

User Guide Supplement

**Model VP6-1470
ValuPoint IoT
Edge Server for BACnet**

Rev. 1.0 – Feb. 2024

VP6-1470 User Guide Supplement Contents

[1 Introduction](#)

- 1.1 How to Use This Guide
- 1.2 Important Safety Notice
- 1.3 Overview of the ValuPoint Devices
- 1.4 Warranty
- 1.5 Required License Information

[2 Installation](#)

- 2.1 Installing the Software

[3 Local Port Page](#)

- 3.1 Connect with BACnet MS/TP
- 3.2 Connect with BACnet IP
- 3.3 Auto Connect (MS/TP)

[4 Who-Is Page](#)

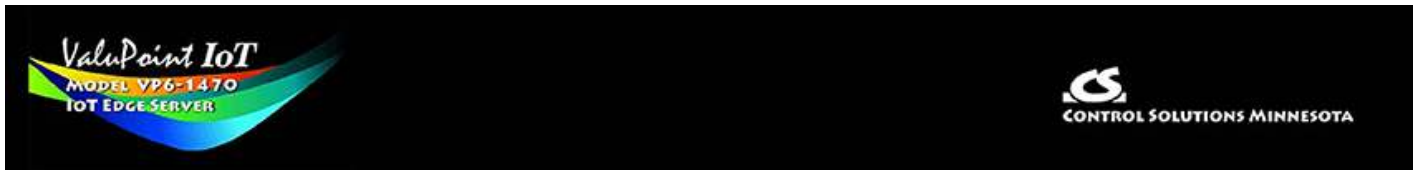
- 4.1 Finding Devices
- 4.2 Select Target
- 4.3 Getting Device Information
- 4.4 Clear Who-Is Cache

[5 Read/Write Page](#)

- 5.1 Read Property
- 5.2 Write Property

[6 Programming Page](#)

- 6.1 Program Loading and Execution
- 6.2 Program Editing and Debugging
- 6.3 Program Capacity
- 6.4 Program States and Error Codes



1 Introduction

1.1 How to Use This Guide

Section 1 gives an overview of the VP6-1470 programmable I/O device. Section 2 talks about installing the configuration software and connecting the VP6-1470. Sections 3 through 12 are guides for each of the tabs found on the screen of the configuration software. Appendix A through F are reference material.

1.2 Important Safety Notice

Proper system design is required for reliable and safe operation of distributed control systems incorporating any Control Solutions product. It is extremely important for the user and system designer to consider the effects of loss of power, loss of communications, and failure of components in the design of any monitoring or control application. This is especially important where the potential for property damage, personal injury, or loss of life may exist. By using ANY Control Solutions, Inc., product, the user has agreed to assume all risk and responsibility for proper system design as well as any consequence for improper system design.

CAUTION: The lithium battery contained in this device may explode if mistreated. DO NOT recharge, disassemble, or dispose of in fire.

No action is required of the user to activate the battery that backs up the real time clock. Important: Replace battery with BR1225A only. Use of another battery may present a risk of fire or explosion.

1.3 Overview of the VP6-1470

This programming tool is a supplement to the main user guide which walks you through the Web User Interface. All configuration of the VP6-1470 is done through the Web UI. Instructions and additional detail are provided in the main user guide. Please refer to that document as this document is only a supplement.

1.4 Warranty

This configuration software and documentation is provided "as is," without warranty of any kind, either expressed or implied, including, but not limited to, the implied warranties of fitness or merchantability for a particular purpose. Control Solutions may make improvements and/or changes in this documentation or in the

product(s) and/or the program(s) described in this documentation at any time. This product could include software bugs, technical inaccuracies, typographical errors, and the like. Changes are periodically made to the information herein; these changes may be incorporated in new editions of the software.

Warranty: All Control Solutions products are warranted against defects in materials and workmanship for a period of time from date of shipment from factory as follows: Two years on non-mechanical parts, one year on mechanical parts (e.g. relays). Defective units will be repaired or replaced, at manufacturer's discretion, at no cost to user except when negligence or improper use has resulted in damage. The express warranty stated herein is in lieu of all other warranties, express or implied, including without limitation any warranties of merchantability or fitness for a particular purpose and all other warranties are hereby disclaimed and excluded by Control Solutions, Inc.

Configuration errors made by customer are not covered under warranty. Damage caused by incorrect electrical connection is not covered under warranty. Removing circuit boards from their enclosures will void the warranty - the complete product with all of its original circuit boards and components must be returned for warranty consideration.

1.5 Required License Information

The VP6-1470 configuration and line programming tools include the SmartWin library (<http://smartwinlib.org>) under the following terms:

License agreement for SmartWin++ (BSD license)

Copyright (c) 2005, Thomas Hansen All rights reserved.

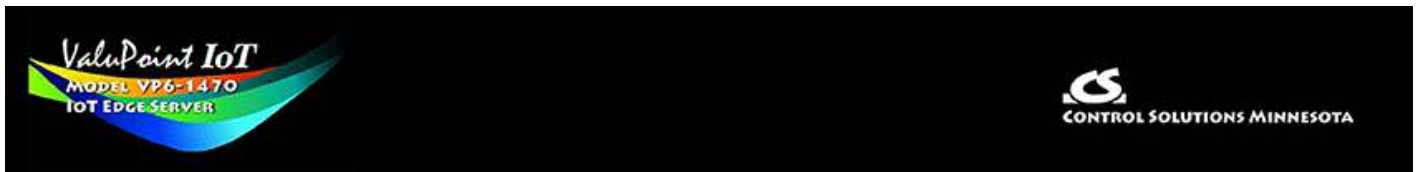
Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

- * Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
- * Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
- * Neither the name of the SmartWin++ nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY

THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

i.CanDrawIt includes, licensed under LGPL, TinyCAD, Copyright 1994-2009 Matt Pyne. Source code is available at <http://tinycad.sourceforge.net>. Open source products included under either GPL or LGPL include TinyCAD v2.70; Unicode/Font Conversions: iconv.dll version 1.9.0.0; PNG Image Support: libpng13.dll version 1.2.8.0; Image compression support: zlib1.dll version 1.2.1.0.



2 Installation

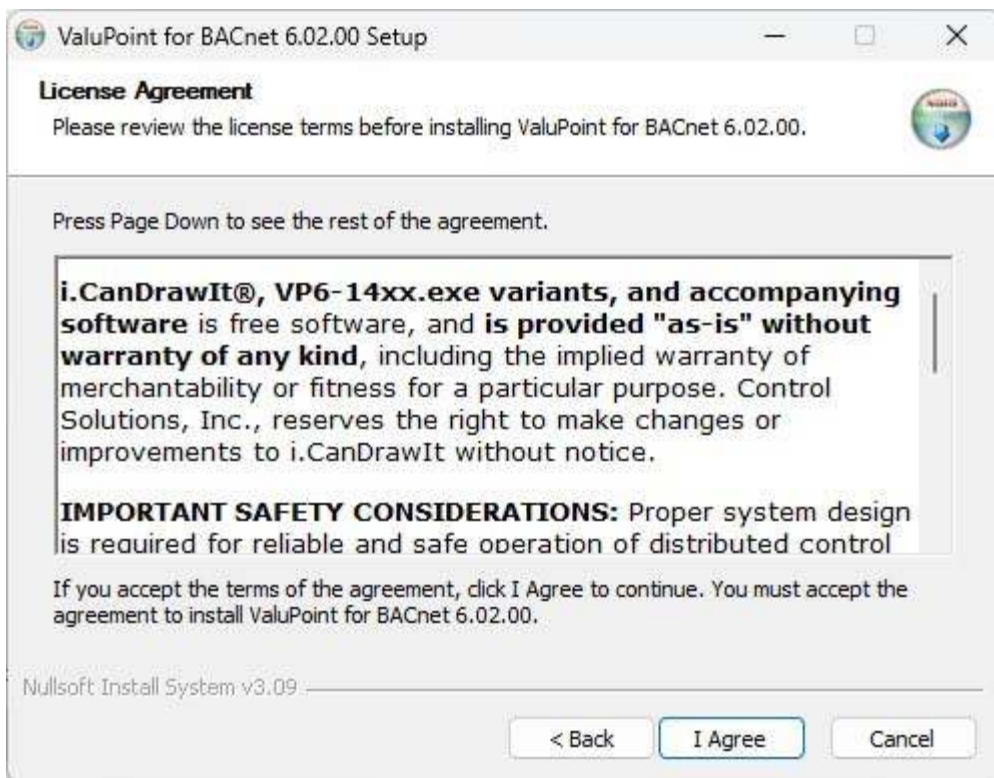
2.1 Installing the Software

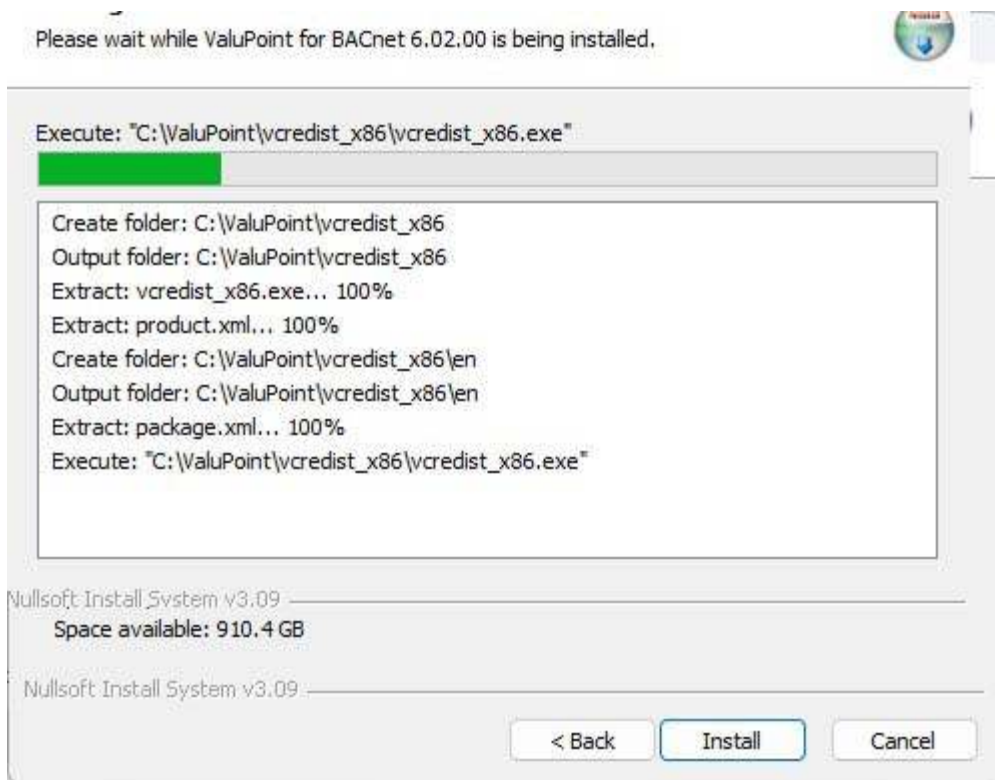
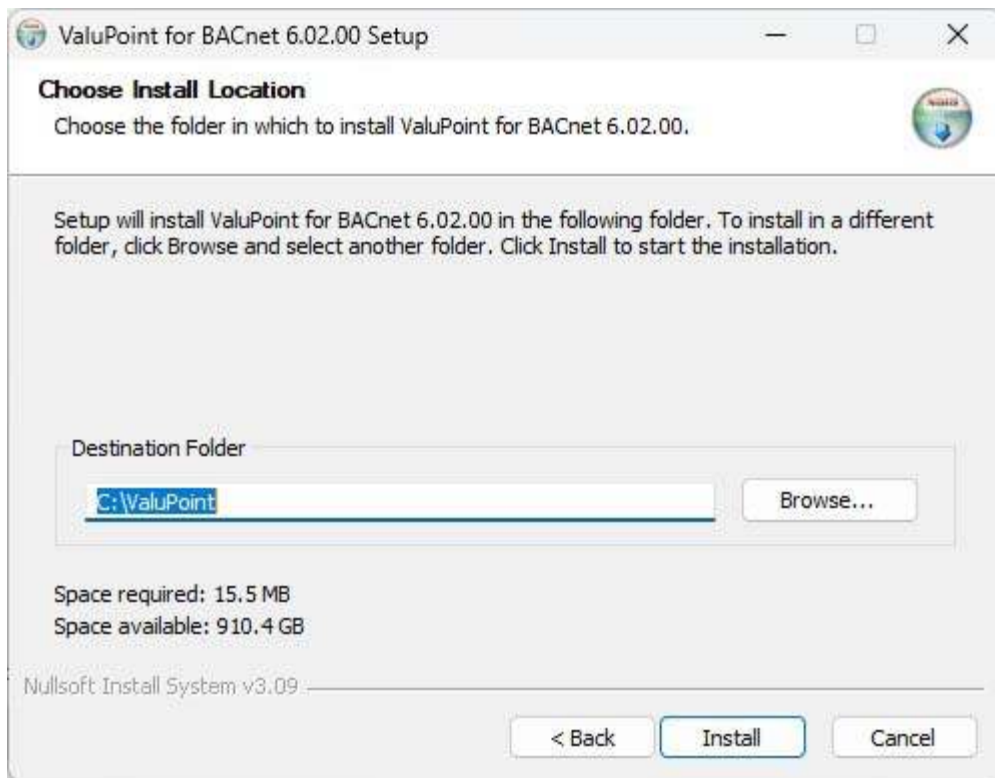
Look for the installer icons in the directory where you unzipped the download that got you to this document. The installer icons look like this:



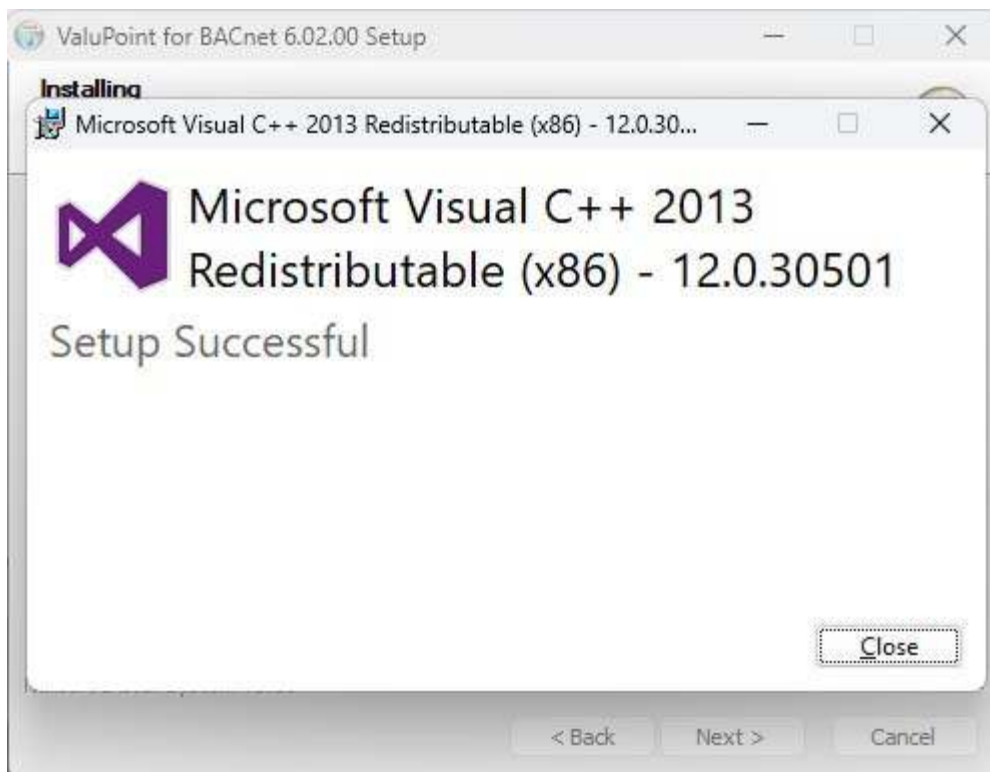
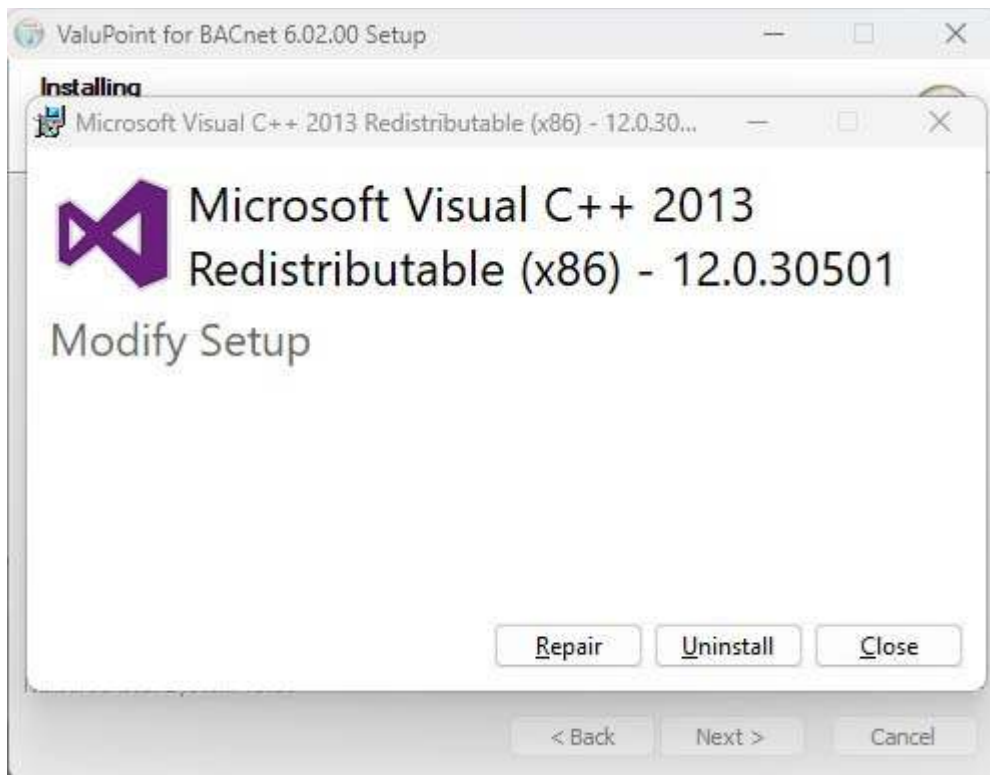
The installation is a 2-step installation. Install VP6_1430 first. Then install i.CanDrawIt second.

Double click the icon to run the setup.exe. You will be questioned about whether to continue because Windows cannot verify the publisher of the software. Permit installation to continue. The sequence of installer screens include the following on Windows 11:

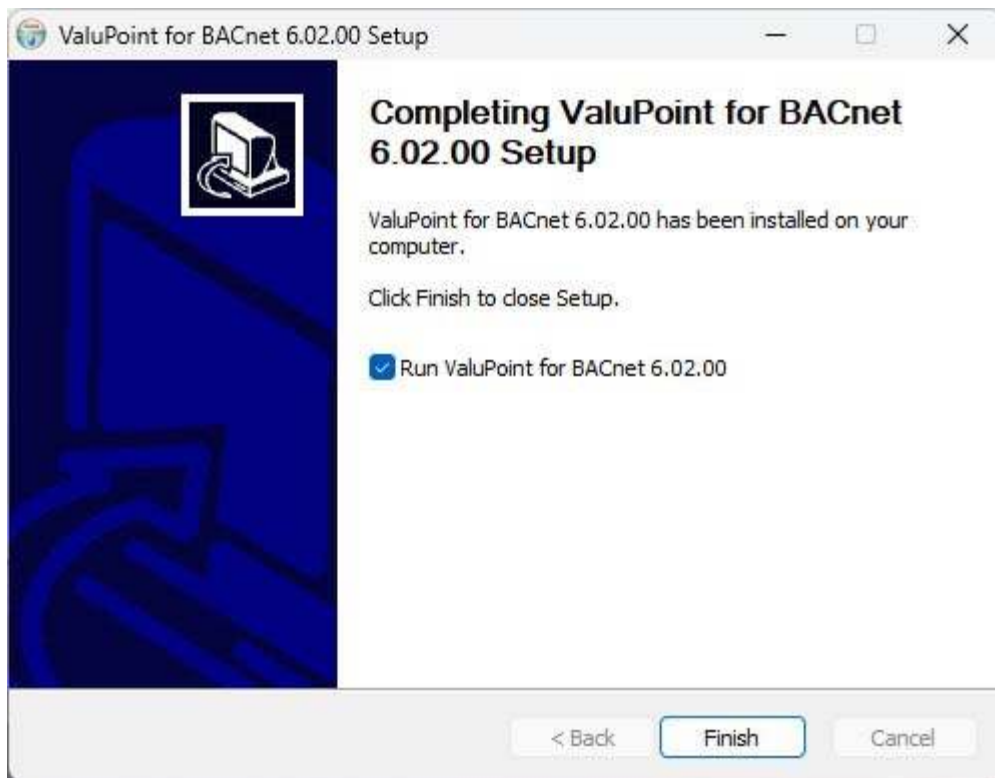




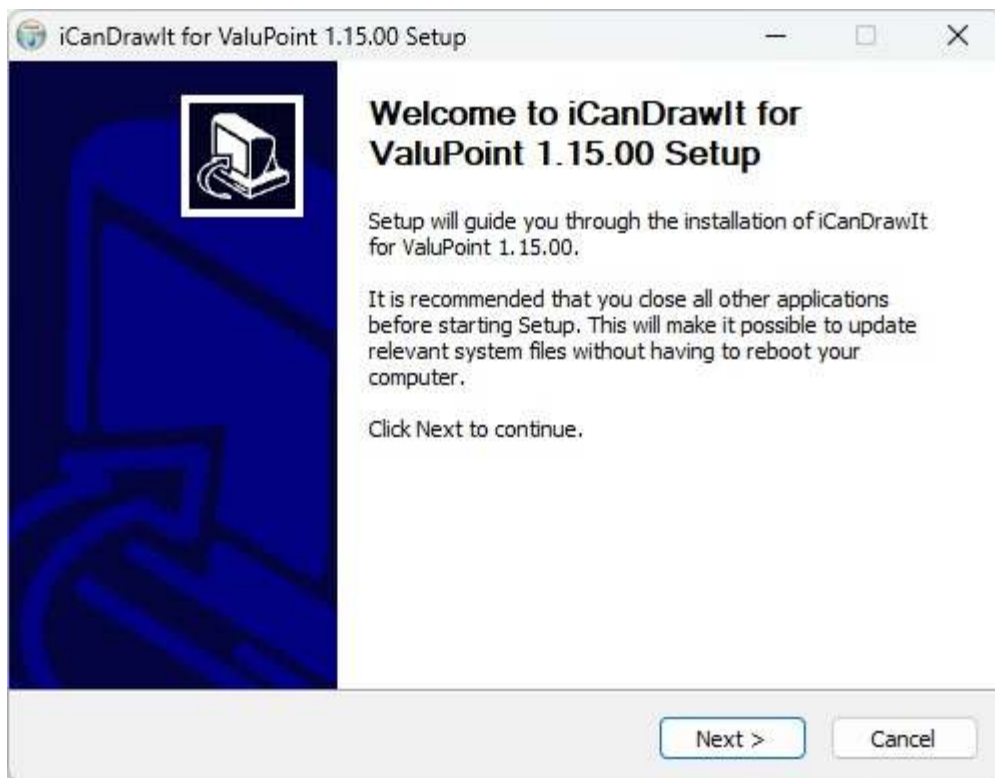
The installer will check to see whether Visual C++ support is already installed on your system, and install it if not. This is standard software provided by Microsoft. If it has already been installed previously, it will ask whether to repair or uninstall. Select Repair.



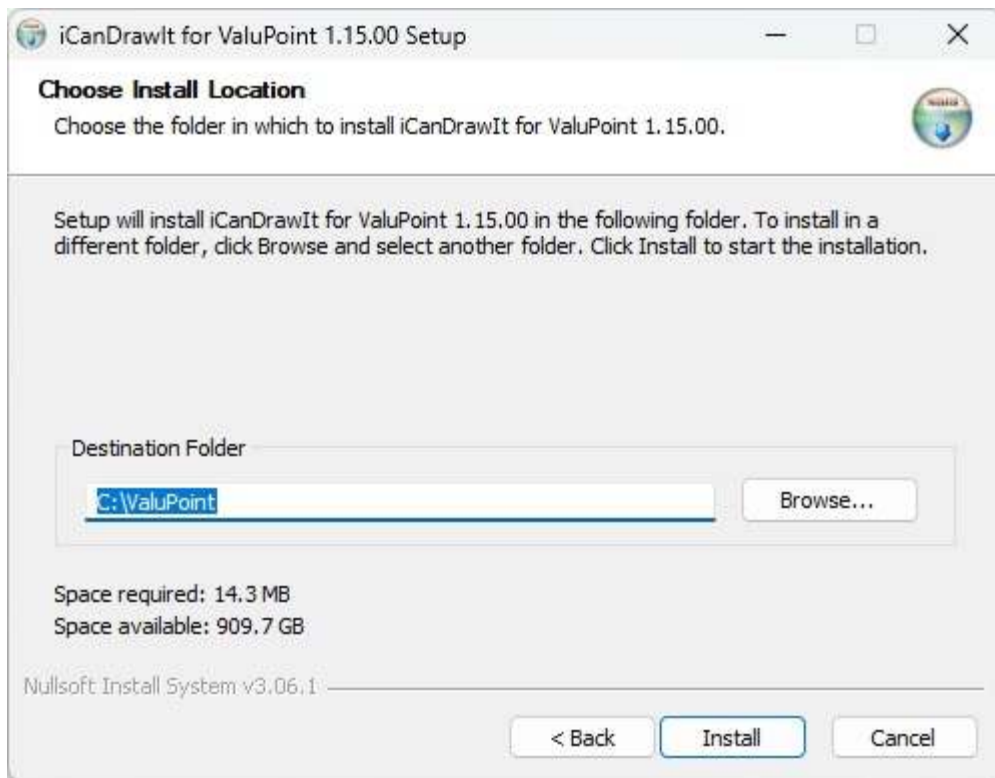
When you get to the "Finish" screen, you are ready to go.



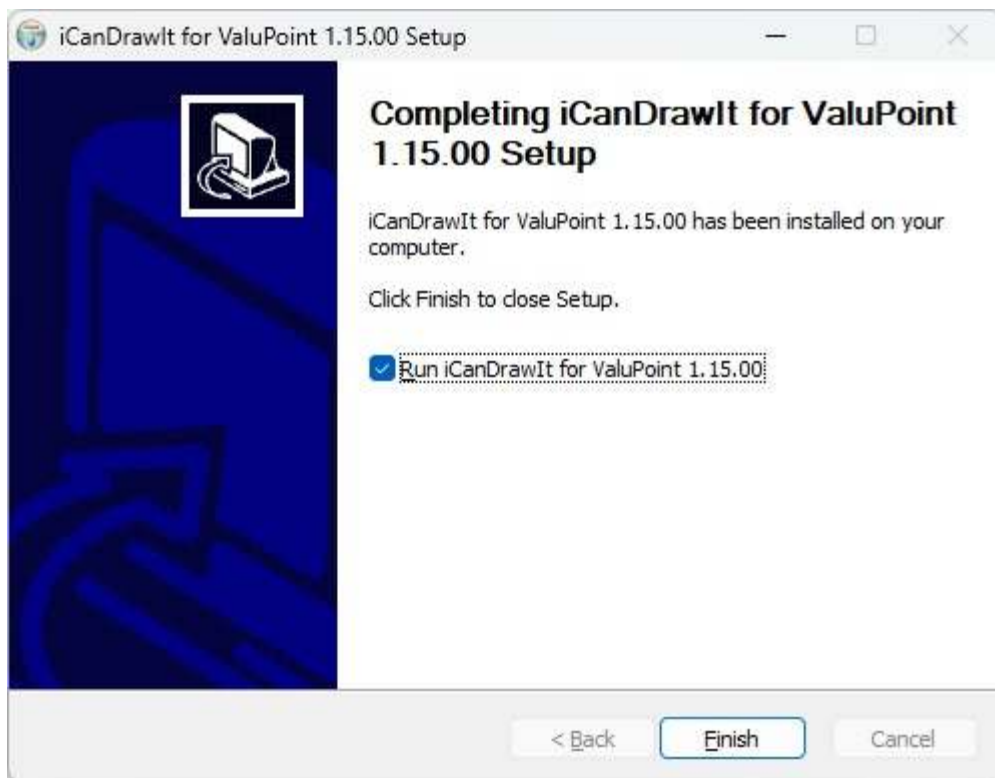
Next, proceed to install i.CanDrawIt. The first installer screen looks like this:

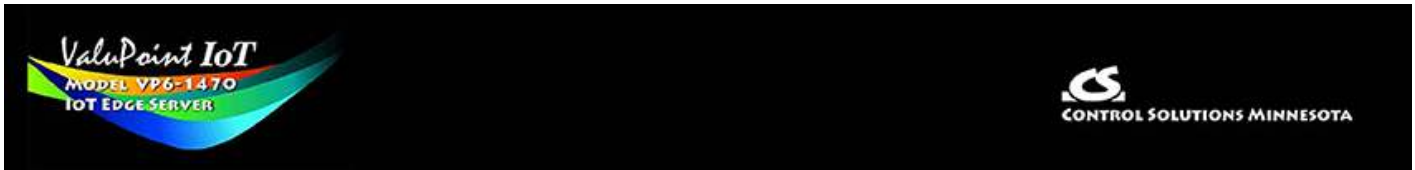


The installation directory should be the same directory that VP6-1430 was installed into.

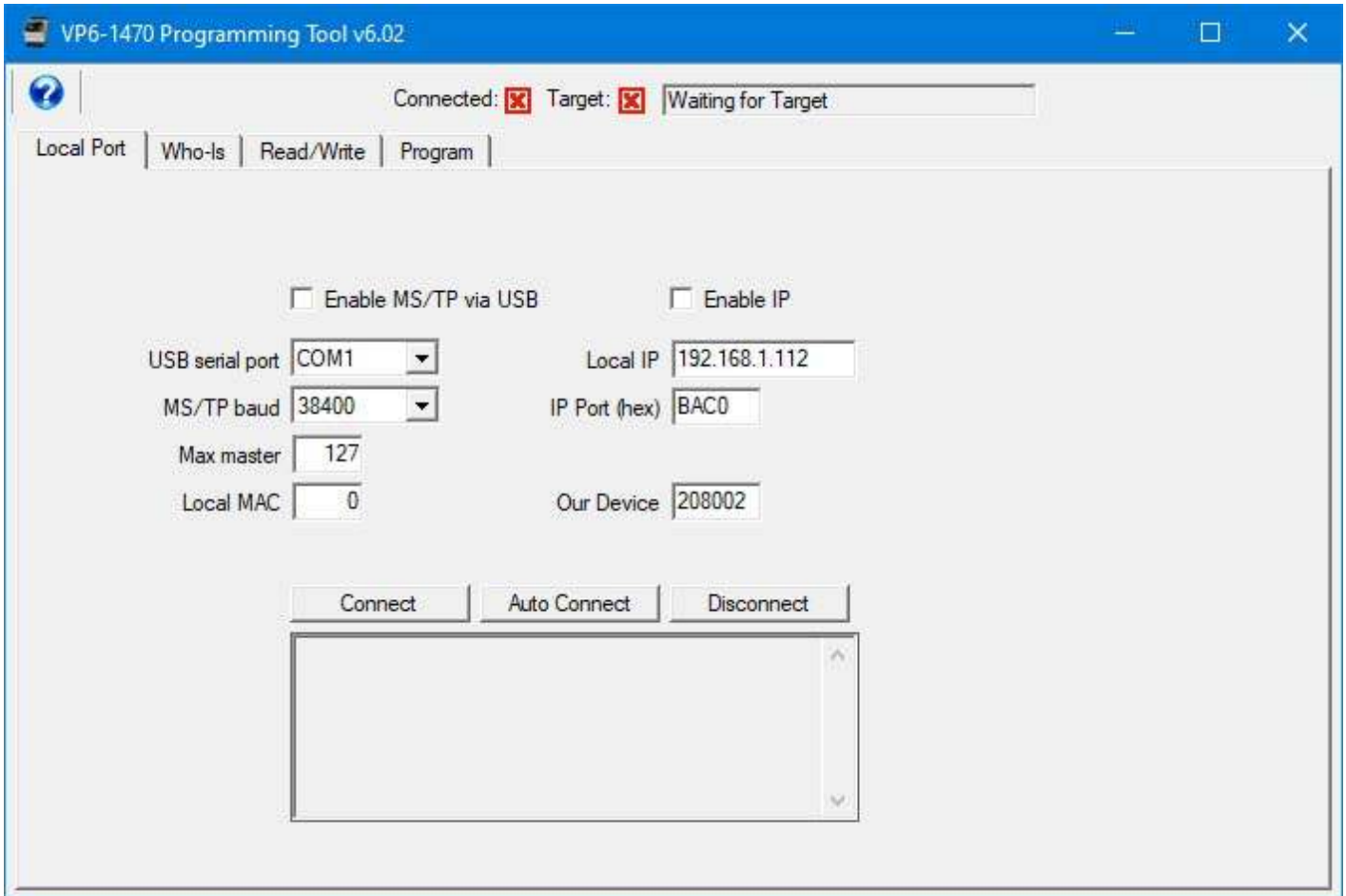


After a few more screens, you will get the familiar 'done' screen.





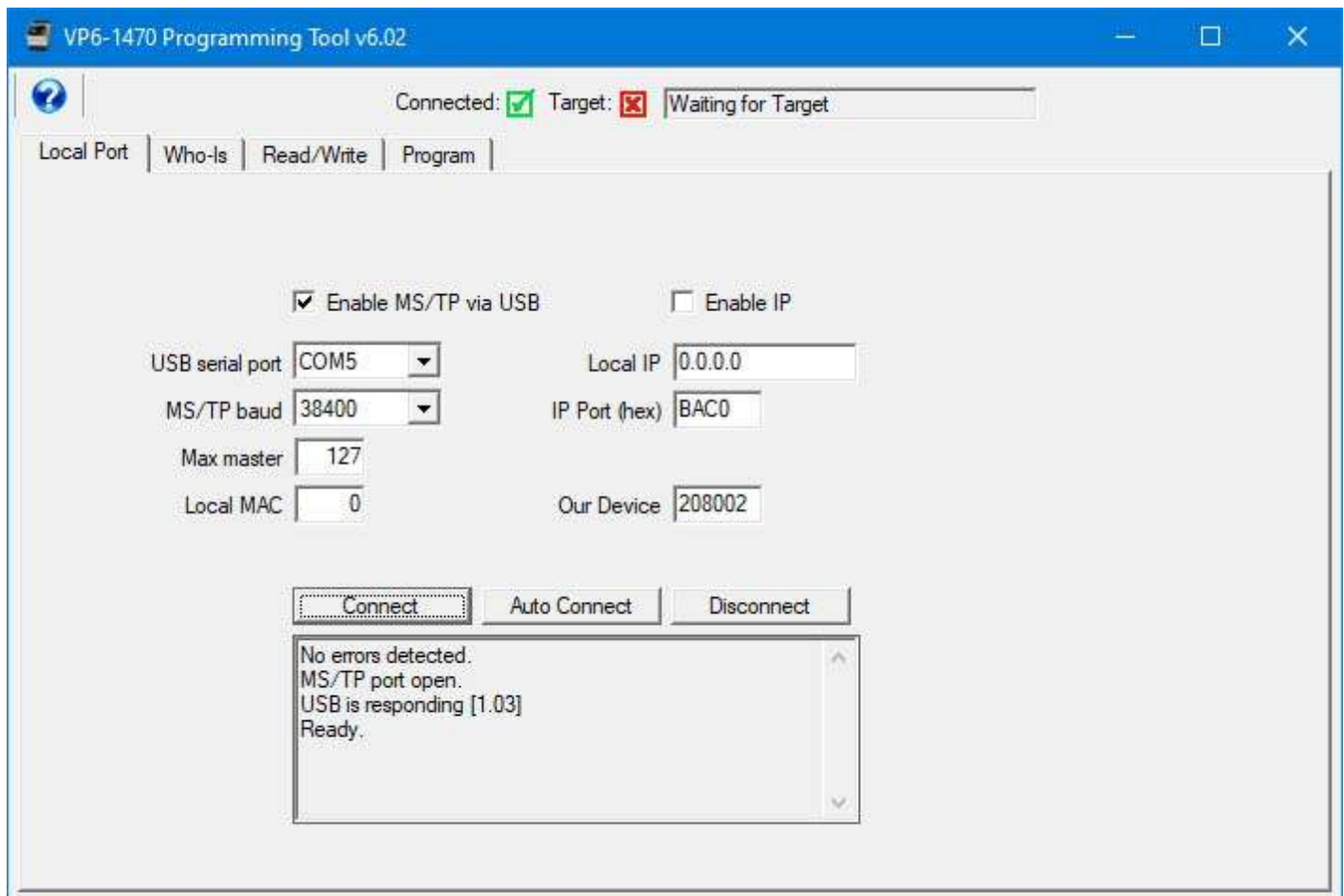
3 Local Port Page



The configuration tool will act as a BACnet device on the network. To connect the configuration tool to the BACnet network, you will use the Local Port page in the configuration tool. The configuration tool's Device instance will be 'Our Device'.

3.1 Connect with BACnet MS/TP

Assuming you have connected and installed the USB to MS/TP adapter ("dongle"), check the box 'Enable MS/TP via USB'. Select the COM port assigned by your PC to your adapter. Find the Device Manager hardware properties in the control panel of your PC and look for the COM port if you have not done so already.



If connecting to a ValuPoint for the first time, as shipped from the factory, leave the remaining settings at their defaults. Otherwise, select the baud rate your ValuPoint is configured to run (default is 38400). Set Max Masters and Local Mac to your known values if they have been changed. If device instance 208002 is already used by some other device, change the 'Our Device' number before connecting.

If the MS/TP MAC address displayed is already used by some other device, enter a different MAC address before clicking Connect. This MAC address will be used by the configuration tool. You should also enter a Max Master value other than the default 127 if you know some other value is in use.

Click the Connect button. You should see the message illustrated above indicating no errors, port open, USB is responding, and Ready. If not, stop here and determine why. Re-check the port number used. See that MTX002 shows up under Ports (COM & LPT) in your PC's Device Manager hardware list. If MTX002 does not show up, revisit the USB installation instructions.

In addition to seeing "Ready", you should see the connected icon at the top change from a red X to a green check mark. When you see "Ready", you are ready to move on to the Who-Is page.

Note: You may see "No errors detected. MS/TP port open." even though your MTX002 is not connected. This means the configuration software found a valid COM port and opened that port, but there is no MTX002 on that port. If you do not see "USB is

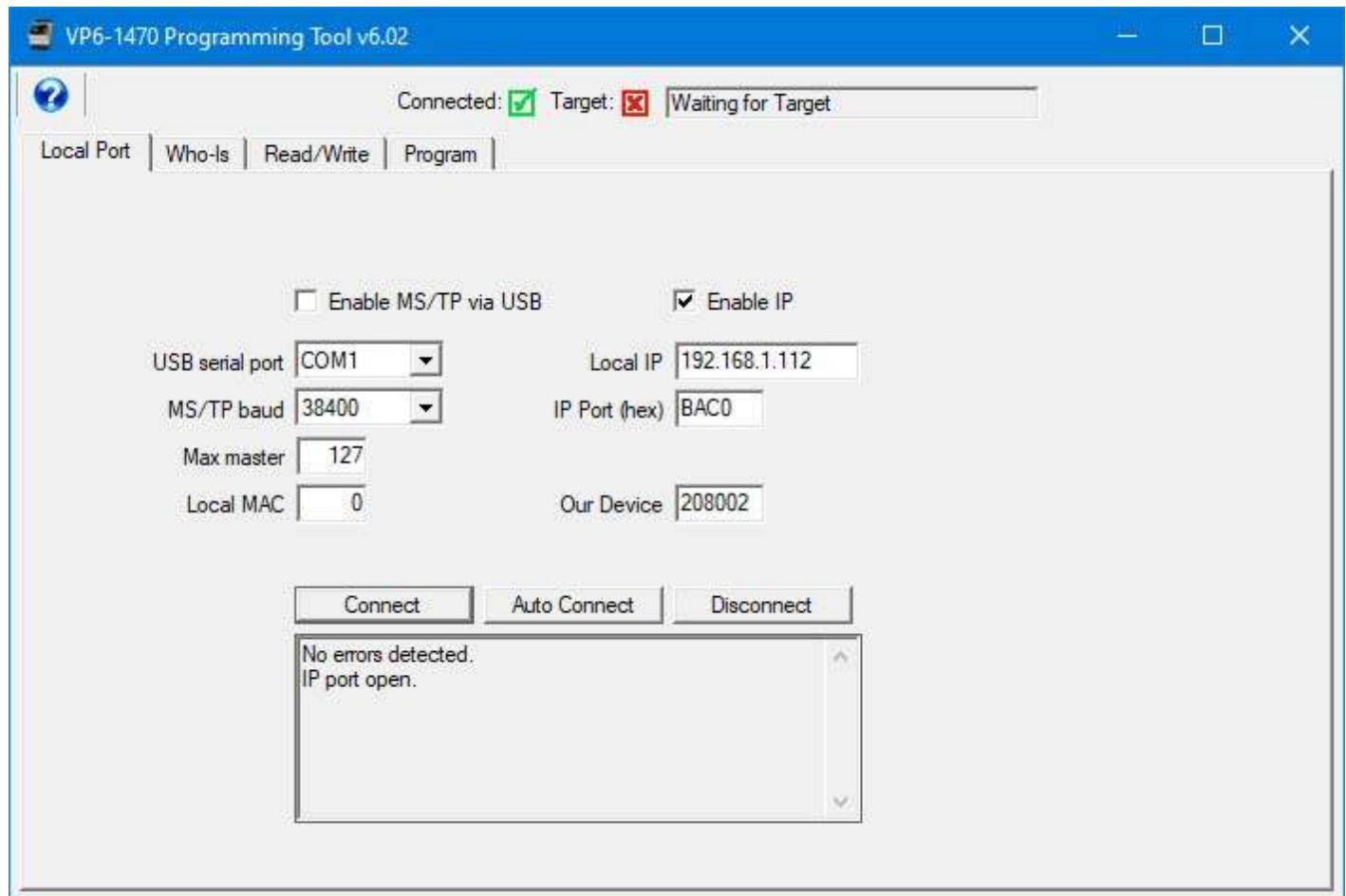
responding. Ready." then you are not ready.

NOTE: The MS/TP interface for this software requires the MTX002. You cannot use a generic USB to RS485 adapter. This is because the MTX002 is an intelligent adapter that handles all of the token passing within the adapter, and only passes APDU's via USB.

3.2 Connect with BACnet IP

Check the 'Enable IP' box if you will be accessing the ValuPoint via BACnet IP. You can access the VP6-1470 using BACnet IP via direct network connection. Enter the port number (0xBAC0 is the default) in hexadecimal. If device instance 208002 is already used by some other device, change the 'Our Device' number before connecting. The configuration tool itself becomes a BACnet device on the network, and therefore requires its own device instance.

Upon opening the program, the Local IP window will display the IP address that Windows has chosen as its default. However, if this is not the desired interface, enter the IP address of the interface you would rather use, then check Enable IP, and click Connect.

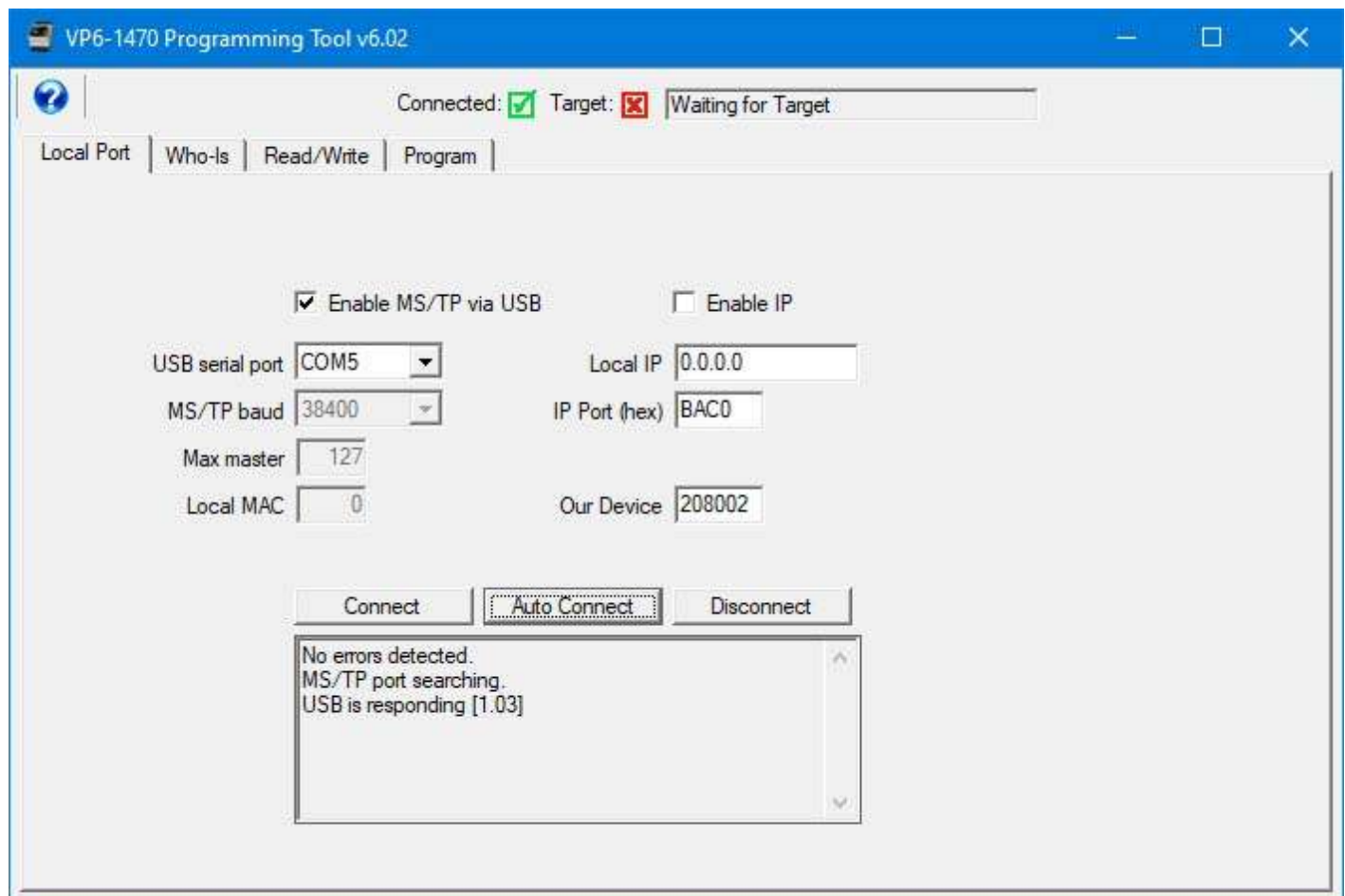


If the IP port is successfully connected, you will see the message "No errors detected. IP port open." You are now ready to move on to the Who-Is page.

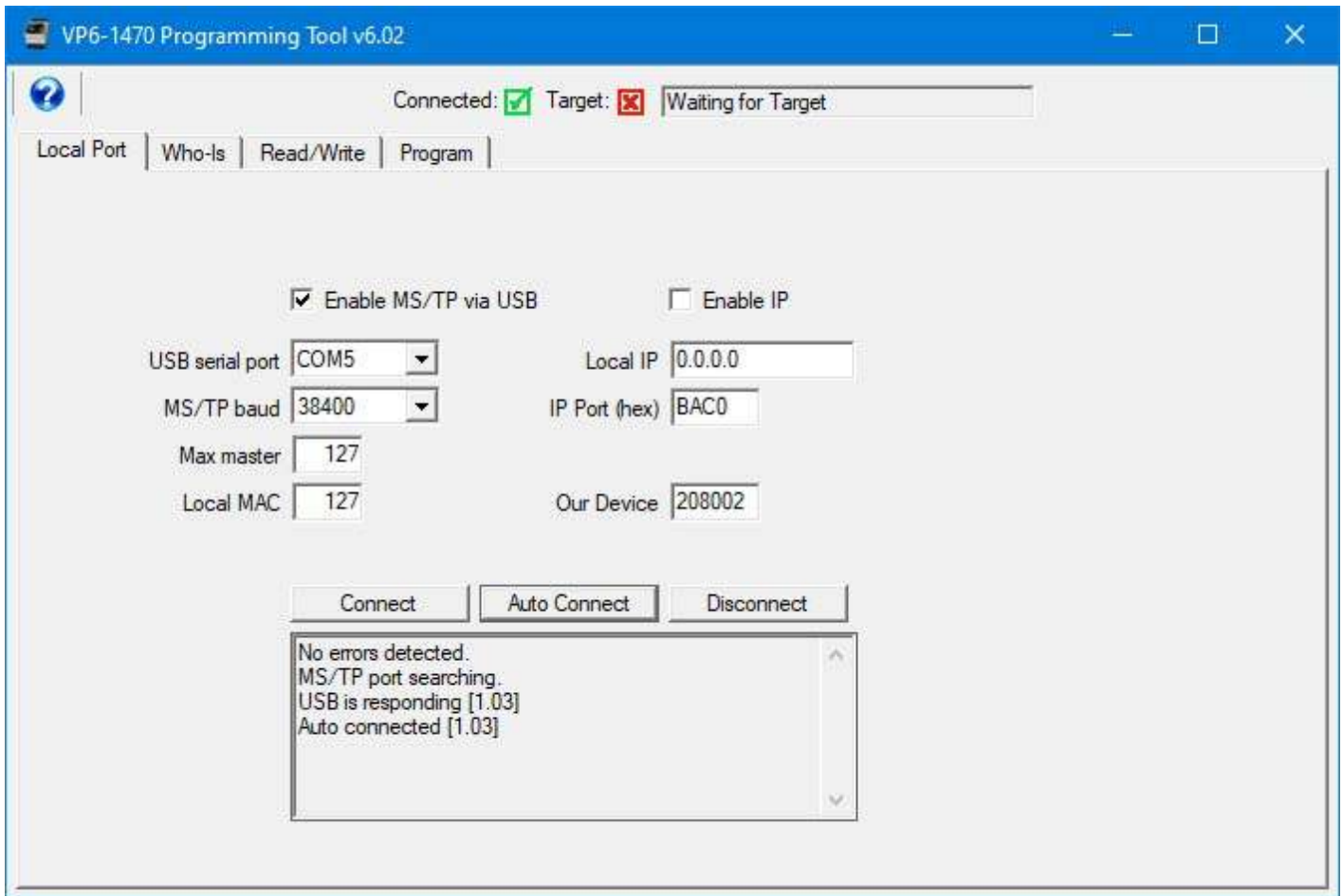
3.3 Auto Connect (MS/TP)

The fastest way to connect is to select the known baud rate, max master setting and some unused local MAC. However, if you are connecting to a ValuPoint that has been previously configured and you do not know its settings, you can use Auto Connect to search for it. The Auto Connect does take some time because it is testing the various possible baud rates. Then when it finds a baud rate that seems right, it spends some time listening to polls for master to determine the max master setting that seems to be in effect. It is also looking for an unused MAC (station ID) to use for the tool itself. The tool will assign itself the highest numbered unused MAC address. Once a seemingly good set of parameters is found, these will be inserted in the appropriate windows for you, and you will see the indication 'Auto connected'. You may now proceed.

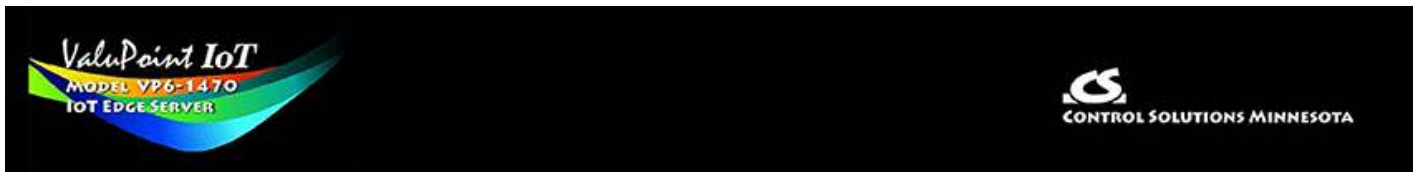
To auto connect, check Enable MS/TP via USB, then click the Auto Connect button. The screen will initially look like this:



Once the auto connect process has completed, the disabled windows will become enabled, and will display the parameters found. When you see the message "Auto connected" show up below "USB is responding", you are ready to move on to the Who-Is window.



Note: The Auto Connect feature is included here simply because it is part of the MS/TP support included for all of Control Solutions' MS/TP products. For the VP6-1470, the faster way to learn the port settings is to simply log into the web UI and look at the port setting page.

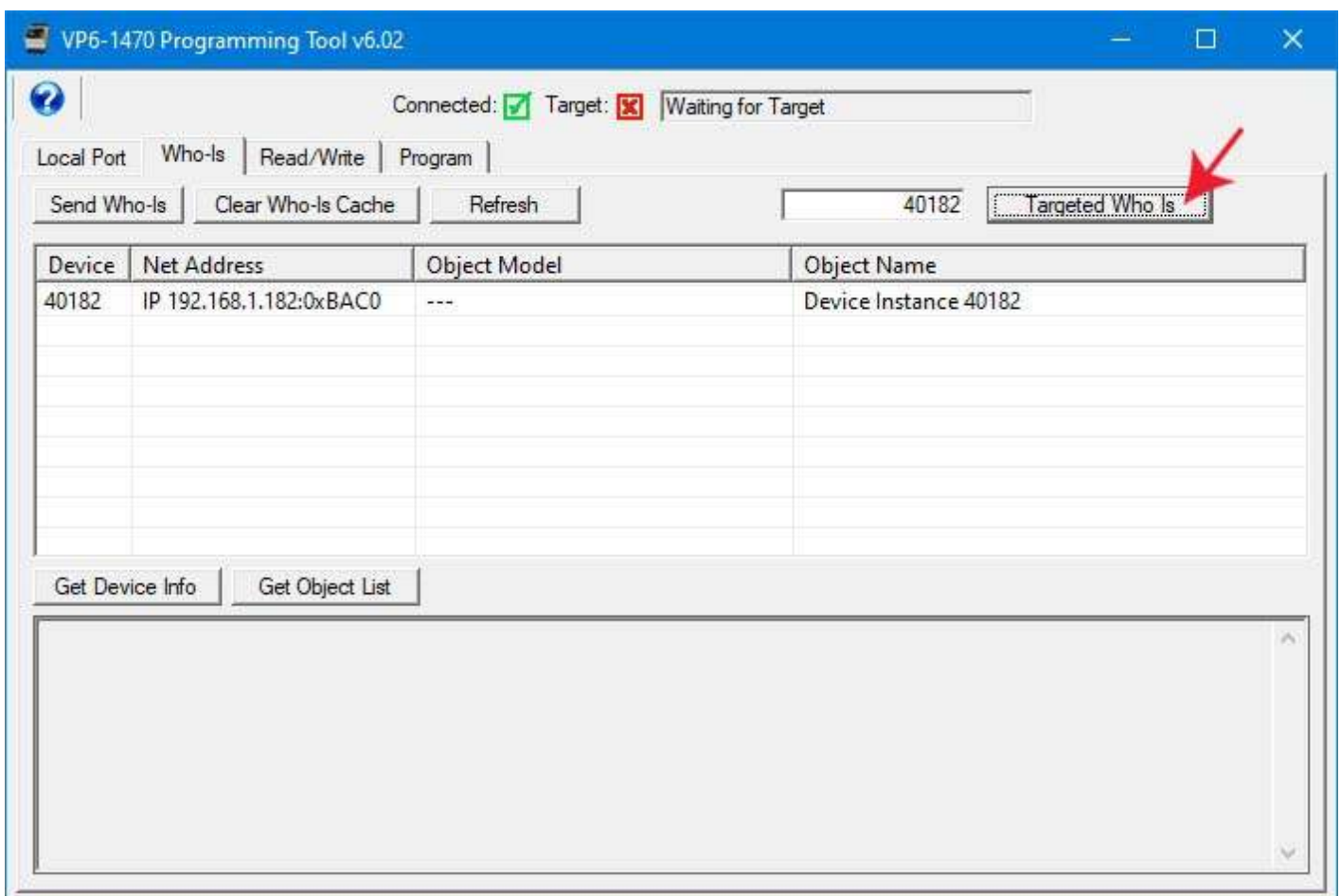


4 Who-Is Page

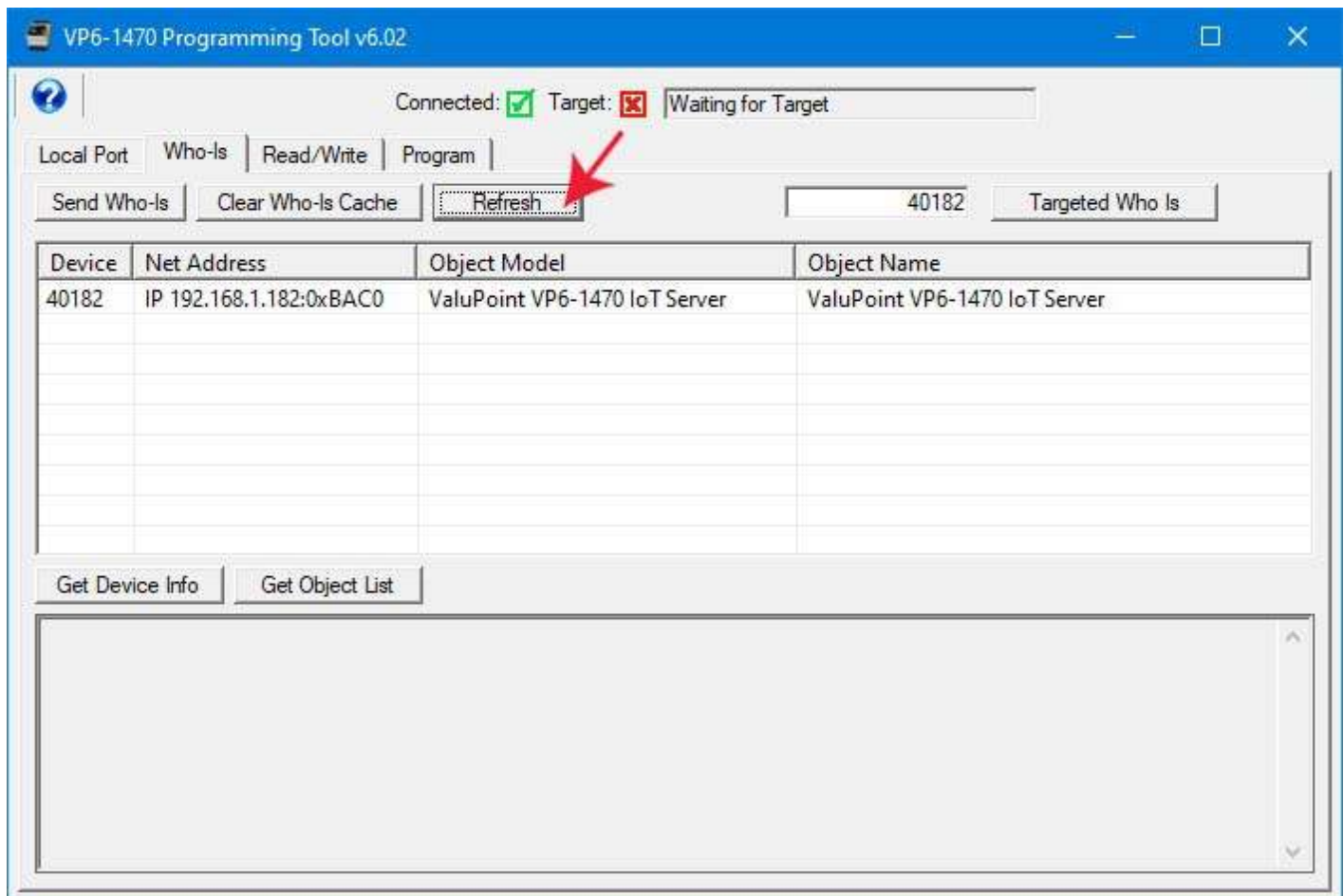
4.1 Finding Devices

Start by connecting the configuration tool to the network as outlined in section 3 of this user guide. Then come here to the Who-Is page. At first, there will be no devices listed. Click the "Send Who-Is" button to discover devices on the BACnet network.

You will end up with a long list of devices if you are connected to a busy network. To simplify the process of finding the device you are interested in, enter its Device Identifier next to the Targeted Who Is window and click that button instead of the global Send Who-Is.

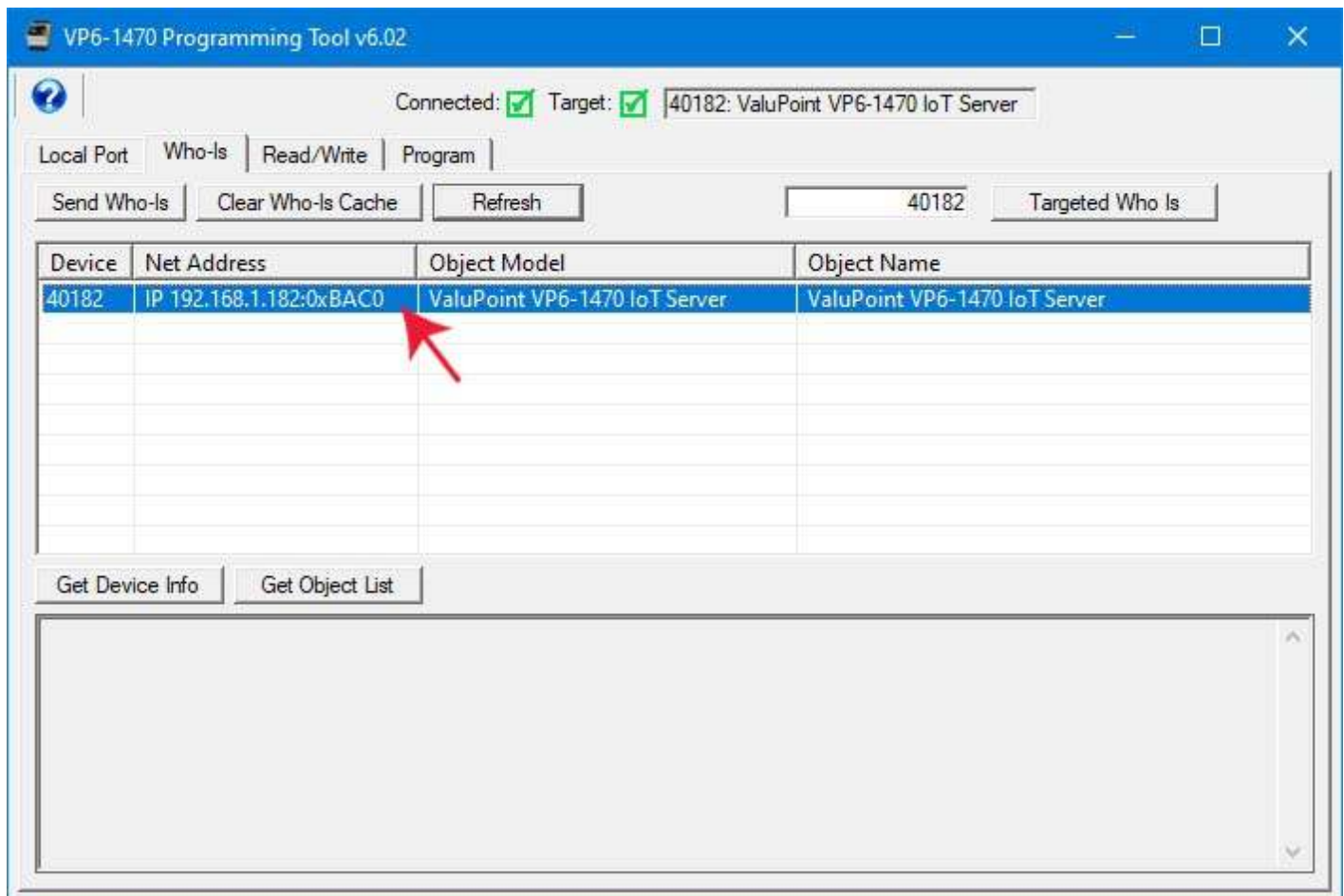


Click the Refresh button to cause the tool to request additional information from each device on the list, namely object model and object name as found in each device's BACnet Device object.



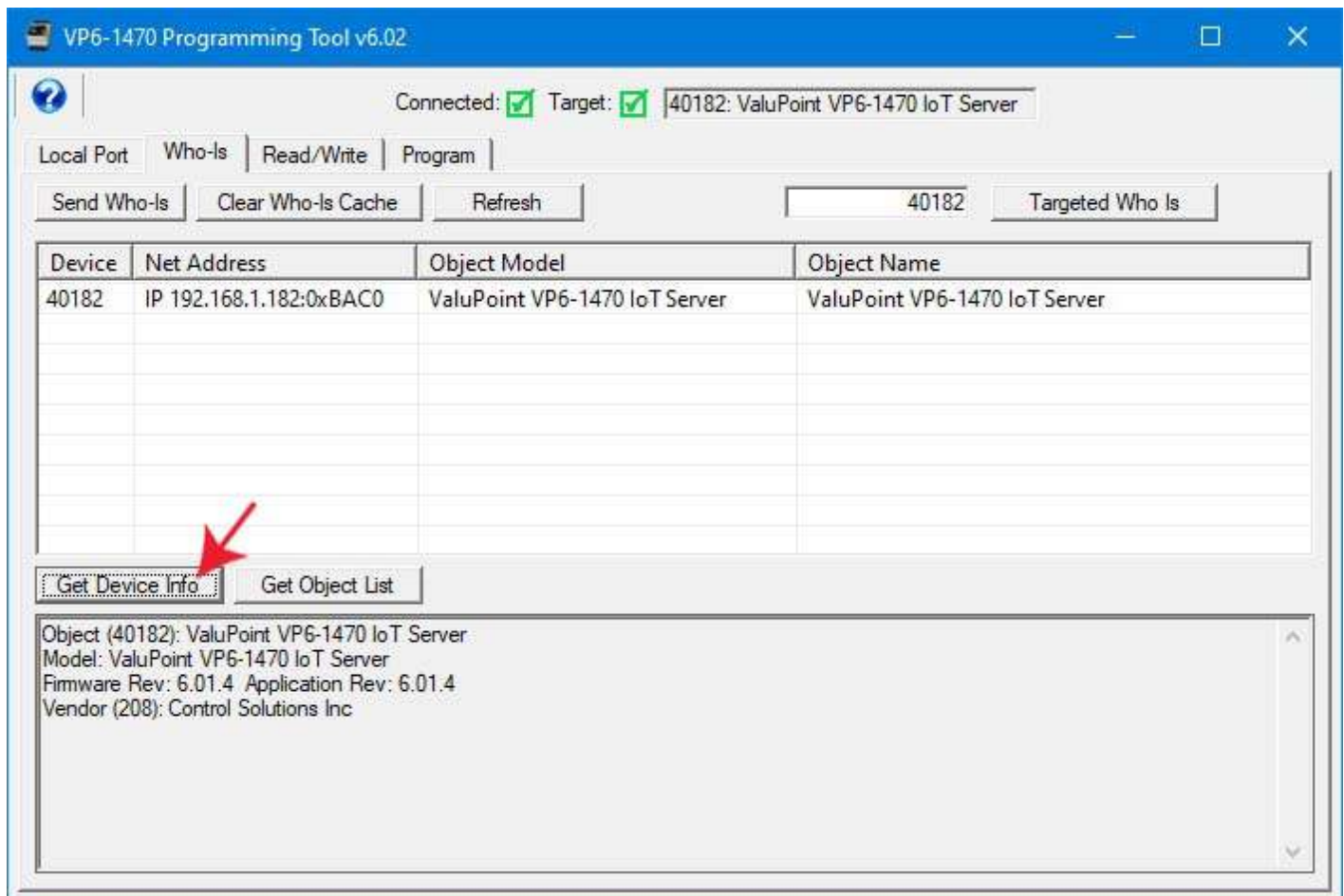
4.2 Select Target

In order to start interacting with a specific device, you need to double click that device on the list. Once selected, the icon to the right of "Target" will turn green, and the device instance and object model will appear in the target window at the top of the screen.



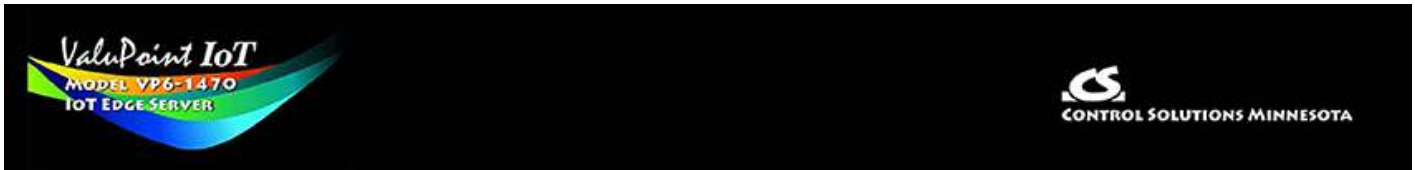
4.3 Getting Device Information

Click the Get Device Info button to get additional information about the selected device, including things like firmware revision. This step is optional and has no bearing on configuring the device.



4.4 Clear Who-Is Cache

If you want to start over on the device discovery process, click the Clear Who-Is Cache button, followed by Send Who-Is, and then Refresh after devices appear. **The only time this step becomes a requirement is if you change the device instance of the device you are in the process of configuring. If the device instance is changed, you must rediscover it under its new identity.**



5 Read/Write Page

This page may be used as soon as a device is selected on the Who-Is page. The device does not have to be a VP6-1430 in order to use the generic object property read/write on this page. This can be a useful diagnostic for any BACnet device.

5.1 Read Property

Select object type, instance, property, and array index if any, and then click 'Read Property'. If the request is successful, the data will be displayed in the log window at the bottom.

A collection of most often used property types are included in the drop list. If the desired property is not shown, select 'Other-->' and enter the property type code in the window next to the list. The property type codes are those defined by the BACnet standard. For example, Present Value can also be obtained using 'Other --> 85'.

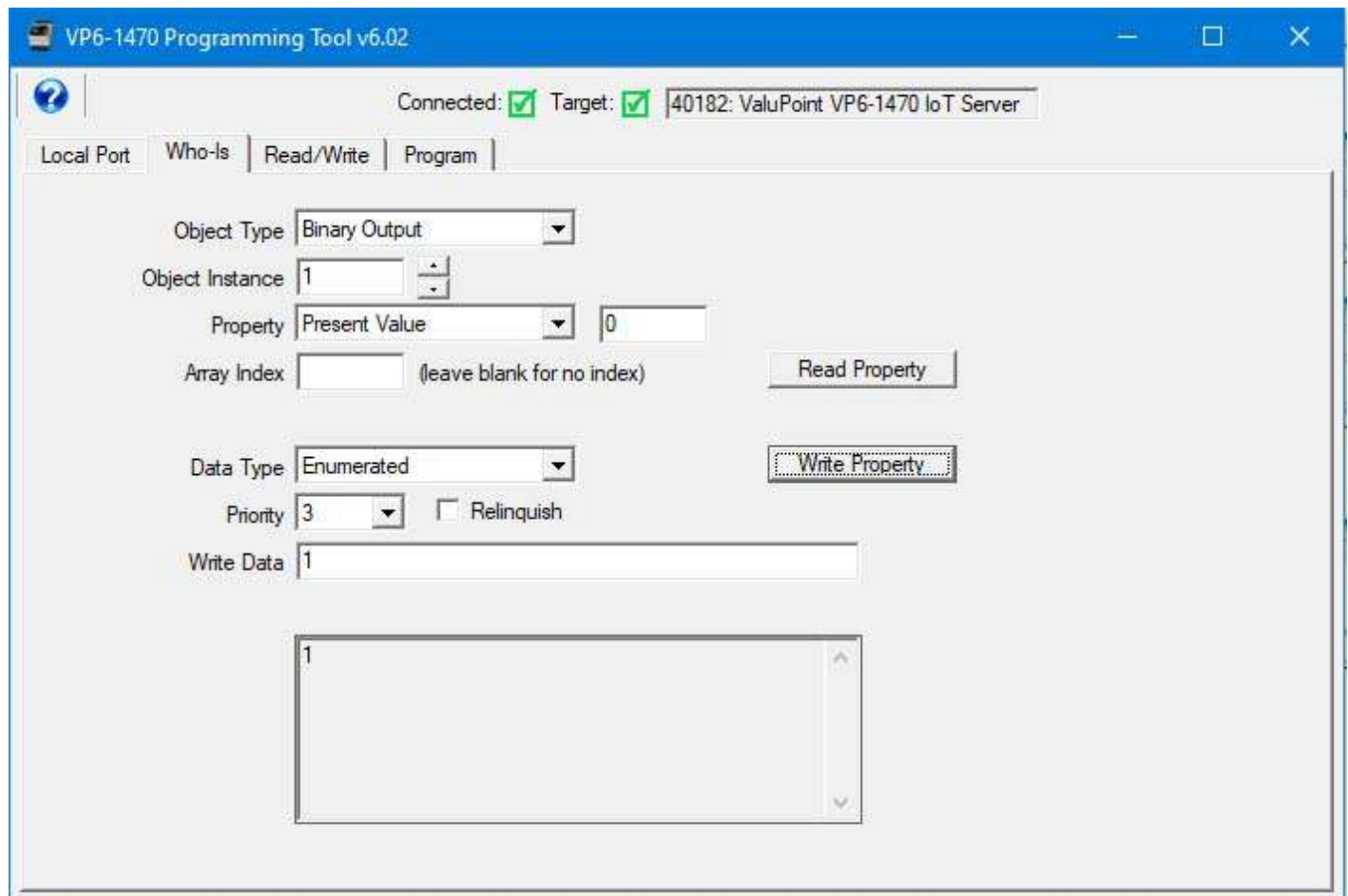
The screenshot shows the 'VP6-1470 Programming Tool v6.02' window. At the top, it indicates 'Connected: [checked] Target: [checked] 40182: ValuPoint VP6-1470 IoT Server'. Below this are four tabs: 'Local Port', 'Who-Is', 'Read/Write', and 'Program'. The 'Read/Write' tab is active. The interface includes several input fields and buttons: 'Object Type' (dropdown menu set to 'Analog Input'), 'Object Instance' (input field with '6'), 'Property' (dropdown menu set to 'Present Value'), 'Array Index' (input field with '(leave blank for no index)'), 'Data Type' (dropdown menu set to 'Null'), 'Priority' (dropdown menu set to 'None' and a 'Relinquish' checkbox), and a 'Write Data' input field. There are two buttons: 'Read Property' and 'Write Property'. At the bottom, a log window displays the value '20.000000'.

5.2 Write Property

Select all of the same settings as you would for Read Property, and in addition, specify data type, priority (if commandable, use 'none' if not commandable), and data to write. Then click 'Write Property'.

Format for most data types is simply a numeric string. The data type 'character string' will consume everything found in the data window as ASCII text copied verbatim. Octet strings should consist of 8-bit values in hexadecimal notation (1 to 2 hex digits), with each octet separated by a comma. Bit strings should consist of a series of T and F characters, optionally separated by a comma. The first T or F is bit zero.

The following is an example of writing a commandable Present Value.



The resulting data priority array can be read as illustrated below.

The screenshot shows the 'VP6-1470 Programming Tool v6.02' window. At the top, it indicates 'Connected: Target: 40182: ValuPoint VP6-1470 IoT Server'. Below this are tabs for 'Local Port', 'Who-Is', 'Read/Write', and 'Program'. The 'Read/Write' tab is active.

Configuration fields include:

- Object Type: Binary Output
- Object Instance: 1
- Property: Priority Array
- Array Index: (leave blank for no index)
- Data Type: Enumerated
- Priority: 3
- Relinquish:
- Write Data: 1

Buttons for 'Read Property' and 'Write Property' are visible. A list box at the bottom shows the current state of the Priority Array:

```
[1] NULL,  
[2] NULL,  
[3] 1,  
[4] NULL,  
[5] NULL,  
[6] NULL,  
[7] NULL,
```

To relinquish the commanded value, select the priority, and check Relinquish. Then click Write Property.

VP6-1470 Programming Tool v6.02

Connected: Target: 40182: ValuPoint VP6-1470 IoT Server

Local Port | Who-Is | Read/Write | Program

Object Type: Binary Output

Object Instance: 1

Property: Present Value 0

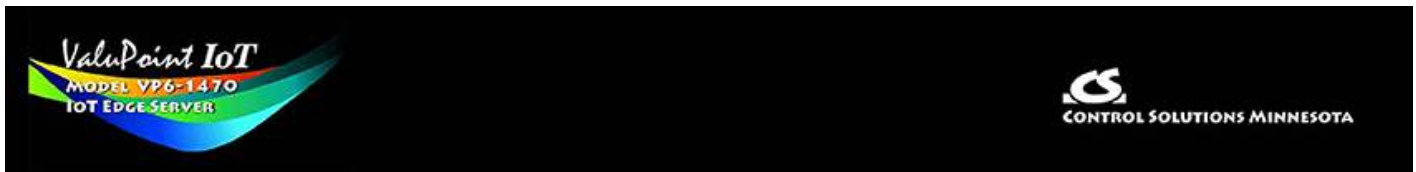
Array Index: (leave blank for no index)

Data Type: Enumerated

Priority: 3 Relinquish

Write Data: 1

NULL

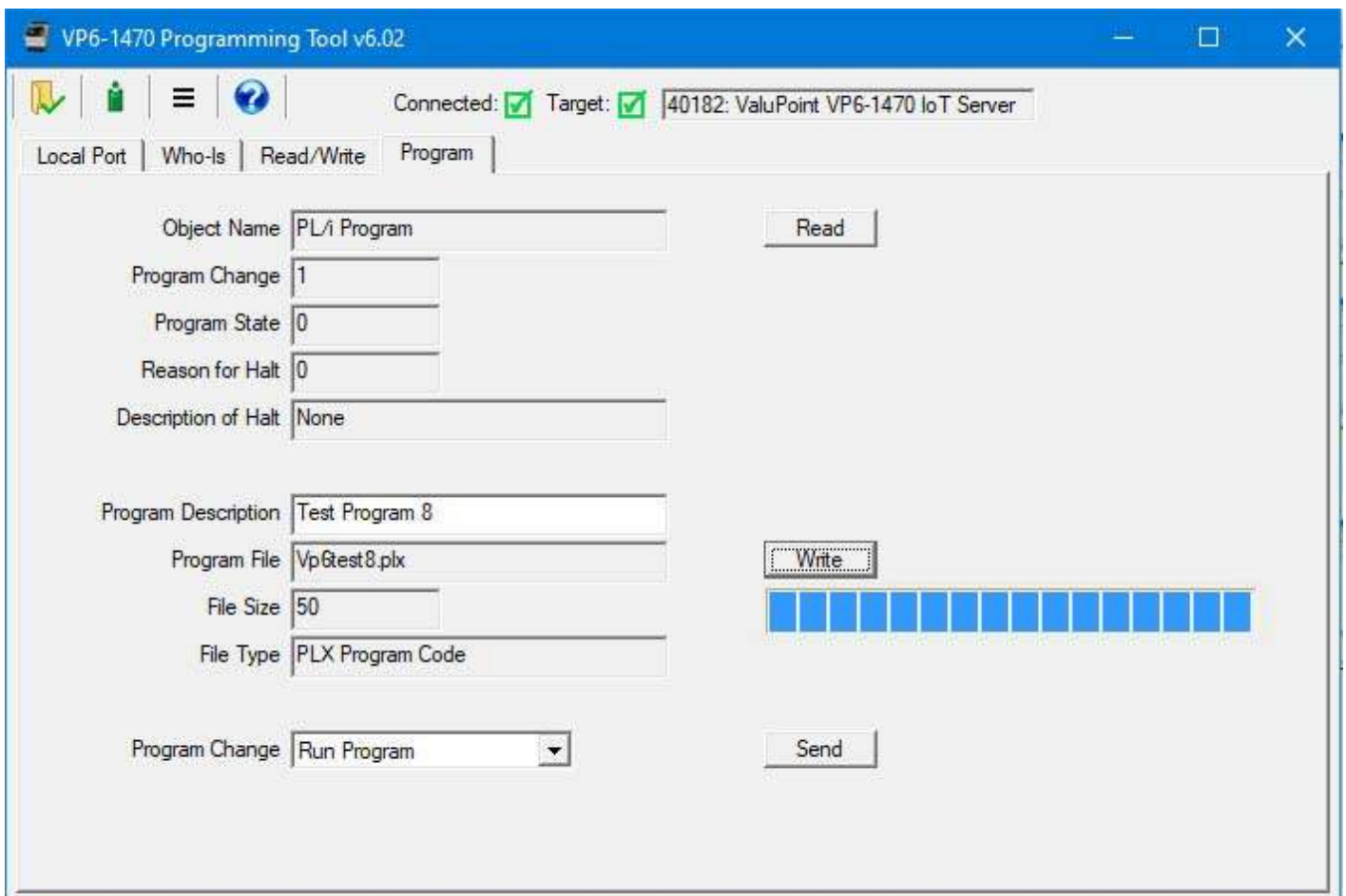


6 Programming Page

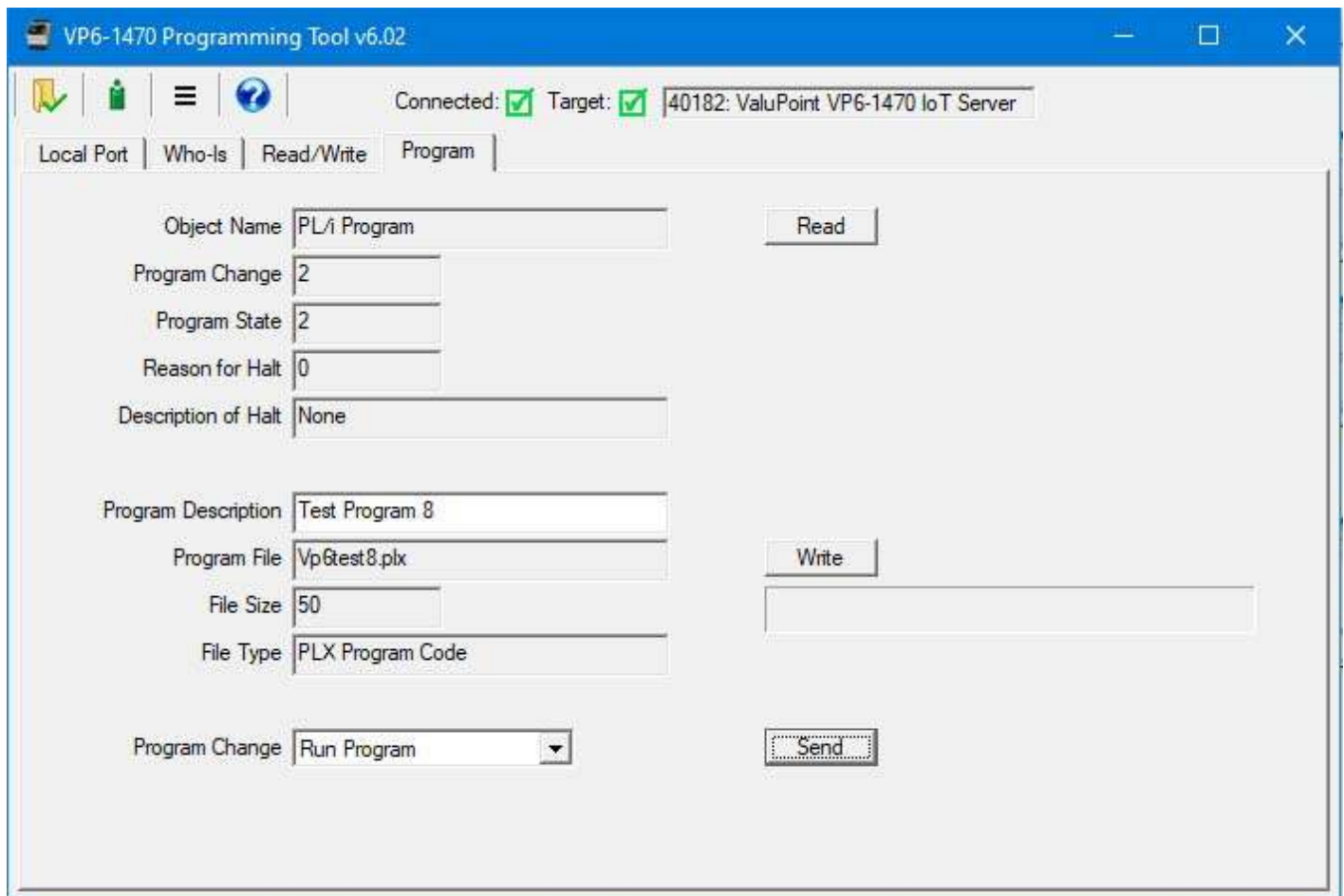
6.1 Program Loading and Execution

Click on the file folder icon at the top left to open a file. The file open dialog will appear. Select a .plx file from the list. If you do not yet have any programs compiled, you will need to use the program editing tools to create and compile a program.

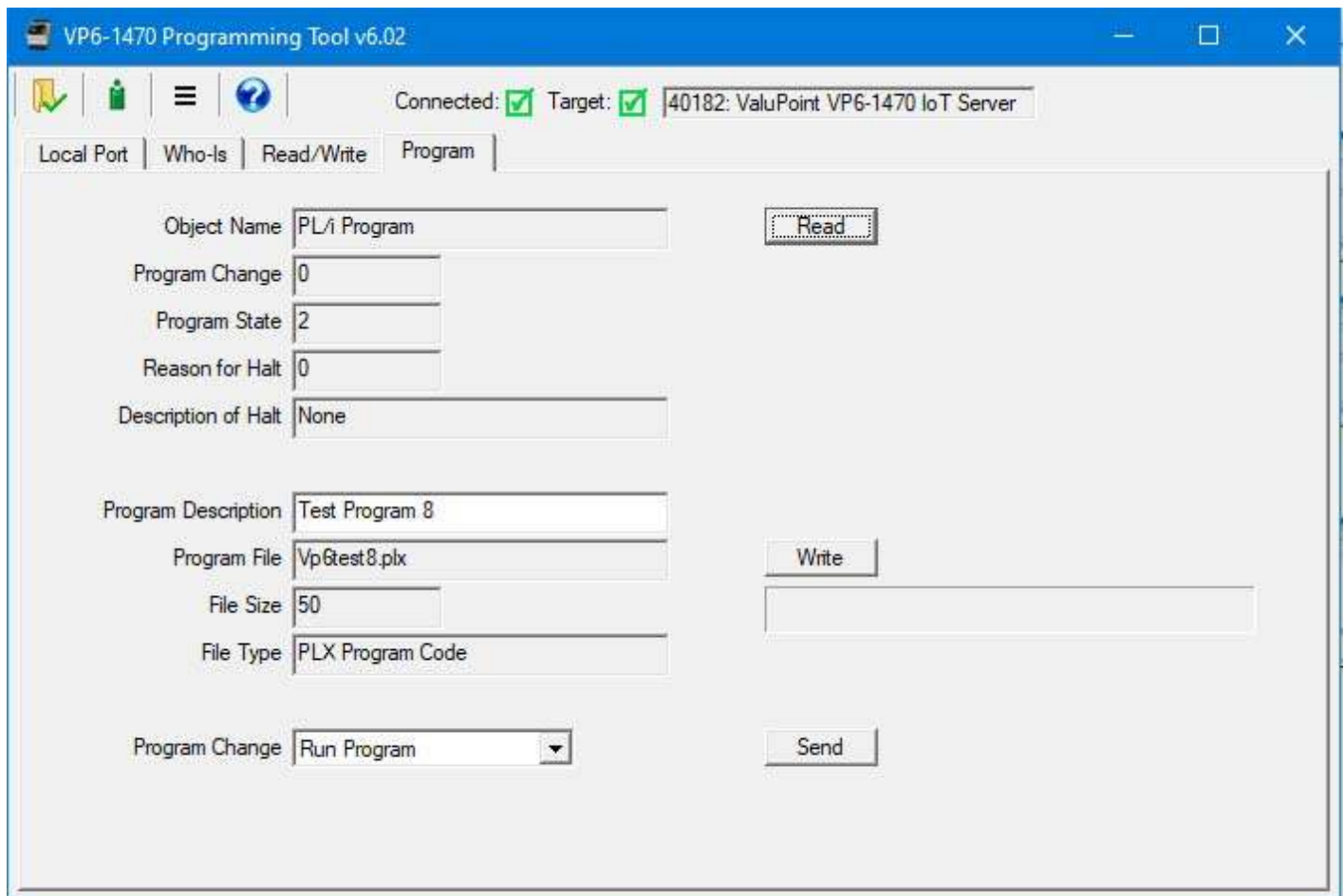
After a program (.plx file) has been opened, click the Upload button to send that program to the ValuPoint. A progress bar will indicate program loading progress.



To invoke execution of the program, select 'Run Program' from the Program Change list, and then click Request.

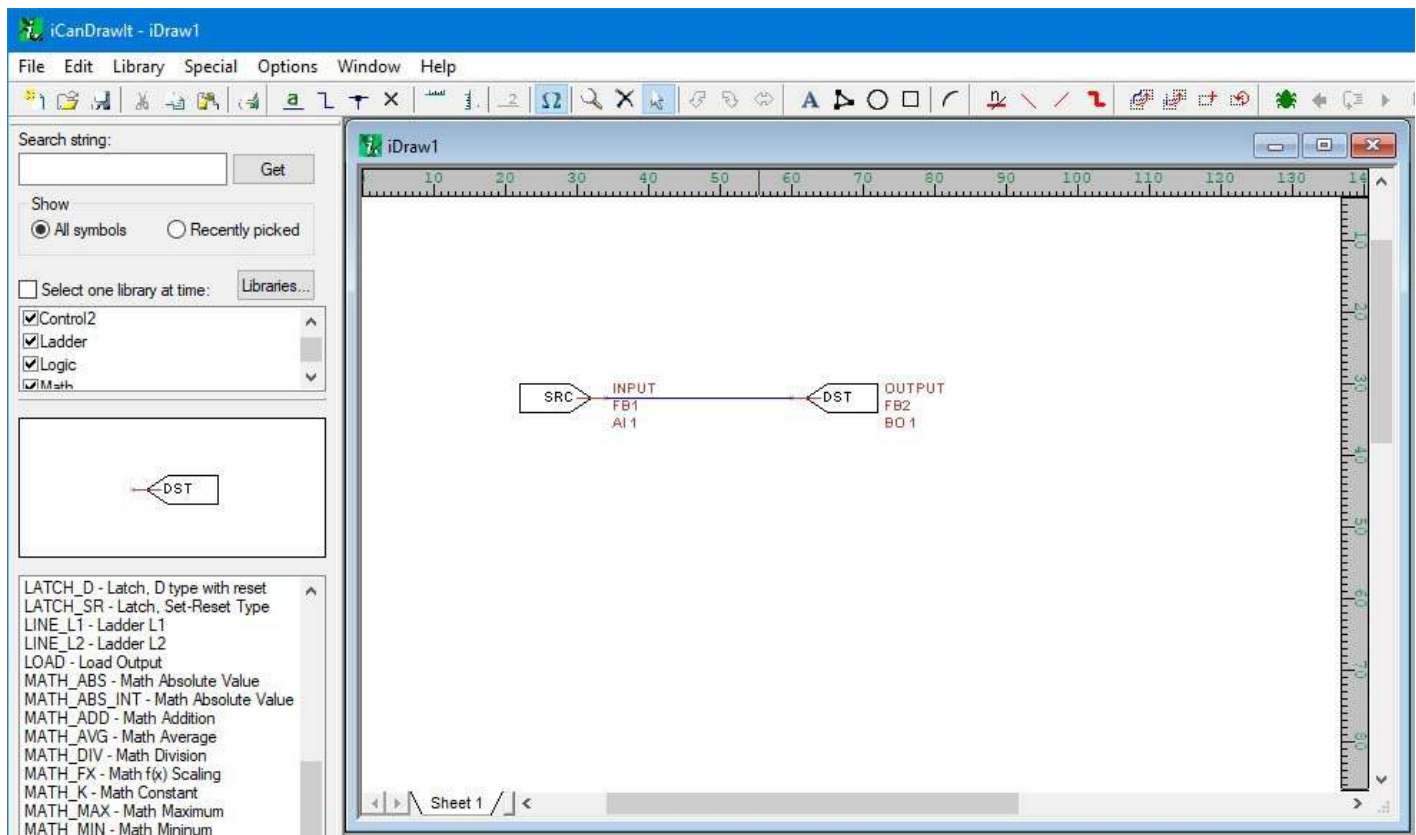


To stop the program, select Halt Program and click Request. To check the status of the program, just click Read.



If the program encounters a fatal error during execution, its error code and description of halt will be displayed after clicking the Read button.

6.2 Program Editing and Debugging



Click the green "i" icon next to the folder icon to open the i.CanDrawIt graphical programming tool that is illustrated below. It has its own set of help pages. Click on the "?" icon in that tool for more about programming. (Note that i.CanDrawIt is a second software package installed after the ValuPoint configuration tool - if you have trouble starting up i.CanDrawIt, be sure it was installed.)

Click the "line" icon next to the "?" icon to open the line programming tool. If you do not want to "draw" a program, but would rather write a program using the native PL/i programming language, you can do this. The line programming tool also has its own set of help pages.

The PL/i programming language is a derivative of PL/1 but is not the same as PL/1. The language is referred to as PL/i with "i" as in i.CanDrawIt.

The above simple example program shows operating the relay output BO #1 (relay output 1) from input AI #1 (hardware input A/UI #1). To configure A/UI #1 for use in the above example, go to the Objects page and configure A/UI #1 for dry contact (or other applicable discrete type).

6.3 Program Capacity

Maximum compiled program (.plx file) size: 65,280 bytes

Maximum RAM available for program variable and stack space: 16,384 bytes

Maximum EEPROM available: 256 bytes

6.4 Program States and Error Codes

Program State codes:

- 0 = idle
- 1 = loading
- 2 = running
- 3 = waiting
- 4 = halted
- 5 = unloading

Reason for Halt codes (Indicated as Error Code):

- 0 = no error (not halted)
- 1 = program load failed
- 64 = normal stop, end of program reached
- 65 = external stop via Program Change
- 66 = debug execution, suspended
- +n = error code (400 or above)

Non-fatal runtime errors:

- 401: subscript out of bounds, non-fatal
- 402: divide by zero, non-fatal
- 406: EEPROM address out of range, operation skipped
- 407: object instance out of bounds, operation skipped

Note: Error codes will show up as "Reason for Halt" even if the error was not necessarily fatal. This is because "Reason for Halt" is the only available standard Program Object property whose purpose is to report errors. Check the Program State to determine if the program is actually halted.

Fatal runtime errors:

- 451: unrecognized opcode, fatal
- 452: stack overflow, fatal
- 453: stack underflow, fatal
- 454: program pc out of bounds, fatal

6.5 Object and Special Register Access

BACnet objects are referenced in i.CanDrawIt programs as a calculated register number. Register numbers are BACnet object type multiplied times 1,000 plus object number starting at #1. Register numbers corresponding to BACnet objects are as follows:

Object Type	Object Number	Register Number
Analog Input	AI 1	1

Analog Output	AO 1	1001
Analog Value	AV 1	2001
Binary Input	BI 1	3001
Binary Output	BO 1	4001
Binary Value	BV 1	5001
Multi-State Input	MI 1	13001
Multi-State Output	MO 1	14001
Multi-State Value	MV 1	19001

A proprietary object type 128 is used to reference the internal real time clock.

The following special registers are available for access to the battery backed real time clock/calendar from the i.CanDrawIt program. Registers 128001-128007 will return the respective element of time as of the register read. The clock could roll over between successive reads, leading to an incorrect overall time stamp. Use the registers in the range of 128001-128007 only if you are basing an algorithm on whether day is the same as previous day, etc. To capture a complete correct timestamp, read registers 128011-128017, and be sure to read 128011 first. Reading register 128011 (year) locks the rest of the time stamp and the remaining registers will return whatever the time/date was when register 128011 was read.

To set the clock/calendar, write all of registers 128011-128017, then write any value to register 128018 to trigger the write. Nothing is done with the content of register 128018 - it is only the trigger to tell ValuPoint to store the content of registers 128011-128017 into the clock/calendar hardware.

Special Reg. No.	Writeable	Description
128001	No	Year
128002	No	Month
128003	No	Day of Month
128004	No	Hour (0..23)

128005	No	Minute
128006	No	Second
128007	No	Day of Week (1=Monday, 2=Tuesday, etc)
128011	Yes	Year (is also lock trigger for read)
128012	Yes	Month
128013	Yes	Day of Month
128014	Yes	Hour (0..23)
128015	Yes	Minute
128016	Yes	Second
128017	Yes	Day of Week (1=Monday, 2=Tuesday, etc)
128018	Yes	Lock trigger for write
128021	No	Minutes since midnight
128022	No	Day of year (Jan 1 = 1, Dec. 31 = 366 if leap year, else 365)