Int		{...}	{...}		UR_IntRegisters	Universal Robot - sub-structure "Int R...		
URO.Float		{...}	{...}		UR_FloatRegisters	Universal Robot - sub-structure "Float...		

You will have the IO Not Responding indication on the PLC until you enable the EthernetIP connection on the robot side. On the robot side the first thing we want to do is set up the IP address.



Network

Select your network method

- DHCP
- Static Address
- Disabled network

Network is connected

Network detailed settings:

IP address:	192.168.0.100
Subnet mask:	255.255.255.0
Default gateway:	0.0.0.0
Preferred DNS server:	0.0.0.0
Alternative DNS server:	0.0.0.0

Initialize Robot

Language

Update

Set Password

Calibrate Screen

Network

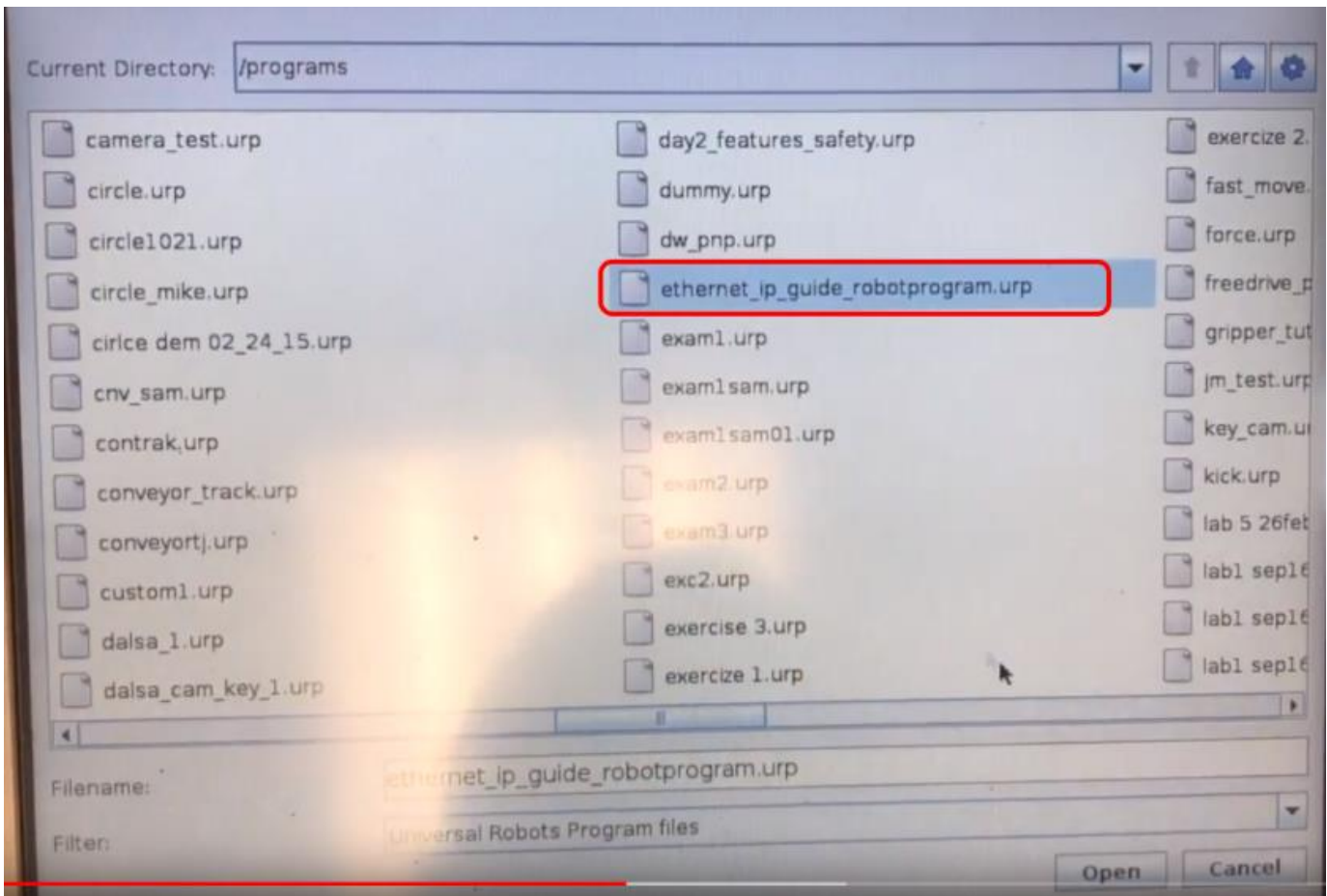
Time

URCaps

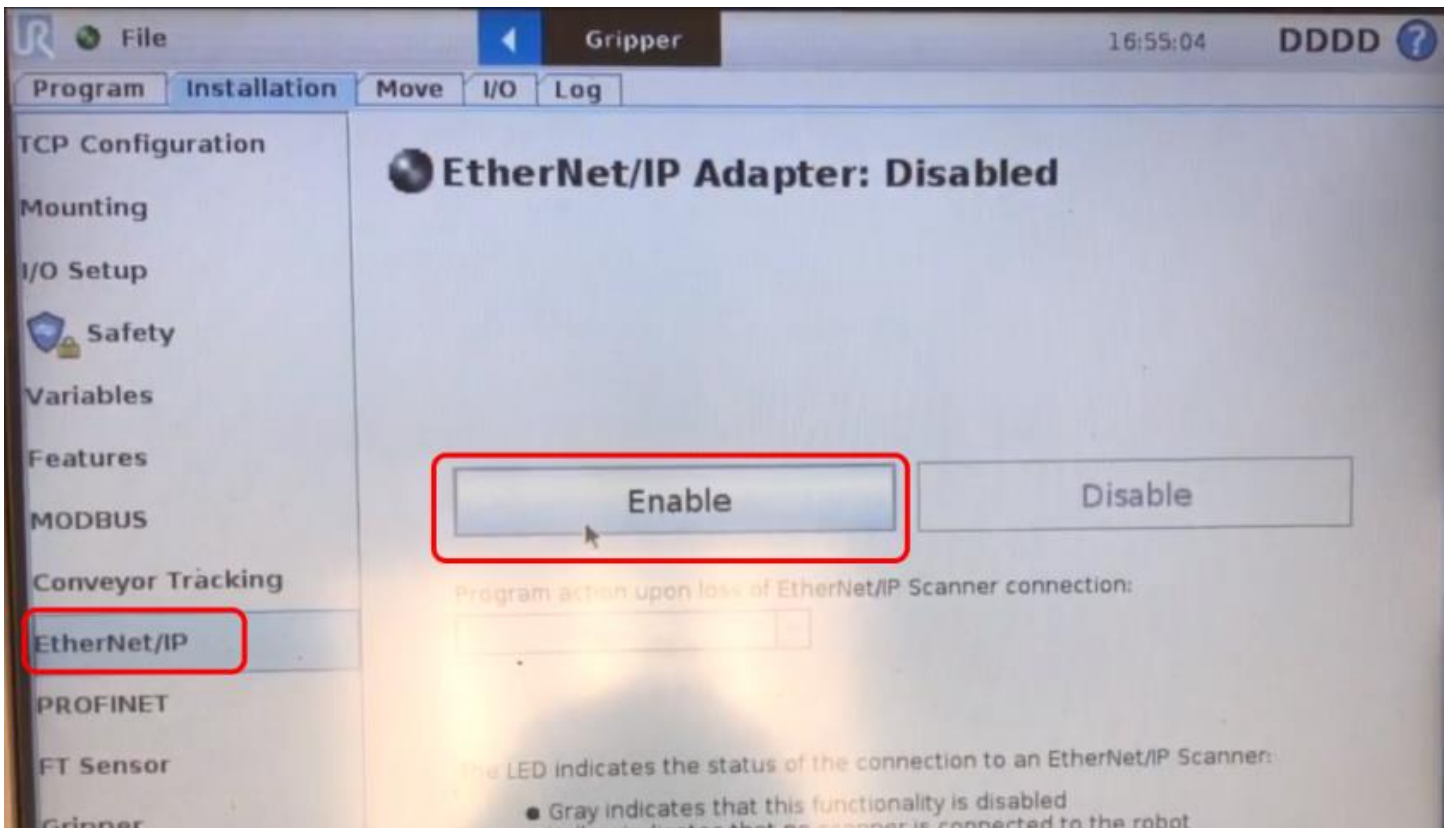
New Program

Load From File

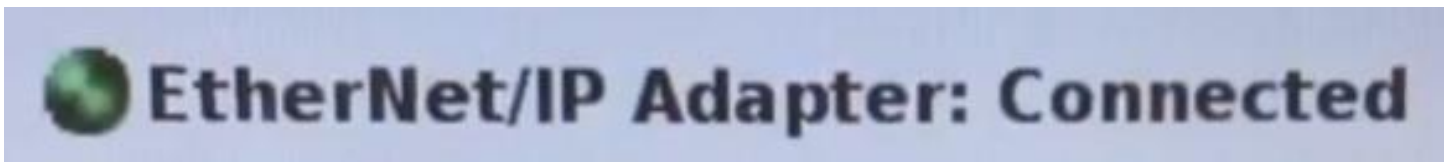
Use Template



Here we are loading the file which gets edited when we enter our IP information as we did above. (think this is how it works).



Under Installation | EthernetIP we click Enable.



Confirms that the driver has been enabled.

Parameter values in PLC are initialized from the robot before each run, this prevents running to old values.

In this example an Input to the PLC is an Output from the robot.

Name	Usage	Value	Force Mask	Style	Data Type	Description
URI TCP		{...}	{...}		UR_T2O_TCP	Universal Robot - T2O sub-structure "...
URI Bit		{...}	{...}		UR_BitRegisters	Universal Robot - sub-structure "Bit R...
URI Bit Registers		{...}	{...}	Decimal	BOOL[64]	Universal Robot - sub-structure "Bit R...
URI.Bit.Registers[0]		0		Decimal	BOOL	Universal Robot - sub-structure "Bit R...
URI.Bit.Registers[1]		0		Decimal	BOOL	Universal Robot - sub-structure "Bit R...
URI.Bit.Registers[2]		1		Decimal	BOOL	Done signal from robot
URI.Bit.Registers[3]		0		Decimal	BOOL	Universal Robot - sub-structure "Bit R...
URI.Bit.Registers[4]		0		Decimal	BOOL	Universal Robot - sub-structure "Bit R...

Name	Usage	Value	Force Mask	Style	Data Type	Description
URI Int		{...}	{...}		UR_IntRegisters	Universal Robot - sub-structure "Int R...
URI Int Registers		{...}	{...}	Decimal	DINT[24]	Universal Robot - sub-structure "Int R...
URI.Int.Registers[0]		10		Decimal	DINT	Parts placed counter
URI.Int.Registers[1]		3		Decimal	DINT	Parts inspected counter
URI.Int.Registers[2]		0		Decimal	DINT	Universal Robot - sub-structure "Int R...
URI.Int.Registers[3]		0		Decimal	DINT	Universal Robot - sub-structure "Int R...