
CYLON

UnitronUC32 OPC

Setup Guide

UNITRONUC32.

Building Management System

© Cylon Controls Ltd. 2006. All Rights Reserved.

UnitronUC32 OPC setup guide (MAN0082 rev 2.13)

Information in this document is subject to change without notice. The software described in this document is furnished under a license agreement or nondisclosure agreement. The software may be used only in accordance with the terms of those agreements. No part of this publication may be reproduced, stored in a retrieval system or transmitted in any form or any means electronic or mechanical, including photocopying and recording for any purpose other than the purchaser's personal use without the written permission of Cylon Controls Ltd.

Cylon Controls Ltd.
Clonshaugh Business and Technology Park
Clonshaugh
Dublin 17
Ireland

TRADEMARKS

All trademarks and tradenames used in this document are acknowledged to be the copyright of their respective holders.

CYLON
www.cylon.com

Contents

Introduction	1
1 What is OPC?	3
1.1 What is an OPC Server?	3
1.2 What is an OPC Driver?	3
1.3 What is an OPC Client?	4
1.4 What is SCADA?	4
1.5 Why would SCADA be used with UnitronUC32?	4
2 Conceptual differences between UnitronUC32 and OPC	5
2.1 Structure of UnitronUC32 data	5
2.2 Structure of OPC data (compared to UnitronUC32 data)	5
Setting up Unitron OPC Server	7
3 Install the UnitronUC32 OPC Server	9
3.1 Install Microsoft .net Framework 2	9
3.2 Run the OPC setup program (OPC Server Setup.exe)	9
4 Verify the OPC server installation	11
4.1 Verify UC32 OPC Point Communication	11
4.2 Verify UC32 OPC Time Schedule Communication	14
4.3 Verify UCU OPC Point Communication	15
Exporting Data to OPC	17
5 About Automatic OPC configuration	19
6 Configure the UnitronUC32 site	20
6.1 Identifying points for automatic export to the SCADA supervisor	20
6.2 Efficiency considerations	20
7 Export the UnitronUC32 database	21
7.1 Preparing for OPC export ('Export to iFix')	21
7.2 Exporting the .gdb and .csv files	24
8 Import the Exported file into OPC Server	25
8.1 Import The UC32 .csv file	25
8.2 Import The UCU .csv file	25
9 Import the Exported file into the SCADA Package	27
10 Optimise the OPC configuration	27
11 Set OPC Server running	28
12 Verify communications with OPC Client	28

Appendices.....	29
13 Appendix - What SCADA packages have been used with the UnitronUC32 OPC server?.....	31
14 Appendix - Requirements for a UnitronUC32 OPC system.....	31
15 Appendix - OPC Capabilities of Unitron Devices	31

Introduction

This page is intentionally blank

1 What is OPC?

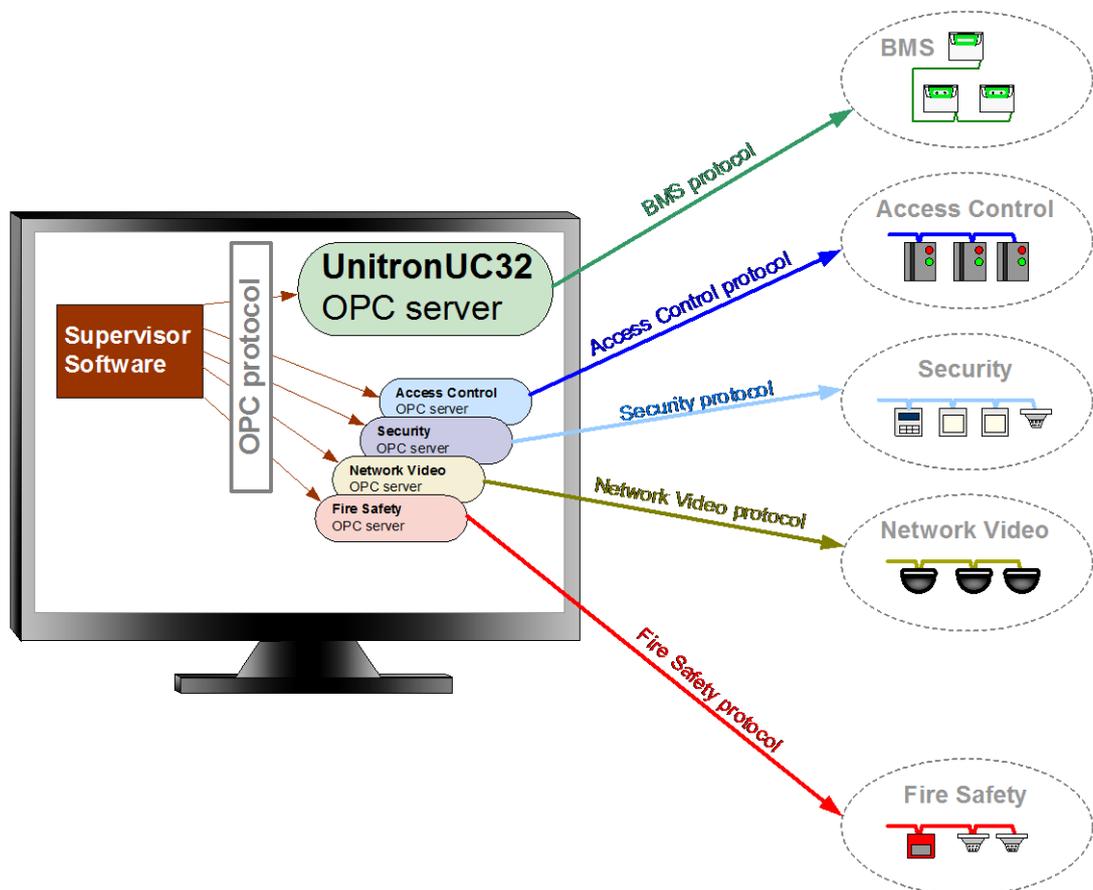
OPC stands for "OLE for Process Control", and provides "Open Connectivity" for process automation and control, allowing data transfer between incompatible proprietary systems (e.g. BMS, Fire systems, security system, Process Control systems etc.).

In the OPC system each proprietary system must have an OPC 'Server', which translates between the proprietary protocol and the OPC protocol. On the OPC side of this server, communication is possible with any OPC-compliant system.

In terms of Cylon's UnitronUC32 BMS, the Unitron OPC Servers are principally used to integrate the BMS with SCADA systems (such as iFix) that are already installed in the building that the BMS will control.

1.1 What is an OPC Server?

An "OPC Server" exposes data from a proprietary protocol system to a standard OPC protocol system. This allows any OPC Client to read and change data on the proprietary system.



1.2 What is an OPC Driver?

"OPC Driver" is a synonym for "OPC Server". Because OPC Servers provide communications between OPC and proprietary protocols used by hardware systems, they have been compared to the hardware "drivers" used by Microsoft Windows - for example printer drivers. Using the term "OPC Driver" can help convey what the OPC Server does, but an OPC Driver is not a separate or different piece of software.

1.3 What is an OPC Client?

An "OPC Client" is any device (usually a piece of software such as a SCADA Supervisor) to read and change data that has been made available to the OPC system by an OPC Server.

1.4 What is SCADA?

SCADA is an acronym for "Supervisory Control and Data Acquisition", and can refer to any system for gathering and analysing real time data. A SCADA system is a "supervisory" rather than a "control" system. SCADA systems are used to monitor manufacturing processes, building environments, telecommunications systems etc.

1.5 Why would SCADA be used with UnitronUC32?

There are several reasons why customers might choose to use a SCADA supervisor to supervise their UnitronUC32 site rather than the standard Unitron Command Centre software:

1. The SCADA system provides a sophisticated and adaptable industry-standard interface for UnitronUC32.
2. If the client site where UnitronUC32 is installed is using SCADA for other purposes (such as manufacturing process control) the client may wish to use the same interface to supervise the building environment.
3. Some SCADA systems (e.g. iFix) provide full audit trail and electronic signature facilities, making UnitronUC32 suitable for controlling pharmaceutical manufacturing environments.

2 Conceptual differences between UnitronUC32 and OPC

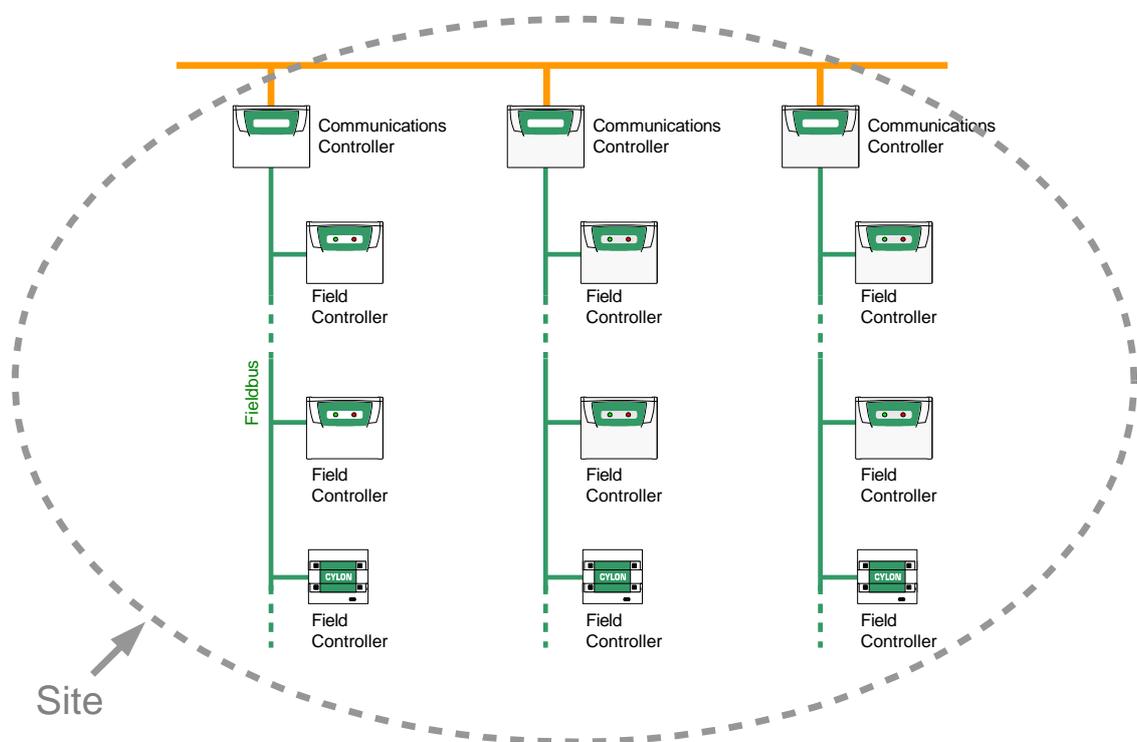
2.1 Structure of UnitronUC32 data

In the UnitronUC32 system, point values originate in **Field Controllers**, which are connected to sensors, switches and equipment throughout the building. Point values are held in the memory of a Field Controller, as part of the controller's 'strategy'.

Field Controllers are networked together to form a **Fieldbus**, which is controlled by a **Communications Controller**.

Communications Controllers, each with several Field Controllers on its Fieldbus, are networked together to form a **Site**.

A **Site** is the primary element on the UnitronUC32 System, and it contains all other elements.



2.2 Structure of OPC data (compared to UnitronUC32 data)

In OPC, the primary element is called a "**Node**".

The **UnitronUC32 OPC Server** acts as a single OPC node, containing any relevant information about all of the sites in the UnitronUC32 system.

The **Node** corresponds to the UnitronUC32 system.

An OPC **Node** connects to OPC-compatible devices through one or more "**Channels**". A Channel can represent a physical connection, such as a serial link (e.g. Modbus RS232) or Ethernet.

In the UnitronUC32 system, each **Channel** represents the connection between the **UnitronUC32 OPC Server** and one UnitronUC32 **Site**.

Each **Channel** corresponds to a **Site**.

An OPC **Channel** contains "**Devices**".

In the UnitronUC32 system, each **Device** represents a UC32.netK Communications Controller.

Finally, the actual Point values are held in "**Datablocks**" within the **Devices**. Each **Datablock** holds a number of point values.

In the UnitronUC32 system, each **Datablock** in a **Device** can contain a number of point values from a single Field Controller on the fieldbus of the UC32.netK Communications Controller that is acting as the **Device**

Note:

In the UnitronUC32 system, the maximum number of points that can be held in a **Datablock** depends on the type of Field controller in which the point values originate:

In UC32 Field Controllers a **Datablock** can hold up to

- 40 contiguous Analog points, or
- 100 contiguous Digital points,

In UCU Field Controllers a **Datablock** can hold up to

- 3 contiguous Analog points, or
- 96 contiguous Digital points.

Setting up Unitron OPC Server

This page is intentionally blank

3 Install the UnitronUC32 OPC Server

If you intend to use the optional Time Schedule interface that is included with the OPC Server, you must first ensure that the Microsoft .net Framework version2 is installed on the Server PC.

3.1 Install Microsoft .net Framework 2

The Microsoft .net framework is required to run the Time Schedule interface. If you wish to use the Time Schedule interface, install the .net framework by running the file

```
dotnet framework 2.exe
```

This program may be downloaded from www.microsoft.com, or run from the OPC Server CD.

3.2 Run the OPC setup program (OPC Server Setup.exe)

This program :

- Installs the UnitronUC32 OPC Server
- Installs the UCU OPC Server
- Adds export functionality to the UnitronUC32 database interface (adds "Export to iFix" option to the database interface menu.)
- Installs the UC32 PowerTool, which is the interface for the UnitronUC32 OPC Server.
- Installs the UCU PowerTool, which is the interface for the UCU OPC Server.
- Installs the UnitronUC32 OPC Time Schedule application.

⇒ *Run the setup program (OPC Server Setup.exe) from the UnitronUC32 OPC CD*

⇒ *When prompted for the location of the UC32 database, enter the directory where UnitronUC32 software is installed. This is usually C:\UnitronUC32*

- This is required for the 'Export to OPC' functionality that will be added to the database interface.

The actual OPC server setup starts.

⇒ *If you accept the Licence agreement, click next*

⇒ *When prompted for the Installation Directory, enter the location of the software with which you intend to use the Unitron OPC Server - e.g. SCADA supervisor (such as iFix).*

⇒ *When prompted for the "Node Type", select "Server"*

⇒ *When prompted for the Program Group enter the name of the program group used by the software with which you intend to use the Unitron OPC Server.*

⇒ *Click the **Finish** button*

You will now be prompted for product activation. You will be asked to enter the activation key printed on the CD cover.

Note: If the activation code is not entered, the software will run in "Evaluation Mode" for a limited period of time, and will prompt for an activation code every time it is started.

The OPC Time Schedule install runs simultaneously with the Activation procedure, and will prompt for automatic startup. It is recommended that you enable Automatic Startup, because this will speed up starting the OPC Time Schedule interface.

4 Verify the OPC server installation

It is advisable, when installing the OPC server on a site for the first time, to verify that points from the Unitron Site are being properly made available over OPC.

To do this you can use the PowerTool to define a **Datablock** of points and to view their values as described below. In order to define the **Datablock**, an OPC **Channel** and at least one OPC **Device** must also be defined, because in the OPC system a **Datablock** is part of a **Device**, which is connected to the Server PC by the **Channel**.

4.1 Verify UC32 OPC Point Communication

Start the UC32 PowerTool

⇒ Start the UC32 PowerTool , by selecting it from the program group created during install (in the example given on page 9, Start > UnitronOPC > UC32 PowerTool) .

Note: The UC32 PowerTool will configure the OPC Server to read points for UC32-type Field Controllers. If points contained within UCU-type Field Controllers are to be read, the UCU PowerTool must be used.

Create a Channel

The OPC **Channel** represents the connection between the Unitron site and the PC running the OPC Server. Only one **Channel** (which is usually a TCP/IP connection) should be to access all of the controllers and points on the Unitron Site (so that an OPC **Channel** represents a UnitronUC32 **Site**).

To create a **Channel** in the UC32 PowerTool :

⇒ Select from the menu

⇒ Enter a name for the **Channel**, e.g. "WestOffice". This should be the same as the name of the Site Directory in the UNitronUC32 system.

⇒ Enter a description of the **Channel**, e.g. "TCP/IP connection to West Office site"

⇒ Enable the **Channel** by ticking the box

Create a Device

In Unitron OPC, a **Device** represents a Unitron Fieldbus, i.e. a Comms Controller with a set of Field Controllers attached to it. At least one Device must be defined if Unitron point values are to be viewed through OPC.

To set up a **Device** in the UC32 PowerTool :

- ⇒ Select **Add Device** from the **Edit** menu
- ⇒ Enter a descriptive name for the **Device**, e.g. "Reception"
- ⇒ Enter a description of the **Device**, e.g. "Fancoil Units and Lights in the Reception Area"
- ⇒ Set the **Primary IP Address** to the IP address of the Comms Controller to which the Field controller containing the points you want to access is connected
- ⇒ Set the **Port** to 4950, unless the port has been specifically altered on the Comms Controller.
- ⇒ Set the **Primary Comms Controller** to the Controller address of the Comms Controller specified by IP address above.
- ⇒ The **Reply Timeout**, **Retries**, and **Delay Time** parameters can be left with their default values.
- ⇒ Enable the **Device** by ticking the **Enable** box

Create a Datablock

In Unitron OPC, points in a Unitron Controller are grouped together into an OPC **Datablock**, which can be read by any OPC Client.

To set up a **Datablock** in the UC32 PowerTool :

- ⇒ Select **Add Datablock** from the **Edit** menu
- ⇒ Enter a name for the **Datablock**, e.g. "FanCoil_A"
- ⇒ Enter a description of the **Datablock**, e.g. "Fancoil Unit beside Reception Desk"
- ⇒ In the **UC32 Controller** box, enter the Fieldbus address of the Field Controller that contains the points you wish to access.
- ⇒ For the **Data Type** select
 - "Float" if the required points are Analog Points,
 - "Digital" if the required points are Digital Points,
 - "Time Schedule" if you are specifying Time Schedule blocks rather than points

- ⇒ In the **I/O Address Setup** section, specify the first and last points in the range of points that will make up the **Datablock**.
- ⇒ If required, you can preserve a **Datablock's** most recently read value by checking the **Latch Data** box.
 - Should a communication failure occur, Data links to unlatched Datablocks display a series of question marks (?????) while Data links to latched Datablocks display the last data polled. This feature is very useful when you need to create reports about your process and require data at all times. You can also use this option to preserve the last values on the screen after a communication failure occurs.
- ⇒ If required, you can prevent output from the OPC Server to this **Datablock** by checking the **Disable outputs** box.
 - You may want to disable outputs:
 - To prevent control outputs that may be hazardous to someone performing maintenance or repair on the process control hardware.
 - To prevent erroneous writes to the Datablock.
 - To isolate a Datablock for debugging.
- ⇒ In the **Polling setup** section, **Primary Rate**, **Phase**, and **Access Time** can be left at their default values.
- ⇒ These parameters can be adjusted if required at a later stage to increase efficiency.
- ⇒ Enable the **Datablock** by ticking the **Enable** box

Start the Unitron OPC server

Before the points in the **Datablock** can be viewed, the OPC Server must be running. To make sure the server is running:

- ⇒ In the **PowerTool**, select **Start** from the **Display Mode** menu.

Check the Communications between Unitron Site and the OPC Server

When a **Channel**, **Device** and **Datablock** have been defined, and the OPC Server is running, you can monitor OPC communications and view point values as follows:

- ⇒ In the **PowerTool**, open the **Statistics** window by selecting **Stats Mode** from the **Display Mode** menu.
- ⇒ In the navigation list at the left hand side of the dialog, select the **Datablock** that was set up earlier.
- ⇒ Check that the "Transmits" statistic starts to increment
- ⇒ Click the **Data Monitor** button to view the value of the points in the **Datablock**.
- ⇒ Check the "Quality" parameter to verify communications. The quality should be reported as "Good".

4.2 Verify UC32 OPC Time Schedule Communication

Set up a Time Schedule Datablock

Next, to verify that Unitron Time Schedules are accessible to the OPC system, a **Datablock** must be set up to represent a Time Schedule in the UC32 PowerTool as follows:

- ⇒ Select **Add Datablock** from the **Edit** menu
- ⇒ Enter a descriptive name for the **Datablock**, e.g. "Lighting_Schedule_Reception"
- ⇒ Enter a description of the **Datablock**, e.g. "Main Time Schedule for Lights in reception area"
- ⇒ In the **UC32 Controller** box, enter the Fieldbus address of the Field Controller that contains the Time Schedule you wish to access.
- ⇒ For the **Data Type** select Time Schedule
- ⇒ In the **I/O Address Setup** section, the Starting Address and Ending Address should be the first block and last block of the Time Schedule respectively.
 - The first and last blocks can be read from the Unitron Database interface, CCDBIF. Start the Database Interface, select the 'Comms Controller Time Schedule' radio button on the right hand side, and then select the Comms Controller that contains the required Time Schedule from the drop-down list on the left-hand side. All of the Time Schedules in that Comms Controller will be listed at the bottom of the Database Interface dialog, along with their Start and end blocks.
- ⇒ If required, you can preserve a **Datablock's** most recently read value by checking the **Latch Data** box.
 - Should a communication failure occur, Data links to unlatched Datablocks display a series of question marks (?????) while Data links to latched Datablocks display the last data polled. This feature is very useful when you need to create reports about your process and require data at all times. You can also use this option to preserve the last values on the screen after a communication failure occurs.
- ⇒ In the **Polling setup** section, **Primary Rate**, **Phase**, and **Access Time** can be left at their default values.
- ⇒ These parameters can be adjusted if required at a later stage to increase efficiency.
- ⇒ Enable the **Datablock** by ticking the **Enable** box

Start the Unitron OPC server

Before the Time Schedule data can be viewed, the OPC Server must be running. To make sure the server is running:

- ⇒ In the PowerTool, select **Start** from the **Display Mode** menu.

Run the Unitron OPC Time Schedule Interface

In order to read and change Unitron Time Schedules through OPC, the **Unitron OPC Time Schedule Interface** must be used. This is a proprietary program, started by running the command `TimeScheduleUI.exe` with the following parameters:

```
TimeScheduleUI.exe /S:[ChannelName] /C:[DeviceName] /TS:[dataBlockName]
```

where `[ChannelName]` , `[DeviceName]` , `[dataBlockName]` are the values set up in the previous sections.

It can be easier to start the **Unitron OPC Time Schedule Interface** from a batch file, to avoid entering all of the parameters each time the Interface is started. A sample batch file is provided with the UntronOPC setup, and can be found in

```
C:\Program Files\UnitronOPCServer\StartSchedule.bat
```

(assuming 'UnitronOPCServer' was given as the install directory during the setup procedure)

Note: To use Time Schedules from the OPC Client, you must insert links to the file `TimeScheduleUI.exe`, with different parameters (as detailed above) embedded in each link for each Time Schedule.

4.3 Verify UCU OPC Point Communication

If you have UCUs on your site, you should also verify the UCU OPC Server in the same manner as described in section 4.1 - *Verify UC32 OPC Point Communication* (on page 11), except using the UCU PowerTool instead of the UC32 PowerTool .

This page is intentionally blank

Exporting Data to OPC

This page is intentionally blank

5 About Automatic OPC configuration

The previous section of this document described how to manually create **Channels, Devices,** and **Datablocks** in the OPC system to represent Points and Time Schedules in the UnitronUC32 system.

However, this process can be time-consuming when a full site is to be integrated with the OPC system, so Cylon provides the facility for automatically exporting UnitronUC32 objects into OPC **Channels, Devices,** and **Datablocks**.

This section describes how to set up the UnitronUC32 site so that automatic export is possible, and then how to export the required configuration files using the Unitron Database Interface.

6 Configure the UnitronUC32 site

The BMS site should be engineered in the UEC in the usual way, and a strategy created for every Field Controller on the site.

However, in order to make it easier to use the Site with a SCADA supervisor, the points that will be used by the SCADA supervisor should be identified in the Controller Strategy. They can then be automatically exported to a file from the UnitronUC32 database, which can then be imported to the SCADA supervisor and the OPC Server.

6.1 Identifying points for automatic export to the SCADA supervisor

Note: If you want to export **all** of the points in the UnitronUC32 Site, it is not necessary to identify each one explicitly, and the procedure described in this section is not required.

If you do not want to export all points in the UnitronUC32 Site to OPC, you can identify the points that are to be exported by giving each of them a name ending with "_i". The name can be any descriptive text, but the last two characters must be "_i".

For example:

boiler_status_i

fanspeed_i

Room Temperature _i

When all of the required points have been named in this manner, download all strategies to their relevant controllers on the BMS site.

6.2 Efficiency considerations

OPC **Datablocks** (see *Conceptual differences between UnitronUC32 and OPC* on page 5) always use contiguous point numbers, because they are defined by a starting point number and a length. When the **Datablock** is requested, each of the points covered by the **Datablock** are polled over the UC32 Fieldbus.

As a result, to maximise efficiency all points marked for export should be assigned consecutive point numbers to minimise

- a) the number of actual point values polled
- b) the number of **Datablocks** used

a) Reducing the number of **Datablocks** used

In addition, if analog points 1, 45, and 90 in a UC32 Field Controller were to be exported, 3 separate **Datablocks** would be required, as the maximum length of an analog **Datablock** is 40 points. Renumbering those points to 903, 904 and 905 would mean that all of the values could be polled by a single **Datablock**.

b) Reducing the number of points polled over the UC32 Fieldbus

It is suggested that any points that are to be exported should be renumbered to contiguous numbers above 900 (or any other high number) using the Unitron Engineering Centre's 'Change Point' feature (right-click on the connecting line between two modules and select Change Point).

7 Export the UnitronUC32 database

The database of points in the UnitronUC32 site must be made available to both the SCADA supervisor system and the UnitronUC32 OPC Server.

Cylon currently supports automatic export of the points database for iFix SCADA supervisor systems. The **Database Interface** module (part of the Unitron Engineering Centre) has a facility to export the Unitron Database into particular formats specifically designed to be easy to import into iFix. (.gdb & .csv files).

7.1 Preparing for OPC export ('Export to iFix')

Some configuration is required **before** the UnitronUC32 Database Interface application can convert the UnitronUC32 database to iFix compatible files. This is because the nature and structure of OPC databases is very different from the structure of the UnitronUC32 database (see *Conceptual differences between UnitronUC32 and OPC* on page 5).

How UnitronUC32 data is mapped to the OPC structure

In order to export the UnitronUC32 database, the following steps must be carried out within the Unitron Database Interface:

1. An OPC **Node** must be defined, along with the **Channel** it will use to connect to the Supervisory PC
2. OPC **Devices** must be created, and mapped to UC32.netK Communications Controllers from which Point values are required.
3. For each **Device** one or more **Datablocks** must be defined, stating the Field Controller containing the required Point Values, and the Addresses of the relevant points within that Field Controller.

Defining an OPC Node (Step 1)

- ⇒ *Open the UnitronUC32 Database Interface.*
- ⇒ *From the File menu, select Export to iFix > Export Current Site*
– This opens the first step of the Export Wizard, which defines the **Node** and Channel
- ⇒ *In the **Node** name box, enter a name that describes the current UnitronUC32 Site (e.g. "BMSPC") (maximum of 12 characters).*
- ⇒ *Enter a name for the Ethernet **Channel** that will be used by UnitronUC32 OPC (this should be the same as the name of the Site directory in the UnitronUC32 system).*
- ⇒ *Enter a description for the Ethernet **Channel** that will be used by UnitronUC32 OPC.*
- ⇒ *Tick the **Assigned** box.*
- ⇒ *Click on the button to move to the **Device** definition step (step 2).*

Defining an OPC Device (Step 2)

- ⇒ *In Step 2 of the Export wizard, enter a name to be used for the first OPC **Device** (e.g. "UC32net1") (maximum of 12 characters).*

- ⇒ Enter a description for the **Device** (optional).
- ⇒ Select the Comms Controller to which this **Device** will map, from the drop-down list of all of the Comms Controllers on the current Site.
- ⇒ Enter the IP address of the selected Comms Controller (e.g. 192.168.7.32), because OPC will talk directly to this address over TCP/IP.
- ⇒ Enter the IP port (usually 4950) used by the Comms Controller to receive Ethernet communications.
- ⇒ If you want the Comms Controller time schedules associated with this Device/Comms Controller to be made available to the Unitron OPC Time Schedule application, check the 'Export Time Schedules' box.
- ⇒ Click the **Assigned** button, to indicate that this **Device** is to be exported.

Note: It is possible to define **Devices** without exporting them, or to decide at later time not to export a **Device** that had previously been 'assigned'.

Note: The timeout parameters for each Comms Controller are set to default values. These can be later edited in UC32 PowerTool if required.

When **Datablocks** have been defined - or if no **Datablocks** are to be defined for this **Device** - you will be able to define another **Device** by entering a new name in the **Device** Name field, (or by selecting a predefined name from the drop-down list) and repeating step 2.

To define **Datablocks** for this **Device** (step 3) click the **Next** button.

Configuring **Datablocks** (step 3)

- ⇒ Clicking **Next** on step 2 of the Export Wizard opens the **Datablock** Configuration page (step 3)
- ⇒ Enter a name for the new **Datablock**(maximum of 12 characters).
- ⇒ Select the Field Controller ("UCxx") that contains the Points whose values will make up this **Datablock**, either by selecting it from the drop-down list or by clicking the **Next UCxx** button until you reach the required Field Controller.

Note: The selected Field Controller must be :

- UC32 type (e.g. UC32.24, UC32.24K, UC32.16 etc.) with firmware version 6.06 or later, or
- UCU Controllers with firmware 5.74 or later.

Unitron2000 field controllers cannot be polled.

- ⇒ Enter a description for the new **Datablock**

Note: There is a 'Select all UCxxs' option which, if selected, will cause **Datablocks** to be created for all points that match the subsequent settings (either manual or "Export points ending in '_i'") from ALL Field Controllers on the Fieldbus of the current '**Device**'.

Warning:

If there are Field Controllers anywhere on your site with addresses above 16, do NOT use the 'Select all UCxxs' option, as the resulting .csv file will not import into UC32 PowerTool.

To avoid this problem, define **Datablocks** individually for each controller.

- ⇒ If you have configured the UnitronUC32 site as described in section 5, check the **Export points ending in '_i'** box. This will automatically create all required **Datablocks** for the selected data type.
- ⇒ Specify the **data type** (Analog or Digital) for the points that will be polled as part of this **Datablock** in the combo box on the bottom right-hand side of the dialog box.
- ⇒ If you have NOT set up points ending in '_i', specify a **start address** and **end address** for the range of points of the specified type that will form this **Datablock**.

Note: You can specify any start and end address, but if there are more than the allowed number of points in the range (40 Analog/100 Digital for UC32 Field Controllers, 3 Analog/96 Digital for UCU Field Controllers) multiple **Datablocks** will be created. All of the **Datablocks** created in this way will use the same name, with an index number appended.

- ⇒ Click the 'Assigned' checkbox, to mark the current **Datablock** for export.
- ⇒ Repeat the above steps for the other **data type** (Analog or Digital) if necessary.
- ⇒ To create another **Datablock** for the current **Device**, enter a new **Datablock** name, and repeat the above steps.
- ⇒ To select another **Device**, click [Back](#).

7.2 Exporting the .gdb and .csv files

⇒ When all of the required points have been added to **Datablocks** in the Export Wizard, click on the **Done** button.

This will create the required files for import into iFix and the UnitronUC32 OPC Server, i.e:

- a UC32 OPC .csv file for the UnitronUC32 OPC Server,
- a UCU OPC .csv file for the UCU OPC Server,
- a .gdb file for the iFix database manager,

Naming scheme for Exported file

The files will be given names based on:

- the type of device the file is intended for - UC32 or UCU (iFix does not need a prefix as iFix files are .gdb format. All other files are .csv)
- the BMS segment covered by the file - Site, Comms Controller or Field Controller
- the OPC **Node** defined in the first step of the Export to iFix wizard e.g. BMSPC
- the Comms Controller address, if appropriate
- the Field Controller address, if appropriate

General pattern:

[UC32|UCU]_[Site|CommsContr|FieldContr]_nodename[_xxx][_yyy].[csv|gdb]

where xxx is the Comms Controller address, and

yyy is the Field Controller address

Examples:

UC32_CommsContr_BMSPC_005.csv

UCU_FieldContr_BMSPC_005_003.csv

Site_BMSPC.gdb

The files will be saved in the iFix sub folder inside the relevant Site folder in the UnitronUC32 software folder on the Supervisory PC (usually C:\UnitronUC32).

Example: C:\UnitronUC32\[sitename]\DBexports\iFix

8 Import the Exported file into OPC Server

To import the Database Interface Export files into the OPC Server, you must first 'open' them in the relevant PowerTool (UC32 PowerTool or UCU PowerTool) then Save them as either .cct or .ucu files, depending on whether they are intended to UC32 controllers or UCU controllers respectively.

Note: there may not always be 2 .csv files exported. This will only happen if points from both UC32 and UCU controllers are required.

8.1 Import The UC32 .csv file

- ⇒ *Open the UC32 PowerTool.*
- ⇒ *Select Open from File menu*
- ⇒ *In the 'Open' dialog, change the file type to "csv"*
- ⇒ *Navigate to the folder of the current site and select the UC32 .csv file exported in section 7 (e.g. C:\UnitronUC32\[sitename]\DBexports\iFix)*
- ⇒ *Click on .*
- ⇒ *Go to options*
- ⇒ *Select Setup*
- ⇒ *Click on Default path tab*
- ⇒ *Set the Default Configuration filename to [nodeName.cct] (e.g. BMSPC.cct)*
- ⇒ *Change the Default path to the directory within the site where the supervisor expects to find Configuration files - e.g. C:\Dynamics\Cylon\pdb*
- ⇒ *Select select Save As... from the File menu*
- ⇒ *Save the .cct file to this location*

Note: This UC32 file must be saved with .cct extension.

8.2 Import The UCU .csv file

- ⇒ *Open the UCU PowerTool.*
- ⇒ *Select Open from File menu*
- ⇒ *In the 'Open' dialog, change the file type to "csv"*
- ⇒ *Navigate to the folder of the current site and select the UC32 .csv file exported in section 7*
- ⇒ *Click on .*
- ⇒ *Go to options*
- ⇒ *Select Setup*
- ⇒ *Click on Default path tab*

UnitronUC32 OPC setup guide

- ⇒ *Set the Default Configuration filename to [nodeName.ucu] (e.g. BMSPC.ucu)*
- ⇒ *Change the Default path to the directory within the site where the supervisor expects to find Configuration files - e.g. C:\Dynamics\Cylon\pdb*
- ⇒ *Select select Save As... from the File menu*
- ⇒ *Save the .ucu file to this location*

Note: This UCU file must be saved with .ucu extension.

9 Import the Exported file into the SCADA Package

From the SCADA interface, import the relevant files according to the manufacturer's instructions.

10 Optimise the OPC configuration

When the UnitronUC32 database is imported from the .csv file, it is likely that it will need to be optimised for your particular iFix SCADA system. This will allow point values to be received more quickly and easily.

In the Database Interface's OPC Export wizard, the Timeout parameters were set to default values. These can now be adjusted for each **Datablock** in UC32 PowerTool.

Suggested adjustments:

Set Primary Rate to 2 seconds for each **Datablock**

Phase each **Datablock** by 0.2 seconds

For Analog **Datablocks** set the Secondary Rate to a value that would reduce the load on the Server (e.g. 10 minutes)

Set the Access Time for Analog **Datablocks** to a value that would reduce the load on the Server (e.g. 05:00).

for Digital **Datablocks** that are being used to trigger alarms, the Access Time should be disabled.

For more information see "Overruns" and "Phasing" in the PowerTool help file.

11 Set OPC Server running

⇒ Press the 'play button' on UC32 PowerTool or the UnitronUC32 OPC Server to connect to the BMS and poll point values.

⇒ In the OPC Client, check the point values that are being returned match the expected values.

If points are not being polled as expected,

- check that: The controllers on the BMS site are on-line and servicing
- check in the UC32 PowerTool, that all of the 'enable' checkboxes (on each Channel, device and **Datablock** page)

12 Verify communications with OPC Client

In your OPC client, create links to Unitron Points and Time Schedules as required, in accordance with the manufacturer's instructions.

Verify that the OPC Client is reading values from the Unitron system as follows:

⇒ Select a point from the OPC Client and view its value.

⇒ In the Unitron Engineering Centre open the Livelog, and select the corresponding point in the Unitron site.

⇒ In the corresponding PowerTool (UC32 or UCU), open the Data Monitor
– Verify that the point value is the same in OPC Client, UEC and PowerTool .

⇒ In the OPC Client, change the value of the point
– Verify that the point has changed in the UEC and PowerTool to match the OPC Client.

Note: If the ethernet cable is unplugged for a period of time while the OPC Server is running, communications with the OPC client may not be restored when the cable is plugged back in. To restore communications, restart the PowerTool software.

Appendices

This page is intentionally blank

13 Appendix - What SCADA packages have been used with the UnitronUC32 OPC server?

iFix

Citect

14 Appendix - Requirements for a UnitronUC32 OPC system

In order to use the UnitronUC32 OPC Server, Field Controllers on the BMS site must have the following firmware:

- UC32 Field Controllers: firmware version 6.06 or later.
- UCU Field Controllers: firmware version 5.74 or later

In addition, if OPC Time Schedules are to be used, the following must be installed on the OPC Client PC:

- Microsoft .net Framework 2

15 Appendix - OPC Capabilities of Unitron Devices

In the Unitron system, the maximum number of points that can be held in a **Datablock** depends on the type of controller in which the point values originate:

In UC32 Field Controllers a **Datablock** can hold up to

- 40 contiguous Analog points, or
- 100 contiguous Digital points,

Note: UC32 PowerTool limits these to 40 Analog and 100 Digital

In UCU Field Controllers a **Datablock** can hold up to

- 3 contiguous Analog points, or
- 96 contiguous Digital points