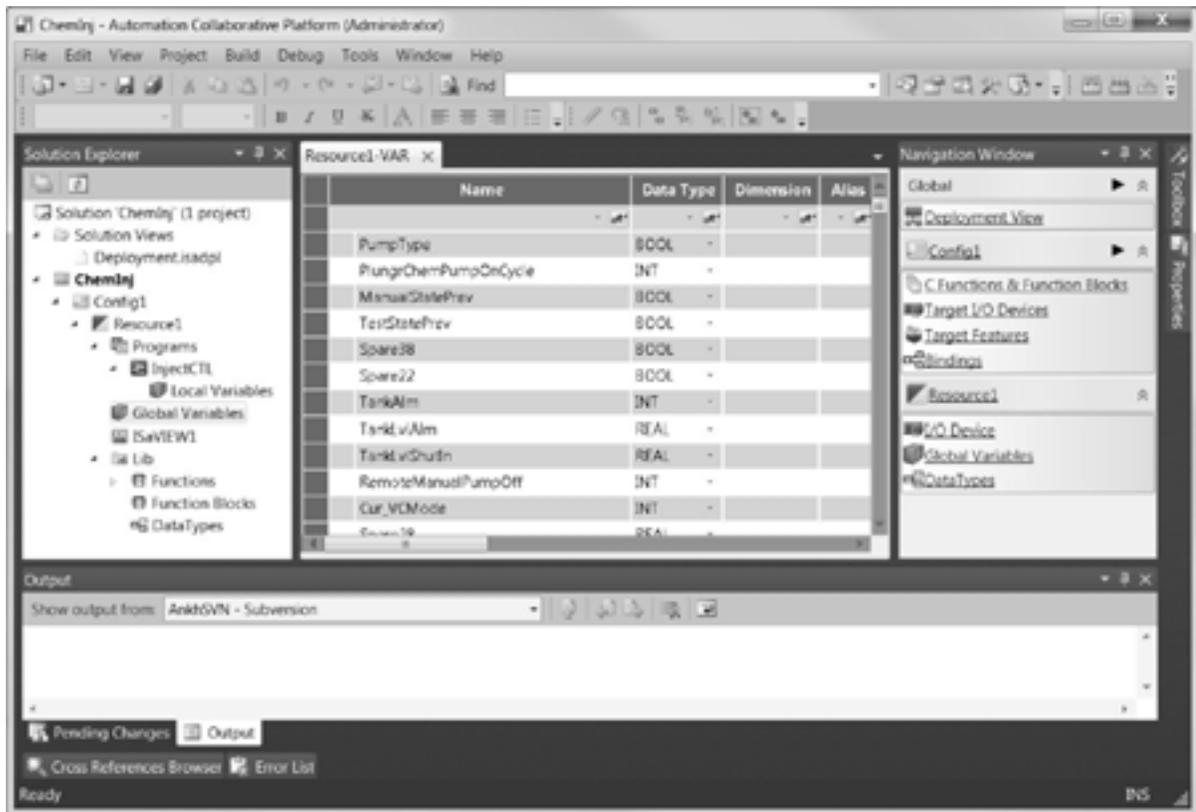


2104713 – rev. AA

# IEC-61131 Developer's Manual

ISaGRAF IEC-61131 on ABB Totalflow G4 Devices  
Application Information



## Intellectual Property & Copyright Notice

©2012 by ABB Inc., Totalflow ("Owner"), Bartlesville, Oklahoma 74006, U.S.A. All rights reserved. Any and all derivatives of, including translations thereof, shall remain the sole property of the Owner, regardless of any circumstances.

The original US English version of this manual shall be deemed the only valid version. Translated versions, in any other language, shall be maintained as accurately as possible. Should any discrepancies exist, the US English version will be considered final.

Notice: This publication is for information only. The contents are subject to change without notice and should not be construed as a commitment, representation, warranty, or guarantee of any method, product, or device by Owner.

Inquiries regarding this manual should be addressed to ABB Inc., Totalflow Products, Technical Communications, 7051 Industrial Blvd., Bartlesville, Oklahoma 74006, U.S.A.

## TABLE OF CONTENTS

<b>INTRODUCTION .....</b>	<b>VII</b>
Organization & Style.....	vii
Chapter Descriptions.....	vii
Getting Help .....	vii
Before Calling .....	vii
Key Symbols .....	vii
<b>1.0 IEC-61131 INTRODUCTION.....</b>	<b>1-1</b>
1.1 Overview.....	1-1
<b>2.0 ISAGRAF INTEGRATED DEVELOPMENT ENVIRONMENT (IDE).....</b>	<b>2-1</b>
2.1 Overview.....	2-1
<b>3.0 SETUP .....</b>	<b>3-1</b>
3.1 Creating a New ISaGRAF Project .....	3-1
3.1.1 Importing the Totalflow.tdb File .....	3-3
3.1.2 Establishing Target Parameter.....	3-4
3.1.3 Establishing the Resources for the PLC.....	3-4
3.2 Configuring I/O Registers Between the IEC Application and the G4.....	3-8
3.2.1 Step-by-Step Instructions .....	3-9
3.3 Annunciator .....	3-16
3.4 Setting Up an IP Address in the G4 Device.....	3-16
3.5 Special ABB Totalflow Variables .....	3-18
3.6 Instantiating the IEC Interface in PCCU32 .....	3-18
3.7 Building, Compiling and Downloading the IEC Application .....	3-19
3.8 Deploying the IEC Application to Other G4 Devices .....	3-20
3.9 Downloading to Other Devices .....	3-21
3.10 IEC Application and INI Files.....	3-21
3.11 Customizing the Custom INI File .....	3-23
3.11.1 Example.....	3-23

**BLANK PAGE**

## TABLE OF FIGURES

Figure 2-1 Opening the ISaGRAF Project Screen of a Pre-Existing Project .....	2-1
Figure 3-1 ISaGRAF Development Environment .....	3-1
Figure 3-2 New Project Dialog Box .....	3-2
Figure 3-3 Project Display .....	3-2
Figure 3-4 Import Totalflowl.tdb Definition File.....	3-3
Figure 3-5 Properties Window .....	3-4
Figure 3-6 Resource Properties Window .....	3-5
Figure 3-7 Code Section - Properties.....	3-5
Figure 3-8 Compiler Options Section - Properties.....	3-6
Figure 3-9 Hardware Section - Properties.....	3-6
Figure 3-10 Settings Section – Properties.....	3-7
Figure 3-11 Hardware Architecture .....	3-7
Figure 3-12 ISaGRAF to Totalflow Register Flow Diagram .....	3-9
Figure 3-13 ISaGRAF Variables.....	3-9
Figure 3-14 Solution Explorer - ExampleProg.....	3-9
Figure 3-15 Global Variables.....	3-10
Figure 3-16 _SYSCYCLECNT – Insert New Record.....	3-10
Figure 3-17 New Variable.....	3-11
Figure 3-18 New Variable Inputs.....	3-11
Figure 3-19 Navigation Window .....	3-11
Figure 3-20 Device Selector .....	3-12
Figure 3-21 Resource1 I/O Window .....	3-12
Figure 3-22 Variable Selector.....	3-13
Figure 3-23 I/O Parameters REAL_Inputs Dialog Box.....	3-13
Figure 3-24 Final Resource1 I/O Device Definition.....	3-14
Figure 3-25 Global Variables – Resource1-VAR.....	3-14
Figure 3-26 Register Pointers within the ABB Totalflow Device as Viewed by PCCU32.....	3-15
Figure 3-27 Resource1-VAR Tab.....	3-15
Figure 3-28 System Setup Dialog Box .....	3-17
Figure 3-29 Network Tab.....	3-18
Figure 3-30 Downloading to the Target Device.....	3-19
Figure 3-31 Download Dialog Box.....	3-19
Figure 3-32 Upload Packages.....	3-20
Figure 3-33 Save As Dialog Box .....	3-21
Figure 3-34 INI File .....	3-22
Figure 3-35 Gen INI G4 Dialog Box .....	3-22
Figure 3-36 PCCU32 View of INI File.....	3-23

**BLANK PAGE**

## LIST OF TABLES

Table 3-1 IEC Application Register Numbers.....	3-8
Table 3-2 ISaGRAF I/O Code.....	3-8
Table 3-3 Annunciators .....	3-16

**BLANK PAGE**

# Introduction

This manual is written to provide information regarding IEC-61131 and ISaGRAF. Along with the relevant information pertaining to these two components, the manual will also offer details about setting up IEC-61131 and ISaGRAF.

## Organization & Style

Each of the chapters in this manual presents information in an organized and concise manner. Readers are able to look at the headings and receive a broad summary of the content without reading every word. Also, there are overviews at the beginning of each chapter that provides the user with an idea of what is in the chapter and how it fits into the overall manual.

## Chapter Descriptions

Chapter	Name	Description
1	Introduction	Provides a description of IEC-61131.
2	ISaGRAF Environment	Provides a description of the ISaGRAF integrated development environment.
3	Setup	Provides the user with a description on how to set up the various components.

## Getting Help

ABB Totalflow takes pride in the ongoing support provided to customers. When purchasing a product, the user receives documentation which should answer their questions; however, ABB Totalflow Technical Support provides an 800 number as an added source of information.

If requiring assistance, call:

USA: (800) 442-3097

International: 001-918-338-4888

### Before Calling

- Know the ABB Totalflow model and serial number. Serial numbers can be found on a plate located on each unit.
- Be prepared to give the customer service representative a detailed description of the problem.
- Note any alarms or messages as they appear.
- Prepare a written description of the problem.
- Know the software version, board and optional part numbers.

## Key Symbols

The following symbols are used frequently in the manual. These are intended to catch the eye and draw attention to important information.

- FYI**  Intended to draw attention to useful information or to clarify a statement made earlier.
- TIP**  Intended to draw attention to a fact that may be useful or helpful in understanding a concept.



Intended to draw attention to a statement that might keep the user from making a mistake, keep the user from destroying equipment or parts or keep the user from creating a situation that could cause personal injury if caution is not used.



Intended to draw attention to a statement regarding the likelihood of personal injury or fatality that could result from improper access or techniques used while working in hazardous locations.

## 1.0 IEC-61131 INTRODUCTION

### 1.1 Overview

In recent years, the process and instrumentation markets have experienced unprecedented growth. Programmable logic controllers (PLC) have provided the backbone for this growth. With so many PLC manufacturers supplying equipment to the industry, the different types of programming languages and techniques that are used have had a tendency to create a certain level of confusion. Additionally, the usage of different controllers has also created communication problems from controller to controller.

IEC-61131 was developed by the International Electrotechnical Commission (IEC) to provide a generic programming environment for the PLC industry. In conjunction with ISaGRAF, ABB Totalflow has adapted their IEC compiler to work with the ABB Totalflow G4 (XSeries<sup>G4</sup>) product line. This alliance has greatly enhanced the functionality of the XSeries<sup>G4</sup> products by providing the end user with the tools with which to build custom applications.

Different user applications may require different programming solutions. One application may lend itself to a graphical solution (Function Block Diagram) while another program may be best addressed using ladder logic (Ladder Diagram). The IEC-61131 solution supports five (5) different programming tools:

- ST – Structured Text
- FBD – Function Block Diagram
- IL – Instruction List
- LD – Ladder Diagram
- SFC – Sequential Function Chart

ISaGRAF's approach to IEC-61131 supports an additional programming tool: Flow Chart (FC) language. Using one of the above programming tools or using them in combination should provide the user with enough flexibility to address any special application problems that their field environment might present.

ABB Totalflow's XFC<sup>G4</sup> and XRC<sup>G4</sup> devices can run multiple IEC-61131 applications on a single application credit. The IEC-61131 licensing is on a "per box" basis. A single IEC-61131 credit allows multiple IEC-61131 applications to run on a single G4 unit.

- One IEC-61131 Interface Turned On = One ISaGRAF Resource
- One IEC-61131 application credit supports multiple instances of the IEC-61131 interface

These application credits can be applied to the flow computer at the factory or by purchasing a credit key (secure flash drive) with the appropriate credits installed.

**BLANK PAGE**

## 2.0 ISaGRAF INTEGRATED DEVELOPMENT ENVIRONMENT (IDE)

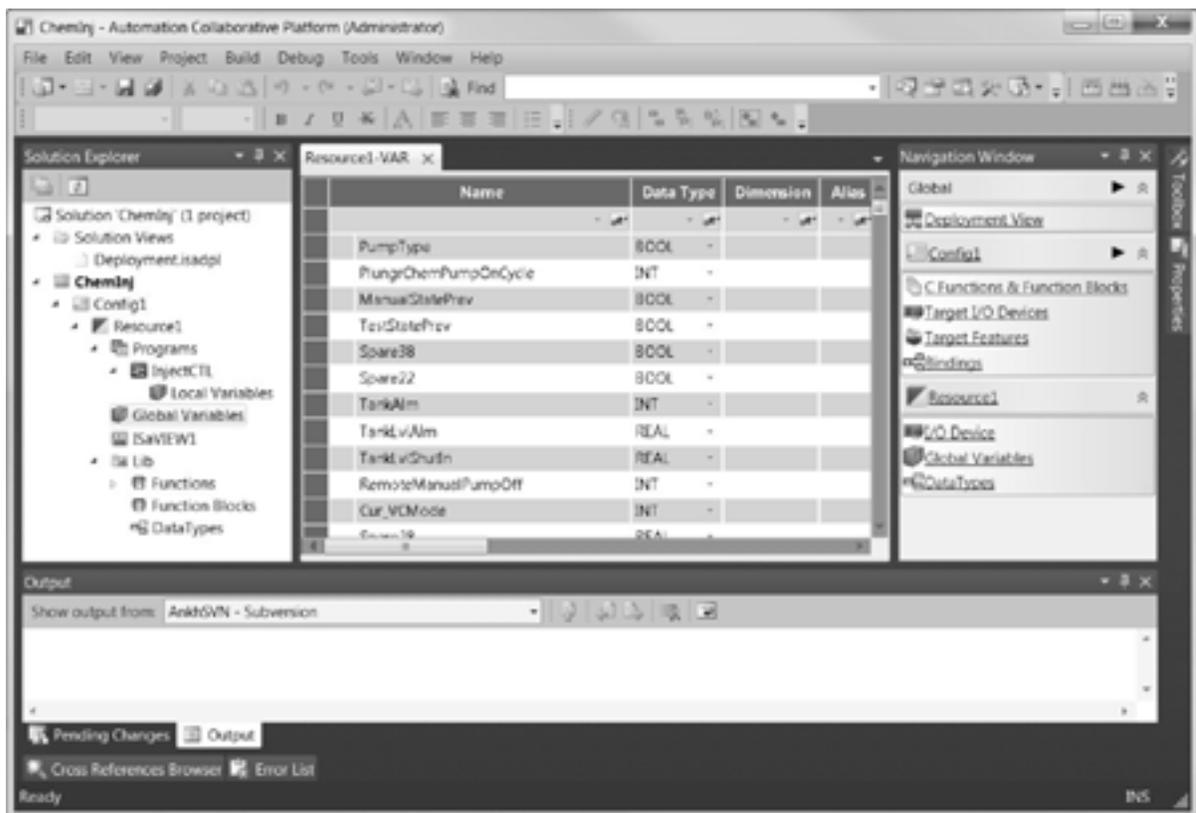
### 2.1 Overview

Upon its initial introduction in 1990, ISaGRAF has become the industry standard for the implementation of IEC-61131. Its design is intended to provide a programming tool that can be used in the PLC and/or microcomputer world. ISaGRAF was designed by ICS Triplex, a Rockwell Automation company.

The following documentation will not attempt to discuss the ISaGRAF environment beyond its interface to ABB Totalflow products. As such, the following topics are discussed:

- Importing the ABB Totalflow library into ISaGRAF (Totalflow.tdb).
- Setting up an IP connection in ISaGRAF (allowing communication with an ABB Totalflow device).
- Downloading (via a 32-bit loader) the IEC-61131 application to the ABB Totalflow device.
- Targeting the G4 device for debugging the IEC-61131 application via ISaGRAF (debugging details are covered in the ISaGRAF documentation).

The following displays the overall view of the ISaGRAF screen after opening a pre-existing project (see Figure 2-1):



**Figure 2-1 Opening the ISaGRAF Project Screen of a Pre-Existing Project**

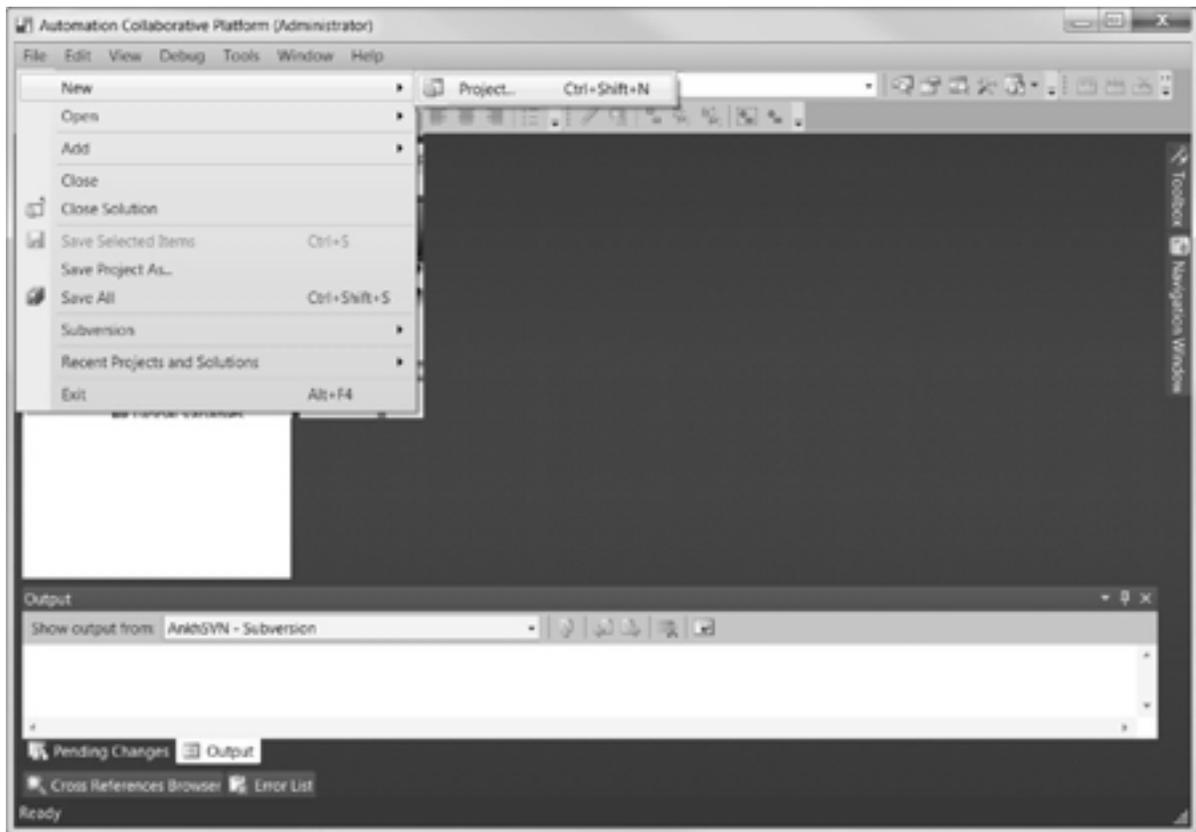
**BLANK PAGE**

## 3.0 SETUP

### 3.1 Creating a New ISaGRAF Project

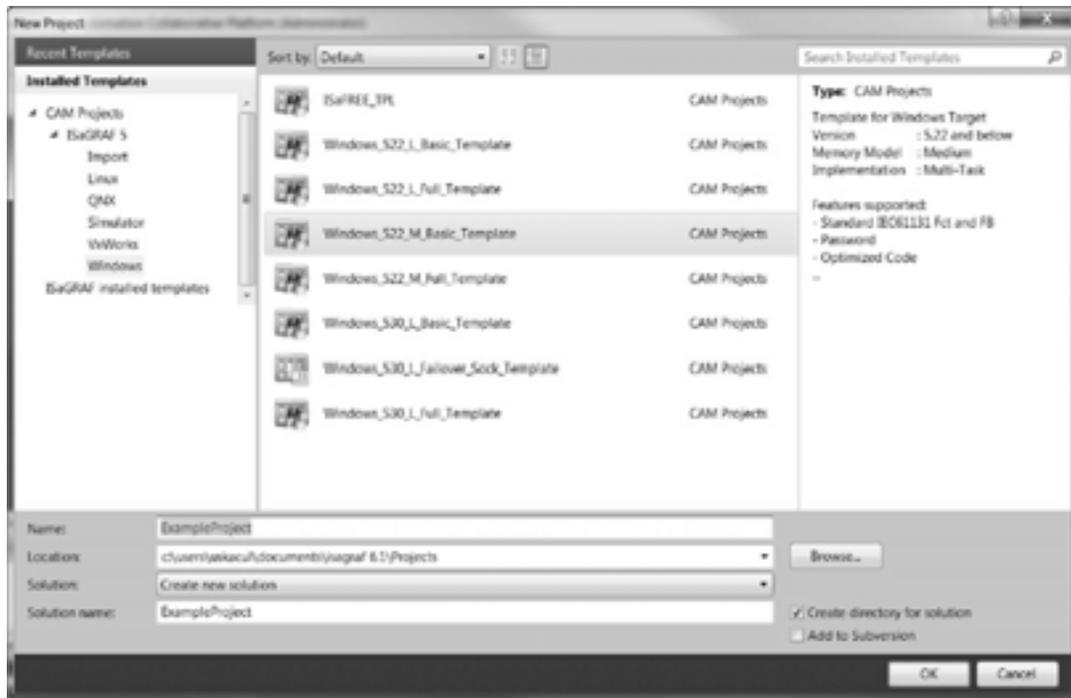
The following information details the procedures necessary to use ISaGRAF.

- 1) To create an IEC-61131 project, open ISaGRAF Development Environment (IDE), and click on File.
- 2) A drop-down menu displays. From the selections, choose New. From the New fly-out menu, select Project (Figure 3-1).



**Figure 3-1 ISaGRAF Development Environment**

- 3) The New Project dialog box displays. Select the Windows\_552\_M\_Basic\_Template from the list of available options. Once the template is selected, browse to the preferred folder in which to save the project. Upon completion, click the OK button (see Figure 3-2).



**Figure 3-2 New Project Dialog Box**

The project that was just created will display as follows (Figure 3-3):

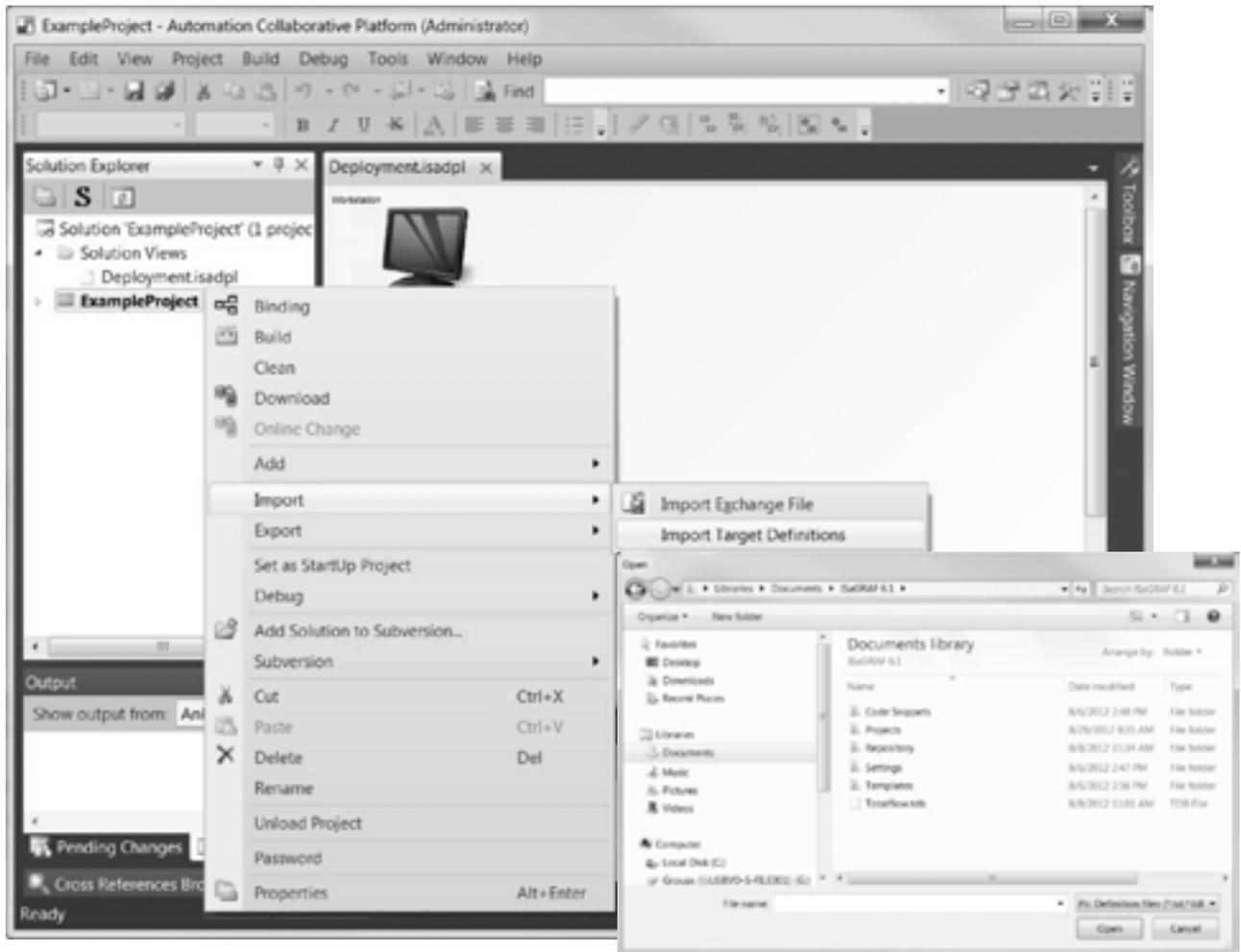


**Figure 3-3 Project Display**

### 3.1.1 Importing the Totalflow.tdb File

A PLC definition now needs to be imported. ABB Totalflow provides a definition file (Totalflow.tdb) that enables the ISaGRAF environment to target the ABB Totalflow G4 devices (XFC<sup>G4</sup>/XRC<sup>G4</sup>).

- 1) From the ISaGRAF Solution Explorer window, right-click on the project that was just created in the following section. From the pop-up menu, select Import.
- 2) From the Import fly-out menu, select Import Target Definitions (see Figure 3-4). An Open dialog box displays. Navigate to where the Totalflow.tdb file is located. Upon location, select it.



**Figure 3-4 Import Totalflow.tdb Definition File**

**FYI**



The Totalflow.tdb file provides functions to read and write to the various data types available in G4 devices. The definition file also defines the Microsoft CE OS, the processor type, memory mapping and the overall target device environment.

'Get' and 'Set' functions support some of the following data types:

- GetREAL – 32-bit floating point

- SetREAL – 32-bit floating point
- GetDINT – Double signed integer (32-bit)
- SetDINT – Double signed integer (32-bit)
- GetUDINT – Double unsigned integer (32-bit)
- SetUDINT – Double unsigned integer (32-bit)
- GetSTRING – Character string (64 bytes)
- SetSTRING – Character string (64 bytes)

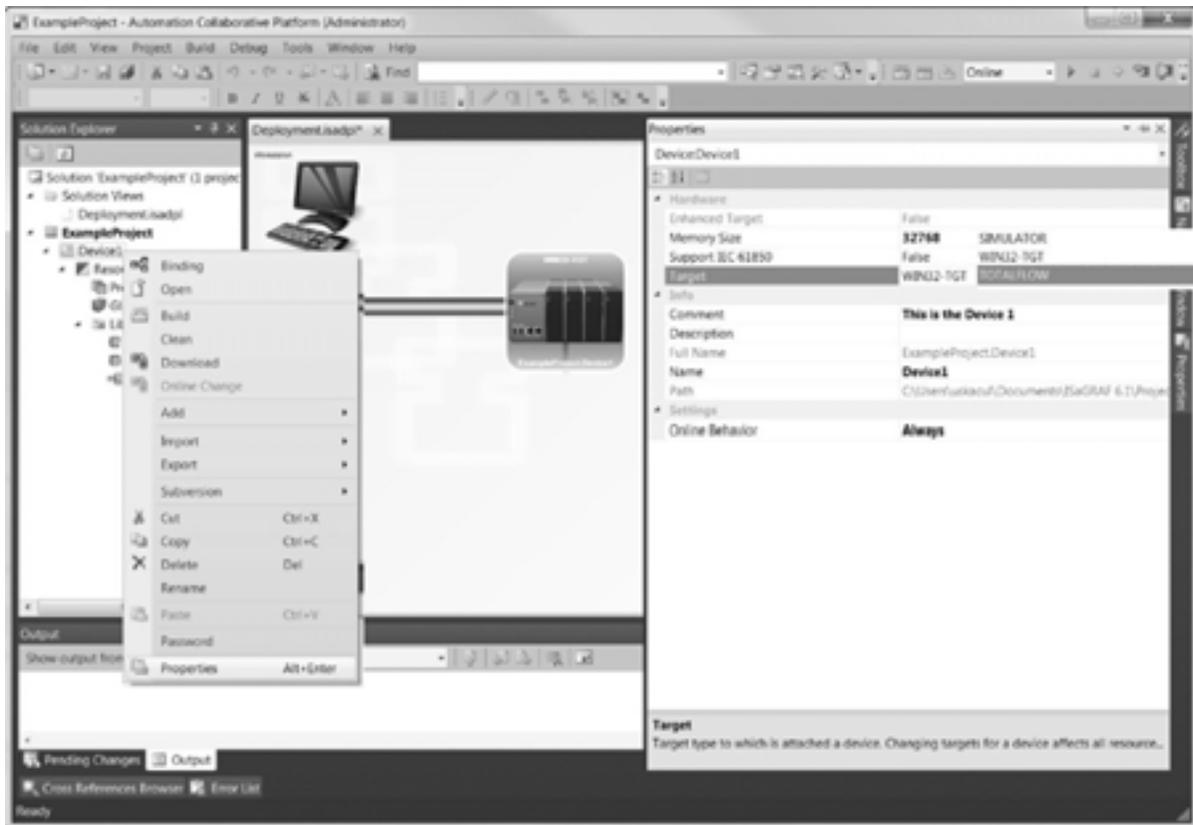


These routines are not recommended. Use I/O points for input/output. See Section 3.2.

### 3.1.2 Establishing Target Parameter

The following details the parameters that must be set in the Properties window.

- 1) Once the Totalflow.tdb is installed, collapse the project (i.e., ExampleProject), and right-click on Device1. From the pop-up menu, select Properties to reveal the Properties window.
- 2) In the Target parameter field, change the default to 'TOTALFLOW' (see Figure 3-5).

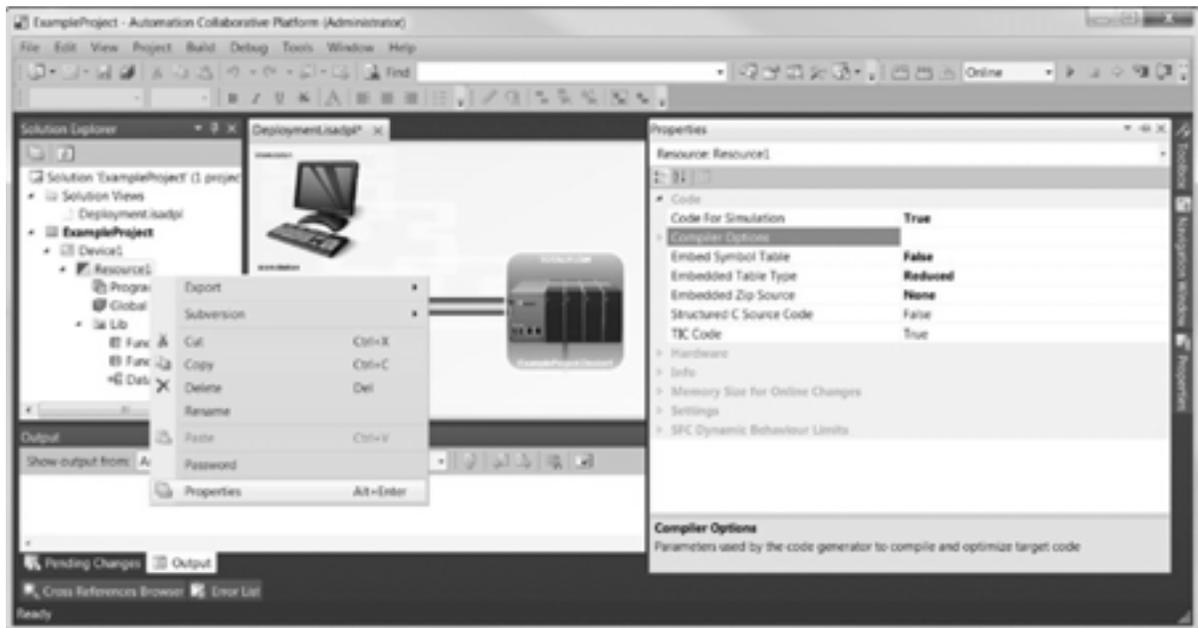


**Figure 3-5 Properties Window**

### 3.1.3 Establishing the Resources for the PLC

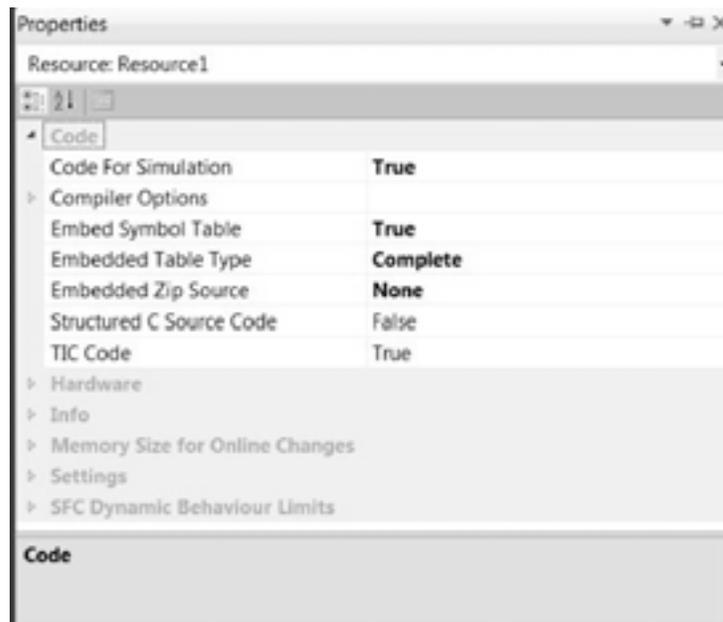
The following details setting up the Resource properties for the PLC.

- 1) Collapse the menus for Device1. Upon completion, right-click Resource1. From the pop-up menu, select Properties to reveal the Resource1 Properties window (see Figure 3-6).



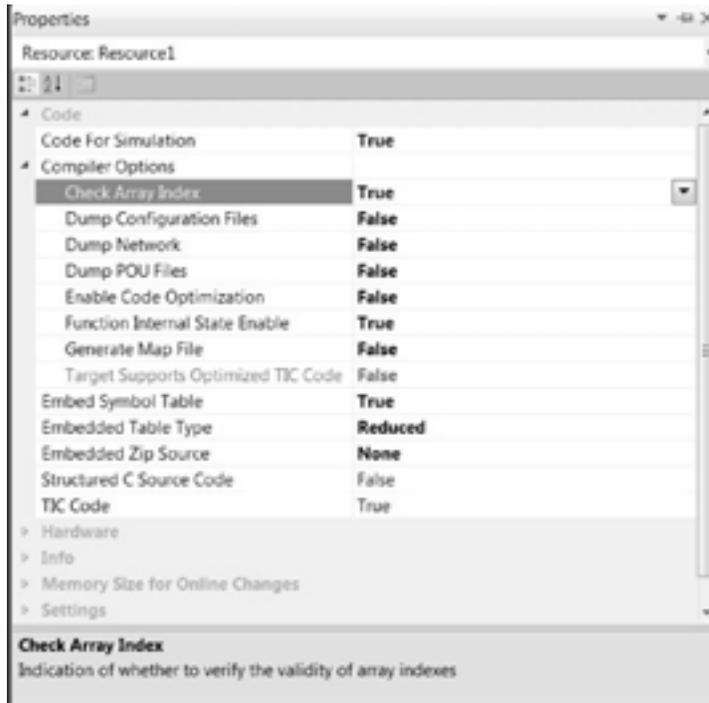
**Figure 3-6 Resource Properties Window**

- 2) In the Resource1 Properties window, within the Code section, change the Embed Symbol Table parameter field to 'True' (see Figure 3-7).
- 3) Next, change the Embedded Table Type to 'Complete' (see Figure 3-7).



**Figure 3-7 Code Section - Properties**

- 4) Upon completion, expand the Compiler Options section, and change the Check Array Index to 'True' (see Figure 3-8).



**Figure 3-8 Compiler Options Section - Properties**

- 5) While still in the Properties window, expand the Hardware section. Once expanded, change the Target parameter field to 'TOTALFLOW' (see Figure 3-9).



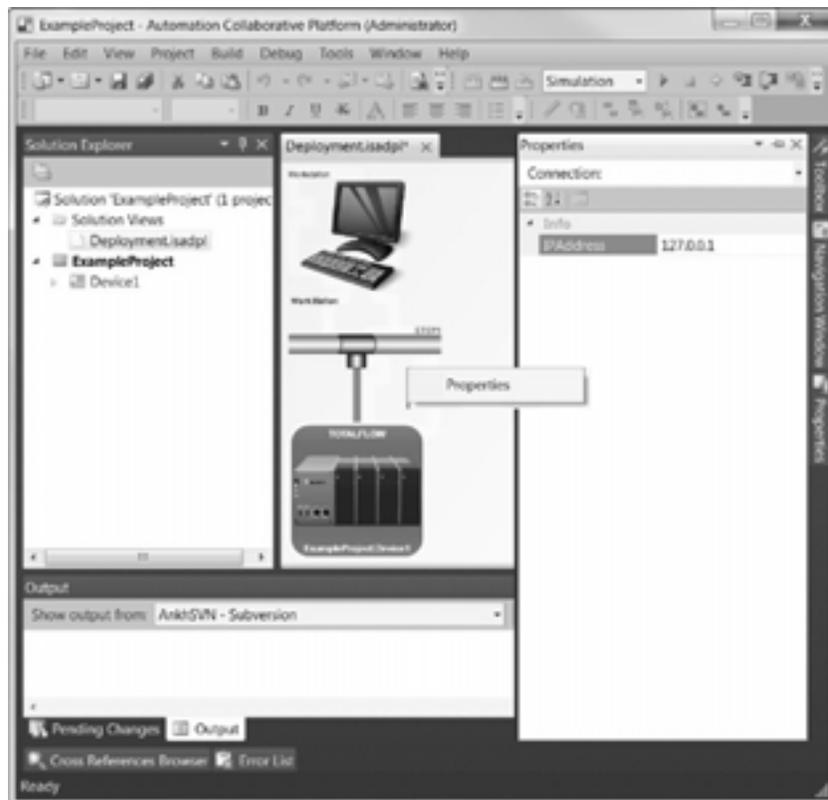
**Figure 3-9 Hardware Section - Properties**

- 6) While still in the Properties window, expand the Settings section. Upon expansion, change the Cycle Time parameter field to '1000' (see Figure 3-10).
- 7) While in the Settings section, change the 'Memory for Retain' to "\tfData\iec-%\Sample", replacing "Sample", with the name of the ini file, i.e., "25900010-001".



**Figure 3-10 Settings Section – Properties**

- 8) Once the user has completed setting up the parameter fields within the Properties window, return to the Solution Views. Once in Solution Views, right-click the vertical bar above the TOTALFLOW icon. From the pop-up menu, select Properties.
- 9) Within the Properties window, the user needs to change the IP address. The user should change the IP address to the network IP address of the intended G4 unit (see Figure 3-11).



**Figure 3-11 Hardware Architecture**

### 3.2 Configuring I/O Registers Between the IEC Application and the G4

The following section discusses defining the IEC registers for input and output data. There are three (3) options in which to use a variable:

- Internal use
- Read/Write value only
- Register address at the PCCU32 level (user-configurable)

The following information attempts to summarize the register address relationship (see Table 3-1):

- **IEC Variable Name** – Static Pressure (SP)
- **IEC User Address** – x9012 hex combination of Totalflow Array \* 1000 hex, (fixed by type) and Register Index, last three digits (hex). This value only needs to be set to ‘lock down’ the Totalflow Register address. If the variable is ‘internal only’ and not used in the INI or another application, the user may leave this field blank. This can also be left blank during development and instead use the automatic INI display.

**Table 3-1 IEC Application Register Numbers**

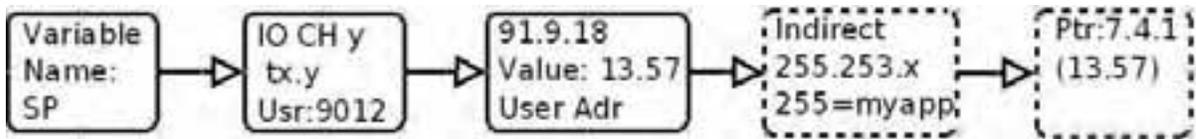
Type	Array	Type	Array	Type	Array
Bool	0	Int32	5	Float	9
Int8	1	UInt32	6	Double	10
UInt8	2	Int32	7	String	11
Int16	3	UInt32	8	Time	12
UInt16	4			Date	13
				Register	253

- **IEC IO Address** – **tdx.y**: **t** is the data type, **d** is the data direction, **x** is the I/O device number and **y** is the channel. This value is automatically set when wiring the I/O in the ISaGRAF environment (see Table 3-2).

**Table 3-2 ISaGRAF I/O Code**

Direction	Code	Type	Code
INPUT	I	Udint/Dint	D
OUTPUT	Q	Real	R
		String	S

- **Totalflow Value Register** – 91.9.18: 91 is the specific application number, 9 is from Table 3-1 and 18 is set by the IEC User Address, if supplied. The address is used by the INI and other applications to set and/or read the value of the IEC variable, SP.
- **Totalflow Indirect Address Register** – 255.253.1 is the programmer-selected value that is placed in the parameter field for this I/O point. If the final register does not need to be set by the user, the Final Register can be placed here (i.e., system date).
- **Totalflow Final Register** – 7.4.1 is the register with the original value and is entered in by the user to point the actual value: AI1.
- **Totalflow Value** – 13.57.



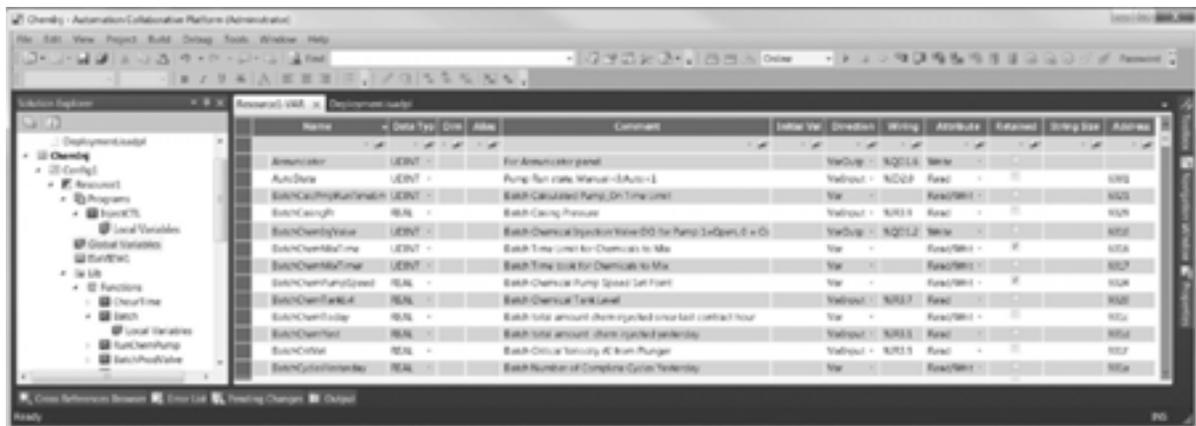
**Figure 3-12 ISAaGRAF to Totalflow Register Flow Diagram**

In Figure 3-12, SP, internal to IEC, has a value of '13.57'. Totalflow shows this value (13.57) at 91.9.18. the user specifies the location of the actual value by placing '7.4.1' into 19.253.1.

IEC runtime reads location 7.4.1 every second while running and places the value into 91.9.18. The IEC program can read this value in its internal variable SP.

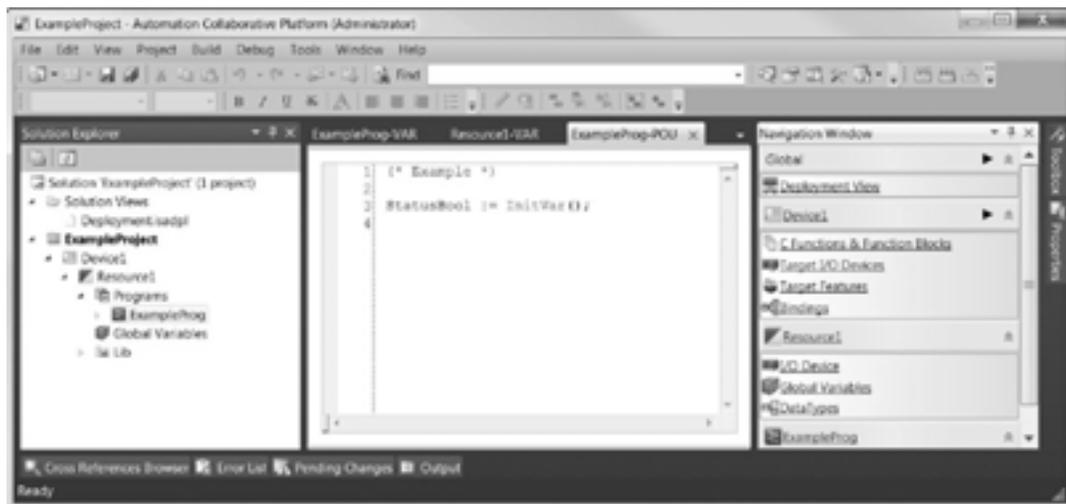
### 3.2.1 Step-by-Step Instructions

Figure 3-13 displays the available variables for a typical IEC application.



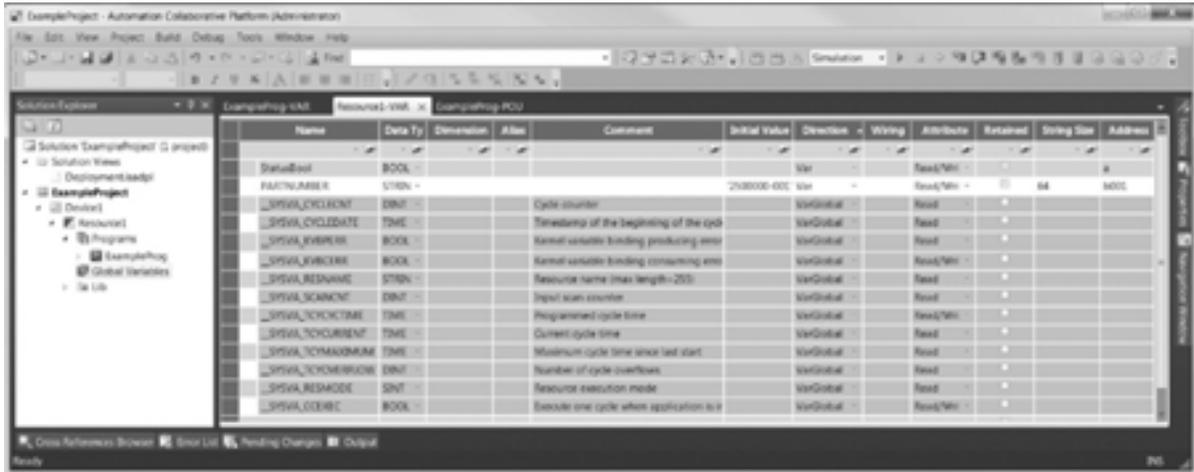
**Figure 3-13 ISAaGRAF Variables**

- 1) In the Solution Explorer section, expand the ExampleProject menu item in the tree-view. Expand the Resources1 sub-menu to reveal Programs. For the purpose of these steps, ExampleProg was added (see Figure 3-14).



**Figure 3-14 Solution Explorer - ExampleProg**

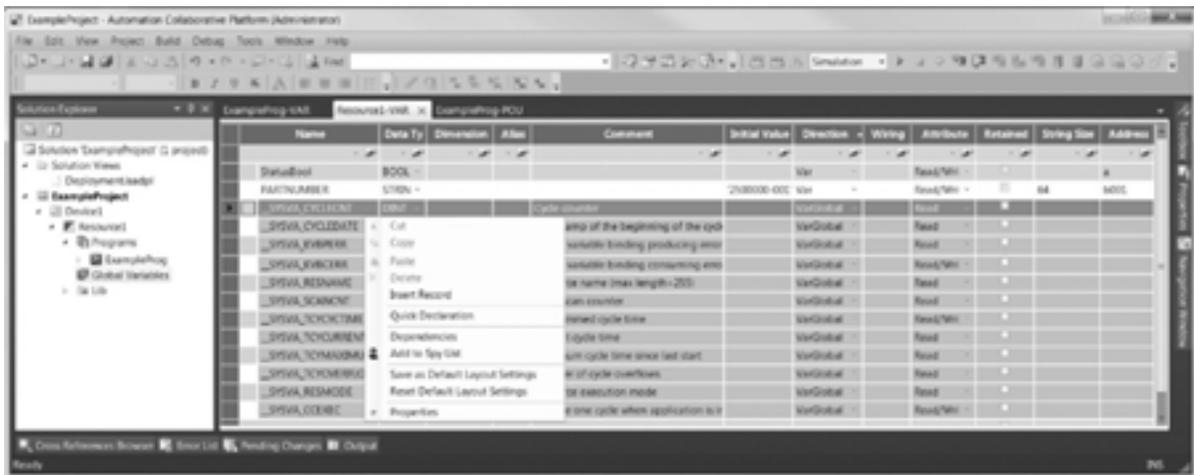
- 2) With the Device1 sub-menu expanded, double-click on Global Variables. All of the variables that are available for Resource1 will display (Figure 3-15).



**Figure 3-15 Global Variables**

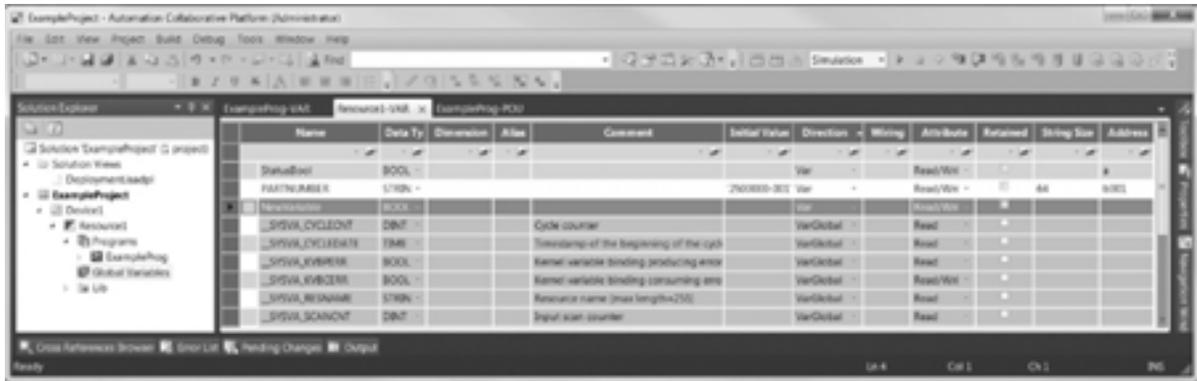
In an effort to simplify the procedures, the following instructions will show how to create the variable, static pressure (SP). This requires the definition of several properties.

- 3) In the Global Variables section, right-click on the '\_SYSVAL\_CYCLECNT'. From the pop-up menu, select Insert Record (see Figure 3-16).



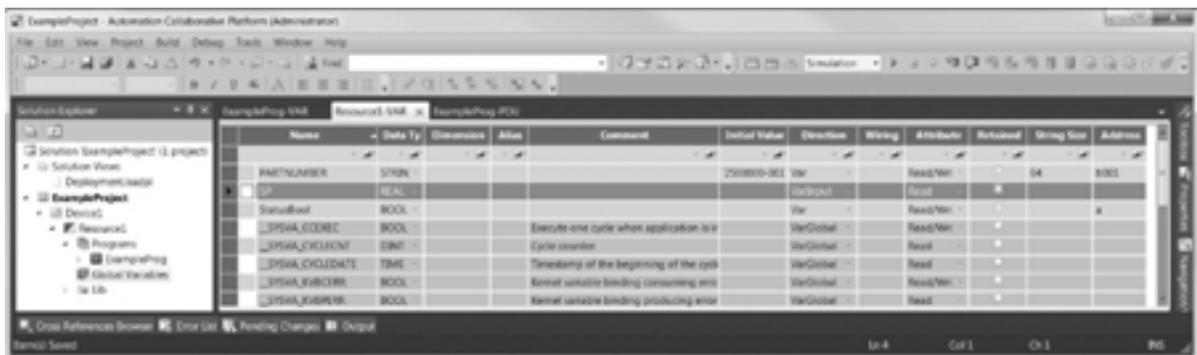
**Figure 3-16 \_SYSVAL\_CYCLECNT – Insert New Record**

- 4) Upon selection of Insert New Record, a new line for variable inputs displays (see Figure 3-17).



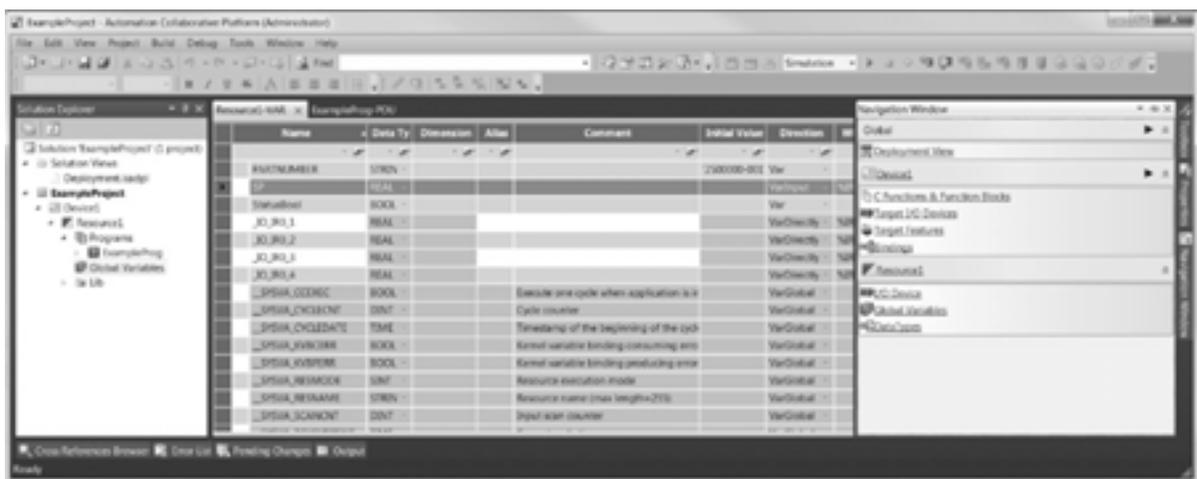
**Figure 3-17 New Variable**

- 5) Within the New Variable parameter field, type in 'SP' for the name of the variable. Next, select 'Real' for the data type, 'Read' for the attribute and 'VarInput' for the direction (see Figure 3-18).



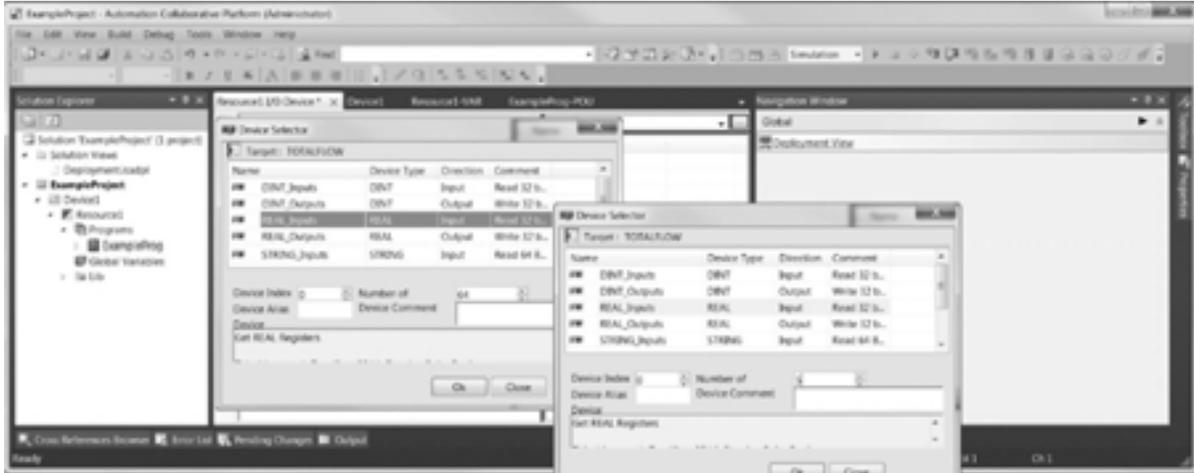
**Figure 3-18 New Variable Inputs**

- 6) After setting the various variable parameter fields, display the Navigation window. From the Navigation window, select Resource1. From the Resource1 menu item, choose I/O Device (see Figure 3-19).



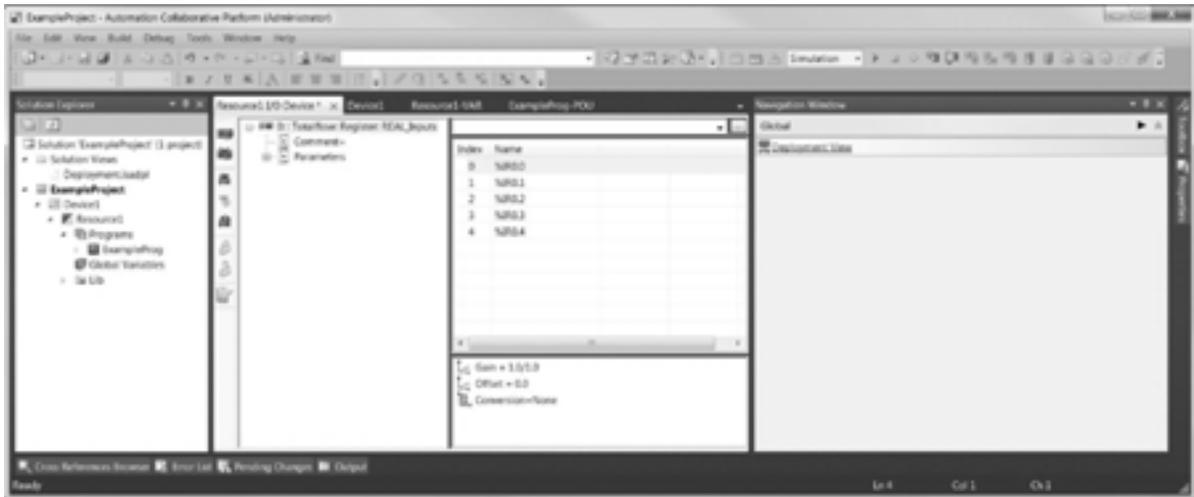
**Figure 3-19 Navigation Window**

- 7) The user now needs to click on the Device Selector icon. The Device Selector dialog box displays. In the dialog box, select REAL\_Inputs. Once completed, move to the Number of field and change the default number. For the sake of these instructions, the number is '5' (see Figure 3-20).
- 8) Click the OK button.



**Figure 3-20 Device Selector**

- 9) Real inputs are now displayed in the Resource1 I/O Device window (see Figure 3-21).



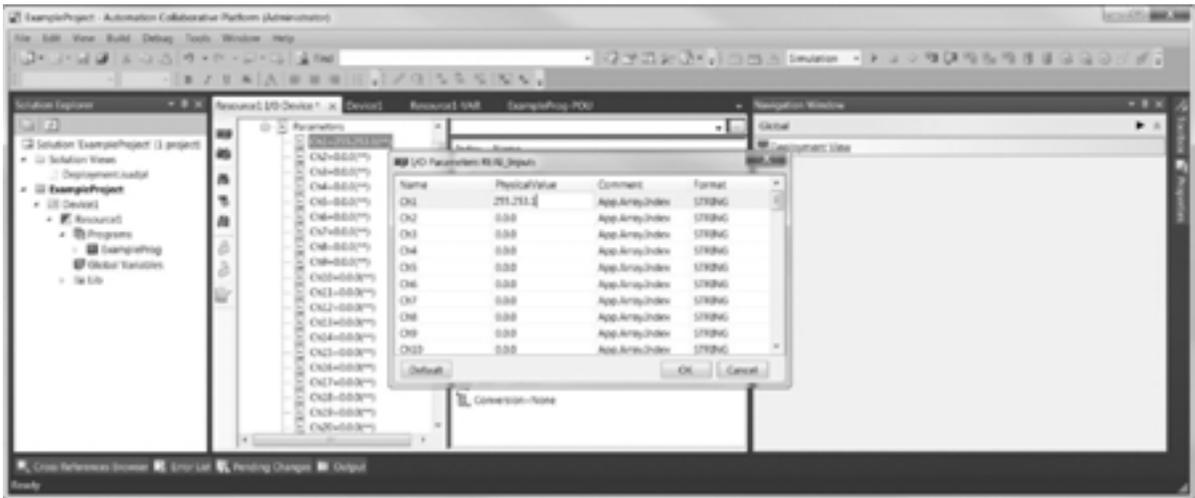
**Figure 3-21 Resource1 I/O Window**

- 10) To place the SP variable into the Resource1 I/O Device window, double-click on %IR0.0.
- 11) The Variable Selector window displays. Select the 'SP' variable, and then click the OK button (see Figure 3-22).



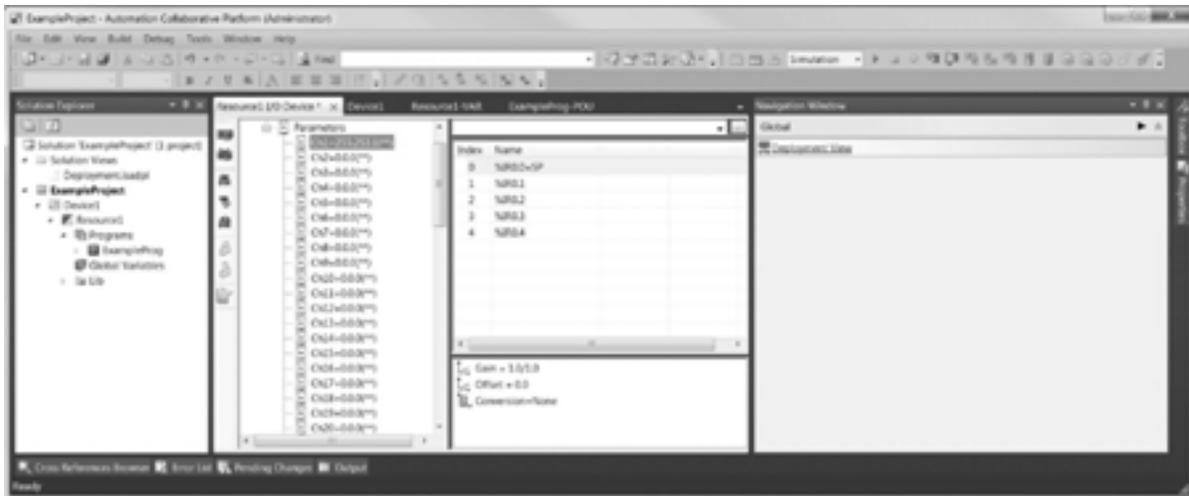
**Figure 3-22 Variable Selector**

- 12) Expand the Parameters menu item in the Resource I/O Device tree-view. Upon expansion, double-click on 'Ch1=0.0(\*\*\*)'. The I/O Parameters REAL\_Inputs dialog box displays.
- 13) Locate Ch1 in the dialog box. In the Physical Value parameter field, type in the following: 255.253.1. Once completed, click the OK button (see Figure 3-23).



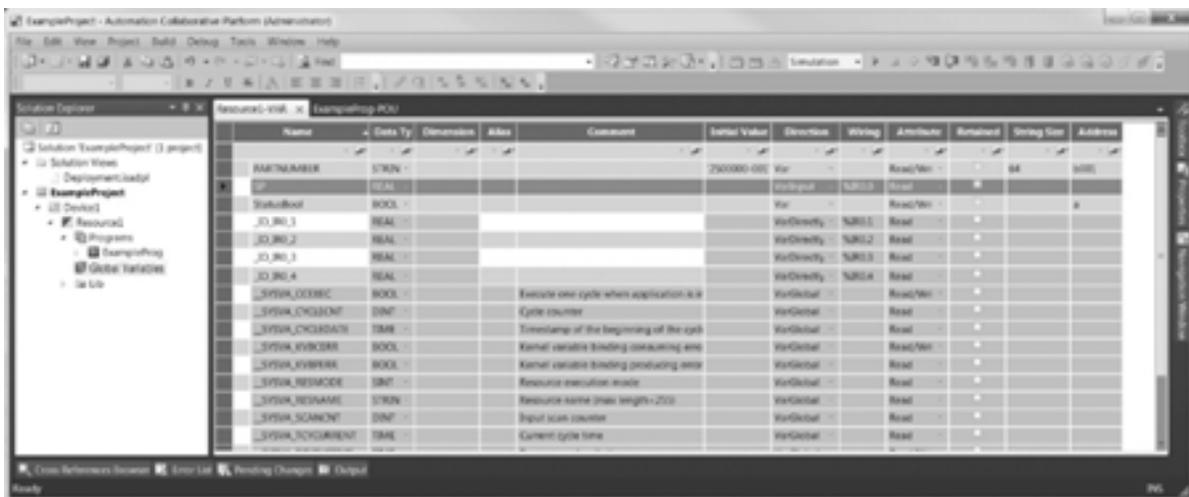
**Figure 3-23 I/O Parameters REAL\_Inputs Dialog Box**

- 14) The final Resource1 I/O Device definition is as follows (see Figure 3-24).



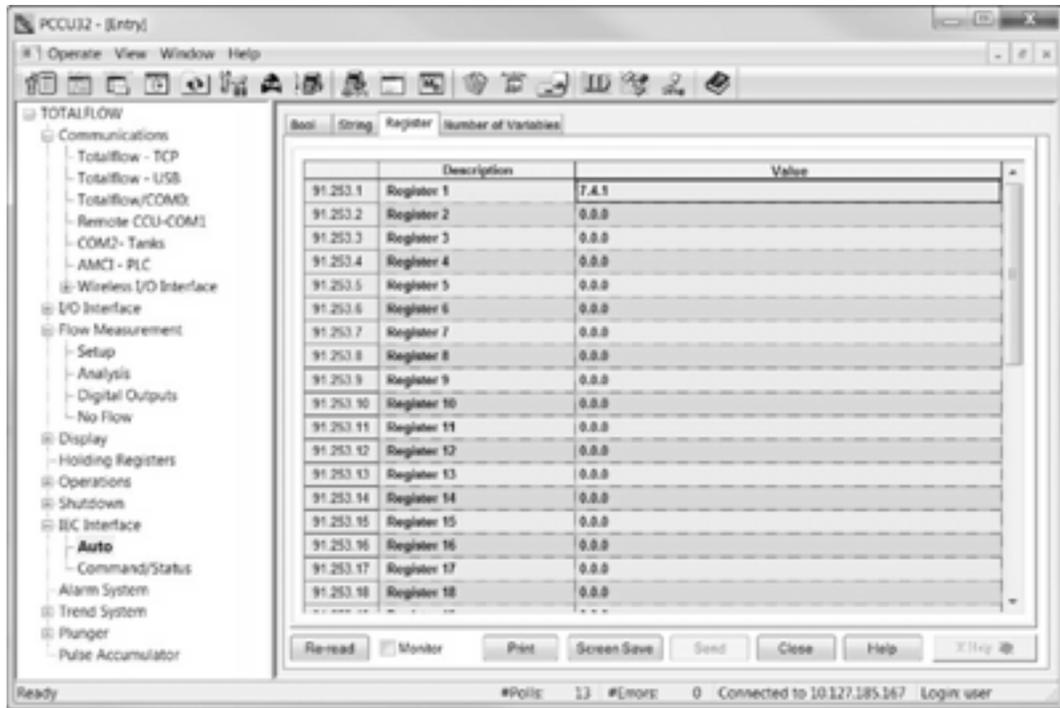
**Figure 3-24 Final Resource1 I/O Device Definition**

15) The Wiring definition also displays in the Global Variables - Resource1-VAR for static pressure (SP) (see Figure 3-25).



**Figure 3-25 Global Variables – Resource1-VAR**

Static pressure (SP) in the IEC project has now been pointed to the registers in the ABB Totalflow target device. In other words, CH1 from the ISaGRAF is used to supply the SP to the IEC application. CH1 has been pointed to the target holding register of 255.253.1. 255 is a special application number which will be changed to the application slot where the specific IEC application slot is instantiated. This subsequently gives the user application slot independence. 253 is the pre-defined array for the register type - holding registers (i.e., indirect addressing). 1 is the programmer-assigned register index for SP.



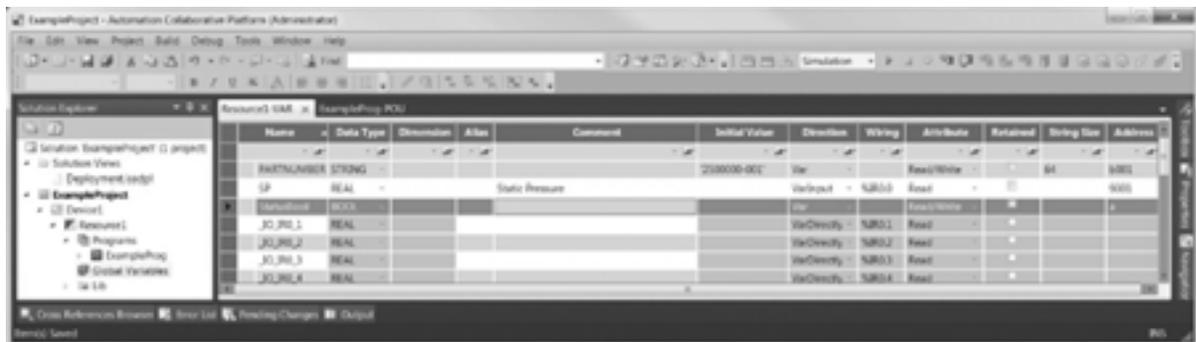
**Figure 3-26 Register Pointers within the ABB Totalflow Device as Viewed by PCCU32**

In Figure 3-26, the user can see register 91.253.1 displayed within PCCU32. The IEC application holds this SP register so that it can then access the hardware I/O in the ABB Totalflow device. Within the PCCU32 Register tab, the user should notice the Value column. The Value column holds the register for the actual hardware I/O point. The IEC application can be mapped to application slots; the recommended range is 91 through 93 in the ABB Totalflow device. In Figure 3-26, application slot 91 is used.



The ABB Totalflow target device generally defaults pressure to the hardware analog input register, 7.4.1. As such, the user should enter '7.4.1' in the Value column.

Returning to the Resource1-VAR, the user needs to set the address for SP. This usually is performed prior to downloading the IEC application to the G4 device (see Figure 3-27).



**Figure 3-27 Resource1-VAR Tab**

The address will lock the address that is seen PCCU32 and the rest of the ABB Totalflow applications. The first digit equals the array number (in hex). The remaining three digits are the register numbers. The register numbers are also in hex (i.e., 9.18 > 9012).

At this point, the user is ready to select the programming mode or tool that will be used to create the IEC application (i.e., Structured Text, Function Block Diagram, Instruction List, Ladder Diagram or Sequential Function Chart).

### 3.3 Annunciator

The status of an application can be displayed optionally on the front panel display. There is a special register referred to as app: 255.251. This is single byte value. This can be most ASCII characters or a special value from Table 3-3.

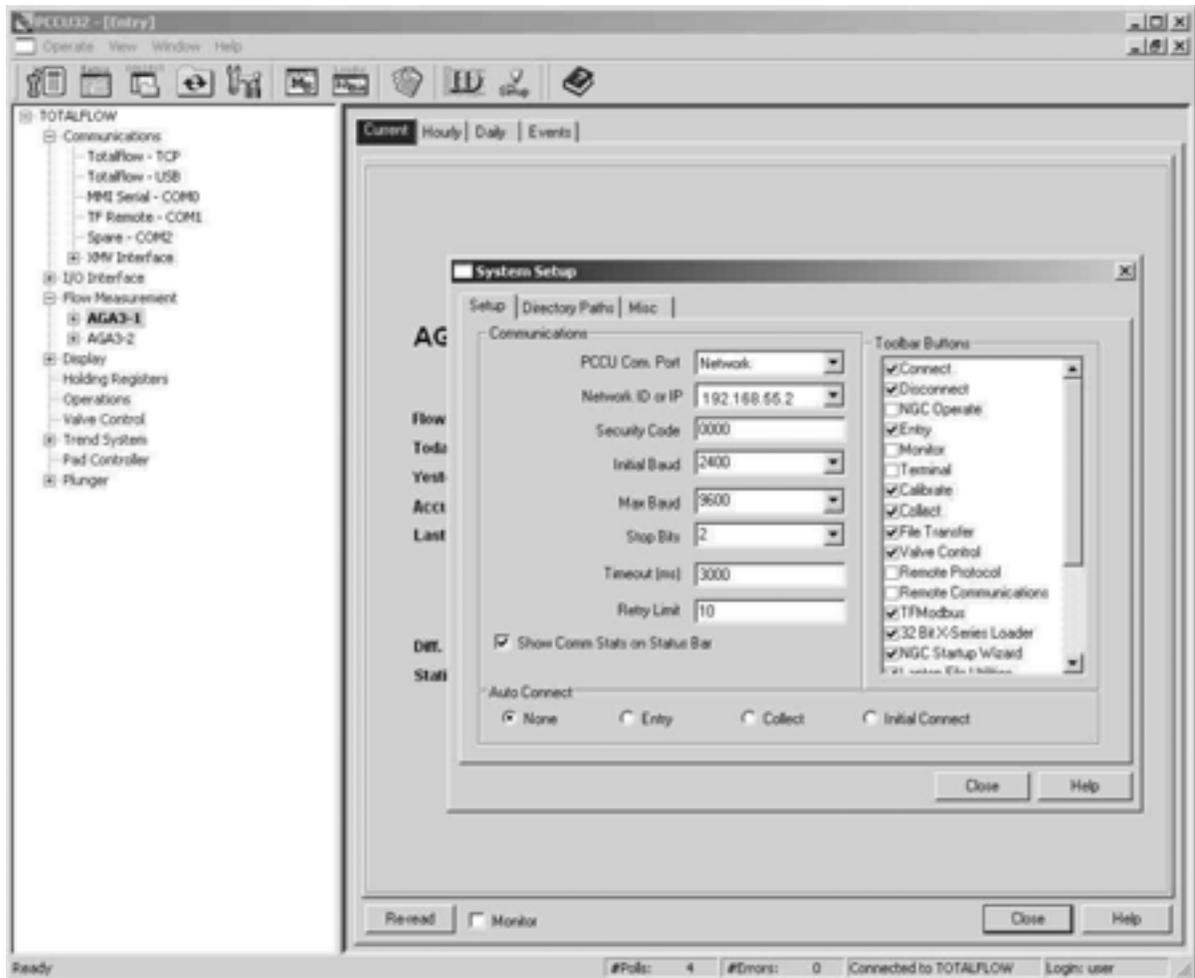
**Table 3-3 Annunciators**

Name	Value	Description
ANN_NONE_C	0x00	None
ANN_OFF_C	0xff	Annunciator off character
ANN_HI_C	0x01	Hi Limit
ANN_LO_C	0x02	Lo Limit
ANN_LL_C	0x03	Low Lithium Battery
ANN_LC_C	0x04	Low Charger
ANN_OV_C	0x05	A/D OV
ANN_BF_C	0x06	Backflow
ANN_LRBATT_C	0x07	Remote Low Battery
ANN_VC_VO	0xF8	Valve Override
ANN_VC_FO	0xA2	Valve Full Open
ANN_VC_FC	0xA3	Valve Full Closed
ANN_VC_OV	0x01	Valve Opening
ANN_VC_CV	0x02	Valve Closing
ANN_VC_LL	0x03	Valve Local Lockout
ANN_VC_LB	0x07	Valve Low Battery
ANN_PL_C	0xb7	Remote Listen
ANN_PT_C	0x7e	Remote Transfer (Send)
ANN_PD_C	0x7f	Remote Download (Receive)
ANN_PS_C	0x5c	Waiting for SYNC
ANN_PA_C	0xCE	Waiting for Ack

### 3.4 Setting Up an IP Address in the G4 Device

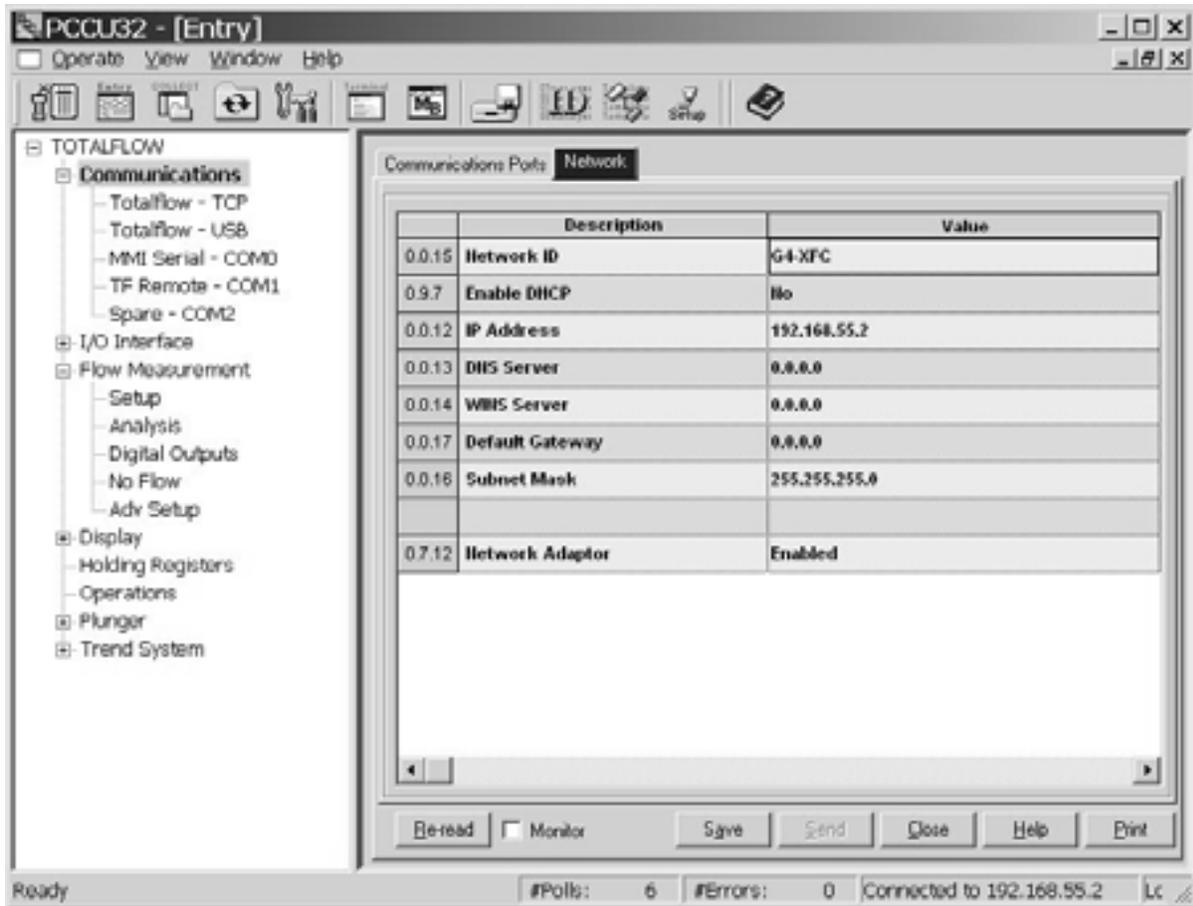
The following information details the steps for establishing an IP address in the ABB Totalflow G4 devices.

- 1) Connect to the G4 device with PCCU32. Click on the Setup icon from the PCCU32 standard toolbar. The System Setup dialog box displays.
- 2) In the System Setup dialog box, click on the Setup tab. Within the Setup tab, the user needs to locate the PCCU Com. parameter field. Once located, click the down arrow, and select Network from the drop-down menu.
- 3) Next, move to the Network ID or IP parameter field. Enter in the network address (see Figure 3-28). Upon completion, click the Close button.



**Figure 3-28 System Setup Dialog Box**

- 4) In the PCCU32 tree-view, select Communications. In the Communications window, click the Network tab.
- 5) Within the Network tab, enter in the network ID, IP address and subnet mask into the corresponding parameter fields (see Figure 3-29).



**Figure 3-29 Network Tab**

### 3.5 Special ABB Totalflow Variables

There are two special variables that ABB Totalflow uses to help the developer.



It should be noted that variables are case-sensitive.

- PartNumber – Used by PCCU32 to determine the INI file for this IEC program.
- NumRegisters – Sets the size of the indirect address array.

### 3.6 Instantiating the IEC Interface in PCCU32

The IEC Interface is an ABB Totalflow application that must be instantiated to run any IEC applications. The IEC serves as the link between the IEC application and the G4 code.

Within G4, application slots have been assigned to various applications that run on the G4 machines. To turn on the IEC Interface application, connect to the G4 device using PCCU32. Once in Entry mode, locate the meter ID within the PCCU32 tree-view. This is the first menu item at the top of the tree-view. Click on the meter ID and then move to the Applications tab.

Within the Applications tab, click the Add App button. This causes the Add New Application dialog box to display. From the Applications to Add parameter field,

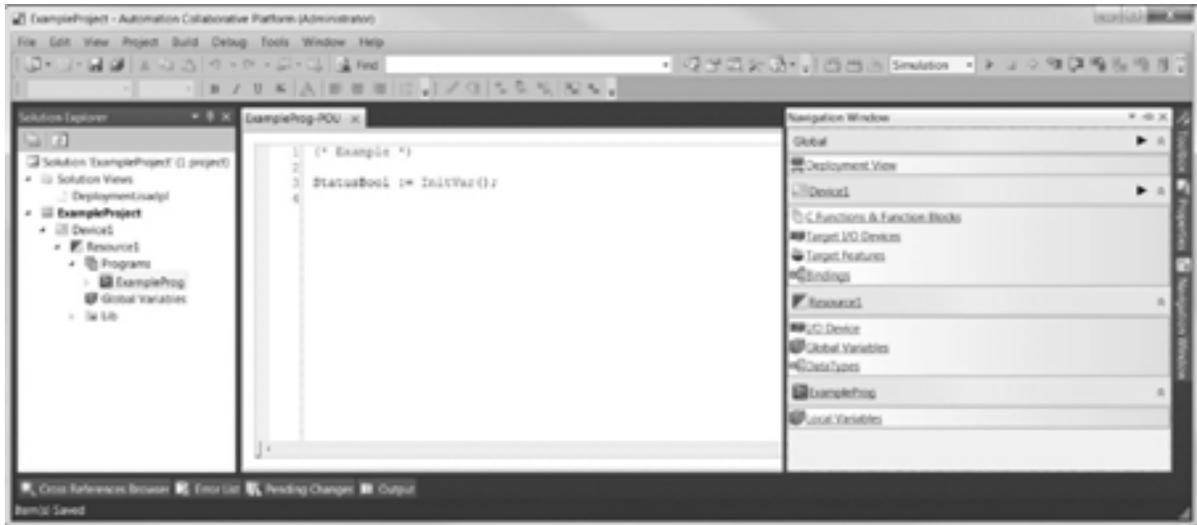
click the down arrow. Select the IEC Interface from the drop-down menu. Upon completion, click the OK button.

The user can then see the IEC Interface application within its assigned application slot. Click the Send button to instantiate the IEC Interface application. The user can then see that the IEC Interface application is within the PCCU32 tree-view.

### 3.7 Building, Compiling and Downloading the IEC Application

The user needs to refer to the ISaGRAF documentation for specific information regarding the various aspects of IEC-61131. When the application has been completed, the user needs to compile the results. This entails the user right-clicking on Project>Build. The necessary executable is then created.

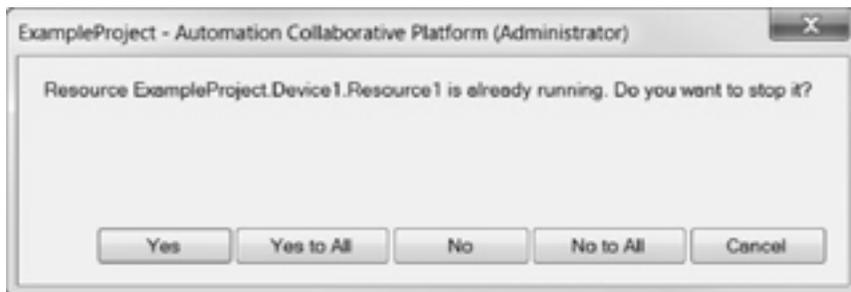
After building and compiling the newly created IEC application, the user needs to download the application to the target device. To download to the target device, click on the Download icon on the standard toolbar (see Figure 3-30).



**Figure 3-30 Downloading to the Target Device**

Upon selection, the file begins to download. In the lower left-hand corner of the screen, the user can see the message "Download Started Project: ExampleProject1". Additionally, the lower right-hand corner of the ISaGRAF development environment displays a download progress bar.

A dialog box will then display. Click the Yes to All button (see Figure 3-31). This completes the download process.



**Figure 3-31 Download Dialog Box**

At this point, the IEC application is downloaded to the G4 device and is ready to be ran or debugged. For further instructions on the debug commands and techniques, refer to the ISaGRAF documentation. While the application code has been downloaded to the G4 device, it still needs to be instantiated before it will run on the G4.

### 3.8 Deploying the IEC Application to Other G4 Devices

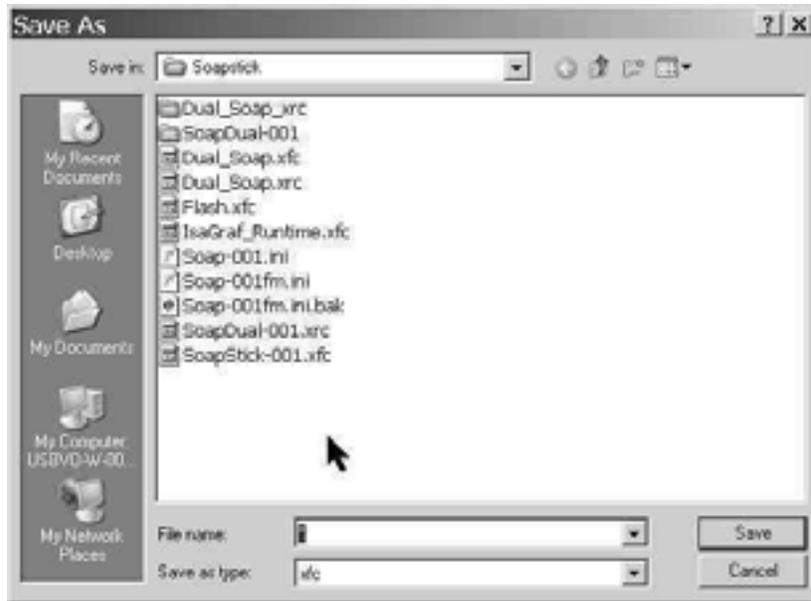
Generally, an IEC application is applicable to more than one site. This means the same IEC application can be downloaded to other G4 devices.

- 1) Open PCCU32. From the PCCU32 standard toolbar, select Package.
- 2) From the Package drop-down menu, select Upload Package (see Figure 3-32).



**Figure 3-32 Upload Packages**

- 3) The Upload Packages dialog box displays. Select IEC Application.
- 4) Click the Start button.
- 5) A Save As dialog box displays. The user can then navigate to the location where they elect to save the file (see Figure 3-33).



**Figure 3-33 Save As Dialog Box**

**FYI**  The file extension associated with the file must match the board type (XFC or XRC).

- 6) Assign a name to the file, and click the Save button. This saves the application to the selected location.

### 3.9 Downloading to Other Devices

The following information details the steps for adding additional locations.

- 1) Open PCCU32. From the PCCU32 standard toolbar, select the 32-Bit Loader icon.
- 2) Within the 32-Bit Loader window, click the Browse (...) button. Move to the location where the user elected to save the Location file.
- 3) Upon location, place a check in the Download ISaGRAF Runtime checkbox.
- 4) Click the Start button.

**FYI**  The device will need to be in a Loader state:

- a) Click on the 32-Bit Loader icon within PCCU32.
- b) Select the Shutdown Totalflow.exe checkbox.
- c) Select the Download ISaGRAF Runtime checkbox.
- d) Select the Start Totalflow.exe checkbox.
- e) Click the Start button to begin the download process.

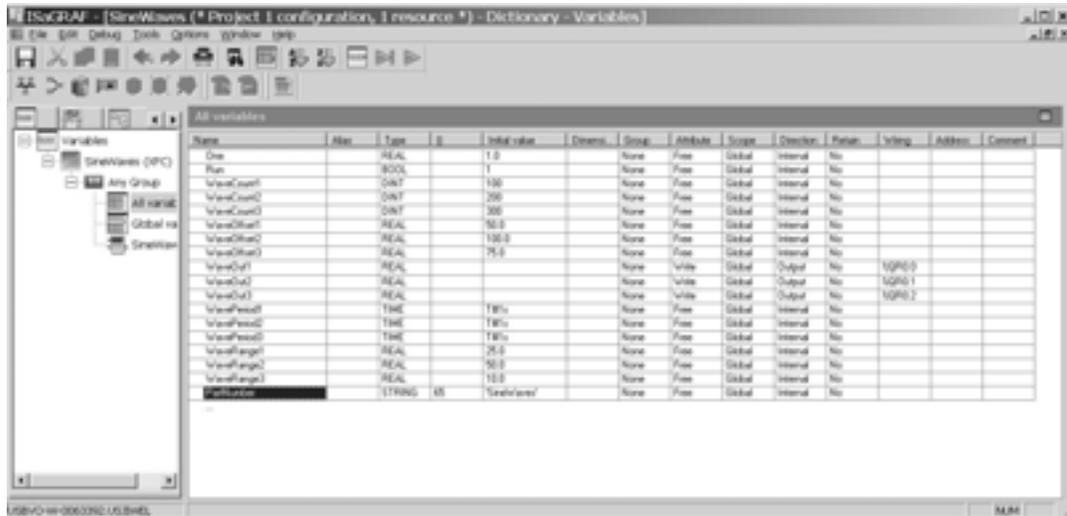
- 5) Upon completion, select the Start Totalflow.exe checkbox.
- 6) Click the Start button. The location loads into the system.

### 3.10 IEC Application and INI Files

ABB Totalflow's PCCU32 uses INI files to define user screens within PCCU32. INI files are text files. PCCU32 parses the INI files to create the various screens within PCCU32.

If a custom INI file is not created for the IEC application, a default INI file is provided.

The first step in creating a custom INI file is to create the variable, PartNumber, within the ISaGRAF development tool (see Figure 3-34).



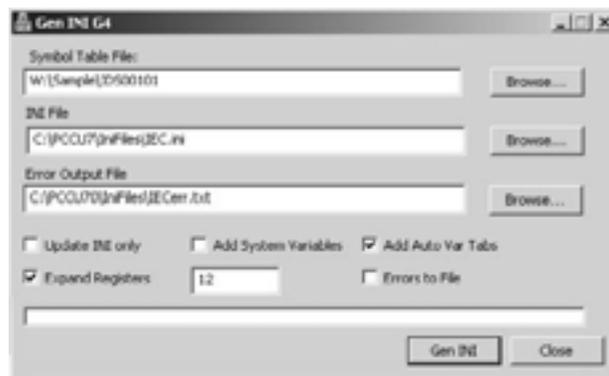
**Figure 3-34 INI File**

Within the IEC Interface application in PCCU32, point to the specific, custom INI file to be used with the user's IEC application. For the purpose of this manual, the custom INI file is named Sine-001.ini or Sine-001fm.ini (Expert view) and should reside in the \PCCU32\INIFiles folder.

PCCU32 can automatically decrement the revision level (-001) to a lower number of the file if the current version does not exist on the current PCCU32 directory.

**CAUTION**  The file prefix (sine in Figure 3-34) cannot be longer than ten (10) characters.

ABB Totalflow supplies a program, Get INI G4, which will read a symbol table file (i.e., IDS00101) and then generate an INI file as a starting point for a developer to customize the INI file (see Figure 3-35). ABB Totalflow recommends that the option "Add Auto Var Tabs" be used to make an Expert view file. Upon completion, the user can make all of the customizations and then duplicate the files as an Advanced view INI file.



**Figure 3-35 Gen INI G4 Dialog Box**

### 3.11 Customizing the Custom INI File

INI files are text files that read by PCCU32 or WinCCU32 to determine how to display data within an application. The INI Programmer's Manual details INI file commands.

#### 3.11.1 Example

In the following example, each lone is described below. The line numbers are for reference points only and are not used in an actual INI file.

```
1 [Status]
2 dsc:Enable Controller;cmd:13.0.1;typ:e;lst:ok=0,ALARM=1;
3 dsc:Alarm;cmd:13.6.1;typ:J;lst:ok=0,ALARM=1;
4 dsc:Stick Available;cmd:13.5.11;typ:l;
5 dsc:VALUES;typ:Z;
6 dsc:SP;cmd:13.9.1;typ:f;
```

1. Indicates the name of the first tab.
2. The description is Enable Controller. The register displayed is array 0, index 1. The 13 represents the App type not the application number. The value is to be displayed as a drop-down box of values: OK or ALARM. The register type is byte integer as indicated by the typ: e;
3. This is similar to line 2 but with a data type of 4-byte integer.
4. The description is Stick Available. The register is displayed as a 4-byte integer.
5. The description is Values and does not display a register value. This causes a header for the next several values to help the user distinguish sections of values.
6. The description is SP. The register is displayed as a floating point value with default width and precision (see Figure 3-36).

	Description	Value
14.0.1	Enable Controller	ok
14.6.1	Alarm	ok
14.5.11	Stick Available	0
	VALUES	
14.9.1	SP	0.0000

Figure 3-36 PCCU32 View of INI File





©Copyright 2012 ABB, All rights reserved

---

Document Title

IEC-61131 Developer's Manual AI

---

Document No.	Date & Rev. Ind.	No. of Pages	Page
2104713	AA	38	38

---