

Bristol OpenEnterprise Reference Guide

Allen-Bradley RDI

Contents

1	Allen-Bradley RDI	3
1.1	Overview	3
1.2	Overview	3
1.3	System Requirements.....	3
1.3.1	Rockwell Software RSLinx ®.....	3
1.3.2	Database Configuration.....	3
1.3.3	Allen-Bradley RDI application (abrdi.exe)	3
1.3.4	Services File Entry.....	3
1.3.5	Session Manager Configuration	3
1.4	RSLinx Driver Configuration	4
1.5	Database Configuration	4
1.5.1	abDriver_Table	4
1.5.2	ABNetwork_Table.....	4
1.5.3	ABdevice_Table	5
1.5.4	dvi_schedule.....	5
1.5.5	ABPollList_Table	6
1.5.6	ABIntegerAnalog_Table	6
1.5.7	ABRealAnalog_Table	7
1.5.8	ABDigital_Table.....	7
1.5.9	ABComplex_Table.....	8
1.6	Running the Allen-Bradley RDI	9
2	Index	10

1 Allen-Bradley RDI

1.1 Overview

The OpenEnterprise Allen-Bradley RDI (filename: abrdi.exe), has been designed to communicate with Allen-Bradley PLC-5s over any of the interface cards and protocols supported by RSLinx ®.

1.2 Overview

The OpenEnterprise Allen-Bradley RDI (filename: abrdi.exe), has been designed to communicate with Allen-Bradley PLC-5s over any of the interface cards and protocols supported by RSLinx ®.

1.3 System Requirements

The following environment is required to run the Allen-Bradley RDI.

1.3.1 Rockwell Software RSLinx ®

As a minimum the OEM type must be installed. See the [RSLinx ® Driver Configuration](#) topic.

1.3.2 Database Configuration

- If the Database Project Builder (DPB) is used to build the database select the Allen-Bradley Device Driver.
- If the DPB is not being used, then you need to use the SQL Client to firstly 'include' the *ab_tables.sql* script, found in the OpenEnterprise Database directory into the OpenEnterprise Database. This script contains the Allen-Bradley RDI schema.
- Then you need to make sure to add a reference to the *ab.cl* file at the correct place in the *poly.cfg* file.
- Finally, you need to include the *ab_insert.sql* script. This will add the Allen-Bradley Driver to the database.

1.3.3 Allen-Bradley RDI application (abrdi.exe)

This is included in the OpenEnterprise Server installation, along with the *ociapi.dll*, which is also required in order for it to function correctly.

1.3.4 Services File Entry

The OpenEnterprise installation should have written an entry for the 'abrdi' to the Services file in *winnt\system32\drivers\etc*. This is the TCP/IP service name to which OE attempts to connect.

1.3.5 Session Manager Configuration

The Allen-Bradley RDI is an application which should be configured to run as part of an OpenEnterprise Session on the Master Server under the control of the Session Manager. Refer to Session Manager Help for further details.

1.4 RSLinx Driver Configuration

To configure a driver, run the Rockwell Software RSLinx ® program and select Communications > Configure Drivers.

Select a driver from the available driver list and complete the information required in the displayed dialog box. All drivers will require a name that will be the same as that name entered in the abNetwork_Table.

The dialog box then varies depending on the driver selected. For instance, for the Ethernet driver the dialog will prompt for PLC TCP/IP address mapping to Driver station numbers.

1.5 Database Configuration

The following tables are added by the default SQL script ab_tables.sql. This is included when the Allen-Bradley option is chosen on the Device Drivers page when running the Database Project Builder wizard, or it can be included from the SQL Client, if building the database manually. The following gives a description of the tables and their use.

1.5.1 abDriver_Table

There is a single driver table entry added by the ABDef.sql script as a part of the initial OE Database build process. This SQL script inserts the mandatory OpenEnterprise Database requirement for using the Allen-Bradley RDI. Other AB table entries are project specific.

The following columns must contain valid data:

drivename - table key.

address - TCP/IP host and service name that the GDI will attempt to connect with. Set to 'abrdi' if the driver is running locally. Make sure the Services file in winnt\system32\drivers\etc contains an entry for 'abrdi'.

retryperiod - This determines how frequently the GDI will attempt to connect with the RDI.

Example:

```
insert into ABDriver_Table(drivename,address,retryperiod)
values ('ABRDI','abrdi',seconds(10));
commit;
```

1.5.2 ABNetwork_Table

There will be a table entry for each configured RSLinx ® Driver. The following columns must contain valid data:

networkname - table key.

drivename - ABDriver_Table key.

driver_id - unique number for RSLinx ® Driver between 0 and 15.

driver_name - name for RSLinx ® Driver assigned during configuration.

Example:

```
insert into ABNetwork_Table
(networkname, drivename, driver_id, driver_name) values
('TEST1', 'ABRDI', 1, 'AB_ETH-1');
```

1.5.3 ABdevice_Table

There will be a table entry for each PLC or redundant pair. The following columns must contain valid data:

devicename - table key.

networkname - ABNetwork_table key.

deviceaddress - the communications path that will be used by the RSLinx ® to access the PLC (see topic entitled Allen-Bradley PLC Communication Paths).

redundant - Single PLC (false) or redundant pair (true).

b_addr_info(required if redundant pair) - path that will be used by the RSLinx ® to access the backup PLC (see Appendix 1 - Allen-Bradley PLC Communication Paths).

main_ind(required if redundant pair) - PLC address for main PLC indicator signal. Set to 1 if main, 0 if standby.

The following columns are optional:-

Heartbeat (optional) - PLC address of the heartbeat signal.

hb_rate (optional) - frequency in seconds for checking heartbeat .

Example:

```
insert into
abDevice_Table (devicename, networkname, deviceaddress, redundant, b_addr_i
nfo, main_ind)
values ('AB1', 'TEST1', 'AB:LOCAL,1', true, 'AB:LOCAL,2', 'N31:0/0');
```

1.5.4 dvi_schedule

There will be a table entry for each poll schedule. A single schedule is not restricted to a single device (PLC) but can be implemented on one or more devices and networks. The following columns must contain valid data:

scheduleid - table key

maxinterval - polling interval -This column is a datetime type ie the value is treated as if offset from midnight on 01-JAN-0001. Thus 10 seconds is equivalent to 00:00:10 01-JAN-0001.

Example:

```
insert into dvi_schedule (scheduleid, maxinterval)
values (1, SECONDS(10));
```

1.5.5 ABPollList_Table

There will be a table entry for each range of PLC memory to be polled on each device. The table row will be matched with a dvi_schedule row to determine the polling interval. The following columns must contain valid data:

name - table key.

devicename - ABDevice_Table key.

regionname - dvi_region key. Must be 'AB_file'.

datumaddress - Allen-Bradley PLC file name eg N7

scheduleid - dvi_schedule key.

start_address - start address in PLC file (defined in datumaddress) for polling.

end_address - end address in PLC file (defined in datumaddress) for polling

Example:

```
insert into ABpolllist_table
(name, devicename, regionname, datumaddress, scheduleid, start_address, end_
address)

values ('N7', 'AB1', 'ab_file', 'N7', 1, 0, 3);
```

1.5.6 ABIntegerAnalog_Table

The entries in this table represent 16 bit word data items in the following PLC memory files:

- Ascii Data (A)
- BCD Data (D)
- Integer (N)
- Status (S)

The following columns must contain valid data:

name - table key.

devicename - ABDevice_Table key.

regionname - dvi_region key. Must be 'AB_iana'.

ab_file - PLC memory address

Example:

```
insert into ABIntegerAnalog_Table
(name, regionname, devicename, ab_file, LowLimit, HighLimit)

values ('PLCN2', 'ab_iana', 'AB1', 'N7:1', 0, 4095);
```

1.5.7 ABRealAnalog_Table

The entries in this table represent data items in the following PLC memory files:

- Floating-point (F)
- Integer (N)

In this table, Integer(N) data items are used to store real numbers. The conversion from PLC to OE Database is calculated as follows:

$$y = mx + c$$

The conversion from Database to PLC is calculated as follows:

$$x = (y-c)/m$$

Where $y(\text{real})$ = output to database, $x(\text{integer})$ = input from PLC,
 $m(\text{real})$ =span, $c(\text{real})$ = zero

The following columns must contain valid data:

name - table key.

devicename - abDevice_Table key.

regionname - dvi_region key. Must be 'ab_rana'.

ab_file - PLC memory address

span - (required for integer/real conversion)

zero - (required for integer/real conversion)

Example:

```
insert into ABrealanalog_table (name,regionname,devicename,ab_file)
values ('PLCF1','ab_rana','ab1','F8:0');
```

1.5.8 ABDigital_Table

The entries in this table represent data items in the following PLC memory files:

- Bit (B)
- Bit (N)
- Output(O)
- Input(I)

In this table Integer(N) data items are used to store digital data.

The following columns must contain valid data:

name - table key.

devicename - abDevice_Table key.

regionname - dvi_region key. Must be 'ab_log'.

ab_file - PLC memory address

Example:

```
insert into abDigital_Table (name,regionname,devicename,ab_file)
values ('PLCB1','ab_log','ab1','B3:0/0');
```

1.5.9 ABComplex_Table

The entries in this table represent Structured Data items in the following PLC memory files:

- Timer (T) - 6 bytes
- Counter (C) - 6 bytes
- Control(R) - 6 bytes
- SFC Status(SC) - 6 bytes
- Blk-transfer Control (BT) - 12 bytes
- Token Data (TD) - 4 bytes
- ASCII String (ST) - 84 bytes
- PID Control (PD) - 164 bytes
- Message Control (MG) - 112 bytes

The data items are transferred as a block of binary data between the OE Database and the RDI. The size of the data block is indicated above for each complex file type. For example, when polling a Timer data item, the RDI will read the PLC memory and send 6 bytes of data to the OE Database. When the OE Database updates a data item it will send 6 bytes of data to the RDI.

The following columns must contain valid data:

name - table key.

devicename - ABDevice_Table key.

regionname - dvi_region key. Must be 'ab_complex'.

ab_file - PLC memory address

Example:

```
insert into ABComplex_Table (name,regionname,devicename,ab_file)
values ('PLCT1','ab_complex','ab1','T4:0');
```


1.6 Running the Allen-Bradley RDI

The program `abrdi` is a windows application. Note that the TCP/IP service name 'abrdi' to which the program binds is hard coded in the program.

To run the Allen-Bradley RDI from the Command line, type the name of the executable file, followed by a space, then by the name of the designated Allen-Bradley TCP/IP service: -

```
C:\> abrdi abrdi
```

When running the `abrdi` as part of an OpenEnterprise Session the *CommandLine* value for the `abrdi` task will also need to contain the text 'abrdi' (without the quotes).

Error and information messages will appear in the `abrdi.exe` program window. The program holds a maximum of 10,000 messages in dynamic memory. These messages will also be written into a disk file `abrdi.log`. This file will be overwritten each time the program is run. Messages are also written into the 'Allen-Bradley Driver' entry in the OE Database `journaltask_table`.

Control Language errors and messages will appear in the RTRTB window.

While the RDI is running, it can accommodate some OE Database Table changes. Any other changes will be ignored. Below is a list of what can and cannot be changed in runtime.

- The single **abDriver_Table** entry should **not** be deleted.
- **abNetwork_table** entries **can** be added and deleted.
- **abDevice_Table** entries **can** be added and deleted. The heartbeat and `hb_rate` columns can be modified. The redundant, `main_ind` and `b_addr_info` columns can be modified once only to turn the device into a redundant pair.
- **Signal Datums** **can** be added or deleted. Note that if a new datum is outside the poll range defined in the associated `abPollList_Table` entry it will not be polled.
- **abPollList_Table** **can** be added or deleted.
- **dvi_schedule** entries **cannot** be added, deleted or modified.

The OpenEnterprise Allen-Bradley Configuration Tool simplifies the process of changing Allen-Bradley data within the OE Database. The user should consult the Allen-Bradley Configuration Tool Reference Guide for further details.

2 Index

A

<i>Allen-Bradley RDI</i>	
<i>Running</i>	8
<i>Allen-Bradley RDI</i>	8

O

<i>OE Database Configuration#ABNetworkTable</i>	2
<i>Overview</i>	1

R

<i>RSLink Driver Configuration</i>	2
<i>Running</i>	
<i>Allen-Bradley RDI</i>	8
<i>Running</i>	8

S

<i>System Requirements</i>	1
----------------------------------	---

Reference Guide

D5092

11-Dec-2007

Allen-Bradley RDI

NOTICE

Remote Automation Solutions ("RAS"), division of Emerson Process Management shall not be liable for technical or editorial errors in this manual or omissions from this manual. RAS MAKES NO WARRANTIES, EXPRESSED OR IMPLIED, INCLUDING THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE WITH RESPECT TO THIS MANUAL AND, IN NO EVENT SHALL RAS BE LIABLE FOR ANY INCIDENTAL, PUNITIVE, SPECIAL OR CONSEQUENTIAL DAMAGES INCLUDING, BUT NOT LIMITED TO, LOSS OF PRODUCTION, LOSS OF PROFITS, LOSS OF REVENUE OR USE AND COSTS INCURRED INCLUDING WITHOUT LIMITATION FOR CAPITAL, FUEL AND POWER, AND CLAIMS OF THIRD PARTIES.

Bristol, Inc., Bristol Babcock Ltd, Bristol Canada, BBI SA de CV and the Flow Computer Division are wholly owned subsidiaries of Emerson Electric Co. doing business as Remote Automation Solutions ("RAS"), a division of Emerson Process Management. ROC, FloBoss, ROCLINK, Bristol, Bristol Babcock, ControlWave, TeleFlow and Helicoid are trademarks of RAS. AMS, PlantWeb and the PlantWeb logo are marks of Emerson Electric Co. The Emerson logo is a trademark and service mark of the Emerson Electric Co. All other trademarks are property of their respective owners.

The contents of this publication are presented for informational purposes only. While every effort has been made to ensure informational accuracy, they are not to be construed as warranties or guarantees, express or implied, regarding the products or services described herein or their use or applicability. RAS reserves the right to modify or improve the designs or specifications of such products at any time without notice. All sales are governed by RAS' terms and conditions which are available upon request.

RAS does not assume responsibility for the selection, use or maintenance of any product. Responsibility for proper selection, use and maintenance of any RAS product remains solely with the purchaser and end-user.

Engineered and supported by:

Remote Automation Solutions,

Bristol Babcock Ltd. Blackpole Road, Worcester, WR3 8YB, UK

Reg office: 2nd Floor, Accurist House, 44 Baker Street, London, W1U 7AL

Registered in England and Wales, Reg No. 2580226

VAT Reg No. GB 880 9839 67

Emerson Process Management
Remote Automation Solutions
1100 Buckingham St
Watertown, CT 06795
T 1 (860) 945 2200
F 1 (860) 945 2278
www.EmersonProcess.com/Remote
binfo@EmersonProcess.com

Emerson Process Management
Remote Automation Solutions
Blackpole Road
Worcester, WR3 8YB
T 44 (0) 1905 856848
F 44 (0) 1905 856930
www.EmersonProcess.com/Remote
oedsupport@EmersonProcess.com

