Quick Start Guide

FTSolution for APT

Version 1.x.x

FTSolution for APT - Hardware Debugger

FTSolution for APT Simulator

FTVersionTrak for APT



www.fast-soft.com 262.238.8088

FTSolution for APT, FTSolution for APT Hardware Debugger, FTSolution for APT Simulator, FTVersionTrak for APT, and Foundations: FTSolution for APT are trademarks of FasTrak SoftWorks, Inc. Siemens APT and APT are registered trademarks of Siemens Industrial Automation, Inc. 2500 Series is a registered trademark of Control Technology, Inc. All other trademarks and trade names are property of their respective owners. All rights reserved.

Contents

Introduction
Compilation Differences7
Installing FTSolution for APT9
Importing an Existing APT Program14
Importing .1 Archive Files15
Using the Solution Explorer16
Displaying and Editing Project Components16 Displaying Multiple Project Components17
PLC Type Configuration
Displaying and Editing Recipe Templates
Displaying and Editing Recipes21
Displaying and Editing Declarations 22
Displaying and Editing Continuous Function Charts23
Displaying and Editing Block Settings in Continuous Function Charts
New Instructions that Separate Blocks
Statement Completion
SFC Magnification 30 SFC Zoom Mode 30 Displaying and Editing Math Statements in Sequential Function Charts 31 Code Regions 31 Flyby Windows Containing Statements Within a Collapsed Code Region 32 New Instructions that Separate Steps and Transitions 33 New Text Editor Features 33 Syntax Coloring 33 Hover Text 34
Incremental Parsing
Statement Completion
Usplaying and Editing Devices
Displaying and Editing Subroutines

New Instructions in Subroutines New Text Editor Features	
Syntax Coloring	
Hover Text	
Incremental Parsing	
Statement Completion	
Compiling an FTSolution for APT Project	
Compliation Report	
Communication Setup	
Downloading a Compiled FTSolution for APT Project	
Debug Mode	
Warning About Debug Mode	
Starting Debug Mode	
Monitoring Variables in Debug Mode	
Modifying I/O in a Watch Window	
Forcing I/O	
Clearing Forced I/O	
Tracing SFC Program Flow	
Logging	
Adding an I/O Module for a 505 Series / CTI Controller	
Defining I/O Points for 505 I/O	
Profibus Configuration	60
Adding GSD files to the Hardware Directory	61
Assigning Slaves to the Profibus Configuration	
Configuring and Addressing Profibus I/O Modules	
Defining I/O Points in Profibus I/O	
Merging a .2bf File	
Viewing and Editing I/O Symbolic Names	
FTSolution for APT Hardware Debugger	
Creating an .AHD file in FTSolution for APT	67
Opening an .AHD file in FTSolution for APT	68
Opening an .AHD file in Hardware Debugger	68
FTSolution for APT Simulator (505 Simulator)	69
Starting 505 Simulator	
Selecting PLC Type	69
Setting Up Communications with 505 Simulator in FTSolution for APT Connecting to 505 Simulator	
FTVersionTrak for APT	
ETVersionTrak Toolbar	
FIVEISUILLAN TUULAL	/2
Setting FTVersionTrak Ontions	······/3
Using Solution Explorer with ETVersionTrok	······/4 7/
	/4

Example of Compare: Comparing at the Project Level	75
Example of History: Historical Information for a Single File	75
PCS Tagging – Marking Objects for Translation	76
Tag Translation and Compile Options	8 0
Location of Install.tag and Append.tag Reports	8 0

Introduction

Thank you for purchasing FTSolution for APT[®] by FasTrak SoftWorks, Inc. FTSolution for APT is part of a family of products and services provided by FasTrak SoftWorks, Inc. to meet the needs of APT language programmers. This family includes:

• FTSolution for APT

Windows-based APT language programming software. FTSolution for APT allows you to create and maintain APT-based PLC programs with a modern Microsoft Windows interface.

• FTSolution for APT Hardware Debugger

Sophisticated PLC hardware debugger utilizing ladder language-based tools. FTSolution for APT - Hardware Debugger allows users to download compiled FTSolution for APT programs into Siemens 505/CTI 2500 Series processors, utilize sophisticated ladder language-based debugging tools, and limit the ability to modify existing programs.

• FTSolution for APT Simulator

PLC Simulator for most APT-supporting Siemens 505 and CTI 2500 Series processors. FTSolution for APT Simulator saves time and hardware expense by running and testing APT projects without the use of a PLC.

FTVersionTrak for APT

Version control for FTSolution for APT. FTVersionTrak for APT safeguards your FTSolution project files with version management, detailed program comparison, and automatic backup.

Contact your FasTrak SoftWorks, Inc. Sales Representative at **262-238-8088** or **sales@fast-soft.com** for assistance in purchasing or upgrading these products. For assistance installing and using FTSolution for APT contact Technical Support at **262.238.8088**, **Option 3** or **techsupport@fast-soft.com**.

Compilation Differences

The following list describes intended differences related to compilation that exist between APT and FTSolution for APT.

FTSolution for APT's advanced compiler

The new compiler tests programs more thoroughly than the original APT compiler, and may detect previously unidentified errors. Depending upon their severity, you may ignore these errors or edit your programs to eliminate them.

Programs that contain specific user-defined I/O modules

In APT, programs can contain user-defined I/O modules with a mix of analog input (WX) and digital output (Y) that total 8. This consists of user-defined I/O modules configured in the following ways:

- WX7 and Y1
- WX6 and Y2
- WX 5 and Y3
- WX4 and Y4
- WX3 and Y5
- WX2 and Y6
- WX1 and Y7

APT compiles and downloads programs with I/O modules configured this way without a warning or an error message. FTSolution for APT will allow import of this condition, but will flag an error on compilation.

SCL Verbs

In APT, SCL verbs can be used as identifiers. (For example, the statement: **AUTO AUTO** is allowed in APT). FTSolution for APT does not allow SCL verbs to be used as identifiers.

Literals of real numbers

In APT, literal real numbers having the form of 1E6 are allowed. FTSolution for APT rejects literal real numbers in the form of 1E6. A decimal point (".") before the E (1.E6), which distinguishes the real literal from an identifier, is required in FTSolution for APT.

ASM Semantic Checking

No ASM semantic checking is implemented in FTSolution for APT. For this reason, only use already semantically-valid ASM when compiling in FTSolution for APT.

User-assigned Y address in a Boolean declaration

APT cannot compile a program when a user-assigned Y address is used in a Boolean declaration. In FTSolution for APT, Y addresses in a Boolean declaration are allowed.

PLC Type Checking on Download

In FTSolution for APT, an error will occur during download if the PLC type set in the program's configuration file does not match the actual PLC to which FTSolution for APT is connected (see PLC Type Configuration on page 18 for additional information.)

PLC Type Checking on Compile

In FTSolution for APT, if an imported APT file has configured memory beyond that of a 555, FTSolution for APT will generate an error on compile. This typically occurs when using a CTI 2500 to emulate a Simatic 555. To prevent an error on compile, a CTI 2500 PLC type must be specified in the program's configuration file (see PLC Type Configuration on page 18 for additional information.)

Note: There is no internationalization in FTSolution for APT. FTSolution for APT supports only United States English Regional and Language options. For example, floating-point numbers must follow US conventions: 1.5 must be used instead of 1,5. String literals must also follow US conventions: double quotes must be used to indicate a string ("") instead of double brackets (<< >>). You may want to change your Regional and Language Options in Windows to reflect United States English while using FTSolution for APT.

Installing FTSolution for APT

Previous versions of FTSolution for APT must be uninstalled before v1.x.x can be installed. Uninstalling previous versions of FTSolution for APT removes the application only. Any APT programs that you imported, edited, or wrote will not be deleted. <u>When using v1.x.x, please re-import and</u> <u>compile any APT programs you have previously used with FTSolution for APT. Version 1.x.x requires</u> <u>re-importing in order to experience full functionality.</u>

Note for Windows XP users: FTSolution for APT requires that Windows XP Service Pack 3 is installed first.

To uninstall the previous version of FTSolution for APT:

Windows 8

- 1. Go to the **Start** menu by pressing the Windows key.
- 2. Right-click **FTSolution for APT**.
- 3. Click Uninstall.
- 4. In the list of installed applications, click **FTSolution for APT**.
- 5. Click Uninstall.
- 6. Follow the prompts. A progress bar shows you how long it will take to safely remove FTSolution for APT. When it is finished, you will see the list again.

Windows XP, Windows Vista, and Windows 7

- 1. On the Start menu, click Control Panel and then under Programs do one of the following:
 - a. Windows 7 and Vista users: Click Uninstall a Program.
 - b. Windows XP users: Click Add or Remove Programs.
- 2. On the list of installed programs, click FTSolution for APT, then do one of the following:
 - a. Windows 7 and Vista users: Click Uninstall.
 - b. Windows XP users: Click the **Remove** or **Change/Remove** tab (to the right of the program).
- 3. Follow the prompts. A progress bar shows you how long it will take to safely remove FTSolution for APT. When it is finished, you will see the list again.

Download Windows 8 Prerequisites:

For Windows 8 users only: Windows 8 users must download additional components before downloading and installing FTSolution for APT with FTVersionTrak.

- 1. Go to Control Panel.
- 2. Click **Program and Features**. (If Control Panel is set to Category View, click **Programs** first, and then click **Program and Features**).
- 3. In the left panel, click **Turn Windows Features On and Off**.



- 4. In the Windows Features dialog box, expand .NET Framework 3.5.
- 5. Select the **Windows Communication Foundation HTTP Activation** and the **Windows Communication Foundation Non-HTTP Activation** check boxes.

🛛 Windows Features 🗕 🗆				
Turn Windows features on or off				
To turn a feature on, select its check box. To turn a feature off, clear its check box. A filled box means that only part of the feature is turned on.				
.NET Framework 3.5 (includes .NET 2.0 and 3.0)				
Windows Communication Foundation HTTP Activation				
Windows Communication Foundation Non-HTTP Activation				
🕀 🔳 🔒 .NET Framework 4.5 Advanced Services				
Active Directory Lightweight Directory Services 🗸 🗸				
OK Canc	el			

- 6. Click **OK**.
- 7. Click Download files from Windows Update.



When these files have downloaded, proceed with the following instructions.

Download and Install FTSolution for APT and FTVersionTrak:

- 1. Access the FasTrak website: www.fast-soft.com.
- 2. Login (or register) to the FasTrak website.
- 3. Select **Software Upgrades** from the **Downloads** menu.
- 4. Click the link to **FTSolution for APT** or **FTSolution for APT Full Version** to download the selfextracting installer to your hard drive. *NOTE:* The full version includes all of the prerequisites (such as .NET framework) necessary to install the product. The regular version does not contain

any prerequisites; if needed they are downloaded on demand during the installation. Either one can be used to install FTSolution for APT. The full version is a larger file and can take longer to download. If you have already installed a previous version of FTSolution for APT, the regular version is recommended.

- 5. If you have purchased FTVersionTrak for APT and it was not previously installed, click the link to **FTVersionTrak** to download the self-extracting installer to your hard drive.
- 6. Run the FTSolution for APT installer after it downloads to your hard drive.
- 7. When the installer window appears, follow the prompts to complete the installation of FTSolution for APT.

APT replacement product - InstallShield Wizard				
	Welcome to the InstallShield Wizard for APT replacement product The InstalShield Wizard will install APT replacement product on your computer. To continue, click Next.	×		
InstallShield	< Back Next > Cancel			

- 8. When the installation of FTSolution for APT is complete, run the FTVersionTrak installer.
- 9. When the installer window appears, follow the prompts to complete the installation of FTVersionTrak.



- 10. When FTVersionTrak has completed installing, start FTSolution for APT using the **Run As Administrator** option:
 - a. Right-click the APT replacement desktop icon.
 - b. Select Run as Administrator from the shortcut menu.

	Open			
	Troubleshoot compatibility			
	Open file location			
۲	Run as administrator			
	Pin to Taskbar			
	Pin to Start Menu			
	Restore previous versions			
	Send to +			
	Cut			
	Сору			
	Create shortcut			
	Delete			
	Rename			
	Properties			

11. When FTSolution for APT starts, make sure your key is attached or enter your site license password in the **Site License** dialog box. This password is case-sensitive. Please contact FasTrak SoftWorks, Inc. if you have questions about your site license.

Site License	
Licensed to:	Alpha E
Located in:	Alpha E
Serial number:	Alpha E
Password:	
	OK Cancel

12. If applicable, Start FTVersionTrak from within FTSolution for APT by clicking the **FTVersionTrak** toolbar button on the FTVersionTrak toolbar.



(If the FTVersionTrak toolbar is not displayed in FTSolution for APT, click the **View** menu, point to **Toolbars**, and select **FTVersionTrak Toolbar**).

13. When FTVersionTrak starts, make sure your key is attached or enter your site license password in the **Site License** dialog box. This password is case-sensitive. Please contact FasTrak SoftWorks, Inc. if you have questions about your site license.

FTVersionTra	k
This product Alpha D	t is licensed to:
Located in: Alpha D	
Serial Numb Alpha D	er:
Password	
	OK Cancel

14. In FTVersionTrak, click the **Create New Repository** toolbar button.

PTVersionTrak	
File Tools Help	
: X 🗞 🙆 🖳 🔚 📲 📑 🕹 X	5 (A 2 L 2
	.::

15. In the **Create New Repository** dialog box, in the **Repository Name** box, type a name for the repository.

Create New Reposito	ory	? 🗙
Server:	FTCLOUDVM\FTVERSIONTRAK	Browse
Repository Name:	Ĭ	
	ОК	Cancel

16. If a Login dialog box is displayed, click **OK**.

Login	? 💌
Server:	FTCLOUDVM\FTVERSIONTRAK
Authentication:	Windows Authentication 👻
Usemam	e: FTCLOUDVM\User
Passwor	d:
	Remember password
	OK Cancel

You can now use FTVersionTrak and FTSolution for APT.

Importing an Existing APT Program

When using v.1.x.x, re-import and compile any APT programs you have previously used with FTSolution for APT. Re-importing is required in order to experience full functionality.

Use the following procedure to import APT programs from an APT repository.

1. Click **Import APT Program** on the **File** menu. The Import APT Program dialog box appears.

🗶 Import APT Program			
APT location:	\\FTSVR05XP\APTesting\Reposito Browse		
APT program:	Program name	Description	
	PLC_1	PLC SYSTEM 1	NEW VERSION 1/96
Project name:	Project2		
Location:	C:\APT		Browse
Solution:	Create new soluti	on 🔹	
Solution name:	Project2		Create directory for solution
			OK Cancel

- 2. Do one of the following:
 - In the **APT location box**, type the location of the APT repository that contains the APT program you want to import.
 - Click Browse to browse to the APT repository that contains the APT program you want to import.

A list of APT programs appears in the **APT program** box after a valid repository is selected.

- 3. In the **APT program** box, click the APT program you want to import. The Project **name** and **Solution name** boxes are automatically configured to match the name of the APT program you are importing.
- 4. Click **OK**.

Importing .1 Archive Files

FTSolution for APT must import programs from an APT database. If your programs are stored as .1 archive files, they must first be restored to an APT database, and then imported from the database.

Use the following procedure to restore your .1 archived files to an APT database:

- 1. Determine the location of your .1 archive file. You will need this path in Step 4.
- 2. Launch APT.
- 3. From the main program menu in APT, press 8 or select **Restore** from the F3 context menu.
- 4. Set the Path to the location of the .1 archive, determined in Step 1.
- 5. Enter the name of the archive file that you wish to restore, and press **ENTER**. If a message indicates that the program already exists in your database, you do not need to restore the program to import it into FTSolution for APT.
- 6. If APT and FTSolution for APT are installed on different computers, copy APT's PROGRAM and DATABASE folders to the APT replacement computer. FTSolution for APT imports programs from these folders.

Using the Solution Explorer

FTSolution for APT stores your PLC programs as Projects. These Projects are stored within Solutions. Each Solution may contain one Project. But Solutions can hold multiple Projects. This allows related PLC programs to be grouped within the same Solution.

You access the Projects in a Solution with the Solution Explorer. This Windows tree-control provides direct access to every component of your PLC program Project.

You can expand and contract the nodes in the Solution Explorer to display and hide lower nodes.

Open Solutions and Programs through items in the File menu, or the Open Project button.

Selecting Projects from the Solution Explorer is similar to choosing PLC programs from the APT Program Directory Level screen. But the Solution Explorer also provides access to all PLC program components, similar to the APT Program Content Level screen.

In the following illustration, the Solution Explorer appears along the right edge. Solution **MasterSolution** contains two Projects, **Manhattan** and **Test Project**.



Displaying and Editing Project Components

Any component within a Project can be displayed in its own editor window.

To edit a Project component, use the up and down arrow keys to highlight the item in the Solution Explorer, and press **Enter**. Or double-click an item to select it.

Displaying Multiple Project Components

Multiple Project components can be opened at the same time, and displayed in their own editor windows. In the following illustration, the editor windows of four components are open. The red circle near the upper left corner indicates the four tabs that provide access to the four editor windows. Click on a tab, or press **Ctrl-Tab** repeatedly, to bring the desired editor window to the foreground.

🕱 MasterSolution - FTSolution for APT							
File Edit View Project Tools Window Help							
🗄 🚰 New Project 👔 Open Project 🕼 Save All 🛛 🕮 Compile 🧠 Download 💡 Debug 💡 Exit Debug 💂							
Solution Explorer 🛛 🔻 구 🗸	Recipes.arut 505 IO Modules.amod Cor	nfiguration.acfg* × IO.aios					
_	Object: MANHATTA	Settings Units to Compile					
Solution 'MasterSolution' (2 projects)		A					
Manhattan	Program Name:	MANHATTA					
Computer Setup acor							
Configuration.acfg	Controller Information						
Process Groups.aprg							
⊿ 😫 IO	Controller Type:	SIMATIC 555 🔹					
📮 IO.aios	Controller Deleger	F					
Modules	Controller Release:	5.x •					
Devices	Memory Size (KB):	384					
Declarations							
 f. Subroutines 	6 1 A 1						
Recipe Templates	Compile Options						
\$ SFC	Force Compile?						
CFC							
👀 Watch	Create Debug Version?						
Reports	Ruid Translate Table?						
Iest Project	Duild franslate rable?						
	Uninterruptible Power (UPS)?						
Modules							
Devices	Do Not Convert TOD to Integer Fields?						
Declarations	Synchronize PLC Clock with Computer?						
Recipes	Synanomizer ze elocit mar compater.						
b f _(x) Subroutines							
Recipe Templates	Reserved Locations						
	Ladder (words):	0					
R® Watch		·					
	Variable (words):	0 *					
Creating project 'Manhattan' project creation suc	ccessful.	ili.					

PLC Type Configuration

PLC type is one of the settings specified in the compile configuration file (**Configuration.acfg**). During an import, the PLC type is automatically configured. For a successful download, the configured PLC Type **must** match the target controller type.

Use the following procedure to specify the PLC type.

- 1. View an FTSolution for APT project in the Solution Explorer.
- 2. Double-click **Compile** in the Solution Explorer. You can also double-click **Configuration.acfg**.

Solution Explorer 🔹 🗖 🗙
Solution 'MasterSolution' (2 projects) Manhattan Manhattan Compile Configuration.acfg Process Groups.aprg Configuration.acfg Process Groups.aprg Modules Devices Devices Devices Devices Configurations Configuration.acfg Process Groups.aprg Configurations Configurations
Test Project

- 3. In the **Controller Type** box, select the type of controller you are using.
- 4. In the **Controller Release** box, select the release of the controller.
- 5. Enter other settings as required.

Project24 - APT replacement produc	t (Administrator)
File Edit View Project Tools Wind	low Help
🔁 New Project 💼 Open Project	🛃 Save Al 🛛 🖽 Compile 🗸 Verify 🦓 Download 🚆
Configuration.acfg* × Error List	
Settings for UNNAMED1 Setting	Js Units to Compile
Program Name:	UNNAMED1
Description:	
	E
Controller Information	
Controller Type:	CTI 2500 •
Controller Name:	
Controller Release:	C400 -
PCS Release:	2.x •
Memory Size (KB):	3072
Compile Options	
Force Compile?	
Create Debug Version?	
id Translate Tak	No month

6. Save the **Configuration.acfg** file by clicking **Save Configuration.acfg** on the **File** menu.

Note: The original APT did not directly support the CTI 2500 family of PLCs. In order for the CTI 2500 Series[®] to work with the original APT, the CTI 2500 PLCs had to emulate the Simatic 505 series. This emulation required the use of a software patch for APT and a DIP switch setting on the CTI 2500 Series PLCs.

As FTSolution for APT directly supports the CTI 2500 Series, FTSolution for APT ignores the setting of the DIP switch. However, in FTSolution for APT, **you must specify a PLC type of CTI 2500 if you are connecting to a CTI 2500 processor**. A mismatched PLC type will cause an error and prevent the program from downloading.

Displaying and Editing Recipe Templates

In the following illustration, **Recipe Templates** in the **STATION_53** Project is selected. Selecting **Recipe Templates** from the Solution Explorer along the right edge of the window opens the Recipe Templates editor on the left side.

X MasterSolution - FTSolution for APT			- # -		
File Edit View Project Tools Window Help					
🔢 Twey Project 🧊 Open Project 📓 Save Al 🛛 🛗 Comple 🥱 Download 💡 Debug 👷 Exit Debug					
CL_STATE.arrt ×			Solution Explorer $ imes \P imes$		
+ New recipe element / Edit O Delet	te		Solution 'MasterSolution' (2 projects)		
Name	Туре	Description	A STATION_53		
BLANK	Integer	blank message	Comple Communications Setup.acom		
EXTENDED	Integer	cylinder extended	Configuration.acfg		
RETRACTED	Integer	cylinder retracted	Process Groups.aprg		
FTE	Integer	cylinder fails to extend	📮 IO.aios		
FAILD	Integer	both limit switched are on	Modules		
ON	Integer	vacuum on/valve on	Declarations		
OFF	Integer	Vacuum off/valve off	A Sectors and		
- New Recipe Element -			⇒ f _(x) Subroutines		
			 Recipe Templates 		
			V CL_STATE.arrt		
			CFC		
			89 Watch		
			Test Project		
Object: BLANK	Settings		-		
Name:	BLANK				
Type:	Integer 👻				
Description:	blank message				
Engineering Units:					
L					
Ready					

The upper portion of this editor lists all the **Recipe Templates** of the **STATION_53** Project. The lower portion of this editor displays the **Settings** of the Recipe Template that is highlighted in the upper portion.

Highlight a Recipe Template in the upper portion, to display its Settings in the lower portion.

Settings are validated as they are edited.

Displaying and Editing Recipes

In the following illustration, **Recipes** in the **STATION_53** Project is selected. Selecting **Recipes** from the Solution Explorer along the right edge of the window opens the Recipes editor on the left side.



The upper portion of this editor lists all the **Recipes** of the **STATION_53** Project. The lower portion of this editor displays the **Settings** or **Recipe Values** of the Recipe that is highlighted in the upper portion.

Press **F10,** or click the **Settings** and **Recipe Values** buttons, to toggle between the Settings and Recipe Values editors.

Highlight a Recipe in the upper portion, to display its Settings or Recipe Values in the lower portion.

Settings and Recipe Values are validated as they are edited.

Displaying and Editing Declarations

In the following illustration, **Declarations** of the **CUPRIC** item in the **POWERLINE** Project is selected. Selecting **Declarations** from the Solution Explorer along the right side of the window opens the Declarations editor on the left side.

🕱 MasterSolution - FTSolution for APT								
File Edit View Project Tools Window Help								
🕴 🛅 New Project 👘 O	pen Project 🏼 🎒 Sa	ve All 🛛 🔛 Compile 🧠 Download 🍚 Debug 😡 I	Exit Debug 🖕					
Declarations.adec ×	Dedarations.adec × Solution Explorer • # ×							
+ New declaration	Solution 'MasterSolution' (2 projects)							
Name	Туре	Description	Address	Marked	▲ STATION_53			
LIFT_DELAY_B	Slow Timer	LIFT TIMER FOR DELAY FEATURE			Computer Com			
PUMP_TIMER_A	Slow Timer	TIMER FOR PUMP A FAILURE			Configuration.acfg			
DUMPER_FTR	Boolean	P603B DUMPER FAULT			Process Groups.aprg			
- New Declaration -					P IO.aios			
					Modules			
					 Devices Declarations 			
					⊿ 💌 Recipes			
					🍮 Recipes.arut			
					$f_{(x)}$ Subroutines			
					A W Recipe Templates			
	PROCESS.arrt							
	1 SFC							
	CFC							
	(Watch							
Object: LIFT_DELAY								
	Name:	LIFT DELAY B		<u>^</u>	01 🛱 🖉			
					Devices			
	Type:	Slow Timer 🔹			Declarations Declarations.adec			
	Description:	LIFT TIMER FOR DELAY			 Recipes 			
		FEATURE			I SFC			
					CFC			
	Reports							
Timeout Preset (x100 msec): 50					Test Project			
Richtdram Automatic								
	PLC Address:	Automatic						
				-				
Ready								

The upper portion of this editor lists all the **Declarations** of the **CUPRIC** item. The lower portion of this editor displays the **Settings** of the Declaration that is highlighted in the upper portion.

Highlight a Declaration in the upper portion to display its Settings in the lower portion.

Settings are validated as they are edited.

Displaying and Editing Continuous Function Charts

To display and edit a Continuous Function Chart, use the up and down arrow keys in the Solution Explorer to highlight a CFC, and press **Enter**. Or double click a CFC to select it.

In the following illustration, the **SERVO** CFC of the **PIER17** item was selected in the Solution Explorer. The CFC diagram appears in the upper portion of the edit window. This editor supports mouse operations, and the same keystrokes available in APT.

MasterSolution - FTSolution for APT	
File Edit View Project Tools CFC Window Help	
🗄 🛅 New Project 👘 Open Project 🞑 Save All 🛗 Comple 🤋 Download 🖓 Debug 🖉 Exit Debug	
SERVO.acfc.fcm* SERVO.acfc* X Dedarations.adec	Solution Explorer 🛛 👻 🕂 🗙
T - Standard 🔻 L - Limiter 👻 V - Valve 💌 Y - Dynamic 👻 🖳 🖓 🙀 🦕 Mark PCS tags	
A - Advanced V S - Selector V M - Math V O - Other V C D ESC	Solution 'MasterSolution' (2 projects)
	STATION_53 Gomple
	ℵ Communications Setup.acom
	Configuration.acfg
	Process Groups.aprg
	Modules
Math Interlock Interlock	Devices
	Declarations
POS_LINK PASSLINK SERVO_EVAL	 Recipes.arut
	▷ $f_{(\infty)}$ Subroutines
	 Recipe Templates
-	CL_STATE.arrt
	V PROCESS.arrt
Press the letter associated with a CFB category to get started.	CFC
	🛞 Watch
	Reports
	CUPRIC
	Devices
	Declarations
	👼 Declarations.adec
	Recipes SEC
No program object is selected.	⊿ Ü CFC
	SERVO.acfc
	SERVO.acfc.cfcm
	SERVO.acfc.diagram
	Reports
	Test Project
	-
Ready	

Displaying and Editing Block Settings in Continuous Function Charts

Use the arrow keys and press **Enter**, or mouse pointer to select a CFC block in the upper portion to edit its **Settings** in the lower portion.

In the following illustration, the **POS_LINK** Math block is selected, and its Settings appear.

Settings are validated as they are entered.

X MasterSolution - FTSolution for APT		
File Edit View Project Tools CFC Winds	w Help	
👔 👔 New Project 👘 Open Project 🛃 Sa	ve All 🛛 🕮 Compile 🦚 Download 🖓 Debug 👷 Exit Debug 💂	
SERVO.acfc.cfcm* SERVO.acfc* X Declar	ations.adec	Solution Explorer $ woheadrightarrow \ensuremath{\mathbb{P}} \times$
T - Standard 🔻 L - Limiter 👻 V - Valve	▼ Y - Dynamic ▼ 🗣 🖓 💦 Mark PCS tags	
A - Advanced 👻 S - Selector 👻 M - Math	▼ 0-Other ▼ C D ESC	Solution 'MasterSolution' (2 projects)
		K Communications Setup.acom
		Configuration.acfg
		IO
		IO.aios
		Modules
Math	Interlock Interlock	Devices Declarations
POS LINK	PASSLINK SERVO EVAL	Recipes
E F G H		🚍 Recipes.arut
* * * *		b f _(x) Subroutines
		V CL STATE.arrt
		PROCESS.arrt
		SFC
Press the letter associated with a CFb category		Se Watch
Object: PO5_LINK	Settings Edit Code (F10)	Reports
Name:	POS_LINK	
Description:		Devices
		 Declarations
		Declarations.adec
Type:	Continuous •	F SEC
Input A:	•	A 🛱 CFC
		SERVO.acfc
Input B:	Ť	SERVO.acfc.cfcm
Input C:	•	Re Watch
Transk Da		Reports
Input D:	•	Test Project
Output E:	v	
	•	
Ready		

Displaying and Editing Math Statements in Continuous Function Charts

Press **F10** or click the **Edit Code (F10)** button in the Settings portion of the editor above to open the CFC Math Language Text Editor, as illustrated.



Code Regions

The text editor in APT displays the math statements for the selected CFC block only. However, the Math Statement Text Editor in FTSolution for APT provides access to the math statements for every block in the CFC.

This editor uses Code Regions to expand and collapse the statements for each block.

Click a + plus sign to expand the statements for that block, and click the – minus sign to collapse the statements. Or locate the cursor within a code region and press **Ctrl+MM** to toggle its expansion and collapse.

Flyby Windows Containing Statements Within a Collapsed Code Region

You can view the statements within a collapsed code region by hovering the mouse pointer over it. As illustrated, a flyby window containing the statements in the collapsed region appears. These flyby windows allow you to examine the statements in a collapsed region without expanding it.



New Instructions that Separate Blocks

The text editor in APT displays the math statements for the selected CFC block only. However, the Math Statement Text Editor in FTSolution for APT provides access to the math statements for every block in the CFC.

Because the math statements of all blocks can be accessed through the same CFC Math Language Text Editor, each block must be separated from the other blocks.

To separate math statement blocks, each block must start with the **#CFB** block_name instruction and end with the **#END_CFB** instruction, as illustrated.

These new instructions are required to separate the math statements of all blocks from each other.

```
=#CFB 05_MAINT
  Date : 12/31/2011
  Author : T Thomas
 Description: Operate station 5 independently.
                                              -----}
 PRAGMA("RLL");
 BEGIN
□ IF (device_num = 295) AND switch_start_pb
                                        THEN
     295sv.MOPEN := TRUE;
 ENDIF:
□ IF (device_num = 295) AND switch_stop_pb
                                         THEN
     295sv.MOPEN := FALSE;
 ENDIF;
#END_CFB;
```

New Text Editor Features

The CFC Math Language Text Editor provides features that increase the speed and accuracy of writing programs.

Syntax Coloring

The text editor uses **Syntax Coloring**, in which keywords, variables, operators, and comments are displayed with different colors, as illustrated. This visually distinct method of displaying statements improves program readability.

Hover Text

The text editor provides **Hover Text**, in descriptive tooltips that appear when the mouse pointer hovers over keywords and variables. These tooltips contain information regarding the item over which the mouse pointer hovers. In the following illustration, the tooltip appeared when the mouse hovered over the variable **x_so4_ind_dn**.

Incremental Parsing

APT identifies errors only after exiting the text editor and compiling. However, FTSolution for APT editor identifies errors immediately with **Incremental Parsing**, in which each line is validated as it is typed. Errors are underlined in red and described in the **Error List** window. Select the **View \ Error List** menu item to display the Error List window. In the following illustration, the red underline in the second TIMER SLOW line identifies the "=:=" syntax error.

```
PRAGMA("RLL");
TIMER SLOW : jglaclat_timer := 15,15,FALSE,FALSE;
TIMER SLOW : jglac2at_timer =:= 15,15,FALSE,FALSE;
BEGIN
```

Statement Completion

The text editor utilizes **Statement Completion**, in which lists of keywords and variables are offered, based upon the characters entered. In the following illustration, the list of keywords and variables appeared, when "t_" was typed into the line. The T_1CESR item is highlighted in the list because it is the closest match to what was typed. Continue typing the remainder of the keyword or instruction or select an item from the list to complete the entry.



Displaying and Editing Sequential Function Charts

To display and edit a Sequential Function Chart, use the up and down arrow keys in the Solution Explorer to highlight the desired SFC, and press **Enter**. Or double-click an SFC to select it. In the illustration, the **04_BARGE** SFC of the **PIER04** item was selected in the Solution Explorer.



The SFC editor supports the same keystrokes that are available in APT. It also supports mouse operations. For example, attach an instruction to the mouse pointer by clicking the instruction toolbar icon at the top of the editor. Then click and drag the mouse pointer in the SFC grid to add that instruction.

The **Pointer** toolbar icon detaches any instruction from the mouse pointer.

The **Check Mark** toolbar icon is equivalent to pressing **Enter**.

The **X** toolbar icon is equivalent to pressing **Escape**.

SFC Magnification

Press **Ctrl** and scroll the mouse wheel to magnify the contents of the SFC grid. In the following illustration, the right screen shot is a magnification of the SFC in the left screen shot.



SFC Zoom Mode

Press **Alt-Z** to enable the SFC zoom mode. This displays Comments next to each step and transition, as illustrated. Comments are the first characters in the statements of each step and transition. Set the Comment width by pressing **Alt-W** and entering a value from o - 68.



Displaying and Editing Math Statements in Sequential Function Charts

Use the arrow keys or mouse to place the SFC cursor on a step or transition, and press **F10** to display its math block in the SFC Math Language Text Editor, as illustrated.

1 STATION_53 - APT replacement product	
Ele Edt Yew Ioop Mudow Beb	
I Gal New Broject (C) Open Broject S Sever Al C Comple - Y Yerly 18 Download > Debug	
01_HOME.adv.dv.m* × 01_HOME.adv. 04_BARGE.adv.*	
WaTNANSITION T11 PS2_NEXT.OLS _AND (Positioner2 to NEXT out);	*
H[FTRANSITION T13 PS2_NEXT.OLS AND [POSITIONER2 to NEXT out]]	
H[FTRANSITION T17 PS2_NEXT.OLS AND (POSITIONER2 to NEXT out)]	
H@TRANSITION T23 FALSE (Wait for e-stop: Safe SFC to clear fault)	
M[ETRANSITION T6 HOME@1_P = 0 ; (STATION1 HOME POINT0]]:	
H[ETRANSITION TIG HOMEGI_P = 1 (STATIONS HOME POINTS)]:	
H[ETRANSITION T12 HOMED_P = 2 (STATION1 HOME POINT2)];	
H[FTRANSITION T16 HOMEB1_P = 3 (STATION1 HOME POINT2]]	
HETRANSITION T22 (HOME&1_P = 2 OR (STATIONI HOME POINT2))	
M(MTRANSITION TS PS2_NOME.OLS AND (POSITIONER2 to HOME out))	
PETRANSITION T4 P32_AUTO.0c5 ADJUSTMENT2 to AUTO out} ADD not(X_P52_2) ADD NOT(Sfc_babt) ADD NOT(sfc_balt[1]) (SFC has not been reset) ADD NOT(sfc_balt[1]) (SFC has not been reset)	
(#END_TRANSITION;	_
R[FTRANSITION T3 SINCHE_THR2.TOUT (STATIONI HOME TIMER2)]	
RETRANSITION T2 SINCHE_THRI.TOUT (STATIONI HOME TIMERI)	
RetRANSITION T1 ps2_evto.CLS;	
HDD_SFC;	
	-
100 % *	2
Ready	Ln 257 C

Code Regions

The text editor in APT displays the math statements for the selected SFC step or transition only. However, the Math Statement Text Editor in FTSolution for APT provides access to the math statements for every step and transition in the SFC.

This editor uses Code Regions to expand and collapse the statements for each step and transition.

Left click a + plus sign to expand the statements for that step or transition, and click the – minus sign to collapse the statements. Or locate the cursor within a code region and press **Ctrl+MM** to toggle its expansion and collapse.

Flyby Windows Containing Statements Within a Collapsed Code Region

You can view the statements within a collapsed code region by hovering the mouse pointer over it. As illustrated, a flyby window containing the statements in the collapsed region appears. These flyby windows allow you to examine the statements in a collapsed region without expanding it.

STATION_53 - APT replacement product	
Ein Edt gew Iosis Window Unip	
: Dimergrapht () Com Brandt I Save Al 22 Corple ∨ Verly 1 Download → Debug	
01_HOME.ede.dox* X BL_HOME.ede DH_BARGE.ede*	
H TRANSITION TIL PS2_NEXT.OLS AND (Positioner2 to MEXT out):	4 *
H WTRANSITION TIS PS2_NEXT.OLS AND (POSITIONER2 to NEXT out):	
H TRANSITION TIT PS2_NEXT.OLS AND {POSITIONER2 to NEXT out}}	
M #TRANSITION T23 FALSE (Walt for e-stop: Safe SFC to clear fault);	
H[ETRANSITION T6 HOMERL_P = 0] [STATIONI HOME FOINTE];	
<pre>H[WTRANSITION TIB HOMEDI_P = 1 (STATIONI HOME POINTI)];</pre>	
H[#TRAMSITION T12 HOMER1_P = 2 {STATION1 HOME POINT2}]	
M TRANSITION TIG HOMERIA TO STATIONI HOME POINTID	
<pre># FTRANSITION T22COMENT FTRANSITION T36 # TTRANSITION T5 PS2_NUTE # TTRANSITION T4. PS2_NUTE # DTRANSITION T4. PS2_NUTE # DFTRANSITION T3 VIONE # DFTRANSITION T3 VIONE # TTRANSITION T3 VIONE # TTRANSITION T3 PS2_NUTE.TOUT (STATION: HOME TDRERI); # FTRANSITION T1 PS2_NUTE.CLS; #EMD_SFC;</pre>	
100 % • 1	יד ד ואס געט

New Instructions that Separate Steps and Transitions

The text editor in APT displays the math statements for the selected SFC step or transition only. However, the Math Statement Text Editor in FTSolution for APT provides access to the math statements for every step and transition in the SFC.

Because the math statements of all steps and transitions can be accessed through the same SFC Math Language Text Editor, each step and transition must be separated from the other steps and transitions.

To separate math statements, step statements must start with the **#STEP step_name** instruction and end with the **#END_STEP** instruction. Similarly, transition statements must start with the **#TRANSITION** transition_name instruction and end with the **#END_TRANSITION** instruction, as illustrated.

These new instructions are required to separate the math statements of all steps and transitions from each other.



New Text Editor Features

The SFC Math Language Text Editor provides features that increase the speed and accuracy of writing programs.

Syntax Coloring

The text editor uses **Syntax Coloring**, in which keywords, variables, operators, and comments are displayed with different colors, as illustrated. This visually distinct method of displaying statements improves program readability.

Hover Text

The text editor provides **Hover Text**, in descriptive tooltips that appear when the mouse pointer hovers over keywords and variables. These tooltips contain information regarding the item over which the mouse pointer hovers. In the following illustration, the tooltip appeared when the mouse hovered over the variable **x_so4_ind_dn**.

Incremental Parsing

APT identifies errors only after exiting the text editor and compiling. However, the FTSolution for APT editor identifies errors immediately with **Incremental Parsing**, in which each line is validated as it is typed. Errors are underlined in red and described in the **Error List** window. Select the **View \ Error List** menu item to display the Error List window. In the following illustration, the red underline in the second TIMER SLOW line identifies the "=:=" syntax error.

```
PRAGMA("RLL");
TIMER SLOW : jglaclat_timer := 15,15,FALSE,FALSE;
TIMER SLOW : jglac2at_timer =:= 15,15,FALSE,FALSE;
BEGIN
```

Statement Completion

The text editor utilizes **Statement Completion**, in which lists of keywords and variables are offered, based upon the characters entered. In the following illustration, the list of keywords and variables appeared, when "**t**_" was typed into the line. The **T_1CESR** item is highlighted in the list because it is the closest match to what was typed. Continue typing the remainder of the keyword or instruction or select an item from the list to complete the entry.

```
resetA_timer.ENABl := t_ ;
resetA timer.RESET :=
                          SW_STOP_PB
                                            ٠
resetC_timer.ENABl :=
                          SWITCH
resetB_timer.RESET :=
                          SYS_CLK
                          SYSTEM_PARAM
                          T_1CESR
                                              AND NOT(re
s04_reset := alarm_res
                          T_ASSY_NG
y_s04_reset := s04_res
                          T_CYC_STOP1
                          T_GATE1_OPR
                          T_INIT_ERR01
{====== Check cycle =======
```

Displaying and Editing Devices

In the following illustration, **Devices** of the **PIER10** item in the **STATION_53** Project is selected. Selecting **Devices** from the Solution Explorer along the right edge of the window opens the Devices editor on the left side.

X MasterSolution	- FTSolution for APT					
File Edit View	Project Tools Window	Help				
New Project	👘 Open Project 💓	Save All Compile 🕲 Download	🎖 Debug 💡 Exit Debug 🥃			
Devices.adev* 🗙	Declarations.adec 04	_BARGE.asfc* Declarations.adec	SERVO.acfc.cfcm SERVO.acfc	Declarations.adec	Ŧ	Solution Explorer 🔹 🖣 🗙
+ New device	🖌 Edit 🖉 Delete	Mark PCS tags				Solution 'MasterSolution' (2 projects)
Name	Type Descrip	otion	Marked			STATION_53
5692V	Valve: Dual Drive / FEC9(F	PUSH/PULL)				 Complex Communications Setup.acom
5690V	Valve: Dual Drive / RBC_P	LT PEPPER				Configuration.acfg
T_253PRS_TMR	Timer: Stopwatch Hold or	n Pallet in Tunnel Signl				
- New Device -						P IO.aios
						Devices
						 Declarations Recipes
						Scheuting
						 W Recipe Templates
						V CL_STATE.arrt
						SFC
						GFC Bit Watch
						Reports
						PIER04
						▲ PIER 10
						Devices
						 Declarations Declarations.adec
Object: 5692	2V	Settings				Recipes
	Name	: 5692V			*	CFC
	Туре	Valve: Dual Drive / Dual Fe 🔻				Watch
	Description	FEC9(PUSH/PULL)				 PIER 17 Test Project
	Open Command	: 5692V.OPENC -				
	Close Command	: 5692V.CL5C •				
	Open Limit Switch	: 5692V.OL5 •				
	Close Limit Switch	: 5692V.CL5 •				
	Normally Open Feedback	?				
	Ignore Feedback Override	?				
	Open Alarm Time (sec)	: 1.0				
	Close Alarm Time (sec)	: 1.0				
					*	
Ready						

The upper portion of this editor lists all the **Devices** of the **PIER10** item. The lower portion of this editor displays the **Settings** of the Device that is highlighted in the upper portion.

Highlight a Device in the upper portion to display its Settings in the lower portion.

Settings are validated as they are edited.

Displaying and Editing Subroutines

In the following illustration, **Subroutines** in the **STATION_53** Project is selected. Selecting **Subroutines** from the Solution Explorer along the right edge of the window opens the Subroutine editor on the left side.

T MasterSolution - FTSolution for AP	т						
File Edit View Project Tools Window Help							
🔋 🛐 New Project 👩 Open Project 🧊 Save Al 🖾 Comple 🦓 Download 🦿 Debug 💂 Exit Debug							
Subroutines.asub* X Devices.adev*	Declarations.adec 04_BARGE	asfc* Declarations.adec SERVO.acfc.cfcm	SERVO.acfc	Declarations.adec	Ŧ	Solution Explorer 🔹 무 🗙	
+ New subroutine / Edit parame	eters 🖌 Edit code 🖉 Delete					Solution 'MasterSolution' (2 projects)	
Name	Type	Description				A STATION_53	
SHIFT_1	Procedure	Shift miss curve data				 Compile Communications Setup.acom 	
MISSDATA	Procedure	Determine Missing Curves				Configuration.acfg	
CHK_DOOR_CLOSE	Procedure	Determine Door Position				Process Groups.aprg ID	
— New Subroutine —						📮 IO.aios	
						Modules Sectors	
						Declarations	
						✓ f _(x) Subroutines	
						Subroutines.asub CHK DOOP, CLOSE Subro	
						MISSDATA.Subroutines.as	
						SHIFT_1.Subroutines.asut Recipe Templates	
						V CL_STATE.arrt	
						CFC	
						Reports	
						P IER04 PIER10	
Object: CHK_DOOR_CLOSE	Settings				Edit Code (F10)	 PIER 17 Test Project 	
	Name: CHK DOOR CLOSE				A		
Der	cription Determine Deer Peritien						
Des	Supton. Determine boor Position						
Parar	meter 1						
	Type: Integer Array	•			E		
	Name: MRS_CURVE_FREE						
An	ray Size: 10						
Des	scription: Hinges Curves Front Palle	t					
					_		
Parar	meter 2						
	Type: Integer Array	•					
	Name: MRS_CURVE_LAST						
An	ray Size: 10				-	۰ III ۲	
Ready							

The upper portion of this editor lists all the **Subroutines** of the **STATION_53** Project. The lower portion of this editor displays the **Settings** of the Subroutine that is highlighted in the upper portion.

Highlight a Subroutine in the upper portion to display its Settings in the lower portion.

Settings are validated as they are edited.
Displaying and Editing Math Statements in Subroutines

Press **F10** or click the **Edit Code (F10)** button to display the math statements in the Subroutine Math Language Text Editor, as illustrated.

STATION_53 - APT replacement product	
Ele Eds New Tools Murgan Deb	
🛛 🕼 Mew Broject 👩 Open Broject 🕼 Save Al 🛛 🗁 Concile 🗸 Verify 🦓 Download 🕨 Debug 💂	
OK_DOOR_CLOSE_Subroutives.asub.subm* X Subroutives.asub*	
#SUBROUTINE CHK_DOOR_CLOSE	王
Progno("Kll");	-
INTEGER : F1,F2,F3,F4,R1,R2,R3,R4; (Pairs of no dose)	
{ 1 pair = 1000 or 0100 or 0010 or 0001 }	
Integer : Pairi_s := #16#8000; Totager : Pairi b := #16#8000;	
Boolean : 25_pair_22f; (1 pair of missing palletized cases)	
Boolean : 60_pair_65f; { 2 pair of missing palletized cases }	
Integer : Foundation_FC_ty; (Missing curves front dollar)	
Facia	
augus .	
(Start of missing curves front curve front pallet)	
<pre>FC_pair_1f := (Foundation_curve_f[10] = Group5_a) (1000)</pre>	
OR (Recedenting survey of the) - Security by (Alter)	
OR	
(Foundation_curve_f[10] = Group8_c) (0010)	_
(Missing curves front pallet)	
If PC Pair 1f Then	
ElsIf FC_pair_2f Then	
Hissing FC_F (= 2)	
Histing FC_f := 0;	
EndIfy	
{	
Missed_FC := Missing_FC_F + Missing_FC_F;	
IF Big_Group_in OR FC_Pair_in THEN R1 (= 1) ELSE R1 (= 0) ENDIF:	
IF Big_Group_2r OR FC_Pwir_2r THEN R2 := 1; ELSE R2 := 0; ENDIF;	
Missed_pairs := (F1 + F2 + F3 + F4 + R1 + R2 + R3 + R4);	
HEND_SUBROUTINE;	-1
100 % • 2	2
Ready	Ln 36 Ca

Code Regions

The Subroutine Math Text Editor in FTSolution for APT groups statements into Code Regions that can be expanded and collapsed.

Left click a + plus sign to expand the statements in that region, and click the – minus sign to collapse the statements. Or locate the cursor within a code region and press **Ctrl+MM** to toggle its expansion and collapse.

Flyby Windows Containing Statements Within a Collapsed Code Region

You can view the statements within a collapsed code region by hovering the mouse pointer over it. As illustrated, a flyby window containing the statements in the collapsed region appears. These flyby windows allow you to examine the statements in a collapsed region without expanding it.

😿 STATION_53 - APT replacement product	
Elle Edit View Iools Window Help	
🗄 🛅 New Project 👘 Open Project 🍠 Save All 🛛 🕮 Compile 🗸 Verify 🤊 Download 🕨 Debug 💂	
CHK_DOOR_CLOSE.Subroutines.asub.subm 🗙 Subroutines.asub*	
OR (Miss_curve_f[9] = Pair3_b) { 0111 } OR (Miss_curve_f[9] = Pair3_c) { 1011 } OR	÷
(Miss_curve_f[9] = Pair3_d);{ 1101 }	
<pre>{ 4 pair = 1111 } BC_Pair_4f := (Miss_curve_f[9] = Pair4_a);{ 1111 }</pre>	
<pre>{ Missing curves front pallet } BIT BC_Pair_1f Then Missing BC_f := 1; {End of missing curves base curve front pallet</pre>	
<pre>(Miss_curve_R 0001 } { 2 pair = 1100 or 0110 or 0011 or 1010 or 0101 } BC_Pair_2R := (Miss_curve_R[9] = Pair2_a) { 1100 } OR (Miss_curve_R[9] = Pair2_b) { 0110 } OR (Miss_curve_R[9] = Pair2_c) { 0011 } OR (Miss_curve_R[9] = Pair2_d) { 1001 } OR (Miss_curve_R[9] = Pair2_e) { 1010 } OR (Miss_curve_R[9] = Pair2_e) { 1010 } OR (Miss_curve_R[9] = Pair2_f); { 0101 } </pre>	
<pre>{ 3 pair = 1110 or 0111 or 1011 or 1101 } BC_Pair_3R := (Miss_curve_R[9] = Pair3_a) { 1110 } OR</pre>	×
Ready	Ln 1 Co

New Instructions in Subroutines

The Subroutine Math Language Text Editor identifies the start of each Subroutine with the **#SUBROUTINE** subroutine_name instruction and identifies the end with the **#END_SUBROUTINE** instruction, as illustrated.

These new instructions are required at the start and end of each Subroutine.



New Text Editor Features

The Subroutine Math Language Text Editor provides features that increase the speed and accuracy of writing programs.

Syntax Coloring

The text editor uses **Syntax Coloring**, in which keywords, variables, operators, and comments are displayed with different colors, as illustrated. This visually distinct method of displaying statements improves program readability.

Hover Text

The text editor provides **Hover Text**, in descriptive tooltips that appear when the mouse pointer hovers over keywords and variables. These tooltips contain information regarding the item over which the mouse pointer hovers. In the following illustration, the tooltip appeared when the mouse hovered over the variable **x_so4_ind_dn**.

Incremental Parsing

APT identifies errors only after exiting the text editor and compiling. However, the FTSolution for APT editor identifies errors immediately with **Incremental Parsing**, in which each line is validated as it is typed. Errors are underlined in red, and described in the **Error List** window. Select the **View \ Error List** menu item to display the Error List window. In the following illustration, the red underline in the second TIMER SLOW line identifies the "=:=" syntax error.

```
PRAGMA("RLL");
TIMER SLOW : jglaclat_timer := 15,15,FALSE,FALSE;
TIMER SLOW : jglac2at_timer =:= 15,15,FALSE,FALSE;
BEGIN
```

Statement Completion

The text editor utilizes **Statement Completion**, in which lists of keywords and variables are offered, based upon the characters entered. In the following illustration, the list of keywords and variables appeared, when "t_" was typed into the line. The T_1CESSR item is highlighted in the list because it is the closest match to what was typed. Continue typing the remainder of the keyword or instruction, or select an item from the list to complete the entry.

```
resetA_timer.ENABl := t_ ;
resetA timer.RESET :=
                                            ٠
                          SW_STOP_PB
resetC_timer.ENABl :=
                          SWITCH
resetB timer.RESET :=
                          SYS CLK
                          SYSTEM_PARAM
                          T 1CESR
                                              AND NOT(re
s04 reset := alarm res
                          T_ASSY_NG
y s04 reset := s04 res
                          T_CYC_STOP1
                          T_GATE1_OPR
                          T_INIT_ERR01
{====== Check cycle ======
```

Compiling an FTSolution for APT Project

The option to compile a project is available when a project (or its compile configuration file) is selected in the Solution Explorer. Please re-import and compile any APT programs you have imported into a previous version of FTSolution for APT.

Solution Explorer 🔹 🗖 🗙
Solution 'MasterSolution' (2 projects)
A STATION_53
Compile
DI 🛱 IO
Modules
Devices
Declarations
Recipes
f(x) Subroutines
Recipe Templates
1 SFC
CFC
Watch
Reports
PIER04
PIER 10
P L PIEKI/
P ist Project

- 1. When a project is selected in the Solution Explorer, a compile of the project can be initiated in several ways:
 - By pressing the 1 key.
 - By clicking Compile on the standard toolbar.



By clicking **Compile** on the **Project** menu.



 By right-clicking a project in the Solution Explorer and then clicking Compile from the shortcut menu.



2. After a compile is initiated, the Compile dialog box is displayed.

201	x
SIMATIC 555	4.x
© No ◎ Append	
Yes Debug	
Force Compile	Cancel
	201 SIMATIC 555 No Append Yes Debug Force Compile

- 3. The **Target** represents the controller specified in the compile configuration file.
- 4. Under Translate, specify the translate option you want to use.
 - When you select **No**, existing marked tags are not modified.
 - When you select **Append**, existing marked tags are placed in the *install.tag* file. Tags marked since the last translate build and tags that were unmarked and re-marked are also placed in a separate file, *append.tag*.
 - When you select **Yes**, all marked tags are placed in the *install.tag* file.
- 5. Under **Options**, select the **Debug** check box to compile for Debug mode.
- 6. Click **Compile** to start compiling the project.

Compilation Report

The compilation report provides information about the compilation process and the compilation output. The following figure is an example of the information contained in the report.

ompile.arpt	× Configuration.acfg	Error Lis	t Watch	1.awch	MA	AIN.asfc	Watch3.awch	
Ph	ase: Assembly code	analysi	3					
Ph	ase: 505 object cod	le gener	ation					
	Summary of compile	d progr	am memory	require	nent	ts:		
	Area	Words	Elements	Used Me	=m	Config 1	1em	
	Ladder (L)	505		2 1	KΒ	805	KB	
	Variable (V)	371		0 1	КB	161	KB	
	Constant (K)	23		0 1	KΒ	161	KB	
	Special (S)	518		1 1	КB	323	KB	
	User (U)	0		0 1	КB	0	KB	
	TMR/CTR/DCAT/MCAT	3	1	0 1	KΒ	5	KB	
	Drum/Edrum	0	0	0 1	KΒ	3	KB	
	Shift register	0	0	0 1	KΒ	1	KB	
	Table move	1	1	0 1	KΒ	2	KB	
	One shot	2	4	0 1	КB	1	KB	
	Nonretentive CR		109					
	Retentive CR		3					
	Y used as CR		0					
	System memory requ	ired by	program:	5 KB				
	Minimum memory con	figurat	ion for pr	ogram: 1	17 I	КB		
	Memory configured	for pro	gram: 3072	KB				
% • 4								•
				n 20		Col 62	Ch 62	INC

To view the compilation report, expand **Reports** in the Solution Explorer, and double-click **compile.arpt**.

Solution Explorer	• 🗆 ×
Solution 'MasterSolution' (2 project	ts)
STATION_53	
I Test Project	
Compile	
⊳ 🛱 IO	
Modules	
Devices	
Declarations	
Recipes	
▷ $f_{(x)}$ Subroutines	
🍸 Recipe Templates	
E SFC	
CFC	
💷 Watch	
Reports	
Compile.arpt	
Dryer1	
Vasher 1	

Communication Setup

Every FTSolution for APT project has a file that defines its communication settings. This file, named **Communications Setup.acom**, is accessible from the Solution Explorer. Communication settings define how FTSolution for APT will communicate to a PLC.

To specify the communication settings:

- 1. View a project in the Solution Explorer.
- 2. Expand **Compile** so that the **Communications Setup.acom** file is visible.



3. Double-click **Communications Setup.acom**. The Communication Setup dialog box is displayed.

X Communication Setup	x
Communication type:	Serial 🔹
COM port number:	1
Response timeout (sec):	6
Retries:	3
	Force NITP
	Dial modem
Modem settings	
Baud rate:	9600 👻
Dial type:	Tone 🔻
Initialization command:	
Phone number:	
	Save Cancel

4. In **Communication Type**, select **Serial** to connect to a PLC using serial communication; select **TCP/IP** to connect to a PLC/Ethernet adapter using TCP/IP.

- 5. If the **Communication Type** is **Serial**, specify the following settings:
 - In the Com Port Number box, enter the communication port you want to use.
 - In the Response Time Out box, enter the response time out you want to use. The response time out specifies the amount of time, in seconds, the software waits for a response from the PLC before returning a time out error.
 - In the Retries box, enter the number of times the software will try to re-establish communications with the PLC after a time-out error.
 - To force the use of NITP, select the **Force NITP** check box. (This specifies ASCII communication for use with Simatic 545, 555, and 575 PLCs.)
 - When the selected form of serial communications is through a modem, select the **Dial** Modem check box, and then specify the following modem settings:
 - In the Baud Rate box, select the rate of communications between the computer and modem.
 - In the **Dial Type** box, select the type of dialing you want to use. Select Pulse dialing only if this is the only type of dialing supported by your phone line.
 - In the Initialization Command box, enter any initialization commands required by your modem. Consult your modem manual for a list of appropriate commands.
 - In the **Phone Number** box, enter the phone number to be dialed.
- 6. If the **Communication Type** is **TCP/IP**, specify the following settings:

X Communication Setup	
Communication type:	TCP/IP •
IP address:	127 0 0 2
TCP port:	1505
Response timeout (sec):	6
Retries:	3
	Use CAMP
	✓ Use packed opcodes
	Save Cancel

- In the IP Address box, enter the IP address of the PLC/Ethernet adapter to which you want to connect.
- In the TCP Port box, enter the TCP port used by the PLC/Ethernet adapter to which you are connecting.
- In the Response Time Out box, enter the response time out you want to use. The response time out specifies the amount of time, in seconds, the software waits for a response from the PLC before returning a time out error.

- In the **Retries** box, enter the number of times the software will try to re-establish communications with the PLC after a time-out error.
- If the PLC/Ethernet adapter to which you are connecting supports CAMP protocol, select the Use CAMP check box.
- If the PLC/Ethernet adapter to which you are connecting supports Packed Opcodes, select the **Use Packed Opcodes** check box.
- 7. Click **Save** to save the communication settings.

Downloading a Compiled FTSolution for APT Project

The option to download a project is only available when a project (or its compile configuration file) is selected in the Solution Explorer and the project has already been compiled.

Solution Explorer	- □ ×
STATION_53	-
4 🛄 Test Project	
Compile	
▷ 📫 IO	
Modules	
Devices	
Declarations	
Recipes	
▷ $f_{(x)}$ Subroutines	=
Recipe Templates	-
SFC	
CFC	
🛞 Watch	
A Reports	
Compile.arpt	
Dryer1	
Washer 1	-

- 1. If a project is currently selected and it has already been compiled, initiate a download by doing one of the following:
 - Pressing the 4 key.
 - Clicking **Download** on the standard toolbar.



Clicking **Download** on the **Project** menu.



 Right-clicking a project in the Solution Explorer and then clicking **Download** from the shortcut menu.

Solu	tion	Explorer		- □ ×
⊳		STATION_5	3	~
4		Co Co	1 - Compile	
	Þ	🔁 IQ 🂡	<u>2</u> - Debug	
			<u>4</u> - Download	
	Þ	물며	Eind Address	
	⊳	🛎 Re	Add	•
	⊳	T _(×) Su ₹ Re →	Cu <u>t</u>	Ctrl+X
		💈 SF 🗙	Remo <u>v</u> e	Del
		Re W	Rename	
	⊿	Re	Info	
	Þ		Description	
	Þ	H Washer	1	-

2. After a download has been initiated, the Download Program dialog box is displayed.

📜 Download Program	×		
Target:	SIMATIC 545		
Communication type:	Serial <u>S</u> etup		
Options:	Download I/O configuration		
	Download incremental changes <u>o</u> nly Grad Discremental changes <u>o</u> nly		
	Download Cancel		

- The **Target** reflects the controller type defined in the compiler configuration file.
- The Communication type displays the currently selected communication type.

Note: If communication settings have not been configured, **Not configured** is displayed for the **Communication type**, and the **Download** button cannot be selected. To set or modify the current communication settings, click **Setup**. (Detailed information about communication setup is provided in the previous section)

- To download the I/O configuration, select the **Download I/O configuration** check box.
- Note: The Download incremental changes only option is not supported in this release.
- To clear the PLC before downloading, select the **Clear PLC before download** check box.
- 3. Click **Download** to begin the download.

Debug Mode

Warning About Debug Mode

Debug mode enables transitioning from Run to Program mode. When using Debug Mode, be aware of the consequences when you transition from Run to Program Mode and outputs are cleared.

Starting Debug Mode

Once an FTSolution for APT solution has been compiled with the Debug Mode option enabled and downloaded to a PLC, Debug Mode can be activated whenever a project or the Compile object is selected in the Solution Explorer. In the following figure, the project named **flipper1** is selected in the Solution Explorer.



If a project or Compile object is selected, activate Debug Mode by doing one of the following:

- Right-clicking the project or the compile object in the Solution Explorer, and then clicking **2**-**Debug** from the shortcut menu.
- Clicking the **Debug Mode** toolbar button on the Standard toolbar.
- Clicking **2-Debug** from the **Project** menu.

Solut	tion I	Explorer			- □ ×
⊳		STATION	_53		*
4		Test Pro	niect		
	\triangleright	🔛 Comp	**	1 - Compile	
	Þ		9	2 - Debug	
	Þ		∢	4 - Download	
	Þ	🧱 Deda		Find Address	
	⊳	🗶 Recip		Add	•
		Recip	¥	Cut	Ctrl+X
		SFC	\times	Remove	Del
		🛞 Wate		Rename	
	4	Repo		Info	
	\triangleright	📑 Drye		Description	
	⊳	📑 Wash	ner 1		T

After activating debug mode, the Connect to PLC dialog box appears.

Communication	n setup	1	
Project:	Project	Connection type	Debug mode
	flipper 1	TCP/IP (192.168.3.204)	Offline

In the Connect to PLC dialog box, select the project you want to debug and then click **Connect to PLC**.

Monitoring Variables in Debug Mode

Watch windows in FTSolution for APT are analogous to the charts used to monitor variables in APT.

Built-In Watches

When Debug Mode is active, FTSolution for APT provides built-in Watch window capabilities. While in Debug Mode, selecting an object in a table editor or the CFC editor causes a **Monitor** tab and a **Sample** tab to be displayed in the Settings Editor.

IO.aios 🗙			Ψ
+ New I/O sym	bol 🖊 Edit	🖉 Delete	
Name	Туре	Description	Address
ANALOGOUTPU	Analog Output	ANALOG1	%WY1
ANALOGINPUT	Analog Input	ANALOG2	%WX9
	Ì		
Object: ANA	ALOGINPUT		Settings (Monitor Sample
.RAW	0 1 5 1	1:42:51.429 A 1:42:51.429 A	M

- Select the **Monitor** tab to display continuously updating data about the object's settings.
- Select the **Sample** tab to display a single instance of data about the object's settings. Click the **Take Sample** button on the **Sample** tab to update the data.

IO.aios X						Ŧ
+ New I/O sym	ibol 🖊 Edit	🖉 Delete				
Name	Туре	Description	Address			
ANALOGOUTPU	Analog Output	ANALOG1	%WY1			*
ANALOGINPUT	Analog Input	ANALOG2	%WX9			
	[
Object: ANA	ALOGINPUT		Settings	Monitor	Sample	
Take sample						
	0 1	1:44:34.490 A	M			
.RAW	5 1	1:44:34.490 A	м			

Custom Watches

To create a custom watch in FTSolution for APT, do the following:

1. Right-click the Watch object in Solution Explorer and select **Add New Watch File** from the shortcut menu.

Solution Explorer	▼ □ ×
Solution 'MasterSolution' (2 projects)	
STATION_53	
I Test Project	
Compile	
> 🔁 IO	
Modules	
Devices	
Declarations	
Recipes	
f(x) Subroutines	
Recipe Templates	
1 SFC	
watch	
Add New Watch File	
Description	
Washer 1	

- 2. In the Add New Item dialog box, enter a name for the Watch file and click **Add**. The Watch file should now be visible in the Solution Explorer.
- 3. Double-click the Watch file in Solution Explorer.
- 4. Enter the variables or addresses you want to add to this Watch window.

flipper1 - APT replacement product			
🛐 New Project 👩 Open Project 🗿 Save	All I Compile ✓ Verif	y 🦚 Download 🤌	🕅 Debug 📱
/atch2.awch × Watch1.awch			
Edit			
Tecent			
Insert Delete	C	Frank	
Variable name or address	Scope	Format	
%WX1		Integer	•
			-
em(s) Saved			

5. Save the Watch window.

In Debug mode, a custom Watch window display a **Monitor** tab, a **Sample** tab, and a **Trend** tab. Clicking the **Trend** tab displays a graphical representation of the data monitored in the Watch window.

The following figure is an example of the information in the Trend tab.

Watch4.awch	MAIN.asfc	Watch3.awch* ×	BOOTSTRP.acfc.cfcm	BOOTSTRP.acfc	MAIN.asfc.sfcm	-
Bet trend	Monitor	Sample	Trend			
			Variable name or	address Scope	Value	
	A A	ЛЛ	% wx1		4	_
	11	11	• %wx9		5	
	//	/ / /	■ %wy1		5	-
10:53:43	10:54	:13 10:	54:43	CUR Program	6	-
	No series	added				
						Ŧ

Modifying I/O in a Watch Window

To modify the value of an input/output point or a variable:

- 1. Add the input/output point or variable to a Watch window. Or, you can open a built-in Watch by selecting an object's editor or a CFC editor.
- 2. In the Watch window, select the input/output point or variable you want to modify, select the **Monitor** tab, and enter the value you want to use in the **Value** box.

Edit	Monitor	Sample	Trend		
Variable nam	e or address		Scope	Value	Timestamp
%WX1				1	01:27:02.503 PM
%WX9				5	01:27:23.748 PM
%WY1				9	01:27:23.748 PM
%WY9				5	01:27:23.748 PM

Forcing I/O

- 1. To force the value of input/output point or a variable, click **Forces** on the **Debug** menu.
- 2. In the Forces dialog box, click **Set Force**.



3. In the Force Address dialog box, enter the input/output point or variable you want to force in the **Variable name** box, and then click **Find**.



4. In the **Value** box, enter the value you want to use.

Variable name:		Scope:			[
%WX1		Program	ı	•	Find
Address: %WX1	Value:		Format: Decimal	•	
%WX1	10		Decimal	•	

5. Click **Force**. This input/output point or variable is now forced. A list of all forced items is displayed in the Forces dialog box.

Clearing Forced I/O

1. Click **Forces** on the **Debug** menu. A list of all forced items appears in the Forces dialog box.

Clear force	Set force	
Address	Name	Scope
%C1		Program
%C2		Program
%WX1		
%WX9		Program

- 2. To clear all forced items, click **Clear all forces**.
- 3. To clear a single force, select the force you want to clear and then click **Clear Force**.

Tracing SFC Program Flow

In Debug Mode, SFC program flow is automatically displayed in the SFC editor. The current SFC step is indicated by a green arrow.



Additional operations, such as the ability to activate and deactivate a step, set and remove break points, and enable and disable Single Step Mode, are accessible using the buttons along the top of the SFC editor. The buttons are displayed whenever a step in the SFC editor is selected.

MAIN.a	sfc X								Ŧ
Enat	ole Single	e Step	Dea	activate		Set brea	kpoint		
	Α	в	С	D	Е	F	G	Н	I
_									* · · · · ·
_		10							
_		7							
		+						C	2
_		9							
_		T							
_		+							
10									
	•								* •

Additional debug operations related to SFC can be accessed using the **Debug** menu.

• To view a list of SFC breakpoints, click **SFC Breakpoints** on the **Debug** menu.



A SFC Breakpoints dialog box is displayed.

SFC Breakpoints	emove breakpoint		- X
SFC	Scope	Step	
MAIN	Program	S1	
Remove all breakp	oints	C	ose

• To view single step SFCs, click **Single Step SFCs** on the **Debug** menu.



A Single Step SFCs dialog box is displayed.

📕 Single Step SF	Cs	23
Go to SFC	Disable Single Step	
SFC	Scope	
MAIN	Program	

Logging

Logging is enabled by clicking **Enable Logging** on the **Debug** menu.



Once Logging is enabled, a Logging Timeline Control bar is accessible from table editors, CFC and SFC editors, and Watch windows.



The Logging Timeline Control bar is used to start or stop recording, begin playback, pause playback data, and jump to a step. The following figure shows the Logging Timeline Control bar in a custom Watch window.

h3.awch X IO.aios MAIN.asfc			
Edit Monitor Sam	ple Trend		
Record		09	:26:53.935 AM 🖉 Discard
Variable name or address	Scope	Value	Timestamp
%wx1		4	09:26:53.754 AM
%wx9		2	09:26:53.754 AM
%wy1		4	09:26:53.754 AM
%wy9		2	09:26:53.754 AM

The following figure shows the Logging Timeline Control bar in a SFC editor.



Adding an I/O Module for a 505 Series / CTI Controller

Use the following procedure to define an I/O module for a 505 Series or CTI controller: (Note: A controller type must be defined in FTSolution for APT solution order to define an I/O module).

- 1. View a project in Solution Explorer.
- 2. Expand % Modules.



3. Double-click a modules table definition file (the default table definition file is named 505 IO *Modules.amod*). A .amod window is displayed.



- 4. In the .amod window, expand **505 IO**.
- 5. Expand a channel.
- 6. Select a base.



7. In a slot you want to define or edit, select or type a module name in the **Type** box. For example, click **2AO** in slot 1 to add an analog output module.

▲ 505 IO ▲ Channel 1 Base 0 Base 1	Cha IO Base	annel: 1 Base: 0			Base Se	ries: 505	•
Base 2 Base 3	Slot	Туре		Address	Slot	Туре	Address
Base 4 Base 5	1	2AO	Ŧ	49	9	•	0
Base 6 Base 7	2	+	~	0	10	•	0
Base 8 Base 9	3		•	0	11	•	0
Base 10 Base 11	4		•	0	12	•	0
Base 12 Base 13	5		•	0	13	•	0
Base 14	6		•	0	14	•	0
Base 15 Master CTI2500	7		•	0	15	•	0
	8		•	0	16	•	0

- 8. Press **TAB** to move to the **Address** box. The address of the first available I/O eight-point boundary is automatically calculated and displayed in the **Address** box. You can accept the default address or enter a different address in the **Address** box.
- 9. Verify that the configured I/O matches the physical I/O. You can press ALT+S to change the width of the slot if necessary, ALT+I to search for I/O symbol names, ALT+M to move the module, and ALT+D to delete the module.

Defining I/O Points for 505 I/O

The I/O point editor is displayed below the module editor. When a slot is selected in the I/O module editor, its associated points are displayed in the I/O point editor.

Pos	Address	Unit	I/O Name	Description
1	WY49	Program-Level 🔻	•	-
2	WY50	Program-Level 🔻	•	▼
	WY51			(not used)
ł	WY52			(not used)
	WY53			(not used)
	WY54			(not used)
,	WY55			(not used)
	WY56			(not used)

To define or edit an I/O point, select the appropriate point and then enter the following information.

- In the Unit box, select the unit you want to use, or select Program-Level.
- In the I/O Name box, select or enter the I/O symbolic name you want to use.
- In the **Description** box, type a description.

Save the I/O configuration, and the associated I/O points, by clicking **Save** on the **File** menu.

Profibus Configuration

You can use FTSolution for APT to create and manage Profibus networks and devices. A compatible PLC serves as the host of a Profibus master module. (Note: A controller type must be defined in FTSolution for APT solution in order to configure a Profibus network).

If you are importing an APT program that contains Profibus configuration, the Profibus configuration will be automatically generated as part of the import process. You will not be able to edit the Profibus configuration within FTSolution for APT, however, without first adding the required .gsd files to the hardware directory and then performing a merge using the associated .2bf file. (Detailed information about performing a merge can be found on page 70).

Use the following procedure to open Profibus configuration screens within FTSolution for APT:

- 1. View a project in Solution Explorer.
- 2. Expand % Modules.

Solution Explorer 🔹 🗖 🗙
Solution Restorer Solution NasterSolution (2 projects) Statution (2 projects) Statution NasterSolution (2 projects) Statution NasterSolution (2 projects) Statution (2 projects) Sta
Vetch Reports Dryer1 Washer1

3. Open a modules table definition file (the default table definition file is named 505 IO *Modules.amod*). A .amod window is displayed.



- 4. In the .amod window, select the PLC type that is acting as the master module. For example, if you have configured a CTI 2500 controller, select **Master CTI2500**. At this point you can add .gsd files to the Hardware Directory and then add slaves to your Profibus network.
 - The master module has a PROFIBUS address of 1.
 - You can enter a **Station name** and **Host name**. These names can help identify this Profibus network when used with other Profibus networks.

505 IO	Station 1	ype: Master		
▲ Master CTI2500	Host	ype: 555-1104		
[3] ABB Drives NPBA-12	Master	Bus Parameters		
 ▷ [4] Commander SK ▷ [5] Gateway 3WN1 ▷ [6] VLT\$ 5000/6000 ▷ [7] FC300 ▷ [8] Gateway 3WN1 	PROFIE S	US address: 1 tation name:		Host name:
[9] LA-824	Slave	Name	Assigned	Add slave
▷ [10] CP 242-8 ▷ [11] LA-824	3	ABB Drives NPBA-12	Yes	Delete slave
 [12] SIPART DR** [13] ABB Drives RPBA-01 [14] HMI TD17 [16] 2500-RBC Profibus RI 	4	Commander SK	Yes	Merge
	5	Gateway 3WN1	No	Delete all
[17] 505/500-6870 DP RB	6	VLT® 5000/6000	No	
[18] ET 200S (IM151)	7	FC300	No	
				*

Adding GSD files to the Hardware Directory

The Hardware directory displays and organizes the Profibus slave devices available for configuration. Each device in the Hardware Directory is a representation of a device master file (.gsd) that has been added to the directory. The information contained in the .gsd determines how and where the device is displayed in the directory. Once the .gsd associated with a device has been added to the hardware directory, that device may be assigned to the Profibus network as a slave.

When a master module is selected in the 505 IO Modules.amod window, use the following procedure to add .gsd files to the Hardware Directory within FTSolution for APT.

1. Click Add slave. The Hardware Directory window appears.

X Hardware Directory	
Controllers Drives Encoders Gateways Gateways Ganeral HMI H/I I/O Ident Systems NC/RC Profibus PA Switching devices Valves	
Add Ha	rdware Remove Hardware
GSD filename:	
Vendor:	
Model:	
	Add Slave Close

- 2. Click Add Hardware.
- 3. **Browse** to the location that contains the .gsd file(s) of the devices you want to add and select them.
- 4. Click **Open** to add the .gsd file(s) to the Hardware Directory. When adding one or more .gsd files to the Hardware Directory, the files are scanned for unsupported or unrecognized information. If any unsupported information is found, a warning message appears, allowing you to view an event log for more details about the problem. In most cases, the device is still added to the Hardware Directory, with any unsupported features ignored. Contact your device manufacturer for more information.

Assigning Slaves to the Profibus Configuration

When a master module is selected in the 505 IO Modules.amod window, use the following procedure to assign slaves to the Profibus configuration:

1. Click Add Slave. The Hardware Directory appears.

X Hardware Directory					
Controllers Drives Encoders Gateways General				•	
→ HMI ▲ VO →					
2500-RBC Pro 2500-RBC Pro 2500-RBC Pro 2500-RBC Pro 2500-RBC Pro 2500-RBC Pro 2500-RBC Pro ≥ SIMATIC ▷ B&R ▷ ET200S	fibus RBC fibus RBC fibus RBC Slave Module fibus RBC fibus RBC				
	Add Ha	irdware	Remove	Hardware	
	GSD filename:	2500RBC-V	L.GSD		
	Vendor:	Control Tec	hnology Inc.		
	Model:	2500-RBC P	rofibus RBC		
		Add	Slave	Close	

- 2. Browse the Hardware Directory to find and select the slave device you want to add. When you select a device, information about the device is displayed in the **GSD filename**, **Vendor**, and **Model** boxes.
- 3. Click Add Slave to add the selected device to the Profibus Configuration.

505 IO Modules.amod* ×				
 505 IO Master CTI2500* [2] C-H SV9000* [3] ABB Drives NPBA-12* [4] Commander SK* [5] Gateway 3WN1* [6] VLT & 5000/6000* [7] FC300* [8] Gateway 3WN1* 	Devi Slave PROF Module	Parameters IBUS address: Station name: Is:		
⊳ [9] LA-824*	Slot	Module	Addre	Insert module
 ▷ [10] CP 242.8* ▷ [11] LA-824* ▷ [12] SIPART DR **** ▷ [13] ABB Drives RPBA-01* ▷ [14] HMI TD17* [15] 2500-RBC Profibus RBC ▷ [16] 2500-RBC Profibus RBC ▷ [17] 505/500-6870 DP RBC* ▷ [18] ET 200S (IM151)* 	1	6ES7 134-4NB50-0AB0 2AI RTI End Slave Configuration	417	Delete module $X \triangleleft - \triangleright WX$ $Y \triangleleft - \triangleright WY$ Unify Compact
- III	4	III	+	Restore

- The first available PROFIBUS address is automatically entered in the **PROFIBUS** address box. If you want, you can type a **Station name** to help identify this device.
- 4. To configure parameters for the device, select the device you just added under the Master module, click the **Parameters** tab, and then enter the necessary information. Not all devices will have configurable parameters.

Once a slave is added, you need to configure the modules installed in the slave, and specify the addresses used by the slave's modules.

Configuring and Addressing Profibus I/O Modules

1. Expand the master module in the 505 IO Modules.amod window.



2. Select the slave device you want to configure.

505 IO Modules.amod* 🗙			
 > 505 IO Master CTI2500* > [2] C-H SV9000* > [3] ABB Drives NPBA-12* > [4] Commander SK* > [5] Gateway 3WN1* > [6] VLT @ 5000/6000* > [7] FC300* > [8] Gateway 3WN1* 	Device File: SI02806A.GSD Slave Parameters PROFIBUS address: 18 Station name: Modules:		
▶ [9] LA-824*	Slot Module	Addre	Insert module
▷ [10] CP 242-8* ▷ [11] LA-824* ▷ [12] STRAFT DP ***	1 6ES7 134-4NB50-0AB0 2AI RTL	417	Delete module
[12] SIPART DR [13] ABB Drives RPBA-01*	End Slave Configuration		$X \triangleleft - \triangleright WX$
 [14] HMI TD 17* [15] 2500-RBC Profibus RBC 			
 [16] 2500-RBC Profibus RBC [17] 505/500-6870 DP RBC* 			Unify
▷ [18] ET 200S (IM151)*			Compact
	<	×.	Restore
< <u> </u>			

- 3. Click the **Slave** tab.
- 4. Click **Insert Module** to add a module for the selected slave device. An Insert Module dialog box appears, which displays modules for the slave.

6ES7 131-48B00-0AB0 2DI DC24V	
6ES7 131-48B00-0AB0* 2DI DC24V	
6ES7 131-4BB00-0AA0 2DI DC24V	=
6ES7 131-4BB00-0AA0* 2DI DC24V	
6ES7 132-48B00-0AB0 2DO DC24V	
6ES7 132-48B00-0AB0* 2DO DC24V	
6ES7 132-48B00-0AA0 2DO DC24V	
6ES7 132-4BB00-0AA0* 2DO DC24V	
6ES7 132-48B30-0AB0 2DO DC24V	
6ES7 132-48B30-0AB0* 2DO DC24V	
6ES7 132-48B30-0AA0 2DO DC24V	
6ES7 132-48B30-0AA0* 2DO DC24V	
6ES7 132-4HB00-0AB0 2DO Rel.	
6ES7 132-4HB00-0AB0* 2DO Rel.	
6ES7 132-4HB10-0AB0 2DO Rel.	
6ES7 132-4HB10-0AB0* 2DO Rel.	
6ES7 131-48D00-0AB0 4DI DC24V	
6ES7 131-48D00-0AB0* 4DI DC24V	
6ES7 131-4BD00-0AA0 4DI DC24V	
6ES7 131-4BD00-0AA0* 4DI DC24V	
6ES7 131-48D50-0AA0 4DImDC24V	
6ES7 131-48D50-0AA0* 4DImDC24V	*

- 5. Select the module you want to add, and then click **Add Module**. The Module is added to the slave.
- 6. For each slot, enter the address you want to use in the **Address** box.

505 IO Modules.amod* 🗙		
505 IO Master CTI2500*	Device File: SI02806A.GSD Slave Parameters	
 ▷ [3] Gateway 3001" ▷ [6] VLT ® 5000/6000* ▷ [7] FC300* ▷ [8] Gateway 3WN1* ▷ [9] LA-824* 	Station name: Modules:	
▷ [10] CP 242-8* ▷ [11] LA-824*	Slot Module Address X Y WX WY Comment 1 6ES7 131-48B00-0AA0 0 8 0 0 0	Insert module
 ▷ [12] SIPART DR*** ▷ [13] ABB Drives RPBA-0 ▷ [14] HMI TD 17* ▷ [15] 2500-RBC Profibus ▷ [16] 2500-RBC Profibus ▷ [17] 505/500-6870 DP R ▲ [18] ET 200S (IM151)* 	End Slave Configura	X <-> WX Y <-> WY Unify Compact
[0] 6ES7 131-4BB00		Restore

- Click X <-> WX to toggle the image register type for the selected module between discrete and word inputs. To ensure that discrete are toggled to words, the discrete must be on a 16-bit boundary.
- Click **Y** <-> **WY** to toggle the image register type for the selected module between discrete and word outputs. To ensure that discrete are toggled to words, the discrete must be on a 16-bit boundary.
- Click **Unify** to move all modules for the selected slave into the first module. Unification is only allowed if all modules are either discrete or words. Only the first address is retained.
- Click **Compact** to move all discrete image register types to word image register types for every module of the selected slave. Then, all modules are unified into the first module. To ensure that discrete are toggled to words, the discrete must be on a 16-bit boundary. Only the first address is retained.
- Click **Restore** to restore modules for the selected slave to their initial configuration, including the image register types. Zero addresses are assumed for all but the first address, which is kept.
- If you have added a Smart Connect slave device, click **Smart connect** to configure it.
- You can press **ALT+I** to search for I/O symbol names and **ALT+D** to delete the module.
- 7. Optional: Enter a comment in the **Comment** box.
- 8. To configure parameters for the module, click the **Parameters** tab, and then enter the necessary information. Not all modules will have configurable parameters.

Device File: SIEMBCD0.GSD					
Sidve	Faranieters				
Offse	t Parameter Name	Value			
0	Discrete I/O Interval msec	1			
1	Word I/O Update Factor	2			
2.0	50X Mismatch Mode	Disable 🔹			
2.1	50X RS232 Comm Port	Disable 💌			

Defining I/O Points in Profibus I/O

Defining an I/O point consists of entering an I/O symbolic name and information for each address you want to access. To begin defining an I/O point go to the I/O Module editor and select a module to display the I/O Point Editor.

505 IO Modules.amod* ×											
505 IO Master CTI2500* ▷ [2] C-H SV9000* ▷ [3] ABB Drives NPBA-12 ▷ [4] Commander SK* ▷ [5] Gateway 3WN1* ▷ [6] VLT® 5000/6000* ▷ [7] FC300* ▷ [6] Cateway 3WN1*	Devi Slave PROF Module	e File: Si Parameter IBUS addre Station nan s:	IEMBCD0.GSD s ss: 17 ne:								
▷ [9] LA-824*	Slot	M	odule	Addre	ss X	Y	WX	WY	Comment	Insert module	
↓ [11] LA-824*	0	CTI-2599	as 8DO AC C	321	0	8	0	0		Delete module	٦.
 [12] SIPART DR*** [13] ABB Drives RPBA-0 [14] LIME TD 178 		End Sla	ive Configura							x ⊲−⊳ wx	1
 [15] 2500-RBC Profibus [16] 2500-RBC Profibus [17] 505/500-6870 DP R 	Slave	17, Slot 0									Aut
[0] 6ES7 131-48B00	Pos	Address	Unit		I/O	Name			Description		
	1	Y321	Program-Lev	el 🔻	IO112		•				
	2	Y322	Program-Lev	el 🔻	LOOP1		•				
	3	Y323	Program-Lev	el 🔹			•				
	4	Y324	Program-Lev	el 🔻			•				E
	5	Y325	Program-Lev	el 🔹			•				
	6	Y326	Program-Lev	el 🔻			•				
	7	Y327	Program-Lev	el 🔹			•				
۰ III	8	Y328	Program-Lev	rel 🔻			•				-

To define or edit an I/O point, select the appropriate point and then enter the following information.

- In the **Unit** box, select the unit you want to use, or select **Program-Level**.
- In the I/O Name box, select or enter the I/O symbolic name you want to use.
- In the **Description** box, type a description.

Save the I/O configuration, and the associated I/O points, by clicking **Save** on the **File** menu.

Merging a .2bf File

After importing a program with Profibus configuration, the Profibus I/O editor will be in limited editing mode. The Profibus configuration cannot be changed in this mode. To enable full editing, the original .2bf file must be merged after the required .gsd files are added to the hardware directory.

To merge a .2bf file:

- 1. Import an APT program.
- 2. Expand %Modules in the Solution Explorer and open the 505 IO Modules.amod file.
- 3. Select the PLC that is acting as the master module.
- 4. Click Merge.
- 5. During the merge, you may be asked to confirm the name of modules. Select the appropriate module in the Confirm Module Name dialog box.



Viewing and Editing I/O Symbolic Names

You can use FTSolution for APT to view and edit the I/O symbolic names used in your FTSolution for APT project. When a project is displayed in the Solution Explorer, you can open the default IO table definition file (named IO.aios), by expanding **IO** and then opening **IO.aios**.



You can then add new I/O symbols, edit existing symbols, or delete symbols that are not needed.

IO.aios* 🗙				
🐈 New I/O Symbol 📝	Edit 👩 Delete			
Name	Туре	Descriptior	Address	
101	Analog Input			
102	Word Input		%WX33	
103	Thermocouple			
10112	Digital Output		%Y321	
LOOP1	Digital Output		%Y322	
UNNAMED1	Analog Input			
UNNAMED2	INNAMED2 Analog Input			
- New I/O Symbol -				
Settings for I0112				
Name	e: 10112			*
Туре	Digital Outp	ut	•	=
Description	1:			
				*

FTSolution for APT Hardware Debugger

FTSolution for APT Hardware Debugger is a sophisticated PLC hardware debugger utilizing ladder language-based tools. FTSolution for APT Hardware Debugger allows users to download compiled FTSolution for APT programs into Siemens 505/CTI 2500 Series processors, utilize sophisticated ladder language-based debugging tools, and limit the ability to modify existing programs.

Creating an .AHD file in FTSolution for APT

To use FTSolution for APT Hardware Debugger you must create a Hardware Debugger file (.ahd file) from within FTSolution for APT. To create an .ahd file for use within Hardware Debugger you must compile your project. During a compile select the option **Create Hardware Debugger file**.

🏋 Compile	Test Project
Target:	CTI 2500 C400 < 8.01
Translate:	 ● <u>N</u>o ● <u>A</u>ppend ● <u>Y</u>es
Options:	Debug Oreate Hardware Debugger File Force
	Compile Cancel

The .ahd file will automatically be placed within the Compile folder of the applicable Solution and Project. For example, the Test Project.ahd file picture below for **Test Project** is located within the folder **\Station_53 \Test Project\Compile**.



Opening an .AHD file in FTSolution for APT

To open the .ahd file within FTSolution for APT open the **Compile** item within the Solution Explorer and double click the **.ahd** file. The Hardware Debugger will launch. In the example below the .ahd file is **Test Project.ahd**.



Opening an .AHD file in Hardware Debugger

You can also open the file within Hardware Debugger.

- 1) Select the **File/Open** menu click the 🖻 toolbar icon, or press [Ctrl-O]. The Open Program dialog appears.
- 2) Click **Browse** and select the *.AHD file.
- 3) Select the Offline radio button or enter information into Connect to PLC. Click Ok.

Progra	m File:	Browse				
Addr	ess Documentation					
Path	r 🗌	Advanced.				
Prog	ram Type					
•	Offline					
C I	Connect to PLC					
	Connection Method	î				
ן ר ר	(TCP/IP)	Setup				
	Use File Associated Connection Settings					
	Options					
	Transfer Logic to PLC					
	E Deservicione					

WARNING: DO NOT use Hardware Debugger to modify an FTSolution for APTcreated .AHD file. An FTSolution for APT-created .AHD file contains unique elements which must be downloaded to a PLC to correctly run an APT language program. Saving an .AHD file with Hardware Debugger omits these critical changes.

FTSolution for APT Simulator (505 Simulator)

FTSolution for APT Simulator (505 Simulator) is designed to run and test APT projects without the use of a PLC.

Starting 505 Simulator

To start the 505 Simulator click the **Start** button, click **All Programs**, click **FasTrak SoftWorks**, and then click **505 Simulator**.

After starting, the 505 Simulator Window is displayed:

📓 505 Simulator	
File Options Help	
PLC Status Current PLC Status Ladder Mode Program Loop Mode Program	PLC Type Selection 555 - 1106 Accept Current PLC Type 555 - 1106
🔽 Following Ladd	er Mode
Connection Status Not Connected PLC Scan Status	Timing Mode Real Time
Ready to change operat	ional mode.

Selecting PLC Type

- 1. In the **PLC Type** Selection box, select the PLC type you want to simulate.
- 2. Click Accept. The following warning appears:

505 Sim	ulator	
?	WARNING: This will clear Are you sure you wish to	your PLC's memory and configuration! continue?
	Yes	No

3. Click **Yes**. The simulator is now simulating the PLC type you selected.

Setting Up Communications with 505 Simulator in FTSolution for APT

505 Simulator uses the TCP/IP communication method to communicate with FTSolution for APT. Do the following to create a new TCP/IP connection in FTSolution for APT to connect to the 505 Simulator:

In FTSolution for APT, open the project and click **Communications Setup.acom** from the Solution Explorer. The Communication Setup dialog box appears. Select **TCP/IP** from the **Communication Type** list. The TCP/IP setup information appears.

X Communication Setup	×
Communication type:	TCP/IP
IP address:	127 0 0 2
TCP port:	1505
Response timeout (sec):	6
Retries:	3
	Use CAMP
	Vse packed opcodes
	Save Caecel
	Save

In the **IP Address** box, enter **127.0.0.1**. In the **TCP port** box, enter **1505** in the TCP Port box. The TCP Port must be set to 1505 to enable connection with the simulator. The other communication settings can be changed as needed; refer to the FTSolution for APT online help for additional information.

Click Save.

Connecting to 505 Simulator

Prerequisites

In order to connect to the 505 Simulator, both FTSolution for APT and 505 Simulator must be currently running. In addition, within FTSolution for APT, a TCP/IP communication method that uses an IP address of 127.0.0.1 must be created.

Connecting

Once these prerequisites have been met, connecting to the simulator is identical to going online with an actual PLC. See *Starting Debug Mode* on page 49 for additional information on how to enter run mode.

Note: 505 Simulator must be running before attempting to connect to it.

FTVersionTrak for APT

FTSolution for APT provides for integration with source control, or version control, software through the use of FTVersionTrak for APT. Contact your FasTrak SoftWorks, Inc. Sales Representative at **262**-**238-8088** or **sales@fast-soft.com** to purchase FTVersionTrak for APT. To use the functionality provided by FTVersionTrak, make sure that FTVersionTrak is listed as the designated source control plug-in for FTSolution for APT.

To specify the source control plug-in:

- 1. Start FTSolution for APT.
- 2. Click **Options** on the **Tools** menu.
- 3. In the Options dialog box, select **Source Control**.
- 4. Select **FTVersionTrak** as the **Current source control plug-in** and click **OK**.

Options	
 Environment Source Control 	Plug-In Selection:
Plug-in Selection FTVersionTrak Options	Specifies the source control plug-in to use with APT replacement product and allows changes to plug-in specific options.
	Current source control plug-in:
	FTVersionTrak 🔹
	OK Cancel

FTVersionTrak Toolbar

Most of the source control-related operations you will need to perform in FTSolution for APT can be accomplished using the FTVersionTrak toolbar. If the toolbar is not displayed, click the **View** menu, point to **Toolbars**, and select **FTVersionTrak Toolbar**.

i 👧 D, 🛿	F 🏷 🛠 🕙 😭 🖻 🔍 🖗 🤻 🖗
b	Get Latest Version. Use this button to retrieve the latest repository version of the selected items in the Solution Explorer.
D 	Get. Use this button to retrieve a user-specified repository version of the selected item in the Solution Explorer.
	View Repository Properties. Use this button to view the repository properties of the selected item in the Solution Explorer.
*	Check Out. Use this button to check out the selected items in the Solution Explorer.
٧	Undo Check Out. Use this button to undo a check out.
*	Check In. Use this button to check in the selected items in the Solution Explorer.
()	History. Use this button to view the history of the selected items in the Solution Explorer.
Ŕ	Compare. Use this button to compare the item currently selected in the Solution Explorer.
\$	Refresh. Use this button to refresh the display.
ū,	Disconnect. Use this button to disconnect from the repository.
P	Launch FTVersionTrak. Use this button to launch FTVersionTrak.
Ŗ	Bind. Use this button to bind an FTSolution for APT project to a repository.
Binding to a Repository

The first step in using FTVersionTrak with FTSolution for APT is to bind an APT project to an FTVersionTrak repository. Binding a project creates a link between the working files of a solution and a repository in the FTVersionTrak database. Instructions for creating a repository were provided earlier in this document, in the procedure for installing FTVersionTrak.

To bind an FTSolution for APT project to a repository:

- 1. Open the solution you want to bind.
- 2. Click the **Bind** toolbar button on the FTVersionTrak toolbar.
- 3. In the Connect to Repository dialog box, select a repository and click **Bind**. You will bind the FTSolution for APT project to this repository.

С	onnect to Repository			? 🔀
	Select a Repository		-0	
	Server	Name	Last Access	Bind
	FTCLOUDVM\FTVERSIONTRAK	Repository 1		Cancel
	FTCLOUDVM\FTVERSIONTRAK	R1		
				Browse

4. In the Add to FTVersionTrak Repository dialog box, click **Bind** to bind the FTSolution for APT project to the root folder of the repository. If you want, you have the ability to bind the project to another folder on the repository.

Add to FTVersiontrak Repository 'Repository 1'	? 💌
Solution Path:	
c:\users\user\documents\apt replacement product\Project4\	
Bind to: 🛄 \$	
· ····································	
Comment:	
Bind	Cancel

Additional icons are displayed in the Solution Explorer of an APT replacement project that has been successfully bound to a repository. These icons are used to indicate the status of the files under FTVersionTrak control. The following figure shows icons that represent the checked-in and checked-out status of the project, solution, and files.

Solution Explorer	* 🗆 X
Solution 'Project4' (1 project)	
🔒 Checked-ir	ı.

Checked-out.

Additional icons are used to represent other statuses.

Setting FTVersionTrak Options

There are many customizable settings that determine how FTVersionTrak operates within FTSolution for APT. Examples of these settings include whether check-ins and check-outs are performed automatically, what occurs when files are deleted, and what restrictions are in place when working offline.

To specify FTVersionTrak options:

- 1. Click **Options** on the **Tools** menu in FTSolution for APT.
- 2. In the Options dialog box, in the left pane, expand **Source Control**.
- 3. Select FTVersionTrak Options.

Environment			
 Source Control Plug-in Selection 	Check In all files when closing a solution	on	
FTVersionTrak Options	 Show Check Qut Options Show Compare Options Show Check In Options 	Show Undo Checkout Options Show History Options	
	On Edit Prompt for checkout icheckout automatically 	On <u>U</u> ndo Checkout Show <u>w</u> arning for modified file Und <u>o</u> checkout automatically	
	On Delete Prompt to delete from repository Delete from repository automatically 	When Working Offline Disable delete repository file Disable rename repository file	

4. Select the options you want to use.

Using Solution Explorer with FTVersionTrak

The operation of many FTVersionTrak operations will vary depending on what is selected in the Solution Explorer. When a project, solution, or unit is selected in the Solution Explorer, the history and compare commands will show information about the entire project, solution, or unit. When an individual file is selected, the history and compare commands will show detailed information about that specific file.

Similarly, the Check-In and Check-Out commands will also work on a project, solution, object, and unit level. For example, when a solution is selected, all of the files contained in that solution will be checked in or checked out at once. If only a single file is selected in Solution Explorer, only that file will be checked in or out.

Example of Compare: Comparing at the Project Level

If a project or solution is selected in the Solution Explorer, clicking the **Compare** toolbar button provides a way to compare the current version of a project or solution with the most recent version in the repository.

The following figure shows the result of a compare project done on a project named **flipper1**. In this comparison, the **Declarations.adec** and **Devices.adev** files do no match the latest repository versions.



Example of History: Historical Information for a Single File

When a single file is selected in the Solution Explorer, such as **Devices.adev**, clicking the **History** toolbar button provides detailed historical information for that file.

The following illustration shows the historical information for the Devices.adev file.

© \$/fl	ipper	1/flipper1/Devices/[Devices.adev		?	x
	No.	Usemame	Computer	Date	Compare	
	6	FTCLOUDVM\User	FTCLOUDVM	1/29/2014	Get	
	5	FTCLOUDVM\User	FTCLOUDVM	1/29/2014		
	4	FTCLOUDVM\User	FTCLOUDVM	1/29/2014		
	3	FTCLOUDVM\User	FTCLOUDVM	1/29/2014		
	2	FTCLOUDVM\User	FTCLOUDVM	1/29/2014		
	1	ETCLOUDV/M\User	ETCLOUDV/M	1/20/2014		
< □	_	III		•	Close	

From this dialog box, it is possible to compare versions of **Devices.adev**. For instance, selecting **No. 4** and then clicking **Compare** will compare the current version with the **No. 4** version in the repository. It is also possible to perform a get operation from this dialog box. Selecting **No. 4** and then clicking **Get** will retrieve the **No. 4** version from the repository and place it in the working folder.

PCS Tagging – Marking Objects for Translation

FTSolution for APT supports the ability to mark simple objects (such as declarations) as well as complex objects (such as CFBs, recipes, or declarations with more than one extension). When marking a complex object, you are able to edit the object and mark the individual extensions that require translation. You are also able to specify an object's attributes and specify the process group to which a tag belongs.

To mark an object for translation, select the item in Solution Explorer and then type the letter **K**. In the following example illustration, the file **IO.aios** is selected in Solution Explorer.



The following items may contain objects that can be marked for translation:

- .aios files (IO.aios is the default)
- .adev files (Devices.adev is the default)
- .adec files (Declarations.adec is the default)
- .arut files (Recipes.arut is the default)
- .acfc files (only CFCs files that certain CFC blocks can be marked for translation)

You can also click the **Mark PCS tags** button to mark an object for translation. The **Mark PCS tags** button appears on several table file definition and configuration screens, as well as the graphical editor for CFCs. The following illustration shows the **Mark PCS tags** button in the **IO.aios** configuration screen.

IO.aios		_	- □×
+ New I/O symbol / Edit		🖉 Delete 🛛 M	lark PCS tags
Name	Туре	Description	Address
METER1	Analog Input	NORTH METER	%WX1476
VALVE1	Digital Output	Valve command	%Y2284
DIG_IN1	Digital Input	WAIT FOR USER	%X2136
DIG_IN2	Digital Input	PROMPT USER	%X2124
DIG_IN3	Digital Input	PROMPT USER	%X396
WORD_OUT_:	Word Output	Word out 8	
E3003 1003			
Object: M	ETER1		Settings
		Name: M	IETER1
		Type: A	nalog Input 🔹
		Description: N	ORTH METER
		_	

Note: To mark an item that is part of a unit, open the unit in Solution Explorer, select the item, and then type the letter **K**. Or, click the **Mark PCS tags** button within the appropriate Unit configuration screen.

After pressing the letter **K** (or clicking the **Mark PCS tags button**), the **Mark PCS Tags** dialog box is displayed. In this dialog box, select the check boxes of the objects you wish to mark for translation. The **Mark PCS Tags** dialog box is equivalent to the Tag Translate editor and sub-editor in APT.

📜 Mark PCS Tags		X
Object	PCS tag information	
METER1 NORTH METER Analog Input		*
VALVE1 Valve command Digital Output		E
DIG_IN1 WAIT FOR USER Digital Input		
DIG_IN2 PROMPT USER Digital Input		
DIG_IN3 PROMPT USER Digital Input		
WORD_OUT_1 Word out 8		-
	0	K Cancel

After selecting the check box of an object, you can enter additional information in the **PCS tag information** column. The information you are able to enter will vary depending on the type of object selected. For example, the following illustration shows the **Mark PCS Tags** dialog box for objects in the **IO.aios** file.

Object	PCS tag information	
METER1 NORTH METER Analog Input		Process groups Extensions
VALVE1 Valve command Digital Output	Name: VALVE1 Desc: Type: DO	Manual set <u>Attributes</u> Process groups
UIG_IN1 Auto user request Digital Input	Name: DIG_IN1 Desc: Type: DI	Manual set <u>Attributes</u> Process groups
DIG_IN2 Auto user request Digital Input		
DIG_IN3 Auto user request		

In the **Mark PCS tags** dialog box, the objects named METER1, VALVE1, and DIG_IN1 are all marked for translation. For the VALVE1 and DIG_IN1 objects, you can type a new name for the PCS tag in the **Name** box and a description for the PCS tag in the **Desc** box.

The first object, METER1, is a complex object. Click the **Extensions** link to mark the individual extensions associated with the complex object. The following illustration shows the extensions associated with the METER1 object.

Object	PCS tag information					
V METER1. NORTH METER Other	Name: 5221_FT0194. Desc: Type: CALC Deadband: 32000	Manual set <u>Attributes</u> <u>Process groups</u>				
METER1.RAW NORTH METER Real	Name: 5221_FT0194.RAW Desc: Type: AI Deadband: 32000	Manual set <u>Attributes</u> <u>Process groups</u>				

To specify the attributes of an object or an object's extension, click the **Attributes** link. After clicking the **Attributes** link, select the check boxes of the attributes you want to use in the **PCS Attributes** dialog box. The PCS Attributes dialog box is equivalent to the tag attributes sub-editor in APT.

T PCS Attributes					×
Attribute	Scan	Upload	AutoLog	20%	
Status					*
Commanded Value					
					*
				Clo	se

The attributes you can specify consist of the following:

- Scan
- Upload
- Autolog
- 20% offset

By default, certain attributes may already be selected, depending on the type of object.

🕱 Mark PCS Tags Object PCS tag information METER1 NORTH METER Analog Input Process groups Extensions VALVE1 Name: VALVE1 Manual set Valve comman Digital Output Attributes Desc: Process groups Type: DO DIG_IN1 Auto user request Digital Input Name: DIG_IN1 Manual set Attributes Desc: Process groups Type: DI DIG_IN2 Auto user request Digital Input DIG_IN3 Auto user request OK Cancel

To specify the use of manual settings, select the **Manual set** check box.

To indicate the process groups to which the object is a member, click the **Process groups** link.

	Group	Group Name	Group	Group Name
1	1		17	
1	2		18	
v	3		19	
J	4		20	
1	5		21	
1	6		22	
V	7		23	
V	8		24	
1	9		25	
V	10		26	
V	11		27	
V	12		28	
1	13		29	
V	14		30	
V	15		31	
1	16		32	

In the **Include tag in Process Groups** dialog box, the check boxes associated with the groups where the tag will be included are selected. By default, all tags are initially assigned to all 32 process groups. To remove a tag from a process group, clear the check box. The **Include tag in Process Groups** dialog box is equivalent to the process groups sub-editor in APT.

Tag Translation and Compile Options

The compile options under **Translate** in the Compile dialog box have a direct effect on PCS tagging. Under **Translate**, specify the translate option you want to use:

🕱 Compile	201	×
Target:	SIMATIC 555	4.x
Translate:	 No Append Yes 	
Options:	Debug	
	Compile	Cancel

- When you select **No**, existing marked tags are not modified.
- When you select **Append**, existing marked tags are placed in the *install.tag* file. Tags marked since the last translate build and tags that were unmarked and re-marked are also placed in a separate file, *append.tag*.
- When you select **Yes**, all marked tags are placed in the *install.tag* file.

Location of Install.tag and Append.tag Reports

Once created, the *install.tag* and *append.tag* files appear in the Solution Explorer within FTSolution for APT. The *install.tag* file is displayed in the following illustration.



Using File Explorer, the *install.tag* and *append.tag* files are also accessible from the Reports folder within the project's folder.



www.fast-soft.com 262.238.8088