

[Tech Note 1022](#)

Configuring the MBTCP DAServer to Receive Unsolicited Messages

All Tech Notes and KBCD documents and software are provided "as is" without warranty of any kind. See the [Terms of Use](#) for more information.

Topic#: 002855

Created: February 2014

Introduction

Unsolicited messaging provides the capability for a PLC to inform the DAServer of critical events immediately, without having to wait for the DAServer to poll for the data. DASMBTCP provides the ability to receive unsolicited messages from a PLC. In order to support unsolicited messages from the PLC, the MBTCP DAServer listens on ethernet port 502. The unsolicited messaging feature is available only to controllers that are directly connected to the TCPIP Port. Controllers under the Modbus Bridge object hierarchy cannot use this feature.

This *Tech Note* provides a sample configuration that will demonstrate unsolicited messaging using a Quantum PLC. In this example, the IP Address of the PLC is 10.2.82.112 and the IP address of the computer running DASMBTCP is 10.2.82.139

Note: This *Tech Note* assumes basic knowledge of DASMBTCP and Modicon PLC programming.

Application Versions

- MBTCP DAServer 3.0 SP1

Configuring the PLC MSTR instruction

In order for the PLC to send an unsolicited message, a MSTR Write instruction must be programmed. The MSTR Write instruction sends data from the PLC to a specified device on the network (DASMBTCP). In this example, the MSTR instruction is programmed to send data from 50 registers to DASMBTCP.

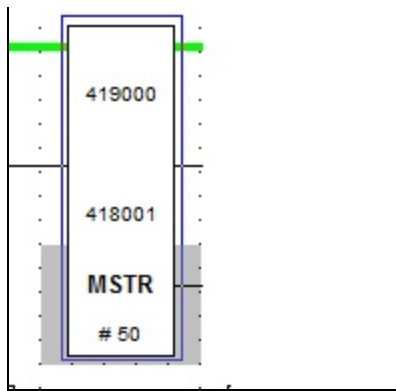


FIGURE 1: MSTR WRITE

- The top node (419000) in the above instruction is the starting address of the Control Block. The Control Block can be configured by going to the DX Zoom screen in the PLC programming software.
- The middle node (418001) is the starting register that holds the source of the data to be sent (Data Area).
- The bottom node (#50) is the length of the data area.
- The top input is used to enable the MSTR operation
- The middle input is used to terminate the MSTR operation
- The top output is on while the instruction is active
- The middle output is on if the instruction is terminated
- The bottom output is on if the operation is successful

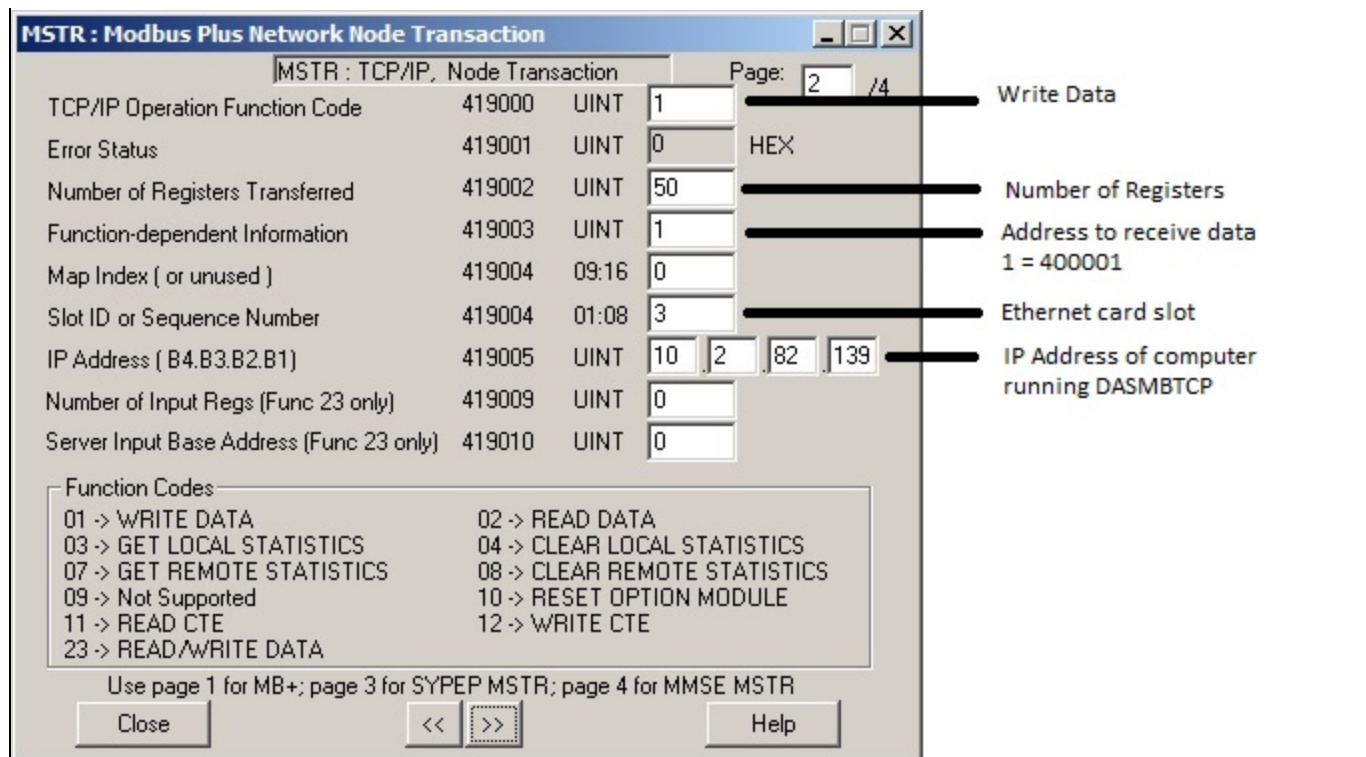


FIGURE 2: CONFIGURING THE CONTROL BLOCK USING DX ZOOM

In this example, the MSTR instruction is configured to write data from 50 consecutive registers starting at 418001 in the PLC to 50 consecutive registers starting at 400001 in the DAServer running on IP address 10.2.82.139

- The **TCP/IP Operation Function Code** is set to 1 (Write Data)
- The **Number of Registers Transferred** is set to 50
- The **Function dependent Information** contains the starting holding register in DASMBTCP that will receive the data. 1 = 400001
- The **Slot ID or Sequence Number** is set to 3. This is the slot in the PLC chassis where the ethernet card resides
- The **IP Address** is the IP address of the computer running DASMBTCP

Configuring the MBTCP DAServer to Accept Unsolicited Messages

Configuring DASMBTCP to accept unsolicited data consists of creating a PLC object, a device group with an update rate of 0 and selecting **Support Unsolicited Messages** for that device group.

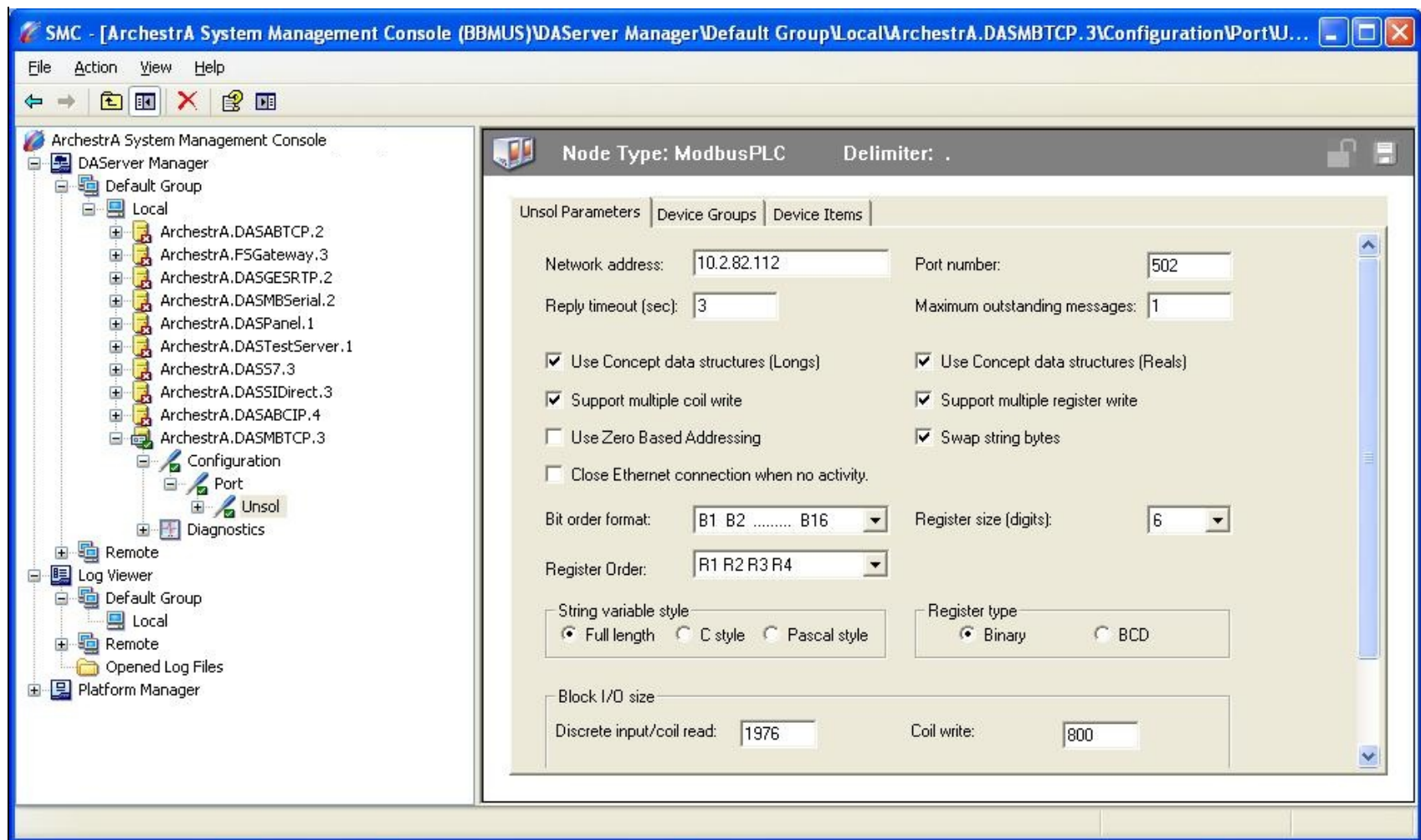


FIGURE 3: DASMBTCP CONFIGURATION

This example uses a ModbusPLC object. The **Network address** is set to the PLC's IP Address (10.2.82.112). All other parameters are using default values.

1. Create a device group named **unsol**. Since we don't need to poll this group, an **Update Interval** of 0 is used.
2. Since this device group is being used to accept unsolicited messages, right-click on the device group name and click **Edit**.
3. Place a check mark next to **Support Unsolicited Messages** then select **OK** then save.

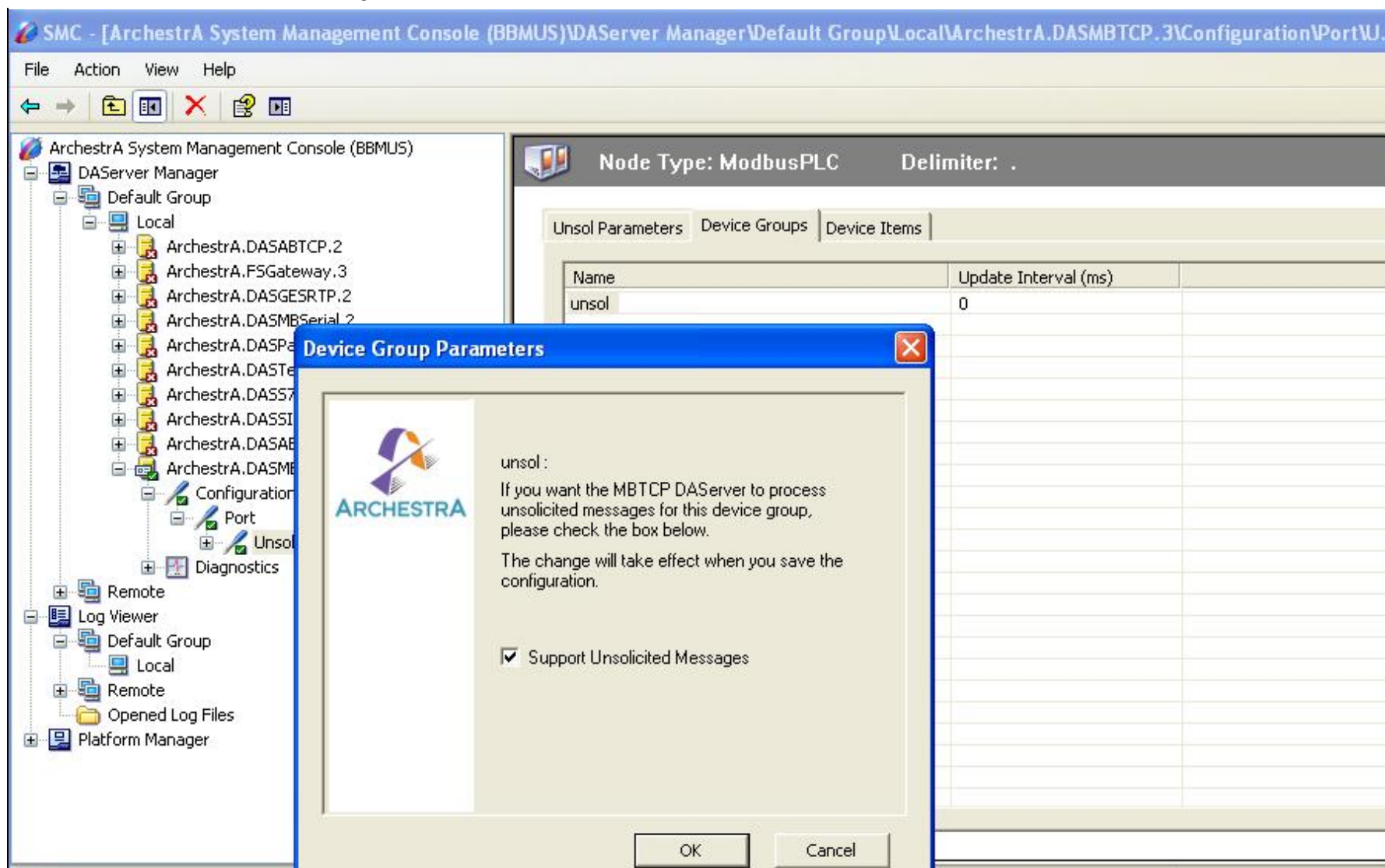


FIGURE 4: SUPPORT UNSOLICITED MESSAGES

After activating DASMBTCP use a client to advise registers 400001 thru 400050. The client application in this example is **WWClient**.

1. Open **WWClient**
2. Select **Connections -> Create**.
3. Type **DASMBTCP** as the Application, type **unsol** as the Topic and click **IOT** as the Connection Type.
4. Click **Create**.
5. Click **Item** then type **400001..400050** in the Item field.

6. Click **AdviseEx**.

You see the following window:

The screenshot shows the Wonderware Client application window. The title bar reads "Wonderware Client" and the menu bar includes "File", "Script", "Connections", "Item", and "Help". The main window displays a list of data points in a table format. The columns represent the data point name, its value, its address, and its timestamp. The data points are listed from 400001 to 400050.

IDT	Address	Value	Timestamp	Hex Value
	\\localhost\dasmbtcp\unsol	0x003a5640	49772	
400001	3324	12:43.29.0892	02/13/2014	0x00c0
400002	3324	12:43.29.0892	02/13/2014	0x00c0
400003	3324	12:43.29.0892	02/13/2014	0x00c0
400004	3324	12:43.29.0892	02/13/2014	0x00c0
400005	3324	12:43.29.0892	02/13/2014	0x00c0
400006	3324	12:43.29.0892	02/13/2014	0x00c0
400007	3324	12:43.29.0892	02/13/2014	0x00c0
400008	3324	12:43.29.0892	02/13/2014	0x00c0
400009	3324	12:43.29.0892	02/13/2014	0x00c0
400010	3324	12:43.29.0892	02/13/2014	0x00c0
400011	1	10:45.58.0770	02/13/2014	0x00c0
400012	0	10:45.58.0770	02/13/2014	0x00c0
400013	1	10:45.58.0770	02/13/2014	0x00c0
400014	0	10:45.58.0770	02/13/2014	0x00c0
400015	1	10:45.58.0770	02/13/2014	0x00c0
400016	0	10:45.58.0770	02/13/2014	0x00c0
400017	1	10:45.58.0770	02/13/2014	0x00c0
400018	0	10:45.58.0770	02/13/2014	0x00c0
400019	1	10:45.58.0770	02/13/2014	0x00c0
400020	0	10:45.58.0770	02/13/2014	0x00c0
400021	764	12:43.32.0346	02/13/2014	0x00c0
400022	16707	12:43.29.0892	02/13/2014	0x00c0
400023	16707	12:43.29.0892	02/13/2014	0x00c0
400024	16707	12:43.29.0892	02/13/2014	0x00c0
400025	16707	12:43.29.0892	02/13/2014	0x00c0
400026	16707	12:43.29.0892	02/13/2014	0x00c0
400027	16707	12:43.29.0892	02/13/2014	0x00c0
400028	16707	12:43.29.0892	02/13/2014	0x00c0
400029	16707	12:43.29.0892	02/13/2014	0x00c0
400030	16707	12:43.29.0892	02/13/2014	0x00c0
400031	0	12:35.57.0684	02/13/2014	0x00c0
400032	0	12:35.57.0684	02/13/2014	0x00c0
400033	0	12:35.57.0684	02/13/2014	0x00c0
400034	0	12:35.57.0684	02/13/2014	0x00c0
400035	0	12:35.57.0684	02/13/2014	0x00c0
400036	0	12:35.57.0684	02/13/2014	0x00c0
400037	0	12:35.57.0684	02/13/2014	0x00c0
400038	0	12:35.57.0684	02/13/2014	0x00c0
400039	0	12:35.57.0684	02/13/2014	0x00c0
400040	0	12:35.57.0684	02/13/2014	0x00c0
400041	0	12:35.57.0684	02/13/2014	0x00c0
400042	0	12:35.57.0684	02/13/2014	0x00c0
400043	0	12:35.57.0684	02/13/2014	0x00c0
400044	0	12:35.57.0684	02/13/2014	0x00c0
400045	0	12:35.57.0684	02/13/2014	0x00c0
400046	0	12:35.57.0684	02/13/2014	0x00c0
400047	0	12:35.57.0684	02/13/2014	0x00c0
400048	0	12:35.57.0684	02/13/2014	0x00c0
400049	0	12:35.57.0684	02/13/2014	0x00c0
400050	0	12:35.57.0684	02/13/2014	0x00c0

FIGURE 5: WW CLIENT

Activating the MSTR instruction in the PLC will now send the data in PLC registers 418001 thru 418050 to DASMBTCP advised registers 400001 thru 400050.

G. Alldredge

Tech Notes are published occasionally by Wonderware Technical Support. Publisher: Invensys Systems, Inc., 26561 Rancho Parkway South, Lake Forest, CA 92630. There is also technical information on our software products at [Wonderware Technical Support](#).

 [Back to top](#)

©2014 Invensys Systems, Inc. All rights reserved. No part of the material protected by this copyright may be reproduced or utilized in any form or by any means, electronic or mechanical, including photocopying, recording, broadcasting, or by any information storage and retrieval system, without permission in writing from Invensys Systems, Inc.

[Terms of Use](#).