SIEMENS

SIMATIC 545

Technical Product Description

Order Number: PPX:545–8103–3
Text Assembly Number: 2592506–0003
Third Edition

Copyright 1992 by Siemens Industrial Automation, Inc.
All Rights Reserved — Printed in USA

Reproduction, transmission or use of this document or contents is not permitted without express consent of Siemens Industrial Automation, Inc. All rights, including rights created by patent grant or registration of a utility model or design, are reserved.

Since Siemens Industrial Automation, Inc. does not possess full access to data concerning all of the uses and applications of customer's products, we do not assume responsibility either for customer product design or for any infringements of patents or rights of others which may result from our assistance.

MANUAL PUBLICATION HISTORY

SIMATIC® 545 Technical Product Description
Order Number: PPX:545–8103–3
Refer to this history in all correspondence and/or discussion about this manual.

Event	Date	Description
Original Issue	03/90	Original Issue (2592506–0001)
Second Edition	08/90	Second Edition (2592506–0002)
Third Edition	09/92	Third Edition (2592506–0003)

LIST OF EFFECTIVE PAGES

Pages	Description	Pages	Description
Cover/Copyright	Third Edition		
History/Effective Pages	Third Edition		
iii — xix	Third Edition		
1-1 — 1-6	Third Edition		
2-1 — 2-57	Third Edition		
3-1 — 3-10	Third Edition		
4-1 — 4-63	Third Edition		
5-1 — 5-19	Third Edition		
6-1 — 6-13	Third Edition		
7-1 — 7-13	Third Edition		
8-1 — 8-23	Third Edition		
A-1 — A-11	Third Edition		
B-1 — B-20	Third Edition		
C-1 — C-22	Third Edition		
D-1 — D-9	Third Edition		
Registration	Third Edition		

Contents

Chapter 3 Input/Output System

Chapter 4 I/O Module Data Sheets

Chapter 5 Programming

Chapter 6 TIWAY Networking

Chapter 7 Direct-Connecting Operator Interface

Chapter 8 Networkable Operator Interfaces

Appendix C Task Codes

Appendix D Communication Protocols

List of Figures

List of Tables

Preface

This Technical Overview provides detailed information on the SIMATIC® 545 programmable control system. Topics include:

- SIMATIC 545 505 Overview
- Controller Description
- I/O Module Subsystem Overview
- I/O Module Data Sheets
- Programming Options
- TIWAY™ Networking
- Operator Interface Products

This document is intended to be used as a technical introduction to the Siemens Energy & Automation product line. It is not intended to provide detailed product information required during installation or operation of the system. Use the appropriate product manuals, and follow all state and local codes to configure and operate your control system.

Chapter 1
Overview

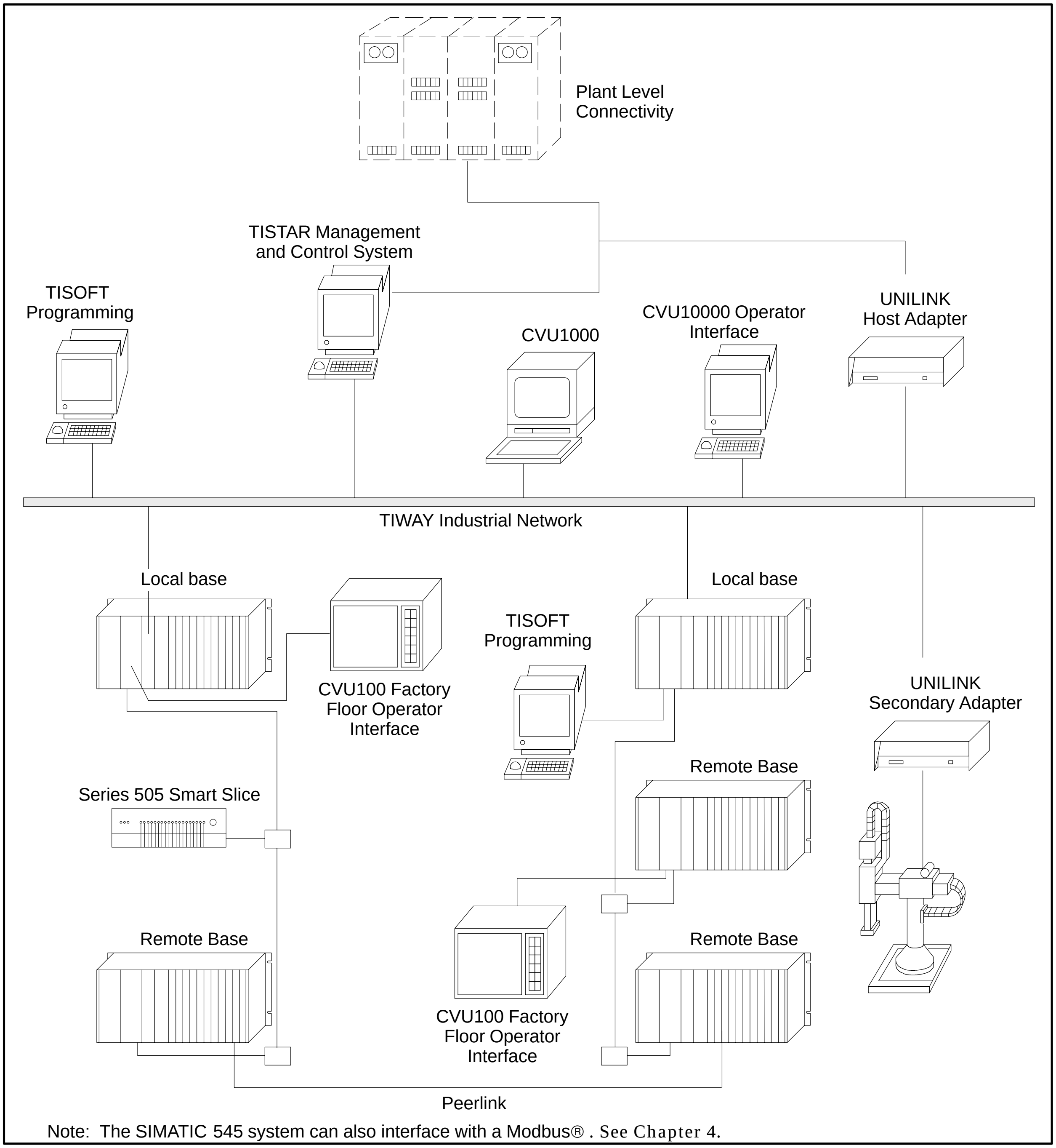
1.1 System Overview

You can use the SIMATIC 545 controller with the SIMATIC® 505 or SIMATIC® 500 Input/Output (I/O) subsystem components. It is best suited for control applications requiring up to 2048 I/O points. A single controller can support up to 16 I/O bases (1 local and 15 remote), which can be located up to 3300 feet from the controller.

The TIWAY local area network links multiple SIMATIC 545 and other Series 505 or Series 500 controllers. These controllers can also be linked together with the CVU™ Operator Interfaces and supervisory or cell computers in a local area network for larger control applications. Peer-to-peer communication is also available with Peerlink™.

TISOFT2™ programming software is used to develop, document, and maintain your SIMATIC 545 control program.

NOTE: The purpose of this manual is to provide an overview for SIMATIC 545 products and is not an installation, operating, or programming manual. For more comprehensive and possibly more up-to-date information on SIMATIC 545 products, consult the appropriate manual.



I000561

Figure 1-1 SIMATIC 545 System Layout

1.2 System Components

Controller

The controller is a high-performance, microprocessor-based PLC. It is well suited for applications requiring any or all of the following types of control.

- High-speed sequencing or logic execution
- Closed-loop or Proportional Integral Derivative (PID) control
- Analog alarm monitoring
- Complex mathematical operations
- Extensive Data Handling

The controller executes user-generated programs to make control decisions based on the state of input sensors (e.g., pushbuttons, limit switches) and control output actuators (e.g., relays, solenoids, motors). The user control programs consist of Relay Ladder Logic (RLL) routines, pre-configured PID and analog alarm blocks, and high-level language Special Function (SF) programs.

I/O Subsystem

The I/O subsystem consists of one or more multi-slot I/O racks, with system power supplies and individual I/O modules. In a typical system, the controller resides in the controller slot of an I/O rack (referred to as the local base).

The local base also contains a power supply and up to 16 individual I/O modules. Bases are available with 4, 8, or 16 slots. Up to 15 additional I/O bases may be connected to the SIMATIC 545 via a built-in remote I/O communications channel.

The I/O subsystem provides a full range of I/O modules to fit your control needs. In addition to discrete, analog, and word I/O, intelligent modules are available for special control tasks such as high-speed counting and communications.

Programming

TISOFT is the factory-floor programming software providing the means for directly entering, debugging, and documenting your ladder logic or special function program. TISOFT is menu-driven, provides block copy and edit functions, both ladder logic and special function programming capabilities, and allows you to add comments to instructions.

TISOFT can run on an IBM® PC/AT®, PC/XT® computer (or a 100% compatible), the CVU 1000/10000,or a TISTAR™ operator station.

The Applications Productivity Tool (APT) is a Computer Aided Software Engineering (CASE) product that shortens the expensive and time-consuming program design task. APT is a graphical, window-oriented tool that supports both continuous and sequential control design. Like TISOFT, APT can also run on an IBM computer (or a 100% compatible), the CVU, or the TISTAR operator station.

Networking

The TIWAY Network allows you to link the pieces of your total control system to supervisory and cell level computers for larger control applications.

The SIMATIC 545 system can also interface with a Modbus® network via a 505 MODNIM. See Chapter 4.

Operator Interfaces

Siemens provides a choice of operator interfaces for monitoring and controlling a SIMATIC 545 control system. The CVU Operator Interface Family provides a choice of control room or factory floor hardware platforms. The CVU1000 and CVU10000 are configurable, fill-in-the-blank systems that provide high performance monitoring and control interfaces for a wide range of discrete and process applications. The CVU1000 and CVU10000 are based on standard MS-DOS® hardware platforms and provide high resolution graphics, combined with less than one second response time and the following capabilities.

- Mimicking the Manufacturing Application
- Displaying and Monitoring of Control Data
- Loop Tuning with Continuous Trending
- Reporting and Recording of Alarm Conditions
- Downloading Recipes or Control Parameters
- Continuous Trending of Process Data
- Data Logging and Archiving
- Historical Trending and Datalogging
- Generating Shift or Batch Reports
- Selecting from Three Levels of Password Protection

The CVU100 Operator Interface is designed to meet the harsh operating conditions of the factory floor. The 35-position integral keypad provides fast and accurate access to any data in a Series 505 programmable controller. The CVU100 can be panel- or rack-mounted in a NEMA 4 or NEMA 12 enclosure.

The CVU100 uses graphic commands that are designed and optimized to provide fast, accurate development of the operator's displays. These commands reduce the time and memory required to develop the control window for the control system.

The CVU100 communicates directly through any of the RS-232 or RS-422 ports on any of the Series 505 or Series 500 programmable controllers. The CVU100 also can be used with any of the Series 305 or Series 405 controllers.

Chapter 2

Operation

2.1 Controller Features

PLC Overview

The SIMATIC 545 is a compact, plug-in controller for Series 505 control systems. The controller provides the following features to fit your control requirements. Figure 2-1 shows a typical controller with I/O modules.

- 96K words of user memory (with battery backup)
- 32,768 Control Relays
- EPROM/EEPROM socket for permanent program storage
- Two serial communications ports for connecting programming equipment, operator interfaces, printers, or annunciators
- Remote I/O communications port
- 2048 I/O points (1024 discrete or analog points, and 1024 discrete-only points)
- 0.78 ms/K word Boolean execution speed
- Relay Ladder Logic programming with both normal and cyclic (timed interrupt) relay ladder tasks
- Floating-point math and high-level programming with the Special Function (Special Function) programming language
- External subroutines generated in C, Pascal, programming languages
- 64 preconfigured PID loops
- 128 preconfigured analog alarm function blocks
- LED indicators to provide easy access to machine status
- UL, FM, and CSA certification
- Mean-Time-Between-Failures of more than 24 years

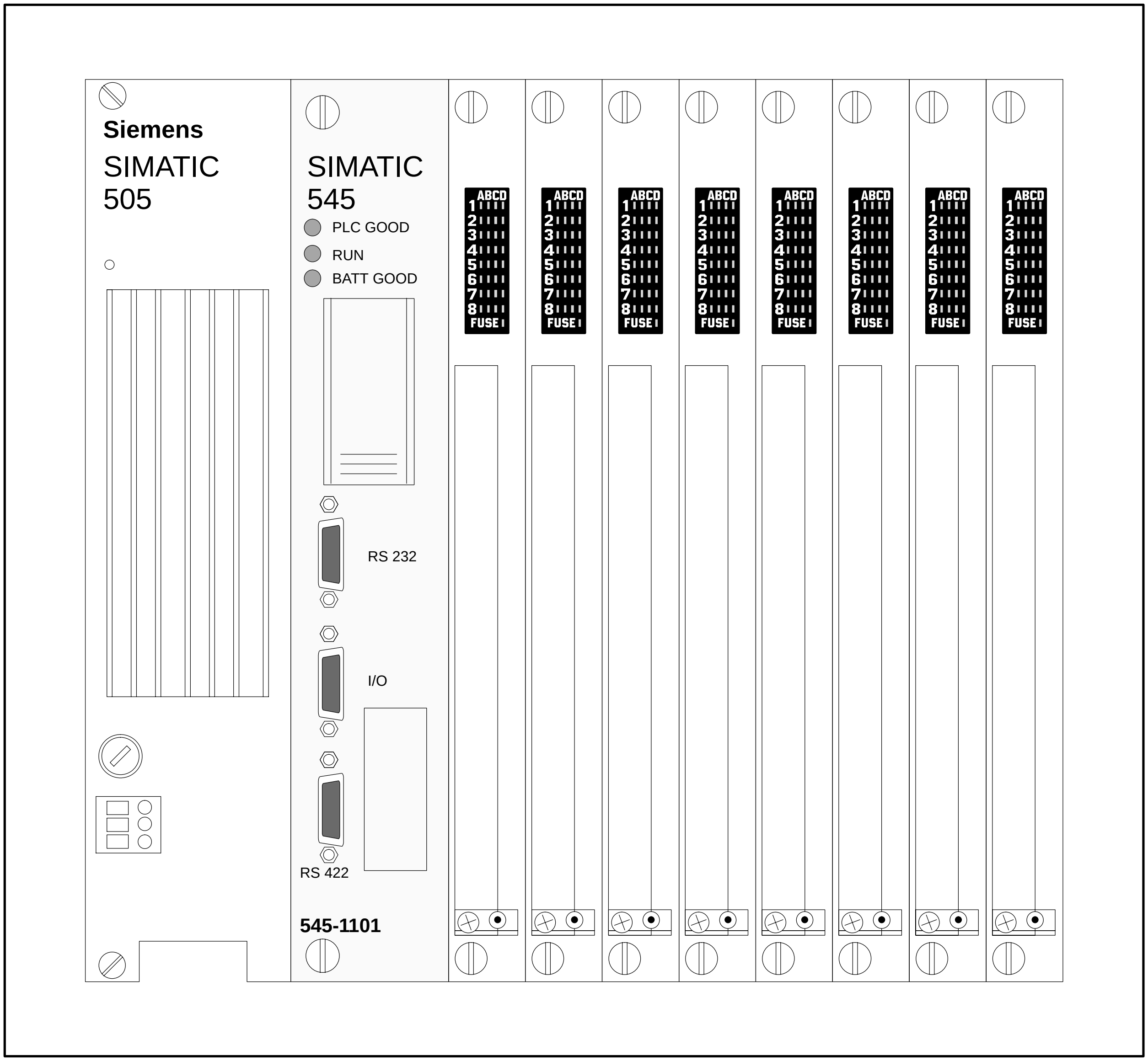


Figure 2-1   SIMATIC 545 Controller

Controller Features (continued)

Configurable Memory

With the SIMATIC 545, you can configure memory (within limits) to fit your application. For example, you can allocate memory sizes for RLL and Special Function programs, variable and constant memory, and working areas for timers, counters, etc.

Program Storage

The SIMATIC 545 offers the option of saving your RLL program in a non-volatile form by downloading it to an Electrically Erasable Programmable Read-Only Memory (EEPROM) integrated circuit. A separate programming device is not necessary. Once programmed, an EEPROM can be removed and used in any SIMATIC 545 as required.

You can also use an Erasable Programmable Read Only Memory (EPROM) programmer to copy the EEPROM to an EPROM integrated circuit. This makes program duplication much easier if you have multiple controllers that use the same program.

To ensure equipment compatibility, use only the EEPROM model supplied by your distributor.

Communication

The SIMATIC 545 has two communications ports. Both ports are configured as Data Terminal Equipment (DTE).

- RS-232/423 (9-pin male) — used to attach programming devices, modems, or operator interfaces. This port can also be used to attach a printer or message display.
- RS-422 (9-pin female) — also used to communicate with programming equipment or operator interfaces; distance from the device can be up to 1,000 feet (305 m). This port may also be configured as an RS-485 communications port.

I/O Port

The SIMATIC 545 has one RS-485 remote I/O port (9-pin female) for connecting up to 15 additional remote bases. The communication rate is 1M baud with a maximum distance of 3300 feet from the controller. This means you can put the controller in the middle of the I/O link and extend 3300 feet in both directions.

Battery Backup

A 3.0 V lithium battery protects all user memory and programming during a power cycle. Battery memory backup typically lasts six months at temperatures ranging from 0° to 60° C.

Accessory Equipment

The following accessory equipment is available as spare parts.

- Programming Cable (part #2601094–8001)
- User EEPROMs (part #2587681–8022) and EPROM (part # 2587681–8023)
- Blank Bezels (part #2587705–8003)

2.2 Controller Setup

Dipswitch Settings

Dipswitches configure communication rates, communication port selection, and battery backup. Figure 2-2 shows the dipswitch locations, and Table 2-1 and Table 2-2 show dipswitch settings. Figure 2-3 shows the pinouts for Ports 1 and 2.

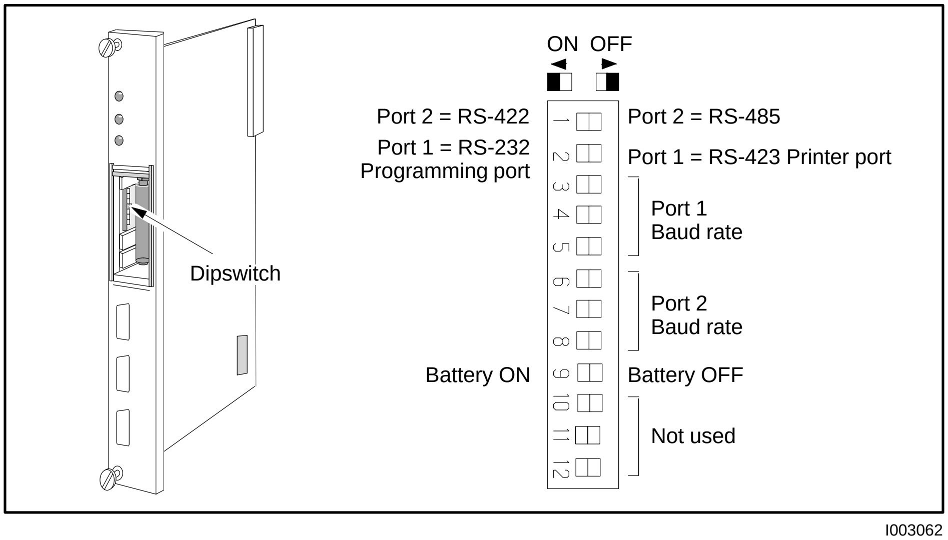


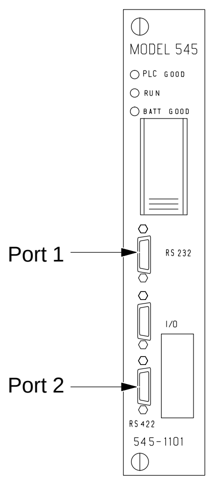
Figure 2-2 Controller Dipswitch Locations

Table 2-1 Port 1 RS–232 Baud Rate Settings

Baud Rate	Dipswitches		
	3	4	5
19200	On	On	On
9600	Off	On	On
2400	Off	Off	On
1200	Off	On	Off
300	Off	Off	Off

Table 2-2 Port 2 RS-422 Baud Rate Settings

Baud Rate	Dipswitches 6	Dipswitches 7	Dipswitches 8
19200	On	On	On
9600	Off	On	On
2400	Off	Off	On
1200	Off	On	Off
300	Off	Off	Off



Port 1
232 Port Pinouts
Controller
Male 9-Pin D Type

Pin	Signal
1	RSD
2	RCV
3	XMT
4	DTR
5	GND
6	DSR
7	RTS
8	CTS

Port 2
422 Port Pinouts
Controller
Female 9-Pin D Type

Pin	Signal
1	DO+
5	DI+
3	GND
7	DO–
8	DI–

Port 2
485 Port Pinouts
Controller
Female 9-Pin Type

Pin	Signal
1	TX/RX+
7	TX/RX–
3	GND

I002080

Figure 2-3 Controller Port Pinouts

Memory Allocation

The SIMATIC 545 uses the following designations for memory types.

- L memory (RLL)
- V memory (variable, can be changed by RLL)
- K memory (constant)
- S memory (Special Function programs, PID loops, analog alarms)
- U memory (externally developed program written in C, Pascal, assembly language, etc.).

System memory is user-configurable. L, V, K, S, and U memory allocations can be set or altered through TISOFT programming software. This allows optimum memory configuration for any application and also allows you to expand memory type capacities without affecting program integrity. Table 2-3 lists the minimum and maximum memory configuration.

Table 2-3 Controller Memory Configuration

Memory Type	Minimum Size	Maximum Size	Block Increments
Ladder (L)	512 words (1K bytes)	29.5K words (59K bytes)	512 words (1K bytes)
Variable (V)	512 words (1K bytes)	88.5K words (177K bytes)	512 words (1K bytes)
Constant (K)	0K words	88K words (176K bytes)	512 words (1K bytes)
Special (S)	0K words	88K words (176K bytes)	512 words (1K bytes)
User (U)	0K words	88K words (176K bytes)	512 words (1K bytes)
TMR/CTR /DCATS/MCAT	1024*	4096	1024*
DRUM/EDRUMS/ MDRMW/MDRMD	64	512	64*
SHIFT REGISTERS	1024*	3072	1024*
MOVE TO/FROM TABLES	1024*	3072	1024*
ONE SHOTS	1024*	7168	1024*
* Number per block There are 32K internal control relays			

Selecting the Scan Type

The SIMATIC 545 operation is divided into discrete and analog categories. The discrete portion consists of RLL execution, I/O updates, intelligent module communications, and Remote Base comm port servicing. The analog portion executes loops, analog alarms, and Special Function programs.

You control the SIMATIC 545 scan time by configuring the timeline. Using AUX Function 19 on your programming unit, you can tune the controller performance by allocating additional execution time in 1 millisecond increments. The following areas can be adjusted.

- PID loop processing
- Analog alarm processing
- Communications tasks
- Special Function program processing

When you select the scan type, you are specifying the relationship between the discrete portion of the scan and the analog portion of the scan. The three scan types are:

- Fixed Scan
- Variable Scan
- Variable Scan with Upper Limit

A more detailed discussion of the controller scan cycle is provided in Appendix A.

2.3 Operating Modes

Program Mode

In program mode, you can edit and transfer programs to controller memory. The controller does not execute the user program instructions. I/O is continuously updated during program mode.

Run Mode

In run mode, the controller executes the user program instructions and receives and sends updates to the I/O channel. You can continue to modify certain variables and constants with the programming software.

Independent Mode Control

The loops, analog alarms, and Special Function programs operate normally in run mode. You can configure the analog portion to follow the discrete portion mode changes. For example, if the discrete portion is placed in run mode, the analog portion is also automatically placed in run mode.

You can also change the modes independently. For example, you may want the discrete portion to enter program mode, but you may want the analog portion to continue processing. This flexibility makes process adjustments much easier.

The controller can also enable or disable individual loops, alarms, and Special Function programs, which allows you to adjust the individual values without affecting the process. For example, you may only want to disable one analog alarm while you recalibrate the field devices.

Run-Time Edit Mode

This mode allows the user to edit while in RUN mode. A syntax check can be performed before going back to RUN mode.

2.4 Ladder Instructions

Instruction Types

The controller executes an advanced ladder logic instruction set, which includes discretes, ladder logic subroutines, and high-level instructions.

I/O Instructions

I/O instructions are available for input, output, and internal control relay logic (Table 2-4).

Table 2-4 I/O Instructions

Instructions	Description
Inputs	The discrete input designator (X) and word input designator (WX) represent the status of input devices. Up to 2048 X points, and 1024 WX points are available.
Outputs	The discrete output designator (Y) and word output designator (WY) represent the commanded status of output devices. Up to 2048 Y points, and 1024 WY points are available.
Control Relays	The controller provides 32,768 control relays. Some of these control relays are retentive and retain their states in the event of a brief power interruption. Others are non-retentive and do not retain their states upon a brief power interruption.
Normally Open Contacts, Normally Closed Contacts	These instructions test the state of discrete inputs (X), internal relays (CR), discrete outputs (Y) and individual bits within register locations in the controller.
Immediate I/O	The Immediate Contact/Coil instructions provide the ability to read discrete inputs from or write discrete outputs to a local base I/O module immediately, without waiting for the normal I/O update cycle at the end of the scan. The IORW (Immediate I/O Read/Write) box instruction provides the ability to move blocks of discrete or word I/O from the specified local base module immediately.
Latch	The SET and RST instructions provide the ability to turn on an output and keep it on until it is reset. These instructions may also be done immediately (SETI/RSTI).
Bit-of-Word	Contact and coil instructions may also reference an individual bit of any word address, i.e. V103.12.
Relational Contacts	Relational Contacts provide the ability to compare a word address value, such as a V- or WX-memory location, with a constant value or another word address value. Power flow is established by the truth of the specified relationship: = <> < <= > >= .
Invert Power Flow	The NOT (Invert Power Flow) instruction provides the ability to logically invert power flow based on the preceding logic, i.e. if power flow is established prior to :NOT: , then it is turned OFF following the instruction, and vice versa.

Conditional and Unconditional Operations

Conditional instructions provide branching capabilities and other alternative program strategies within RLL programs. Table 2-5 lists Conditional/Unconditional Instructions.

Table 2-5 Conditional/Unconditional Operations

Instruction	Description
Jump	Freezes all discrete outputs in its field of control. The end jump instruction may be either conditional or unconditional.
Master Control Relay	Turns off all discrete outputs within the zone of control between the MCR instruction and the end MCR. The end MCR instruction may be either conditional or unconditional.
Skip Forward to LBL	The Skip Forward to Label and Label instructions provide a means of selecting specific segments of a program to be executed during a scan. When the controller reads the SKIP instruction, no ladder logic is executed until the Label instruction is found.

Cyclic RLL Tasks

The TASK box instruction marks the beginning of a Cyclic RLL TASK. It allows you to segment RLL into separate segments. TASK segments the RLL into two sections, TASK1 is the main ladder program, TASK2 is the cyclic task. One use of a Cyclic RLL TASK may be to read/write Immediate I/O on a deterministic schedule shorter than the overall program scan. Table 2-6 lists the Cyclic RLL Task.

Table 2-6 Cyclic RLL Task

Instruction	Description
Task	The Cyclic RLL Task executes on the timed interval specified in the instruction. The interval can be a constant or any word address. (Task 1 is the Main RLL program with a non-changeable timed interval of 0.)

Bit Instructions

These instructions allow you to read the state of a specific bit or bits within a data word, and to set or reset individual bits. Table 2-7 lists the Bit Instructions,

Table 2-7 Bit Instructions

Instruction	Description
Bit Clear	Clears a selected bit of word to 0
Bit Pick	Tests a selected bit of a specified word to determine its status
Bit Set	Sets a selected bit of a word to 1
Bit Shift Register	Creates a bit shift register containing up to 1023 bits

Counter/Timer Instructions

Timers (TMR) operate on a base of either 1/10th of a second or 1 millisecond. Counters (CTR) may count up to 32,767. In addition, the Up/Down Counter (UDC) saves programming time by allowing one UDC to count up *and* count down, depending on which of two input signals are received. Table 2-8 lists the Counter/Timer Instructions.

Table 2-8 Counter/Timer Instructions

Instruction	Description
Counter	Counts recurring events up to 32,767
Up/Down Counter	Counts the number of events (up or down) that have occurred between zero and 32,767
Timer	Decrements in .1-second or 1-millisecond steps for timing of events
Discrete Control Alarm Timer	Used with discrete control valve that provides feedback on whether the valve is open or closed. The output of the instruction should be the output to the valve or motor; the input should be from the logic that determines the state of the valve.
Motor Control Alarm Timer	Similar to the Discrete Control Alarm, but provides additional functions to use with motor driven valves, driving in each direction or reversing motors.

Drum Instructions

Drum timers and event-driven drums are commonly used sequencing devices. Unlike most controllers which force you to program sequencing with extensive ladder logic, the Series 505 instruction set provides simple, easy-to-use box instructions for these functions. Table 2-9 lists the Drum Instructions.

Table 2-9 Drum Instructions

Instruction	Description
Drum	Simulates operation of an electromechanical drum, and provides 15 output coils with 16 steps. The steps are index on multiples of the time base set up for drum.
Event Drum	Simulates operation of an electromechanical drum and provides 15 output coils with 16 steps. The steps can be indexed by a timer only, an event contact only, or a combination of an event contact and a timer.
Maskable Event Drum-Discrete	Functions in the same manner as the event drum with the additional ability to specify a mask for each step to allow selection of the outputs to be controlled by the step.
Maskable Event Drum-Word	Functions just as the discrete drum, but gives the option of specifying a word rather than a set of discretes.

Matrix Instructions

For fault diagnostics or sequence error checking it is often necessary to compare the actual condition of external or internal I/O points with the expected state. Table 2-10 lists the Matrix Instructions.

Table 2-10 Matrix Instructions

Instruction	Description
Scan Matrix Compare	Compares up to 16 user-defined bit patterns against the current states of up to 15 I/O points. If a match is found, the step number is entered into the V-memory location specified.
Index Matrix Compare	Compares the bit pattern of a user-specified step to the current state of up to 15 I/O points. The instruction is indexed by loading a step number into the location specified.

Math Instructions

The ladder logic math functions operate on integers between –32,768 and +32,767. Square root, absolute value, and a comparison block are provided in addition to the standard four functions of add, subtract, multiply, and divide. In most cases, the operands can be either memory locations or a constant (signed or unsigned integer value). Table 2-11 lists Math Instructions.

Table 2-11 Math Instructions

Instruction	Description
Add	Adds a signed integer (positive or negative) in memory location A to either a signed integer in memory location B or a constant and stores the result in memory location C. The signed integers on which the addition is performed are not affected by the operation and retain their values in the original locations.
Subtract	Subtracts either a signed integer (positive or negative) in memory location B or a constant from a signed integer in memory location A and stores the result in location C. The signed integers on which the subtraction is performed are not affected by the operation and retain their values in the original location.
Divide	Divides a dividend in two memory locations, A and A+1, by a divisor in memory location B. The quotient and remainder are stored in two memory locations, C and C+1. The dividend and divisor values are not affected and retain their values in memory after the division is complete.
Multiply	Multiplies a signed integer in memory location A by a signed integer in memory location B. The product is stored in two memory locations, C and C+1. The signed integer values in memory locations A and B are not affected by the multiplication and retain their values in the memory after multiplication is complete.
Compare	Compares a number in memory location B to a number in memory location A. The comparison is made for equal to, less than, or greater than. Values in the A and B memory locations are not affected.
Square Root	Takes the square root of a positive integer and stores the result in a specified memory location. The integer retains its original value.
Absolute Value	Replaces the value of the specified word with its absolute value.

Move Instructions

Data may be easily moved between the I/O buffer and working memory. Constants may be loaded into working memory using the Load Data Constant (LDC) instruction. Block moves of up to 256 words are supported. Tables may be established, complete with indirect addressing. Tables may also be used as First In, First Out (FIFO) registers. Data may be moved between any storage areas, including timer/counter/ drum values, variable storage, and the I/O buffer. Table 2-12 lists the Move Instructions.

Table 2-12 Move Instructions

Instruction	Description
Load Data Constant	Loads the specified value into the A memory location.
Move Discrete IR to Word	(Move Image Register to Word) instruction shifts a specified number of bits from the discrete image register (X, Y, or C) to a specified word in memory.
Move Word to Discrete IR	(Move Word to Image Register) allows a specified number of bits to be shifted from a word memory location to specified discrete image register locations in a single memory scan.
Move Word (memory)	Moves up to 256 words from a designated memory location to another designated location.
Move Word from Table	Moves words from one table location in V-memory to another location in V-memory, as specified by a table address pointer. One word is moved each memory scan.
Move Word to Table	Moves words from a word source in V-memory to a table destination address specified by a pointer in V-memory. One word is moved during each scan.
Indirect Move Word	A table-to-table block move. In a single scan, it moves a block of up to 256 words from within a source table to within a destination table, then increments both table pointers by the number of words moved. In batch applications, this instruction greatly simplifies the management of large tables of recipe parameters.
Move Image Register to/from Table	Moves blocks of image register bits (up to 256 words or 4090 bits per scan) into or out of tables in V-memory and increment the table pointers by the number of words moved. For example, these instructions allow efficient bit-level interlocks across Peerlink networks, or elegant machine diagnostics with the table search instructions described below.
Move Element	The Move Element is a general move command. Most source and destination fields are supported, as well as indirect addressing. This command in most instances will be the only move instruction needed.

Word Instructions

Information from devices such as weight scales, bar code readers, or Binary-Coded Decimal (BCD) devices can best be interpreted by word-manipulation instructions. These instructions can also be particularly useful in handling alarm priorities. Table 2-13 lists the Word Instructions.

Table 2-13 Word Instructions

Instruction	Description
Convert Binary to BCD	Converts binary inputs to equivalent Binary Coded Decimal values. Binary integer values up to 32,767 are converted to equivalent BCD values.
Convert BCD to Binary	Converts BCD inputs to equivalent binary integer values. BCD inputs up to 9999 are converted to their binary integer equivalents.
Word AND	(WAND) logically ANDs a word in memory location A with a word in memory location B. The WAND instruction then places the result in memory location C.
Word OR	(WOR) logically ORs a word in memory location A with a word in memory location B. The result of the WOR instruction is placed in memory location C.
Word Exclusive OR	(WXOR) operates on a word in memory location A and a word in a second location B. The result of the WXOR instruction is placed in memory location C.
Move Image Register to/from Table	Moves blocks of image register bits (up to 256 words or Register to/from 4090 bits per scan) into or out of tables in V-memory Table and increment the table pointers by the number of words moved. These instructions allow for efficient bit-level interlocks across Peerlink networks, for example, or elegant machine diagnostics with the table search instructions described below.
Word Rotate	Rotates to the right 4-bit segments of the word location specified.
Word Shift Register	Shifts from 1 to 1023 words from memory location A to Register V-memory, beginning at location B.
One shot	Provides an output for one memory scan.

Ladder Logic Subroutines

Subroutines provide additional control of conditional operations with definition, go to, and return statements. Table 2-14 lists the Subroutines available.

Table 2-14 Subroutines

Instruction	Description
Go to Subroutine	Enables you to write ladder logic programs preceded by a subroutine number and to call them as needed. Can have up to 255 subroutines.
Parameterized Go to Subroutine	Similar to Go to Subroutine instruction, except that parameters may be passed to the subroutine.
External Subroutine Call	Allows you to pass parameters to a subroutine that is developed offline in a non-RLL programming language, such as C or Pascal, and then calls the subroutine for execution.
Subroutine	Is placed before a set of ladder logic instructions that are to be executed only when called with the Go to Subroutine or Parameterized Go to Subroutine instruction.
Return	Brings execution of the ladder logic program back to the network following the subroutine call. The Return may be either conditional or unconditional.

Clock Instructions

The controller has a real-time clock which can be checked and set via ladder logic using the clock instructions. For example, end-of-shift reports or time driven events can easily be initiated. Table 2-15 lists the Clock Instructions.

Table 2-15 Clock Instructions

Instruction	Description
Date Compare	Compares the current date of the real-time clock with the value contained in the designated V locations.
Time Compare	Compares the time of the real-time clock with the values contained in the designated V-memory locations.
Time Set	Sets the time portion of the real-time clock to the values contained in designated V-memory locations.
Date Set	Sets the date portion of the real-time clock to the values contained in designated V-memory locations.

Table Instructions

Tables may be established, complete with indirect addressing, and may also be used as FIFO registers. Data may be moved between any storage areas, including timer/counter/ drum values, variable storage, and the I/O buffer. Table 2-16 lists the Table Instructions.

Table 2-16 Table Instructions

Instruction	Description
Table to Table AND	ANDs the corresponding bits in two tables and places the results in a specified third table.
Table to TABLE OR	ORs the corresponding bits in two tables and places the results in a specified third table.
Table to Table EXCLUSIVE OR	Exclusively ORs the corresponding bits in two tables and places the results in a designated third table.
Table Complement	Inverts the status of each bit in the first table and places the results in the second table. The complement of 0 is 1; of 1, 0.
Word to Table	Places a duplicate of a word at the address specified in the destination table.
Table to Word	Duplicates a specified word in a table to another word location.compares each bit in a source word to the corresponding bit of a designated word in a table. The results are placed in a destination table.
Word to Table OR	Logically ORs the corresponding bits of a source word with a designated word in a source table. The results are placed in a destination table.
Word to Table OR	Exclusively ORs the corresponding bits in a specified word and a word from a source table. The resulting word is placed in a destination table.
Table Search for Equal	Locates the next occurrence of the word in a table that is equal to the source word. The address of the match is shown by a pointer.
Table Search for Not Equal	Locates the next occurrence of the word in a table that is not equal to the source word. The mismatch is shown by a pointer, and the value is copied to another specified word.

2.5 PID Loops

Number of Loops Supported

The controller supports 64 PID loops. The PID equation has two operating modes: Position and Velocity. You can select the mode that best fits your application.

- *Position*—The output of the standard PID control rule is an actuator position which may be sent directly to a WY-output location and out to a standard valve or final control element. This form of the equation (known as the position equation) is used in most cases.
- *Velocity*—Derived by taking the first derivative of the position equation, it then computes a *change* in valve position from one sample time to the next. The output of the velocity form may be sent through blocks of time proportioning logic to produce timed pulses to move a motor-driven valve to a new position.

Loop Flag

A loop flag address can be configured to contain status information about the loop. The individual bits in the loop flag can be used in ladder logic to determine if the loop is in manual, auto, or cascade mode, or if the loop is in an alarm condition (see Loop Modes). The loop operating mode (manual, auto, cascade) can also be controlled from ladder to determine logic by setting the appropriate loop flag bits.

Ramp/Soak Operation

A Ramp/Soak option is also available to allow automatic setpoint (SP) changes over a period of time. Figure 2-4 shows a ramp/soak scenario.

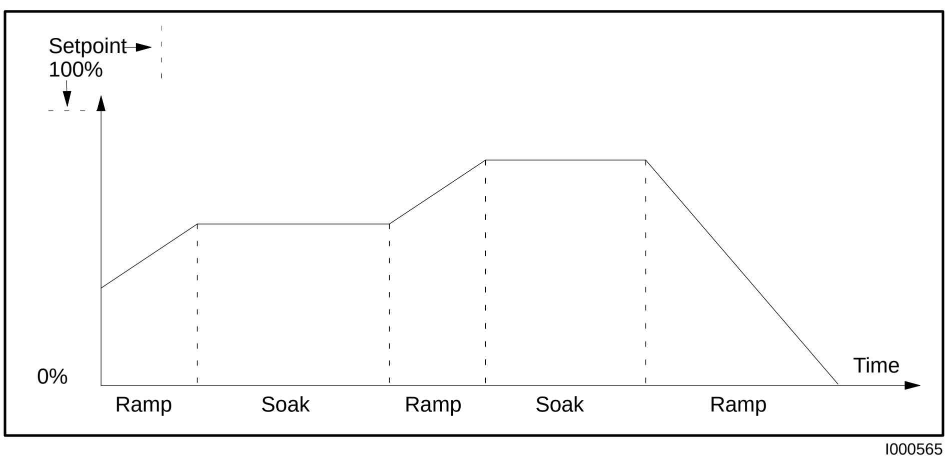


Figure 2-4 Ramp/Soak Example

Setpoints and Ramp Rates

You can define the setpoints and the ramp rates. A total of 256 Ramp/Soak steps are allowed per PID loop. Jog, Hold, and Restart options are also provided. There are status bits available in the Ramp/Soak menu to determine where you are at a given time in the Ramp/Soak routine.

Derivative Gain Limiting

Noise on the process variable usually limits or even prohibits the use of derivative control in simple controllers. Derivative gain-limiting provides a filter applied to the process variable, only in the derivative term, to limit this disturbance. This also allows full three-mode PID control even in noisy environments.

Special Function Calls

Advanced process control applications often require additional processing and/or complex mathematical operations, in conjunction with standard mode PID control, for implementing adaptive control strategies, process optimization, and pre- or post-processing of loop variables. The controller provides the option of calling individual programs (written in a BASIC-like high level language called Special Function programming) from each of the PID loops configured in the system.

Special Function programs provide extensive integer, and floating point math capabilities, as well as complete access to all loop parameters, including process variable, setpoint, loop output, bias, gain, rate and reset values, as well as all other controller data.

Special Function programs may be called from any loop to operate on the setpoint, process variable (PV), or loop output. The controller calls Special Function programs that act on loop output following loop execution but prior to writing the new output value.

Special Function programs that operate on the setpoint or process variable are called when the setpoint or process variable is accessed. However, the Special Function program does not necessarily have to perform an operation on either item. The setpoint is accessed each time the loop PID calculation is done as required by the sample rate.

The process variable is accessed at the sample rate, or at least every 2 seconds if the sample rate is greater than 2 seconds. In this case, Special Function programs can be used to scale incoming values or to change process variable or setpoint values depending on the status of certain external conditions (e.g., thumbwheel switches).

Setpoint Limits

A clamp setpoint limits option is available to limit the setpoint range that may be changed by an operator interface or Special Function program.

Loop Modes

Loops can be operated in either manual, automatic, or cascade modes. Also, different combinations of PID control (e.g., P, P+I, P+I+D) are available. Appendix B provides a more detailed description of loop modes of operation.

Reset Windup

Reset windup protection is included in the PID loop control block, with an option to actively back-calculate or simply freeze the integral (“bias”) term when the output reaches zero or 100%.

Process Variable Alarms

Process Variable alarms are available with four absolute limits, two deviation limits, a rate of change alarm, and a broken transmitter alarm. In addition, the alarm deadband is selectable from 0.2% of span to 5.0% of span.

PID Loop Entry

Programming a PID loop in a SIMATIC 545 begins with a Loop Directory that identifies the loops by number and title. After selecting a loop to program, you just fill in a two-page menu with the required information. Figure 2-5 shows the first of the two menu pages. Once these menus are completed, the loops are ready to run; no ladder logic or additional programming is required to invoke the loop algorithm.

PID LOOP 1 TITLE:		REMOTE SETPOINT:	NONE
		CLAMP SP LIMITS: LOW:=	+0.00000
POS/VEL PID ALGORITHM:	POS	HIGH=	+0.00000
LOOP VFLAG ADDRESS:	NONE		
SAMPLE RATE (SECS):	+1.00000	LOOP GAIN:	+1.00000
		RESET (INTEGRAL TIME):	+INF
PROCESS VARIABLE ADDRESS:	NONE	RATE (DERIVATIVE TIME):	+0.00000
PV RANGE: LOW=	+0.00000	FREEZE BIAS:	NO
HIGH=	+100.000		
		DERIVATIVE GAIN LIMITING:	NO
PV IS BIPOLAR:	NO	LIMITING COEFFICIENT:	+10.0000
SQUARE ROOT OF PV:	NO		
20% OFFSET ON PV:	YES	SPECIAL CALCULATION ON:	NONE
		SPECIAL FUNCTION:	NONE
LOOP OUTPUT ADDRESS:	NONE		
OUTPUT IS BIPOLAR:	NO	LOCK SETPOINT:	NO
20% OFFSET ON OUTPUT:	YES	LOCK AUTO/MANUAL:	NO
		LOCK CASCADE:	NO
RAMP/SOAK PROGRAMMED:	NO		
RAMP/SOAK FOR SP:	NO	ERROR OPERATION:	NONE
		REVERSE ACTING:	NO
S-MEMORY AVAILABLE: 4096			ENABLED
		545 NETDATA	PG
EXIT–F1	EDIT–F2	COMMNT–F7	EN/DIS–F8

I000566

Figure 2-5 Loop Entry Screen

2.6 Analog Alarms

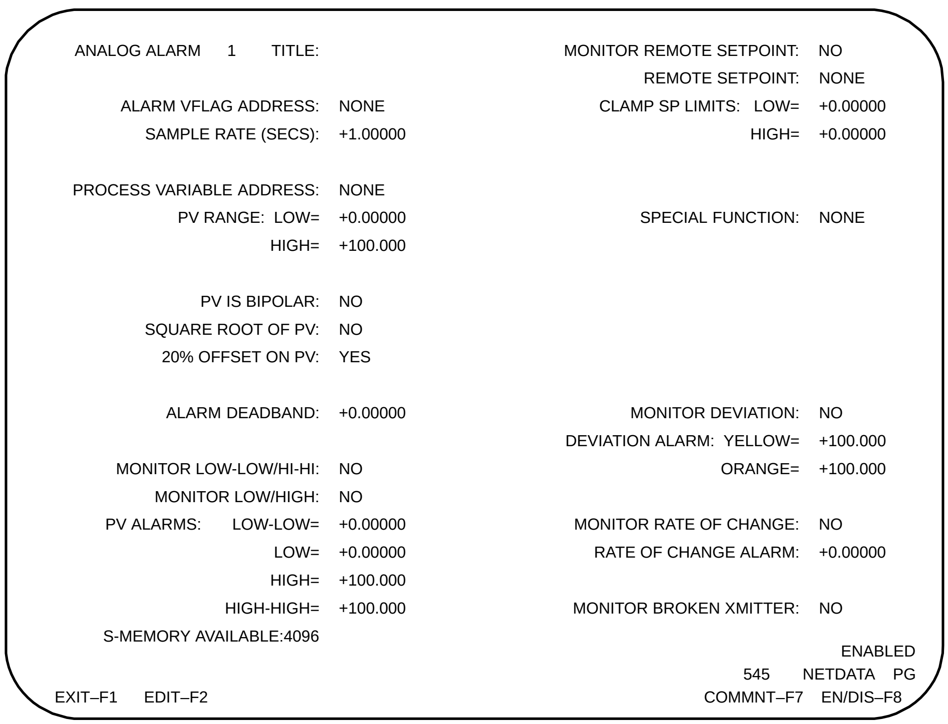
Analog Alarms

The controller provides 128 Analog Alarm function blocks that input and register values. These analog alarms provide four limit alarms, two deviation alarm bands, as well as rate-of-change and broken transmitter alarms.

The alarm limits and configuration is in Engineering Units. An option is also available to initiate a special function calculation. In this manner, the timing and scaling capabilities of the analog alarm algorithm can be used in conjunction with Special Function programming to create custom, PID-type control.

Analog Alarm Entry

Programming an analog alarm in a SIMATIC 545 begins with an analog alarm directory listing the alarms by number and title. After selecting the analog alarm for programming, complete a sample menu (see Figure 2-6).



I000567

Figure 2-6 Analog Alarm Menu

2.7 Special Function Programs

Overview

In addition to the standard relay ladder logic, the programming instruction set offers more than 30 high-level, user-oriented operations. The Special Function instructions replace programming steps that once required hundreds of words of memory. More importantly, these instructions permit you to use the power of the controller to perform complex functions, which makes program development and debugging faster and easier.

User-defined Special Function programs consist of Special Function statements. In addition, Special Function programs may call Special Function Subroutines. Each subroutine is assigned a unique number and can be called from one or more Special Function programs. Special Function subroutines can also call other subroutines up to a total of four levels deep.

The controller also provides a high-level language programming environment, called Special Function Programming, ideally suited for implementing supervisory tasks and advanced process control strategies. The controller allows up to 1023 individual Special Function programs (and an additional 1023 subroutines) to be defined in the controller. Special Function programming is a BASIC-like self-documenting language offering extensive integer and floating point math operations, flow control instructions, and complex operations including if-then-else constructs, lead/lag function blocks, and sequential or correlated data table operations.

Versatile Calls

Special Function programs can be called on a timed basis (cyclic), from ladder logic, from PID loops, or from analog alarms. Additionally, Special Function programs can be identified as Priority or Non-priority.

2.8 Special Function Statements

Special Function programs consist of one or more statements. The statements can be combined to obtain the control effects needed for your situation.

COMMENT

Provides comment capabilities similar to the “REM” function in BASIC. Comments are stored in the programmable controller to provide online documentation of the program. Comments are often required in some control areas to provide information about the actions taking place.

BINARY-BCD

Performs a Binary to BCD conversions.

SCALE

Converts Binary to Engineering units.

UNSCALE

Converts Engineering units to Binary.

MATH

The math statements allow you to enter math equations in a natural algebraic manner using the rules of operator precedence. Parenthesis are allowed, as well as subscripted variables for reading or writing to a number of values. Variables and constants may be either integer or real (32-bit floating point). Math statements support the following operators, which are performed in full IEEE 32-bit floating point precision format.

Example: math V100. : = V12.* (Sin V12. + LOG V200.) / V65.

**	exponentiation
*	multiplication
/	division
+	addition
–	subtraction/unary minus
:=	assignment
NOT	Unary NOT. The expression “not X” returns the one’s complement of X
>>	Shift right
<<	Shift Left
*	Multiplication
MOD	Modulo arithmetic. The expression “X mod Y” returns the remainder of X after division by Y.
WAND	Bit-by-bit AND of two words
WOR	Bit-by-bit OR of two words
WXOR	Bit-by-bit XOR of two words

Relational operators (=, >, <, >=, <=, <>, AND, OR)

Special Function Statements (continued)

In addition to the previous operators, the Math Special Function provides the following operations:

- Absolute value (ABS)
- Truncate, return integer (TRUNC)
- Return fraction (FRAC)
- Round, return integer (ROUND)
- Return smallest integer greater than or equal to x (CEIL)
- Return largest integer less than or equal to x (FLOOR)
- Square root (SQRT)
- Exponential (EXP)
- Natural logarithm (LN(Base e))
- Common logarithm (LOG(Base 10))
- Sine – in radians (SIN)
- Cosine – in radians (COS)
- Tangent – in radians (TAN)
- Inverse sine – in radians (ARCSIN)
- Inverse cosine – in radians (ARCCOS)
- Inverse tangent – in radians (ARCTAN)

IMATH

With integer only math, parenthesis, constants and subscripted variables are allowed. Logical functions are also added and include:

NOT	Unary NOT. The expression “not X” returns the one’s complement of X
>>	Shift right
<<	Shift Left
*	Multiplication
/	Integer division. Any remainder left over after the division is truncated
MOD	Modulo arithmetic. The expression “X mod Y” returns the remainder of X after division by Y.
+	Addition
–	Subtraction/unary minus
WAND	Bit-by-bit AND of two words
WOR	Bit-by-bit OR of two words
WXOR	Bit-by-bit XOR of two words
:=	Assignment

IF-THEN-ELSE, ENDIF

The IF-THEN-ELSE, ENDIF special functions are used together to program conditional statements. The ENDIF statement ends the IF-THEN loop. The IF statement may include valid mathematical operations as a part of the expression, for instance:

IF V200 ≤ SIN (V12) / V25

SDT Sequential Data Table

The Sequential Data table maintains an index into a table. Each time the statement is executed, the table index is incremented by one and the next entry in the table is used as output. When the end of the table is reached, the index is reset to the first entry in the table and a discrete bit, called the restart bit, is turned off. The restart bit is turned on if the index is at any point other than the beginning of the table.

CDT Correlated Data Table

The Correlated Data Table uses an input and an output table. The CDT routine locates the entry in the input table that is greater than or equal to a specified input value. It then writes the corresponding entry in the output table to the output variable. Both tables must have the same number of entries and the values in the input table must be in ascending order; i.e., the lowest value in the lowest numbered memory location to the highest value in the highest memory location.

SSR Sequential Shift Register

Sequential shift registers shift all of the data within the register one position each time a shift command is executed. The first (vacated) position in the table is set to zero and the data shifted out of the last position is lost. The register is considered empty when it contains all zeros, and the status bit is turned on to indicate that the register is empty. The sequential shift register is especially useful for following parts down an assembly line.

FSTR IN Fall Thru Shift Register Input and Output

FTSR-IN and FTSR-OUT are used to operate asynchronous shift registers. FTSR-IN is used to add an entry to the table while FTSR-OUT is used to remove an entry from the table. The asynchronous shift registers are tables containing binary words. The first words entered into the table with FTSR-IN are the first words output with FTSR-OUT.

FTSR OUT Fall Thru Shift Register Output

FSTR-OUT is used to remove an entry from an asynchronous shift register. This is done on a first in/first out basis as noted above.

PRINT ASCII Message Print

Print is used to send a message through the ASCII message port. This statement may be used to print both text and the contents of integer and real variables.

CALL

Call Special Function command is used to call an Special Function subroutine. Up to 1023 subroutines can be created in the controller. Each subroutine is given a unique number and once the Special Function SUB is defined, it can be called using its unique number. Four levels of subroutine nesting are allowed. The call function can include up to 5 parameters which are passed and/or returned from the subroutine.

PACK

Pack Special Function provides a means to easily pack data into contiguous blocks. This Special Function can pack Xs, Ys, Cs, WXs, WYs, or K or V words into blocks. This Special Function is most useful for efficient transmission of data on a communications network such as TIWAY. Pack can also be used to unpack data stored in contiguous blocks.

LEAD/LAG

The Lead/Lag Special Function allows you to perform shaping and filtering on any analog variable. Typically used in process control applications as a compensator in dynamic Feed-Forward control, this Special Function may only be invoked from loops, analog alarms, or cyclic Special Function programs. In specifying the Lead/Lag Special Function the Input and Output words are defined along with the lead-time in minutes and the lag-time in minutes plus a gain.

RETURN

Return Special Function returns from a subroutine. If invoked from an Special Function program, the Special Function program is terminated.

PACK LOOP

The Pack Loop command packs loop data from a loop into contiguous blocks or unpacks loop data from contiguous blocks and stores it in the loop.

PACK ALARM

The Pack Analog Alarm Special Function packs Analog Alarm data from an analog alarm into contiguous blocks, or as in Pack Loop, except substitute analog or alarm for loop.

GOTO/LABEL

The GOTO statement is used to continue program execution of a specified LABEL statement.

2.9 Special Function Program Directory

Special Function Program

Programming a Special Function program in the SIMATIC 545 begins with displaying the directory that lists the programs by title and number, as shown in Figure 2-7.

SPECIAL FUNCTION PROGRAM/SUBROUTINE DIRECTORY S–MEMORY AVAILABLE:9908

ENABLE/ DISABLE	PROGRAM		ENABLE/ DISABLE	PROGRAM	ENABLE/ DISABLE	SUBROUTINE		ENABLE/ DISABLE	SUBROUTINE
	1	MTR CTRL				1	PUMP 101		
	2					2	FLOWRATE		
	3					3	BOILER 1		
D	5	BOILER 2				10			
	6	PAINT				14	LMT SW 7		
	7	PRINT				64			
	17	IF TEST				87			
	23				D	187	BURNER 2		
	27					237			
	77					287			
	100	VALUE 34							
	200								
	250								
	300								

SELECT: SFSUB0010

545 NETDATA PG

EXIT–F1 SHOW–F2 BLOCK–F3 EN/DIS–F4 PG/RN-F5 LOOP–F6 ALARM–F7 SFPGM–F8

I000568

Figure 2-7 Special Function Program Directory

Entering a Special Function Program

Special Function programs are entered with menus, as shown in Figure 2-8.

```
TITLE:     BOILER 2                                                   SF PROGRAM        5

                              CONTINUE ON ERROR (Y,N):   YES
                         ERROR STATUS ADDR (Y,C,WY,V):   V500 – V502
                                PROGRAM TYPE (N,P,C,R):   CYCLIC
                                      CYCLE TIME (SEC):   0.5
_______________________________________________________________________________________

00001   *        CALCULATE FLOW AND MASS FLOWRATE TO BOILER 2.  GET
                 SIGNAL, TAKE SQUARE ROOT, ADD CORRECTION AND SAVE.
00002   MATH     V200  := SQRT ( WX100 ) + V250
00003   *        THE CORRECT FLOWRATE IS NOW SAVED IN CUBIC FEET/MIN.
                 CALL SUBROUTINE 2 TO CALCULATE MASS FLOWRATE FROM
                 TEMP AND FLOWRATE.
00004   CALL      SFSUB  ............ :  2                P1 ................ :  V200
                  P2 ................. :  V110           P3 ................ :  V202
                  P4 ................. :                 P5 ................ :
00005   *        THE MASS FLOWRATE IS STORED IN LOCATION V202, IN (LBS/MIN).
                 IF THERE IS FLOWRATE MISMATCH WITH BOILER 1, SET FLAG.
00006   IF       V202 – V102 > 150
00007   IMATH    V100  := 1
00008   ELSE
00009   *        ELSE RESET FLAG
S–MEMORY AVAILABLE:3052

                                                                                ENABLED
                                                                  545    NETDATA    ED
  EXIT–F1    EDIT–F2          FIND–F4     DELST–F5    INSST–F6     COMMNT–F7    EN/DIS–F8
```

I000569

Figure 2-8   Special Function Program Example

The error status can be designated as a single discrete bit or a group of three consecutive word locations. Specifying a bit for error status tells only that an error has occurred. With three words designated for error status, information relevant to the error can be identified and corrected.

- Word 1 — contains an error code determining what type error was encountered.
- Word 2 — contains the number of the Special Function program or Special Function subroutine where the error was encountered.
- Word 3 — contain the last fully executable line before encountering the error.

2.10 Auxiliary Functions

Overview

There are several auxilliary function available for various controller diagnostics and self checks. Some of the diagnostics and self-checks are accessible through the Auxiliary Function menu on the TISOFT programming software. The following Auxiliary Functions are available.

- Power-up restart — AUX 10
- Partial restart — AUX 11
- Complete restart — AUX 12
- Compare PLC to Disk — AUX 17
- PLC Scan Time — AUX 19
- Run PLC diagnostics — AUX 20
- Display failed I/O — AUX 25
- Show PLC diagnostic cell — AUX 29

Power-up Restart

Function 10 (power-up restart) clears all unforced X, Y, and non-retentive C elements on power-up or restart of the PLC. Retentive control relays are not cleared. The WX and WY elements are not affected. See Table 2-17 for AUX Function 10 restart states.

Partial Restart

Function 11 (partial restart) clears all discrete elements except retentive C and forced elements. The word elements and presets are not reset. See Table 2-17 for AUX Function 11 restart states.

Complete Restart

Function 12 (complete restart) clears all discrete elements and word elements, including retentive C elements. Forced word elements are not reset. See Table 2-17 for AUX Function 12 restart states.

Table 2-17 Effects of Using AUX Functions 10, 11, and 12

System Characteristics	Power-up Restart AUX 10 Battery ON	OFF	Partial Restart AUX 11 Battery ON	OFF	Complete Restart AUX 12 Battery ON	OFF
PLC Mode (RUN, PGM HOLD, RAM, ROM	No change*	PGM*	No change*	No change*	No change*	No change*
Loop Mode	No change*	PGM/LOCK*	No change*	No change*	No change*	No change*
Discrete I/O (Xs and Ys)	Cleared	Cleared	Cleared	Cleared	Cleared	Cleared
Word I/O (WXs and WYs)	No change	Cleared	No change	No change	Cleared	Cleared
Non-Retentive Control Relays	Cleared	Cleared	Cleared	Cleared	Cleared	Cleared
Retentive Control Relays	No change	Cleared	No change	No change	Cleared	Cleared
Forced Elements	No change	Cleared	No change	No change	No change	No change
TMR/CTR/DRUM Presets	No change	Cleared	No change	No change	Downloaded from L–memory value	Downloaded from L-memory value
L,V,K,S Memory	No change	Cleared	No change	No change	No change	No change
Memory Configuration	No change	Default Value(s)	No change	No change	No change	No change
I/O Configuration	No change	Cleared	No change	No change	No change	No change
Scan Time	No change	Default Value(s)	No change	No change	No change	No change
Time of Day	No change	Cleared	No change	No change	No change	No change
Task Codes/Scan	No change	Default Value(s)	No change	No change	No change	No change
Watchdog Value	No change	Default Value(s)	No change	No change	No change	No change
Port Lockout	No change	UNLOCKED	No change	No change	No change	No change
Fatal Error	No change	Cleared	Cleared	Cleared	Cleared	Cleared

* If fatal error was present then Program Mode
** If programmed EEPROM (EPROM) present, then RUN Mode using program in EEPROM (EPROM)

Compare PLC to Disk

AUX Function 17 (Compare PLC to Disk) compares program disk and PLC data. It allows you to save and load your VERIFY options to and from the disk or directory.

PLC Scan Time

Function 19 (PLC Scan Time) allows you to configure the SIMATIC 545 scan cycle for optimum performance. Appendix A provides additional information on the SIMATIC 545 scan cycle.

Run PLC Diagnostics

AUX Function 20 (run PLC diagnostics) initiates the PLC self-checks. The PLC must be in PROGRAM mode in order to execute self-tests. The PLC executes the equivalent of an AUX Function 20 at every power-up. If any area fails, a message detailing the failure is displayed. AUX Function 20 makes the following tests.

- Valid RAM locations are verified
- Pre-coded ROM checksum values are checked
- The hardware force function is checked
- Internal timer operation is verified
- Operating system ROM cyclic redundancy check code is checked

Display Failed I/O

AUX Function 25 (display failed I/O) displays the locations of any failed I/O modules which are capable of diagnosing and indicating failure. Some modules, such as word and analog modules, report that they have failed if the user-supplied voltage is not correct.

AUX Function 25 also displays I/O mismatches, indicating that one or more installed module(s) do not agree with the I/O configuration. Verify that the configuration data for listed modules is correct before considering that the module may actually be malfunctioning.

Show PLC Diagnostic Cell

AUX Function 29 checks the PLC status and displays the results. The screen displays the information shown in Figure 2-9.

```
                         545 OPERATIONAL STATUS

          SCAN  =   52 MSECS/VARIABLE          BATTERY      =   GOOD
          KEY   =   UNLOCKED                   DOWNLOAD     =   OFF
          MODE  =   PROGRAM                    PROGRAM IN RAM
          S-MEMORY CONFIGURED = 4096           S-MEMORY AVAILABLE = 3052

          SF/LOOP MODE = PROGRAM
          SF/LOOP FOLLOWS PLC TO PROGRAM MODE

          FATAL ERROR            =   NONE
     LOOP FATAL ERROR            =   NONE
          NON FATAL ERROR        =   NONE
     LOOP NON FATAL ERROR        =   NONE



                                                        545  NETDATA    PG
  EXIT–F1
```

I000570

Figure 2-9 Typical PLC Status Screen (AUX 29)

- Scan = scan time reported by AUX Function 29; it is the highest scan time recorded since the last PLC reset or PROGRAM–RUN mode transition. Status Word 10 (STW10) contains a continuously updated report of the scan time.
- Key = always unlocked; the SIMATIC 545 does not have a key.
- Mode = Run, Program, or Hold; shows operating mode.
- Special Function/Loop Mode = Run indicates the system is running in the Special Function/Loop Mode.
- Special Function/Loop Follows PLC to Program Mode — informational comment.
- PLC Fatal Error = PLC GOOD LED turns off, I/O ports disabled, discrete outputs turned off, word outputs held in their last valid state, communication ports cleared and re-initialized, pending or queued tasks cleared.
- PLC Non-fatal Error = Scan overrun, I/O base failure, Special Function port failure, I/O module failure, I/O table mismatch.
- Loop Fatal Error = a memory, operating system, or diagnostic failure in PID loop or Special Function program processing. Status Word 161 provides status for the fatal error conditions.
- Loop Non-fatal Error = a port communication error, a math error such as divide-by-zero, an improper Special Function program call, or a scan timeslice overrun. Word 162 provides status for the non-fatal errors.
- Battery = ON indicates the backup battery is switched **ON** and good**;** OFF indicates the backup battery is switched **OFF** or bad**.**
- Download = OFF (not used in the SIMATIC 545).
- Program in XXX — RAM if system is operating from RAM memory; ROM if system is operating from an EEPROM (EPROM).
- S memory configured — number of bytes in S memory actually for loops, analog alarms, and SF programs.
- S memory available — Number of free bytes remaining in S-memory.

2.11 Status Words

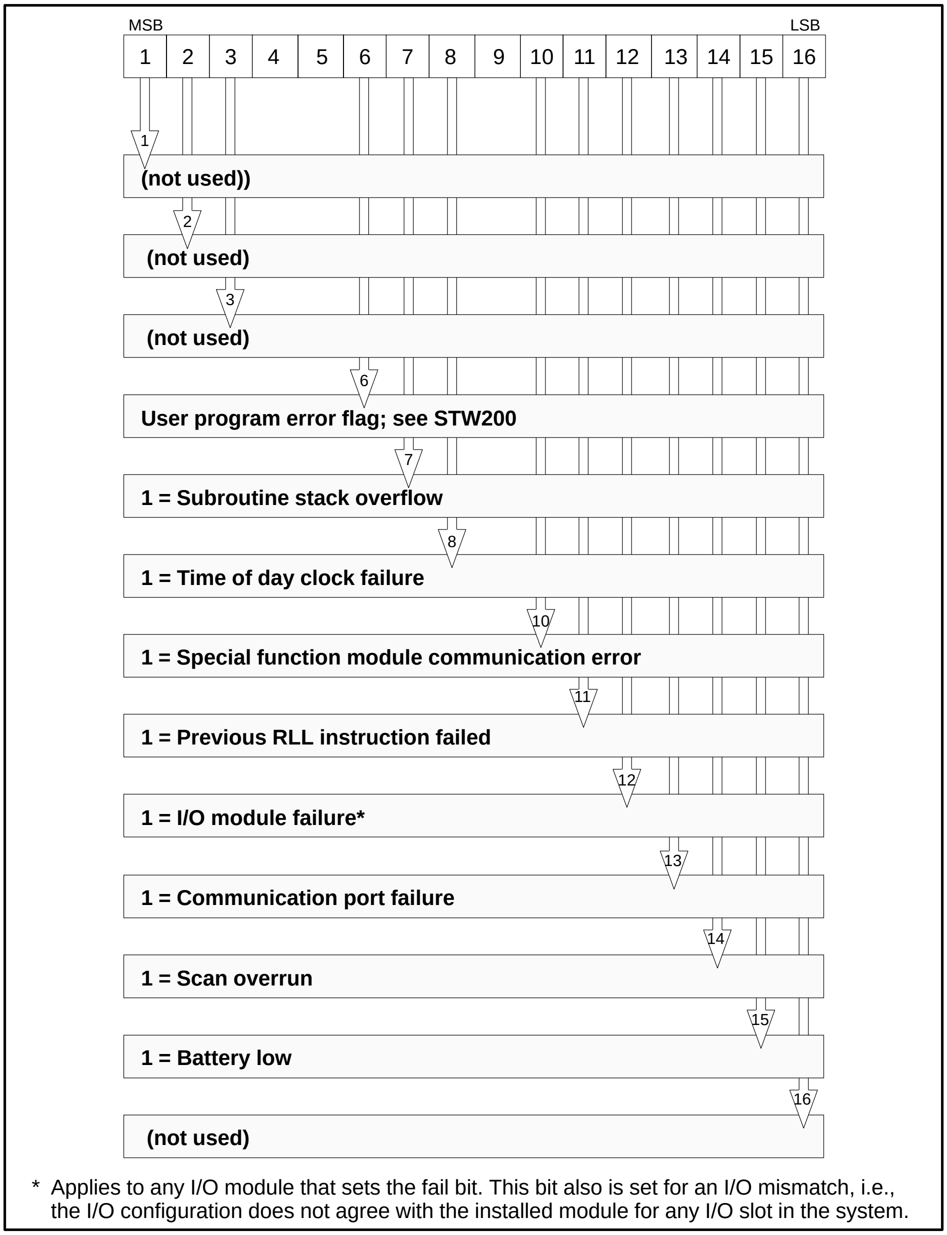
Overview

In addition to auxiliary functions, the controller provides operation information in the form of 16-bit status words. Status words can be read with your programming device. Status words can also be contained within a RLL program which allows the system to execute diagnostics during run-time conditions.

NOTE: To provide a consistent view of status words format, this section shows all words with actual binary data as it appears in 16-bit memory locations; however, TISOFT allows you to display status words in integer format (where appropriate). This frees you from having to manually translate binary data.

Each status word description explains the function or purpose of each bit within the word. If a bit is not used, it is not described. If several bits perform a single function, they are described by a single definition. If a status word is not used, it is noted accordingly.

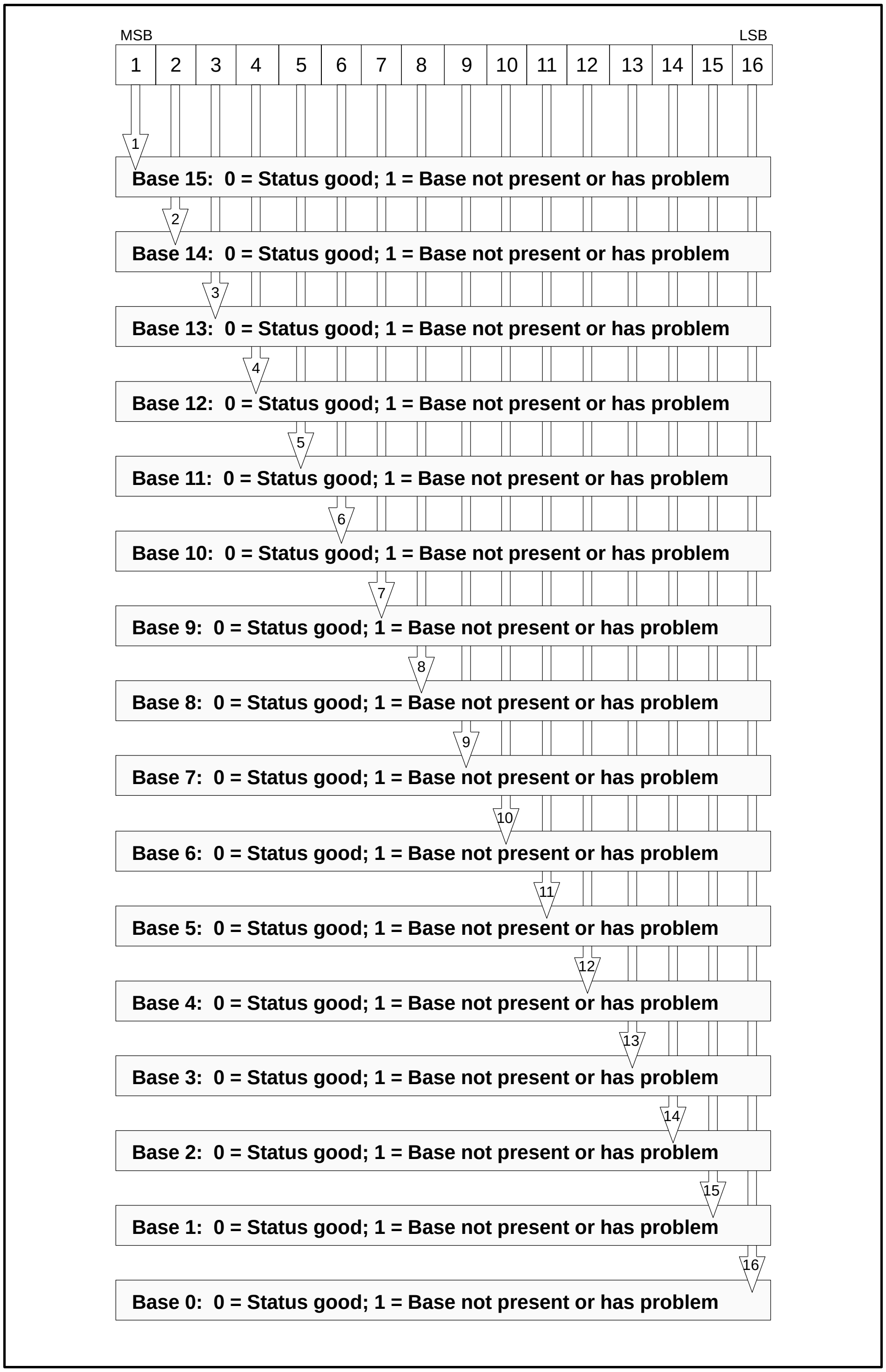
STW01: Non–fatal Errors



I003519

NOTE: STW1 cannot be accessed by a multi-word move instruction, e.g., MOVE, MOVW. STW1 is a local variable that is only valid within a given RLL task. Do not do multiple-word move operations that begin with STW1.

STW02: Base Controller Status



I003520

STW03 – STW09:

Status words 03 through 09 are not used.

STW10: Dynamic Scan Time

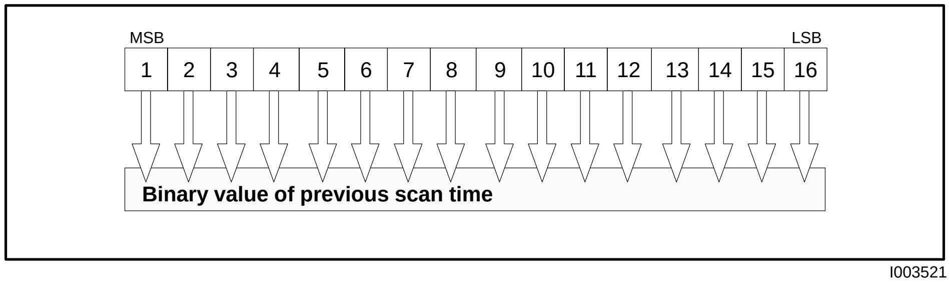


Figure 2-10 illustrates an example of STW10 containing a scan time of 145 ms.

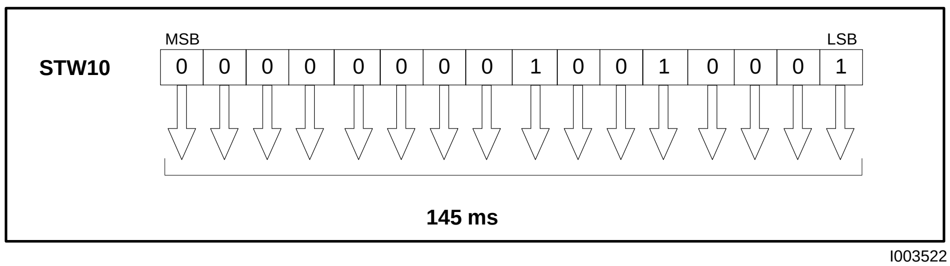


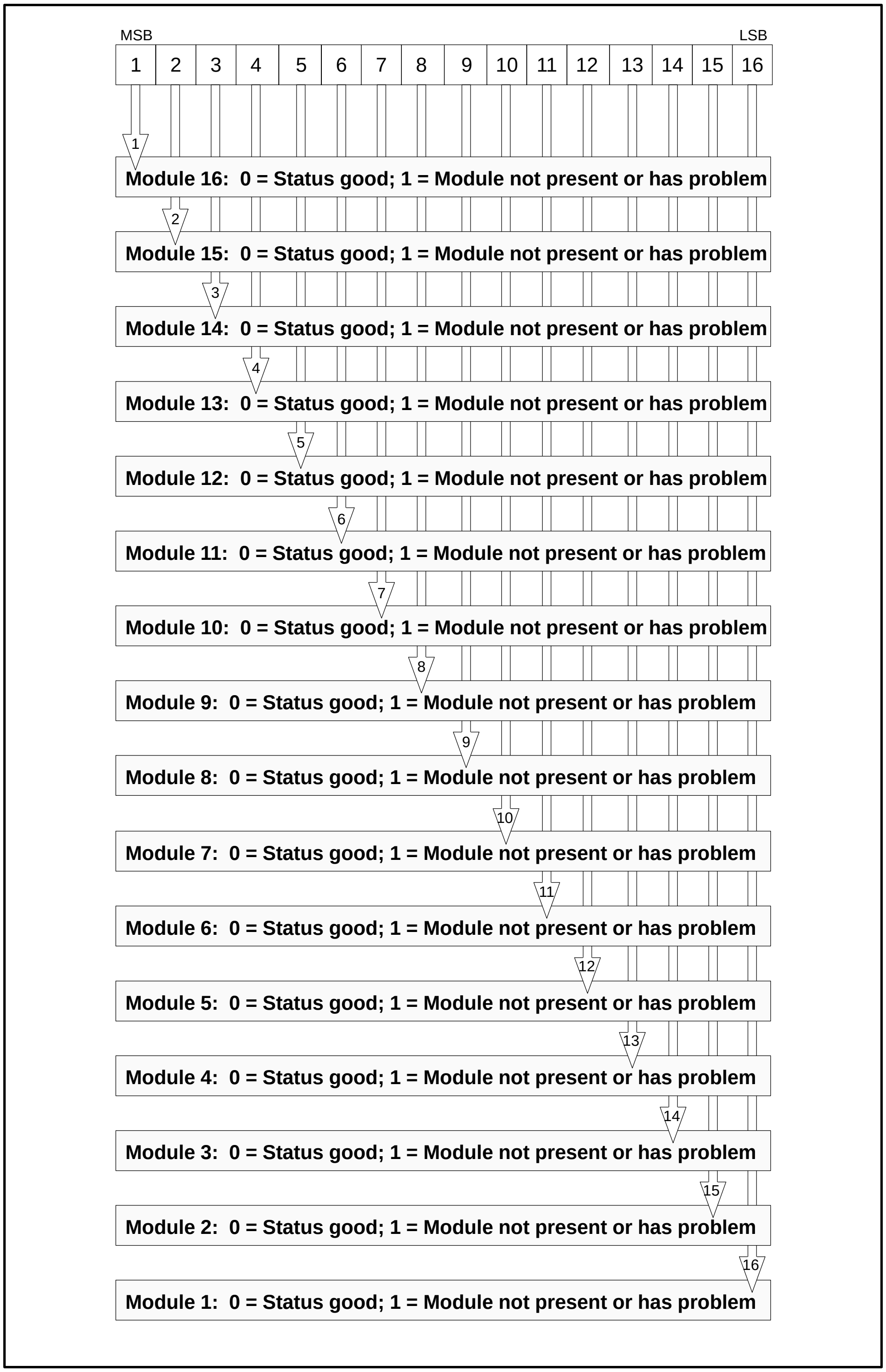
Figure 2-10 Example of Status Word Reporting Scan Time

STW11 – STW26: I/O Module Status

Status words 11 through 26 indicate the status of the modules present in each base. Table 2-18 lists the status words that correspond to the bases. The illustration on the next page shows the content of these status words.

Table 2-18 Status Words 11 Through 26

Channel 1 I/O modules	
Status word	**Indicates**
11	Modules on Base 0
12	Modules on Base 1
13	Modules on Base 2
14	Modules on Base 3
15	Modules on Base 4
16	Modules on Base 5
17	Modules on Base 6
18	Modules on Base 7
19	Modules on Base 8
20	Modules on Base 9
21	Modules on Base 10
22	Modules on Base 11
23	Modules on Base 12
24	Modules on Base 13
25	Modules on Base 14
26	Modules on Base 15



MSB
LSB
1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16
Module 16: 0 = Status good; 1 = Module not present or has problem
Module 15: 0 = Status good; 1 = Module not present or has problem
Module 14: 0 = Status good; 1 = Module not present or has problem
Module 13: 0 = Status good; 1 = Module not present or has problem
Module 12: 0 = Status good; 1 = Module not present or has problem
Module 11: 0 = Status good; 1 = Module not present or has problem
Module 10: 0 = Status good; 1 = Module not present or has problem
Module 9: 0 = Status good; 1 = Module not present or has problem
Module 8: 0 = Status good; 1 = Module not present or has problem
Module 7: 0 = Status good; 1 = Module not present or has problem
Module 6: 0 = Status good; 1 = Module not present or has problem
Module 5: 0 = Status good; 1 = Module not present or has problem
Module 4: 0 = Status good; 1 = Module not present or has problem
Module 3: 0 = Status good; 1 = Module not present or has problem
Module 2: 0 = Status good; 1 = Module not present or has problem
Module 1: 0 = Status good; 1 = Module not present or has problem

I003523

STW11–26: (continued)

The controller reports an I/O mismatch (an installed module does not agree with the I/O configuration) as a failed I/O module. Although the module has not actually failed, you must enter correct I/O configuration data or install the proper module to correct the failure report.

In Figure 2-11, the 1 in Bit 10 indicates that slot 7 in Base 4 contains a defective or incorrectly configured module (I/O mis-match). All other slots either contain a working module or no module at all.

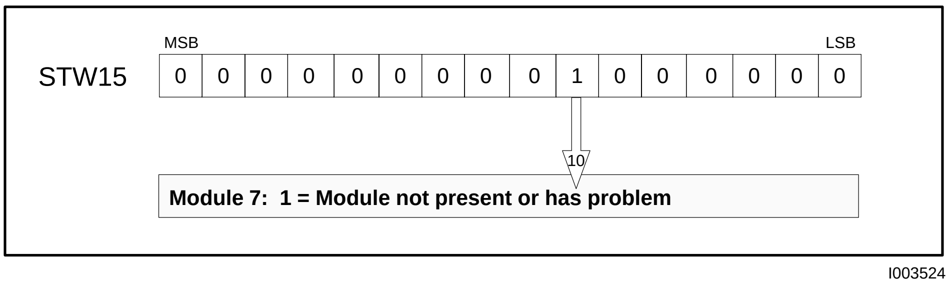


Figure 2-11 Example of Status Word Reporting a Module Failure

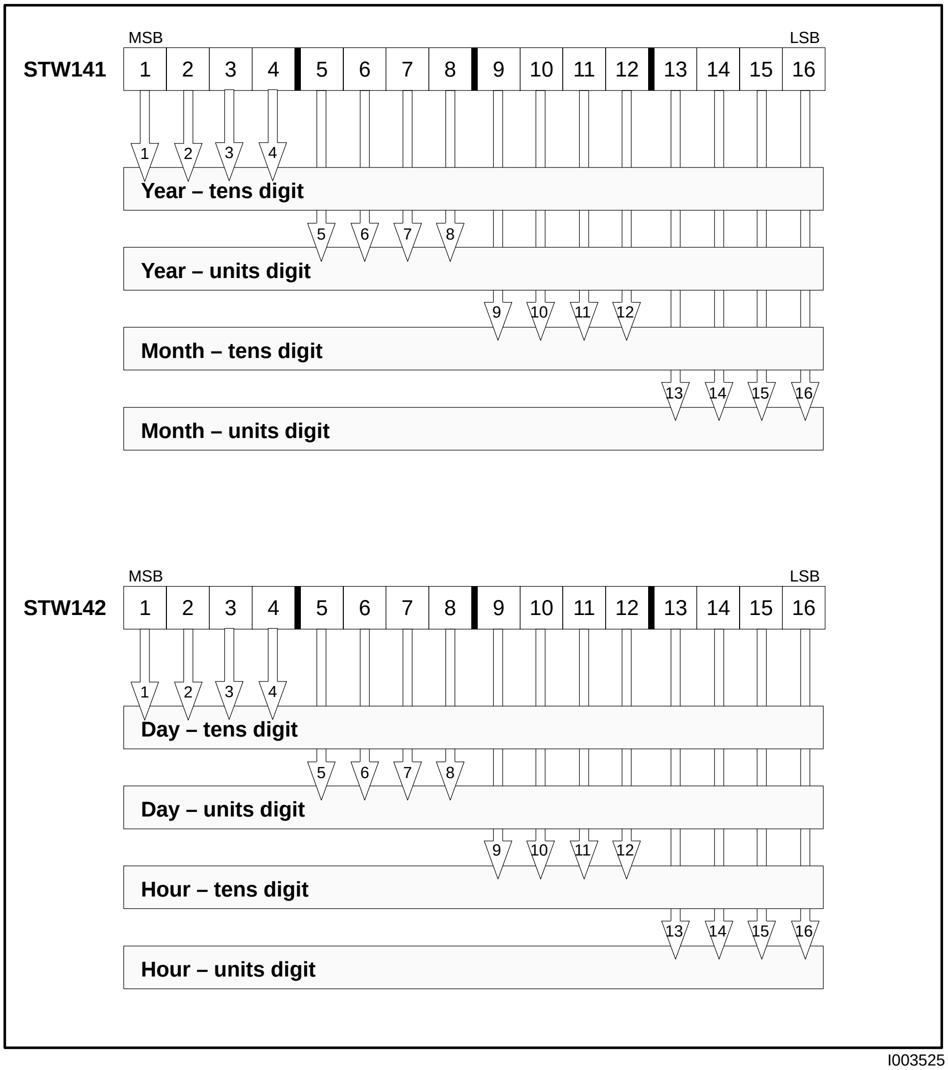
---

NOTE: When a remote base loses communication with the controller, the appropriate bit in STW02 shows a 1. The bits in the status word (STW11–STW26) corresponding to the modules in that base show zeroes, even if modules on that base have failed or been incorrectly configured. That is, base modules are not shown as failures in a base that is not communicating.

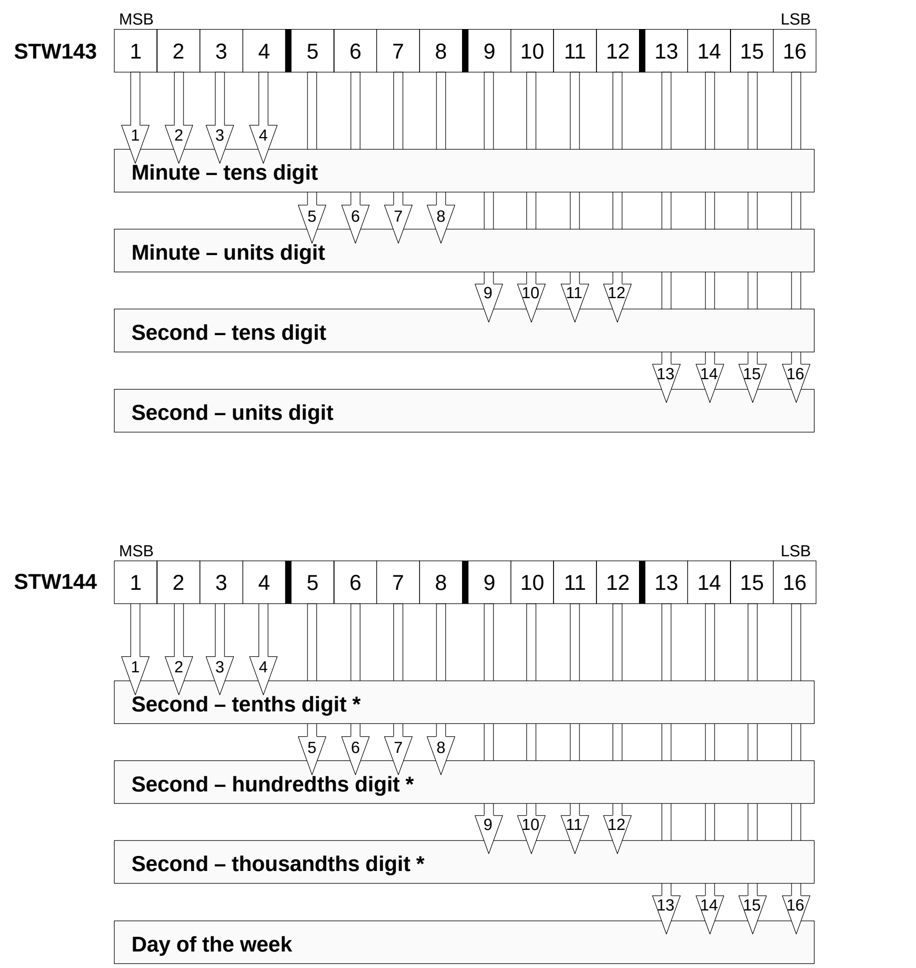
When you disable a base from the TISOFT I/O Configuration Screen, all bits in the status word (STW11–STW26) that corresponds to that base are set to zero.

STW27 – 140: Status words 27 through 140 are not used.

STW141–STW144: Date, Time, and Day of Week



Status Words (continued)



* The resolution of these units of time is controller specific. A controller fills a field with zeros for time units that it does not support.

I003526

Figure 2-12 illustrates how the clock information is displayed, using BCD, for the controller on the date: Monday, 5 October, 1992 at 6:39:51.767 P.M. Note that the 24-hour (military) format is used, and Sunday is assumed to be day 1.

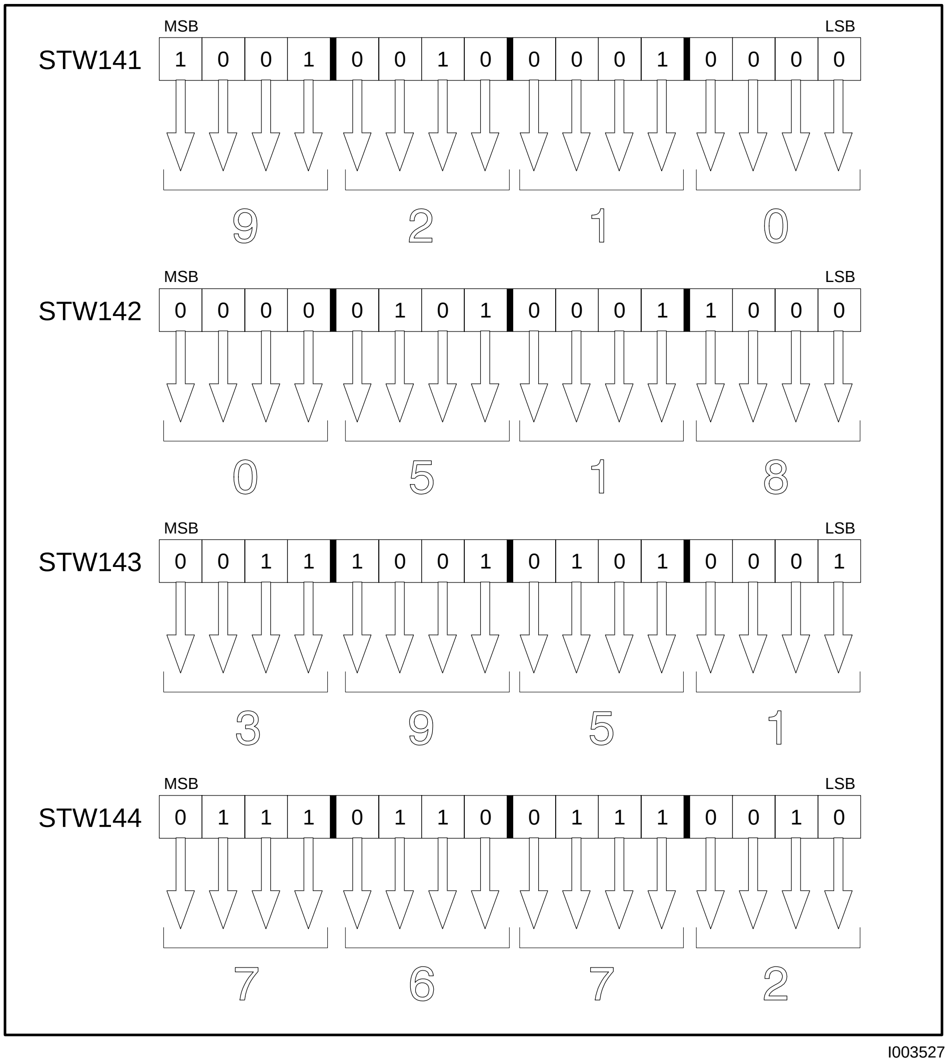
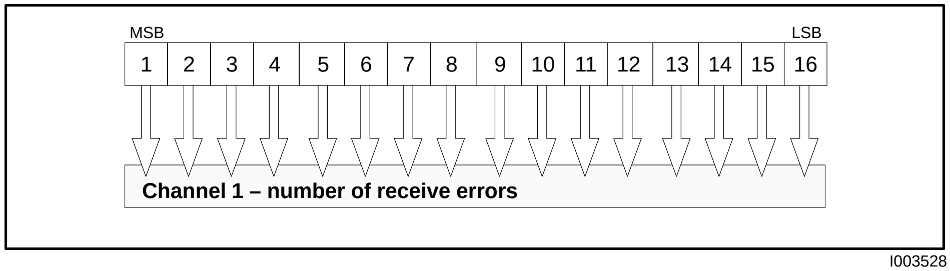


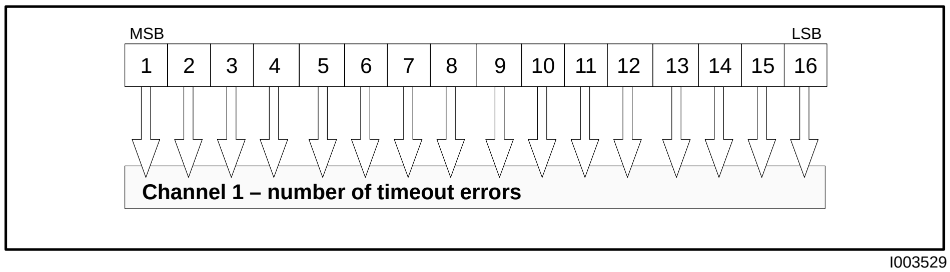
Figure 2-12 Example of Status Words Reporting Time

Status Words (continued)

STW145: Receive Errors for Channel 1



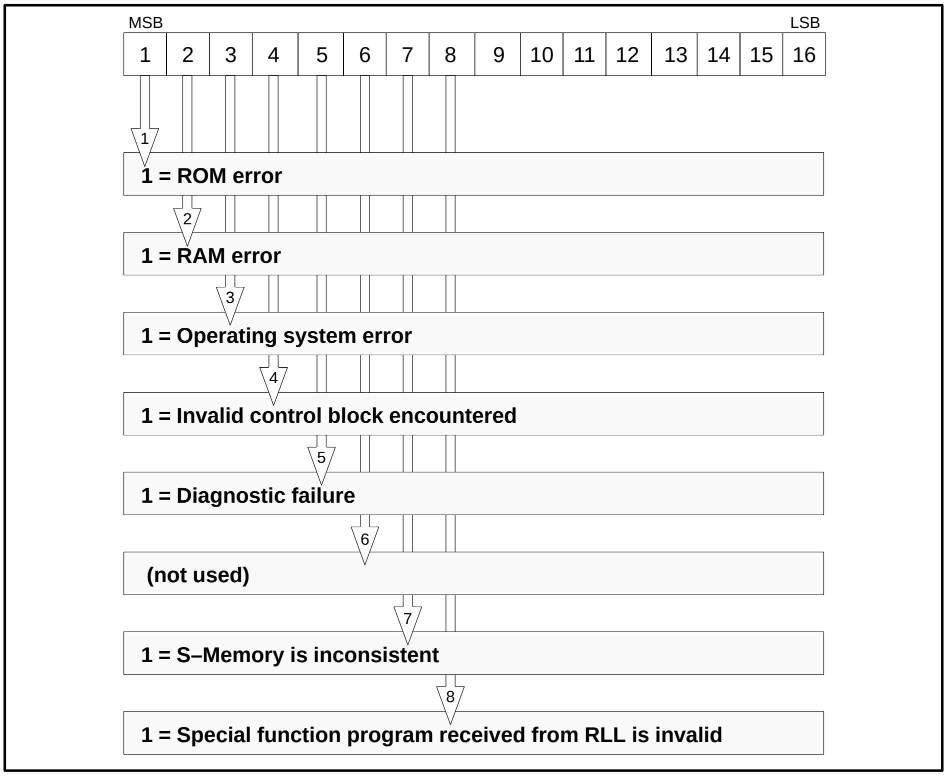
STW146: Timeout Errors for Channel 1



NOTE: A typical controller system should have no more than one detected and corrected error over the I/O link per 20,000 scans. If this error rate is exceeded, it may indicate a possible wiring or noise problem. Three consecutive errors to an RBC causes the base to be logged off and the corresponding bit in STW2 to be set.

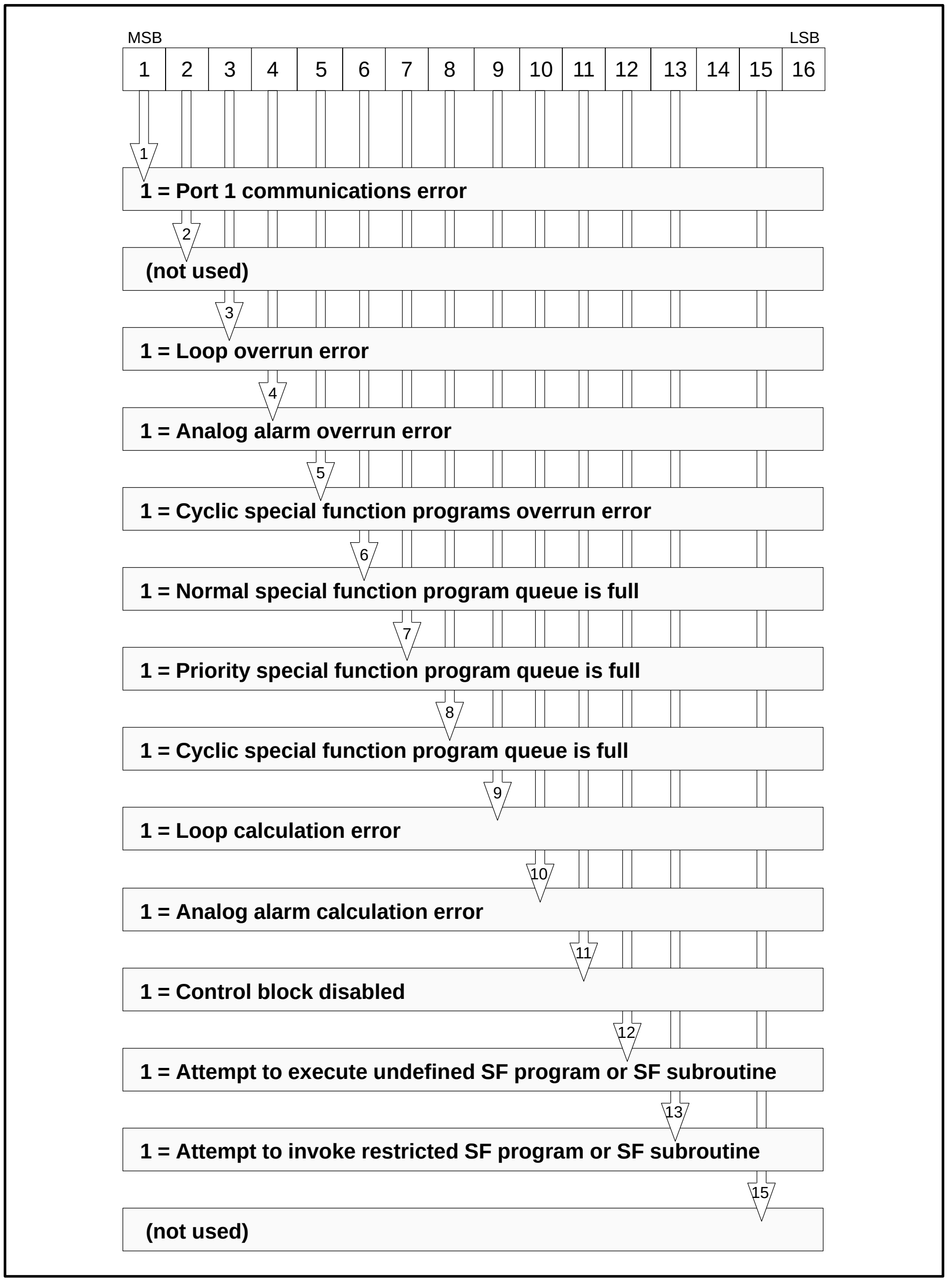
STW147–160 Status words 147 through 160 are not used.

STW161: Special Function Processor Fatal Errors



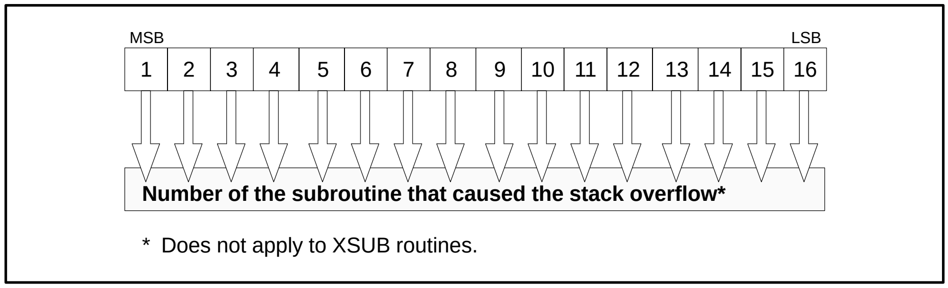
I003544

STW162: Special Function Processor Non-fatal Errors



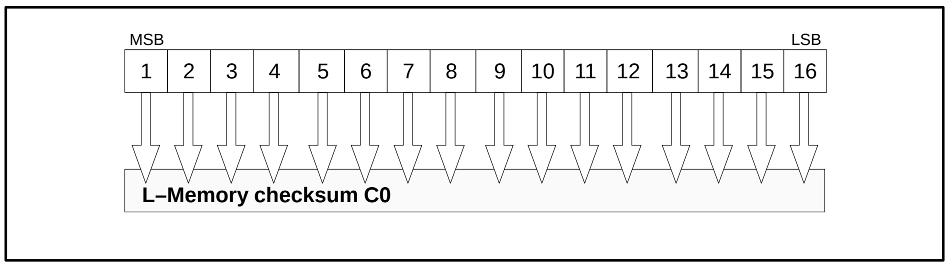
I003545

STW163:
RLL Subroutine
Stack Overflow



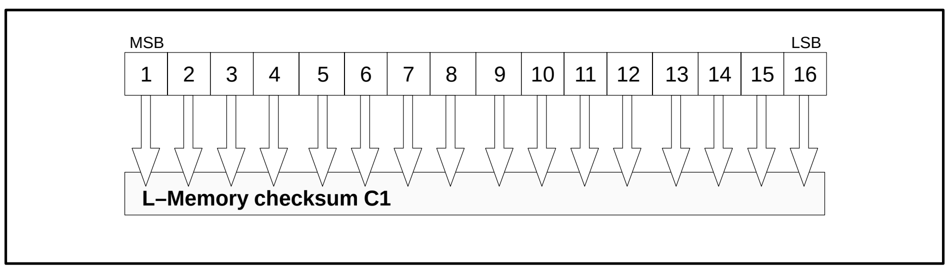
I003546

STW164 – STW165:
L–Memory
Checksum C0



I003547

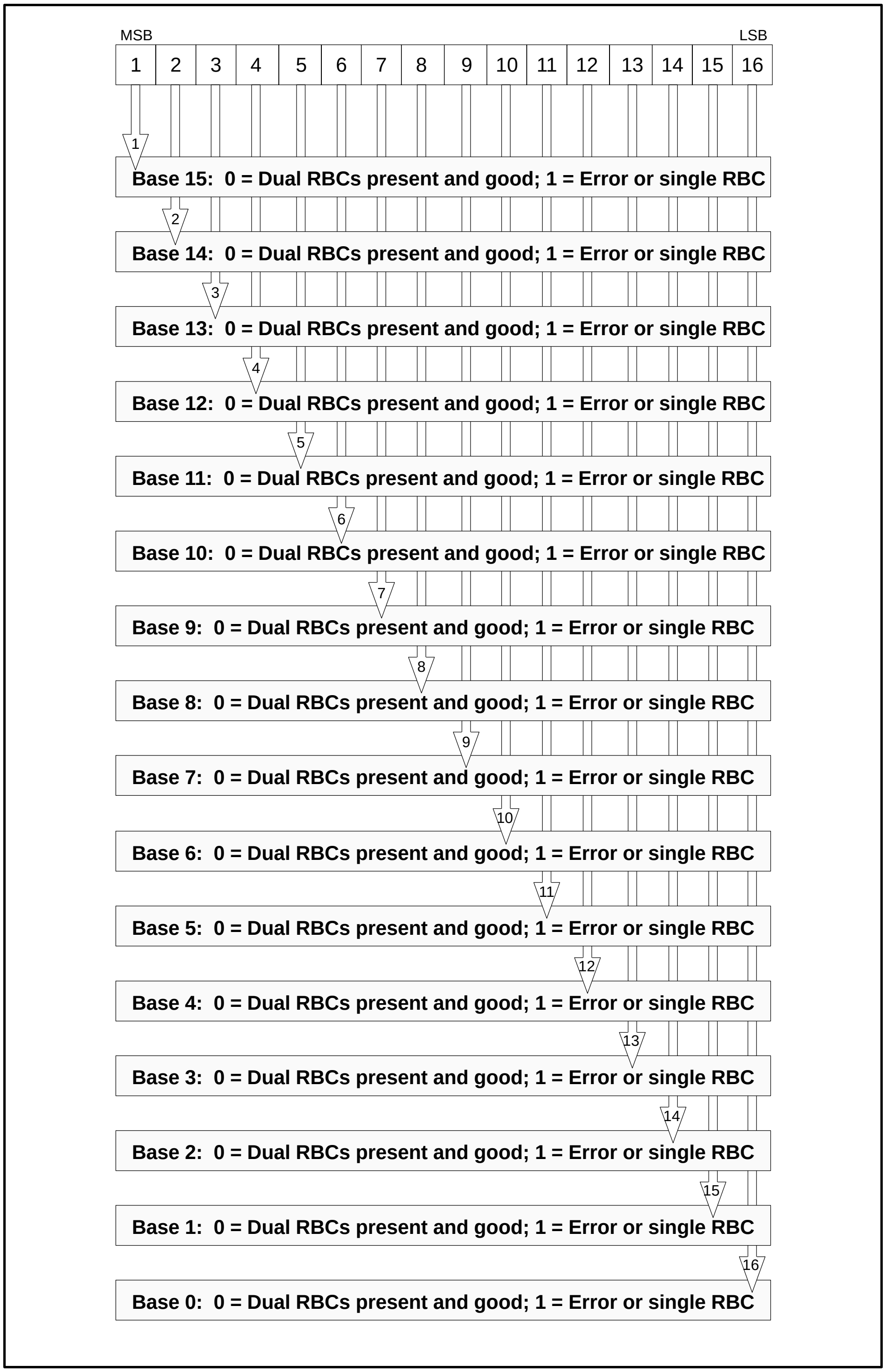
STW166 – STW167:
L–Memory
Checksum C1



I003548

Status Words (continued)

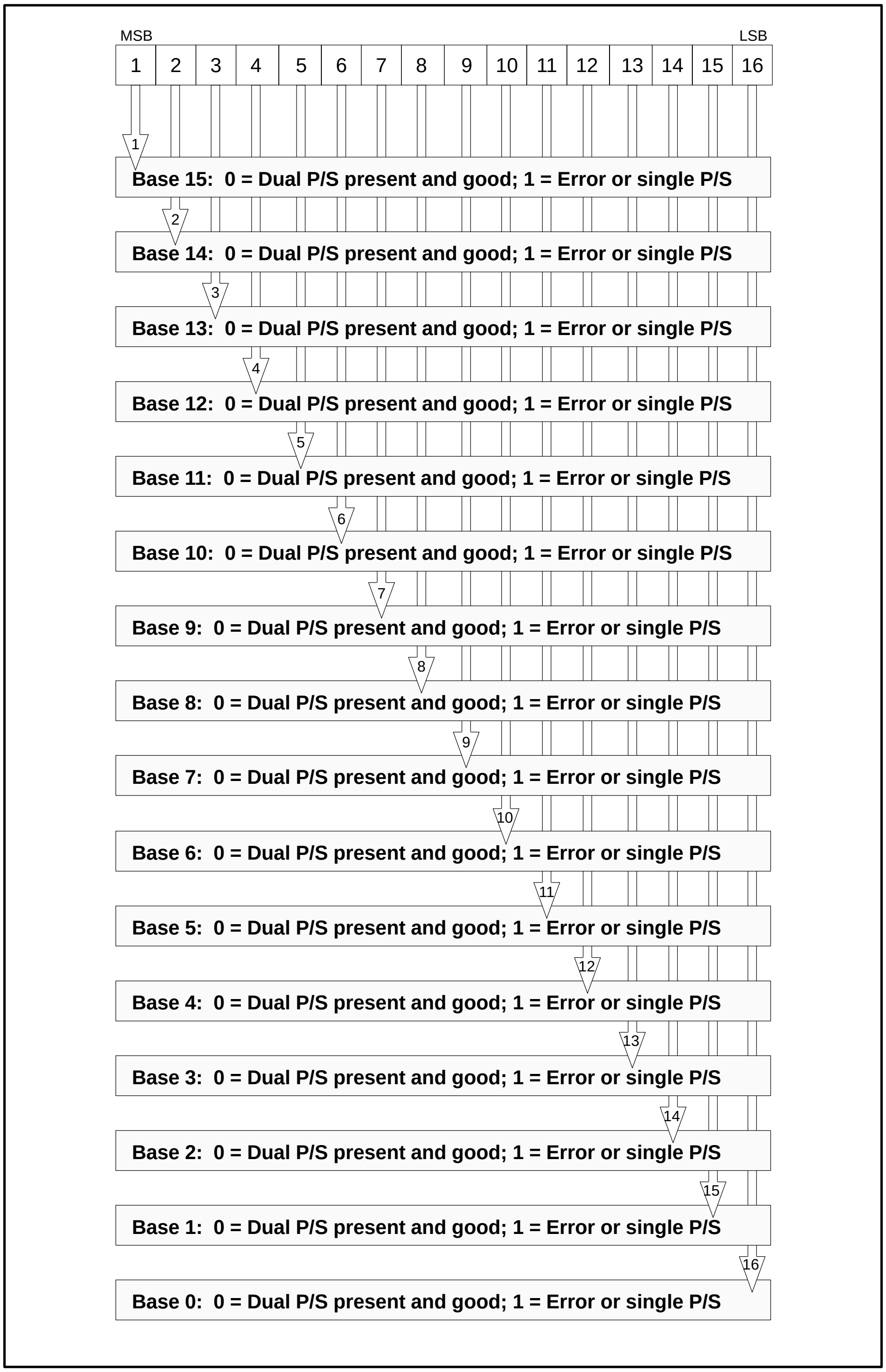
STW168: Dual RBC Status



I003549

STW169 – 175: Status words 169 through 175 are not used.

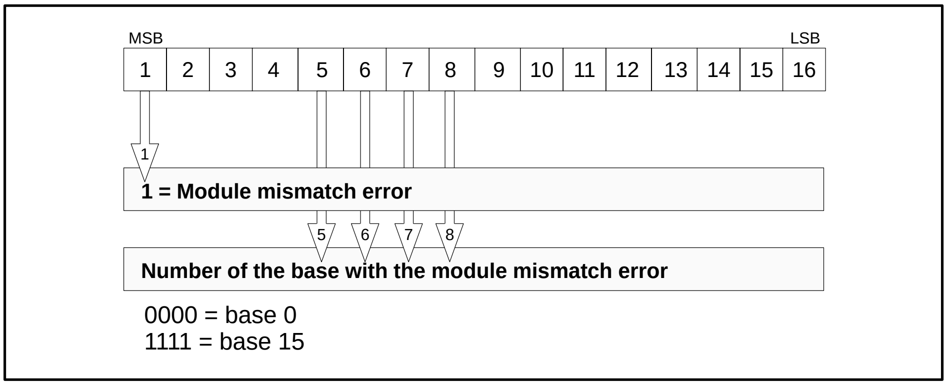
STW176: Dual Power Supply Status



I003550

STW177 – STW183: Status words 177 through 183 are not used.

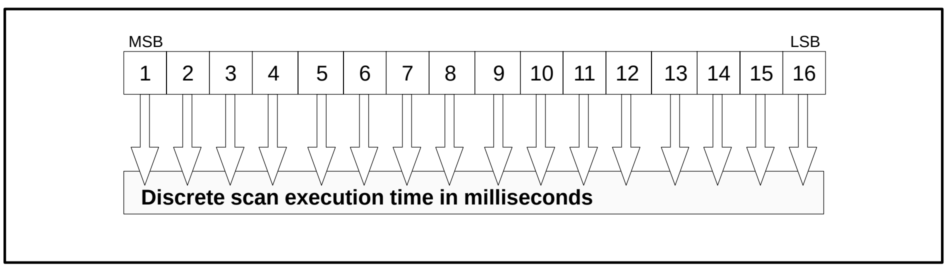
STW184: Module Mismatch Indicator



I003551

STW185 – STW191: Status words 185 through 191 are not used.

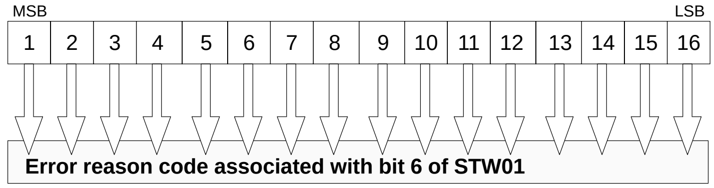
STW192: Discrete Scan Execution Time



I003552

STW193 – STW199: Status words 193 through 199 are not used.

STW200: User Error Cause



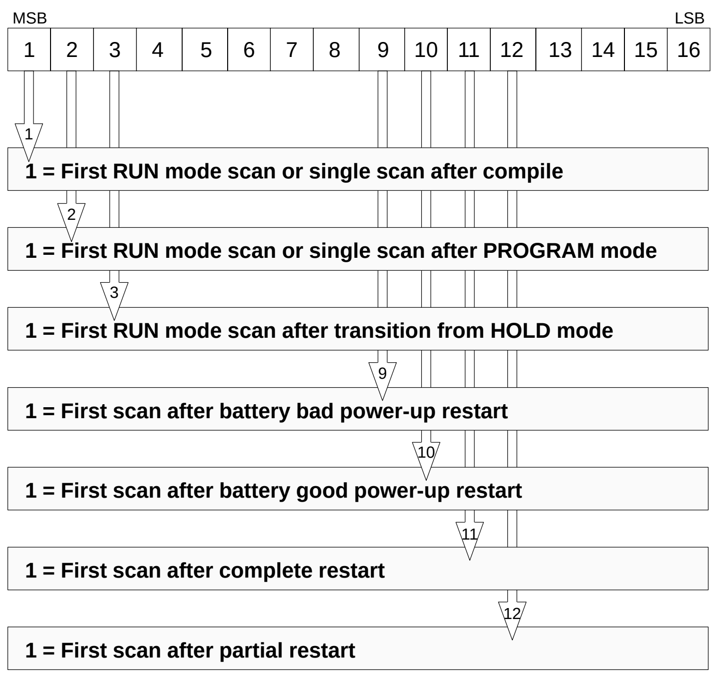
Valid reason codes:

0	No error
1	(Not used with 545)
2	(Not used with 545)
3	(Not used with 545)
4	Exceeded subroutine nesting level
5	Table overflow
6	Attempted to call a non-existent subroutine

STW200 reports the first error that occurs when you run the program. After you correct the problem that causes the first error, recompile and run the program again. If there is a second problem, the error code for this problem is recorded in STW200. Subsequent errors are recorded accordingly.

I003553

STW201: First Scan Flags

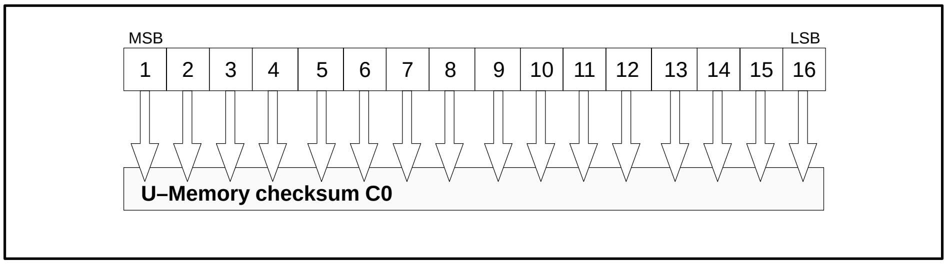


I003554

Status Words (continued)

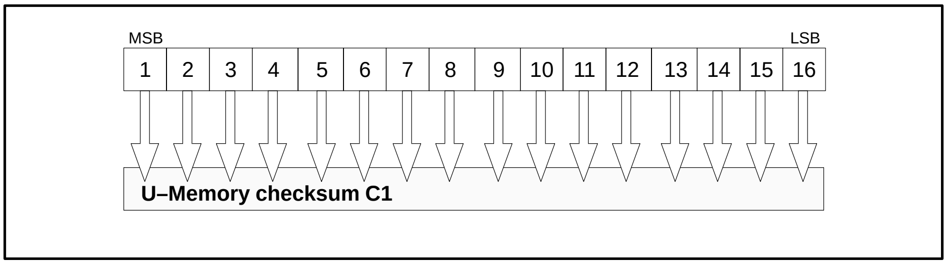
STW202 – STW205: Status words 202 through 205 are not used.

STW206 – STW207: U–Memory Checksum C0



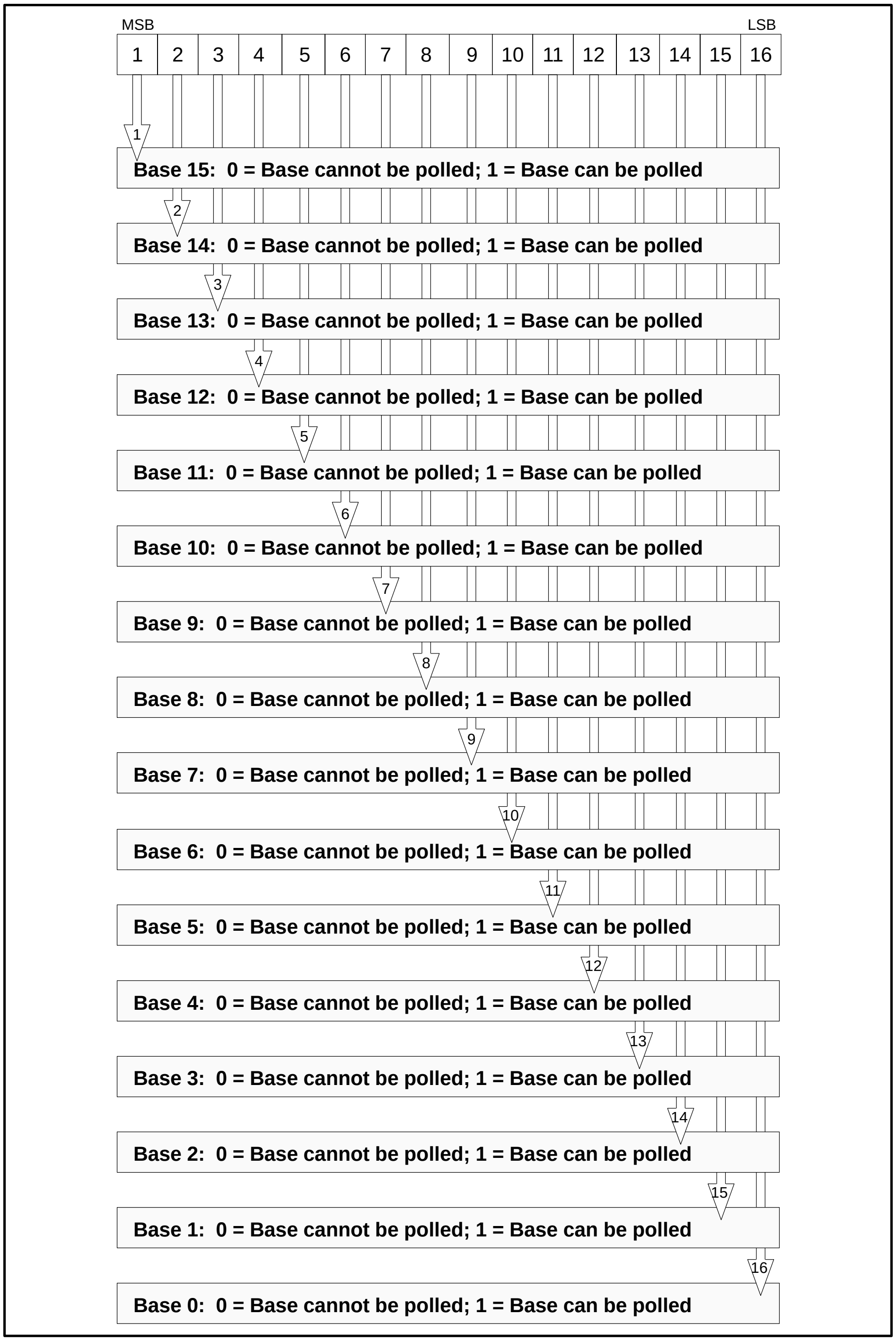
I003559

STW208 – STW209: U–Memory Checksum C1



I003560

**STW210:
Base Poll Enable Flags**

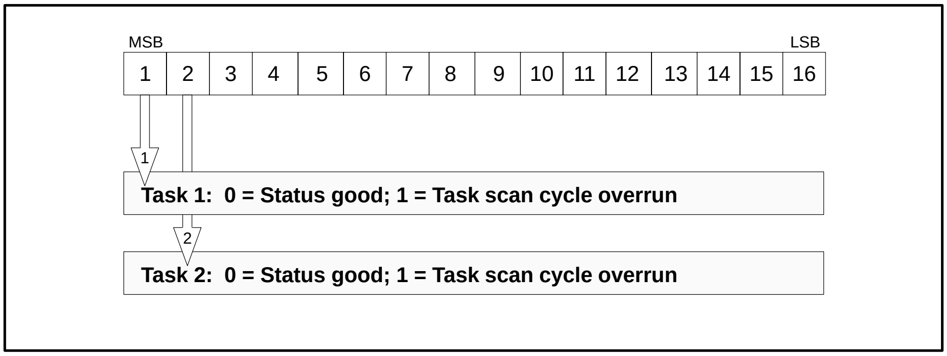


I003561

STW211 – STW218: Status words 211 through 218 are not used.

Status Words (continued)

STW219: Cyclic RLL Task Overrun



I003562

2.12 Specifications

Table 2-19 Environmental Specifications

Operating Temperature	0° to 60° C (32° to 140° F)
Storage Temperature	–40° to +70° C (–40° to 158° F)
Relative Humidity	5% to 95% non-condensing
Pollution Degree	2, IEC 664, 684A
Vibration Sinusoidal Random	IEC 68-2-6, Test Fc; 0.15 mm, peak-to-peak, 10–57 Hz; 2.0g 57–150Hz NAVMAT P-9492 or IEC 68-2-34 Test Fdc with 0.04g^2/Hz, 80–350Hz, and 3dB/octave rolloff, 80–20Hz and 350–2000 Hz at 10 min/axis
Impact Shock	IEC, 68-2-27, Test Ea; Half sine, 15g 11ms
Electric Noise Immunity Conducted noise: Radiated noise: Electrostatic discharge:	 IEC 801, Part 4, Level 3 MIL STD 461B, Part 4; CS01, CS02, CS06 IEC 255-4 EEC 4517/79 Com(78) 766 Final, Part 4 IEEE 472, 2.5kV IEC 801, Part 3, Level 3 MIL STD 461B, Part 4; RS01, RS02 IEC 801, Part 2, Level 4, (15kV)
System Isolation	Isolation (user-side to controller-side): 1500V rms
Corrosion Protection	All parts are of corrosion resistant material or are plated or painted as corrosion protection.
Agency Approvals	UL Listed (UL508 industrial control equipment) CSA Certified (CSA142 process control equip.) FM Approved (Class I, Div. 2, Haz. Location)

Table 2-20 SIMATIC 545 Electrical Specifications

Maximum Power Drain	4 W @ +5 VDC, 0.2 W @ –5 VDC
Battery Backup	3.0 V Lithium battery (part 2587678–8005)
Communication Ports	RS-232-C/RS-423 RS-422
User Memory	96Kwords, RAM (user configurable) 64K words, EEPROM (EPROM) total
Memory Backup	EEPROM (part #2590852–8003) EPROM (part #2590848–8001) Battery backed static RAM

Chapter 3
Input/Output System

3.1 Description

Versatility

The Series 505 I/O subsystem is a versatile, high-speed industrial input/output system which provides a durable, electrical noise-resistant interface between the control system and the external world. With a wide range of plug-in I/O modules, you can control or gather data from almost any factory floor sensor, actuator, or intelligent device.

Series 505 Base Assemblies

The controller and all I/O modules are housed in a Series 505 I/O base assembly. An I/O base assembly has slots for the PLC or Remote Base Controller (RBC), a power supply, and the I/O modules. Four base assembly models are currently available.

- 4 I/O slots (PPX:505–6504,)
- 8 I/O slots (PPX:505–6508)
- Redundant I/O base with 11 I/O slots (PPX:505–6511)
- 16 I/O slots (PPX:505–6516)

The 8 I/0 Slots (PPX:505–6511) provides 2 power supply slots, 2 Remote Base controller slots, and 11 I/O slots. It has the same physical dimensions as the 16-slot base. The power supplies can be any combination of the 24 VDC supply (PPX:505–6663) or the 110/220 VAC Redundant AC supply (PPX: 505–6660–A). The 110/220 VAC supply (PPX:505–6660) cannot be used in a redundant configuration.

The Remote Base Controller (PPX:505–6851A) can be used in a stand-alone configuration or a redundant media mode. A Series 505 controller may also be used in the Redundant I/O rack (only one controller) with dual power supplies.

Any base assembly may be used at any point in the system, depending on the number of I/O points required.

Local and Remote Operation

The I/O subsystem is grouped into local and remote I/O categories, which are defined by their physical location. The local I/O consists of those modules located in the same base assembly as the controller. If you install the controller in a 16-slot base assembly (PPX:505–6516), the local I/O may consist of as many as 16 I/O modules.

You can connect up to 15 additional remote I/O base assemblies to the system via an RS-485 communication link. The I/O modules in these bases make up the remote I/O as shown in Figure 3-1.

Individual I/O modules in the remote bases communicate with the controller through RBCs (PPX: 505-6851A). The Smart Slice emulates a remote base providing ten inputs and six outputs. The RBC and Smart Slice receive new output updates, and returns the current input status once every controller scan. The controller remote I/O consists of one channel, which provides up to 2048 I/O points.

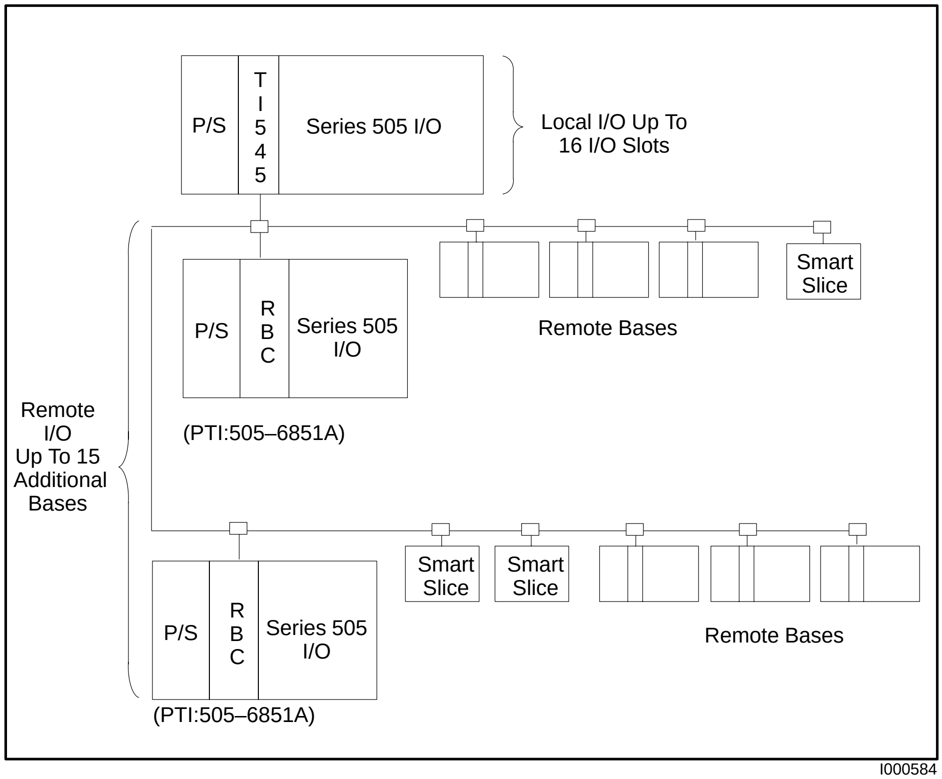


Figure 3-1 I/O System Layout

Synchronous I/O

Unlike some systems, all Series 500 and Series 505 controllers lock I/O updates into step with the logic scan. All I/O points, both local and remote, are updated with real-time data every logic scan, thus avoiding the difficult debug and troubleshooting problems which result from asynchronous I/O systems.

The remote I/O system operates without scan latency at a data rate of 1 megabaud. A remote I/O system with 256 discrete input and 256 discrete output points located on two remote bases can be updated in approximately 3 milliseconds—even at the maximum distance of 3,300 feet.

Data Integrity

The Series 505 I/O System is highly resistant to electrical and radio frequency noise commonly found in industrial environments. Unlike other I/O systems which use simple parity protection, the Series 505 remote I/O system contains sophisticated error checking protection. The I/O communications channel uses a Miller encoding technique, an extended message preamble, and a 16-bit Cyclic Redundancy Check (CRC-16). This results in an undetected error rate of about one undetected error every 100 years.

Base Log-out

If a transmission error occurs during remote I/O data transfer (e.g., a CRC error or no response), the controller will try to re-establish communications up to three times. If all three attempts fail, the remote base is logged-out of the I/O map. Then the controller will periodically attempt to reconnect logged-out bases and will resume communications once the remote base returns to normal operation.

Selectable Base Polling

An option is provided to selectively enable or disable communications to individual I/O racks in normal operation. The selectable polling option permits tailoring the I/O update cycle to only the selected racks.

Configurable Addressing

Your control program may be developed independently of the physical I/O system. I/O is configured just prior to system startup without altering the ladder logic. This allows you to keep your original control program intact while expanding or modifying the system.

The Series 505 control system uses a simple I/O addressing scheme. Each individual I/O point is addressed by a type identifier and an address. The type identifiers are:

- X — Discrete input addresses (1–2048)
- Y — Discrete output addresses (1–2048)
- WX — Word/Analog input (1–1024)
- WY — Word/Analog output (1–1024)

Configuration of the I/O subsystem consists of assigning addresses to the individual I/O points on a module-by-module basis. Series 505 I/O modules may be logically configured differently from their physical configuration—in other words, I/O labels (X1, X2, X3...) are not determined by the base slot in which they reside.

For example, an 8-point discrete input module, located in slot #2 of remote base #3, may be assigned as X1 – X8, or as X2041 – X2048.

3.2 I/O Communication

Cabling

The Series 505 I/O System supports up to 16 bases (including the local base). Remote I/O bases may be located as far as 3,300 feet from the controller CPU. The I/O system uses shielded twisted pair cabling for the trunk and droplines. The trunk line is spliced with terminal blocks for drop line installation. Each drop line can support multiple remote bases. Figure 3-2 shows an example of Trunk and Drop Line.

NOTE: Termination resistors must be placed at the ends of the main trunk.

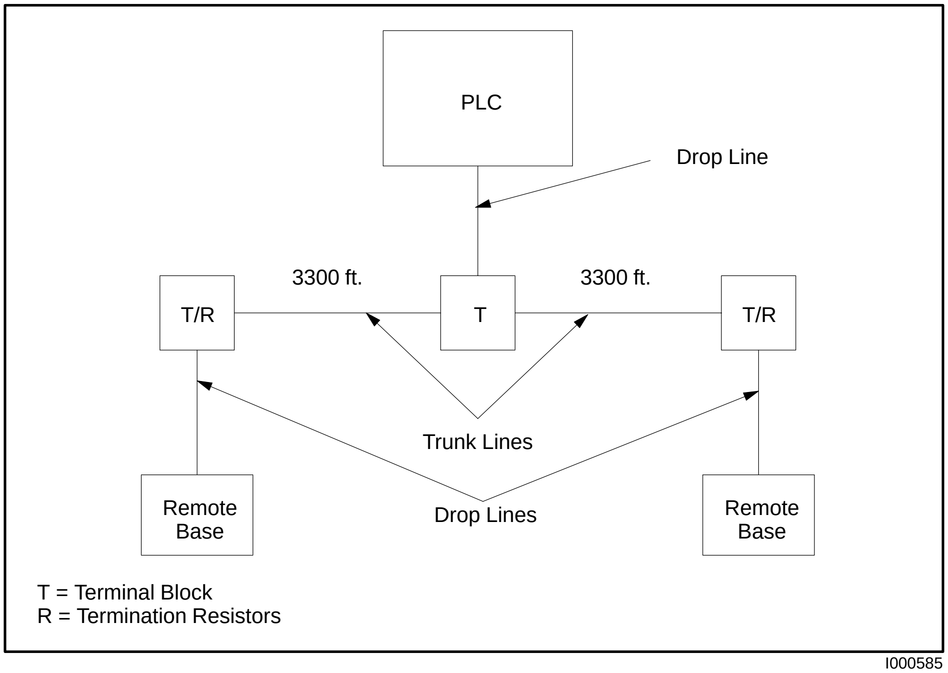


Figure 3-2 Trunk and Drop Line Example

Trunk Lines

The cable that connects terminal blocks is called a trunk line.The maximum trunk length depends on the cable type and number of drop lines used in your installation. See Table 3-1.

Table 3-1 Maximum Cable Length for Trunk Lines

Number of Drop Lines	Maximum Distance in Feet (Meters)*		
	9860 cable	*(Belden Cables) 9271 cable*	*9182 cable*
2 - 5	3300 (1000)	1100 (330)	2200 (660)
6	3200 (970)	1067 (320)	2133 (640)
7	3100 (940)	1033(310)	2067 (620)
8	3000 (910)	1000 (300)	2000 (600)
9	2900 (880)	967 (290)	1933 (580)
10	2800 (850)	933 (280)	1867 (560)
11	2700 (820)	900 (270)	1800 (540)
12	2600 (790)	867 (260)	1733 (520)
13	2500 (760)	833 (250)	1667 (500)
14	2400 (730)	800 (240)	1600 (480)
15	2300 (700)	767 (230)	1533 (460)
16	2200 (670)	733 (220)	1467 (440)
17	2100 (640)	700 (210)	1400 (420)
* Figures in parentheses indicate distance in meters.			

NOTE: Cable types 9860 and 9271 can be mixed on the trunk line, but cable type 9182 must be used alone.

PLC and RBC Drop Lines

The drop line is the cable that connects the PLC or RBC to a terminal block. Use Belden cable type 9271 or 9182 for all drop lines. Each drop line must not exceed 30 feet (9.1 M) in length.

Cable Layout

There are several methods of routing communication cables for the RS-485 trunk line. Some methods provide better accessibility, while other methods provide better noise immunity. Regardless of the routing method, one of important design factor is cable distance.

Measure the maximum length (listed in Table 3-1) from the controller to the most distant tap. See Figure 3-3.

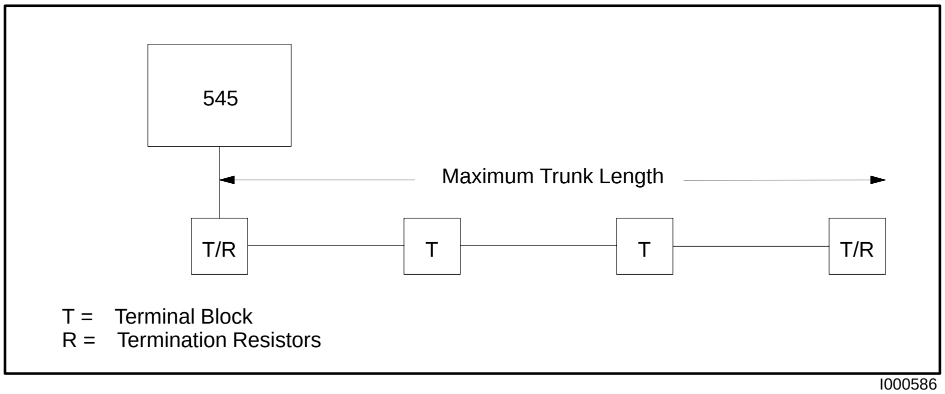


Figure 3-3 Maximum Trunk Length

By using a “T” configuration, you can increase the total trunk line length to twice the length specified in Table 3-1. Figure 3-4 shows a “T” configuration.

Place a maximum of two terminal blocks per 50 feet (15 M) on the trunk line.

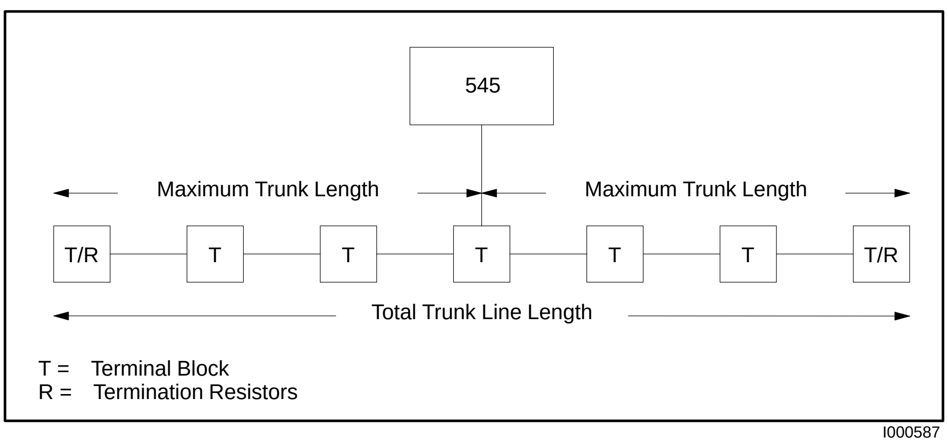


Figure 3-4 “T” Configuration

3.3 I/O Configuration

Entering the Configuration

One of the advantages of the Series 505 system environment is that you can use the programming software to configure the I/O devices. The menu-driven screens and coded hard keys make this an easy task. Figure 3-5 shows a sample I/O configuration screen.

```
                I/O MODULE DEFINITION FOR CHANNEL . . . . .  1      BASE . . . 00
                              I/O        NUMBER OF BIT AND WORD I/O          SPECIAL
     BASE       SLOT       ADDRESS      X          Y         WX        WY     FUNCTION
   ENABLED
                 01  . . . . .  0033  . . . . .  00 . . . . . .  32 . . . . . .  00 . . . . . .  00 . . . . . . . .   NO
                 02  . . . . .  0001  . . . . .  32 . . . . . .  00 . . . . . .  00 . . . . . .  00 . . . . . . . .   NO
                 03  . . . . .  0000  . . . . .  00 . . . . . .  00 . . . . . .  00 . . . . . .  00 . . . . . . . .   NO
                 04  . . . . .  0000  . . . . .  00 . . . . . .  00 . . . . . .  00 . . . . . .  00 . . . . . . . .   NO
                 05  . . . . .  0000  . . . . .  00 . . . . . .  00 . . . . . .  00 . . . . . .  00 . . . . . . . .   NO
                 06  . . . . .  0000  . . . . .  00 . . . . . .  00 . . . . . .  00 . . . . . .  00 . . . . . . . .   NO
                 07  . . . . .  0000  . . . . .  00 . . . . . .  00 . . . . . .  00 . . . . . .  00 . . . . . . . .   NO
                 08  . . . . .  0000  . . . . .  00 . . . . . .  00 . . . . . .  00 . . . . . .  00 . . . . . . . .   NO
                 09  . . . . .  0000  . . . . .  00 . . . . . .  00 . . . . . .  00 . . . . . .  00 . . . . . . . .   NO
                 10  . . . . .  0000  . . . . .  00 . . . . . .  00 . . . . . .  00 . . . . . .  00 . . . . . . . .   NO
                 11  . . . . .  0000  . . . . .  00 . . . . . .  00 . . . . . .  00 . . . . . .  00 . . . . . . . .   NO
                 12  . . . . .  0000  . . . . .  00 . . . . . .  00 . . . . . .  00 . . . . . .  00 . . . . . . . .   NO
                 13  . . . . .  0000  . . . . .  00 . . . . . .  00 . . . . . .  00 . . . . . .  00 . . . . . . . .   NO
                 14  . . . . .  0000  . . . . .  00 . . . . . .  00 . . . . . .  00 . . . . . .  00 . . . . . . . .   NO
                 15  . . . . .  0000  . . . . .  00 . . . . . .  00 . . . . . .  00 . . . . . .  00 . . . . . . . .   NO
                 16  . . . . .  0000  . . . . .  00 . . . . . .  00 . . . . . .  00 . . . . . .  00 . . . . . . . .   NO

FROM  RAM
                                                                     545            NETDATA
   EXIT–F1 SHOW–F2 READRM–F3 WRITRM–F4   EN/DIS–F6 DELCHN–F7 CLRBS–F8
```

I002092

Figure 3-5 I/O Module Definition Menu

You can configure the I/O on-line (communicating with the controller) or off-line (changes saved to disk). When you use on-line configuration, you can perform those tasks that require access to the controller. For instance, you must use on-line configuration to the controller to read the configuration of a base from the base itself. The enabling / disabling of each I/O base can be done on-line or off-line. If it is off-line, the enable/disable status of the base is saved to disk along with the other address information.

Reading the Configuration

You can display individual base configuration. For example, you can read base 1 and examine each slot to determine which I/O points are configured for that slot. You can also use TISOFT to read, clear, or compare the controller I/O and base configurations. Figure 3-6 shows a sample screen for a base-to-controller comparison.

I/O MODULE COMPARISON FOR CHANNEL . . . . 1 BASE 00

		FROM P/C					FROM BASE				
	SLOT	X	Y	WX	WY	SF	X	Y	WX	WY	SF
**	01	00	00	00	00	NO	00	08	00	00	NO
**	02	00	00	00	00	NO	00	08	00	00	YES
	03	00	00	00	00	NO	00	00	00	00	NO
**	04	00	00	00	00	NO	00	00	07	01	YES
	05	00	00	00	00	NO	00	00	00	00	NO
	06	00	00	00	00	NO	00	00	00	00	NO
	07	00	00	00	00	NO	00	00	00	00	NO
	08	00	00	00	00	NO	00	00	00	00	NO
	09	00	00	00	00	NO	00	00	00	00	NO
	10	00	00	00	00	NO	00	00	00	00	NO
	11	00	00	00	00	NO	00	00	00	00	NO
	12	00	00	00	00	NO	00	00	00	00	NO
	13	00	00	00	00	NO	00	00	00	00	NO
	14	00	00	00	00	NO	00	00	00	00	NO
	15	00	00	00	00	NO	00	00	00	00	NO
	16	00	00	00	00	NO	00	00	00	00	NO

COMPARISON NOT EQUAL – FAILED MODULE IN BASE

545 NETDATA RN

EXIT–F1

I002094

Figure 3-6 Base and Controller Comparison

3.4 I/O Cycle Description

During the normal I/O update cycle, the controller stores input data obtained from the I/O bases into the image registers and transfers output data from the image registers to the I/O bases. The length of the I/O update cycle is dependent upon the number of bases and types of modules (analog, discrete or intelligent). All I/O points are fully updated each scan.

The normal I/O update is handled automatically by the controller and I/O subsystem for all types of I/O modules (analog, discrete, and intelligent). Special RLL programming is not required to access any I/O device.

The normal I/O update occurs just prior to the execution of the main RLL program.

Immediate I/O

The immediate I/O feature allows your RLL application program to access an I/O point in the I/O module multiple times per controller scan. This feature enables you to sample fast-changing inputs more often, providing a faster response to the application.

For immediate I/O updates the controller does the following:

- Reads discrete inputs directly from the module as part of power flow computations; does not update the image register (immediate contacts).
- Reads discrete and word inputs directly from the module into the image register during the RLL program execution (IORW instruction).
- Writes discrete and word outputs from the image register to the module (IORW instruction).
- Copies power flow to both the image register and the I/O module (immediate coils).
- Immediate Set/Reset Coils – set or reset a specified bit immediately

Chapter 4

I/O Module Data Sheets

4.1 Module Selection Considerations

I/O Power Budget Requirements

The power supply (PPX:505–6660) provides up to 55 watts at +5VDC and 3.75 watts at –5 VDC for use by the controller and the I/O modules. The power supply operates at either 110V or 220V AC on user-supplied power. A jumper on the inside of the back of the module is used to select voltage. See Table 4-1.

Table 4-1 Series 505 I/O Modules, RBC, and SIMATIC 545 Controller

Model Number	Description	Immediate I/O	Power consumption (watts)	
			+5 V	–5V
PPX:505–4008	24 VAC Discrete Input (8 point)	√	2.0	–
PPX:505–4016	24 VAC Discrete Input (16 point)	√	2.0	–
PPX:505–4032	24 VAC Discrete Input (32 point)	√	2.0	–
PPX:505–4108	12 VDC Discrete Input (8 point)	√	2.0	–
PPX:505–4116	12 VDC Discrete Input (16 point)	√	2.0	–
PPX:505–4132	12 VDC Discrete Input (32 point)	√	2.0	–
PPX:505–4208	110 VAC Discrete Input (8 point)	√	2.0	–
PPX:505–4216	110 VAC Discrete Input (16 point)	√	2.0	–
PPX:505–4232	110 VAC Discrete Input (32 point)	√	2.0	–
PPX:505–4308	24 VDC Discrete Input (8 point)	√	2.0	–
PPX:505–4316	24 VDC Discrete Input (16 point)	√	2.0	–
PPX:505–4332	24 VDC Discrete Input (32 point)	√	2.0	–
PPX:505–4408	220 VAC Discrete Input (8 point)	√	2.0	–
PPX:505–4416	220 VAC Discrete Input (16 point)	√	2.0	–
PPX:505–4432	220 VAC Discrete Input (32 point)	√	2.0	–
PPX:505–4508	24 VDC Discrete Output (8 point, 0.5A)	√	2.5	–
PPX:505–4516	24 VDC Discrete Output (16 point, 0.5A)	√	2.5	–
PPX:505–4532	24 VDC Discrete Output (32 point, 0.5A)	√	2.5	–
PPX:505–4608	110 VAC Discrete Output (8 point, 0.5A)	√	2.5	–
PPX:505–4616	110 VAC Discrete Output (16 point, 0.5A)	√	2.5	–
PPX:505–4632	110 VAC Discrete Output (32 point, 0.5A)	√	2.5	–
PPX:505–4708	24 VDC Discrete Output (8 point, 2.0A)	√	5.0	–
PPX:505–4716	24 VDC Discrete Output (16 point, 2.0A)	√	5.0	–
PPX:505–4732	24 VDC Discrete Output (32 point, 2.0A)	√	5.0	–
PPX:505–4808	110/220 VAC Discrete Output (8 point, 2.0A)	√	5.0	–
PPX:505–4816	110/220 VAC Discrete Output (16 point, 2.0A)	√	5.0	–
PPX:505–4832	110/220 VAC Discrete Output (32 point, 2.0A)	√	5.0	–
PPX:505–4908	Relay Output Form C (8 point)	√	2.5	–
PPX:505–4916	Relay Output Form A (16 point)	√	2.5	–

Table 4–1 Series 505 I/O Modules, RBC, and SIMATIC 545 Controller (continued)

Model Number	Description	Immediate I/O	Power consumption (watts) +5 V	–5V
PPX:505–4932	Relay Output Form A (32 point)	√	2.5	–
PPX:505–5100	Turboplastic		7.0	–
PPX:505–5103	Turboparison		7.0	–
PPX:505–5417	Relay Output Form C	√	2.5	–
PPX:505–6010	Input Simulator	√	2.0	–
PPX:505–6011	Output Simulator	√	2.5	–
PPX:505–6208A	Analog Input (8 points)		4.0	–
PPX:505–6208	Analog Output (8 point)		5.0	–
PPX:505–6208A	Analog Output (8 points)		2.0	–
PPX:505–6308	Word Input		4.0	–
PPX:505–6408	Word Output		5.0	–
PPX:505–6660	110/220 VAC Redundant Power Supply		–	–
PPX:505–6660A	110/220 VAC Redundant Power Supply		–	–
PPX:505–6663	24VDC Redundant Power Supply		–	–
PPX:505–6851A	Remote Base Controller		5.0	0.2
PPX:505–7002	High-Speed Counter and Encoder		2.5	–
PPX:505–7012	Analog Eight input, Four output	√ *	2.5	0.3
PPX:505–7016	High-speed Bipolar analog (8 inputs/4 outputs)	√	5.0	0.125
PPX:505–7028	Thermocouple		2.5	0.01
PPX:505–7038	RTD		2.2	–
PPX:505–7101	BASIC		6.0	0.125
PPX:505–7190	505/7MT Interface		4.0	–
PPX:505–7339	Network Interface Module, Local Line		8.0	–
PPX:505–7340	Network Interface Module, RS–232–C		8.0	–
PPX:505–7354	Peerlink		8.0	–
PPX:505–9201	505 Smart Slice 24VDC (10 inputs/6 outputs)		–	–
PPX:505–9202	505 Smart Slice 220VAC (10 inputs/6 outputs)		–	–
PPX:545–1101	Programmable Logic Controller		4.0	0.2
PPX: 505/ATM–0220	Coprocessor		11.0	0.2
PPX: 505/ATM–0440	Coprocessor		11.0	0.2

* Only inputs should be used for immediate I/O.

Module Selection Considerations (continued)

Special Function Modules

Special Function modules, such as the Network Interface module, Peerlink, or BASIC module, can reside in any base. You can have up to 256 Special Function modules, but there is usually a large amount of data transferred between the controller and these special modules and optimum performance may be extremely difficult to achieve. If system performance is critical, place Special Function modules in the local I/O base to minimize any possible affects on scan time.

4.2 Series 505 I/O System Specifications

The Series 505 I/O subsystem is tested against some of the most stringent standards in the world. The module data sheets have specifications concerning power consumption, output or input points, etc. Table 4-2 lists specifications that are common to Series 505 I/O devices, unless otherwise noted on the individual module specifications.

Table 4-2 Environmental Specifications

Operating Temperature	0° to 60°C (32° to 140°F)
Storage Temperature	–40° to +70°C (–40° to 158°F)
Relative Humidity	5% to 95% non-condensing
Pollution Degree	2, IEC 664, 684A
Vibration Sinusoidal Random	IEC 68-2-6, Test Fc; 0.15 mm, peak-to-peak, 10–57 Hz; 2.0 g 57–150 Hz NAVMAT P-9492 or IEC 68-2-34 Test Fdc with 0.04 g^2/Hz, 80–350 Hz, and 3 dB/octave rolloff, 80–20 Hz and 350–2000 Hz at 10 min/axis
Impact Shock	IEC, 68-2-27, Test Ea; Half sine, 15g 11 ms
Electric Noise Immunity Conducted noise: Radiated noise: Electrostatic discharge:	 IEC 801, Part 4, Level 3 MIL STD 461B, Part 4; CS01, CS02, CS06 IEC 255-4 EEC 4517/79 Com(78) 766 Final, Part 4 IEEE 472, 2.5 kV IEC 801, Part 3, Level 3 MIL STD 461B, Part 4; RS01, RS02 IEC 801, Part 2, Level 4, (15 kV)
System Isolation	Isolation (user-side to controller-side): 1500 V rms
Corrosion Protection	All parts are of corrosion resistant material or are plated or painted as corrosion protection.
Agency Approvals	UL Listed (UL508 industrial control equipment) CSA Certified (CSA142 process control equip.) FM Approved (Class I, Div. 2, Haz. Location)

4.3 Remote Bases

Description

The controller is housed in a Series 505 base assembly. A base assembly has slots for the PLC or remote base controller (RBC), a power supply, and the I/O modules. Four base assembly models are currently available. Each has the following number of slots for I/O modules. See Figure 4-1.

- 16-slots (PPX:505–6516)
- 8-slots (PPX:505–6508
- 4-slots (PPX:505–6504)
- 11-slot Redundant Base (PPX:505–6511) (the base houses two power supplies and two CPUs)

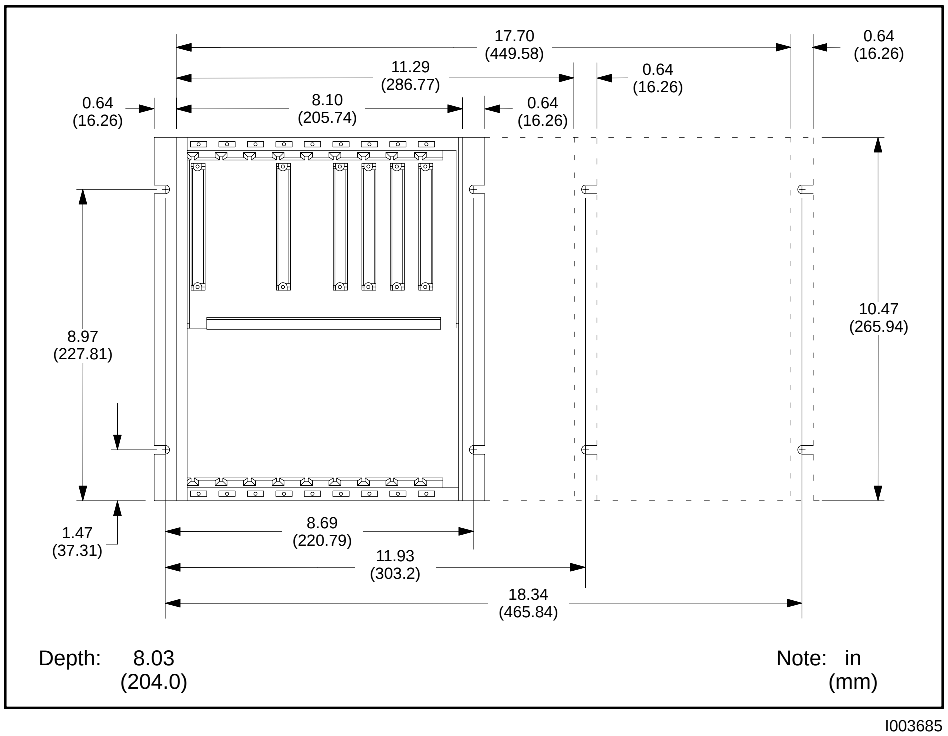


Figure 4-1 Screw-hole Dimensions for 4-, 8-, and 16-slot Bases

4.4 Power Supplies

There are three power supplies available in Series 505:

- 110/220 VAC (PPX:505–6660)
- 24 VDC (PPX:505–6663)
- 110/220 VAC Redundant Power Supply (PPX:505–6660A)

The PPX:505–6660 and PPX:505–6660A are the same except the PPX: 505–6660A can be used in a redundant power supply configuration; the complementary supply can be either a PPX:505–6663 or another PPX:505–6660A. The PPX:505–6660 cannot be used in a redundant architecture. (Use the PPX:505–6511 Redundant I/O base if dual power supplies are required.) A jumper on the inside of the back of the AC power supplies is used to select the voltage.

All three power supplies provide up to 55 watts at +5 VDC and 3.75 watts at –5 VDC for use by the controller and the I/O modules.

Series 505 power supplies provide up to 60 watts of power. When externally supplied power fails, a signal is sent to the controller to warn of the oncoming power failure. When line power is restored, the 505 system resumes normal operation.

When an overload occurs, the power supplies will automatically shut down. Once the overload is removed and power is restored, the power supply resumes normal operation.

The PPX:505–6660 and PPX:505–6660A power supply can be selected for either 110 VAC or 220 VAC.

The PPX:505–6663 is a 24 VDC power supply. The user-provided power source may be a battery/charger or other DC power supply in the range of 20-30 volts DC provided it meets the stated requirements for current and maximum ripple.

Table 4-3 PPX:505–6660, PPX:505–6660A Specifications

Input Voltage Select	110/220 VAC (User-selectable)
Voltage Range	85–132 VAC (110) or 170–264 VAC (220)
Frequency	47–63 Hz
Input Current Peak Inrush Steady state Volt-Amp rating	 50A maximum 2A rms maximum, 8 A zero to peak 200 V-A
Output	55 W at +5 VDC 3.75 W at –5 VDC
Fuse	3.0A/250 V, slow-blow, 3AG

Table 4-4 PPX:505–6663 Specifications

Voltage Range	20–30 VDC (24)
Input Voltage Ripple	< 10% (2.4V) from 94–126 Hz
Input Current Peak Inrush Steady state Wattage rating – Model PPX:505–6663	 20A maximum 5A DC maximum, 100 W maximum
Output	55 W at +5 VDC 3.75 W at –5 VDC
Fuse	8.0A/250 V, normal-blow, 3AG

4.5 Discrete Input Modules

The Series 505 Discrete Input modules provide 24 VAC, 110 VAC, 220 VAC, 6–12 VDC, and 24 VDC selections in 8-, 16-, and 32-point models. The input circuits are grouped into four commons per module. (Isolation is provided between each of the four commons.)

24 VAC Discrete Input

Model Number	Number of Inputs	I/O Slots
PPX:505–4008	8	1
PPX:505–4016	16	1
PPX:505–4032	32	1

Type of Inputs
IEC TYPE 1

Rated Voltage
24–48V AC

Input Voltage Range for “ON”
20.0V AC UH min
56.0V AC UH max

Input Voltage Range for “OFF”
0.0V AC UL min
5.0V AC UL max

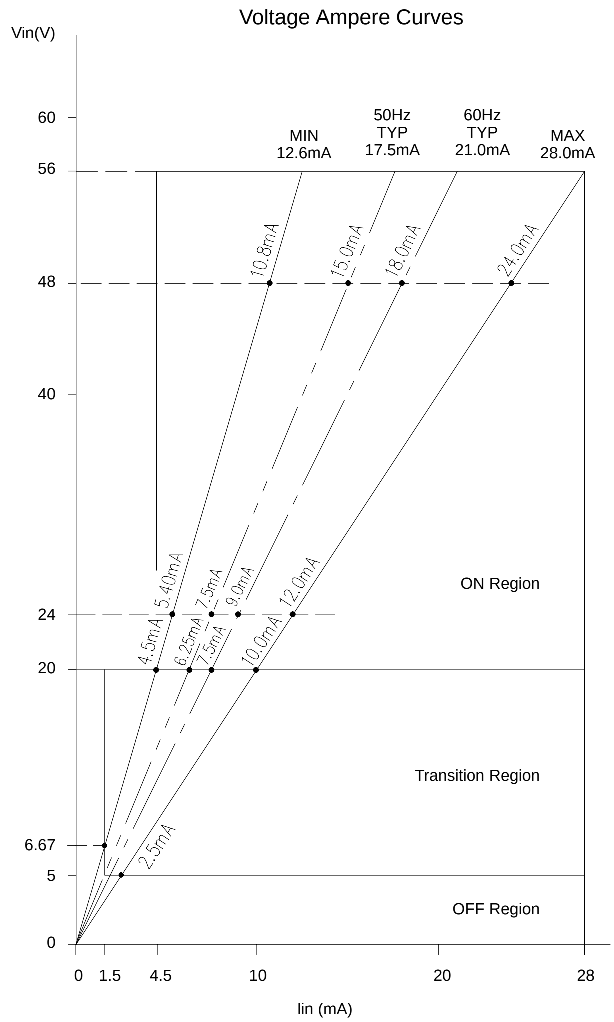
Input Current Limits during “ON”
4.5 mA IH min
28.0 mA IH max

Max Input Current for “OFF”
1.5 mA IT min

Delay Time through Module
06 ms min
30 ms max

Max Module Power from Base
2.0 watts

Operating Frequency
47 to 63 Hz



I000594

Discrete Input Modules (continued)

Model Number	Number of Inputs	I/O Slots
PPX:505–4108	8	1
PPX:505–4116	16	1
PPX:505–4132	32	1

Type of Inputs
IEC TYPE 1

Rated Voltage
6–12V DC

Input Voltage Range for “ON”
4.0V DC UH min
15.0V DC UH max

Input Voltage Range for “OFF”
0.0V DC UL min
1.5V DC UL max

Input Current Limits during “ON”
3.0 mA IH min
22.0 mA IH max

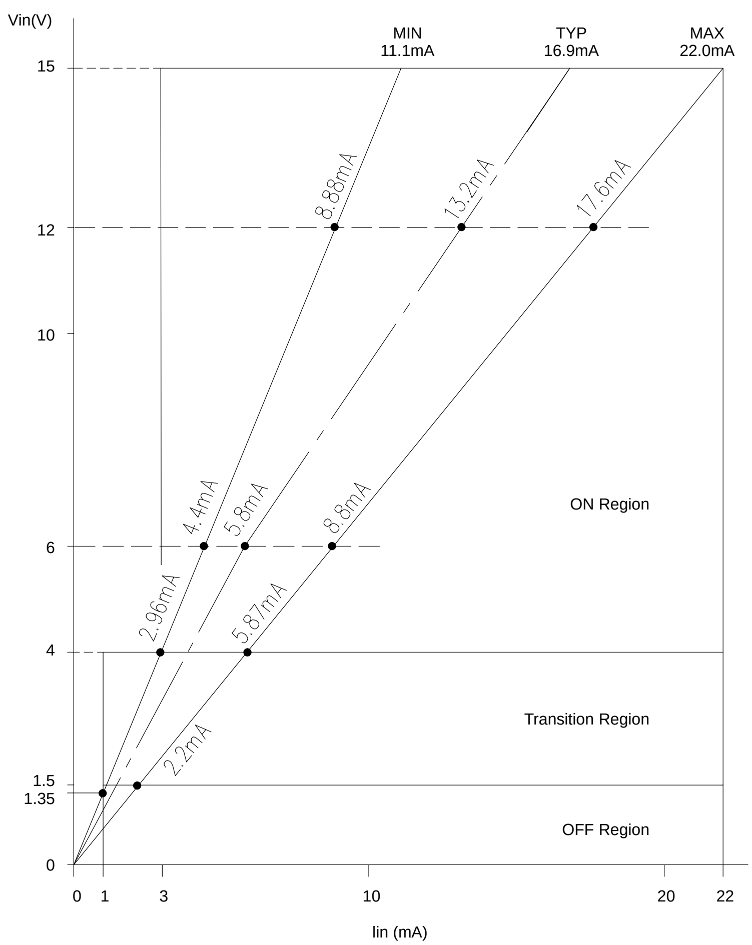
Max Input Current for “OFF”
1.0 mA IT min

Delay Time through Module
0.5 ms min
3.0 ms max

Max Module Power from Base
2.0 watts

6–12 VDC Discrete Input

Voltage Ampere Curves



I000595

Model Number	Number of Inputs	I/O Slots
PPX:505–4208	8	1
PPX:505–4216	16	1
PPX:505–4232	32	1

110 VAC Discrete Input

Type of Inputs
IEC TYPE 1

Rated Voltage
100–115V AC

Input Voltage Range for “ON”
79.0V AC UH min
132.0V AC UH max

Input Voltage Range for “OFF”
0.0V AC UL min
20.0V AC UL max

Input Current Limits during “ON”
4.0 mA IH min
15.0 mA IH max

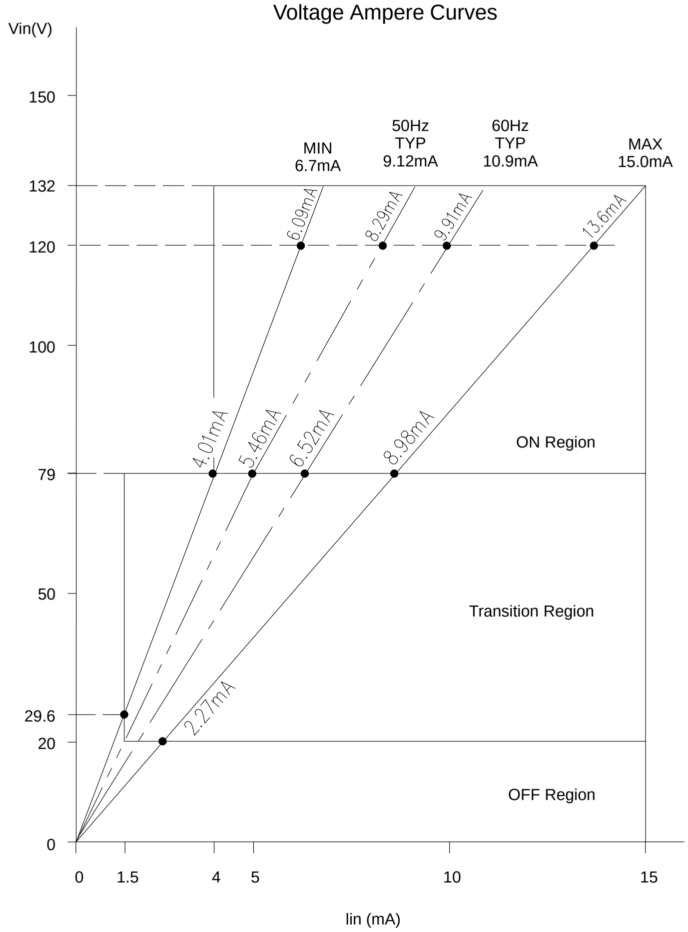
Max Input Current for “OFF”
1.5 mA IT min

Delay Time through Module
06 ms min
30 ms max

Max Module Power from Base
2.0 watts

Operating Frequency
47 to 63 Hz

Voltage Ampere Curves



I000596

Model Number	Number of Inputs	I/O Slots
PPX:505–4308	8	1
PPX:505–4316	16	1
PPX:505–4332	32	1

Type of Inputs
IEC TYPE 1

Rated Voltage
24V DC

Input Voltage Range for "ON"
14.0V DC UH min
30.0V DC UH max

Input Voltage Range for "OFF"
0.0V DC UL min
5.0V DC UL max

Input Current Limits during "ON"
2.0 mA IH min
15.0 mA IH max

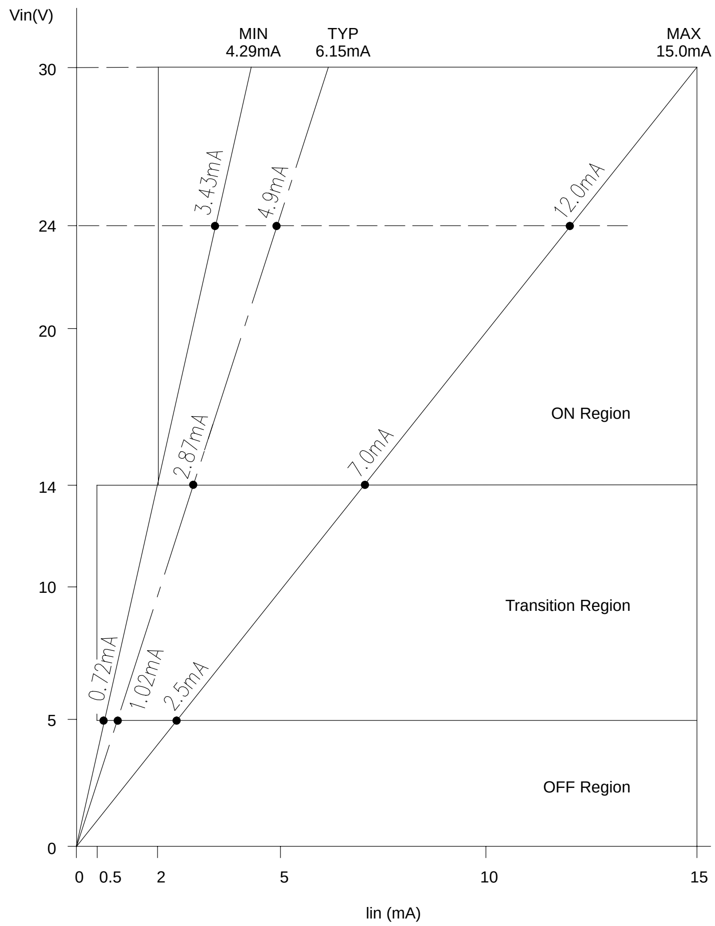
Max Input Current for "OFF"
0.5 mA IT min

Delay Time through Module
03 ms min
10 ms max

Max Module Power from Base
2.0 watts

24 VDC Discrete Input

Voltage Ampere Curves



I000597

220 VAC Discrete Input

Model Number	Number of Inputs	I/O Slots
PPX:505–4408	8	1
PPX:505–4416	16	1
PPX:505–4432	32	2

Type of Inputs
IEC TYPE 1

Rated Voltage
200–230V AC

Input Voltage Range for “ON”
164.0V AC UH min
265.0V AC UH max

Input Voltage Range for “OFF”
0.0V AC UL min
40.0V AC UL max

Input Current Limits during “ON”
6.0 mA IH min
20.0 mA IH max

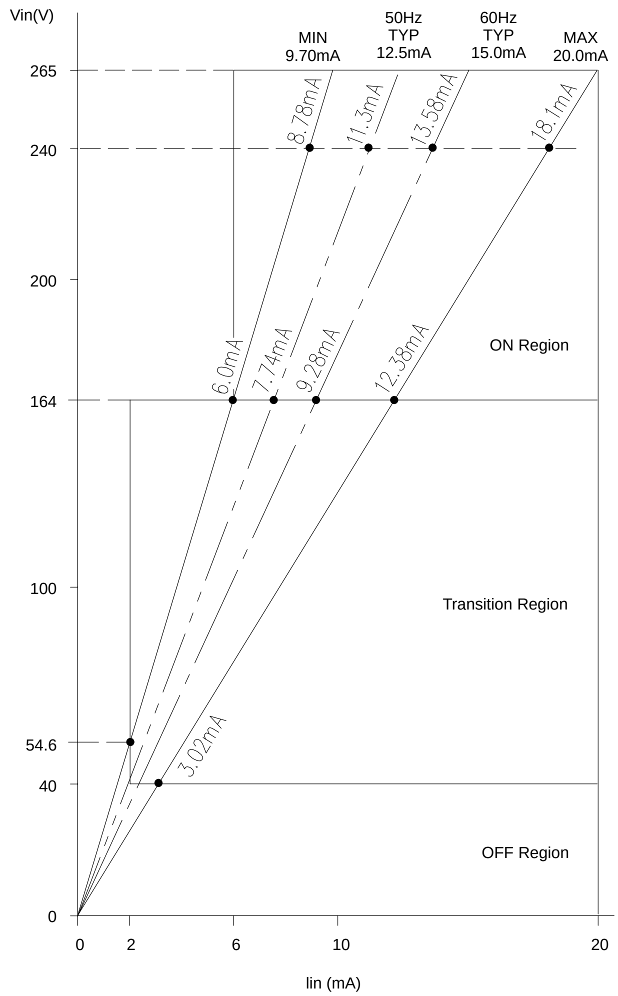
Max Input Current for “OFF”
2.0 mA IT min

Delay Time through Module
06 ms min
30 ms max

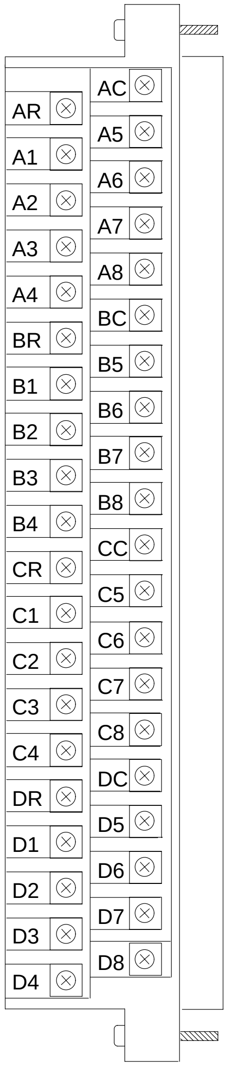
Max Module Power from Base
2.0 watts

Operating Frequency
47 to 63 Hz

Voltage Ampere Curves



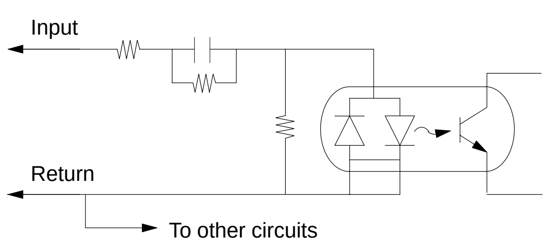
I000598



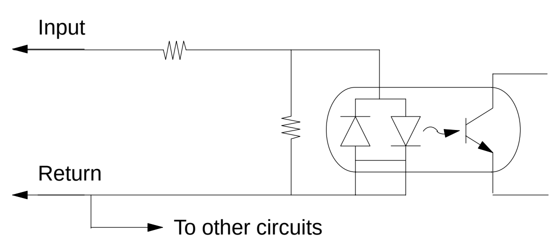
Terminal	8-point	16-point	32-point
AR	Return 1–2	Return 1–4	Return 1–8
A1	Input 1	Input 1	Input 1
A2	Input 2	Input 2	Input 2
A3	Not used	Input 3	Input 3
A4	Not used	Input 4	Input 4
AC	Not used	Not used	Not used
A5	Not used	Not used	Input 5
A6	Not used	Not used	Input 6
A7	Not used	Not used	Input 7
A8	Not used	Not used	Input 8
BR	Return 3–4	Return 5–8	Return 9–16
B1	Input 3	Input 5	Input 9
B2	Input 4	Input 6	Input 10
B3	Not used	Input 7	Input 11
B4	Not used	Input 8	Input 12
BC	Not used	Not used	Not used
B5	Not used	Not used	Input 13
B6	Not used	Not used	Input 14
B7	Not used	Not used	Input 15
B8	Not used	Not used	Input 16
CR	Return 5–6	Return 9–12	Return 17–24
C1	Input 5	Input 9	Input 17
C2	Input 6	Input 10	Input 18
C3	Not used	Input 11	Input 19
C4	Not used	Input 12	Input 20
CC	Not used	Not used	Not used
C5	Not used	Not used	Input 21
C6	Not used	Not used	Input 22
C7	Not used	Not used	Input 23
C8	Not used	Not used	Input 24
DR	Return 7–8	Return 13–16	Return 25–32
D1	Input 7	Input 13	Input 25
D2	Input 8	Input 14	Input 26
D3	Not used	Input 15	Input 27
D4	Not used	Input 16	Input 28
DC	Not used	Not used	Not used
D5	Not used	Not used	Input 29
D6	Not used	Not used	Input 30
D7	Not used	Not used	Input 31
D8	Not used	Not used	Input 32

Typical Input Circuits

PPX:505–4008, 4016, 4032
PPX:505–4208, 4216, 4232
PPX:505–4408, 4416, 4432



PPX:505–4108, 4116, 4132
PPX:505–4308, 4316, 4332



I000599

4.6 Discrete AC Output Modules

Description

The Series 505 Discrete AC Output modules provide 24–110 VAC (0.5A), and 110–220 VAC (2.0A) voltage selections in 8-, 16-, and 32-point models. The output circuits are grouped into four commons per module.

24–110 VAC, 0.5 A Discrete Output

Model Number	Number of Outputs	I/O Slots
PPX:505–4608	8	1
PPX:505–4616	16	1
PPX:505–4632	32	1

Rated Voltage
24 to 115V AC

Operating Voltage Range
20 to 132V AC

Min Load per Point
5.0 mA

Temporary Overload
5 A rms for 2 cycles

Max On–State Voltage Drop
1.8V, 0 to peak (@ 0.5 A)

Max Off–State Leakage Current
2.0 mA @ 40° C
6.0 mA @ 60° C

Kickback Protection
Diode

Max Delay Time through Module
11 ms ON to OFF
1.7 ms OFF to ON

Frequency Range
47 to 63 Hz

Max Module Power from Base
2.5 W

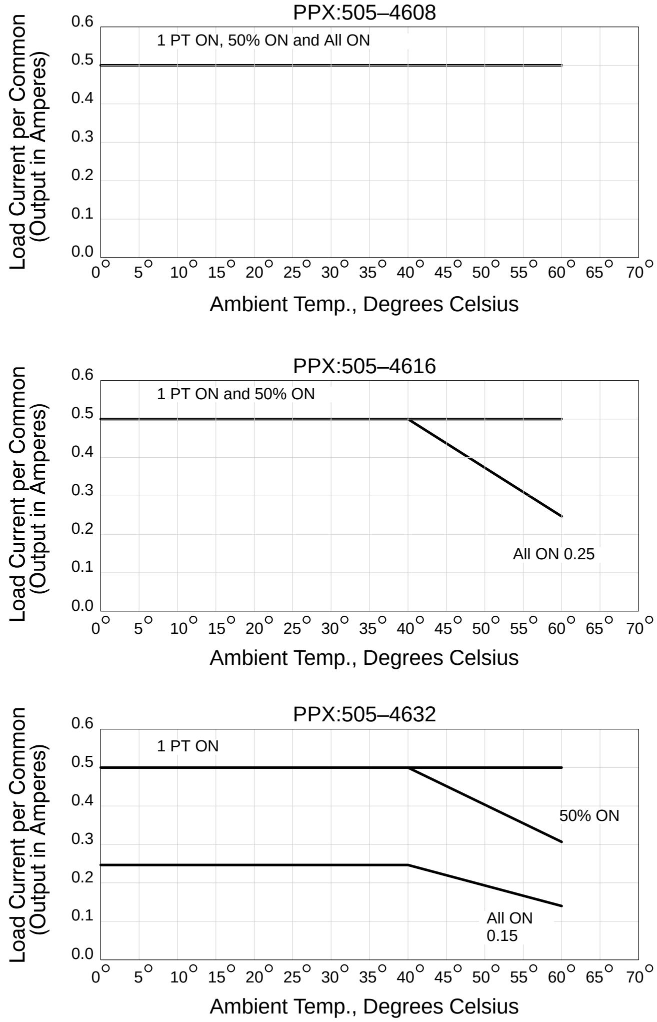
Surge Suppressor
R–C

Type of Outputs
Non–latching type, unprotected

Output Fuse Rating
3.15 A 125V
5 X 20mm, Normal Blow

dv/dt for Main Triac
100V/us

Derating Curves



I000600

Discrete AC Output Modules (continued)

Model Number	Number of Outputs	Module Width
PPX:505–4808	8	2
PPX:505–4816	16	2
PPX:505–4832	32	2

110–220 VAC, 2.0 A Discrete Output

Rated Voltage
100 to 230V AC

Operating Voltage Range
85 to 265V AC

Min Load per Point
50 mA

Temporary Overload
20 A rms for 2 cycles

Max On–State Voltage Drop
2.0V, 0 to peak (@ 2 A)

Max Off–State Leakage Current
6.0 mA

Max Delay Time through Module
11 ms ON to OFF
2 ms OFF to ON

Frequency Range
47 to 63 Hz

Max Module Power from Base
5 W

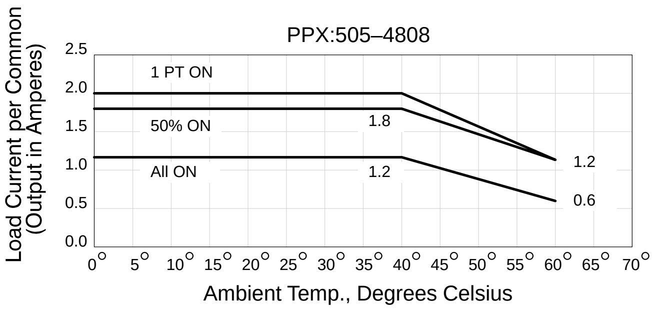
Surge Suppressor
R–C

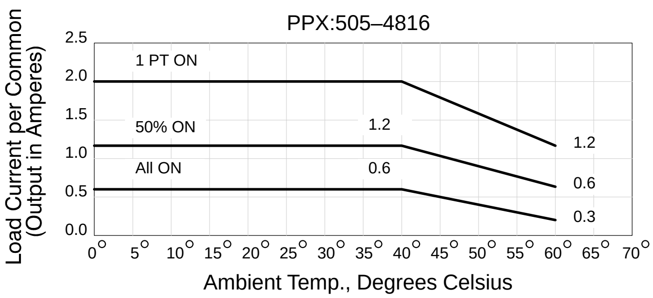
Type of Outputs
Non–latching type, unprotected

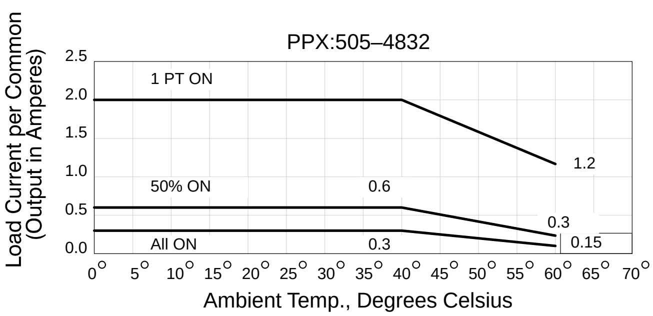
Output Fuse Rating
5.0 A 250V
.25 X 1.25 in., Normal Blow

dv/dt for Main Triac
100V/us

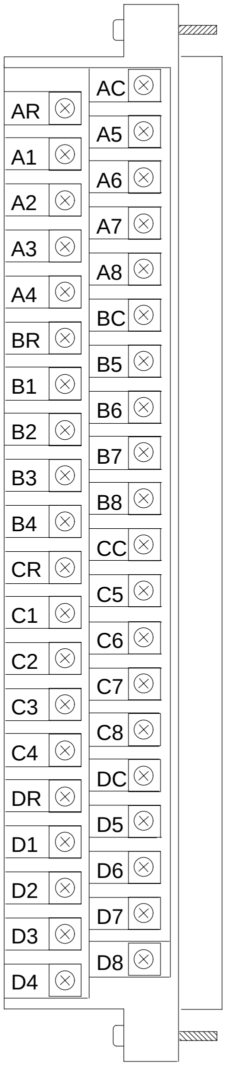
Derating Curves







I000601

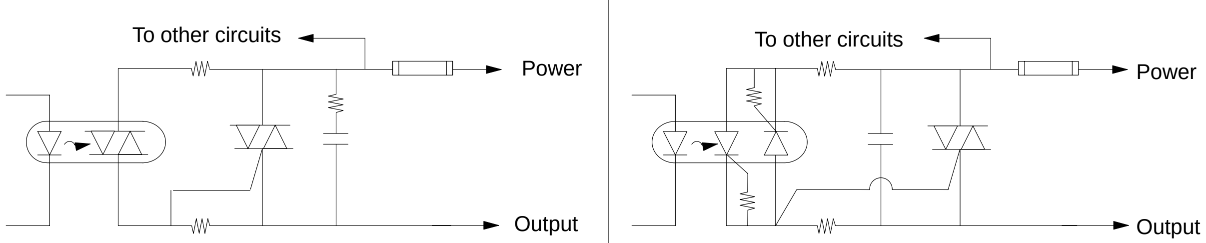


Terminal	8-point AC Out	16-point AC Out	32-point AC Out
AR	Not used	Not used	Not used
A1	Output 1	Output 1	Output 1
A2	Output 2	Output 2	Output 2
A3	Not used	Output 3	Output 3
A4	Not used	Output 4	Output 4
AC	User Power	User Power	User Power
A5	Not used	Not used	Output 5
A6	Not used	Not used	Output 6
A7	Not used	Not used	Output 7
A8	Not used	Not used	Output 8
BR	Not used	Not used	Not used
B1	Output 3	Output 5	Output 9
B2	Output 4	Output 6	Output 10
B3	Not used	Output 7	Output 11
B4	Not used	Output 8	Output 12
BC	User Power	User Power	User Power
B5	Not used	Not used	Output 13
B6	Not used	Not used	Output 14
B7	Not used	Not used	Output 15
B8	Not used	Not used	Output 16
CR	Not used	Not Used	Not used
C1	Output 5	Output 9	Output 17
C2	Output 6	Output 10	Output 18
C3	Not used	Output 11	Output 19
C4	Not used	Output 12	Output 20
CC	User Power	User Power	User Power
C5	Not used	Not used	Output 21
C6	Not used	Not used	Output 22
C7	Not used	Not used	Output 23
C8	Not used	Not used	Output 24
DR	Not used	Not used	Not used
D1	Output 7	Output 13	Output 25
D2	Output 8	Output 14	Output 26
D3	Not used	Output 15	Output 27
D4	Not used	Output 16	Output 28
DC	User Power	User power	User Power
D5	Not used	Not used	Output 29
D6	Not used	Not used	Output 30
D7	Not used	Not used	Output 31
D8	Not used	Not used	Output 32

Typical Output Circuit

PPX:505–4608, 4616, 4632

PPX:505–4808, 4816, 4832



I000602

4.7 Discrete DC Output Modules

Description

The Series 505 Discrete DC Output modules provide 6–24 VDC (0.5A), and 6–24 VDC (2.0A) voltage selections in 8-, 16-, and 32-point models. The output circuits are grouped into four commons per module. (Isolation is provided between each of the four commons.)

Model Number	Number of Outputs	I/O Slots
PPX:505–4508	8	1
PPX:505–4516	16	1
PPX:505–4532	32	1

Rated Voltage
6 to 24V DC

Operating Voltage Range
4.5 to 34V DC

Temporary Overload
2.0 A for 1 ms

Max On–State Voltage Drop
1.8V DC

Max Off–State Leakage Current
0.2 mA

Kickback Protection
Diode

Max Delay Time through Module
(with 5 mA min load)
1 ms On to Off
1 ms Off to On

User Power Current with No Load
3.0 mA/Common

Max Module Power from Base
2.5 W

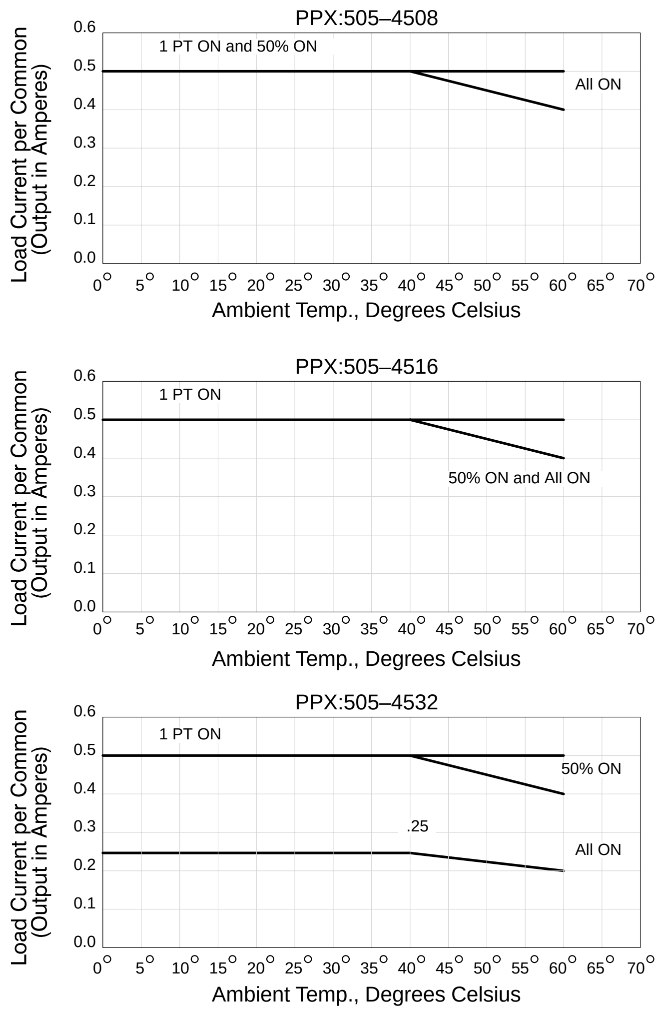
Reverse Voltage Protection at
Power Terminals
34V DC

Type of Outputs
Non–latching type, unprotected

Output Fuse Rating
3 A 125V
5 X 20mm, Normal Blow

24 VDC, 0.5 A Discrete Output

Derating Curves



I000603

Model Number	Number of Outputs	I/O Slots
PPX:505–4708	8	2
PPX:505–4716	16	2
PPX:505–4732	32	2

24 VDC, 2.0 A Discrete Output

Rated Voltage
6 to 24V DC

Operating Voltage Range
4.5 to 34V DC

Temporary Overload
5.0 A for 1 ms

Max On–State Voltage Drop
2.0V

Max Off–State Leakage Current
0.2 mA

Kickback Protection
Diode

Max Delay Time through Module
(with 5 mA min load)
1 ms ON to OFF
1 ms OFF to ON

User Power Current with No Load
3.0 mA/Common

Max Module Power from Base
5 W

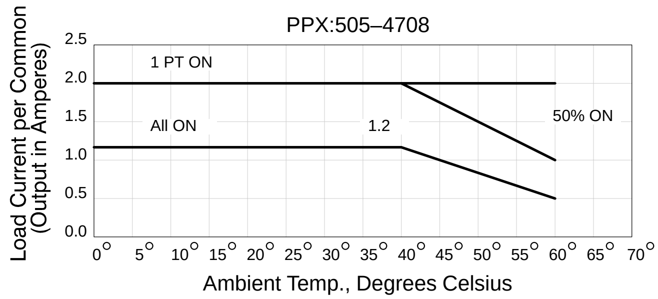
Reverse Voltage Protection at
Power Terminals
34V DC

Type of Outputs
Non-latching type, unprotected

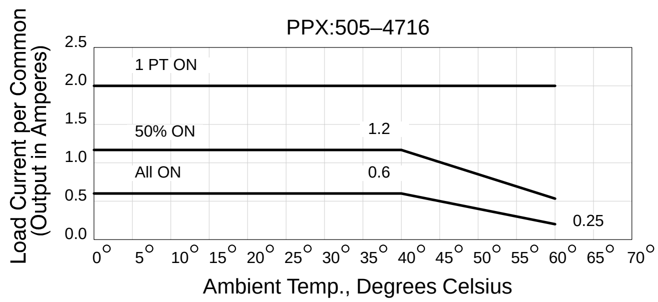
Output Fuse Rating
3.15 A 125V
5 X 20mm, Normal Blow

Derating Curves

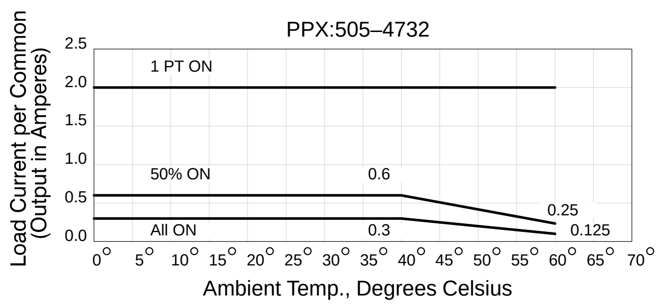
PPX:505–4708



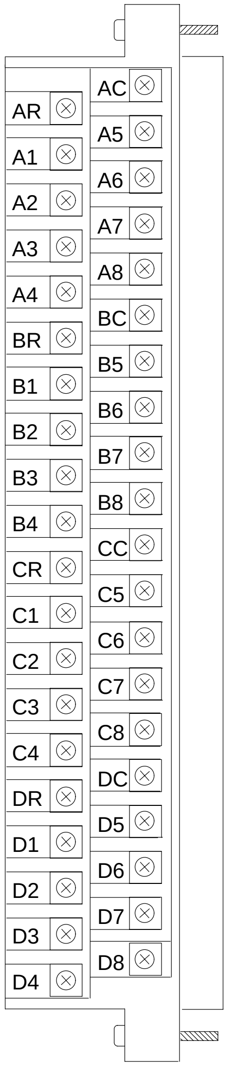
PPX:505–4716



PPX:505–4732



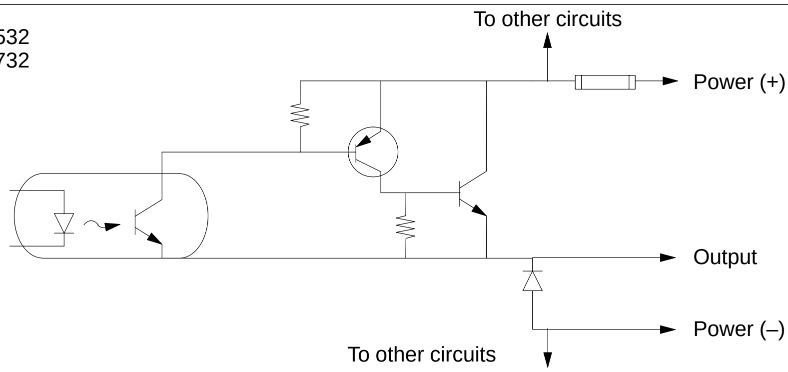
I000604



Terminal	8-point DC Out	16-point DC Out	32-point DC Out
AR	User Power (–)	User Power (–)	User Power (–)
A1	Output 1	Output 1	Output 1
A2	Output 2	Output 2	Output 2
A3	Not used	Output 3	Output 3
A4	Not used	Output 4	Output 4
AC	User Power (+)	User Power (+)	User Power (+)
A5	Not used	Not used	Output 5
A6	Not used	Not used	Output 6
A7	Not used	Not used	Output 7
A8	Not used	Not used	Output 8
BR	User Power (–)	User Power (–)	User Power (–)
B1	Output 3	Output 5	Output 9
B2	Output 4	Output 6	Output 10
B3	Not used	Output 7	Output 11
B4	Not used	Output 8	Output 12
BC	User Power (+)	User Power (+)	User Power (+)
B5	Not used	Not used	Output 13
B6	Not used	Not used	Output 14
B7	Not used	Not used	Output 15
B8	Not used	Not used	Output 16
CR	User Power (–)	User Power (–)	User Power (–)
C1	Output 5	Output 9	Output 17
C2	Output 6	Output 10	Output 18
C3	Not used	Output 11	Output 19
C4	Not used	Output 12	Output 20
CC	User Power (+)	User Power (+)	User Power +)
C5	Not used	Not used	Output 21
C6	Not used	Not used	Output 22
C7	Not used	Not used	Output 23
C8	Not used	Not used	Output 24
DR	User Power (–)	User Power (–)	User Power (–)
D1	Output 7	Output 13	Output 25
D2	Output 8	Output 14	Output 26
D3	Not used	Output 15	Output 27
D4	Not used	Output 16	Output 28
DC	User Power (+)	User power (+)	User Power +)
D5	Not used	Not used	Output 29
D6	Not used	Not used	Output 30
D7	Not used	Not used	Output 31
D8	Not used	Not used	Output 32

Typical Output Circuit

PPX:505–4508, 4516, 4532
PPX:505–4708, 4716, 4732



I000605

4.8 Relay Output Modules

Description

The Series 505 Relay Output modules can switch either DC or AC power at each common.

Model Number	Number of Outputs	I/O Slots
PPX:505–4908	8	1
PPX:505–4916	16	1
PPX:505–4932	32	2
PPX:505–5417	16	1

Operating Voltage Range
- 20 to 265V AC
- 4.5 to 30V DC
- 10 to 125V AC (Model 5417)
- 0 to 120V DC (Model 5417)

Temporary Overload
- 5 A for 1ms
- 2A, 1sec (Model 5417)

Off-State Leakage Current
- 0.0 mA

Total Relay Contact Resistance
- 250 milliohms (Models 4916, 4932)
- 300 milliohms (Model 4908)
- 50 milliohms (Model 5417)

Max Delay Time through Module
- 10 ms ON to OFF
- 10 ms OFF to ON
- 3 ms ON to OFF (Model 5417)
- 5 ms OFF to ON (Model 5417)

Max Module Power from Base
- 2.5 W
- 3 W (Model 5417)

Output Fuse
- (1 fuse per point)
- Rating: 3 A, 250V (Model 4908)
- Rating: 4A, 250V (Model 5417)
- Size: 1.25” X .25”, Normal Blow (Models 4908, 5417)

Type of Contact
- Form–A (Models 4916, 4932)
- Form–C (Models 4908, 5417)

Type of Outputs
- Non–latching type, Unprotected

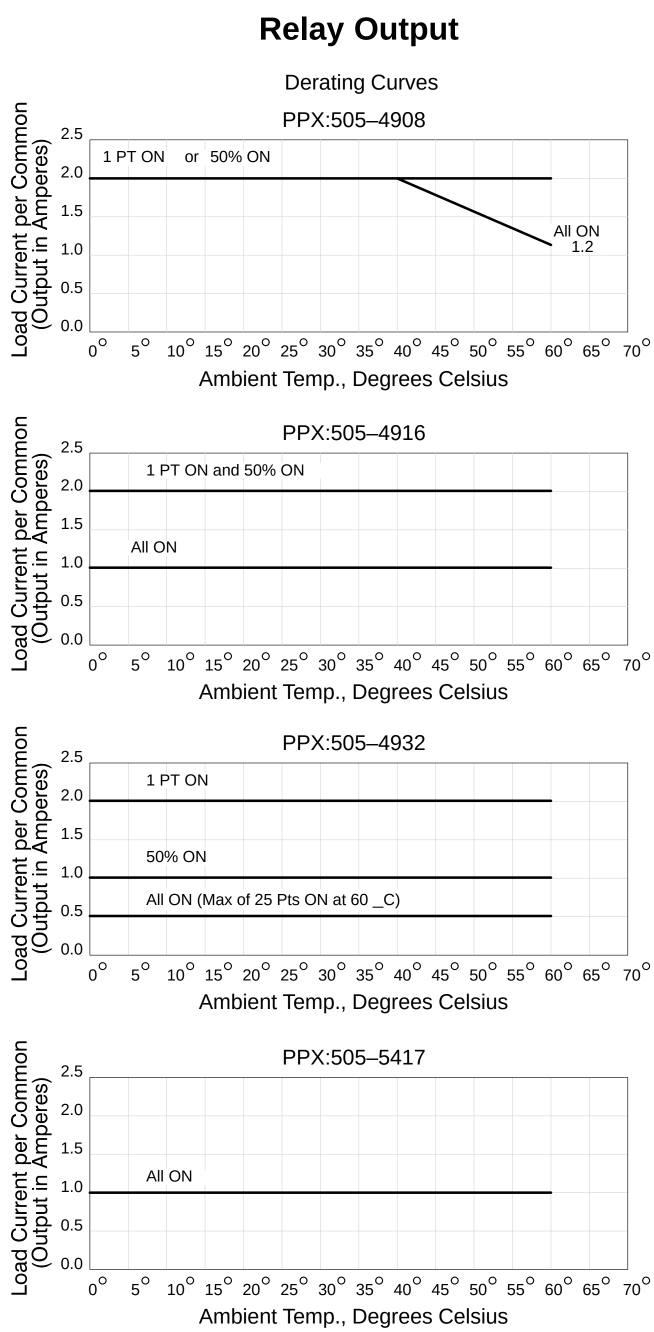
Repetition Rate
- 6 Hz max

User Power Supply,
- 20–28 VDC, 0.5A, UL class 2 (Models 4908, 4916, 4932)
  - Current: 8 pt = 100 mA
  - 16 pt = 200 mA
  - 32 pt = 400 mA

21.6–26.4 VDC, 300mA (Model 5417)

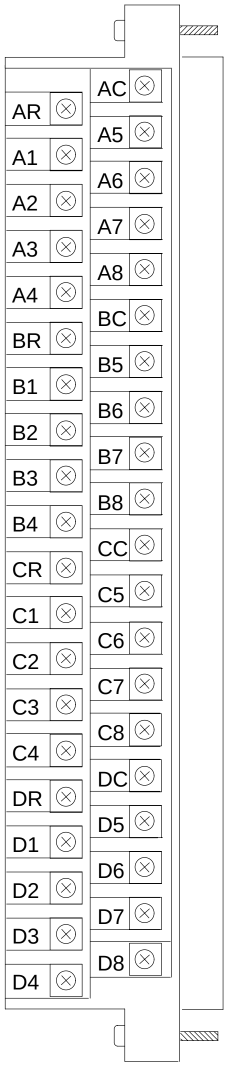
Life Cycles @ 1 Hz repetition rate (Model 4908, 4916, 4932)

Full Rated Current	100,000
@ 0.5 A Models 4916, 4932	300,000
@ 0.5 A Model 4908	500,000
@ 0.1 A Model 4908, 4916, 4932	1,000,000



I000606

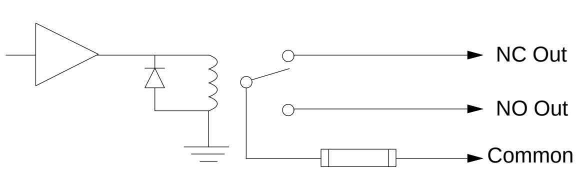
Relay Output Modules (continued)



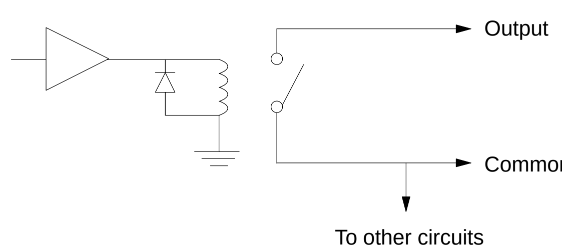
	PPX:505–4908	PPX:505–4916	PPX:505–4932	PPX:505–5417
Terminal	**8-point Relay**	**16-point Relay**	**32-point Relay**	**16–point Relay**
AR	Common 1	Common for 1–4	Common for 1–8	User coil– (gnd)
A1	NO 1 (LED A1)	NO 1 (LED A1)	NO 1 (LED A1)	NO 1 (LED A1)
A2	NC 1 (LED A1)	NO 2 (LED A2)	NO 2 (LED A2)	NO 2 (LED A2)
A3	Not used	NO 3 (LED A3)	NO 3 (LED A3)	NO 3 (LED A3)
A4	Not used	NO 4 (LED A4)	NO 4 (LED A4)	NO 4 (LED A4)
AC	Common 2	Common for 5–8	Common for 9–16	Common 1–4
A5	NO 2 (LED A2)	NO 5 (LED B1)	NO 9 (LED B1)	NC 1 (LED A5)
A6	NC 2 (LED A2)	NO 6 (LED B2)	NO 10 (LED B2)	NC 2 (LED A6)
A7	Not used	NO 7 (LED B3)	NO 11 (LED B3)	NC 3 (LED A7)
A8	Not used	NO 8 (LED B4)	NO 12 (LED B4)	NC 4 (LED A8)
BR	Common 3	Not used	NO 5 (LED A5)	User coil+ (24VDC)
B1	NO 3 (LED B1)	Not used	NO 6 (LED A6)	NO 5 (LED B1)
B2	NC 3 (LED B1)	Not used	NO 7 (LED A7)	NO 6 (LED B2)
B3	Not used	Not used	NO 8 (LED A8)	NO 7 (LED B3)
B4	Not used	NO 9 (LED C1)	NO 17 (LED C1)	NO 8 (LED B4)
BC	Common 4	Not used	NO 13 (LED B5)	Common 5–8
B5	NO 4 (LED B2)	Not used	NO 14 (LED B6)	NC 5 (LED B5)
B6	NC 4 (LED B2)	Not used	NO 15 (LED B7)	NC 6 (LED B6)
B7	Not used	Not used	NO 16 (LED B8)	NC 7 (LED B7)
B8	Not used	NO 13 (LED D1)	NO 25 (LED D1)	NC 8 (LED B8)
CR	NO 5 (LED C1)	NO 10 (LED C2)	NO 18 (LED C2)	Not Used
C1	NC 5 (LED C1)	NO 11 (LED C3)	NO 19 (LED C3)	NO 9 (LED C1)
C2	Common 5	NO 12 (LED C4)	NO 20 (LED C4)	NO 10 (LED C2)
C3	Not used	Not used	NO 21 (LED C5)	NO 11 (LED C3)
C4	Not used	Not used	NO 22 (LED C6)	NO 12 (LED C4)
CC	NO 6 (LED C2)	NO 14 (LED D2)	NO 26 (LED D2)	Common 9–12
C5	NC 6 (LED C2)	NO 15 (LED D3)	NO 27 (LED D3)	NC 9 (LED C5)
C6	Common 6	NO 16 (LED D4)	NO 28 (LED D4)	NC 10 (LED C6)
C7	Not used	Not used	NO 29 (LED D5)	NC 11 (LED C7)
C8	Not used	Not used	NO 30 (LED D6)	NC 12 (LED C8)
DR	NO 7 (LED D1)	Not used	NO 23 (LED C7)	Not Used
D1	NC 7 (LED D1)	Not used	NO 24 (LED C8)	NO 13 (LED D1)
D2	Common 7	Common for 9–12	Common for 17–24	NO 14 (LED D2)
D3	Not used	Not used	Not used	NO 15 (LED D3)
D4	User Coil Power (+)	User Coil Power (+)	User Coil Power (+)	NO 16 (LED D4)
DC	NO 8 (LED D2)	Not used	NO 31 (LED D7)	Common 13–16
D5	NC 8 (LED D2)	Not used	NO 32 LED D8)	NC 13 (LED D5)
D6	Common 8	Common for 13–16	Common for 25–32	NC 14 (LED D6)
D7	Not used	Not used	Not used	NC 15 (LED D7)
D8	User Coil Power (–)	User Coil Power (–)	User Coil Power (–)	NC 16 (LED D8)

Typical Output Circuit

PPX:505–4908, 5417



PPX:505–4916, 4932



I003686

4.9 Analog Input Modules

Description

The Series 505 Analog Input module provides an interface with voltage or current sensing devices to monitor pressure, position, flow, temperature, or speed. This module translates an analog input into a digital word, which is used in the controller.

Analog Input

Model Number	Number of Inputs	I/O Slots
PPX:505–6108A	8	1

Signal Range
±5VDC/0 to 20 mA or ±10VDC

Accuracy (25_ C)
voltage, +/– 0.5% of full scale at 25° C
current, +/– 0.7% of full scale at 25° C

Temperature Coefficient
Voltage input, 58 ppm/°C
Current input, 83 ppm/°C

Repeatability
+/– 0.1% of full scale over operating range

Input Resolution
12 bits, 1.25 mV, 5 µA (±5V range)

Input System Conversion
Delay = Update + filter delay (75 ms)
250 ms + 75 ms = 325 ms typical

DC Input Resistance
1 megohm minimum on ±5V range
500K megohm on ±10V range

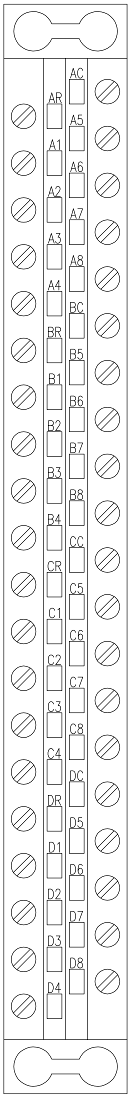
Input Protection (clamp diodes, optic isolation)
Voltage: overvoltage to +/– 30 VDC
Current: overcurrent to 30 mA

Module Power from Base
4 W max, 2.5W typical

Channel-to-channel Crosstalk Threshold
Up to 30 VDC

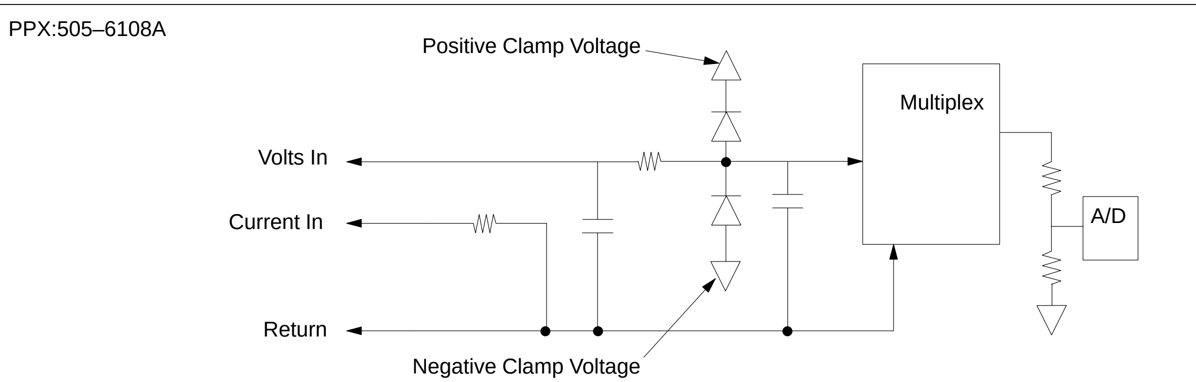
Input signal wiring
shielded, twisted-pair cable (14–24 AWG or 0.18–1.5 mm^2, either stranded or solid-type)

I000608



Terminal	8-channel Input
AR	Not Used
A1	I1 current input point 1
A2	V1 voltage input point 1
A3	Return for point 1
A4	Not used
AC	Not Used
A5	I5 current input point 5
A6	V5 voltage input point 5
A7	Return for point 5
A8	Not used
BR	Not used
B1	I2 current input point 2
B2	V2 voltage input point 2
B3	Return for point 2
B4	Not used
BC	Not used
B5	I6 current input point 6
B6	V6 voltage input point 6
B7	Return for point 6
B8	Not used
CR	Not used
C1	I3 current input point 3
C2	V3 voltage input point 3
C3	Return for point 3
C4	Not used
CC	Not used
C5	I7 current input point 7
C6	V7 voltage input point 7
C7	Return for point 7
C8	Not used
DR	Not used
D1	I4 current input point 4
D2	V4 voltage input point 4
D3	Return for point 4
D4	Chassis
DC	Not used
D5	I8 current input point 8
D6	V8 voltage input point 8
D7	Return for point 8
D8	Chassis

Typical Input Circuit



I000609

4.10 Analog Output Modules

Description

The Series 505 Analog Output module provides an interface with voltage or current sensing devices to control pressure, position, flow, temperature, or speed. This module translates a digital word into an analog output signal.

Analog Output

Model Number	Number of Outputs	I/O Slots
PPX:505–6208A	8	1

Signal range
0 to 10 VDC and 0 to 20 mA simultaneously

Accuracy (25_ C)
voltage, +/– 0.5% of full scale at 25° C
current, +/–0.5% of full scale at 25° C

Temperature coefficient
Voltage output, 136 ppm/° C
Current output, 204 ppm/° C

Resolution(output)
12 bits, 2.5 mV, 5 µA

Update time
29 ms (min)
PC scan time + 56 ms (max)

Settling time
Current: 2.0 ms (max)
Voltage: 0.2 ms (max)

Load resistance
Voltage: 5000 ohms min; no max
Current: 10 ohms min; 600 ohms max
(600 ohms min., 1000 ohms max. if extra
10 VDC power supply present in circuit)

Load
capacitive (voltage outputs)100 pF max
inductive (current outputs) 1.0 mH max

Maximum module power from base
2W max, 1W typical

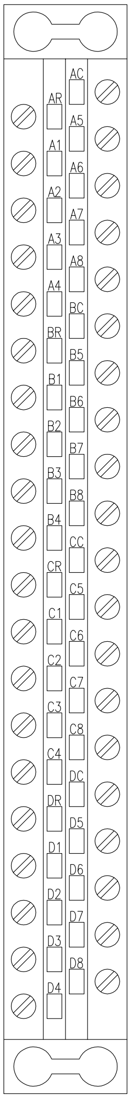
User power supply
20 to 28 VDC at 0.5 A, UL Class 2

Output Fuse Rating
0.5 A 250V, 5 X 20mm, Fast Acting

Voltage protection
overvoltage to 30 VDC, Reverse voltage protected

Output signal wiring
shielded, twisted-pair cable
(14-24 AWG or 0.18-1.5 mm@,
either stranded or solid-type).

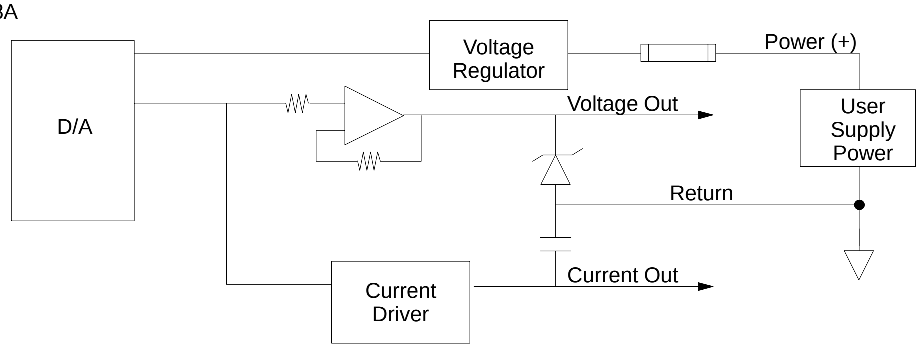
I000610



Terminal	8-channel Out
AR	User Power (–)
A1	I1 current output 1
A2	V1 voltage output 1
A3	Return for point 1
A4	Not used
AC	User Power (+)
A5	I5 current output 5
A6	V5 voltage output 5
A7	Return for point 5
A8	Not used
BR	Not used
B1	I2 current output 2
B2	V2 voltage output 2
B3	Return for point 2
B4	Not used
BC	Not used
B5	I6 current output 6
B6	V6 voltage output 6
B7	Return for point 6
B8	Not used
CR	Not used
C1	I3 current output 3
C2	V3 voltage output 3
C3	Return for point 3
C4	Not used
CC	Not used
C5	I7 current output 7
C6	V7 voltage output 7
C7	Return for point 7
C8	Not used
DR	Not used
D1	I4 current output 4
D2	V4 voltage output 4
D3	Return for point 4
D4	Shield
DC	Not used
D5	I8 current output 8
D6	V8 voltage output 8
D7	Return for point 8
D8	Shield

Typical Output Circuit

PPX:505–6208A



I000611

4.11 8-channel Input, 4-channel Output Analog Module

Description

The Series 505 8-channel input, 4-channel output module provides both an input and output interface with voltage or current sensing devices to monitor and control pressure, position, flow, temperature, or speed.

Model Number	Number of Inputs/ Outputs	I/O Slots
PPX:505–7012	8 / 4	1

Analog Input and Output

Module Features

Isolation
1500 Vrms channel-to-base, output-to-output, output-to-PLC, output-to-input

Accuracy
0.1% typical

PLC data can be scaled

Calibration
Lockout prevents unauthorized calibration
Calibrate using dumb terminal through RS–232–C port on module

Signal wiring
shielded, twisted-pair cable (14-24 AWG or 0.18-1.5 mm@, either stranded or solid-type).

Maximum module power from base
2.5W (+5V), 0.3W (–5V)

Output Section Features

User power supply
18 to 35 VAC or VDC (four supplies required for isolation between outputs)

Voltage and Current Outputs (Unipolar)
0 – 10 V, 0 – 20 mA

Resolution
12 bits

Parameter	Voltage	Current
Update Time	25 ms max.	25 ms max.
Settling Time	10 ms max.	2 ms max.
Load Resistance	5 K min.	900 max @ 24VDC

Module holds last value or 0 mA on communication failure

Outputs can withstand driving open and shorted conditions

Input Section Features

Impedance Isolated Differential Inputs

Voltage and Current Inputs (Bipolar)
ranges of 50 mV, 100 mV, 500 mV, 1 V, 2 V, 5 V, and 10 V

Resolution
15 bits (1 in 32000)

Repeatability
3 counts

Over current protection
1/2 watt resistor, allows continuous 32 mA on single channel

Update time
20 ms per input at 60 and 300 Hz
24 ms per input at 10 and 50 Hz

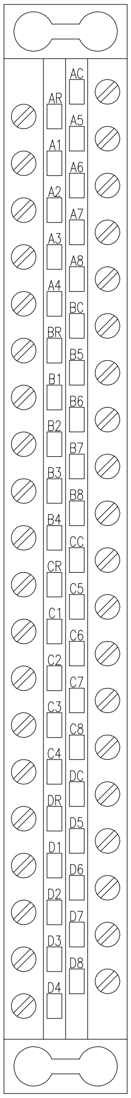
Noise rejection
Common mode, 80 dB @ 60Hz
Normal mode, 40 dB @ 60Hz

Software filtering
10Hz, 50Hz, and 60Hz selectable

Low-pass filter
10Hz hardware filter

Zero variable to +/– 2 V of offset in 1 mV increments

I000612



Terminal	Connection
AR	Input 1 (–)
A1	Input 2 (–)
A2	Input 3 (–)
A3	Input 4 (–)
A4	Input 5 (–)
AC	Input 1 (+)
A5	Input 2 (+)
A6	Input 3 (+)
A7	Input 4 (+)
A8	Input 5 (+)
BR	Input 6 (–)
B1	Input 7 (–)
B2	Input 8 (–)
B3	User Power A (–)
B4	Shield
BC	Input 6 (+)
B5	Input 7 (+)
B6	Input 8 (+)
B7	User Power A (+)
B8	Output A Voltage (+)
CR	Return A
C1	User Power B (–)
C2	Shield
C3	Return B
C4	User Power C (–)
CC	Output A Current (+)
C5	User Power B (+)
C6	Output B Voltage (+)
C7	Output B Current (+)
C8	User Power C (+)
DR	Shield
D1	Return C
D2	User Power D (–)
D3	Shield
D4	Return D
DC	Output C Voltage (+)
D5	Output C Current (+)
D6	User Power D (+)
D7	Output D Voltage (+)
D8	Output D Current (+)

I000613

4.12 High-speed Bipolar Analog Module (preliminary)

The Series 505 High Speed Analog provides 8 high-speed, bipolar (5 or 10 VDC) inputs, and 4 bipolar (5 or 10 VDC) or unipolar (4 to 20 mA) outputs. It provides an interface to voltage and current sensing devices to monitor and control pressure, position, flow, and other high-speed analog applications.

Table 4-5 PPX:505–7016 Input Specifications

Item	Description
Input points	8 differential bipolar analog points
Input signal	Current 0-20mA; voltages +/–5V and +/–10V 250Ω 0.5W internal resistor for current input
User power	None required, but loop power is not provided by this module
Isolation	1500V rms inputs-to-controller 1MΩ inputs-to-inputs
Accuracy, overall	≤0.1% of full-scale typical ≤0.3% of full-scale over full operating temperature range
Resolution	14 bits (unipolar), 13 bits plus sign (bipolar)
Repeatability	220 ppm (0.02%) at calibrated temperature
Overcurrent protection	0.5W resistor allows continuous 32mA on a signal point
Update time	1ms or 10ms for all points (inputs and outputs)
Common-mode rejection	100V rms with <0.1% additional inaccuracy
Common-mode rejection ratio	>80dB
Normal-mode rejection ratio	40 dB at 60 Hz
Filtering	1ms or 10ms
Hardware filtering	4K Hz
Overvoltage withstand	200V peak, differential or common-mode
Out-of-range inputs	Reported as error words
Calibration	ASCII terminal, Analog 0 compensation, Autoranging span selection during calibration: 50m V, 100m V, 500m V, 1V, 2V, 5V, 10V
Scaling	Data can be scaled to engineering units

High-speed Bipolar Analog Module (preliminary) (continued)

Table 4-6 PPX:505–7016 Output Specifications

Item	Description
Output points	Four differential, bipolar analog points
Output signal	Current 0-20mA; and voltages +/–5V, +/–10V, both available on separate terminals with a common return
Accuracy	≤0.1% of full-scale typical ≤0.25% of full-scale over full operating temperature range
Isolation	Fully isolated when each has separate isolated power supply 1500V rms output-to-controller
Resolution	12-bits (unipolar), 12 bits plus sign (bipolar)
Overvoltage withstand	Driving open and shorted conditions, on either or both output types per point.
Output value on failure	Selectable to 0V/mA or last value
Scale	Data can be scaled to engineering units
Update time	All outputs are updated 1 ms or 10ms
Output or error word	Read back in high-density configuration
Output loads	Volt > 5kΩ, Current <600Ω*, Capacitance <0.1µF
Compliance voltage	User supply –(9V)max
* Current up to 600Ω with a 24V nominal power supply. For increased load drive: increase power supply voltage (up to 40V) such that V power supply	

4.13 Thermocouple Module

Description

The Series 505 Thermocouple module has 8 high-accuracy inputs for thermocouple sensors or millivolt sources. The module senses small voltages generated by the thermocouple sensors and converts them into a 16-bit signed integer value. This value is represented in the controller as a process variable or monitoring point.

The 8 input channels are configured as a group via dipswitch settings. Calibration is accomplished using pushbuttons and an LED display.

Thermocouple

Model Number	Number of Channels	I/O Slots
PPX:505–7028	8	1

Thermocouple Input Types
J, K, T, E, mV, R, S, N

Input Range
+/– 50 mV, all channels

Channel–Channel Isolation
300 V (rms)

Input Impedance
>10 M

Rejection
Common mode, 120 dB @ 60Hz
Normal mode, 60 dB @ 60Hz

Resolution
14 bits minimum (1 in 16000)

Repeatability
10 counts or 0.06%

Accuracy
0.1%, +/– 0.5 C typical

Update time
250 ms, 1 – 8 channels

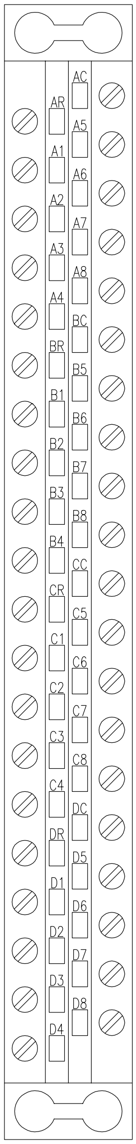
Cold junction compensation
automatic on all readings, calibration requires precision mV source only, no ice bath required

Report format
signed integer (mV), degrees C x10, degrees F x 10

Averaging
selectable (last four readings)

Power Required from Base
2.5 W

I000614



Terminal	Connections
AR	Ground (CJCAL –)
A1	Not used
A2	Input 1 (–)
A3	Not used
A4	Input 2 (–)
AC	CJCAL (+)
A5	Not used
A6	Input 1 (+)
A7	Not used
A8	Input 2 (+)
BR	Not used
B1	Input 3 (–)
B2	Not used
B3	Input 4 (–)
B4	Not used
BC	Not used
B5	Input 3 (+)
B6	Not used
B7	Input 4 (+)
B8	Not used
CR	Input 5 (–)
C1	Not used
C2	Input 6 (–)
C3	Not used
C4	Input 7 (–)
CC	Input 5 (+)
C5	Not used
C6	Input 6 (+)
C7	Not used
C8	Input 7 (+)
DR	Not used
D1	Input 8 (–)
D2	Not used
D3	Not used
D4	Chassis ground
DC	Not used
D5	Input 8 (+)
D6	Not used
D7	Not used
D8	Chassis ground

I000615

4.14 RTD Module

The RTD Module (PPX:505–7038) provides 8 RTD inputs in a single-wide module. It is compatible with 2-, 3-, and 4-wire RTDs and automatically compensates for lead wire resistance with 3-wire and 4-wire RTDs. The RTD module is dipswitch selectable for 4 types of platinum RTDs, 2 types of nickel probes, or copper. There are 8 different zero degree ohm configurations. The stimulation current (1mA or 2.5mA) is generated from the base power; no external power source is required. The RTD module automatically detects open, shorted, or out-of-range inputs and reports these to the controller as either upscale error codes or separate error words (user's choice). RTD measurements are reported as 0°F, 0°C, Ω, or scaled integer.

Table 4-7 RTD Features

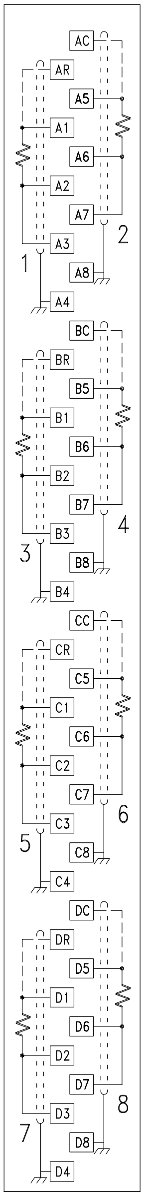
Model Number	PPX:505–7038
Inputs	8
Form factor	Single-wide
RTD compatibility	Platinum (4 types); copper; nickel (2 types); 2-, 3-, or 4-wire RTDs
RTD zero degree ohms	Configurable for eight different values
Input data format to the controller Refer to Table 1-2 for probe ranges	°F x 10 °C x 10 Ω x 10 (1.0 to 2000.0) Ω x 100 (1.00 to 320.00) Scaled counts, 0 to 32000
External power	None required
Error reporting	Configurable for separate error words or upscale error codes Module automatically detects open, shorted, or out-of-range inputs
Input noise filtering	Configurable for 50/60 Hz rejection
Internal resolution	0.003 Ω (19-bit)
Lead wire compensation	500 Ω max (<50 Ω recommended)

Table 4-8 Probe Temperature Ranges

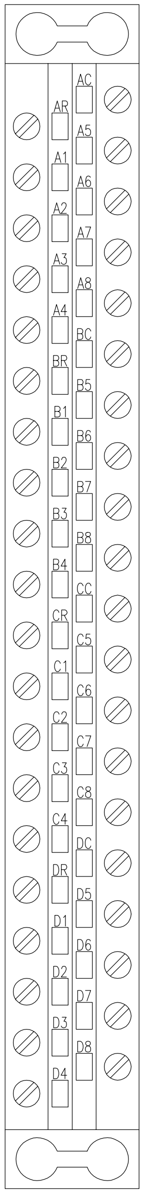
	Platinum	Copper	Nickel
°F x 10	–328.0 to 1562.0	–328.0 to 500.0	–112.0 to 527.0
°C x 10	–200.0 to 850.0	–200.0 to 260.0	–80.0 to 275.0

Table 4-9 Performance Specifications

	Typical 20–30° C	Typical 68–86° F	Specification Limit 0–60° C	Specification Limit 32–140° F
Mean accuracy (100 Ω platinum RTD)	0.1°C 0.2°F 0.03 Ω		0.6°C 1.2°F 0.25 Ω	
Repeatability (100 Ω platinum RTD)	0.1°C 0.2°F 0.03 Ω		0.3°C 0.5°F 0.11 Ω	
Absolute accuracy 100 Ω platinum Ω x 100 Ω x 10	 0.2°C 0.4°F 0.06 Ω 0.1 Ω + 0.05% of input Ω		 0.9°C 1.7°F 0.36 Ω 0.1 Ω + 0.2% of input Ω	
Common-mode rejection	100 dB		–	
Normal-mode rejection at line frequency ±0.01 Hz	80 dB		–	
Normal-mode rejection at line frequency ±3 Hz	25 dB		–	
Update time per active input: all inputs < 470Ω	< 110 ms @ 60 Hz filter		< 120 ms @ 50 Hz filter	
Step response time: 100Ω platinum RTD	250 ms X number of active inputs			
Step response time: inputs to 2KΩ	625 ms X number of active inputs			
Base power +5 V –5 V	 2.2 W max, 1.6 W typical 0 W			
Isolation (inputs to controller)	1500 Vrms			



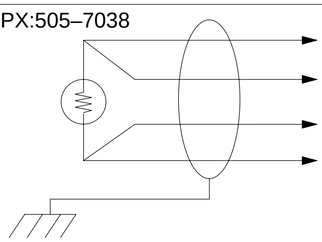
Wiring diagram



Terminal	Connections
AR	RTD1 sense–
A1	RTD1 return
A2	RTD1 sense+
A3	RTD1 source
A4	shield
AC	RTD2 sense–
A5	RTD2 return
A6	RTD2 sense+
A7	RTD2 source
A8	shield
BR	RTD3 sense–
B1	RTD3 return
B2	RTD3 sense+
B3	RTD3 source
B4	shield
BC	RTD4 sense–
B5	RTD4 return
B6	RTD4 sense+
B7	RTD4 source
B8	shield
CR	RTD5 sense–
C1	RTD5 return
C2	RTD5 sense+
C3	RTD5 source
C4	shield
CC	RTD6 sense–
C5	RTD6 return
C6	RTD6 sense+
C7	RTD6 source
C8	shield
DR	RTD7 sense–
D1	RTD7 return
D2	RTD7 sense+
D3	RTD7 source
D4	shield
DC	RTD8 sense–
D5	RTD8 return
D6	RTD8 sense+
D7	RTD8 source
D8	shield

Typical RTD Sensor Wiring Diagram

PPX:505–7038



Sense– High impedance negative voltage sense line, used only in 4–wire mode

Return Current return, common to all channels, negative reference in 3–wire mode

Sense+ High impedance positive voltage sense line

Source Excitation current and 3–wire voltage sense line

For accurate 3–wire lead compensation, match return and source lead impedances and temperature coefficients.

I000615

4.15 Parallel Word Input Module

The Parallel Word Input Module accepts 8 16-bit parallel words of binary data. Since there are 16 data lines (DBs), each module can multiplex 8 16-bit inputs by using strobe signals. All 8 16-bit inputs are wired in parallel, and the strobing activity selects which input is read.

Model Number	Number of Inputs	I/O Slots
PPX:505–6308	8	2

Parallel Word Input

Data Lines
TTL,CMOS (up to 28 VDC)

Strobe Lines
TTL, CMOS (up to 28 VDC)

Data Line Protection
Overvoltage to 40 VDC,
Reverse voltage protected

Update Time
8 milliseconds maximum

Data Active Level
Low true or high true (selectable)

Internal Resistance
Pull up or pull down (selectable)

Strobe Active Level
Low true or high true (selectable)

Strobe Source
Module-supplied or user-supplied

Channel Operation
Single or multiplexed

Input Word Length
16 bits

User Power Supply
20 to 30 VDC @ 0.35 A with
maximum ripple of +/– 0.4 V
UL listed Class 2

Maximum Power from base
4 W

Jumper Selections

Option	Jumper Installed	Jumper Not Installed
(1) Data active	High true	Low true
(2) Strobe active	High true	Low true
(3) Strobe source	User-supply	Module supply
(4) Internal Resistance	Pull-down	Pull-up

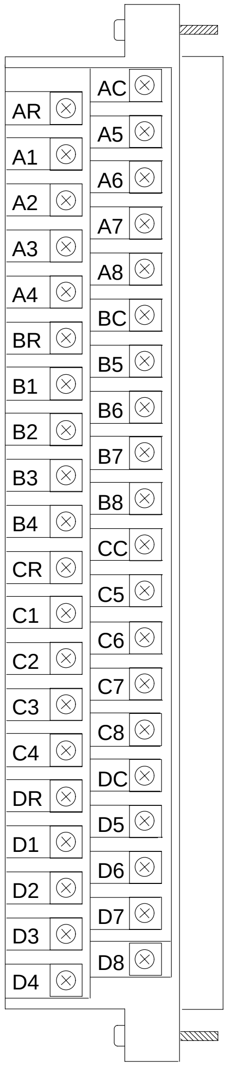
Data Line Voltage Ranges (VDC)

State	0 – 0.8 V	>0.8 – 2.0 V	>2.0 – 28.0 V
Low true	Set to 1	Indeterminate	Set to 0
High true	Set to 0	Indeterminate	Set to 1

Strobe Line Voltage Ranges (VDC)

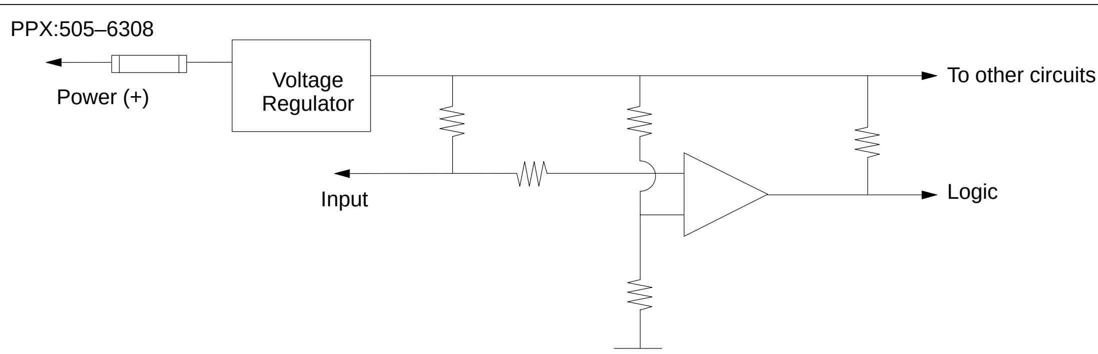
State	0 – 0.8 V	>0.8 – 2.0 V	>2.0 – 5.25 V
Low true	Read data	Indeterminate	Not reading
High true	Not reading	Indeterminate	Read data

I000616



Terminal	8-Word Input
AR	User Power (–)
A1	Option 1 Data Active jumper
A2	Option 1
A3	Option 2 Strobe Active Level jumper
A4	Option 2
AC	User Power (+)
A5	DB1
A6	DB2
A7	DB3
A8	DB4
BR	Not used
B1	Option 3 Strobe Source jumper
B2	Option 3
B3	Option 4 Internal Resistance jumper
B4	Option 4
BC	Signal common
B5	DB5
B6	DB6
B7	DB7
B8	DB8
CR	Signal common
C1	Strobe 1
C2	Strobe 2
C3	Strobe 3
C4	Strobe 4
CC	Signal common
C5	DB9
C6	DB10
C7	DB11
C8	DB12
DR	Signal common
D1	Strobe 5
D2	Strobe 6
D3	Strobe 7
D4	Strobe 8
DC	Signal common
D5	DB13
D6	DB14
D7	DB15
D8	DB16

Typical Input Circuit



I000617

4.16 Parallel Word Output Module

The Series 505 Parallel Word Output module connects field devices, such as seven-segment displays or digital-to-analog converters, to the controller. Any device that uses a multi-bit parallel word, such as Binary Coded Decimal (BCD) or Gray code, may be connected. Each module can multiplex eight 16-bit outputs by using strobe signals.

Model Number	Number of Outputs	I/O Slots
PPX:505–6408	8	2

Parallel Word Output

TTL Output Signal Drive
- Logic high
  - 4.5 mA max source at 2.4 VDC
- Logic low
  - 30 mA sink current at 0.55 VDC

CMOS Output Signal Drive
- Logic high
  - 0.35 mA max source at 4.4 VDC
- Logic low
  - 3.5 mA max sink at 0.4 VDC

Update Time
- 8 milliseconds maximum

Data Active Level
- Low true or high true (selectable)

Strobe Active Level
- Low true or high true (selectable)

Strobe Source
- Module-supplied or user-supplied

Channel Operation
- Single or multiplexed

Output Word Length
- 16 bits

User Power Supply
- 20 to 30 VDC @ 0.35 A,UL Class 2

Voltage Protection
- Overvoltage to 40 VDC,
- Reverse voltage protected

Maximum Power from base
- 5 W

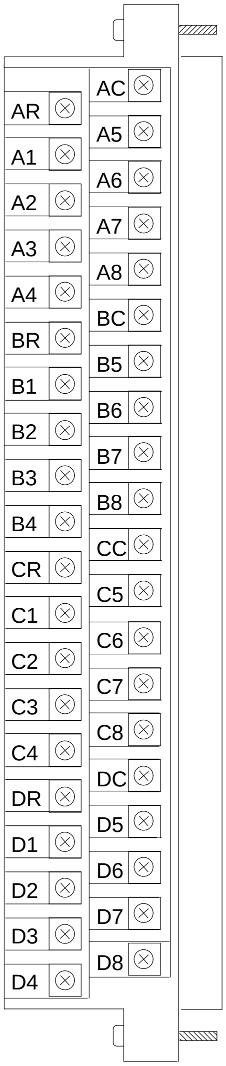
Jumper Selections

Option	Jumper Installed	Jumper Not Installed
(1) Data active	High true	Low true
(2) Strobe active	High true	Low true
(3) Channel Operation	Single channel	Eight channels

Drive Capabilities of Data/Strobe Lines

Output Current	Voltage Level	Logic Level
4.5 mA source	2.40 VDC	TTL high true
30 mA sink	0.55 VDC	TTL low true
0.35 mA source	4.40 VDC	CMOS high true
3.5 mA sink	0.40 VDC	CMOS low true

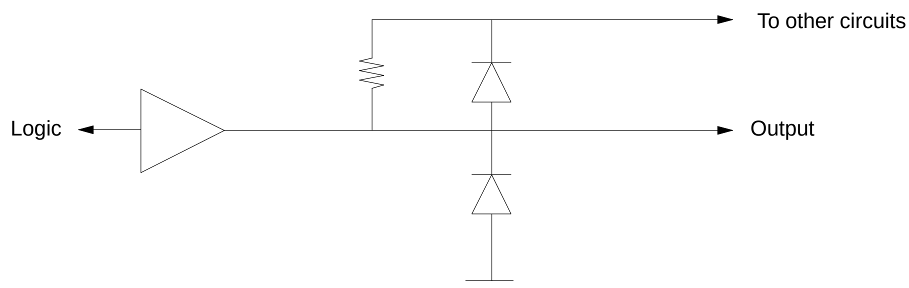
I000618



Terminal	8-Word Output
AR	User Power (–)
A1	EPF out
A2	Not used
A3	Option 1 Data Active Level jumper
A4	Option 1
AC	User Power (+)
A5	DB1
A6	DB2
A7	DB3
A8	DB4
BR	Not used
B1	Option 2 Strobe Active Level jumper
B2	Option 2
B3	Option 3 Channel Operation
B4	Option 3
BC	Signal common
B5	DB5
B6	DB6
B7	DB7
B8	DB8
CR	Signal common
C1	Strobe 1
C2	Strobe 2
C3	Strobe 3
C4	Strobe 4
CC	Signal common
C5	DB9
C6	DB10
C7	DB11
C8	DB12
DR	Signal common
D1	Strobe 5
D2	Strobe 6
D3	Strobe 7
D4	Strobe 8
DC	Signal common
D5	DB13
D6	DB14
D7	DB15
D8	DB16

Typical Output Circuit

PPX:505–6408



I000619

4.17 High-Speed Counter and Encoder Module

The Series 505 High-Speed Counter (HSC) and Encoder module provides two independent high-speed counter and output control channels. The module operates asynchronously from the controller scan. Four counter modes, four inputs, and two outputs are provided.

Model Number	Number of Channels	I/O Slots
PPX:505–7002	2	1

High-Speed Counter

(Count Inputs)

Frequency (maximum)

Duty Cycle	Count Mode Frequency	Quadrature Mode Frequency
10%	10 Khz.	4 Khz
20%	25 Khz	8 Khz
25%	30 Khz	10 Khz
30%	35 Khz	12 Khz
40%	50 Khz	15 Khz
50%	45 Khz	15 Khz
60%	35 Khz	15 Khz

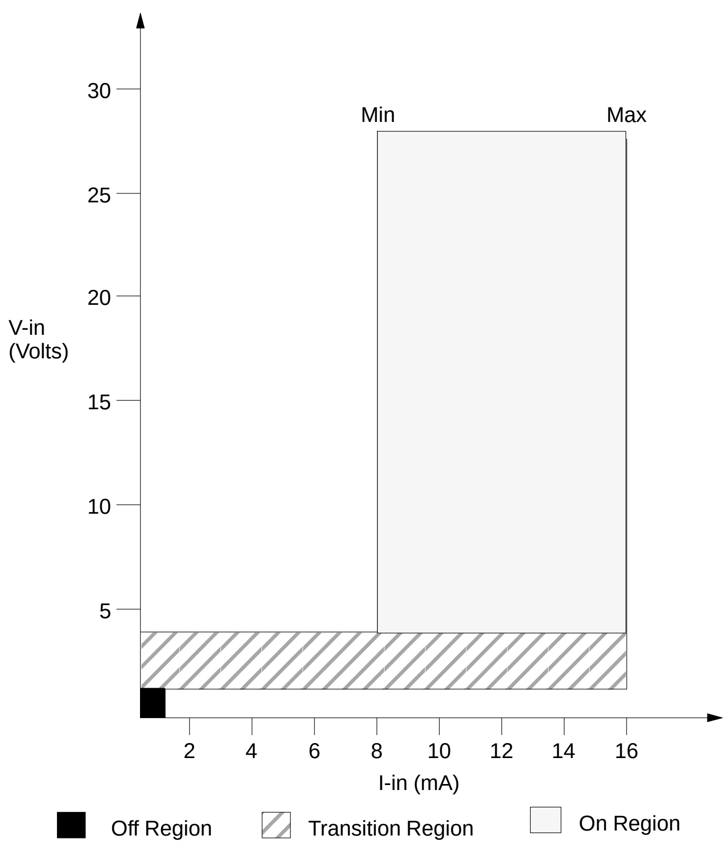
On voltage
4 to 28 VDC, Class 2 supply

Current on
8 to 16 mA

Off voltage
–1.5 to 1.5 VDC

Current off
0 to 1 mA

Volt–Ampere Curve for Count Inputs



I000620

Model Number	Number of Channels	I/O Slots
PPX:505–7002	2	1

High-Speed Counter

(Reset and Inhibit Inputs)

On voltage
4 to 28 VDC, Class 2 supply

Current on
2 to 40 mA

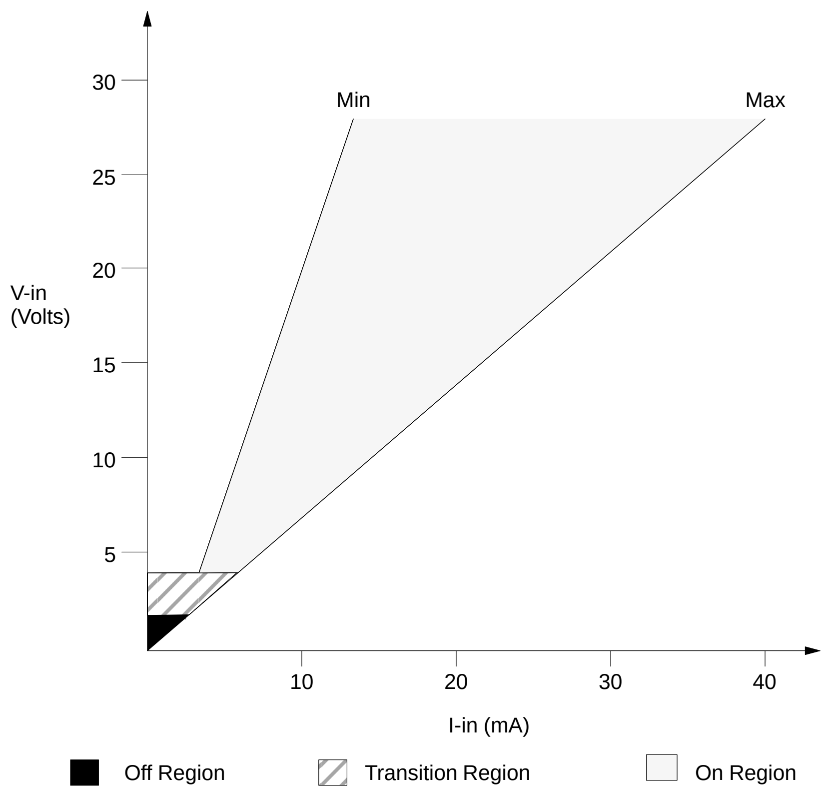
Off voltage
–1.5 to 1.5 VDC

Current off
0 to 0.75 mA

Reset delay time
unfiltered
950 µs on
950 µs off
filtered:
3.75 µs on
5.00 µs off

Inhibit delay time:
200 µs on
200 µs off

Volt–Ampere Curve for Reset and Inhibit Inputs



I000621

Model Number	Number of Channels	I/O Slots
PPX:505–7002	2	1

User voltage
28 VDC maximum

Voltage drop
1.8 VDC max. @ 400 mA
1.0 VDC max. @ 25 mA

Leakage current
500 µA max.

Peak current
500 mA for 1 ms

Isolation
1500V rms User-side to P/C-side
1500V rms Channel 1 to 2 (inputs)
0V rms Channel 1 to 2 (outputs)
1500V rms Inputs to outputs

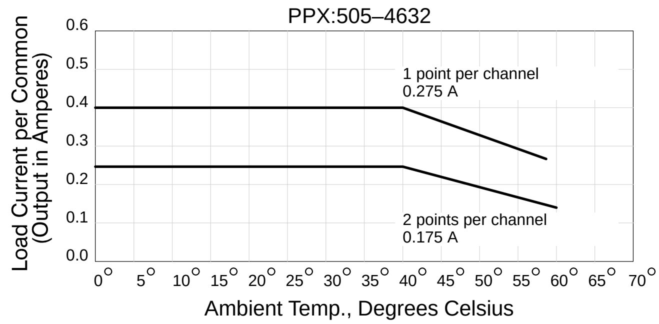
Power Required from Base
2.5 W

Response Time
1.1 ms
(Input pulse to output action)

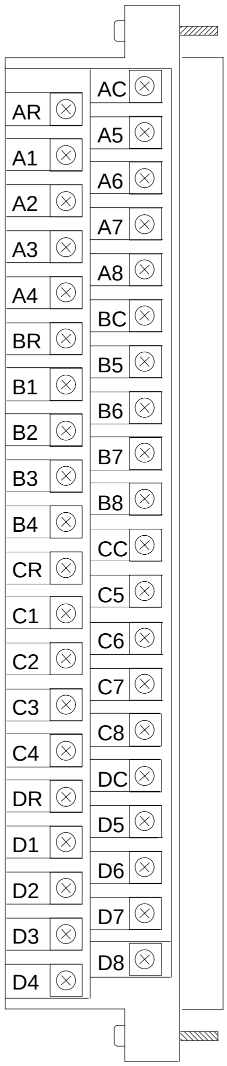
High-Speed Counter

(Outputs)

Derating Curves



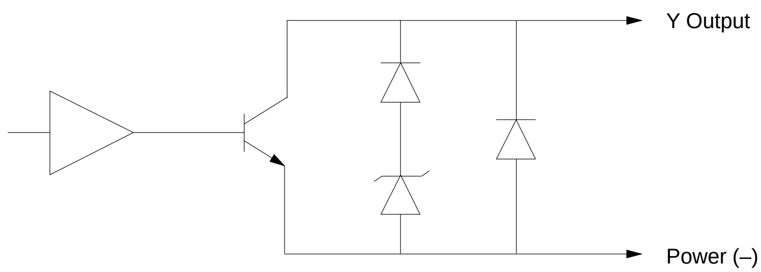
I000622



Terminal	Connections
AR	Not used
A1	Channel 1, A input (–)
A2	Channel 1, B input (–)
A3	Channel 1, Reset (–)
A4	Channel 1, Inhibit (–)
AC	Not used
A5	Channel 1, A input (+)
A6	Channel 1, B input (+)
A7	Channel 1, Reset (+)
A8	Channel 1, Inhibit (+)
BR	User Power (–)
B1	Channel 1, output 1 (Y1)
B2	Channel 1, output 2 (Y2)
B3	Not used
B4	Not used
BC	Not used
B5	Not used
B6	Not used
B7	Not used
B8	Not used
CR	Not used
C1	Channel 2, A input (–)
C2	Channel 2, B input (–)
C3	Channel 2, Reset (–)
C4	Channel 2, Inhibit (–)
CC	Not used
C5	Channel 2, A input (+)
C6	Channel 2, B input (+)
C7	Channel 2, Reset (+)
C8	Channel 2, Inhibit (+)
DR	User Power (–)
D1	Channel 2, output 1 (Y1)
D2	Channel 2, output 2 (Y2)
D3	Not used
D4	Not used
DC	Not used
D5	Not used
D6	Not used
D7	Not used
D8	Not used

Typical Output Circuit

PPX:505–7002



I000623

4.18 BASIC Module

The Series 505 BASIC module is an intelligent co-processor for the SIMATIC® 505™ controllers. Applications requiring complex math, data handling, or external device interfaces (serial RS-232-C) are well-suited for the BASIC module. The module executes applications programs, written in BASIC, independently of the controller scan unless the application program requests an exchange of information with the controller.

BASIC Features

The BASIC operating system includes a full complement of floating-point math, Boolean, and ASCII string handling instructions. This includes multi-dimensional numeric and ASCII array handling. For/Next loops and subroutines are also provided.

User Memory

The module has 28K bytes of user program and data storage. This memory has battery backup to protect the application program and retentive array data during power outages. The module also provides an automatic restart feature that is useful if system power outages occur.

Communication

Two standard RS-232-C communication ports are included. Both ports provide input and output character buffers, baud rates from 110 to 19.2K baud, selectable parity, and selectable character lengths. The ports are configured as Data Terminal Equipment (DTE).

PLC Access

The BASIC module has two levels of access to the controller. Both levels are directly accessible through simple BASIC statements added to the application program. Normal I/O access is available with the IOREAD and IOWRITE program statements. These commands give direct access to the RLL program through WX and WY I/O points assigned to the BASIC module.

Access to the controller data areas, including V-memory, timers, counters, drums, word and discrete I/O image registers, is available through PCREAD and PCWRITE statements.

EEPROM User Program Storage

User programs can be stored in non-volatile EEPROM.

Clear Input/Output Buffers

Call statements allow the BASIC program to delete the data from either or both communications port I/O buffer data.

Print Null Character

The print NULL character feature allows you to print NULL characters out of Port1 or Port2.

Model Number	Number of Outputs	I/O Slots
PPX:505–7101	0	1

Programmable BASIC

User Memory
28k bytes, battery backed RAM
EEPROM Prog Storage Optional

Ports
2, RS–232–C/423

Memory Buffers
Input: 28 ASCII characters
Output:
128 ASCII characters, port 1
1024 ASCII characters, port 2

Baud Rate
110 to 19.2K

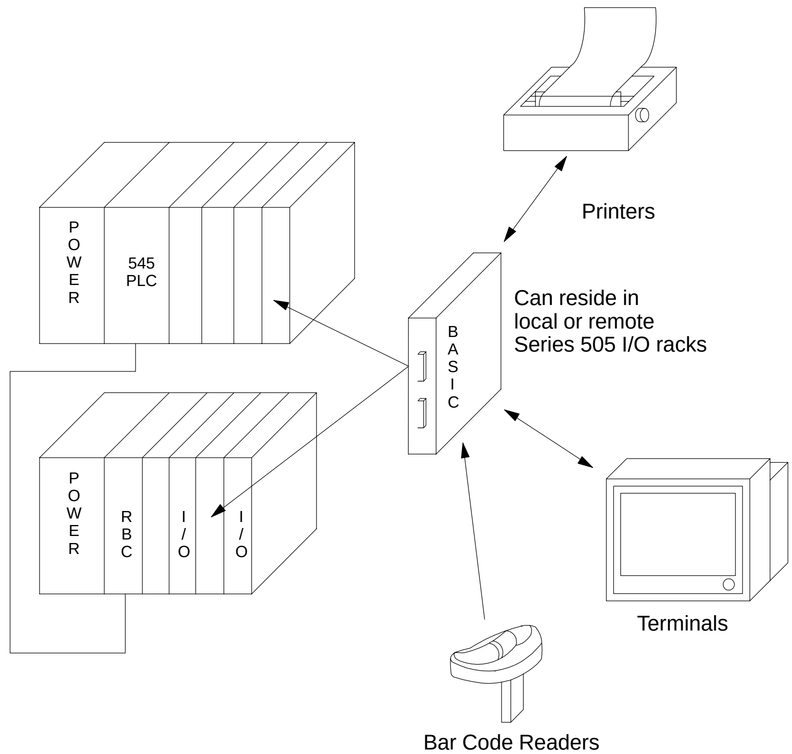
Communication Distance
40–220 m (131–721 ft.)

Parity
Even, odd, or no parity

Bit size
7 or 8 bit characters

Operation Mode
DTE

Maximum Power from base
6 W



I000624

4.19 Coprocessors

The Series 505 386/ATM Coprocessor is a general-purpose, high-speed IBM PC/AT® compatible computer in an I/O module form factor with a real-time interface to the 505 family of programmable controllers. The Series 505 386/ATM addresses the need to integrate high performance computing with the real-time control of a programmable controller

The 386/AT contains an 80C386SX™ microprocessor which operates at 8 or 16 Mhz and comes in two models: 2 megabytes of DRAM and a 20 megabyte hard disk (PPX:505/ATM–0220); and 4 megabytes of DRAM and a 40 megabyte hard disk (PPX:505/ATM–0440). Both models have a high density 3-1/2 inch floppy disk drive, EGA/VGA video, battery-backed real-time clock, and ports for keyboard, video, parallel and serial outputs. The 386/ATM plugs directly into the PLC chassis and draws power from the chassis power supply over the I/O backplane.

Features

The 386/ATM Coprocessor offers the following features:

- IBM-PC/AT compatibility provides a well-defined programming and operations environment.
- Series 505 I/O family form factor in a triple-wired module with direct plug into the PLC chassis.
- Direct PLC backplane interface permits maximum data transfer rate per PLC scan.
- Industrial hardened; designed to minimize the effects of electronic noise; tested for EMI, RF, ESD, thermal, humidity, shock and vibration.
- Built-in internal diagnostics on power up and continuous DRAM parity checking.
- Will operate without a keyboard or monitor if the application does not require them.

Table 4-10   386/ATM Coprocessor Specifications

CPU	80C386SX (socket for 80C87SX math coprocessor)	
Memory	PPX:505/ATM–0220	PPX:505/ATM–0440
DRAM w/parity	2M byte	4M byte
Hard Disk	20M byte	40M byte
Floppy Disk Drive	3–1/2 inch 1.44M byte	
Operating System	MS-DOS 3.3	
Communication Ports	2 Serial Communication Ports (RS-232/423) Rates: 110 to 57600 baud 1 Parallel printer port	
Other Ports	VGA analog video port 1 IBM-PC/AT compatible keyboard port	
I/O Bus Communication	Integrated Interface to the PLC I/O Bus	
Channels Per Module	8 analog I/O (4WX, 4WY)	
Data Communication rate over PLC I/O Bus (Max. per PLC scan)	2048 bits +8 analog I/O or 480 16-bit words –8 analog I/O or combinations of both	
Power Consumption (Typical)	11 watts @ 5 VDC; 0.2 watts @ –5 VDC	
Diagnostic	Internal diagnostic on power up. Continuous DRAM parity check.	
Operating Temperature	5 to 45° C (41 to 113° F)*	
Storage Temperature	–40° to 60° C (–10 to 140° F)	
Relative Humidity	20% to 80%, noncondensing	
Size	Triple-wide Series 505 I/O Module	

*5 to 50° C (41 to 122° F) when operating from the hard disk and not the floppy disk. the operating temperature, 5 to 45° C, is due to the limitations of the floppy disk media and not the 386/ATM Coprocessor.

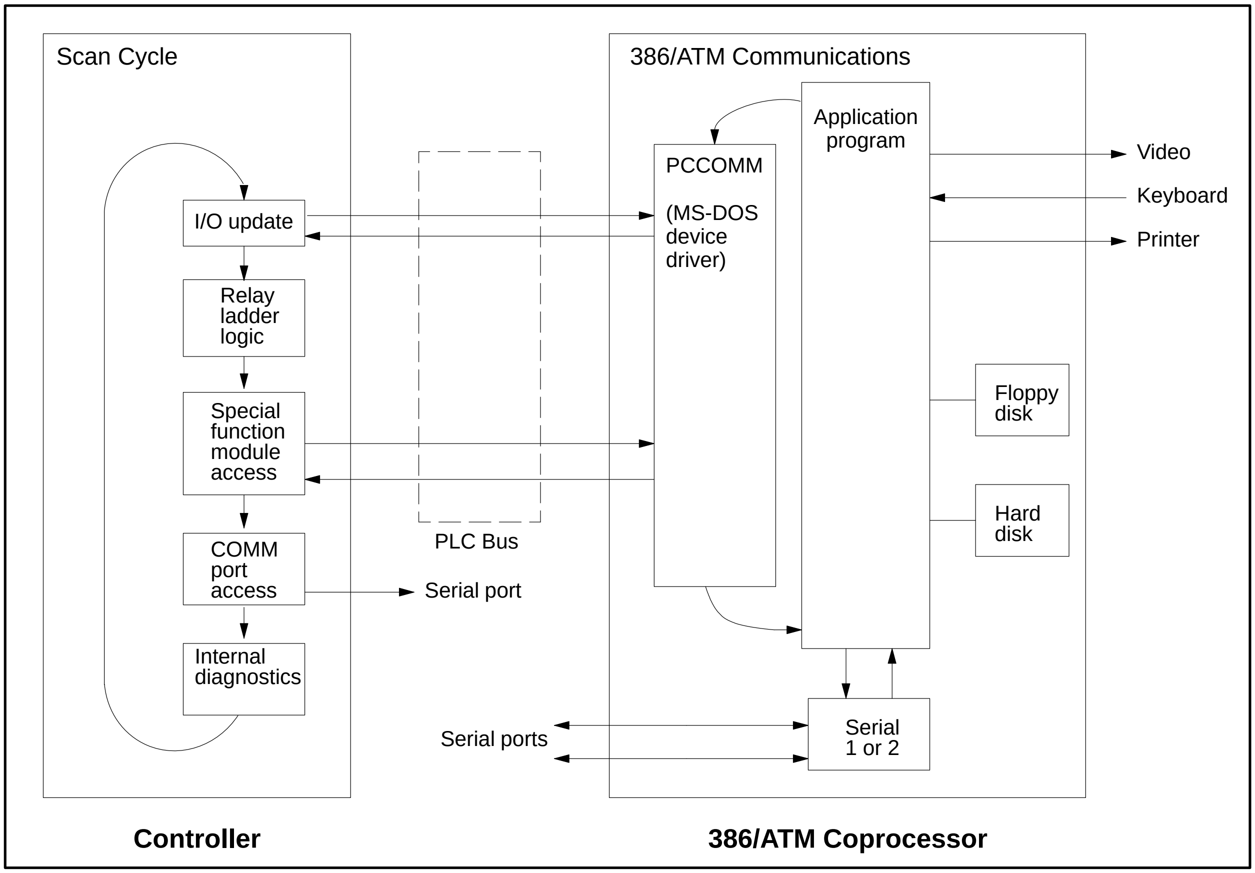


Figure 4-2 Interaction—386/ATM Coprocessor and Controller

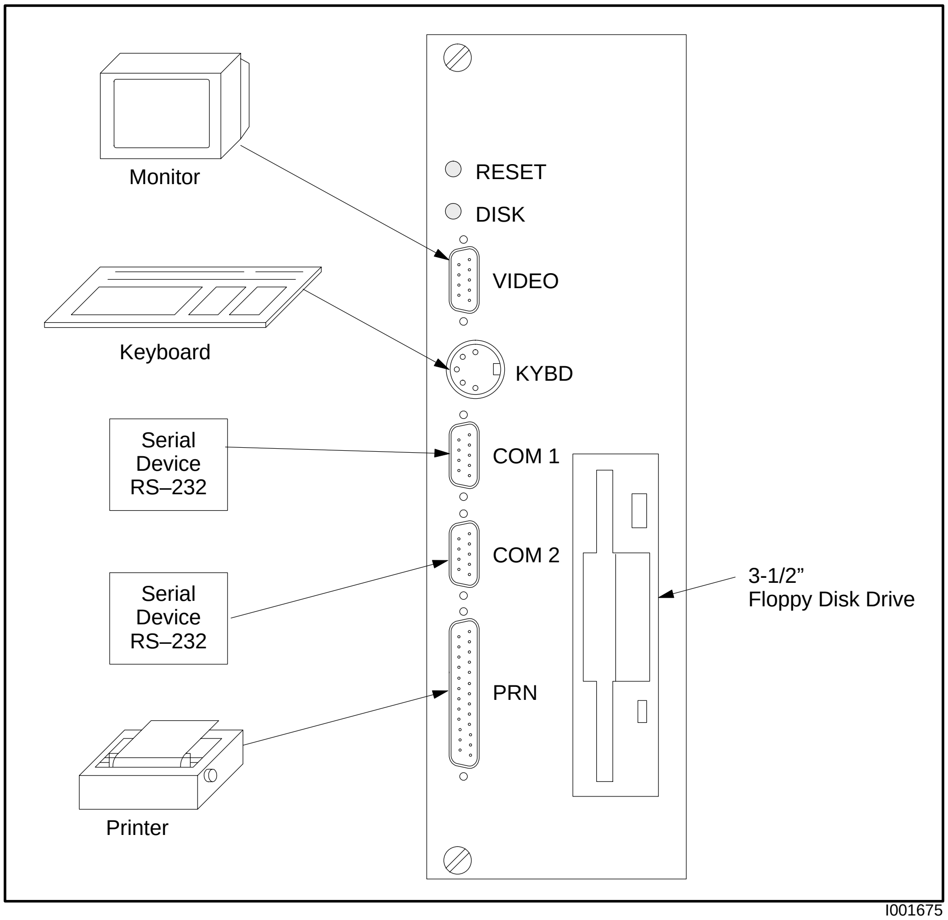


Figure 4-3   Bezel Features

4.20 Network Interface Module

The Series 505 Network Interface Module (NIM) provides an interface from a local or remote base to the TIWAY network. The NIMs accept high-level commands generated by a network primary and convert these commands to corresponding task codes which communicate with the PLCs. The purpose of the high-level commands is to remove as many differences as possible between secondaries on a TIWAY network. The application programs at the host level can then treat all TIWAY secondaries in a like manner.

Dual Local Line or Dual RS-232-C

The local line version provides two differential line ports for use with shielded, twisted-pair cable. This interface provides line isolation and exceptional noise immunity for installations requiring high data transmission rates or communication distances up to 25,000 feet.

The dual RS-232-C version provides two RS-232-C/423 ports for communicating over wide geographic areas using modems. The dual media selection supports redundant communication paths to the controller.

Modem Interface

The modem interface provides standard signals for using both half- and full-duplex modems. This interface is switch-selectable for either synchronous or asynchronous operation.

Switch-selectable Communication Rates

Both models of the Series 505 NIM provide switch-selectable data rates from 110 baud to 115.2k baud. This range of communication rates provides data rate matching to most commonly available modems and increased throughput via higher data rates available over Siemens Local Line™.

Universal Command Language (UCL)

The Series 505 NIMs communicate with host computers using Universal Command Language (UCL). The UCL high-level commands access data types in the same manner as all TIWAY conformant secondaries. This results in more efficient data access when compared to individual device drivers at the host application level.

Model Number	Media	I/O Slots
PPX::505–7339	Dual Local Line	1
PPX:505–7340	Dual RS–232–C	1

Data Rates
110; 150; 300; 600; 1200; 2400;
4800; 9600; 19,200;
38,400; 57,600; 115,200 bps

RS-232-C/423 Ports
Configured as DTE,
synchronous or asynchronous,
full or half duplex,
NRZ or NRZI encoding.

Local Line Ports
default to asynchronous
half duplex, with NRZI encoding

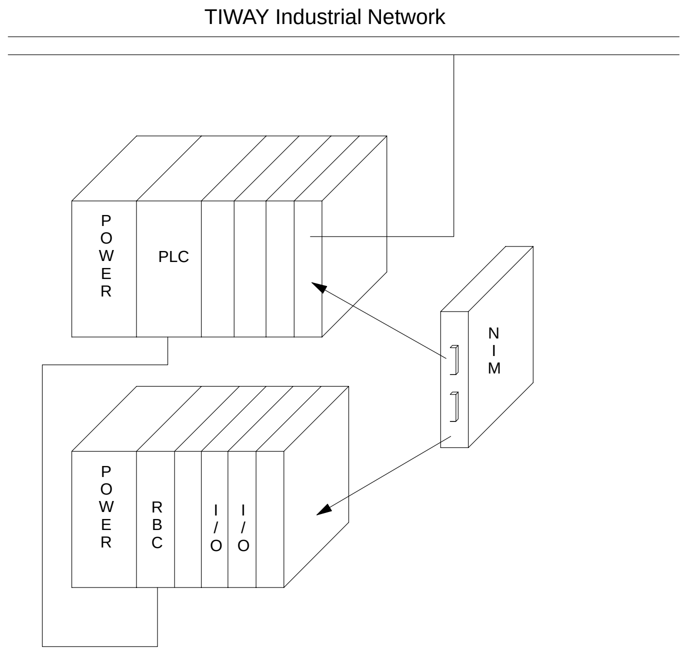
Data Link Protocol
High level data link
(HDLC, ISO 4335) unbalanced
normal response mode, ADCCP
(UN,4) class of procedure per
Chap. 11 of ANSI Standard
X3.66–1979.

Network Media
Shielded, twisted-pair (local line),
appropriate cable for RS-232-C

Undetected Bit Error Rate
6×10^{-13} (calculated) with specified cable

Maximum Power from base
8 W, +5 VDC

Network Interface



I000625

4.21 Peerlink Module

The Series 505 Peerlink module provides peer-to-peer data transmission from any controller on the network to all other controllers on the network. This module is perfect for transfer line applications or sequential processes which require controller interlocking.

The controllers or stations on the network are interconnected through a multidrop differential line current loop using twisted shielded pair cable as the media. The total maximum network distance is 10,000 feet, using premium cable.

Communication

The Peerlink network uses a broadcast method to achieve peer-to-peer communication. This eliminates the additional expense of a network host and avoids the protocol, hardware and software complexities of a token passing structure. With the broadcast method, one module on the network is designated as the active monitor. This active monitor polls each station on the network, and each station responds by broadcasting its data to all other stations.

The broadcast message of each station is mapped directly into designated variable memory locations of each controller CPU. This variable memory location is selectable through a word location in the controller. Each controller can act on the information from any other PLC on the network. No additional controller programming is required by the user.

Fault Tolerance

The Series 505 Peerlink module provides two methods of defined fault tolerance. The standard dual local line ports make redundant media connection possible. If communications on one channel are interrupted, the module will automatically transfer communications to the second channel.

In addition to the dual media, the module also includes an active-passive monitor option to achieve levels of fault tolerance. The designated active monitor assigns a second Peerlink module as the passive monitor. If an active monitor failure is detected, the passive monitor becomes the active monitor and assigns a new passive monitor allowing continued communications.

Model Number	Media	I/O Slots
PPX:505–7354	Dual Local Line	1

Peerlink

Channels per Module
Dual Local Line

Network Transmission Media
Twisted shielded pair

Total Network Distance
10,000 feet
(Belden 9860 or equivalent)
3,250 feet
(Belden 9271 or equivalent)

Number of Modules per Network
16 maximum

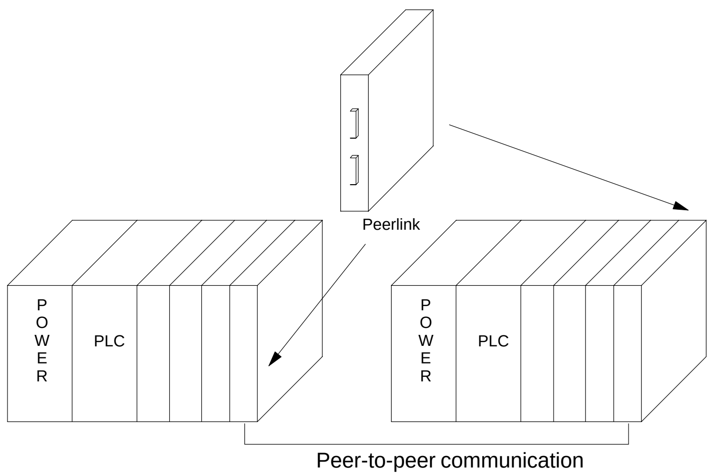
Data Transfer per Station
1 to 16 16-bit words
(switch selectable)

Predicted Network Update Time
20 ms — 4 stations at 8 words/station
35 ms — 8 stations at 8 words/station
45 ms — 8 stations at 16 words/station
55 ms — 16 stations at 8 words/station
75 ms — 16 stations at 16 words/station

Data Rate
115.2 Kbits per second

Undetected Bit Error Rate
6×10^{-13} (calculated)

Maximum Power from base
8 W, +5 VDC



I000626

4.22 MODNIM Module

The Series 505 MODNIM (PPX:505–5184) allows the SIMATIC 545 to be interfaced with a Modbus® host via a Modbus communication network. The configuration can be point-to-point or networked with the addition of modems. Modems allow a multi-drop network where multiple slave nodes are connected to the master host computer. See Figure 4-4.

Communications

The Modbus system is a Local Area Network (LAN) designed to work in an industrial environment. It is a master/slave network where a number of slave nodes (controllers) are connected to a master node (host computer).

Characteristics

The Modbus Protocol determines how messages are passed between the host computer and the secondary nodes. Up to 247 slaves can reside on a single network. Each node must have a unique address between the range of 1 to 247.

Message Types

Two types of messages are used, the Query/response type and the Broadcast/no response type.

- In the Query/response type of message a single node is addressed
- In the Broadcast/no response type all nodes are addressed, but none respond.

Protocol

Each message contains a envelope where the data is contained. The envelope directs the data to the correct address on the network. The envelope contains the information to determine if the contents were received correctly and instructions about what to do with the data.

Transmission Modes

Two types of transmission modes are available for a particular network, though only one can be selected. Each node must conform to that transmission mode. Selection of transmission mode is based on the requirements of the host computer. The two types of transmission are:

- ASCII (American Standard Code for Information Interchange)
- RTU (Remote Terminal Unit)

The Series 505 MODNIM is switch selectable to use either mode.

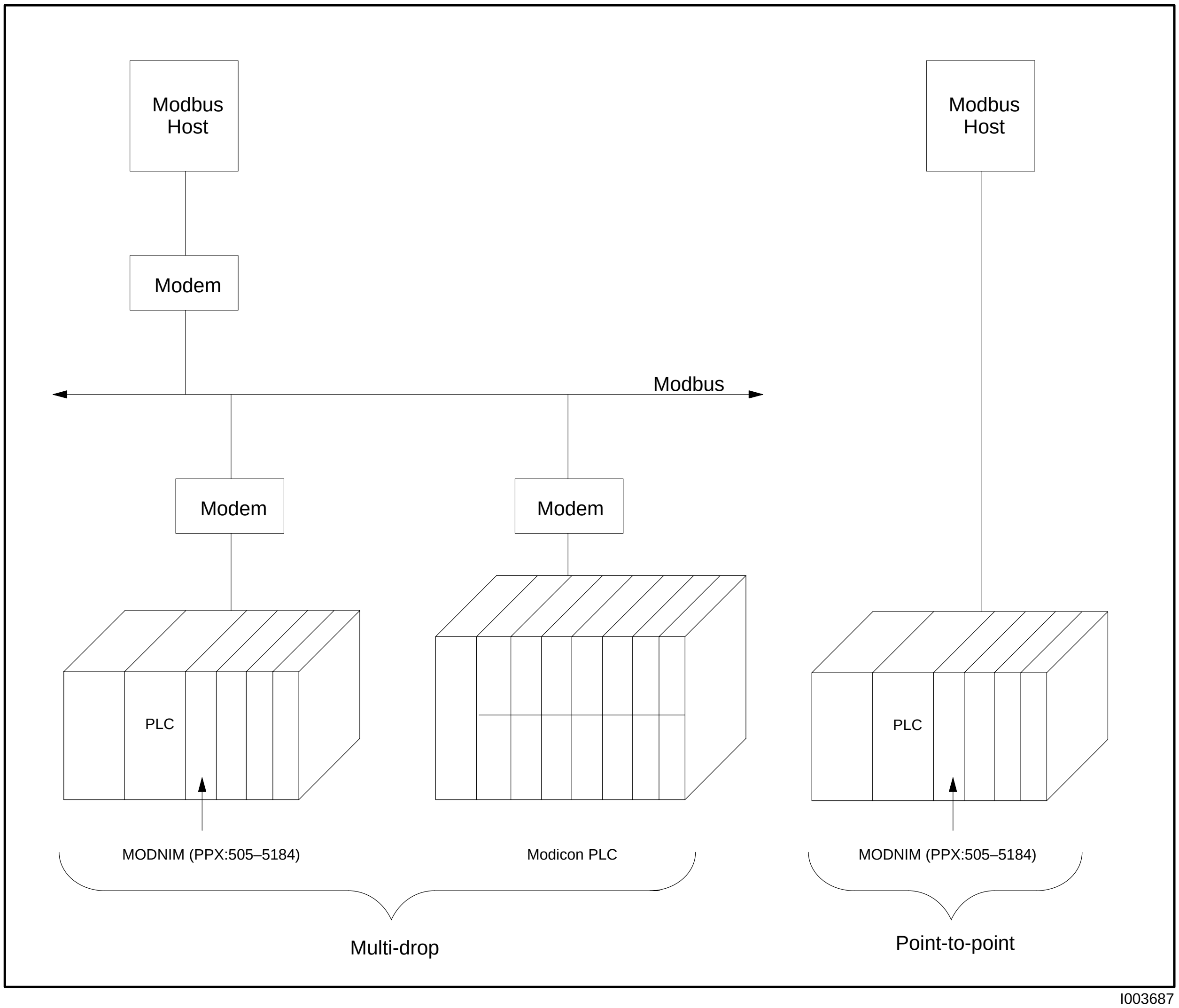


Figure 4-4 Multi-drop and Point-to-point Configurations

Transmission Mode Characteristics

ASCII and RTU transmission mode details are given in Table 4-11. The selection of a particular mode is determined by dipswitch settings

Table 4-11 Transmission Mode Details

Characteristic	ASCII	RTU (8 Bits)
Code used	Hexadecimal (ASCII printable)	8 bits binary
Bits per Character		
Start	1	1
Data	7	8
Stop	1 or 2	1 or 2
Parity	Optional	Optional
Checksum	LRC	LRC
Start of Transmission	:	Time period of 3.5 characters to determine start and finish of a message.
End of Transmission	CR/LF	

Supported Functions

There are 13 functions that a Series 505 MODNIM can perform on a receipt of a message from the host computer. These functions are described in Table 4-12. Element types supported include X, Y or CR, WX, and V.

Table 4-12 Modbus Functions Supported

Code	Type	Description
01	Read Coil Status	Get current status of a group of coils
02	Read Input Status	Get current status of a group of discrete inputs
03	Read Holding Register	Get current values from holding registers
04	Read Input Register	Get current values from input registers
05	Force a Single Coil	Change the state of a logic coil to on or off, forced on or forced off, or unforced.
06	Write a Single Register	Write a value into a holding register
07	Read Exception Status	Get the 8 internal status coil values
08	Execute Diagnostics	Send diagnostic tests to a slave
11	Comms. Even Counter	Enable the success or failure of a query to be determined
12	Get Comms. Log	Get the communications log for Modbus network transactions
15	Write Multiple Coils	Change a number of consecutive coils
16	Write Multiple Registers	Write values into a series of consecutive holding registers
17	Report Slave I.D.	Get the slave type and the condition of its run light

Table 4-13 MODNIM Specifications

Item	Description
Model Number	PPX:505–5184
Media	Dual RS–232C/423
I/O Slots	1
Data Rates	50, 75, 110, 150, 200, 300, 600, 1200, 1800, 2400, 3600, 4800, 7200, 9600, 19200
RS–232C/423 Ports	Configured as DTE, asynchronous, full or half (RTS/CTS) duplex, external modem support
Data Link Protocol	ASCII and RTU
Network Media	Appropriate cable for RS–232C
Maximum power from base	8W, +5VDC

4.23 Remote Base Controller

The remote I/O port on the SIMATIC 545 controls all communications between the controller and the Remote Base Controllers (RBCs).

The controller can address up to 2048 I/O points with one local I/O base plus up to 15 remote I/O bases accessed through the built-in remote I/O controller. All I/O points, whether local or remote, can be any mix of analog or discrete (except that no more than 1024 analog I/O points may be specified).

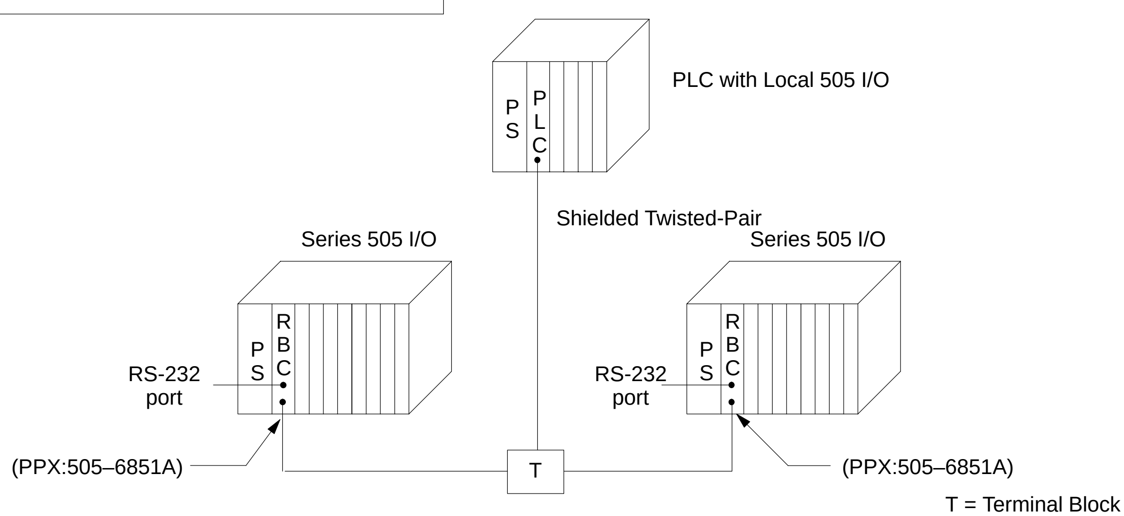
The shielded twisted-pair communication link is limited to 3300 feet from the controller. These distances are measured from the source to the most distant destination. The remote I/O bases can be up to 6600 feet (shielded twisted-pair) from each other by placing the controller in the middle of the cable run.

Series 505 RBCs install in the CPU slot of the I/O base. The communications cable attaches to the front of the RBC. Before installing, determine if the RBC will hold outputs at their last value or drive them to zero in case the RBC and CPU lose communications. The RBC has a watchdog timer which times out around 500 msec; if is desired that the outputs should hold the last value, move the on-board jumper from its default (go-to-zero) position.

The communication link uses shielded twisted-pair cable; cable with the characteristics of Belden 9860, 9271, or 9182 is recommended.

Model Number	Number of Outputs	I/O Slots
PPX:505–6851A	N/A	0, put in PLC slot

Remote Base Controller



I000627

4.24 Smart Slice

The Series 505 Smart Slice I/O module allows access to input and output points that are not located near a Series 500/505 base. The module attaches directly to the RS-485 remote I/O link and emulates a remote base. The module has ten inputs and six outputs. Power is derived from field excitation voltage.

The sinking inputs have a high-side (line or positive) terminal and the sourcing outputs have a low-side (neutral or negative) terminal. A signal replaceable fuse protects all the inputs and outputs. Status indicators are provided for each I/O point, the fuse, remote I/O communications, and module power.

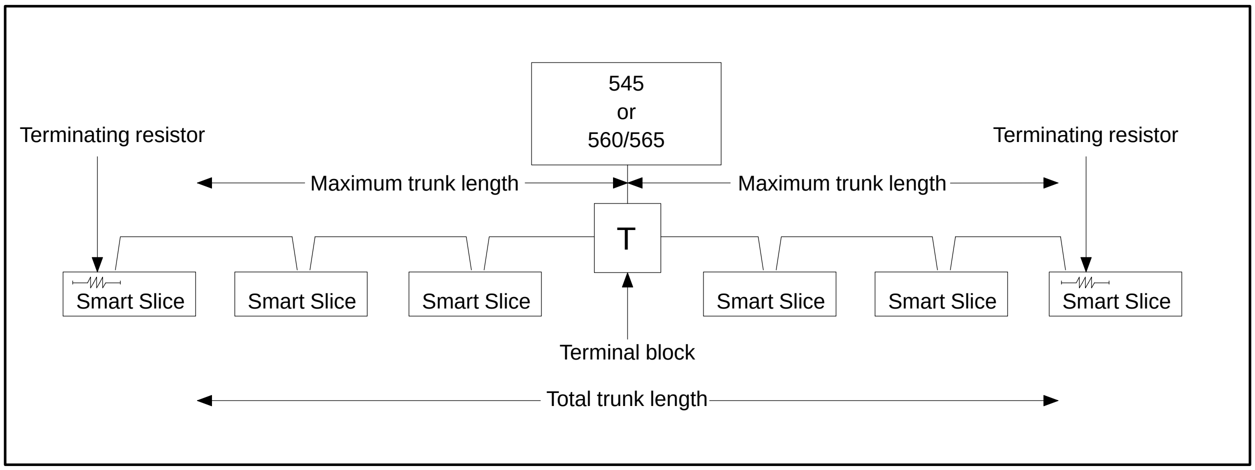


Figure 4-5 Communications Cable Configuration

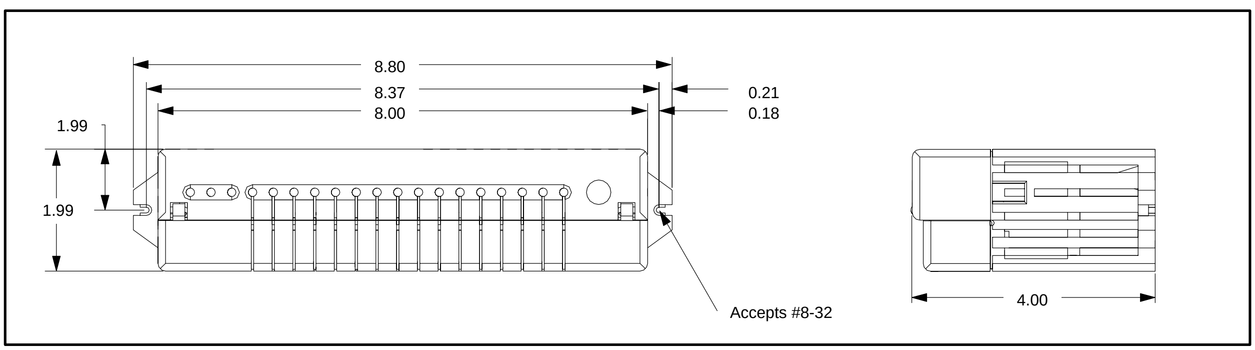


Figure 4-6 Mounting Tabs and Dimensions

Table 4-14 24VDC Smart Slice (PPX:505–9201) Specifications

Description	24 VDC, 10 inputs/6 outputs
Rated voltage	20–30 VDC
Input type	Sinking, high-side common
Input on voltage range	15 VDC min., 30 VDC max.
Input on current range	2.0 mA min., 6.0 mA max.
Input off voltage range	0.0 VDC min., 5.0 VDC max.
Input off current range	0.5 mA max.
Input delay time	2 ms max.
Output type	Sourcing, low-side common
Output current	0.5 A max at 60° C, 1.0 A max at 40° C
Output temporary overload	3.0 A for 1 ms
Output on voltage drop	1.0 VDC
Output off leakage current	0.02 mA max.
Output delay time	4.0 ms on-to-off, 2.0 ms off-to-on

Table 4-15 110VAC Smart Slice (PPX:505–9202) Specifications

Description	110 VAC, 10 inputs/6 outputs
Rated voltage	90–132 VAC, 47-63 Hz
Input type	Sinking, high-side common
Input on voltage range	79 VAC min., 132 VAC max.
Input on current range	4.0 mA min., 13.0 mA max.
Input off voltage range	0.0 VAC min., 20.0 VAC max.
Input off current range	1.0 mA max.
Input delay time	6 ms min., 40 ms max.
Output type	Sourcing, low-side common
Output current	0.5 A max at 60° C, 1.0 A max at 40° C
Output temporary overload	5.0 A rms for 2 cycles
Output DV/DT	300 V/µs
Output on voltage drop	1.4 Vrms at 1.0 A
Output off leakage current	2.0m A max.
Output delay time	11ms on-to-off, 2.0 ms off-to-on

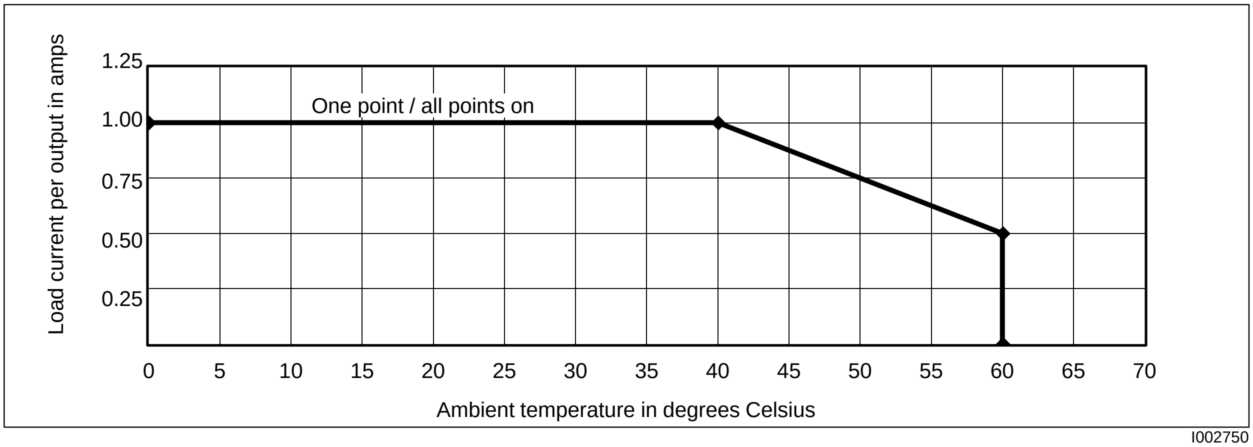


Figure 4-7 Module Derating

Chapter 5

Programming

5.1 TISOFT

Operation

TISOFT is the standard programming, documentation, and diagnostics tool for the SIMATIC TI programmable controllers. You use TISOFT for entering and editing your programs, as well as for accessing programming utilities such as “cut and paste” block operations and entering program comments. Menus enable simple step by step entry of program data, as well as easy access to the many TISOFT utilities. TISOFT operates on IBM PC/XT/AT® and compatible personal computers. For example, the menu chart in Figure 5-1 illustrates the ease of access to editing functions.

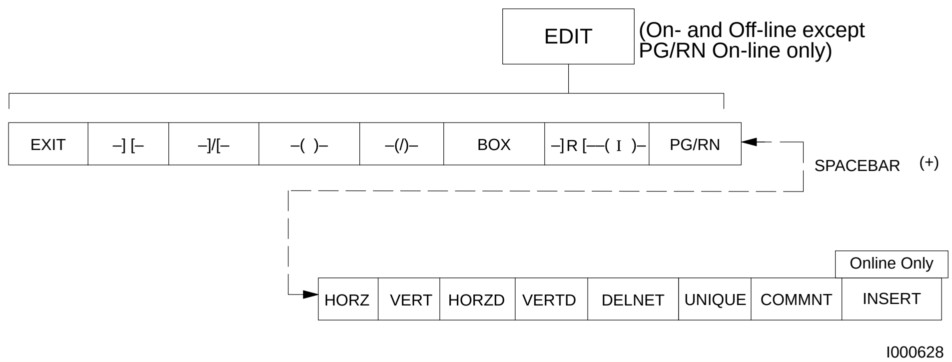


Figure 5-1 Accessing Program Functions

For more comprehensive information, consult the TISOFT manual.

Capabilities

TISOFT provides both off-line and on-line program development of user control programs. In addition to entering RLL and Special Function programs, you can also configure the 64 PID loops and 128 analog alarms. The PID loops and analog alarm blocks are easily configured from simple fill-in-the-blanks menus. Editor functions including block move and copy operations, and search and replace options allow you to quickly assemble and modify program segments from multiple sources to construct new control programs. When you have certain program segments that are used repeatedly, you can store the keystrokes in a macro to further reduce the time spent on program entry. Complete program annotation and documentation support is also included in the TISOFT package.

You also use TISOFT for configuration of the control system, including memory configuration and I/O subsystem configuration. Program maintenance and troubleshooting is also simplified with the TISOFT package.

Program File Management

Program file management features allow you to use program and maintenance chart names that relate to the process or machine being controlled (see Figure 5-2). This greatly reduces confusion and makes it easy to match program files to machines. Programs can be copied, renamed, and deleted from within TISOFT. You can also store multiple programs in a single directory, multiple directories, or on floppy diskettes.

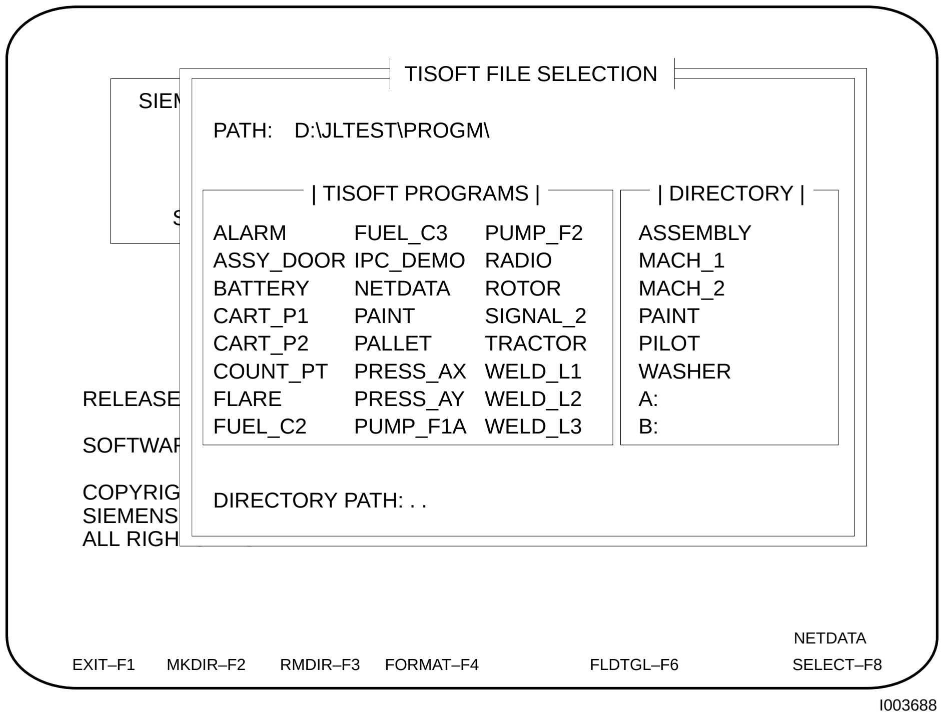


Figure 5-2 Program File Management

On-line Debugging and Maintenance

With TISOFT you can monitor and debug your RLL program on-line. Rung element status, as well as status charts of program variables, may be created to monitor discrete points, register locations, timers and counters, and word I/O points. TISOFT also allows you to simultaneously select and display multiple non-adjacent rungs.

A split screen format also allows simultaneous display of RLL rungs and predefined register status charts (see Figure 5-3). Ladder elements can be forced or unforced while being observed in status mode. The rung cross reference feature may also be invoked from the main ladder display. When used in conjunction with the non-adjacent rung selector, this provides a quick method of tracing program ladder elements.

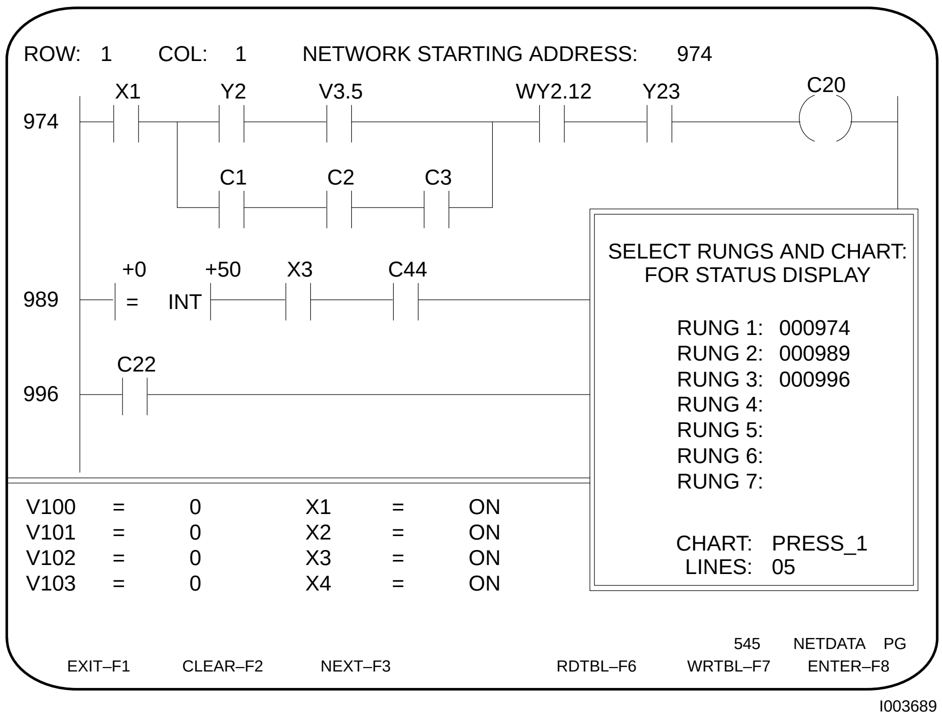


Figure 5-3 Split Screen Status Charts

Program Annotation and Documentation

TISOFT provides extensive program documentation options and features to permit you to fully document your control program. Program documentation features include 3-line by 11-character synonyms and 3-line by 48-character descriptors for rung elements, as well as 16-line by 60-character rung comments.

You enter synonyms and descriptors through a pop-up synonym editor, available from any program location. Synonyms for register memory locations allow documentation of program variables and constants. TISOFT also permits comment documentation of program PID loops, analog alarms, as well as Special Function programs and subroutines. Figure 5-4 shows an example of the pop-up synonym editor.

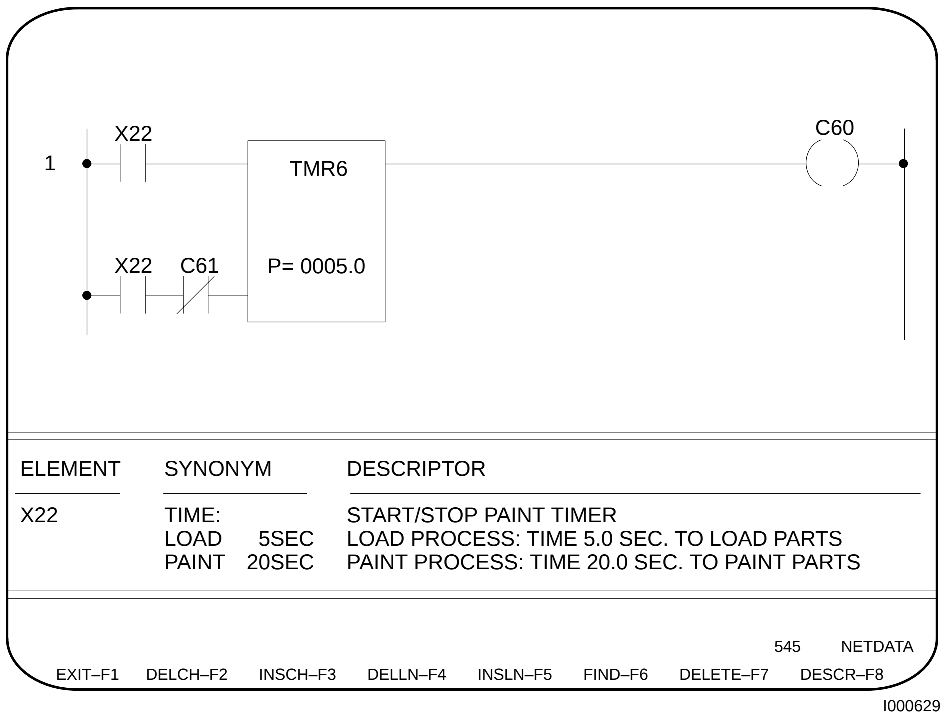


Figure 5-4 Example Synonyms and Descriptors

You can easily generate full program listings, as well as element cross-reference listings. An on-line cross reference for any RLL rung is also available with a single keystroke. Also, search and replace allows you to quickly and easily replace an element for a specific range or throughout the entire program. Search and replace operations can be performed individually, or you can complete a table that will search for and replace several program elements automatically. The miniature cross reference lists addresses and output usage for any input element on the rung along with input usage and locations for any output element in the rung. Figure 5-5 shows an example of a cross reference.

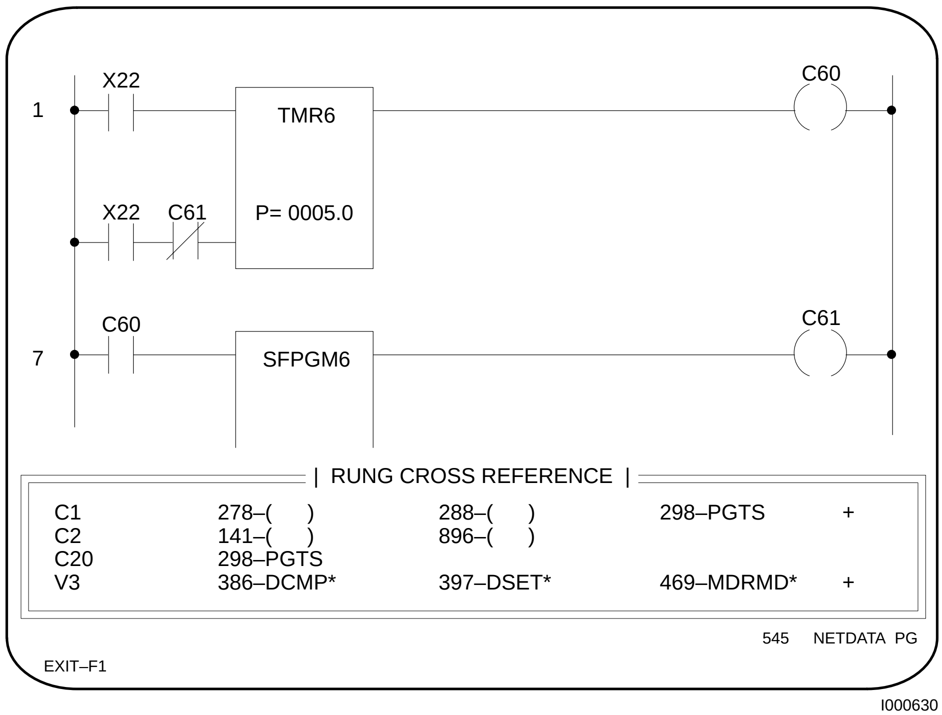


Figure 5-5 Cross-reference Display

Hard Keys

TISOFT gives quick access to many frequently used operations through a single key on the programming device keyboard. The following list describes some of the commonly used functions available with hard keys.

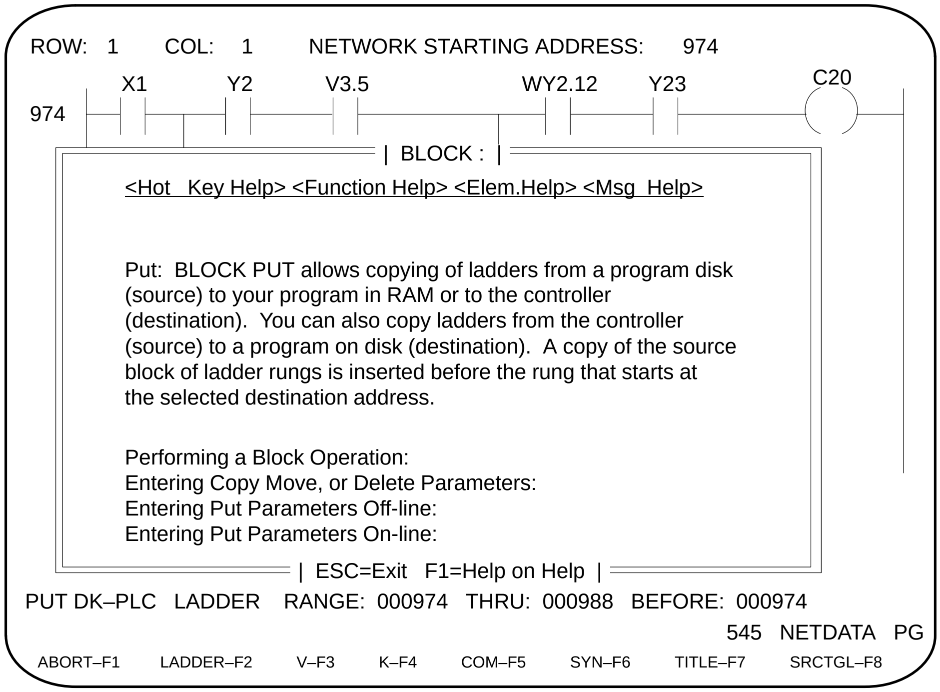
- The [Esc] key returns to the previous level of menu operations.
- The [PgUp] [PgDn] keys scroll the screen displays.
- The [←] ARROW keys move the cursor from field to field.
- The [Return] key enters displayed information or begins a requested operation.
- The [INS] [DEL] keys allow you to insert information or delete information.

Whenever a function appears on the menu, the hard key equivalent of the function is enabled. You can use the hard key to invoke the function rather than using a function key available on a TISOFT menu.

Comprehensive Help System

TISOFT offers various forms of assistance during program entry. In addition to automatic range checking for instruction parameters, detailed information about how to actually use the instruction is available through a comprehensive help system (see Figure 5-6.)

The help system is also context sensitive. This saves you time because you do not have to scroll through a help menu to find the topic you need. For example, if you are using the block put function and request help by pressing the [ALT] [H] key sequence, the help system automatically displays information associated with the operation. The [SHIFT] [?] key sequence also shows options at user-entered fields. See Figure 5-7.



I003690

Figure 5-6 Comprehensive LIST System

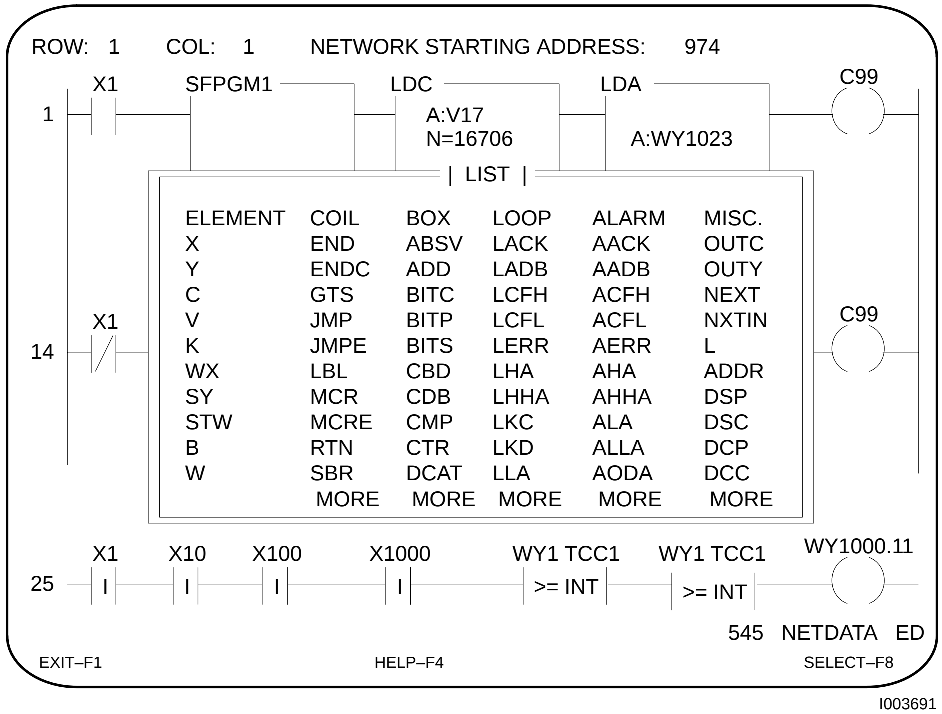


Figure 5-7 Typical Element LIST Screen

TISOFT Hardware Bases

You can use TISOFT on either an IBM PC/XT/AT® or compatible personal computer or a CVU10000. PC-DOS or MS-DOS® version 2.1 or higher operating system is required.

TISOFT requires a minimum of 540K of available RAM and supports color or monochrome graphics. Communications to the programmable controller may be established through a standard asynchronous communications adapter card, or you may optionally choose TIWAY network access using the TIWAY PC Adapter Card.

5.2 Applications Productivity Tool

The Applications Productivity Tool (APT) is a coordinated design and implementation environment that contains several different graphical and textual editors for specific design tasks. APT allows design and implementation of PLC control without having to program in Relay Ladder Logic (RLL). Experienced process-control users can be applying APT within a few days. It can cut control-solution design and implementation from months to weeks. APT can be used independently or as part of the TISTAR solution in configuring the SIMATIC 545 and SIMATIC 560T/565T controllers.

For more comprehensive information, consult the APT manual.

Graphical Programming Environment

APT is a graphical programming environment that eliminates the need to work in relay ladder logic while generating RLL for maintenance. Since APT compiles into structured RLL, it generates a language already familiar to the electrician for troubleshooting. APT represents a familiar structure for process and control engineers, allowing them to become more intimately involved in up front design work, thereby minimizing translation errors. Such terms as PID loops, devices and control blocks are used to facilitate better communication during the design cycle. APT serves as a productivity-enhancer and minimizes the need for re-training of process, control, and maintenance personnel.

Modular Programming

The APT environment supports modular programming techniques, allowing process/control engineer to program their control system from the top down. This eliminates the need to configure specific devices or I/O points before designing an overall control strategy. Devices, I/O, and other specific details can be easily configured later using the APT definition editors. APT allows the engineer to switch back and forth freely between the different editors, providing flexibility to configure the system details in any order. For example, an engineer can configure a motor type (device editor), then move to the I/O module editor, then to the recipe editor—there is no pre-defined order required for programming a control application.

Partitioning

APT programs are built from multiple units—groups of major process equipment which operate semi–independently (see Figure 5-8). Reactor systems or weighing systems are examples of units. Each process that can be controlled by one controller comprises one APT program. Partitioning a process not only simplifies debugging the program but reduces compilation time significantly. The process and instrumentation diagram (P&ID) above illustrates a process partitioned into 3 units.

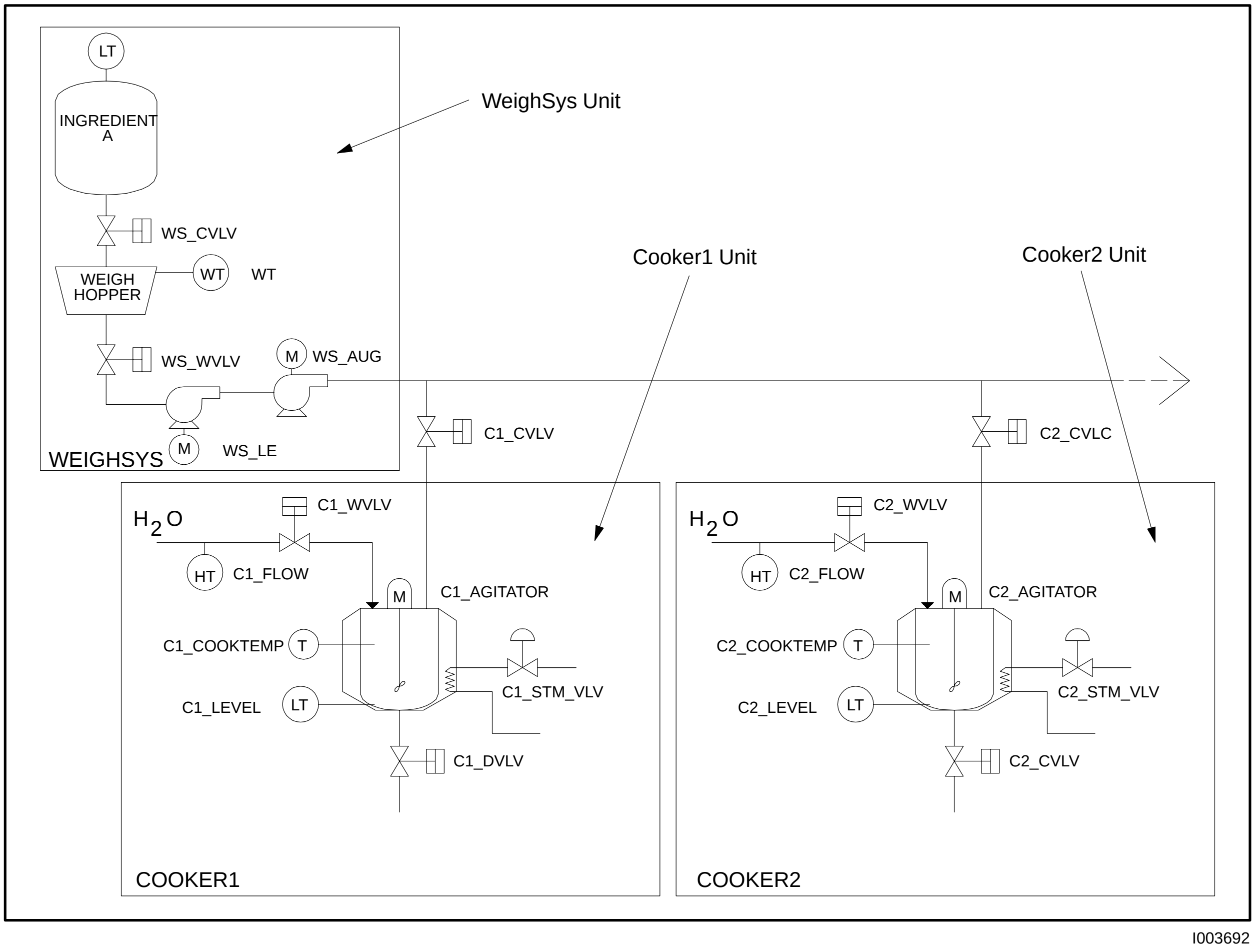


Figure 5-8 Example of Process Partitioning

APT Screen Hierarchy

APT is hierarchical in structure and uses overlapping windows to display three different levels of information (see Figure 5-9).

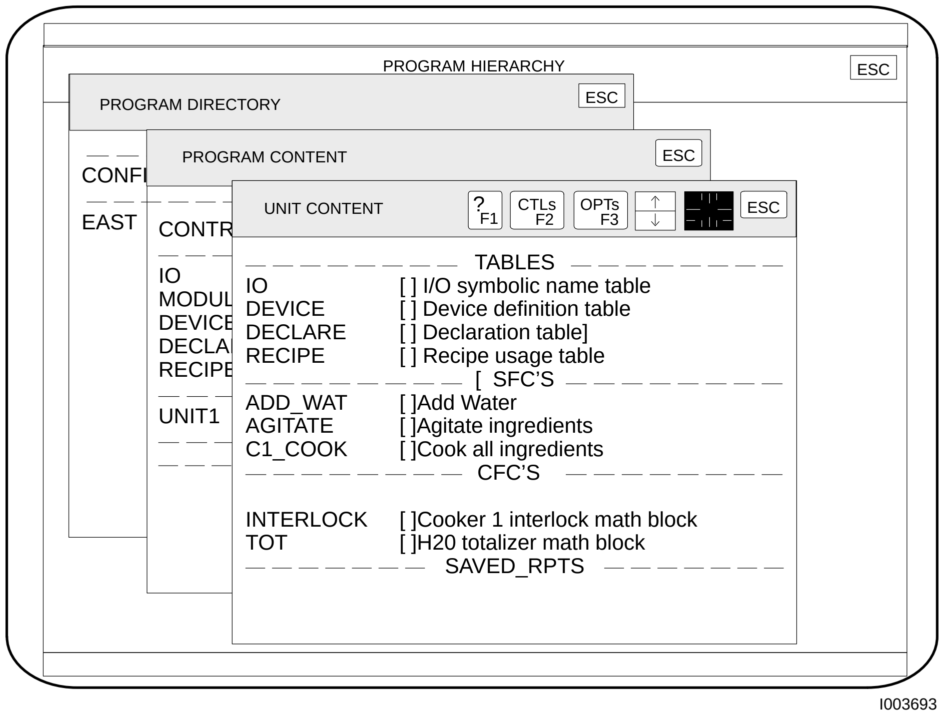


Figure 5-9 APT Screen Hierarchy

- **Program Directory** Top level in the hierarchy, provides access to system-wide information and program names that are resident on the APT disk, allows the manipulation of entire programs and their constituent parts; expands to show the contents of a program, and expands to perform functions on program parts.
- **Program Content** Mid-level in the hierarchy, provides information about a specific program, displays the names of a program's constituent parts, provides access to the editors and utilities that define program components, and facilitates moving to the other hierarchy levels.
- **Unit Content** Lowest level in the hierarchy, allows definition of the logic for each independent process (unit operation) or machine in the control program, and contains the logical sub-components of program.

Pull-down Option/Function Menus

Pull-down option/function menus are accessed by selecting the appropriate icons (see Figure 5-10). These icons display the various options and functions available to the user within a particular level of the hierarchy. Valid options and functions are color-coded to disallow invalid selections.

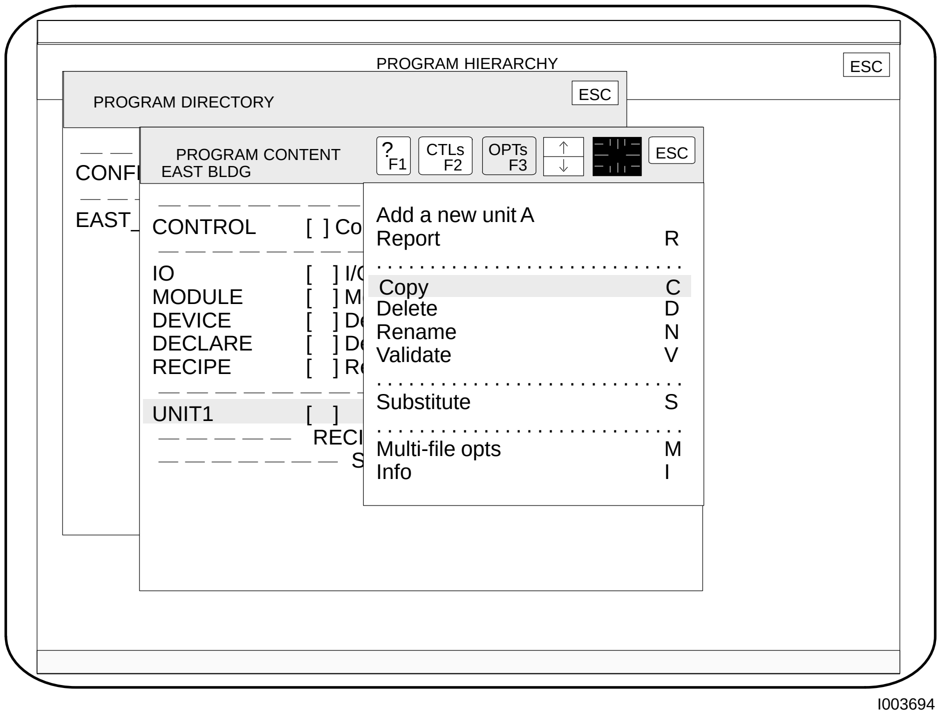


Figure 5-10 Pull-down Option/Function Menus

Programming in APT

APT supports two main languages for programming and defining solutions to process control problems—sequential function language and continuous function language.

Sequential Function Chart (SFC) Programming

A Sequential Function Chart (SFC) is a graphical representation of a state-oriented control problem. The SFC represents a module or collection of the processing actions within the total program. Each block in the SFC represents a state of operation in the control process. The APT SFC editor provides 10 drawing icons for creating the various SFC structures.

Following each step is a transition condition.

APT also provides a state command language used to specify the APT state control commands for each step of a SFC.

Safe-state Processing

Safe-state SFCs allow the process/control engineer to handle exceptions that occur during a normal process (see Figure 5-11). User-defined conditions stop processing of the current SFC and transfer control to a new specified SFC. For example, safe-state SFCs can be used to handle emergency shutdowns or any level of equipment failures. Safe-state SFCs also have priorities. The priority assignment designates the order of safe-state execution. The safe-state SFC trigger with the highest priority always takes precedence and will stop the processing of any lower-level safe-state SFCs (see Figure 5-12).

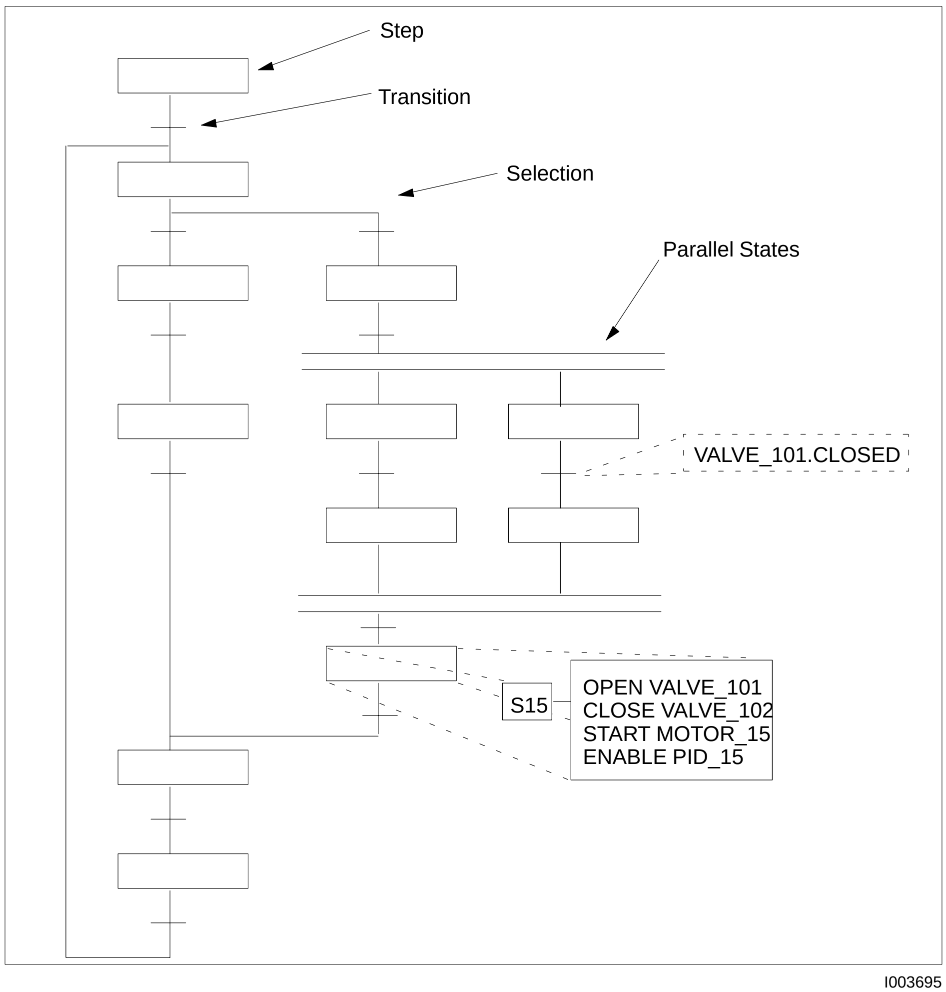


Figure 5-11 Sequential Function Chart (SFC)

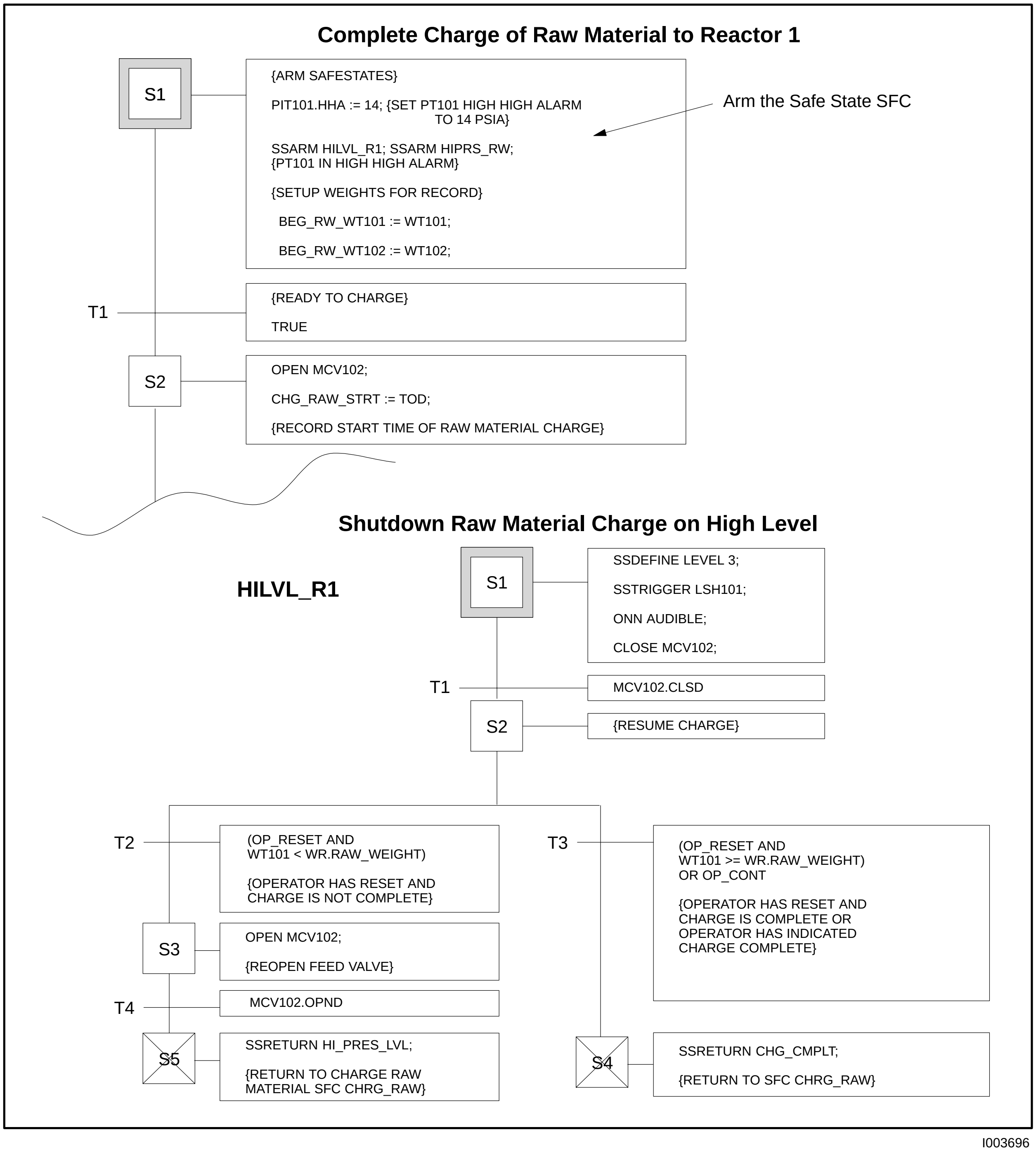
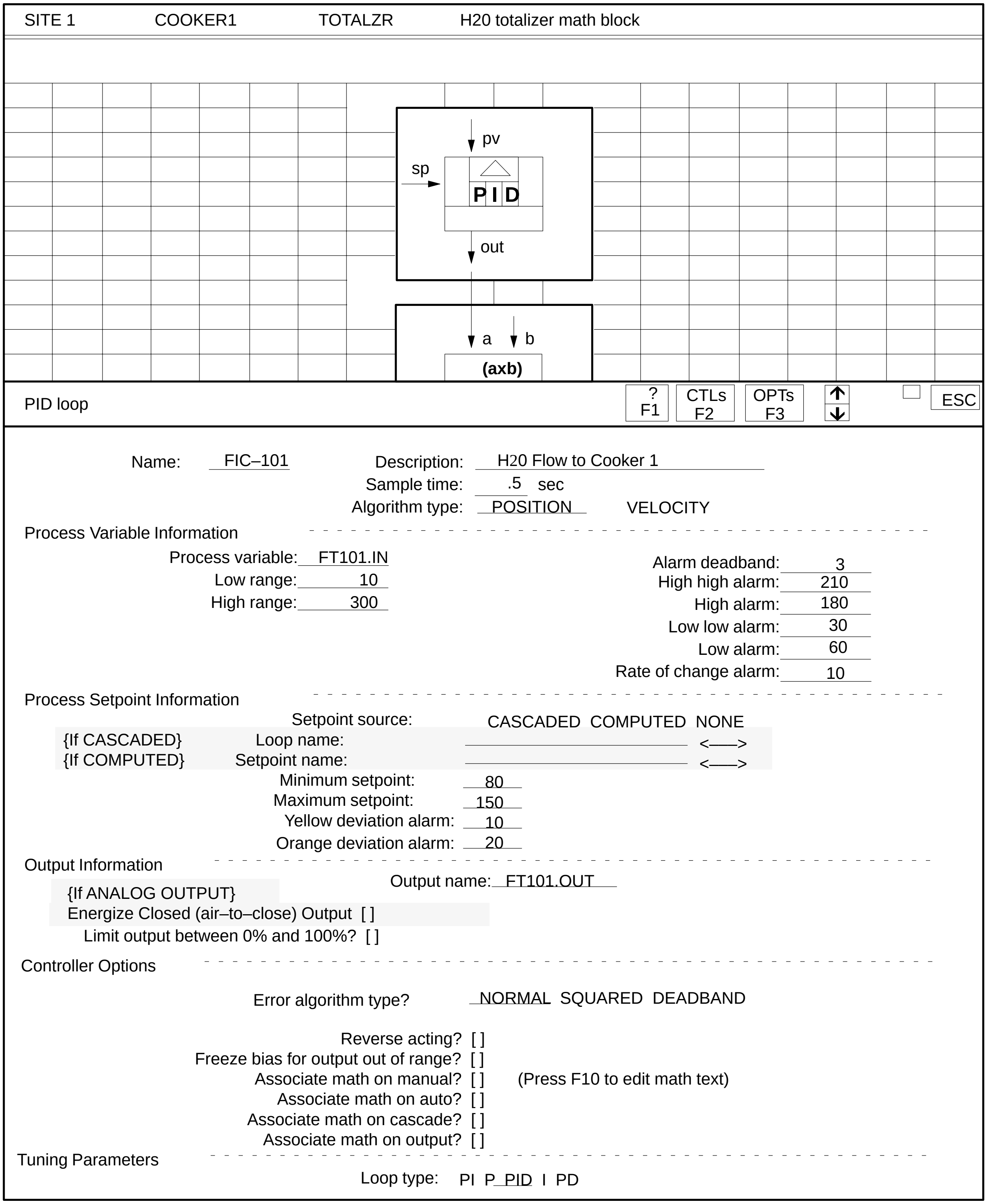


Figure 5-12 Safe-State Processing

Continuous Function Chart (CFC) Programming

A Continuous Function Chart (CFC) is a graphical representation of continuous control strategies (PID loops, physical flow transmitters) used in control problems. CFC function blocks are enabled and disabled by the Sequential Function Charts. Some typical CFC function blocks follow:

- Simple (PI loop, P loop, PID loop, etc.)
- Advanced (cascade, dead-time compensator, etc.)
- Limiter (output limiter, rate limiter,etc.)
- Selector (average select, high select, low select, etc.)
- Valve (motor position control, proportional time control, split range, etc.)
- Math (absolute value, calculator, divider, etc.)
- Dynamic (continuous dead-time delay, derivative, discrete dead-time delay, etc.)
- Other (analog alarm, correlated lookup table, etc.)



I003697

Figure 5-13 Continuous Function Chart (CFC)

Math Language

The math function block is a specialized CFC function block without predetermined internal specifications. It allows the creation of function blocks that meet application-specific requirements (see Figure 5-14). The math block has an associated form with a free-format text field for entering user-defined calculations. Math Blocks may also be used in SFC Steps. In SFCs the logic executes only while the step is active.

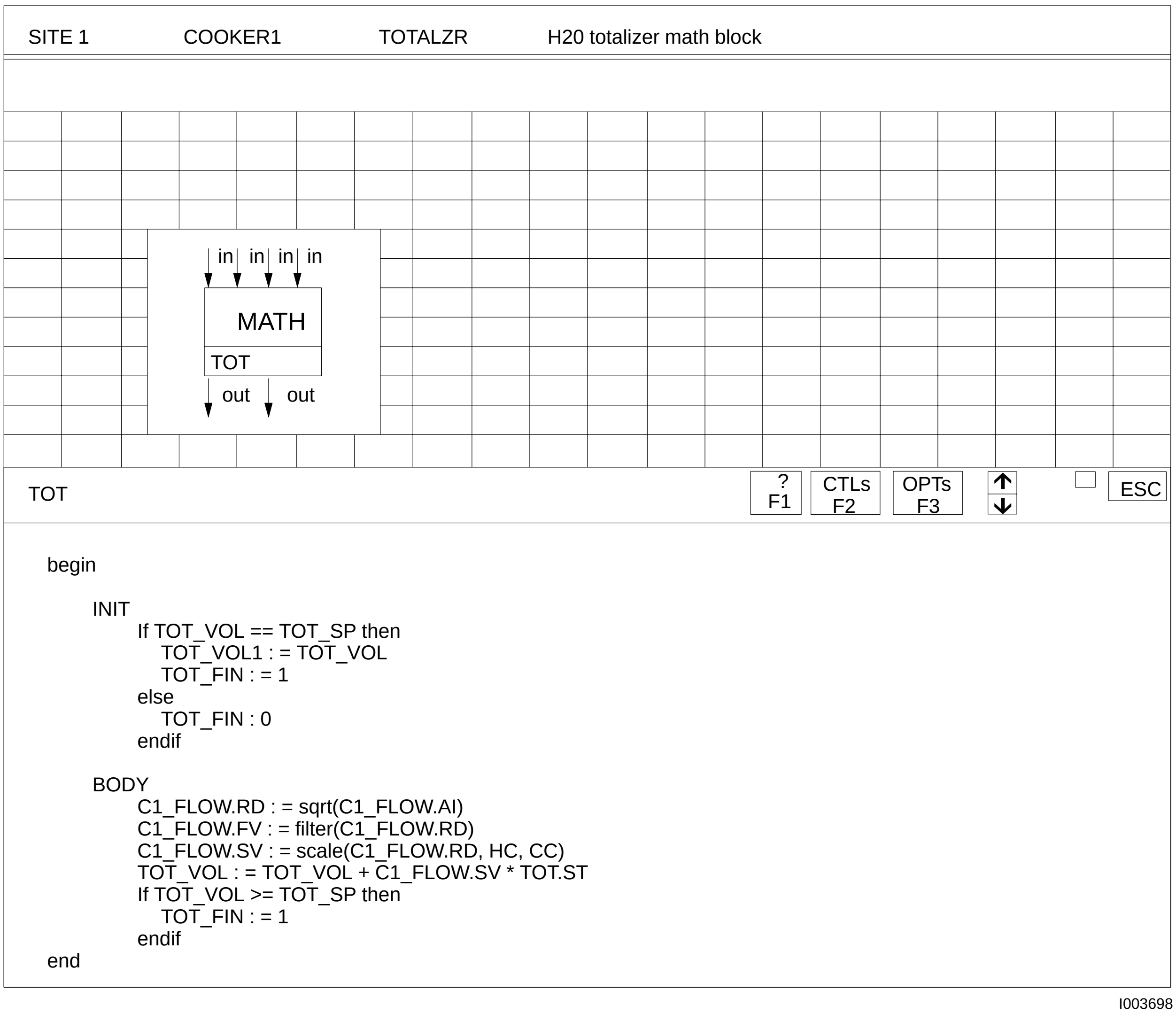


Figure 5-14 Math Function Block/Language

Graphical Debug Tools

APT provides graphical tools for debugging control programs. Typical debug tools include the following.

- Animate—allows you to watch your program execute. Each SFC step is displayed as it becomes active.
- Step-logger—similar to a tape recorder: it logs every step during the execution of an SFC as it becomes active. Step-logger also allows you to stop the record of your program and rewind it back to any point. A reset option also allows you to erase previously logged information and start over.
- Breakpoint—allows you to manually interrupt the transition flow between states in a SFC program so you can stop your program at specified intervals.
- Activate—allows you to activate an individual SFC or CFC function block in a program.
- Sample—polls the controller and displays the values of a user-specified group of variables.

MAITT

MAITT is a verification tool that can communicate directly to a controller to simulate inputs and verify program outputs. It has been added to the APT environment to give the user the ability to verify APT programs after they have been compiled and downloaded.

MAITT can help to answer questions like:

- Is my control logic correct for this application?
- Will my program properly shut down the flow pump when the pipeline valve fails to open?
- Are my I/O modules wired correctly?
- Did the change I made to my control program really work as I wanted or did it corrupt other parts of the program?

With MAITT, the user writes a program with a series of simple, English-like instructions.

Chapter 6
TIWAY Networking

6.1 Description

Industrial Local Area Network

TIWAY is an industrial Local Area Network (LAN) designed to satisfy today's industrial automation requirements. TIWAY has successful installations in all types of industrial environments, including discrete, batch, and process control areas. The TIWAY network provides a convenient method of obtaining, evaluating, modifying, and replacing data stored in the separate program memories of interconnected Siemens controllers. TIWAY also provides an interface to non-Siemens devices such as robots, Computer Numerically Controlled (CNC) machines, and instrumentation systems.

For more information about networking, consult the appropriate networking manual.

Components

TIWAY system components consist of:

- Network Primaries (UNILINK Host Adapters and Gateways)
- Network Secondaries (Network Interface Modules and UNILINK Secondary Adapters)
- Network media
- Host Resident software packages

Hosted Network

TIWAY is a "hosted" network. This means the primary (UNILINK Host Adapter, configured as the Master Host Interface) is the network manager and controls communication to the 254 available nodes. These nodes may be any combination of other primaries and secondaries. Standard host software packages enable network primaries to gather, analyze, and control data from any other network secondary.

Figure 6-1 shows these components in a simplified network scenario.

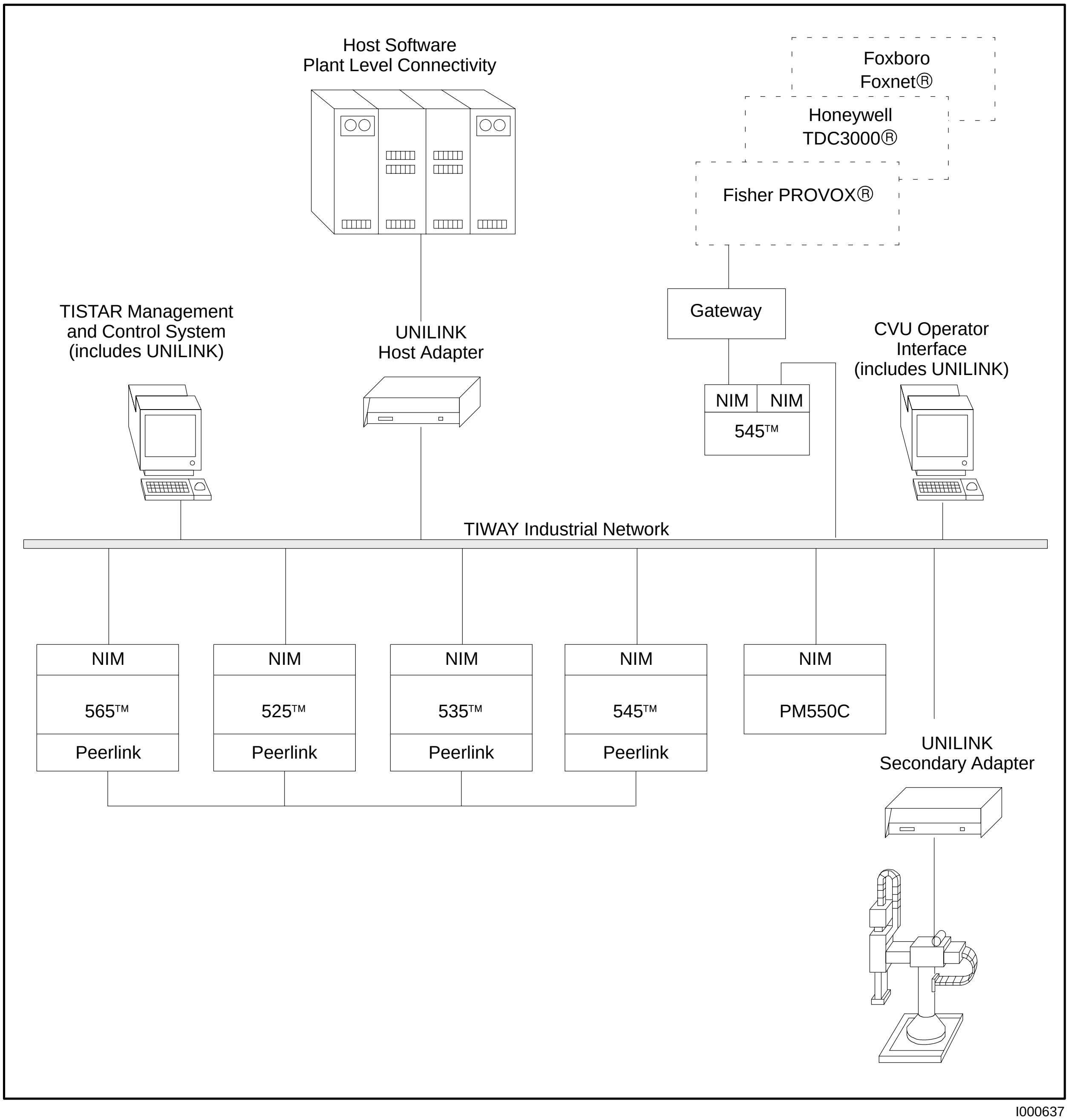


Figure 6-1 TIWAY Components

Network Architecture

TIWAY is a multi-drop, bus topology that supports Local Line™ and/or modem communication. Local Line geographic coverage is dependent on cable selection, number of secondary nodes, baud rate selection, and node placement. Modems can be attached (via RS-233-C), with the coverage being dependent on your selection of modem.

Universal Command Language

A key element of the TIWAY network is the Universal Command Language (UCL). Functional control of the secondaries (controllers and UNILINK Secondary Adapters) is performed with UCL standard command codes. The UCL consists of high-level commands that are recognized and interpreted by the complete Siemens line of controllers. TIWAY uses the industry standard High-level Data Link Control (HDLC) protocol. Excellent data throughput rates are realized by use of Siemens Universal Command Language and a data transfer rate of 115.2K bps. This is in addition to exceptional data accuracy, with an undetected bit error rate of 6×10^{-13} .

By using a standard “language,” two benefits are realized.

- Communication is simplified since more information can be accessed in a standard format in a given period of time.
- The host computer programmer can now treat all secondaries as logical devices.

This means that the same program can be used to communicate with any controller without treating each controller type uniquely. To perform an operation on the controller, the host program simply supplies the controller address and the UCL command to be executed.

For example, the registers in the 545™ and 525™ can be accessed with exactly the same command, regardless of the fact that the capabilities of the controller are considerably different. Only a subset of UCL commands need to change for each type of controller. This approach provides software flexibility and reduces system reconfiguration and expansion.

6.2 UNILINK Host Adapter

The UNILINK Host Adapter (PPX:500–6224) allows factory-floor and mainframe computers to access Siemens equipment as well as other vendor's products that are linked to the TIWAY network. Figure 6-2 shows the functional area that utilizes the UNILINK Host Adapter.

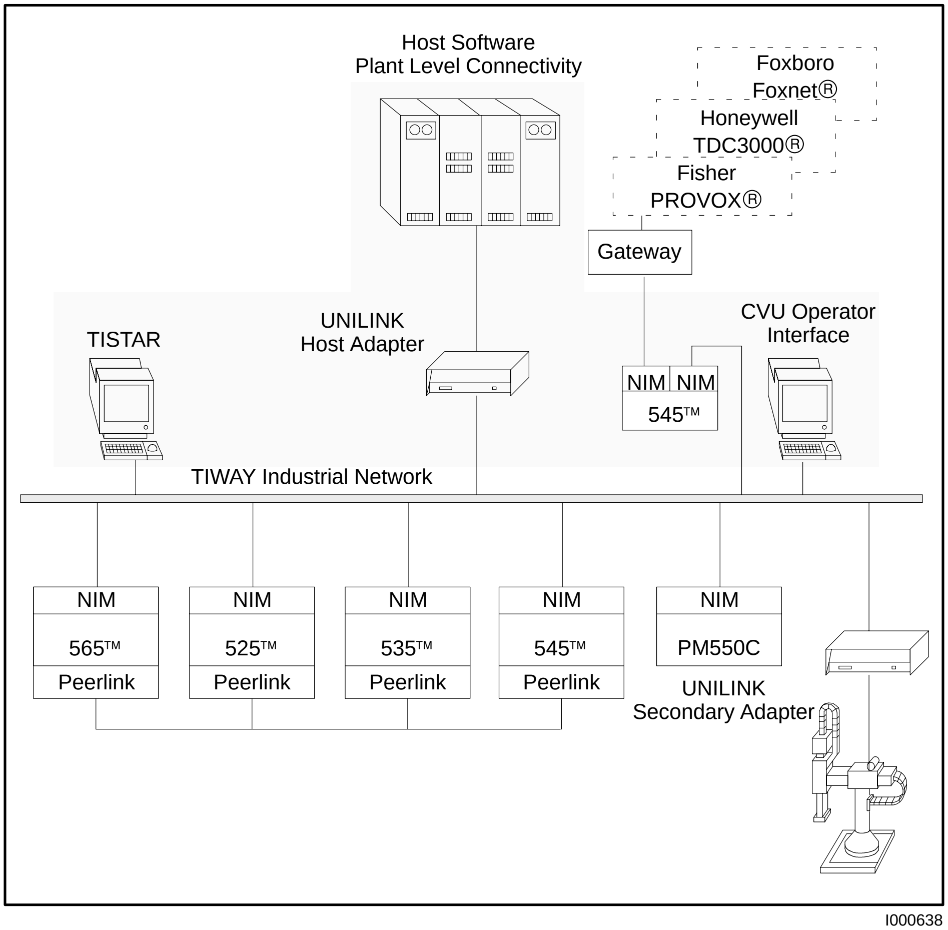


Figure 6-2 UNILINK Host Adapter

The adapter communicates with a wide range of host computers through a serial binary data interface using RS-232-C control signals and RS-423 drivers. The communication port is configurable for baud rates up to 38.4K bps, synchronous or asynchronous communication, full- or half-duplex, with even, odd, or no parity. The adapter supports both a non-intelligent terminal (NITP) and a BDLC protocol. The BDLC protocol is a byte-oriented subset of the ANSI X3.66 standard protocol.

6.3 Host Software Programming Packages

Prepackaged configurable systems such as the CVU Operator Interface, TISTAR systems typically provide TIWAY support as part of the package. However, in many cases you may want to implement application specific functions outside of a prepackaged environment. To reduce application software development cost and complexity, Siemens offers a series of TIWAY communication support packages.

TIWAY host software packages contain a set of tools to help the application developer. TIWAY device drivers, communications utilities, and a high-level language interface, allow the developer to work on the application without having to spend time developing the lower-level communication software.

The high-level language interface supplies TIWAY access via convenient subroutine calls. For example, the TIGET subroutine provides an easy way to get data from a 545™. The application developer calls the subroutine and supplies parameters that specify the controller data location, the length of the data block, and the array name that will contain the returned data. When control returns from the subroutine, the controller data has been moved into the array and is ready for use.

There are 4 models of UNILINK Host Adapters.

- 115 VAC, RS-232-C in, local line out (PPX:505–7111)
- 115 VAC, RS-232-C in and out (PPX:505–7112)
- 230 VAC, RS-232-C in, local line out (PPX:505–7113)
- 230 VAC, RS-232-C in and out (PPX:505–7114)

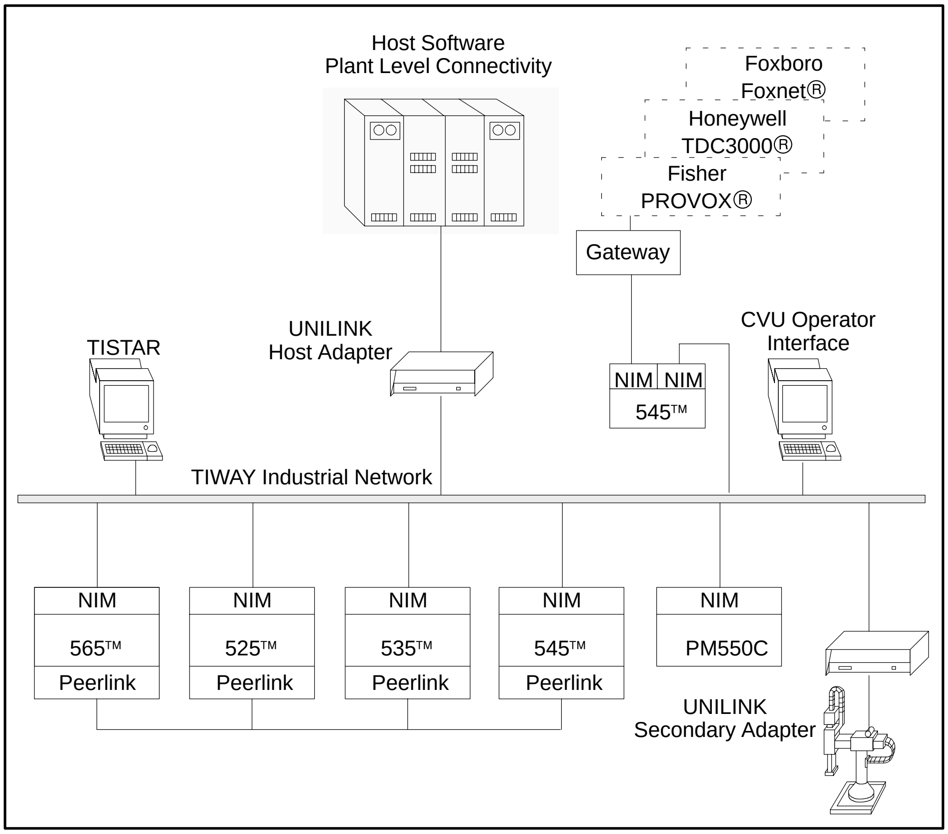


Figure 6-3 Host Software Packages

Siemens offers TIWAY Host Software packages to support DEC™ VAX™/MicroVAX™ (VMS™ 4.x, 5.x), DEC (RSX–11), and IBM PCs and compatibles (MS–DOS®). TIWAY support is also available through third-party vendors for other popular host computer systems.

6.4 Network Interface Module

The Series 505 Network Interface Module (NIM) provides an interface from a local or remote base to the TIWAY network. Series 505 NIMs accept high-level commands generated by a network primary and convert these commands to corresponding task codes which communicate with the controller. Figure 6-4 shows the functional area that utilizes the NIM.

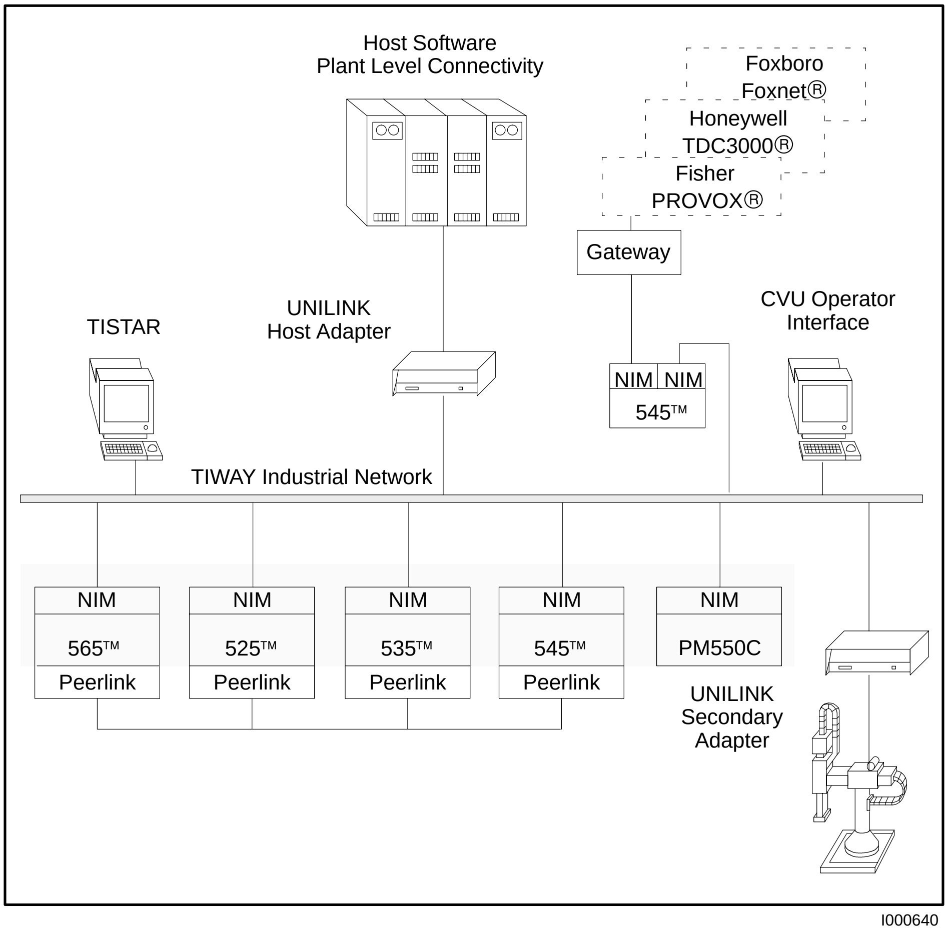


Figure 6-4 Network Interface Module

There are 2 models of 505™ NIMs.

- Dual local line version (PPX:505–7339)
- Dual RS–232–C version (PPX:505–7340)

6.5 UNILINK Secondary Adapters

The UNILINK Secondary Adapter allows many previously incompatible factory-floor devices (robots, CNC machines, and instrumentation systems) to be linked to the TIWAY network. One of the biggest features of the adapter is the Personality Interface Module (PIM). The PIM is a small plug-in module that translates the TIWAY commands into the intelligent device specific commands. Figure 6-5 shows the functional area that utilizes the UNILINK Secondary Adapter.

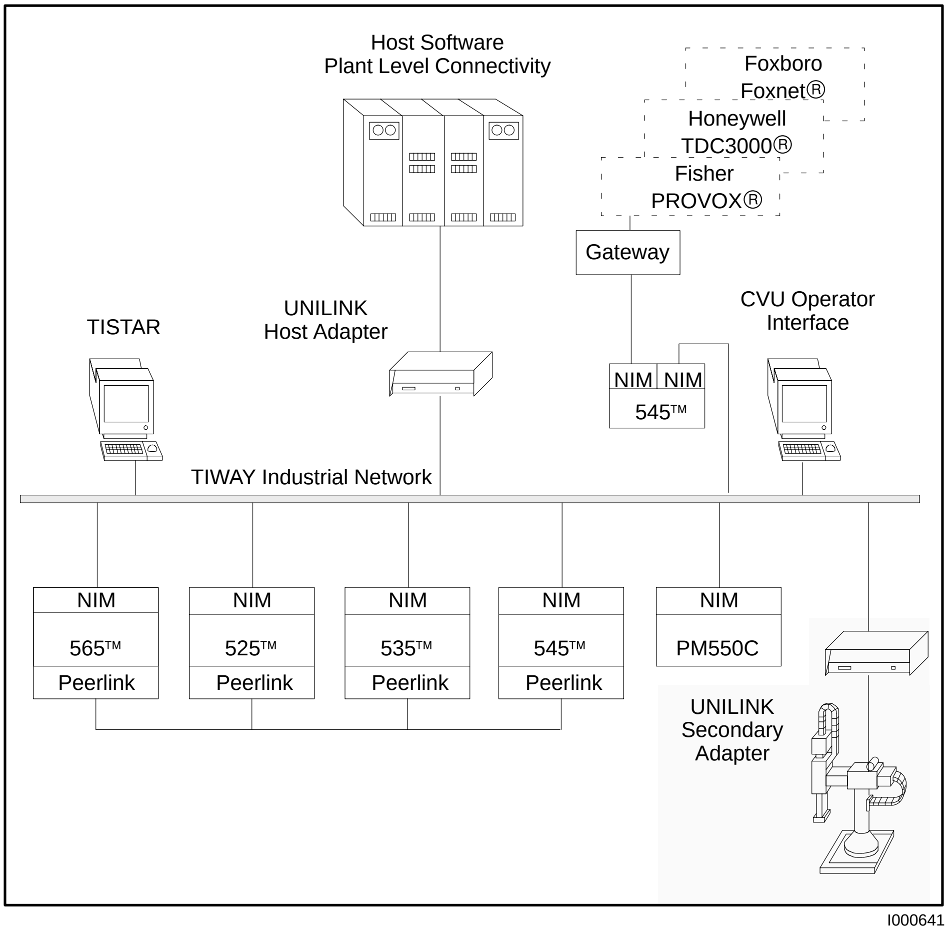


Figure 6-5 UNILINK Secondary Adapter

6.6 Gateways

The TIWAY I Gateway provides an interface between the SIMATIC TIWAY I network and a distributed control system host using Modbus® protocol. Figure 6-6 shows the relationship of the Gateway to one of several possible Modbus-compatible host systems.

The TIWAY I Gateway translates Modbus commands from a host into the TIWAY I protocol format. Since the Modbus and TIWAY I systems are entirely different in protocol and interface requirements, the Gateway serves as a protocol translator and as a type of network monitor.

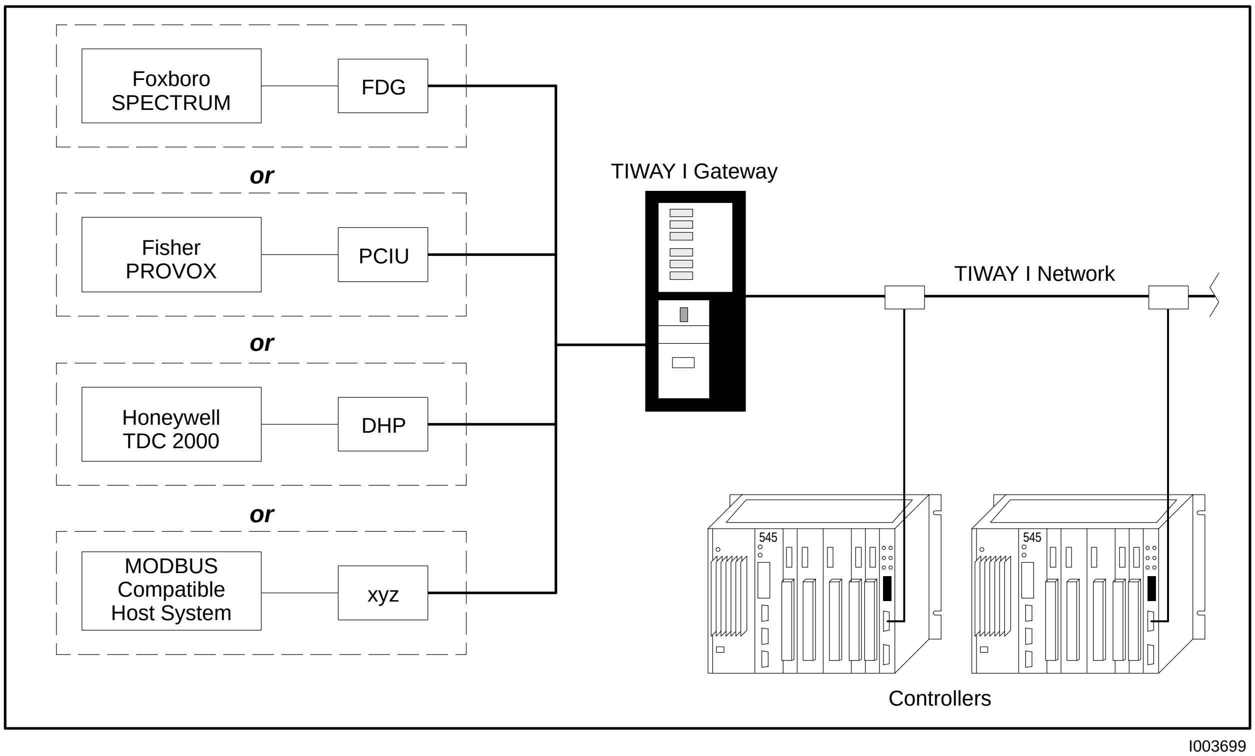


Figure 6-6 TIWAY Gateways

Host Systems

The TIWAY I Gateway can provide protocol translation for the following host systems. The commands supported are described in Table 6-1.

- The Honeywell TDC 2000® Data Highway Port (DHP) with one Gateway can monitor up to eight controller stations with one Network Interface Module (NIM) for each controller.
- The Foxboro SPECTRUM™ FOXNET® Device Interface (FDG) with one Gateway can monitor up to 64 stations equipped with NIMs.
- The Fisher PROVOX® Programmable Controller Interface Unit (PCIU) can monitor up to 8 stations with NIMs.

Table 6-1 Commands Supported by TIWAY I Gateway

Code	Type	Description
01	Read Coil Status	Get current status of a group of coils
02	Read Input Status	Get current status of a group of discrete inputs
03	Read Holding Register	Get current values from holding registers
04	Read Input Register	Get current values from input registers
05	Force a Single Coil	Change the stat of a logic coil to on or off, forced on or forced off, or unforced.
06	Write a Single Register	Write a value into a holding register
07	Read Exception Status	Get the 8 internal status coil values
08	Execute Diagnostics	Send diagnostic tests to a slave
11	Comms. Even Counter	Enable the success or failure of a query to be determined
12	Get Comms. Log	Get the communications log for Modbus network transactions
15	Write Multiple Coils	Change a number of consecutive coils
16	Write Multiple Registers	Write values into a series of consecutive holding registers

Gateways (continued)

Models

The TIWAY I Gateway is available in four models, offering a choice of communication ports and voltage supplies, as listed in Table 6-2.

Table 6-2 TIWAY I Gateway Models

Model Number	Communication Ports	Supply Voltage
PPX:500–7301	RS-232-C/Local Line	120 VAC
PPX:500–7302	Dual RS-232-C	120 VAC
PPX:500–7303	RS-232-C/Local Line	240 VAC
PPX:500–7304	Dual RS-232-C	240 VAC

Data Transmission Rates Supported

The TIWAY I Gateway supports data transmission rates from 110 bits per second (bps) to 19.2K bps with the host, and from 110 to 115.2K bps on the network interface.

Types of Data Accessed

The Gateway allows access to 1K discrete I/O points (X and Y), 28K holding registers (V-memory), 1K word input and output registers (WX and WY), as well as performance statistics for each controller network interface. Information can be stored in variable (V) memory locations, retrieved, and changed from the operator’s console of the distributed control system.

6.7 Peerlink

Peerlink is a local area network that provides an inexpensive and easily implemented solution for small controller networks. Typically, a major factor in this application is rapid, frequent transfer of data between controllers. By using an HDLC-like protocol for data integrity and rapid transmission, the Peerlink network can transmit 16 words to 16 controllers in less than 75 ms at maximum load. The Peerlink network can operate at distances up to 10,000 feet. Figure 6-7 shows the function areas that utilize Peerlink communication.

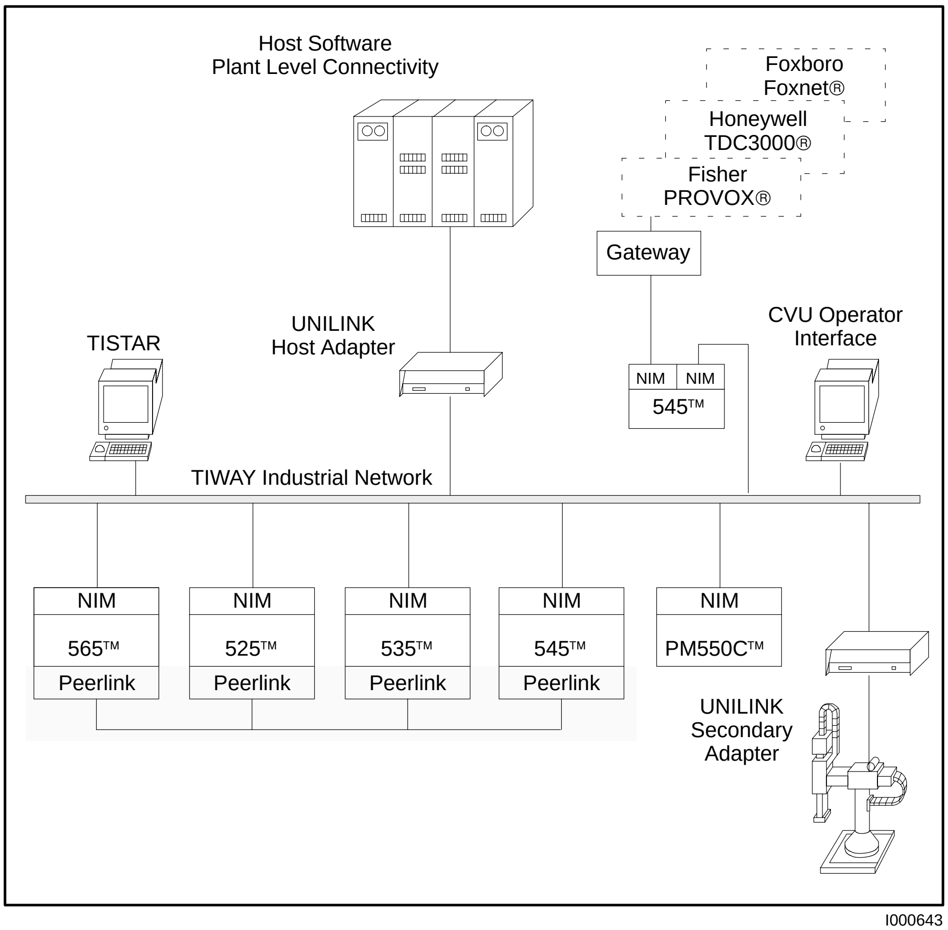


Figure 6-7 Peerlink

Peerlink is transparent to your applications program and transmits the data directly to the controller variable memory. No programming is necessary, since you only have to define the starting address. Peerlink provides two fault tolerance methods; media redundancy, or active/passive monitoring.

Chapter 7

Direct-Connecting Operator Interface

7.1 Overview

Factory-floor Operation

The CVU100 is a factory-floor operator interface designed to communicate with any of the Series 505 or Series 500 controllers. The CVU100 can replace pushbuttons, pilot lights, and bar indicators, as well as function as a loop access interface. Shown in Figure 7-1, the CVU100 has an integral 35 position keypad, 9-inch amber CRT, and meets NEMA 4X/12 standards as tested by UL.

The CVU100 has a built-in macro programming language that supports fast generation of operator displays for even the most complex control applications. To program the CVU100, you use any standard IBM PC/XT compatible keyboard.

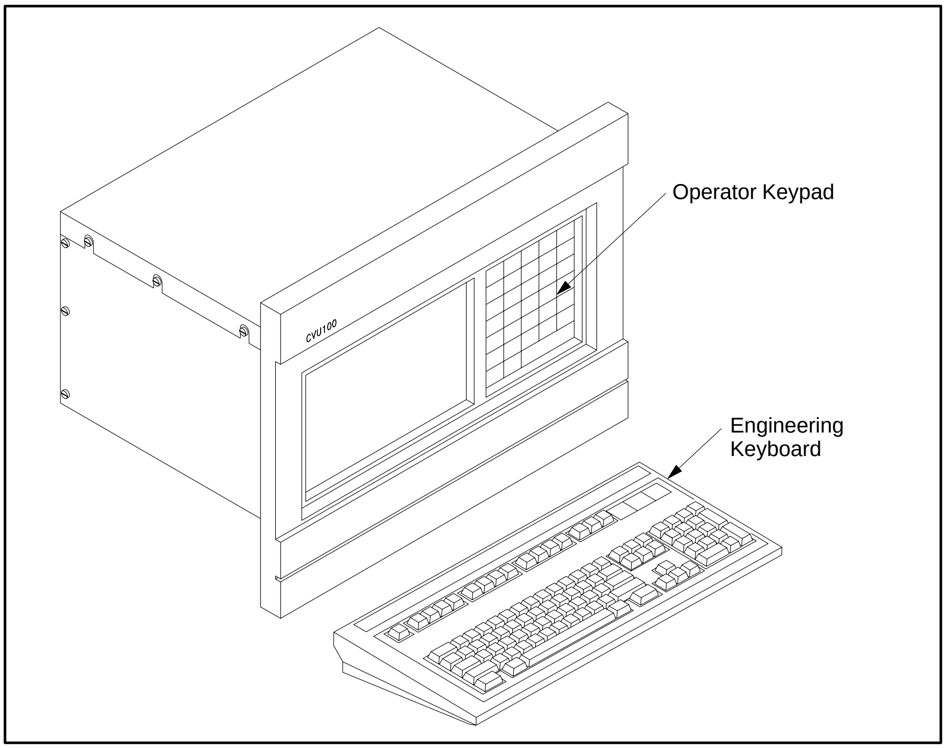


Figure 7-1 CVU100 System

Communication with Controllers

The CVU100 reads and writes to the controller variable memory and control relay registers. The CVU100 is fully compatible with all memory types supported by the SIMATIC 545 and SIMATIC 565 controllers. Information is shared by both the controller and the operator interface without additional programming in the controller. The CVU100 communicates via any of the RS-232 ports available on the Series 505 and Series 500 controllers or Remote Base Controllers (RBC). Figure 7-2 illustrates how the CVU100 connects to the RBC. Figure 7-3 and Figure 7-4 show the pin-outs for connecting the CVU100 to the SIMATIC 565 and SIMATIC 545.

The CVU100 can initiate control actions or monitor the status of an input and display the status or value in the form of a discrete or analog message, bar graph, or flashing symbol.

For more comprehensive information consult the CVU100 manual.

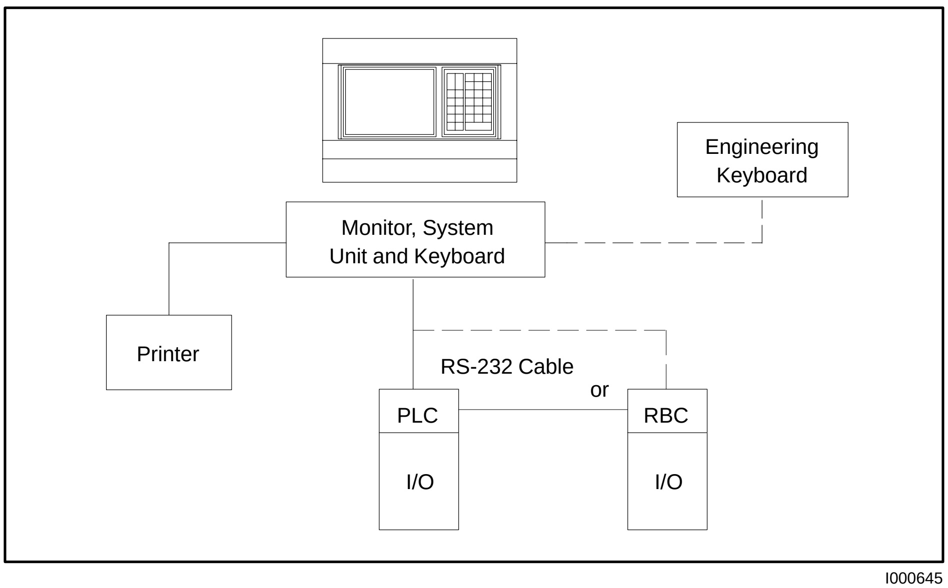
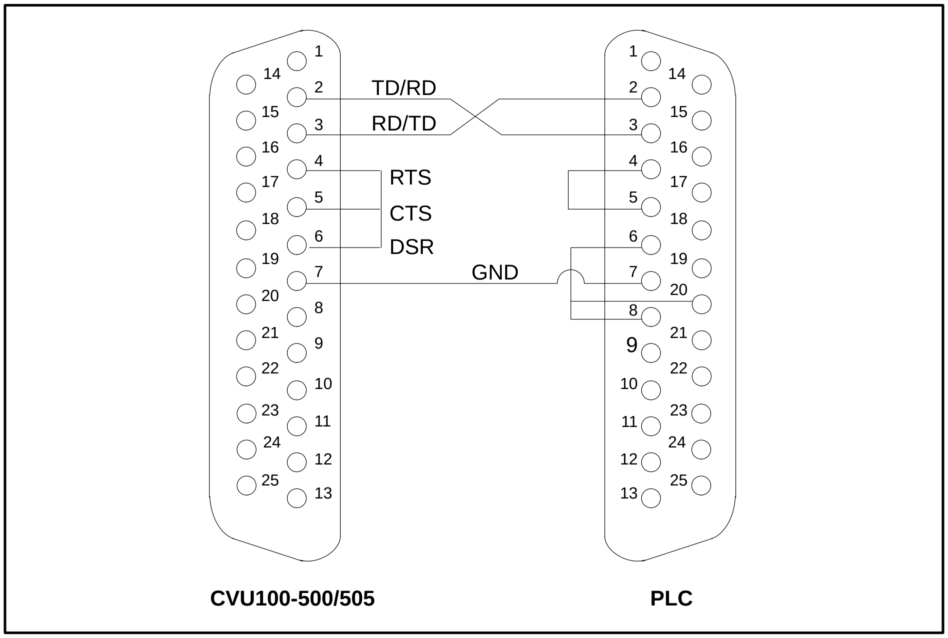
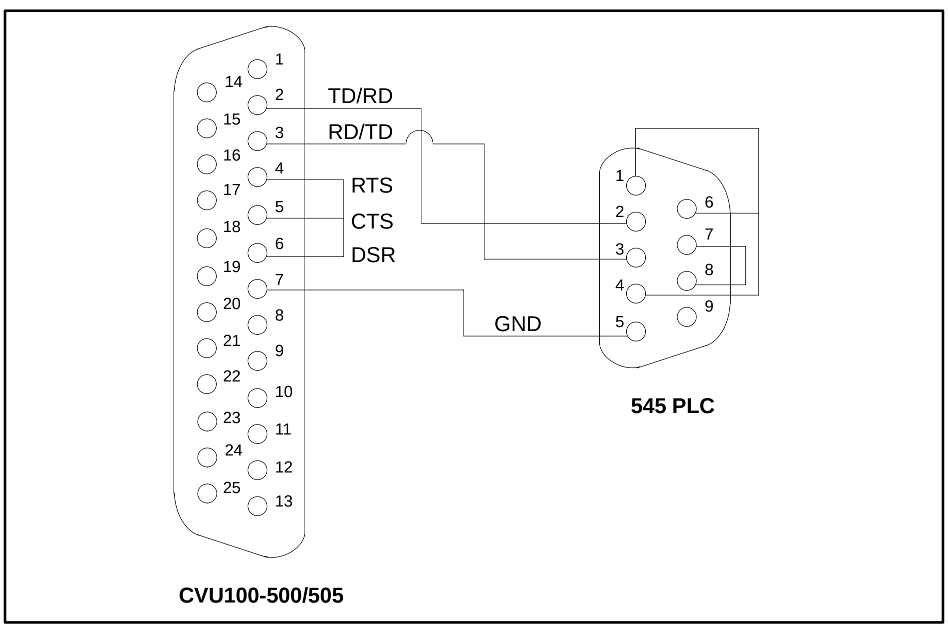


Figure 7-2 CVU100 Connected through RBC



I000646

Figure 7-3 Series 500 & 505 PLC/CVU100 Wiring Diagram (except SIMATIC 545)



I000647

Figure 7-4 545 PLC/CVU100 Wiring Diagram

7.2 Macro Configuration

The CVU100 uses macro configuration to develop screen programs. A single screen program can be used to format several displays, or you can use several smaller programs to develop a single display. The 55K bytes of battery-backed screen memory provides storage for up to 255 screen programs.

A program macro can initiate block reads of data from the programmable controller, write data entered from the keypad or from a value calculated by the CVU100, or the controller can initiate reads or writes using a program activated by a control bit.

The commands shown in Table 7-1 are used to read and write data to the controller.

Table 7-1 Commands Used to Read/Write Data to the Controller

Command	Description
GET	Read a single memory location
GETIO	Read an I/O point (X, Y, C)
GETMIO	Read multiple I/O points (X, Y, WX, WY)
GETR	Read up to 128 continuous memory locations
GETS	Read a loop value or analog alarm value from S-memory
PUT	Write to a single memory location
PUTIO	Write to an I/O point (C)
PUTR	Write values to a maximum of 128 memory locations
PUTS	Write to a loop variable or an analog alarm variable in S-memory

Using the preconfigured symbols for pumps, fans, valves, and other graphics functions, such as box, circle, plot and bar commands, allows generation of displays easily and quickly.

The CVU100 can also generate reports to a serial printer or write data to a personal computer via an RS-232 link.

7.3 Alarm Management

The CVU100 can be configured to monitor several alarm bits continuously and display the status on an alarm summary display. By using the GETIO command for each alarm point, you can use the status of the control point to present a flashing message to the operator or to activate the alarm status area of the prompt line. Figure 7-5 shows an alarm message example.

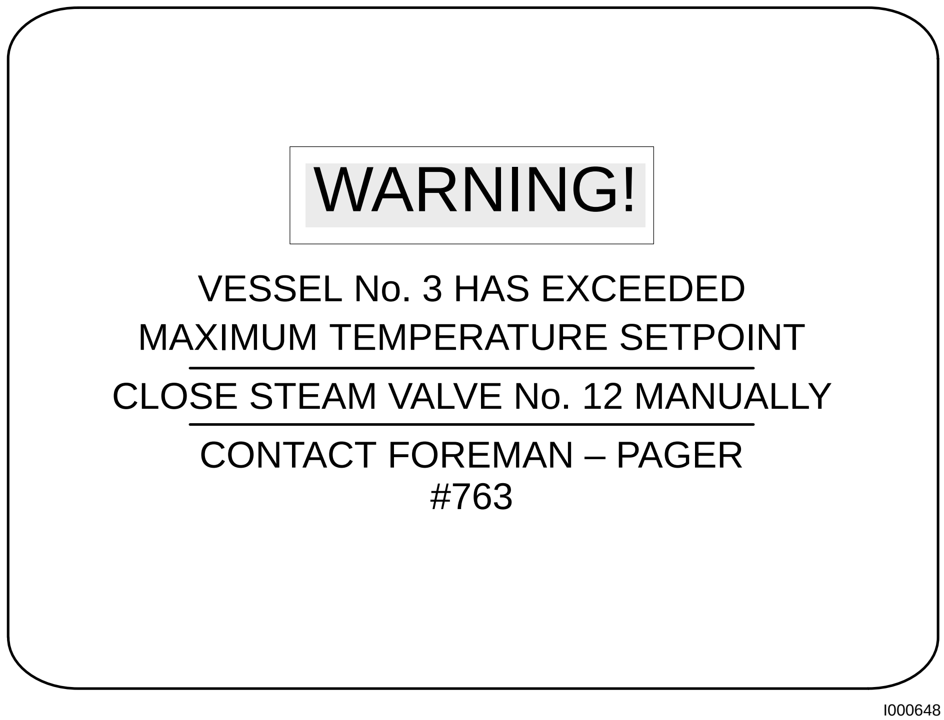


Figure 7-5 Alarm Message Example

7.4 Tuning Loops

The CVU100 has full access to S memory for tuning loops and analog alarms. This allows you to create your own tuning displays. From a single display, you can control the mode of a loop, change both the setpoint and output, or acknowledge loop alarms. Figure 7-6 illustrates the type of display you can create for each loop. The text side of the display can be used to give additional information about your particular application.

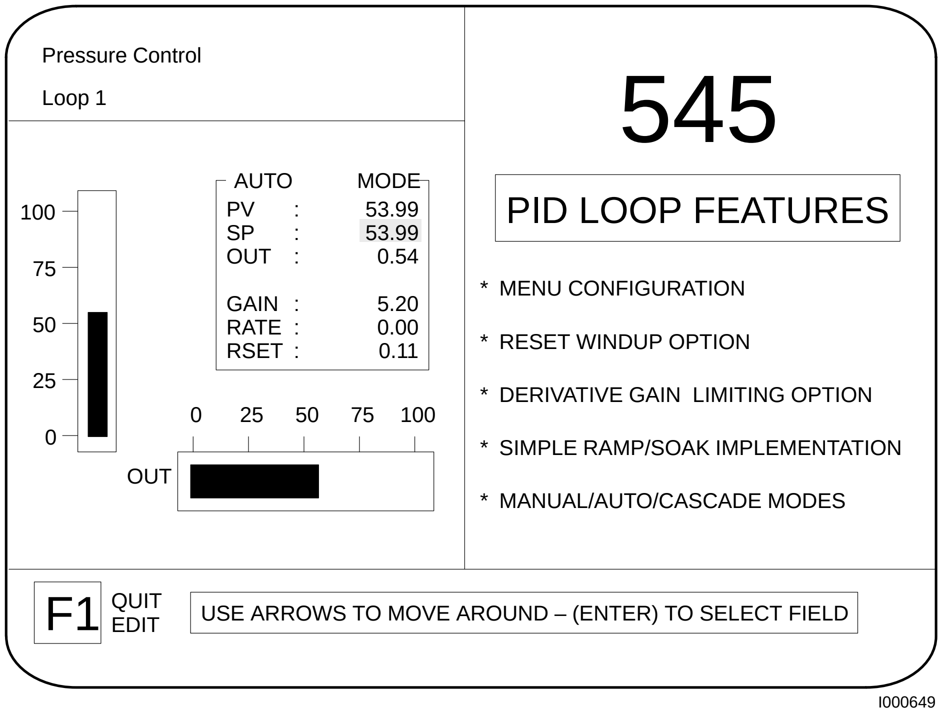
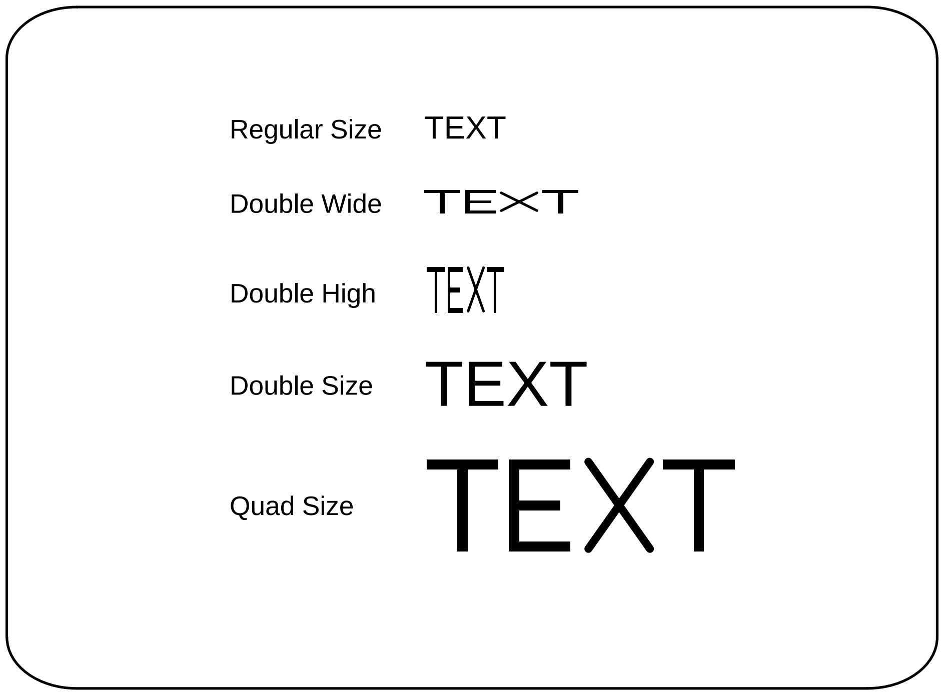


Figure 7-6 Loop Tuning Display

7.5 Monitoring the Control System

Display Pages and Messages

The display page is divided into 25 rows and 80 columns. The 25th row is designated as a status line and can remain the same for all operator displays. Text messages can be displayed using any of the five different text formats, as illustrated below in Figure 7-7.



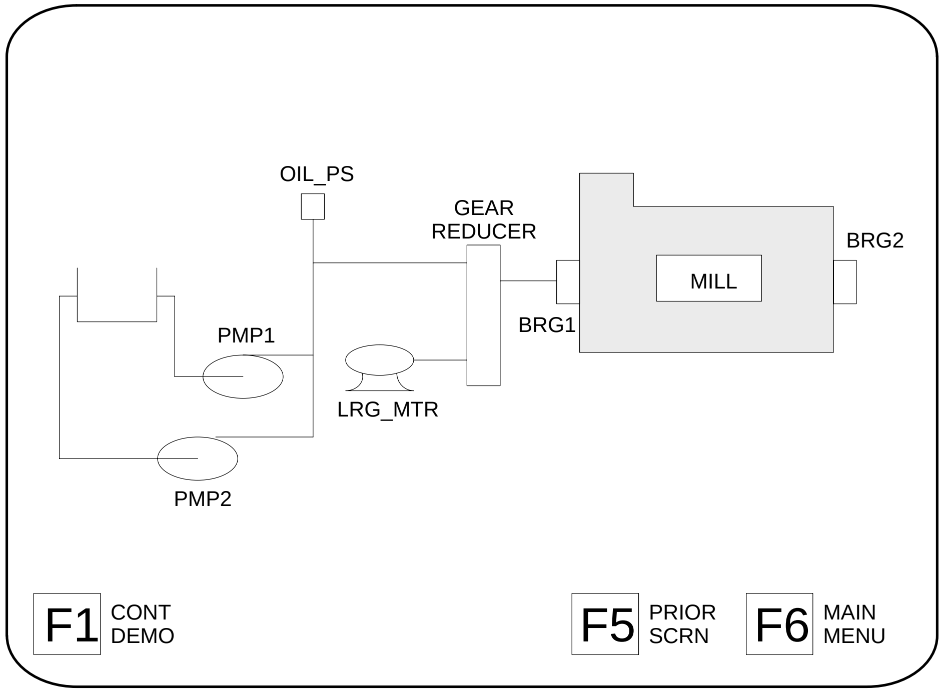
I000650

Figure 7-7 Text Formats

Text can be formatted to change status based on an I/O point. The message can be displayed as flashing text, changing intensity, or changing to reverse video.

Graphic Displays

The macro configuration utilities of the CVU100 are more memory efficient than standard basic commands for developing custom graphic displays. Figure 7-8 illustrates a typical application in graphic format.

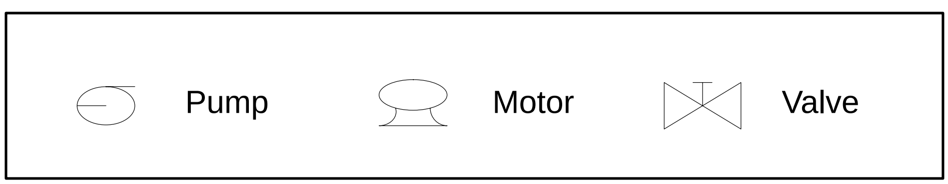


I000651

Figure 7-8 Typical Application Graphic

Designing Graphic Displays

The graphic macro commands support generation of complex graphic displays. The 91 graphic symbols include pumps, compressors, turbines, motors, tanks, valves, and box or circle utilities. The combination of these symbols with two line thickness reduces memory used per display and improves the response time for the operator. Figure 7-9 shows three of the symbols commonly-used in many applications.



I000652

Figure 7-9 Graphic Symbol Examples

Displaying Variable Data

Analog values can be displayed using data read directly from the controller. You can display the data as a scaled value using the math capability of the CVU100, as an animated bar graph using the vertical bar command, or as a trend plot using the plot command.

In Figure 7-10, for example, one set of data is displayed using each of the four different display methods.

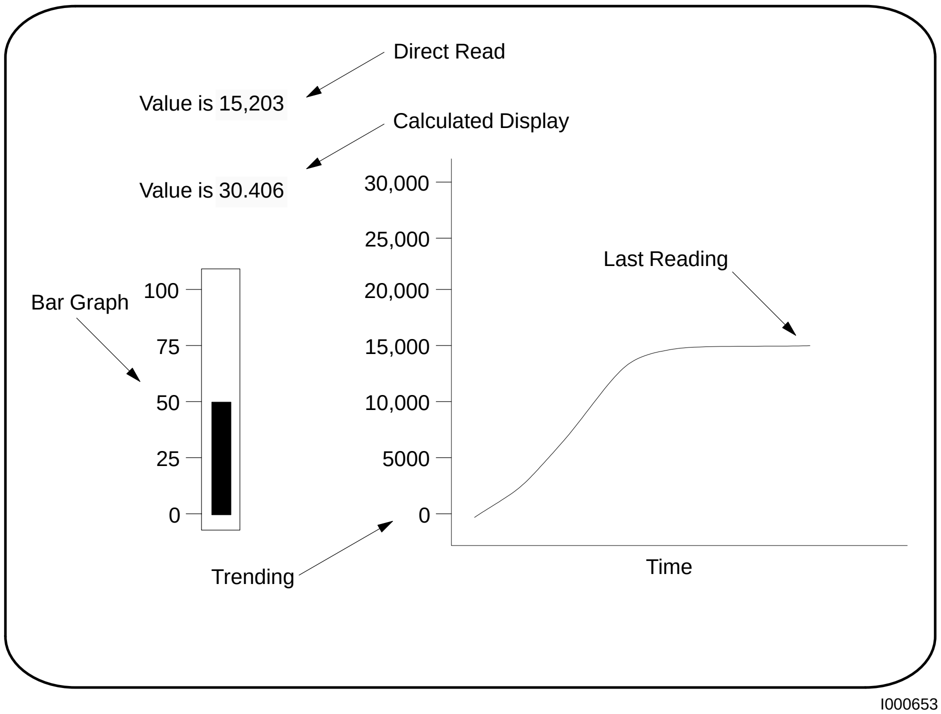


Figure 7-10 Variable Data Formats

7.6 Interfacing to the Control System

The 35-position sealed membrane keypad (Figure 7-11) has 14 function keys that can be used to initiate status changes of discrete I/O points, acknowledge alarms, change loop modes, or position the cursor on an analog value.

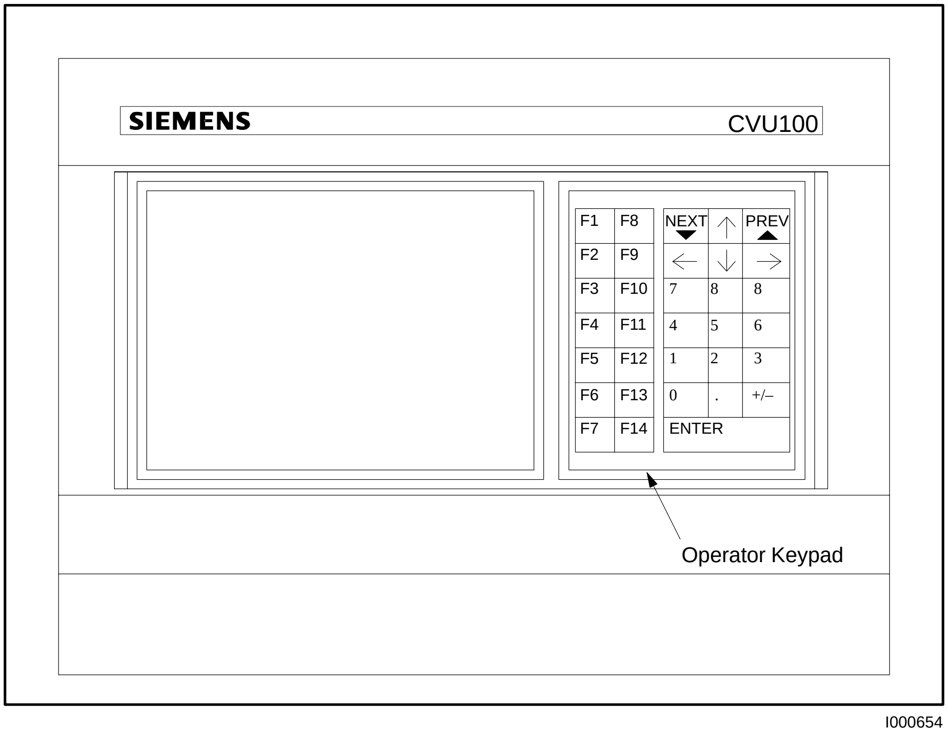


Figure 7-11 CVU100 Keypad

7.7 CVU100 Specifications

Table 7-2 CVU100 Specifications

Characteristic	Specification
Dimensions: Height Width Depth	 12.24 inches (311 mm) 16.47 inches (418 mm) 11.24 inches (286 mm)
Weight	21.0 lbs. (9.52 kg)
Mounting	Panel or rack (requires filler adapters)
Power requirements	115 (90–132) VAC at 47–63 Hz 230 (180–264) VAC at 47–63 Hz
Power consumption	37 watts
Temperature	Operating Conditions 0° to 50° C (32° to 122° F) Storage Conditions –40° to 65° C (–40° to 149° F)
Humidity	5 to 80% relative, non-condensing
Vibration (5 to 2kHz)	Operating Conditions 0.006 inches peak-to-peak, 1.0g maximum Storage Conditions 0.015 inches peak-to-peak, 2.5g maximum
Agency listings	UL® listed, CSA certified (pending)

The CVU100 is equipped to withstand most harsh environments found on the factory floor. However, you can install it in a 19-inch National Electrical Manufacturers Association (NEMA) cabinet. Figure 7-12 shows the mounting dimensions.

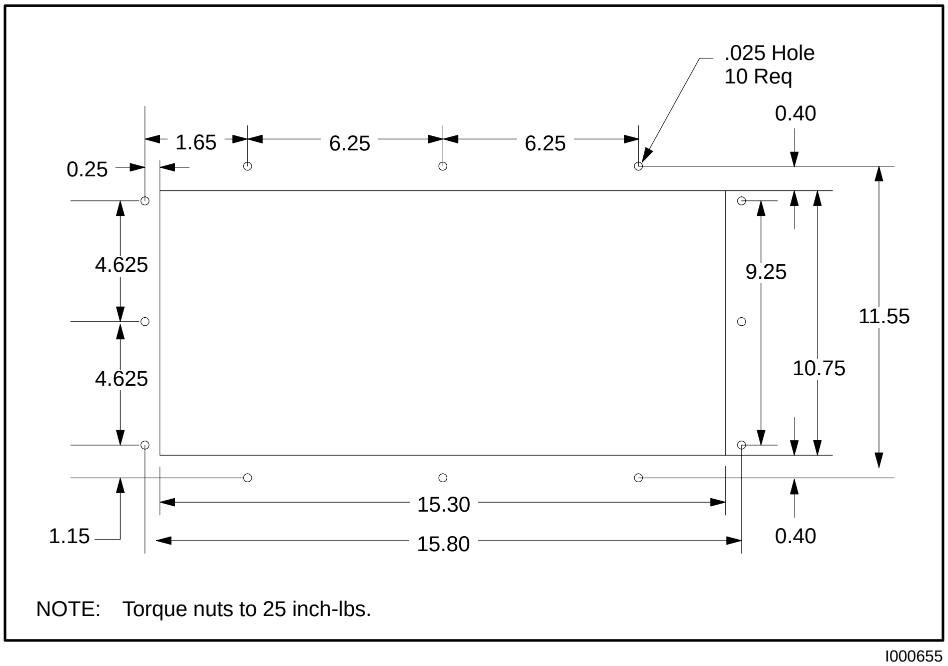


Figure 7-12 CVU100 Dimensions

Chapter 8

Networkable Operator Interfaces

8.1 Overview

Control Room or Factory Floor

The CVU product line is a family of high-performance, color-graphic, operator interfaces designed for use with SIMATIC 505 and SIMATIC 500 controllers. The CVU10000 is designed for control room environments, while the CVU1000 is targeted for the harsh environments found on the factory floor. The CVU software can also be installed on a 386/ATM Coprocessor. The 386/ATM coprocessor is a high-speed computer in an I/O module form factor. All of these units are IBM-compatible and offer extensive control capabilities. See Figure 8-1.

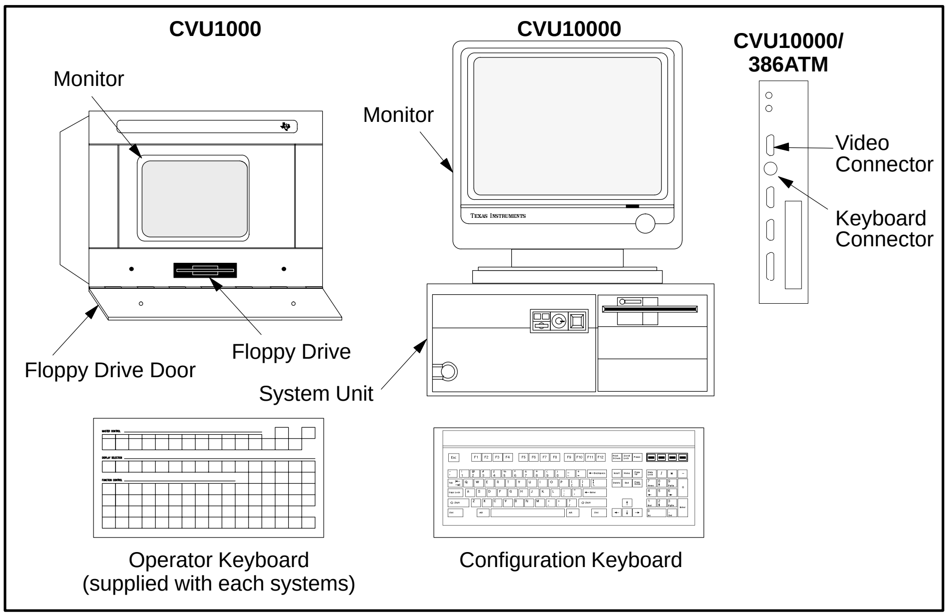


Figure 8-1 The CVU System

The CVU can replace the lights, dials, switches, buttons, and annunciators that the operator normally uses to observe and control the application. As shown in Figure 8-1, the CVU10000 includes a system unit with an AT-style keyboard, 14- or 19-inch monitor (optional), a hardened operator keyboard, and all necessary software. The CVU1000 includes the system unit with enclosed monitor, a hardened operator keyboard, and all necessary software. The CVU1000 can use most any AT-style keyboard for configuration.

For more comprehensive information, consult the appropriate CVU manual.

Reads and Writes Controller Variables

The CVU reads and writes to the controller variable memory and input/output image registers. In this way, information is shared by the CVU and the controller. The controller RLL program may receive new setpoint values or other instructions from the operator through the operator keyboard or via a new recipe. The CVU can read information generated by the RLL and display it in the variety of ways. Figure 8-2 is a basic representation of how the CVU can connect to the controller network.

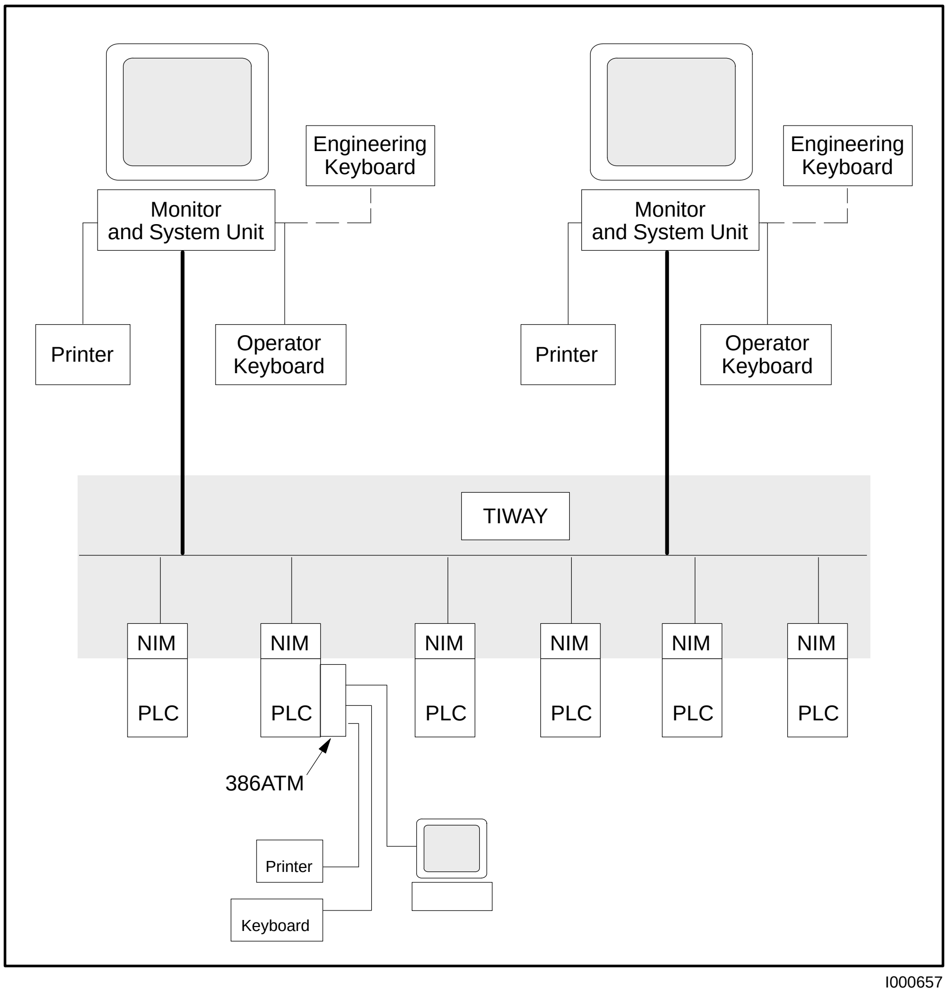


Figure 8-2 Multiple Controller, Multiple CVU Network

Controller Communication

The CVU communicates with SIMATIC controllers over the TIWAY Local Area Network (LAN). All Series 500/505 controllers are supported.

The CVU software running on a 386/ATM can only communicate with the controller in its immediate I/O system and not with other controllers through TIWAY.

Through the networking capability, you can easily configure several types of CVU applications.

- Single programmable controller with one CVU
- Single programmable controller with multiple CVUs
- Multiple programmable controllers with one CVU
- Multiple programmable controllers with multiple CVU hosts

Easy Configuration

Information from the controller is presented on the display pages in the form of text messages, numeric fields, animated symbols, and bar graphs. These display elements are configured by simply entering the appropriate memory addresses, data types, and scaling factors into related databases. Because the CVU does not execute program instructions as the controller does, its setup is referred to as “configuration.” CVU configuration requires four fundamental steps.

1. Determine the control requirements by examining the needs of the operator and determining what portions of the process should be monitored. Also, determine what alarms are necessary, what data should be displayed, etc.
2. Design the process graphic pages and determine the information to be incorporated into the databases.
3. Enter the Configuration (addresses, data types, scaling factors, etc.) information in the databases.
4. Build the run-time files necessary for operation. (The CVU “compiles” the database files for run-time operation.)

Predefined Display Pages

The CVU allows a maximum of 100 configurable display pages and also provides more than 29 preconfigured system pages. The preconfigured pages include:

- PID loop overview pages (eight total)
- Analog alarm overview pages (sixteen total)
- One directory page
- One discrete alarm stack
- One loop tuning page
- One analog alarm page
- One controller status page

Powerful Graphics

Any of the 100 user configurable display can be graphics pages. A graphics editor simplifies the generation of these process overview pages and allows you to easily create graphic representations of your process. A versatile array of options are available to allow animation of pumps, valves, tanks, etc.

An Instrument Society of America (ISA) symbol table is included with the graphics package. Also, you can copy these symbols and edit them to make custom versions.

Recipe Management

With the recipe management features, you can create, edit, copy, document, and load recipes. These recipes can be loaded into the controller program, which makes it easy to produce variations of a product. A recipe normally consists of a list of the controller program variables and the values to be loaded (or verified) in controller memory. The CVU can store up to 999,999 recipes with 400 discrete or analog values.

Alarm Management

The CVU monitors the controller operation for the following alarm conditions.

- PID loop alarms
- Analog alarms
- User specified discrete alarms

An alarm stack is used to track the most recent 512 discrete alarms. Alarms are annunciated on the status banner at the bottom of every display page. They may also be used (through a discrete output) to operate an audible or visual alarm.

Report Packages

You can design up to 20 individual report templates for documentation of process operation. The following report types are available.

- Timed reports
- Event-driven reports
- Operator-initiated reports

The reports can contain controller data values, date and time, and free-form comments. The reports can be up to 80 columns wide and 66 lines long. Report pages can perform mathematical operation on variables with the result printed in a specified location of the report.

Loop Tuning

The CVU has several loop tuning and trending features that allow you to closely monitor and tune your process variables. Preconfigured loop tuning pages are available that allow graphic display of process setpoint, process variable, and process output. A trending feature allows you to view several variables over a period of time, which makes it much easier to monitor process interaction.

Real Time Trending

Up to four variables and the stat of four discrete I/O points can be viewed as a function of time. This page is similar in format to a strip chart recorder. For example, the trending pages can plot the setpoint, output, or process variable of a loop.

Historical Trending

Up to 396 variables can be monitored and displayed, and archived to 99 different files each containing up to four variables. All files can be recalled and display data in a graphical and numerical format for each sample.

Data Logging

The CVU can log data to disk that can be displayed using the offline or online graphical display utilities. Logged data can be automatically exported to LOTUS.PRN (ASCII) files which can then be imported to LOTUS 1-2-3®. The CVU is also Ethernet-compatible.

Password Protection

Different levels of password protection are available to provide security for your control system. There are levels for system design, which provides access to all levels of configuration and operation, and operator levels that reduce the risk of accidental system design changes. The password protection also allows you to protect specific control operations or display screens.

8.2 Monitoring the Process

Display Pages

The operator must be given sufficient information to monitor and control a system. The CVU provides user-configured pages on which the operator can observe applications. An example of a run-time page is shown in Figure 8-3.

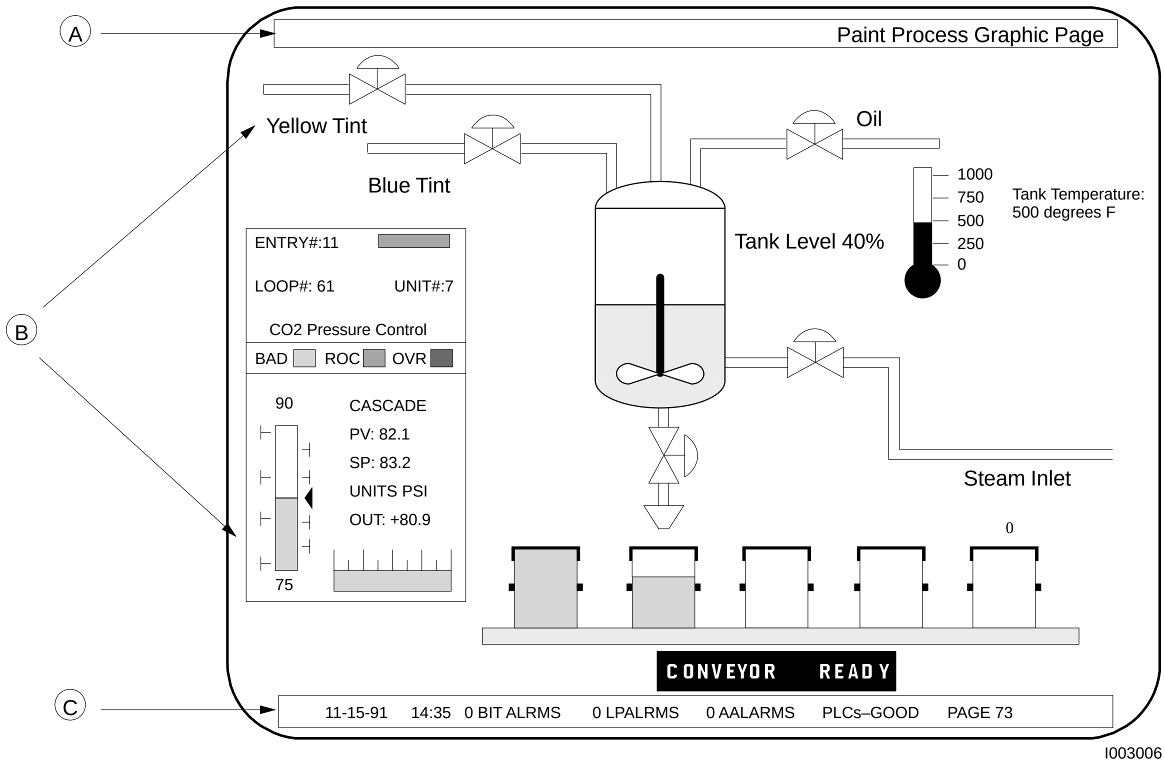


Figure 8-3 Example of a Run-time Graphics Page

There are three sections on the screen shown in Figure 8-3. At the top is the prompt/title line.

A. Prompt/Title Line — shows the title of the current display on the right side of the screen. The left side of this line is the area where prompts appear for the operator to enter page numbers, passwords, or data values.

B. Display Area — can be configured to show process information displayed as text, data, or graphics.

C. Status Line — shows the printer status, date, time, analog alarm status, loop alarm status, bit alarm status, PLC status, the page number, and the network status of the current display.

Displaying Variable Data

The CVU can display the value of a variable in (A) text lines, (B) graphic displays, or (C) faceplates as illustrated in Figure 8-3. The CVU automatically scales and formats the value to the specifications that you designate. Text and free-form graphics pages can contain up to 256 variables each.

Displaying Text

A text line can indicate the state of a discrete I/O point by color and a phrase. A text line can also be animated to flash and/or change text and background color as the discrete I/O point changes states. A text line may be one of the following different types.

- ON/OFF TEXT changes its attributes as the discrete I/O point changes state. The text can flash, change color, display one of two phrases, and show the value of a variable.
- A BIT WINDOW can display one of several different text lines in a location, depending upon the status of one or more specified bits in the controller.
- A VARIABLE WINDOW can display one of several different text lines depending on the value of a variable in the controller.
- COMMENT LINES can show titles or additional information.

Displaying Graphics

The CVU makes it easy for you to configure graphics pages. With this graphics capability, you can construct displays that dynamically represent the state of the process. These displays can include both static and animated elements drawn in any of eight colors, using either standard symbols from the pre-programmed symbol library or symbols that you define.

The CVU already has generic ISA symbols for pumps, valves, and motors. These symbols make graphics page configuration much easier. Also, you can copy and edit these symbols to make custom symbols for your application.

Static elements are non-changing entities used to create a realistic background on a graphics page. Symbols, lines, pipes, boxes, and ellipses are some examples of elements which may be static or animated.

Animated elements are those that have been programmed to change state. For example, text lines can be made to flash; drums can be filled or drained; indicator bars can rise or fall with varying conditions. Symbols may be animated and used as shown in Figure 8-4.

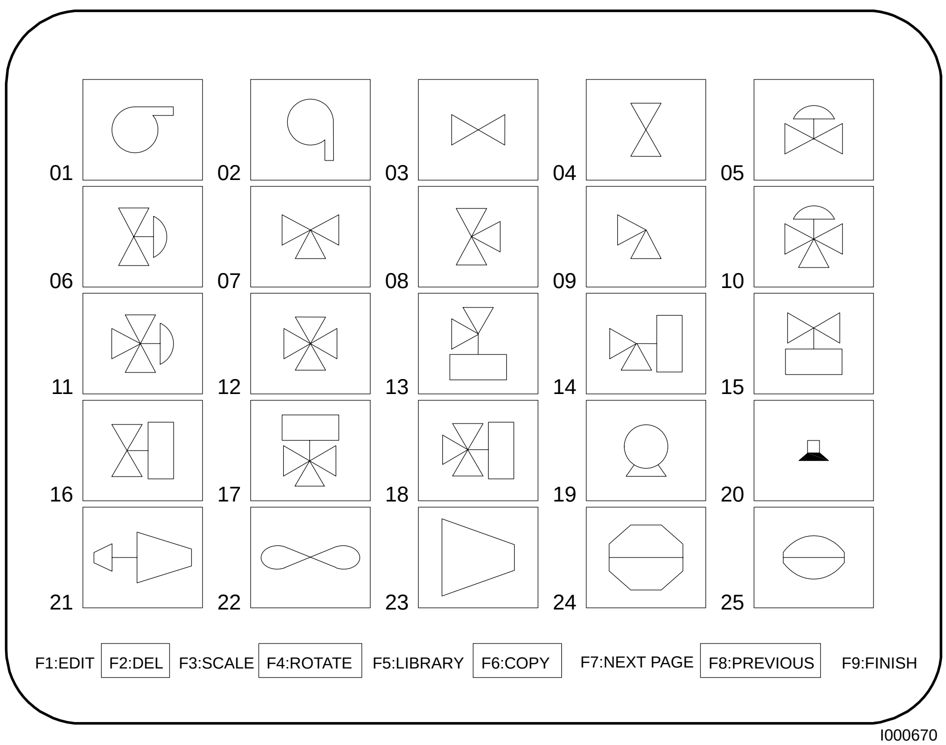


Figure 8-4 Typical Symbol Set Display

Displaying Faceplates

A faceplate is a box containing one or two bar graphs with associated text and a nameplate. The height of the bar graph changes as the value of a variable changes. You can specify the scale factors and engineering units for each bar graph.

Within an indicator display page, the CVU can show up to 8 counter or timer faceplates. On a loop or analog alarm overview page, the CVU can show up to 8 loop or analog alarm faceplates. You can place up to eight loop and analog alarm faceplates on any graphics page.

Figure 8-5 shows a loop faceplate.

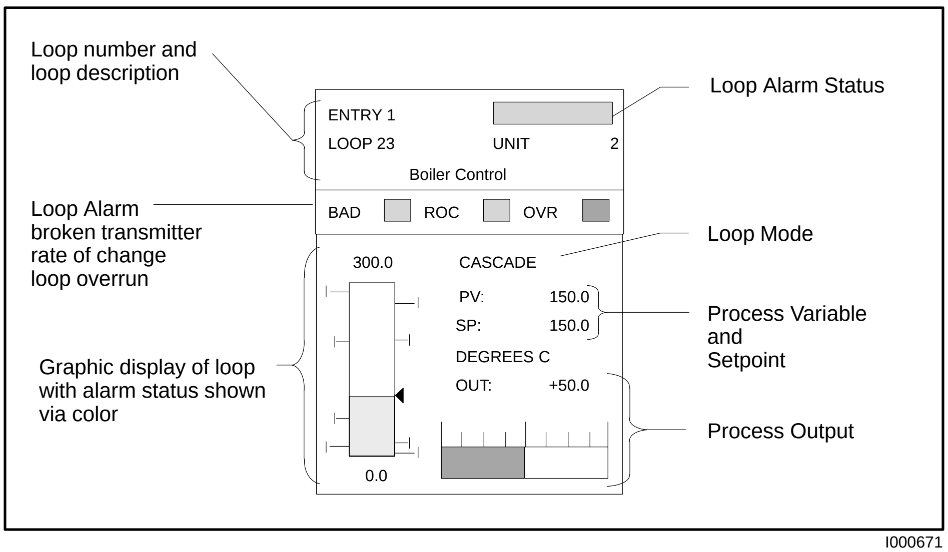


Figure 8-5 CVU Run Time Loop Information Block

Displaying Real Time Trending Data

The CVU has trending pages which can display the values of four variables and the state of four discrete I/O points as a function of time. This page is similar in format to a strip chart recorder. For example, the trending pages can plot the setpoint, output, or process variable of a loop.

You can select from separate (side-by-side) or interspersed (overlaid) trends. These trends, and example of which is shown in Figure 8-6, can be printed by the operator during run time.

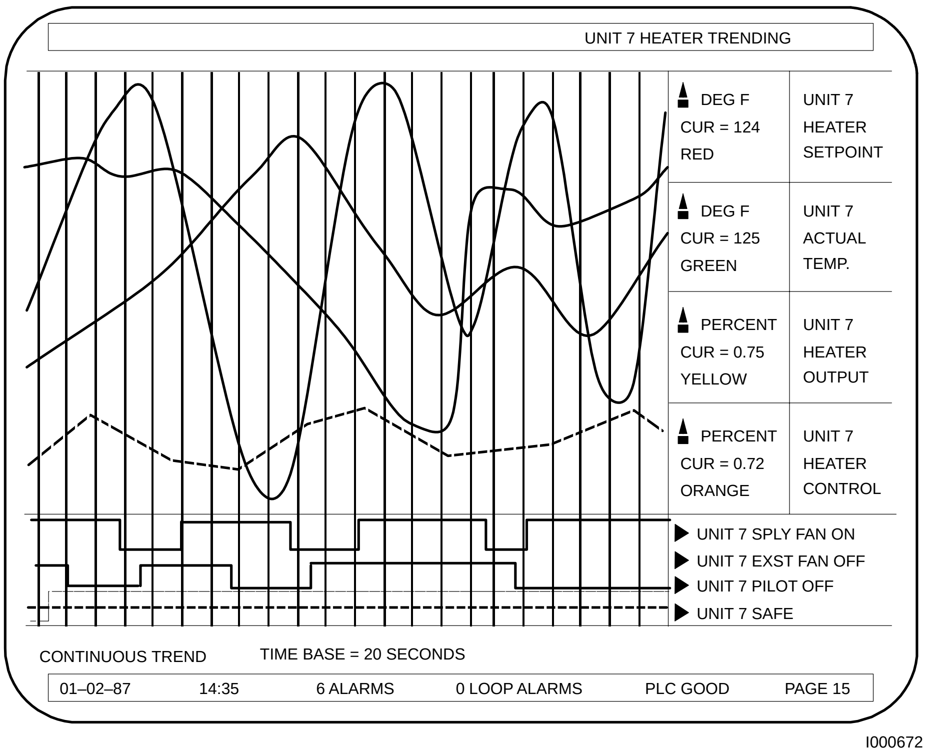


Figure 8-6 Interspersed Trending Example

Displaying Historical Trend Data

The CVU can provide historical trending for up to 396 variables. The system provides the ability to review previously recorded data from any of the 99 files of 99 groups of historical data residing on the hard disk (99 x 99 total max).

The trend display page is a dedicated operator's display that will be used for the historical trend display. You can use one display for all historical trends or for each group configured. See Figure 8-7 for an example of the historical trend display.

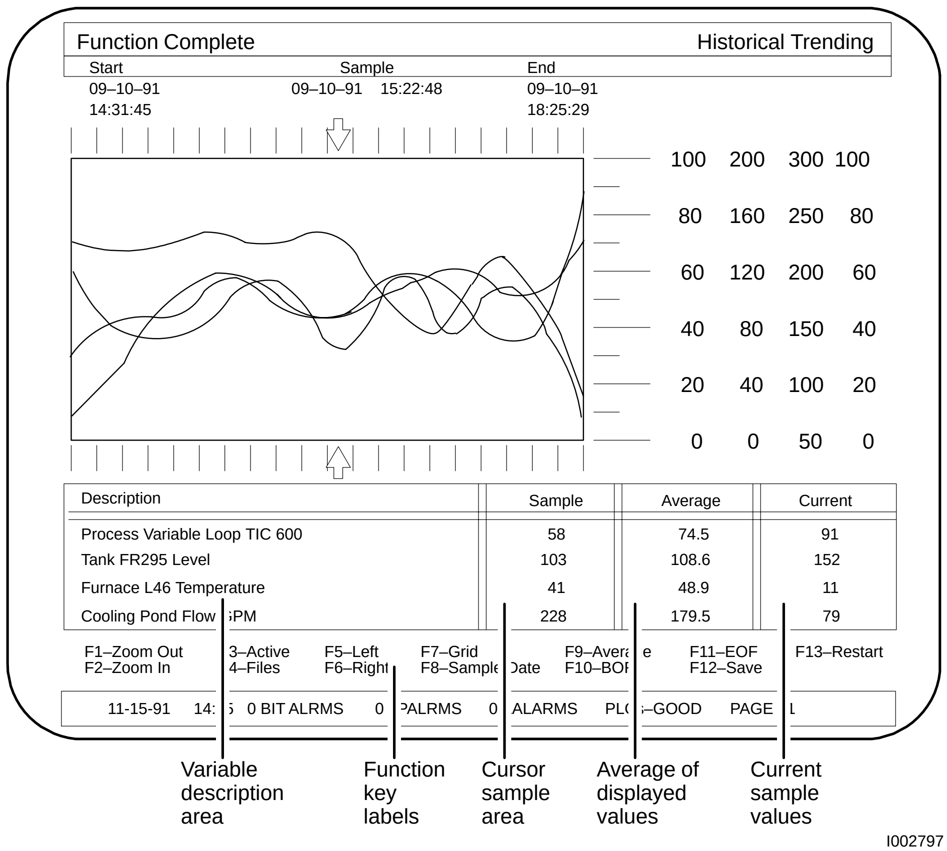


Figure 8-7 Historical Trend Display

8.3 Run-Time Page Formats

Status Page

The controller status page, shown in Figure 8-8, provides a quick status overview for up to 64 controllers connected to the CVU. The display shows network errors, controller mode of operation, fatal errors, base failures, and individual module failures.

PROGRAMMABLE CONTROLLER STATUS

UNIT	DESCRIPTION	LOGGED ON NET	NET ERROR	OP MODE	FATAL ERROR	BASE FAIL	MOD FAIL
1	▶ NIM #1	545		RUN	NO	NO	NO
2	▶ NIM #2	545		RUN	NO	NO	NO
3	▶ NOT USED	NO					
4	▶ NOT USED	NO					
5	▶ NOT USED	NO					
6	▶ NOT USED	NO					
7	▶ NOT USED	NO					
8	▶ NOT USED	NO					

FACILITY ERRORS (F) 07=COM COMMON 02000B=TIMEOUT 050000=HIU 099800A=MODE
02=CVUTA 05=NIM 06=DOS 09=PLC ERRORS 020088=OFFLINE 098002=OUT OF RANGE

02–15–92 13:29 0 BIT ALRMS 9 LPALRMS 1 AALARMS PLCs–GOOD PAGE COM

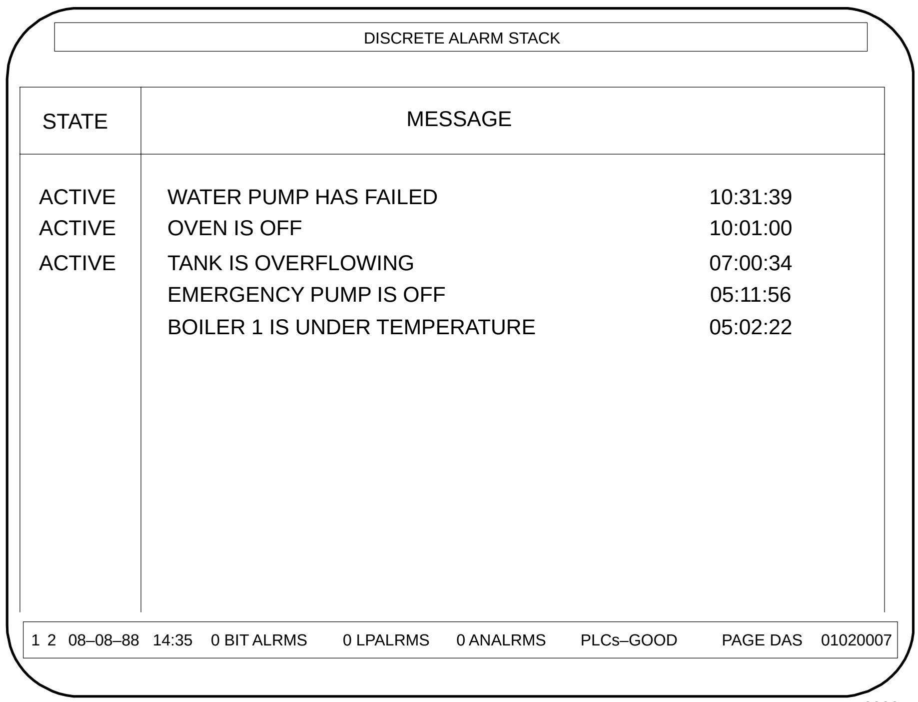
I000673

Figure 8-8 Controller Status Page

Alarm Pages

You list critical I/O points on the alarm pages when it is important to alert the operator if the status of these points changes.

The CVU has one Discrete Alarm Page. This page allows the operator to view up to 512 alarms, 20 at a time. You scroll up and down to show additional alarms and to select which alarms to acknowledge and reset. Figure 8-9 is an example of a discrete alarm page. You can also display the time the alarm occurred in the controller. This will simulate “first out” alarm indication in the event multiple alarms are received at the same time by the CVU.



I000674

Figure 8-9 Example of a CVU Discrete Alarm Page

Loop Tune and Loop Overview Pages

The CVU can display up to two loops on the Loop Tune Page, shown in Figure 8-10. The CVU also has 8 preconfigured Loop Overview Pages that contain up to 8 loop overview blocks per page. The information contained in the Loop Overview pages has less detail than the Loop Tune Page. For example, the trending data is not shown. In both cases, the faceplates are similar in format to the Loop Information Block.

With these pages, the engineer can tune a loop during run time and can also acknowledge a loop alarm.

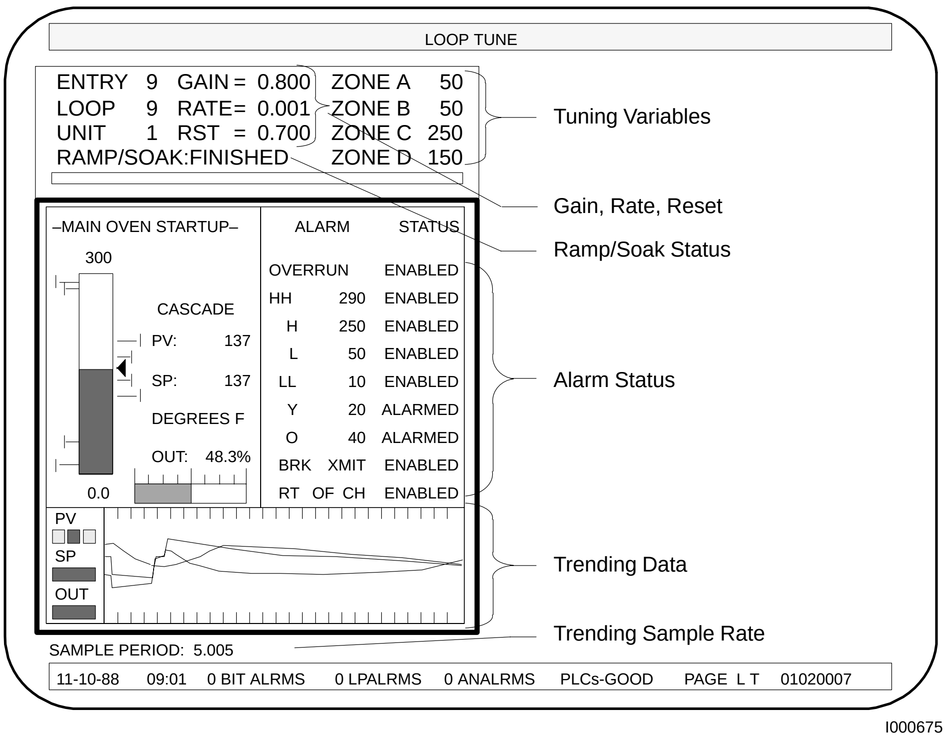


Figure 8-10 CVU Loop Tune Page

You can adjust the Sample Period for the loop during run time. The range is 0.055–999 seconds. Because the Sample Period is based on a 55 millisecond clock, the value that the CVU uses and displays will vary slightly from the value that you enter.

Configurable Graphics Pages

Once you have created the custom graphic screens, the operator can view any of the user-configured pages using several options.

The PAGE NUMBER key displays the directory page and prompts you for a page number; enter the number of the page desired. Figure 8-11 is an example of a user-configured page.

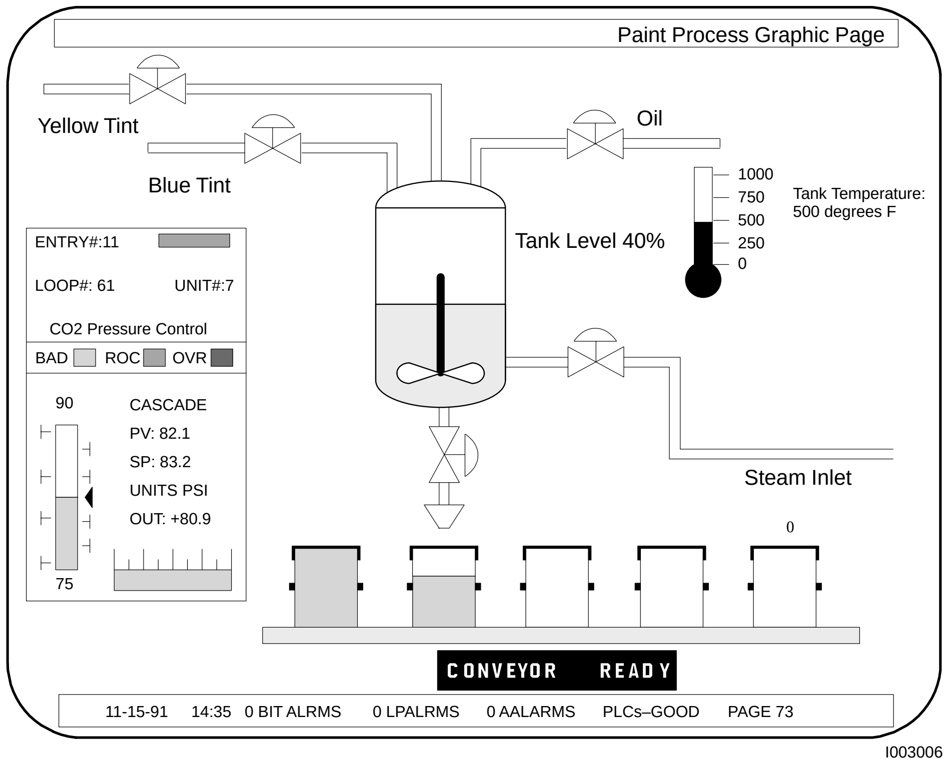
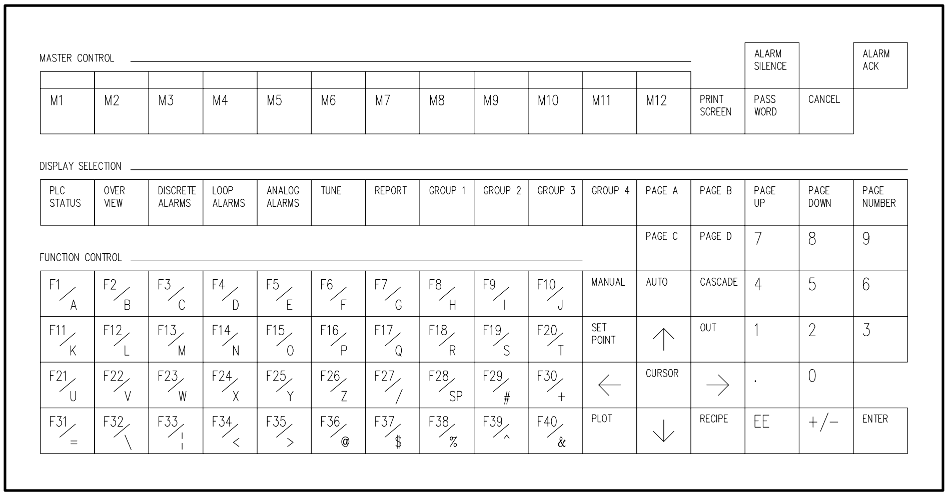


Figure 8-11 Example of a User-configured Page

8.4 Controlling the Process

Operator Keyboard

The operator can control discrete devices, change variables, and tune loops from the Operator Keyboard, which is shown in Figure 8-12. The Operator Keyboard is a sealed membrane type keyboard that is suitable for the harsh environments found on the factory floor.



I000677

Figure 8-12 CVU Operator Keyboard

Discrete Bit Control

The operator can control a discrete device, such as a pump or a valve, by using control keys. The 12 Master Control keys (M1–M12) at the top of the operator keyboard are operational regardless of the current page. These keys are intended to control major functions of the application.

The 40 Function Control keys (F1–F40) may be assigned a different operation for each page. A function assigned to a key for a particular page operates only when that page is displayed.

Variable Control

By using the cursor and numeric keys, the operator can change the value of a variable from the keyboard. The operator enters the new value in the targets shown on the faceplate. The CVU automatically performs any necessary scaling.

Loop Mode Control

The operator can change the mode of the loop to manual, automatic, or cascade. The operator does this from either the loop tune page or a loop indicator faceplate.

8.5 Data Output

Running Log

It is often necessary to maintain records relating to the operation and state of the process. You can configure a “running log” by specifying text lines to be printed when a discrete I/O point (or Control Relay) changes state. The CVU then generates a log (with a time and date stamp) of all of these events. The printed text lines can include on/off phrases or variables. Figure 8-13 shows an example of a text line log sheet.

06–22–88	12:34	MIXER #7 READY TO DUMP
06–22–88	12:34	MIXER #4 READY FOR NEW BATCH
06–22–88	12:35	MIXER #3 READY FOR NEW BATCH
06–22–88	12:37	MIXER #7 READY FOR NEW BATCH
06–22–88	12:39	MIXER #2 READY TO DUMP
06–22–88	12:39	MIXER #5 OUT OF SERVICE
06–22–88	12:42	FLOUR TRANSFER SYSTEM START
06–22–88	12:45	MIXER #2 READY FOR NEW BATCH
06–22–88	12:48	MIXER #6 READY TO DUMP
06–22–88	12:53	SUGAR TRANSFER SYSTEM START
06–22–88	13:02	FLOUR TRANSFER SYSTEM STOP
06–22–88	13:09	MIXER #4 READY TO DUMP

I000678

Figure 8-13 Example of a Text Line Log

Report Formats

A report can include the values of variables, the time and date, and free-form comments. The CVU report may be up to 80 columns wide and 66 lines long. If a report directory is necessary, you can generate one by using a configurable page and listing all configured reports.You can design formats for up to 20 reports during configuration. Figure 8-14 shows an example report.

Reports are printed as a result of one of the following conditions.

- The operator requests a report by pressing the REPORT key and, when prompted, entering the number of the desired report.
- An event occurs, signaling a report.
- A report is configured to print out at a certain time.

Acme Bakery Daily Shift Report

02–10–92 10:13

LINE	PRODUCT CODE	1ST SHIFT	2ND SHIFT	3RD SHIFT
1	3443	218	199	200
2	7889	77	150	120
3	0	0	0	0
4	2332	97	90	82
5	7886	112	60	146
6	6770	56	29	0
7	4501	123	169	176
TOTALS		683	697	724

Turn in all production reports before noon on the following day. Thanks!

I000679

Figure 8-14 Report Example

Report pages can perform mathematical operations on variables with the result printed in a specified location of the report. All mathematical expressions supported by MS-BASIC™ are included as standard.

Table 8-1 Math Operations

<table>
<tr><td>Addition of two operands</td><td>+</td></tr>
<tr><td>Multiplication of two operands</td><td>*</td></tr>
<tr><td>Exponentiation – one operand raised to the power of another operand</td><td>^</td></tr>
<tr><td>Absolute value of an operand</td><td>ABS(X)</td></tr>
<tr><td>Arctangent of an operand</td><td>ATN(X)</td></tr>
<tr><td>Tangent of an operand</td><td>TAN(X)</td></tr>
<tr><td>Logarithm of an operand</td><td>LOG(X)</td></tr>
<tr><td>Return the truncated integer of an operand
• For negative X,FIX returns the first negative integer greater than X</td><td>FIX(X)</td></tr>
<tr><td>Generate a random number
• RND generates the next random number in the sequence when X > 0 or is omitted
• RND returns the last number generated when X = 0
• RND returns the same number for a given X when X < 0</td><td>RND(X)</td></tr>
<tr><td>Return the integer value of the remainder generated by module division; for example, 35.69 | 7.9 = 2 (36 / 8 with remainder of 4)</td><td>| (X)</td></tr>
<tr><td>Subtraction of two operands</td><td>–</td></tr>
<tr><td>Floating point division of one operand by another</td><td>/</td></tr>
<tr><td>Truncated integer division of an operand by another
• The operands are rounded to integers and the result is truncated to an integer</td><td>\</td></tr>
<tr><td>Square root of an operand</td><td>SQR(X)</td></tr>
<tr><td>Cosine of an operand</td><td>COS(X)</td></tr>
<tr><td>Sine of an operand</td><td>SIN(X)</td></tr>
<tr><td>E to the power of an operand</td><td>EXP(X)</td></tr>
<tr><td>Return the truncated integer of operand
• For negative X, INT returns the first negative integer less than X</td><td>INT(X)</td></tr>
<tr><td>Return the sign of the operand
• SGN returns +1 when X > 0
• SGN returns –1 when X < 0
• SGN returns 0 when X = 0</td><td>SGN(X)</td></tr>
</table>

See Table 8-2 for a description of Feature Specifications.

Table 8-2 Feature Specifications

Pixel based Graphic Editor	640 x 480
Real-time Trending	100 pages (8 points/page)
Historical Trending Variables	99 pages (4 points/page)
Automatic Data Logging	99 logs (99 files/log)
Recipe Download	999,999 (400 points/recipe)
Report Package with Integral Math Calculations	Yes
Full Page Symbol Editor with Zoom Function	Yes
Pre-configured Displays	92
Pre-configured Loop Monitoring Displays	8 pages (8 loops/page)
Pre-configured Loop Tuning Displays	Yes
Pre-configured Discrete Alarm Stack	512 alarms
Pre-configured Discrete Alarm Acknowledge	Printer and/or PLC
PLC Communication Status	64 units
Analog Alarm Overviews	16 pages (8 alarms/page)
Animated Symbols per Page	256
Animated Bar Graphs per Page	256
Analog Values per Page	256
Text Messages per Page	256
Intelligent Transmitter Support	Yes
Printer Support	HP Deskjet™ HP Paintjet™ HP Laserjet™ Epson Color Epson FX Series TI 800 Series

8.6 Hardware Specifications

Refer to Table 8-3 for a description of CVU Specifications.

Table 8-3 CVU Specifications

Supply Voltage	90–132 VAC, 180–264 VAC	
Supply Frequency	47–63 Hz	
Operating Temperature	10° to 33° C (50° to 91° F)	
Storage Temperature	–10° to 60° C (14° to 140° F)	
Relative Humidity	20% to 80%, noncondensing (operating and storage)	
Vibration	10 to 25 Hz .006” 25 to 500 Hz 0.2 Gs	
Graphics Software Interface	VGA	
Graphics Resolution	640 x 480 pixels	
Communication	TIWAY @ 115.2 k baud	
Distance between CVU and Controller	10,000 feet maximum	
Monitor	14″	19″
Power Consumption	100 Watts	100 Watts
Supply Voltage	120–230 VAC	90–250 VAC
Supply Frequency	50–60 Hz	45–70 Hz
CRT Dimensions	14.1″ W x 14.2″ D x 12.2″ H	19″ W x 20.5″ D x 18.6″ H
Agency Listings	FCC Class A, UL listed, CSA Certified	

The CVU1000 is equipped to withstand most harsh environments found on the factory floor. However, you can install it in a 19-inch National Electrical Manufacturers Association (NEMA) cabinet. Figure 8-15 shows the mounting dimensions.

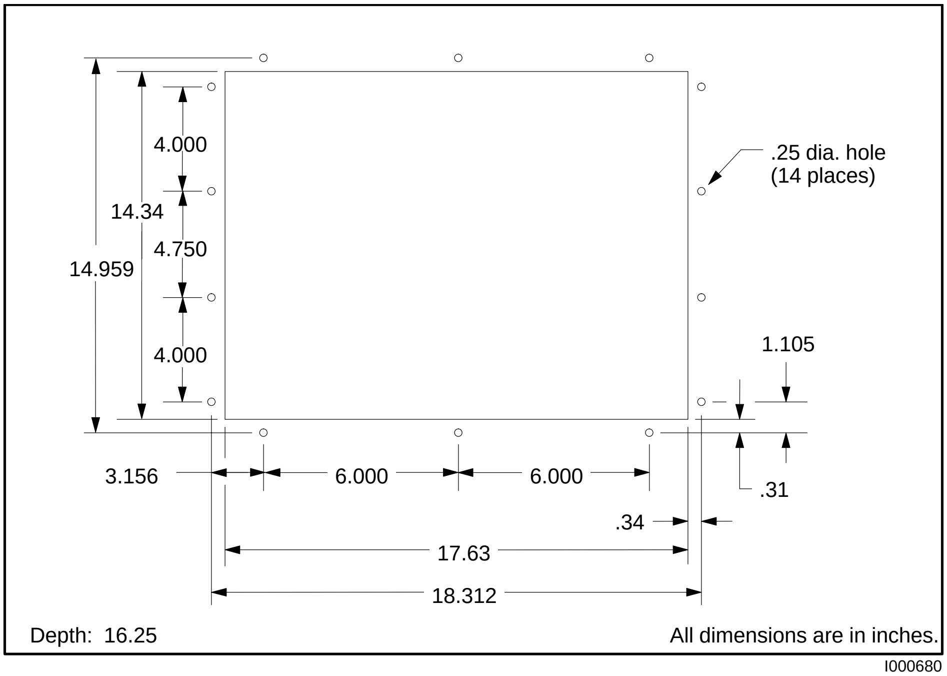


Figure 8-15 CVU1000 Dimensions

Appendix A

SIMATIC 545 Timeline Configuration

A.1 Overview

Scan Portions

The SIMATIC 545 controller scan cycle is divided into a discrete portion and an analog portion as shown in Figure A-1.

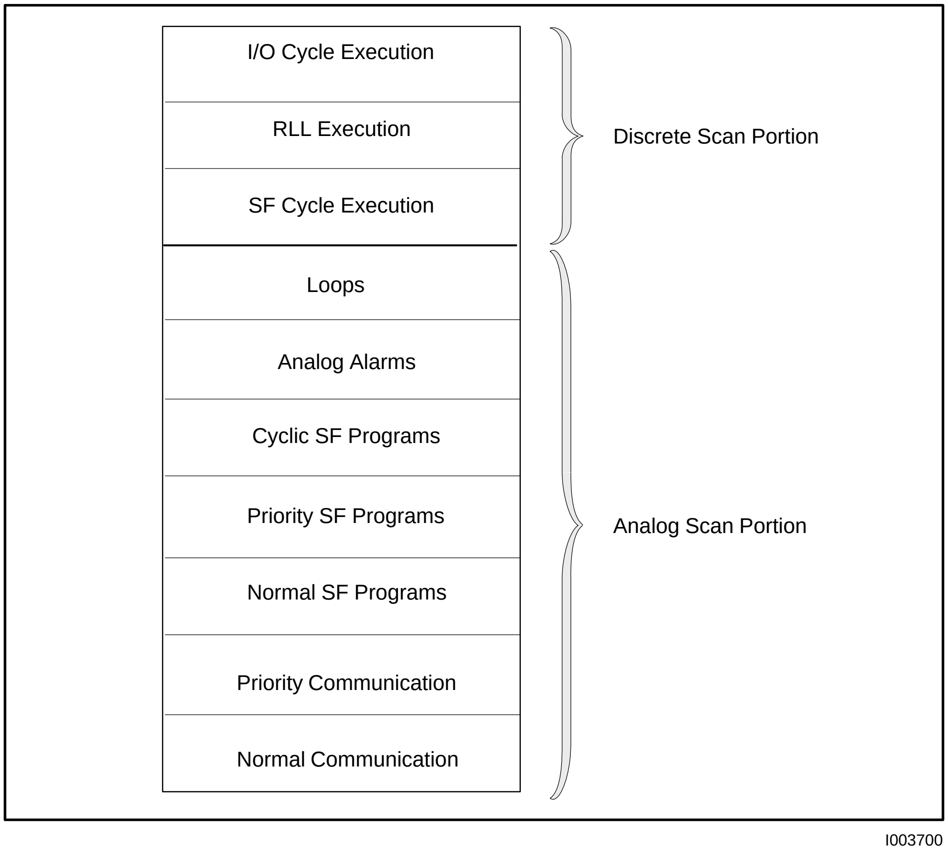


Figure A-1 Scan Portions

The discrete portion is often referred to as the “logic side” and used to service the I/O subsystem, execute RLL, and service intelligent I/O module updates.

The analog portion is often referred to as the “regulatory control side” and controls the PID loop execution, Special Function program execution, and network communication.

A.2 Discrete Scan

The discrete scan consists of three primary tasks that are executed sequentially (Figure A-2) and at a rate that can be user-specified.

Normal I/O Update During the normal I/O cycle update, the programmable controller writes data from the image registers to the outputs, and stores data from the inputs into the image registers. The length of the I/O update cycle is dependent upon the number of bases and types of modules (analog, discrete or intelligent). All I/O points are fully updated each scan.

Main Ladder Logic Cycle The programmable controller executes the main RLL task.

Special Function Module Communication Communication with special function (SF) modules, e.g., NIM, BASIC, PEERLINK, etc., consists of the following actions.

- Service requests from a previous scan for which processing has been completed are transmitted to the SF modules.
- Remote bases are polled for initial SF module service requests.
- Remote base communication ports are polled for service requests.
- Service requests from SF modules and remote base communication ports are processed.

Each SF module that requires service increases the scan time, depending upon the type of module and task. Each type of module is allowed a certain number of service requests per scan. Once these are completed, this function is terminated. Some service requests can be deferred, and these are processed during the analog task time slice described below.

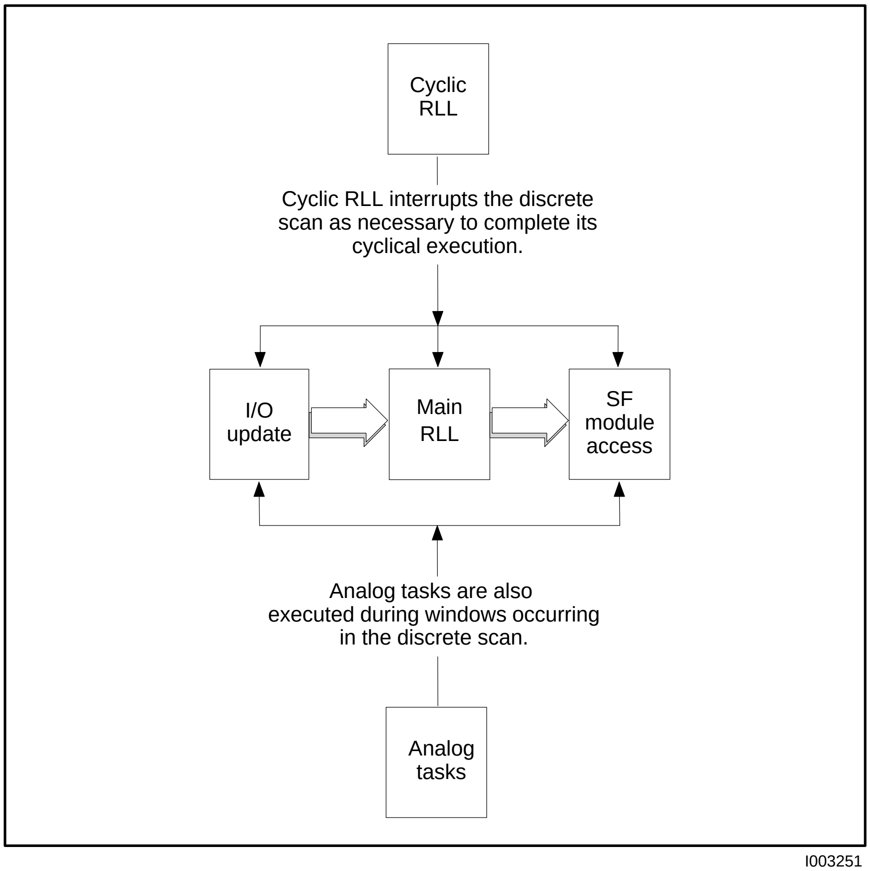


Figure A-2 Discrete Scan Sequence for the SIMATIC 545 Controller

Each module that requires service during this period adds scan time according to the type of module and the type of task. Each type of module is allowed a certain number of requests, block transfers, or store-and-forward operations per scan. Once these are completed, the intelligent I/O cycle is terminated.

A.3 Analog Scan

Analog Task Processing

The analog portion of the scan is composed of five general types of tasks (Figure A-3), which are cyclical or noncyclical in their execution.

Analog tasks are guaranteed execution once per scan, following the discrete scan, provided there is processing to be done. Analog tasks are also processed during windows of halted activity that occur during the normal I/O and SF portions of the scan. RLL execution is not interrupted by analog tasks.

You can adjust the amount of time spent per controller scan for all analog tasks, except diagnostics, with a programming unit and using AUX Function 19. The time allocation for a given analog task is referred to as its **time slice**.

Cyclic Analog Tasks

The following types of processes are executed cyclically. Each has a sample rate which determines how often it is executed.

- Loops
- Analog alarms
- Cyclic SF programs

The programmable controller has an analog task that executes each type of cyclic process. When enabled, each cyclic process is placed in the execution queue that is managed by the analog task responsible for executing that type of process.

The cyclic processes are time-ordered on their individual queues according to when each process is rescheduled for execution, relative to the other cyclic processes within the same queues. The process with the highest priority (closest to overrunning) is executed first. The process is executed until it is completed or until the time specified for that particular task's time slice expires. If the executing process is completed before the time slice expires, the process with the next highest priority is executed. If the time slice expires before the process is completed, the process (and the task) is put on hold in its current position.

The programmable controller then advances to the next analog task. When the programmable controller sequences through its operations and returns to an analog task with a cyclic process on hold, the process resumes execution from the hold point, unless a higher priority process was scheduled since the last respective time slice. If a process in a cyclic time slice is not finished executing when it is scheduled to execute again, an overrun flag is set.

Restricted SF programs, which are called by loops or analog alarms, are executed from within the loop or analog alarm tasks. Therefore, their execution time is included within the loop or analog alarm time slice.

SF subroutines, which are called by SF programs or other SF subroutines, are processed during the calling program's time slice.

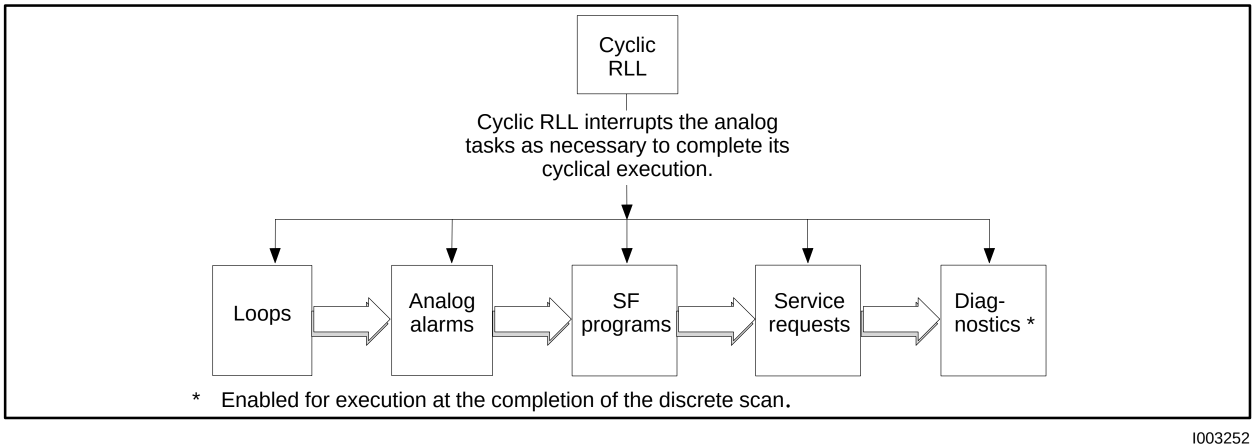


Figure A-3 Analog Task Scan Sequence for the SIMATIC 545 Controller

Noncyclic Analog Tasks

The following types of processes are executed noncyclically.

- Priority/Non-priority SF programs
- Service request messages
- Run-time diagnostics

Priority and Non-Priority SF Programs are non-cyclic processes that are queued when the SFPGM RLL box instruction receives power flow. There is an analog task that executes priority SF programs, and another analog task that executes non-priority SF programs. These processes are executed in the order they are queued on the appropriate task's execution queue. When the programmable controller completes one of these processes, it removes the process from the respective queue and turns on the SFPGM output. There are no overrun flags associated with these processes.

Service Requests received from the communication ports are placed on one of two communications queues. Read and write commands are placed on the priority communication queue for fastest response. Commands that may require several scans to complete, e.g., program edits and the TISOFT **FIND** function, are placed on a non-priority communications queue.

Run-time Diagnostics are enabled for execution at the completion of the discrete scan. The time slice for diagnostics is 1 ms and cannot be changed.

Setting the Scan

The SIMATIC 545 scan is defined as the time between normal I/O updates. You can set the scan as follows.

- Fixed — The programmable controller starts a new discrete scan at the specified time interval. The controller executes the discrete scan once and then cycles to the analog scan portion, executing the analog tasks at least one time. If the analog tasks are completed within the specified time, the controller goes into a loop mode (processing analog tasks or idling) until time to start the next scan.

  A scan overrun status bit is set (bit 14 in Status Word 1) if the total execution time for the discrete scan portion and the first execution of the analog scan portion exceeds the fixed scan time.

- Variable — The programmable controller executes all tasks once and then starts a new scan. All discrete and analog tasks are guaranteed one execution per scan. Specify variable scan for the fastest possible execution of the discrete scan.

- Variable with upper limit — The programmable controller executes the discrete scan once and then executes the analog tasks. The programmable controller remains in the analog portion of the scan as long as there are analog tasks to be done. When the upper time limit expires, or no analog tasks require processing, a new scan is begun.

  The analog scan portion is executed at least one time. A scan overrun status bit is set if the total execution time for the discrete scan portion and the first execution of the analog scan portion exceeds the upper limit.

Cycle time for the cyclic RLL can be a fixed value or a user-specified variable. As a variable, the cycle time can be changed by logic in your application program. If the cyclic RLL completes execution in less than the specified cycle time, execution does not resume until the next cycle begins. The programmable controller scan time is extended by the amount of time to execute the cyclic RLL multiplied by the number of times the cyclic RLL is executed during the programmable controller scan.

The timing relationship of the scan operations is shown in Figure A-4. Refer to the Section A.4 for details about how to configure the time slices.

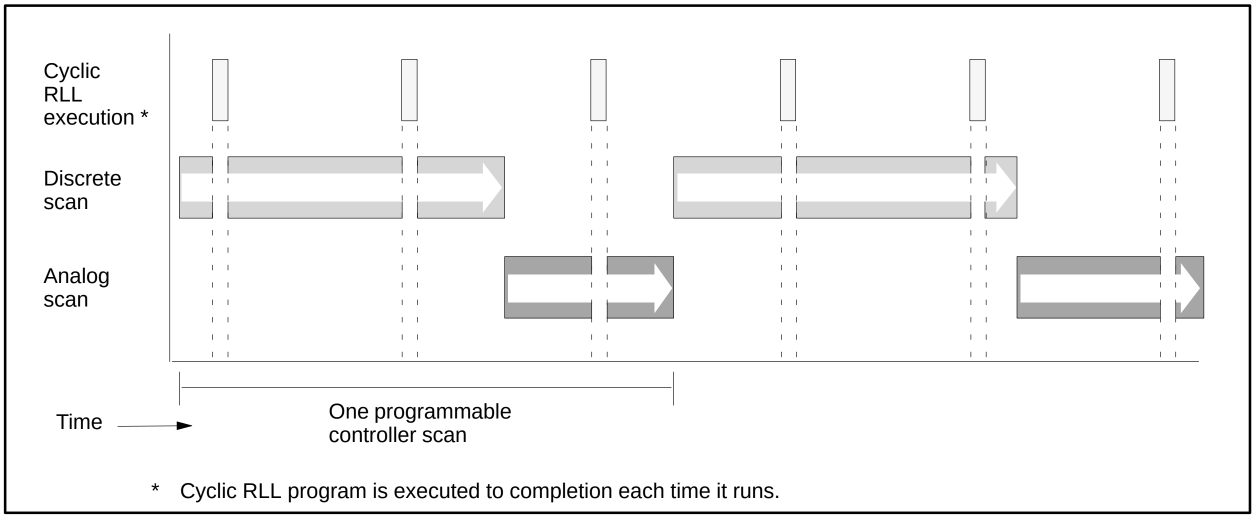


Figure A-4 Timing Relationship of the Controller Scan Operations

A.4 Setting Time Slices

Basic Strategy

For most applications, you will not need to adjust the default timeslices. After you have made your best predictions for analog process execution times (loops, analog alarms, SF programs, etc.), you may still want to make adjustments in the timeline, based on empirical data. You have the option of fine-tuning the sub-slices of the analog timeslice to ensure that these analog processes are executed as quickly as possible and do not overrun. The sections that follow describe some suggestions about how to approach the fine-tuning.

When you set the timeslices, you are also affecting the length of the overall controller scan. Shorter analog timeslices reduce the overall scan, which result in a faster I/O update. Typically, you want to reduce the analog portion of the scan as much as possible to reduce the overall scan time. However, do not allow too little time for the analog portion. Loops and analog alarms will begin to overrun, and the time for SF programs to execute after scheduling will be longer.

Using Peak Elapsed Time Words The controller stores the peak elapsed time for a process to execute. The peak elapsed time is the time from when a process is scheduled (placed on the queue) until the process completes execution. The controller updates these words each time the process is scheduled and executed.

- $LPET_n$ for loops (n = 1–64)
- $APET_n$ for analog alarms (n = 1–128)
- $PPET_n$ for SF Programs (n = 1–1023)
- $TPET_n$ for RLL cyclic tasks (n = 1–2)

You can determine if the loops, analog alarms, or cyclic SF programs are coming close to overrunning. If the value in the APET, the LPET, or the PPET approaches the sample time, you can increase the timeslice for the analog alarms, the loops, or the cyclic SF programs. Alternatively, you can decrease the other timeslices. This reduces the overall scan, allowing the analog alarms or loops to run more often in a given time. The time needed for the discrete portion of the scan limits how much you can reduce the overall scan.

If the PPET indicates that an SF program is taking an excessive amount of time for execution, you can increase the timeslice appropriately. If the SF program is critical, move that SF program to the Priority queue.

PPET is only valid for an SF program queued from RLL (priority, non-priority, or cyclic SF programs). The time for executing an SF program called from a loop or analog alarm is included in LPET or APET, respectively.

$TPET_n$ contains the peak elapsed time for the execution of an RLL task, $TPET_1$ for the main RLL task (TASK1) and $TPET_2$ for the cyclic RLL task (TASK2).

Using the Status Words Check the status of the following bits in Status Word 162 (STW162) to see if these analog processes are overrunning.

- Bit 3 Loops are overrunning.
- Bit 4 Analog Alarms are overrunning.
- Bit 5 Cyclic SF programs are overrunning.
- Bit 6 Non-priority SF program queue is full.
- Bit 7 Priority SF program queue is full. All priority and non-priority SF programs will be executed in turn.
- Bit 8 Cyclic SF program queue is full.

Check bit 14 in Status Word 1 (STW01) to see if the overall scan is overrunning. When the bit is true (= 1) the scan time required to execute the entire program is greater than the designated scan time.

The instantaneous discrete execution time (the time to execute the discrete portion of the scan) is reported in status word 192 (STW192). The instantaneous total scan time is reported in status word 10 (STW10).

Table A-1 summarizes the performance and overrun indicators.

Table A-1 Performance and Overrun Indicators

Performance Overrun Indication	Status Word/Aux Function
Discrete scan overrun indicator	STW01 and AUX 29
Previous discrete scan time	STW192 and AUX 19
Previous total scan time	STW10 and AUX 19
Peak discrete and total scan time	AUX 19
Cyclic process overrun indicators	STW162
Individual cyclic process overrun indicators	V–Flags and T6
SF queue full	STW162
Process peak elapsed time	LPET, APET, PPET, SPET
Scan watchdog	AUX14

Calculating the Timeline

SF Modules When you determine the base location for SF modules, consider the impact on the controller scan. Update time for an SF module is an order of magnitude faster when you install the module in the local base, versus a remote base, resulting in less extension of the controller scan.

If all SF modules cannot be installed in the local base, consider placing low-activity SF modules, such as the ASCII, BASIC, or DCP modules, in a remote base. Locate high-activity modules, such as the NIM or PEERLINK, in the local base.

SF Program Execution Time Your calculation of an SF program execution time based on the statement times is the actual execution time required for the controller to run the SF program. The time from when the SF program program is placed on the queue until the point at which execution begins can vary. This wait depends upon the number of SF programs scheduled, how long they take for execution, how long the timeslice is, and the priority of other analog tasks scheduled for processing.

Priority/Non-Priority SF Program Queues The two SF program queues provide a means of separating critical SF programs, (needing to run quickly) from less important SF programs. Keep the number of priority SF programs as small as possible, and if it is not essential that an SF program be executed very rapidly, assign it to the other queue.

You can increase the timeslice for the Priority SF programs to ensure that queued programs are executed as quickly as necessary.

Cyclic SF Program Queue The controller allows you to queue up to 32 cyclic SF programs at once. If you create more than 32, only the first 32 that are queued will be executed.

Do Not Overload the Controller Remember that the controller has a finite set of resources. The controller supports 64 loops, but you cannot run them all at 0.1 second intervals without adversely affecting the execution of the other analog processes. You cannot run all 128 analog alarms at 0.1 second intervals for the same reason.

RLL versus SF Math The controller processes RLL math much faster than SF program math. When possible, use RLL for integer mathematical calculations for faster response time.

Timeslice Resolution Timeslices have a resolution of 1 ms. When you program a 4 ms timeslice, that timeslice will be executed for four 1 ms clock pulses. The time from the beginning of the timeslice to the first pulse can vary from zero time to a full 1 ms pulse. Therefore, the actual time in a 4 ms timeslice is greater than 3 but less than or equal to 4 ms.

Each transition between timeslices takes approximately 200 μs of overhead. This overhead is included in the time allotted to each timeslice and does not have an additional impact on the overall scan.

Appendix B

Loop Operation

B.1 Loop Algorithms

Overview

PID loops are truly stand-alone control blocks, since they do not require additional ladder logic or other programming. This appendix describes the implementation of PID control loops in the SIMATIC 545 controller. The discussion includes algorithms used, options that may be selected, and performance.

I/O Points Variables and Scaling

The controller has three kinds of I/O points—discrete, analog, and word. Discrete points (denoted by Xn for inputs and Yn for outputs) are single bits with 0 corresponding to off and 1 corresponding to on. Word points (denoted by WXn for inputs and WYn for outputs) are 16-bit words, with the interpretation of the bits being dependent on the type of I/O module. Analog points (also denoted by WXn for inputs and WYn for outputs) are 16-bit integers scaled by type of I/O point as follows:

- 0% OFFSET – Stored as an integer variable in the the range of 0 to 32000. (This would typically correspond to an actual range of 0 to +5 volts.)
- 20% OFFSET – Stored as an integer variable in the range of 6400 to 32000. (This would typically correspond to an actual range or 1 to +5 volts or 4 to 20 milliamps.)
- BIPOLAR – Stored as an integer variable in the range of –32000 to 32000. (This would typically correspond to an actual range of –5 to +5 volts.)

All of these representations are referred to as scaled integers.

Loop Variables

There are two primary variables that the loop must access. These are the input to the loop—called the Process Variable, and the output from the loop—called the output. Both are analog variables. For each loop, the SIMATIC 545 allows you to specify the location of the variable and the manner in which it is scaled.

When the controller reads the Process Variable for a loop, it will automatically convert it from a scaled integer to a floating point number in the range 0.0 to 1.0 (0.0 represents PV Low Range; 1.0 represents PV High Range). These values are referred to as normalized real numbers. All loop calculations are performed using normalized reals. The output from the loop is also a normalized real, which will be converted to a scaled integer when stored to memory.

The controller allows you to specify the range of the Process Variable in engineering units (for example, degrees centigrade). External references to loop variables are always in engineering units. The normalized real format is only used internally to the controller for the purpose of loop calculations.

PID Control

The controller provides feedback loops using the PID (Proportional-Integral-Derivative) algorithm. The controller output M is computed from the measured process variable PV as follows:

Let:

$$K_c = \textit{Proportional gain}$$

$$T_i = \textit{Reset or integral time}$$

$$T_d = \textit{Derivative time or rate}$$

$$SP = \textit{Setpoint}$$

$$PV(t) = \textit{Process Variable at time } t$$

$$e(t) = SP - PV(t)$$

Then:

$$M(t) = \textit{controller output at time } t$$

$$M(t) = K_c\left[e(t) + \frac{1}{T_i}\int_0^t e(x)dx + T_d\frac{d}{dt}e(t)\right] + M(0)$$

By proper choice of the T_i and T_d values, the integral and/or derivative action may be eliminated resulting in the other common types of loops—P, PI, and PD. The controller also provides a mechanism whereby the I, ID, and D loops may be obtained.

Standard PID (Discrete Form)

The output $M(t)$ above may be approximated using a discrete form of the PID equation.

Let:

$$T_s = \textit{Sample rate}$$

$$K_c = \textit{Proportional gain}$$

$$K_i = \textit{Coefficient of the integral term}$$

$$= K_c(T_s/T_i)$$

$$K_r = \textit{Coefficient of the derivative term}$$

$$= K_c(T_d/T_s)$$

$$T_i = \textit{Reset or integral time}$$

$$T_d = \textit{Derivative time or rate}$$

$$SP = \textit{Setpoint}$$

$$PV_n = \textit{Process Variable at nth sample}$$

$$e_n = \textit{Error at nth sample } (SP - PV_n)$$

$$M_0 = \textit{Initial value (also referred to as the controller reference value) to which the controller output has been initialized}$$

Then:

$$M_n = K_c e_n + K_i \sum_{i=1}^{n} e_i + K_r(e_n - e_{n-1}) + M_0$$

This form of the PID equation is referred to as the position form since the actual actuator position is computed. The controller also provides a velocity form of the PID equation which computes the change in actuator position. The velocity equation is described later in this section. The controller modifies the standard equation slightly to use the derivative of the Process Variable instead of the error as follows.

$$M_n = K_c e_n + K_i \sum_{i=1}^{n} e_i - K_r(PV_n - PV_{n-1}) + M_0$$

These two forms are equivalent unless the setpoint is changed. In the original equation, a large step change in the setpoint will cause a correspondingly large change in the error resulting in a bump to the process due to derivative action. This bump is not present in the second form of the equation.

The controller also combines the integral sum and the initial output into a single term called the bias (denoted in this document by Mx). This results in the following set of equations.

$$Mx_0 = M_0$$

$$Mx_n = K_i e_n + Mx_{n-1}$$

$$M_n = K_c e_n - K_r(PV_n - PV_{n-1}) + Mx_n$$

The controller will always keep the output M in the interval [0.0,1.0]. This is done by clamping M to the nearer of 0.0 and 1.0 whenever the calculated output falls outside this range.

Reset Windup Protection

Reset windup can occur if reset action is specified and the computation of the bias term Mx is performed exactly as shown in the equation below.

$$Mx_n = K_i e_n + Mx_{n-1}$$

For example, consider an application in which the output is controlling a valve and the PV goes to some value PV_a such that $PV_a > SP$ and does not change for a period of time. Since the error e will always be negative, this will cause the bias term Mx to constantly decrease until the output M goes to 0 closing the valve. However, since the error term is still negative, the bias will continue to decrease becoming more negative. When the PV finally does come back down below the SP, the valve will stay closed until the error e is positive long enough to cause the bias Mx to become positive. The controller is guaranteed to undershoot. A similar problem will occur if the error becomes positive for an extended period of time because the PV is less than the SP.

One way to solve the problem is to clamp the bias between 0.0 and 1.0. The controller does this. However, if this is the only thing that is done, then the output will not move off 0.0 (thus opening the valve), until the PV has become less than the SP. This will also guarantee an undershoot.

The controller solves the undershoot (and the corresponding overshoot) problem in one of two ways depending on how the loop is programmed—by freezing the bias term, or by actively back-calculating the bias term.

Freeze Bias When Output Goes Out of Range Option

If the “Freeze Bias When Output Goes Out of Range” option is selected, the controller simply stops changing the bias Mx whenever he computed output M goes outside the interval [0.0, 1.0]. When this option is selected, the computation of the output M and bias Mx is done as follows.

Trial bias term: $\overline{Mx_n} = K_i e_n + Mx_{n-1}$

Trial output: $\overline{M} = K_c e_n - K_r(PV_n - PV_{n-1}) + \overline{Mx}$

Actual output:

M_n if= 0.0 $\quad \overline{M} < 0.0$,

M_n if= $\overline{M}$ $\quad 0.0 \le \overline{M} \le 1.0$,

M_n if= 1.0 $\quad \overline{M} > 1.0$.

New bias term:

Mx_nif = $\overline{Mx_n}$ $\quad 0.0 \le \overline{M} \le 1.0$,

$M_n = Mx_{n-1}$ otherwise .

In this example, the bias will probably not go all the way to zero; when the PV *does* begin to come down, the loop will begin to open the valve sooner than it would, had the bias been allowed to go all the way to zero. This action has the effect of lessening the amount of overshoot.

Adjusting the Bias

The default action of the controller is a slightly more sophisticated operation which makes the computation of the bias term conditional on the computation of the output as follows:

Trial bias term $\overline{Mx} = K_i e_n + Mx_{n-1}$

Trial output $\overline{M} = K_c e_n - K_r(PV_n - PV_{n-1}) + \overline{Mx}$

Actual output $M_n = 0.0$ if $\overline{M} < 0.0$

$M_n = \overline{M}$ if $0.0 \leq \overline{M} \leq 1.0$ *(not wound up)*

$M_n = 1.0$ if $\overline{M} > 1.0$.

New bias term $Mx_n = \overline{Mx}$ if $0.0 \leq \overline{M} \leq 1.0$ (*no windup)*

$Mx_n = M_n - (K_c x e_n - K_r(PV_n - PV_{n-1}))$

if output $M_n \leq 0.0$

With this method, the output device begins to respond as soon as the PV begins to come down. If the loop is properly tuned, overshoot can be eliminated entirely. (This assumes that the setpoint is not changing. If the output went out of range due to a setpoint change, then the loop probably will oscillate because you must wait for the bias term to stabilize again.)

The choice of whether to use the default loop action or to freeze the bias is dependent on the application.

NOTE: If large step changes to the setpoint are anticipated, it is probably better to select the freeze bias option. Otherwise, the default action is best**.**

Eliminating Proportional, Integral, or Derivative Action

It is not always necessary (or even desirable) for full three-mode PID control of a loop. Parts of the PID equation may be eliminated by choosing appropriate values for the gain (K_c), reset (T_i) , and rate (T_d) thus yielding a P, PI, PD, I, and even an ID and a D loop.

Eliminating Integral Action. The inner loops in a cascade strategy probably do not need the extra bit of sluggishness and inertia brought in by integral action, and often are specified as Proportional-only loops. The units on the integral term are Minutes per Repeat and so the larger the number entered the smaller the contribution of the integral term; T = 9999.99 means very little integral action, while 0.01 means a lot. Integral action may be eliminated, then, by entering a very large number for the integral time.

Eliminating Derivative Action. The contribution to the output due to derivative action may be eliminated by setting T_d = 0.

Eliminating Proportional Action. The contribution to the output due to the proportional term may be eliminated by setting Kc = 0. Since K_c is also normally a multiplier of the integral coefficient (K_i) and the derivative coefficient (K_r), the SIMATIC 545 makes the computation of these values conditional on the value of K_c as follows:

$$K_i = K_c(T_s/T_i) \quad \text{if} \quad K_c \neq 0 \,,$$

$$ = T_s/T_i \quad \text{if} \quad K_c = 0 \,, \quad \textit{(for I or for ID control)}$$

$$K_r = K_c(T_d/T_s) \quad \text{if} \quad K_c \neq 0 \,,$$

$$ = T_d/T_s \quad \text{if} \quad K_c = 0 \,, \quad \textit{(for ID or D control)}$$

Position and Velocity Algorithm

The controller implements both the position and the velocity forms of the PID algorithm. For the position algorithm, the *position* of the device being controlled is computed based on the error. The velocity form of the PID algorithm computes the *change in the device position* based on the error.

PID Position Algorithm

For the position form of the PID equation, the controller output M_n is computed as follows.

$$M_n = Kc \times e_n + Ki \sum_{j=1}^{n} e_j - Kr\,(PV_n - PV_{n-1}) + M_0$$

Variable	Definition	Loop Variable Mnemonic
Ts	Sample rate	LTS
Kc	Proportional gain	LKC
Ki	Coefficient of the integral term: Kc × (Ts / Ti)	
Kr	Coefficient of the derivative term: Kc × (Td / Ts)	
Ti	Reset or integral time	LTI
Td	Derivative time or rate	LTD
SP	Setpoint	LSP
PV_n	Process Variable at nth sample	LPV
e_n	Error at nth sample: $SP - PV_n$	
M_0	Output at sample time 0	
M_n	Output at sample time n	LMN

The controller combines the integral sum and the initial output into a single term called the bias (Mx). This results in the following equations which define bias and output at sample time.

Bias At Sample Time n: $Mx_n = Ki \sum_{j=1}^{n} e_j + M_0$

Output At Sample Time: $M_n = Kc \times e_n - Kr\,(PV_n - PV_{n-1}) + Mx$

The following is an example of the computation done by the controller during a single sample period for a loop. The rate portion of the algorithm is usually used for special cases and is set to 0 in this example.

Variable	Definition	Value
Ts	Sample rate	1 second
Kc	Proportional gain	.01
Ti	Reset or integral time	1 minute
Td	Derivative time or rate	0
SP	Setpoint	.5
PV_n	Process Variable for this sample	.75
PV_{n-1}	Process Variable for previous sample	.77
e	Error for this sample: SP – PV	.5 – .75 = –.25
Mx	Bias	.5
Ki	Coefficient of integral term: Kc × (Ts / Ti)	.01 × (1s / 60s) = .00017
Kr	Coefficient of derivative term: Kc × (Td / Ts)	.01 × (0s / 1s) = 0

New Bias

$$= Mx = Ki \times e_n + Mx_{n-1}$$
$$= Mx = (.00017 \times -.25) + .5$$
$$= Mx = .4999$$

New Output

$$= M_n = Kc \times e_n - Kr \times (PV_n - PV_{n-1}) + Mx_n$$
$$= M_n = .01 \times -.25 - 0 \times (.75 - .77) + .4999$$
$$= M_n = .49746$$

The new bias is 0.4999 and the new output is 49.746%.

PID Velocity Algorithm

The standard PID equation computes the actual actuator position. An alternative form of the PID equation computes the change in actuator position rather than the actual position. This form of the equation is referred to as the velocity PID equation and is obtained by subtracting the equation at time (n–1) from the equation at time (n).

$$\Delta M_n = M_n - M_{n-1} =$$

$$Kc\left[(e_n - e_{n-1}) + \frac{Ts}{Ti} \times e_n - \frac{Td}{Ts}\left(PV_n - 2PV_{n-1} + PV_{n-2}\right)\right]$$

Variable	Definition
Mn	Loop output at the nth sample
Ti	Reset time
Kc	Proportional gain
Td	Rate Time
e_n	Error (SP–PV) at the nth sample
Ts	Sample time
PV_n	Process Variable at the nth sample

The output of the velocity equation usually should not be sent as-is to a WY location to drive a valve directly. This algorithm calculates how much to *change* the value position and so may be fed through some time-proportioning logic to produce forward or reverse drive pulses to a motor-driven valve.

B.2 Loop Alarms

Alarm Conditions

The controller allows the user to specify alarms conditions that are to be monitored for each loop. The alarms monitored are as follows:

- Broken Transmitter – This alarm occurs if the input Process Variable is outside of its valid range depending on the type of scaling performed (i.e., 0% offset, 20% offset, or bipolar).
- High-High – This alarm occurs if the PV rises above the programmed High-High Alarm Limit.
- High – This alarm occurs if the PV rises above programmed High Alarm Limit.
- Low – This alarm occurs if the PV falls below the programmed Low Alarm Limit.
- Low-Low – This alarm occurs if the PV falls below the Low-Low Alarm Limit.
- Yellow Deviation – This alarm occurs if the PV is further than the programmed Yellow Deviation Alarm Limit from the Setpoint.
- Orange Deviation – This alarm occurs if the PV is further than the programmed Orange Deviation Alarm Limit from the Setpoint.
- Rate of Change – This alarm occurs if the PV changes faster than the programmed Rate-of Change Alarm Limit.

Alarm Hysteresis

The controller also provides hysteresis on all alarms except the Rate-of-Change alarm to prevent them from chattering when the PV is near one of the alarm limits. This is done by allowing the user to specify an alarm deadband. The loop will not exit the alarm condition until the PV has come inside the alarm limit minus the deadband. This is shown graphically in Figure B-1.

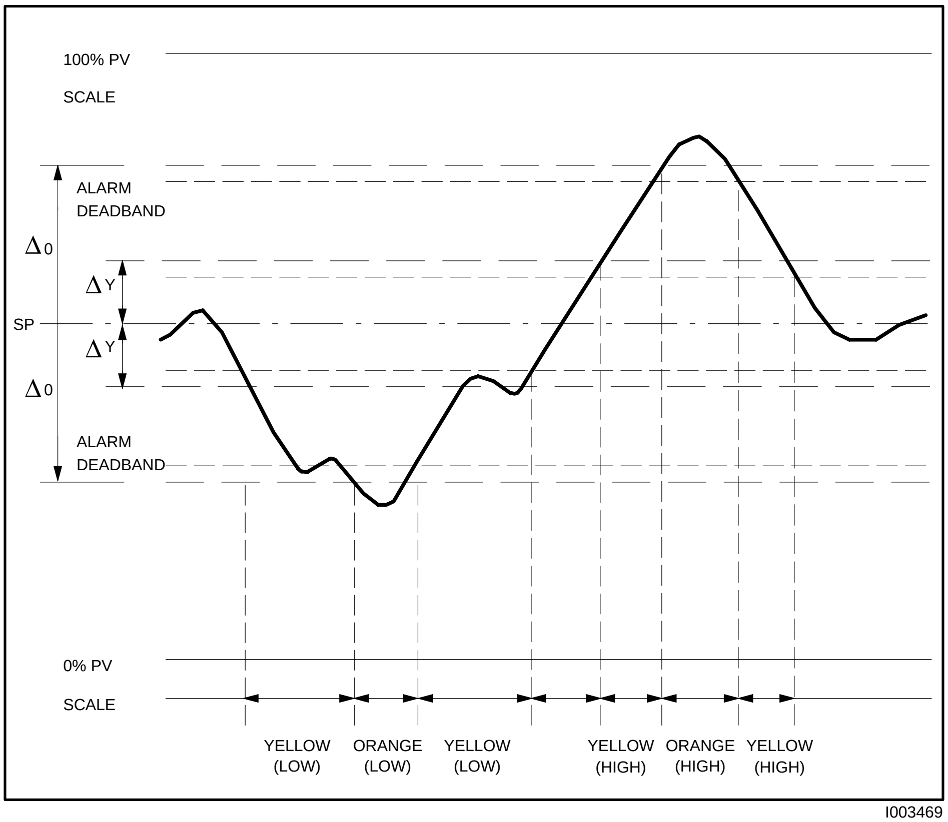


Figure B-1 Alarm Deadband

B.3 Loop Modes

While the controller is in RUN mode, the loops operate in one of three modes.

- Manual
- Automatic
- Cascade

Manual Mode

In this mode, the loop output is not calculated by the controller, but comes from the operator through an operator interface, like the CVU or TISTAR system. While a loop is in MANUAL, the controller still monitors the Broken Transmitter, High-High, High, Low, Low-Low, and Rate-of-Change alarms. The Yellow and Orange deviation alarms are not monitored.

Automatic Mode

In this mode, the controller computes the loop output. The setpoint (SP) for the loop comes from either an operator, Special Function Program, or from a RAMP/SOAK Table. RAMP/SOAK is discussed later in this document. All alarms are monitored.

Cascade Mode

In this mode, the controller computes the loop output. The setpoint for the loop comes from a user specified location called the remote setpoint. For truly cascaded loops, the remote setpoint is the output of another loop. The controller also allows the remote setpoint to be some other variable in the controller. (Such loops are not truly cascaded but the same terminology is used. The controller does some things for truly cascaded loops that are not done for a simple remote setpoint. These differences will be discussed later.) All alarms are monitored.

For cascaded loops, the loop whose output is used as the setpoint for another loop is called the outer loop; and the loop that uses the output of another loop for its setpoint is called the inner loop. Loops can be cascaded as necessary. For example, if you have 10 PID loops, you can cascade 9 of the loops (the final loop is the output to the field device).

If an inner loop of a cascade is placed in AUTO or MANUAL, then all of its outer loops must be placed in MANUAL to prevent reset windup. Similarly, an outer loop cannot be placed in AUTO until all inner loops are in CASCADE. This can become complicated, so the logic to handle opening and closing of cascades is built into the controller as follows:

1. If an inner loop is switched out of CASCADE then all of its outer loops are switched to MANUAL.
2. A request to place an outer loop in AUTO or CASCADE is denied unless the inner loop is in CASCADE.

If a loop is not truly cascaded but is simply using a remote setpoint, changes to and from CASCADE mode are allowed.

Mode Changes

The controller allows the loop mode to be changed by an Special Function program, ladder logic, or an operator interface device using task codes. While the loop may be requested to enter any mode from any other mode, the SIMATIC 545 actually only performs the following mode transitions:

MANUAL	→	AUTO
AUTO	→	CASCADE
CASCADE	→	AUTO
AUTO	→	MANUAL

The other requests (MANUAL → CASCADE and CASCADE → MANUAL) are handled as transitions to AUTO and then to the final mode as follows:

MANUAL → CASCADE	is done as	MANUAL → AUTO → CASCADE
CASCADE → MANUAL	is done as	CASCADE → AUTO → MANUAL

Bumpless Transfer of Control

The controller provides for bumpless mode changes (this means that the controller output does not change immediately after the mode change) as described below.

Position PID Algorithm

MANUAL to AUTO The Setpoint and Bias are initialized as follows:

$$SP \leftarrow PV$$

$$Mx \leftarrow M$$

AUTO to CASCADE The remote Setpoint (i.e., the output of the outer loop) is initialized as follows:

$$M_{outer\ \ loop} \leftarrow SP_{inner\ \ loop}$$

Velocity PID Algorithm

MANUAL to AUTO The Setpoint is initialized as follows:

$$SP \leftarrow PV$$

AUTO to CASCADE The Remote Setpoint (i.e., the output of the outer loop) is initialized as follows:

$$M_{outer\ \ loop} \leftarrow SP_{inner\ \ loop}$$

B.4 Special Computations on Output, PV, or Error

Forward and Reverse Acting Loop

The controller allows a loop to be programmed as reverse acting. With a reverse acting loop, the output is driven in the opposite direction of the error. When a loop is programmed to be reverse acting, the controller performs the initial loop calculation as follows:

NOTE: Programming a loop to be reverse acting is equivalent to programming a negative proportional gain.

$$\overline{Mx} = -K_i e_n + Mx_{n-1}$$

$$\overline{M} = -K_c e_n + K_r(PV_n - PV_{n-1}) + \overline{Mx}$$

Different manufacturers define forward and reverse acting controller responses in different ways. The definition used here describes the controller's output response to a change in Setpoint. A SIMATIC 565T controller with the forward-acting option specified will respond to an increase in Set Point with an increase in its output. A negative-acting loop will decrease its output when the Setpoint is increased.

Square Root of the Process Variable

The controller will let the user specify that the square root of the Process Variable is to be used instead of the raw process variable. This is useful when the input for the Process Variable is from a device (such as an orifice meter) that requires a square root calculation to determine the correct value to be used.

Error Squared Control

When error squared control is selected, the error (e_n) is calculated as:

$$\bar{e} = SP - PV_n$$

$$e_n = \bar{e} \text{ x } abs(\bar{e})$$

Since $e_n <= \bar{e}$, a loop using the error squared is less responsive than a loop using just the error. In fact, the smaller the error, the less responsive the loop. Error squared control would typically be used in a PH control application.

Error Deadband Control

When error deadband control is selected and YDEV is the yellow deviation alarm limit, the error is calculated as:

$$\bar{e} = SP - PV_n$$

$$e_n = 0 \quad \text{if} \quad abs(\bar{e}) < YDEV$$

$$= \bar{e} - YDEV \quad \text{if} \quad \bar{e} > YDEV$$

$$= \bar{e} + YDEV \quad \text{if} \quad \bar{e} < -YDEV$$

With error deadband control, no control action is taken if the PV is within the yellow deadband area around the setpoint.

Derivative Gain

There are many applications where the process dynamics call for the stabilization and braking action of derivative control. However, many applications are prevented from using derivative control because of noise on the process variable signal. If the process variable signal contains rapidly varying noise spikes of small amplitude, then the derivative term effectively amplifies those noise pulses and passes them directly to the output device.

The Derivative Gain Limiting option on the controller addresses this problem and allows the use of derivative control action in many cases. This option allows the user to specify a filter coefficient *Kd* which is also known as the Derivative Gain Limiting coefficient. When derivative gain limiting is used, loop calculation is modified as follows:

Position Algorithm

$$Y_n = Y_{n-1} + \frac{T_s}{Ts + (Td/Kd)}(PV_n - Y_{n-1})$$

$$\overline{Mx} = K_i e_n + Mx_{n-1}$$

$$\overline{M} = K_c e_n - K_r(Y_n - Y_{n-1}) + \overline{Mx}$$

Velocity Algorithm

$$Y_n = Y_{n-1} + \frac{T_s}{Ts + (Td/Kd)}(PV_n - Y_{n-1})$$

$$\Delta M_n = K_c(e_n - e_{n-1}) + K_i e_n - K_r(Y_n - 2Y_{n-1} + Y_{n-2})$$

This effectively places a first-order filter on the derivative term only, with a time constant of K_d sample times. The default value of K_d is 10, so then if the loop sample time were 0.5 sec., this would result in a net filter time constant of 5 seconds. Since this definition takes into account the loop sample time and only affects the derivative term the default value of K_d = 10 is good for most applications requiring derivative control.

Special Computations on Output, PV, or Error (continued)

Ramp/Soak

For each loop, the controller allows you to program a RAMP/SOAK table to indicate how the setpoint is to change with time. A RAMP/SOAK table consists of entries called steps: there are two types. In a RAMP step, the setpoint is changed from its current value to a user-specified final value at a user-specified rate. In a SOAK step, the setpoint is held constant for a user-specified period of time. For a SOAK step, you also specify that guaranteed soaking is desired by specifying a deadband around the setpoint. The Process Variable must be within this deadband limit of the setpoint in order for SOAK time to be measured.

Special Function Programming

At various points in the execution of a loop, the controller allows you to break out of the normal loop execution sequence to perform your own processing. The mechanism provided is through Special Function Programs. A Special Function Program is a sequence of BASIC-like statements that allow the user to manipulate controller variables, including all of the parameters to a loop.

The points in the execution of a loop in which an Special Function Program may be invoked are as follows.

- After reading and performing any indicated conditioning on the Process Variable. This type of Special Function Program would typically be used to perform filtering on the Process Variable. (SF on PV).
- Before using the Setpoint to compute the error. This type of Special Function Program would typically be used in a ratio-control application. (SF on SP).
- Before storing the output. (SF on output).

Appendix C

Task Codes

C.1 Task Code Parameter Descriptions

Task Codes are the command/response messages sent to the controller through the RS-232 and RS-422 communication ports (ports 1 and 2). Task Codes are used to perform the following functions:

- Program the controller
- Initiate modes of operation
- Perform diagnostics
- Read/Write controller data

This appendix only presents the Read/Write data Task Codes and applicable status Task Codes.

This section defines the symbols used to represent the various fields of the task codes. If the symbols used in the task code are not defined in the task code description, then the definitions given in this section apply. Symbols defined with the task code descriptions take precedence for that task code.

NOTE: Upper case and lower case of the symbols are significant.

C.2 Word Area Address Descriptors - AAAA, aaaa, and wwww

Word Codes

Task Codes that access user word areas within the controller require a Word Code to identify the desired value(s). Word codes are separated into three categories:

- Category 1 – Those that access memory types supported by a discrete machine such as the SIMATIC® 530C or SIMATIC® 560.
- Category 2 – Those that access Loop and Analog Alarm Variables supported only in batch machines such as the SIMATIC® 565, SIMATIC 545, and SIMATIC® 575.
- Category 3 – Those that access timeline variables supported by controllers with tunable timelines such as the SIMATIC 545 and the SIMATIC 575.

Word codes are represented by three different symbols as follows:

- AAAA – Represents a Word Code used in Task Codes initially intended to access memory types in Category 1 by earlier Series 500/505 controllers. Later controllers also use these task codes to access Category 2 and Category 3 word codes. AAAA may be either 16 or 32-bits long.
- wwww – Represents a Word Code used in Task Codes intended to access Loop and Analog Alarm Variables supported only in a batch machine such as the SIMATIC 565 and SIMATIC 545. If the most-significant bit is set then wwww is treated as the address of a real (32-bit) value. If the most-significant bit is reset then wwww is treated as the address of an integer (16-bit) value. The Word Code wwww may be either 16- or 32-bits long.
- aaaa – Represents a Word Code or an I/O Code used in Task Codes dealing with SF Programs where either an I/O Code or Word Code can specify an error address. If the most-significant bit of aaaa is reset then aaaa represents an I/O-code. If the most-significant bit of aaaa is set then aaaa represents a word-code. aaaa may be either 16- or 32-bits long.

The correspondence between the categories and word code representation for the task codes described in this appendix. See Table C-1.

Table C-1 Word Code Categories on SIMATIC 545

Task Code	Controller	
	Categories	Representation
01-02, 50-51	1,2,3	AAAA or wwww
5A, 7E, 7F	1,2,3,	wwww

Category I – Word Code Description

Category 1 user words within the controller are arranged in pages with page sizes dependent on the type of memory. Page sizes for each memory type are defined in Table C-2.

Table C-2 Memory Type Page Sizes

Word Code Definition	Page Size	First Location
0000 – V Memory	1024	0
0001 – K Memory	1024	0
0010 – DCP Memory	480	HEX 10
0011 – DCC Memory	1024	1
0100 – STW Memory	1024	1
0101 – G Memory	1024	0
0110 – Reserved (PSWA)	–	–
0111 – Undefined	–	–
1000 – TCP/TCC Memory	128	1
1001 – Reserved (Constant)	–	–
1010 – DSP/DSC Memory	30	1
1011 – Word Code Expansion with Expanded Offset		
1100 – WX Memory	1024	1
1101 – Word Code Expansion		
1110 – WX Memory	1024	1
1111 – Offset Expansion		

The first page (Page 0) is accessed by a single word format and the subsequent pages are accessed by a two word extended format as described below.

Where the format for Word Codes 0 thru E is:

MSB

bits –>	1	2–5	6–16
	0	Word Code	Word Offset

This format is used to access the first page of each type of word area. To access successive pages, Word Code F is used as the expansion code and the format becomes:

MSB

bits –>	1	2–5	6	7–16
	0	1 1 1 1	0	Ten Bit Page Number (Page)
	0	Word Code		Word Offset

Page and Offset The PAGE and OFFSET fields identify the specific data element (of the memory type specified by WORD CODE).

The encoding of the PAGE and OFFSET fields of the word address is dependent upon the value of the WORD CODE field. Equations using the following operations are given below for each memory type.

- a .DIV. b = integer—divide of a by b
- a .MUL. b = integer—multiply of a by b
- a .MOD. b = remainder of a .DIV. b

V/K/G Encoding

For V and K memory, the PAGE/OFFSET fields are determined by the following equations:

PAGE = (N-1).DIV.1024   OFFSET = (N-1).MOD.1024
(Where N is the data element number, e. g. 4073 in V4073.)

The data element number for G is determined by:

*N = Application * 32768 + Application Offset*
Application number for A = 1, B = 2, etc. (0 means current application)
Application Offset ranges from 1 to 32768.

Given PAGE and OFFSET, N is determined by the following:

N = (PAGE.MUL.1024)+1+OFFSET

DCP Encoding

For DCP memory the PAGE/OFFSET fields are determined by the following equations:

PAGE = (N-1).DIV.30   OFFSET = (((N-1).MOD.30)+1).MUL.16+(S-1)
Where N is the drum number and S is the drum step number.

Given PAGE and OFFSET, N is determined by the following:

N = (PAGE.MUL.30)+(OFFSET.DIV.16)

WX/WY/DCC/STW Encoding   For DCC, STW, WX, and WY memory, the PAGE and OFFSET fields are determined by the following equations:

PAGE = (N-1).DIV.1024
OFFSET = ((N-1).MOD.1024) + 1
(Where N is the data element number, e. g. 53 in WX53.)

Given PAGE and OFFSET, N is determined by the following:

N = (PAGE.MUL.1024)+OFFSET

TCP/TCC Encoding

The following equations specify the PAGE and OFFSET fields for word addresses referencing TCP/TCC memory.

NOTE: TCP and TCC memory share a single word code. Because of this, the encoding of the OFFSET field is used to distinguish between these word types.

PAGE = (N-1).DIV.128
OFFSET = (N-1).MOD.128 + Z
Where N is the Timer/Counter number and Z is 1 (TCP memory) or 129 (TCC memory).

Given PAGE and OFFSET, N is determined by the following:

N = (PAGE.MUL.128)+((OFFSET-1).MOD.128)+1

DSP/DSC Encoding

The following equations specify the GROUP and OFFSET fields for word addresses referencing DSP/DSC memory.

NOTE: DSP and DSC memory share a single word code. Because of this, the encoding of the OFFSET field is used to distinguish between these word types.

PAGE = (N-1).DIV.30   OFFSET = ((N-1).MOD.30) + Z
Where N is the drum number and Z is 1 (DSP memory) or 31 (DSC memory).

Given PAGE and OFFSET, N is determined by the following:

N = (PAGE.MUL.30)+((OFFSET-1).MOD.30)+1

Category 2 Word Code Description

Category 2 words are distinguished by bits 1 through 5 all set to one. To access loop and analog alarm data the following formats are used:

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
f	1	1	1	1	1	w	w	w	w	n	n	n	n	n	n

The single word format is used to access types defined by word codes 0-E in bits 7-10. Word code F in bits 7-10 is used to extend the addressing past 64 variables of a given type or to reach additional types. The double word format is as follows:

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
f	1	1	1	1	1	1	1	1	1	w	w	w	w	w	w
n	n	n	n	n	n	n	n	n	n	n	n	n	n	n	n

The fields in the category 2 word-code are as shown in Table C-3 thru C-5.

Table C-3 f – Variable Data Format

0	Integer
1	Real

Table C-4 wwww – Variable Data

0000 = T	Temporary
0001 = LPV	Loop Process Variable
0010 = LSP	Loop Setpoint
0011 = LMN	Loop Output
0100 = LMX	Loop Bias
0101 = LERR	Loop Error
0110 = LKC	Loop Gain
0111 = LTD	Loop Rate
1000 = LTI	Loop Reset
1001 = LVF	Loop V-flags
1010 = LRSF	RAMP/SOAK flags
1011 = APV	Analog Alarm Process Variable
1100 = ASP	Analog Alarm Setpoint
1101 = AVF	Analog Alarm flags
1110 = P	SF Subroutine Parameter
1111	Expansion code (see wwwwww)

Table C-5 wwwwww – Extended Variable Type

00xxxx	xxxx is the same as wwww above
010000	Constant. If f is 0 then the next word contains an integer constant. If f is 1 then the next two words contain a real constant.
010001 = LPVL	Loop Process Variable Low Limit
010010 = LPVH	Loop Process Variable High Limit
010011 = APVL	Analog Alarm Process Variable Low Limit
010100 = APVH	Analog Alarm Process Variable High Limit
010101 = LTS	Loop Sample Rate (seconds)
010110 = ATS	Analog Alarm Sample Rate (seconds)
010111 = LHA	Loop High Alarm Limit
011000 = LLA	Loop Low Alarm Limit
011001 = LODA	Loop Orange Deviation Alarm Limit
011010 = LYDA	Loop Yellow Deviation Alarm Limit
011011 = LSPL	Loop Setpoint Low Limit
011100 = LSPH	Loop Setpoint High Limit
011101 = LCFH	Most-significant word of Loop C-flags
011110 = LCFL	Least-significant word of Loop C-flags
011111 = LHHA	Loop High-High Alarm Limit
100000 = LLLA	Loop Low-Low Alarm Limit
100001 = LRCA	Loop Rate-of-Change Alarm Limit (engineering units / minute)
100010 = LADB	Loop Alarm Deadband
100011 = AHA	Analog Alarm High Alarm Limit
100100 = ALA	Analog Alarm Low Alarm Limit
100101 = AODA	Analog Alarm Orange Deviation Alarm Limit
100110 = AYDA	Analog Alarm Yellow Deviation Alarm Limit
100111 = ASPL	Analog Alarm Setpoint Low Limit
101000 = ASPH	Analog Alarm Setpoint High Limit
101001 = ACFH	Most-significant word of Analog Alarm C-flags
101010 = ACFL	Least-significant word of Analog Alarm C-flags
101011 = AHHA	Analog Alarm High-High Alarm Limit
101100 = ALLA	Analog Alarm Low-Low Alarm Limit
101101 = ARCA	Analog Alarm Rate-of-Change Alarm Limit (engineering units/ minute)
101110 = AADB	Analog Alarm Alarm Deadband
101111 = AERR	Analog Alarm Error
110000 = SFEC	Special Function Program Error Code
110001 = LKD	Loop Derivative Gain-limiting coefficient

Table C-5 wwwwww – Extended Variable Type (continued)

110010 = LRSN	Loop RAMP/SOAK Step Number
110011	Expression (see below)
110100 = X	Discrete input, accessed as a word
110101 = Y	Discrete output, accessed as a word
110110 = C	Control relay, accessed as a word
110111 = LACK	Loop Alarm / Alarm Acknowledge flags
111000 = AACK	Analog-alarm Alarm / Alarm Acknowledge flags
111001 = LPET	Loop Peak Elapsed Time Value - Represents the elapsed time from when the process is scheduled until it completes execution (545 only)
111010 = APET	Analog Alarm Peak Elapsed Time Value - Represents the elapsed time from when the process is scheduled until it completes execution (545 only)
111011 = PPET	SF PGM Peak Elapsed Time Value - Represents the elapsed time from when the process is scheduled until it completes execution (545 only)
111100-111110	unassigned reserved
111111	illegal

nn...n - Variable number 1 equals the first variable

The format for an expression is as follows:

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
f	1	1	1	1	1	1	1	1	1	1	1	0	0	1	1
Length of Following Expression (bytes)															
MATH or IMATH expression															

where:
f indicates the type of expression

1 indicates that the expression is a MATH expression
0 indicates that the expression is an IMATH expression

Category 3 Word Code Description

Category 3 Word Codes allow configuration of the SIMATIC 545 Timeline. Two of the spare category 1 word codes are used to provide expansion to Category 3 words. The first word code used (1101) indicates an 11 bit data type identifier and a 16 bit offset, while the second code (1011) indicates an 11 bit identifier and a 32 bit offset. In both cases, the first offset is 0.

The 2-word format is:

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
f	1	1	0	1		W			W				W		
n	n	n	n	n	n	n	n	n	n	n	n	n	n	n	n

and the 3 Word Code format is

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
f	1	0	1	1		W			W				W		
n	n	n	n	n	n	n	n	n	n	n	n	n	n	n	n
n	n	n	n	n	n	n	n	n	n	n	n	n	n	n	n

The fields in the word-code are as follows:

f - Integer/Real Flag A 0 in the MSB of the first word indicates integer. A 1 in the MSB of the first word indicates floating point and thus restricts that variable to S memory only.

WWW 11 bit word code (3 digit hex number) which specifies one of the following variables.

000 = FV = Fixed/Variable Scan Type selection. The offset (nnn...n) for this variable is meaningless and therefore set to zero. FV(0) may contain one of three valid values as follows (note this is the RRRR field in task code 01 or 02):

0000 =	Fixed Scan
0001 =	Variable Scan
0002 =	Variable Scan with Upper-Limit.
0003-FFFF =	invalid ; previous setting retained.

Battery bad power up shall always default to Variable Scan. When this variable is allowed to be modified by RLL, an invalid value will default to Variable Scan.

001 = Timeline configuration parameters. These are programmed by the user to select how often the scan is repeated and the maximum time spent in each part of the timeline. Battery bad power up default values shall be specified by the individual controller product. Each parameter is a 16 bit integer and is represented by a different offset (nnn...nn) as follows:

0000 = DS	DS - Discrete Scan Time = 1-255 ms (Valid for FV = 0 or 2). This value specifies how often the I/O Cycle, RLL, SF Module Cycle and Guaranteed Comm is performed.
0001 = LS	Loop Time Slice = 0-255 ms
0002 = AS	Analog Alarm Time Slice = 0-255 ms
0003 = CS	Cyclic SF PGM Time Slice = 0-255 ms
0004 = PS	Priority SF PGM Time Slice = 0-255 ms
0005 = NS	Normal SF PGM Time Slice = 0-255 ms
0006 = SS	Ladder SF Subroutine Time Slice = 0-255 ms
0007 = CN	Normal Communication Time Slice = 0-255 ms
0008 = CP	Priority Communication Time Slice = 0-255 ms
0009 = RS	Ladder SF Subroutine 0 Time Slice = 0-255 ms

002-7FE: undefined

7FF: Reserved for expansion

NOTE: Writing a value of 0 to a Timeline parameter shall be interpreted as no change to the existing value.

C.3 Task Code Definition

TC 01
Read Word Memory Random

Format: 01 AAAA [AAAA] ...
Response: 01 RRRR [RRRR]

Response values are returned in respective address positions.

NOTE: For SIMATIC 545 and SIMATIC 575 controllers, the *wwww* wordcode-descriptor can be used instead of the *AAAA* in which case the data descriptor *dddd* replaces *RRRR*.

TC 02
Write Word Memory Area Random

Format: 02 AAAA RRRR [AAAA RRRR] ...
Response: 02

NOTE: For SIMATIC 545 and SIMATIC 575 controllers, the *wwww* wordcode-descriptor can be used instead of the *AAAA* in which case the data descriptor *dddd* replaces *RRRR*.

TC 30
Read Operational Status

Format: 30 [xx]
Response: 30 UUUU UUUU ...

where xx = not coded = Return STATUS WORDS 1 thru 15.
= 00 = Return STATUS WORDS 1 thru 15
= 01 = Return STATUS WORDS 16 thru 30.

Response contains up to 15 status words depending upon the controller, with the following definition.

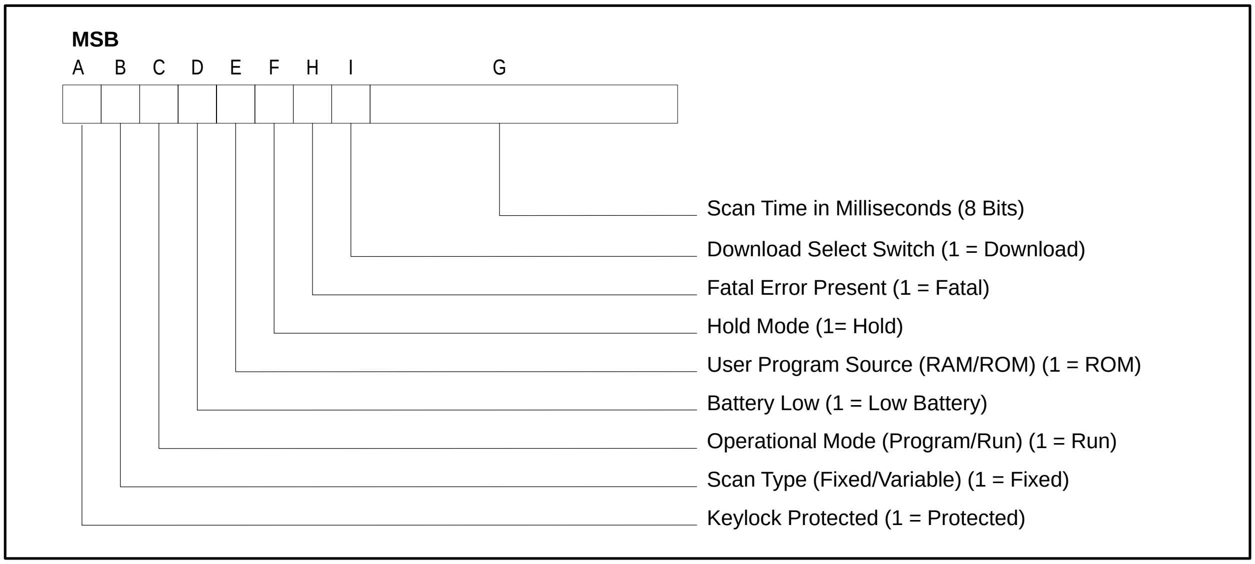


Figure C-1 Status Word 1

The scan time reported in field G is modulo 256 in earlier controller releases.

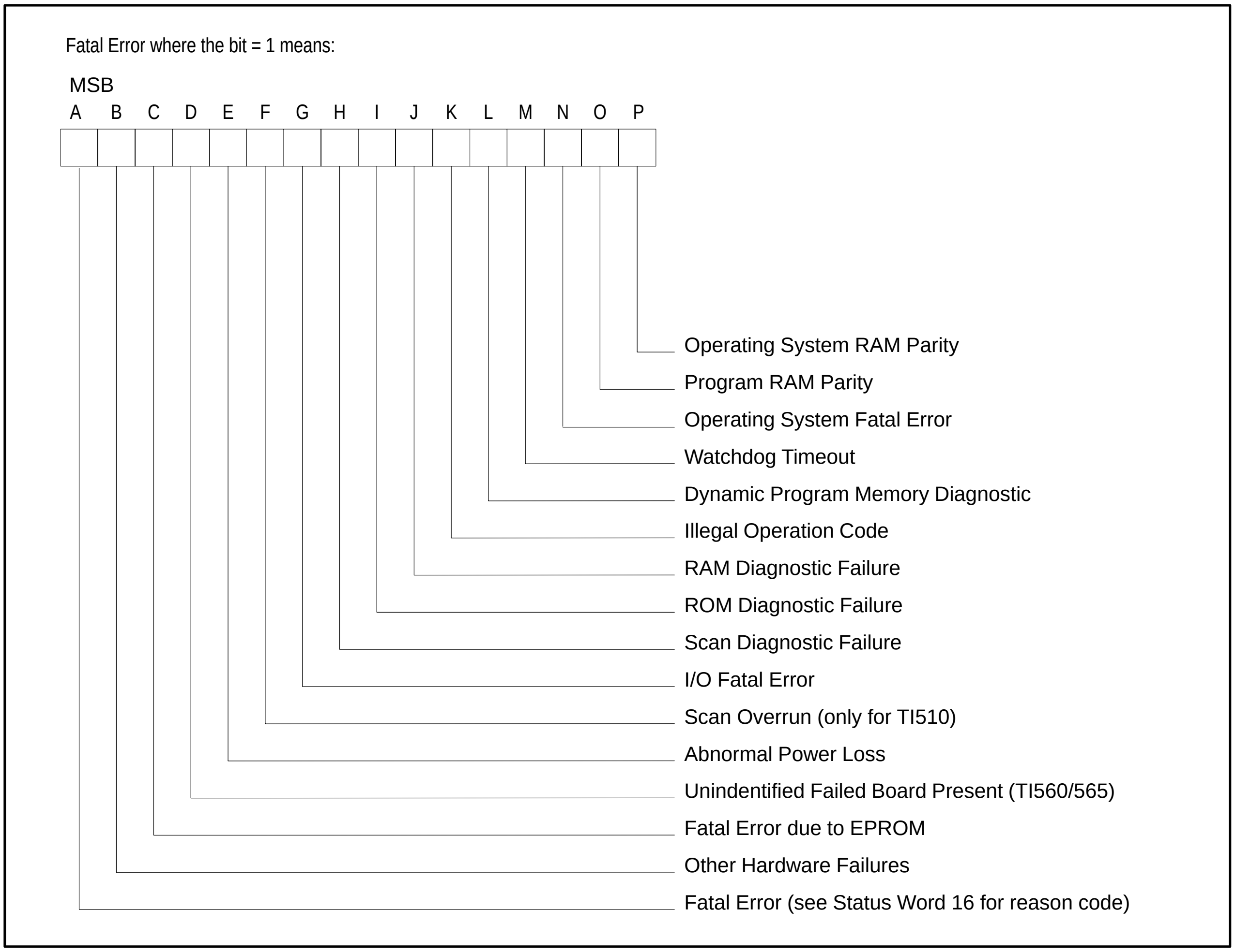


Figure C-2 Status Word 2

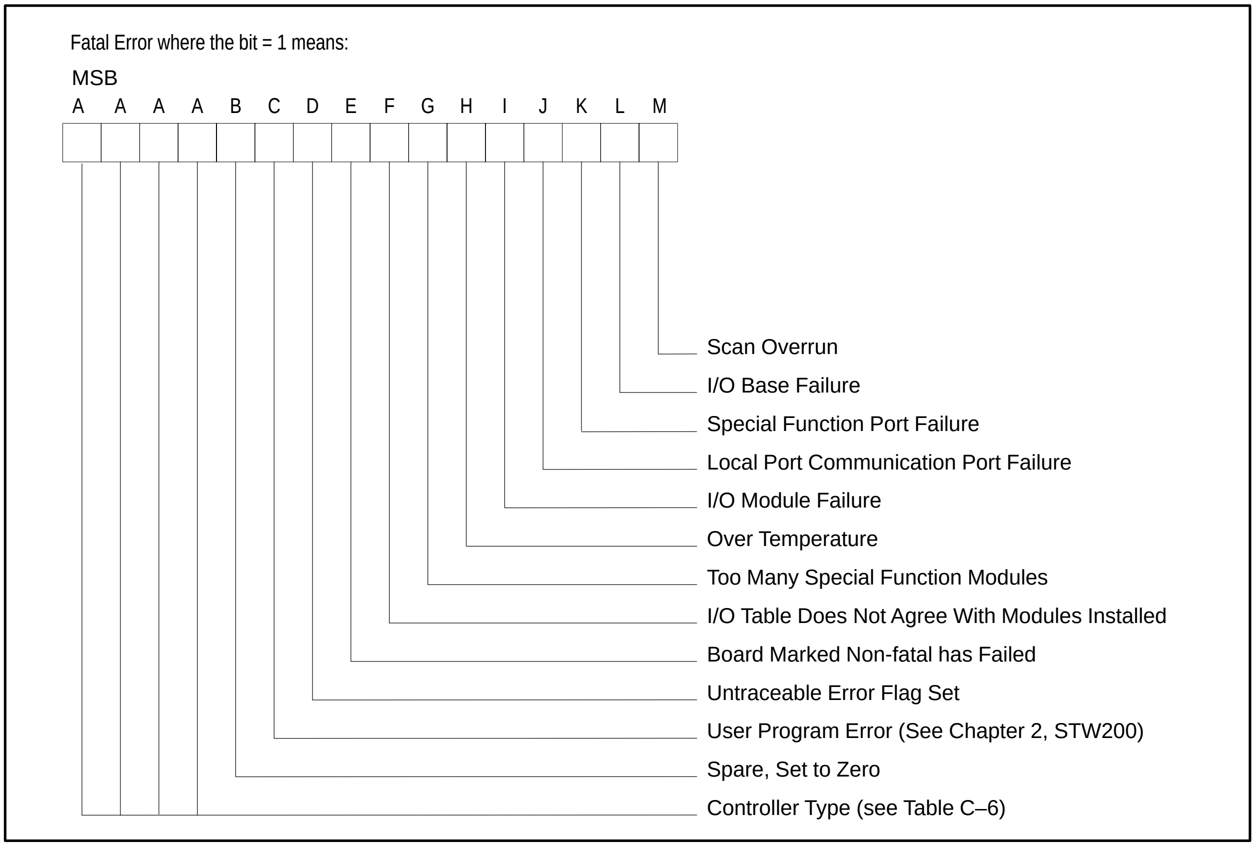


Figure C-3 Status Word 3

Table C-6 Controller Type

0000 – 5TI
0001 – 510
0010 – 520/525
0011 – 530/535
0100 – 540/545
0101 – 550
0110 – 560/565
0111 – 570
1000 – 570
1000 – 1101 = Future Expansions
1110 – Non Programmable Controller
1111 – PM550

Status Word 4: The Ladder Logic Memory Size. (See also Status Word 7).

Status Word 5: User V-memory Size. (See also Status Word 12.)

Status Word 6: Highest configured I/O Point.

Status Word 7: Most significant 16 bits of the Ladder Logic Memory Size. Concatenated with Status Word 4 to indicate memory sizes larger than 65,535 words.

Status Word 8: Remaining I/O points not configured.

Status Word 9: Expanded Controller Type Identification.

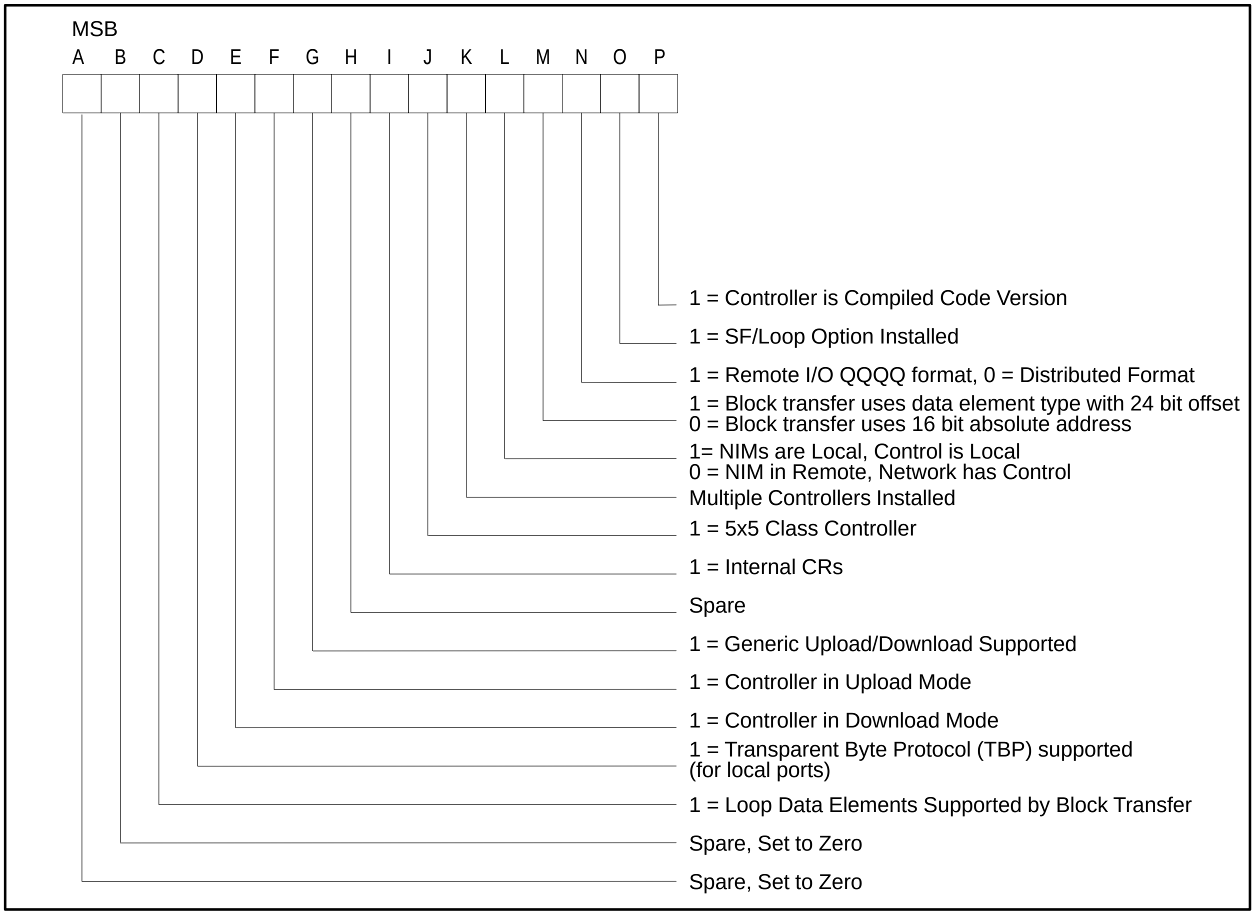


Figure C-4 Status Word 9

Status Word 10: HBU Mode.

0001 = Active with no Standby (Stand Alone)
0002 = Active with Off-line Standby
0003 = Active with On-line Standby
0004 = Standby On-line with healthy Active
0005 = Standby Off-line with healthy Active

Status Word 11 (Valid only if Word 10 = 2 or 5)**:** Reason for Off-line.

0000 = No Special Reason (entered on power up)
0001 = Off-line due to Hardware Mismatch
0002 = Off-line due to User Command
0003 = Off-line due to Active unit in PROGRAM mode
0004 = Off-line requesting On-line but Inhibited by user program in Active unit
0005 = Off-line due to failure in Standby
0006 = Off-line due to loss of HBU Communications

Status 12: Most significant 16 bits of the User V-memory size. Concatenated with Status Word 5 to indicate memory sizes larger than 65535 words.

Status Word 13 and 14: Operational status of each card in a multicard chassis. The numbered bits below represent the corresponding physical slot within the controller. If the bit is 0, a card is installed and functional. If the bit is 1, the card is failed or not present.

MSB WORD 13 LSB

x	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1

WORD 14

x	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16

Status Word 15: Peak discrete scan for a SIMATIC 545™
or SIMATIC 575. It has no meaning for the other controllers. It is the peak time required to execute the I/O Cycle, RLL and SF Cycle for a SIMATIC 545 or SIMATIC 575.

Status Word 16: Provides the reason for fatal error if bit “A” in Status Word 2 is set. Error numbers range from 0 to 65535.

Status 17 thru 30: Undefined and are set to zero.

TC 50
Read User Word Area Block

Format: 50 AAAA
Response: 50 RRRR [RRRR] ...

As many locations are returned as will fit in the task code length or until the end of the memory type is reached.

NOTE: For the SIMATIC 545 and SIMATIC 575 controllers, the *wwww* wordcode-descriptor can be used instead of the *AAAA* in which case the data descriptor *dddd* replaces RRRR.

TC 51
Write User Word Area Starting at Address

Format: 51 AAAA RRRR [RRRR] ...
Response: 51

As many locations can be written as will fit in the task code length or until the end of the memory type is reached.

NOTE: For the SIMATIC 545 and SIMATIC 575 controllers, the *wwww* wordcode descriptor can be used instead of the *AAAA* in which case the data descriptor *dddd* replaces RRRR.

TC8
Set Controller Time of Day Clock

Format: 58 GGGG HHHH IIII JJ
Response: 58

where GGGG = Year/Month =

bit	1	8	9	16
	xxxx	xxxx	xxxx	xxxx
	tens	ones	tens	ones
	Year		Month	

HHHH = Day/Hour =

bit	1	8	9	16
	xxxx	xxxx	xxxx	xxxx
	tens	ones	tens	ones
	Day		Hour	

IIII = Min/Sec =

bit	1	8	9	16
	xxxx	xxxx	xxxx	xxxx
	tens	ones	tens	ones
	Minute		Second	

JJ= Day of Week =

bit	1	8
		xxxx
	spare	

TC 5A
Write Block

Format: 5A wwww dddd dddd ... dddd
Response: 5A

Description: Write block beginning at the address specified by wwww.

NOTE: When writing loop or analog alarm data, the data corresponding to undefined loops and analog alarms is ignored. It must be present in the block as a placeholder.

TC 71
Read Controller Time of Day Clock

Format: 71
Response: 71 GGGG HHHH IIII JJ

For GGGG HHHH IIII JJ definition, see Task Code 58.

TC 7D
Read SF/Loop Processor Mode

Format: 7D
Response: 7D 00*mm ffff nnnn ssssssss rrrrrrrr wwww xxxx yyyy zzzz*

Description: TC7D returns the current operational mode of the Loop/SF processor (or the Loop/SF function on single board controllers such as the SIMATIC 545). The fields returned are as follows:

mm	Mode Descriptor. 1... .... Loop card is following CPU's operational mode .1.. .... Loop card is in PROGRAM mode. ..1. .... Loop card is in RUN mode. ...1 .... Loop card is in HOLD mode. .... 1... Loop card is in FAULT mode. .... .000 Unused.
ffff	Fatal Error vector
nnnn	Non-fatal error vector
xxxxxxxx	Size of S-memory in bytes.
rrrrrrrr	Number of bytes of S-memory available.
wwww	Maximum number of loops supported.
xxxx	Maximum number of analog-alarms supported.
yyyy	Maximum number of SF Programs supported.
zzzz	Maximum number of SF Subroutines supported.
vvvv	Control Block ID of the last undefined SF Program or SF Subroutine that the user tried to execute. 0000 is returned if no attempt has been made to execute an undefined SF Program or SF Subroutine.
rrrr	Control Block ID of the last restricted SF Program that the user tried to invoke from RLL. 0000 is returned if no attempt has been made to invoke a restricted SF Program from RLL.
qqqq	Control Block ID of the first disabled control block in S-memory. 0000 is returned if there are no disabled control blocks.

TC 7E
Read Random

Format: 7E wwww [wwww] ... [wwww]
Response: 7E dddd dddd ... dddd

Description: Read values given the the wwww fields. wwww may be either normal Series 500 word-codes or SF/Loop word-codes.

NOTE: Real values for undefined loops or analog alarms are returned as NANs. Integer values for undefined loop or analog alarms are returned as zero.

If an error occurs when reading a variable, TC7E will return an error response as follows:

00 *ee wwww*

where ee is the error code and wwww is the word-code on which the error occurred. Note that wwww may be 16-, 32-, or 48-bits long. If the error occurred because wwww was a partial word-code at the end of the task code then TC7E will return a full-sized word-code padded on the end with zeros.

TC 7F
Read Block

Format: 7F *wwww* [nn]

Response: 7F *dddd dddd ... dddd*

Description: Read nn values beginning at the address specified by *wwww*. *wwww* may be either a normal Series 500 word-code or an SF/Loop word-code. If nn is not specified than as many values as will fit in the task code buffer are returned.

NOTE: Real values for undefined loops or analog alarms are returned as NANs. Integer values for undefined loops or analog alarms are returned as zero.

TC 88
Select Number of SF Module Task Codes Per Scan

Format: 88 CN [CN] ...
Response: 88

Where: C = Channel Number = 1, 2, 3, ... 8 or F
N = Number of Task Codes per Scan = 1, 2, 3, ... 8

NOTE: This allows a different number of task codes per scan for each channel, but all SF modules on that channel will be allowed the same number. If an un-installed channel is programmed, a range error is returned.

The hex value of F for the channel number is a wild card and indicates the I/O channel over which the request is received. If this request is from any comm port (local or remote) other than an SF module, it will be rejected as containing invalid data.

Task Code Definition (continued)

TC 89
Read Number of SF Module Task Codes Per Scan

Format:	89 [CO] ...
Response:	89 CN [CN]

Where: C = Channel Number = 1, 2, 3, ... 8 or F
N = Number of Task Codes per scan = 1, 2, 3, ... 8

NOTE:

- Multiple CNs are returned in the order of the coded COs.
- If CO is note-coded in the request, then one CN is returned for each of the installed channels in order of channel number (2 per RCC).
- If the channel is coded as F, the N for the channel over which the request is received is returned.
- If the requester is on a local port and codes the channel as F, then an invalid data error is returned.
- If an uninstalled channel is coded, a range error is returned.

The following restart/reset conditions apply.

- Cold / Warm / Hot Restarts : Use previous value
- Power Up with Low Battery: Default to 2 task codes per scan
- PLC Clear: Default to 2 task codes per scan

Appendix D
Communication Protocols

D.1 Overview

The SIMATIC 545 supports two serial communication protocols on the RS-232 and RS-422 communication ports (ports 1 and 2):

- Non-intelligent Terminal Protocol (NITP)
- Transparent Byte Protocol (TBP).

D.2 Non-intelligent Terminal Protocol (NITP)

NITP Format

NITP is a simple, character-oriented method of data link communications using standard 7-bit ASCII codes. Both command and response messages consist of starting and ending delimiters, a character count or message length field, the body of the message, and an error-checking code field as shown in Figure D-1.

:	Character Count	Message Body	ECC	;

Figure D-1 Command and Response Messages

NITP Character Set

NITP uses the subset of standard ASCII codes, shown in Table D-1, to communicate with a wide variety of host devices, from ASCII terminals to more intelligent machines.

Table D-1 Standard ASCII Codes

7–bit ASCII Code	Displayed Character	7–bit ASCII Code	Displayed Character
30	0	39	9
31	1	3A	:
32	2	3B	;
33	3	41	A
34	4	42	B
35	5	43	C
36	6	44	D
37	7	45	E
38	8	46	F

NOTE: ASCII characters other than those in the NITP character set (such as a carriage return or line feed) may be sent between the ending delimiter and the next beginning delimiter to control special network devices. The controller ignores these characters.

Hexadecimal values must be translated into two ASCII codes or characters: For example, 0E (hex) equals 30 (ASCII) 45 (ASCII)

Non-intelligent Terminal Protocol (NITP) (continued)

Message Delimiters

A colon (:) marks the beginning of a message and a semicolon (;) marks the end of a message. Any characters between a colon and the next semicolon are interpreted as a valid message, while any characters between a semicolon and the next colon are ignored. This allows the host to use any parameters required by its software between lines of output. When transmitting data to the host, the UNILINK Host Adapter sends carriage return and line feed characters after the terminating semicolon to scroll the response on ASCII terminals. More intelligent host devices can be set to filter out the carriage return and line feed.

Character Count

To aid in error control, the colon (:) at the beginning of the message is followed by a two-character count field representing the total number of printable characters in the message, including the colon, character count, message body, error checking code, and the terminating semicolon. The two characters represent an 8-bit hexadecimal value such that a count of 50 characters will be represented as 3332 (32 hex). Default length of NITP messages are 72 characters.

Message Body

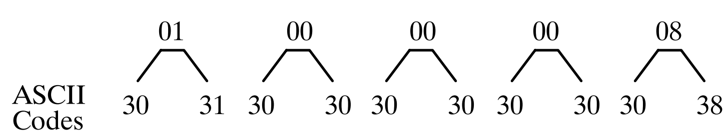
The message body consists of ASCII character pairs from the NITP character set representing a single binary byte value. The binary values from each character pair is the adapter command string.

Error-checking Code

Following the message body is an ASCII four-character error-checking code (ECC) in the form of a 16-bit hexadecimal number that is included at the end of the message just before the semicolon terminator. The ECC is a checksum computed by both the sending and receiving stations:

1. Divide the character count and the message body into blocks of four characters, left-justified and zero-filled. The beginning and ending delimiters are not included in the calculation.
2. Treat each block as a four-digit hexadecimal number.
3. Sum the resulting numbers (blocks).
4. Take the two's complement of the sum to get the ECC.

As an example, consider a message whose body is the Read Word Memory Random Task Code (TC01) reading V1 and V9:



The total character count is calculated by adding together the number of characters in the message, the four characters for the ECC, the two characters for the character count, and the two characters used to delimit the message. The total character count for a message containing "0100000008" is:

Starting Delimiter	:	=	01	(1)*
Character Count		=	02	(2)
Message Body – 0100000008		=	0A	(10)
Error-checking Code		=	04	(4)
Ending Delimiter	;	=	01	(1)
Total Character Count			12	(18)

* Numbers in parentheses are the corresponding decimal values.

After determining the total character count, insert the character count at the beginning of the message body as "120100000008". The ECC is given by:

Two's complement of (1201 + 0000 + 0008) = EDF7

so that the complete message is the character string:

:120100000008EDF7;

or

ASCII CODES	3A	31	32	30	31	30	30	30	30	30	30	30	38	45	44	46	37	3B
CHARACTERS	:	1	2	0	1	0	0	0	0	0	0	0	8	E	D	F	7	;

NOTE: If the number of characters in the concatenated character count and message body string is not evenly divisible by 4, then the fill characters "00" are added to the end of the string for use in the ECC calculation. These fill characters are not actually placed in the final message.

Table D-2 summarizes the NITP message structure:

Table D-2 NITP Message Structure

Field	Contents	No. of Characters
Beginning delimiter	Colon (:)	1
Character count	Hexadecimal	2
Body of Message	Hexadecimal	72 (default)
Error Checking Code	Hexadecimal	4
Terminator	Semicolon (;)	1

D.3 Transparent Byte Protocol

Transparent Byte Protocol (TBP) is a message protocol for point-to-point communications using eight bit characters transmitted serially. Compared to NITP, TBP improves the efficiency of the information exchange between a controller and an operator interface device.

UART Initialization Requirements

In order to support TBP information exchange, the communications controlling device (Universal Asynchronous Receiver-Transmitter, or UART) is initialized to the following conditions:

- 8 data bits per character
- no parity
- 1 start bit
- 1 stop bit

Message Format

The TBP message format is implemented as:

3F RL LL TC dd dd ... dd CK SM

The 3F or ASCII question mark is the first character of all TBP messages. The first byte serves as the protocol identifier which allows the NITP messages to be distinguished from the TBP messages. Therefore, any device supporting TBP initializes the UART as described above. If the protocol is NITP, the first byte received is hexadecimal BA or the ASCII colon sent with odd parity. If the protocol is TBP, the first byte is hexadecimal 3F or the ASCII question mark. Any other first character causes the message to be discarded.

Initiator

The initiator of a request message may choose the protocol (NITP or TBP) that is to be used. The responder must respond to the initiator in like protocol. In the case that the request was made using TBP and the responder does not support TBP, no response to the request is made.

Responder

For the responder which supports TBP, a request that is made using NITP is acknowledged using NITP. Since the NITP character set is limited to 20 characters, the responder makes provisions to check parity even though the UART is set up as defined above.

RL Byte

The RL byte defines the request/response maximum length in bytes. In a message that is a task code request, the RL byte specifies the maximum length of the task code response. In the task code request the RL byte is a hexadecimal number ranging from 6 (hexadecimal 06) to 256 (hexadecimal 00) with 255 being represented by hexadecimal FF. In the task code response the RL byte defines the maximum length of response that the device supports. Only two values are allowed for the RL byte in the response. The values are 73 (hexadecimal 49) or 256 (hexadecimal 00).

With this definition, a responding device with a maximum buffer size of 72 bytes indicates this restriction to the requesting device. In the event that the request overflowed the 72 byte buffer length, the responding device returns error code hexadecimal 09 (incorrect amount of data sent with request).

The requesting device is able to use the RL byte to restrict the amount of data that the responding device returns. If the requesting device has a buffer length of 72 bytes, then the requestor does not request a response longer than 73 bytes. The requesting device may also use the RL byte to specify a short response where only a small amount of data is required. For instance, if the current status (program or run mode and fatal error present or not) of the responding device is required by the requestor, sending task code 30 with the RL byte set to hexadecimal 07 causes the responder to return only the first byte of the otherwise 30-byte response which contains this essential information.

LL Byte

The LL byte defines the message byte count including the 3F and the CKSM. The byte count is a hexadecimal number ranging from a minimum of 6 (hexadecimal 06) to a maximum of 256 (hexadecimal 00) with 255 being represented by hexadecimal FF.

TC Byte

The TC byte is a hexadecimal number ranging from 00 to FF which identifies the task code command/response message.

The dd dd ... dd bytes is hexadecimal numbers representing the data associated with the task code request/response.

The CK and SM bytes forms hexadecimal numbers ranging from 0 to 255. The checksum bytes is generated by the sending device utilizing the Fletcher checksum ones complement as defined by the following equation. The exclamation point preceding the sum of the terms in parentheses in the equation for CK means to take the one's complement of the resulting sum.

CK = !(B1 + B2 + B3 + ...+ Bn + SM)
SM = (n)B1 + (n - 1)B2 + (n - 2)B3 + ... + Bn

When a carry results from adding two bytes, the carry is added into the sum or the sum is incremented by one. For instance if the message consisted of the following bytes:

3F 49 06 30
CK would be calculated as !(3F + 49 + 06 + 30 + 15) = 2C.
SM would be calculated as 4*3F + 3*49 + 2*06 + 30 = 15.
The transmitted message would become: 3F 49 06 30 2C 15.

The receiver calculates both the CK and the SM checksum bytes as shown in the following equations.

CK = B1 + B2 + B3 + ...+ Bn
SM = (n)B1 + (n - 1)B2 + (n - 2)B3 + ... + Bn

When a carry results from adding two bytes, the carry is added into the sum or the sum is incremented by one. For the transmitted message from the above example, the receiver would produce the following results:

CK would be calculated as 3F + 49 + 06 + 30 + 2C + 15 = FF.

SM would be calculated as 6*3F + 5*49 + 4*06 + 3*30 + 2*2C + 15 = FF.

If the checksum calculated by the receiver does not equal hexadecimal FF, an error has occurred and the message is discarded.

Message Timeouts

The receiver starts a timer upon receipt of the first character of a message. If the entire message is not received before the timer times out, the message is discarded and the receiver is initialized to receive the next message. The timeout value is baud rate dependant and message length dependant. The values for the maximum length message are shown in Table D-3, along with the baud rate and character times.

Table D-3 Maximum Length Message Values

Baud Rate	Character Time (ms)	Message Timeout Value (sec)
19,200	0.521	0.150
9,600	1.042	0.300
2,400	4.167	1.200
1,200	8.333	2.400
300	33.333	9.600

Message Turn Around Time

After receipt of a message, the response is not initiated for 0.25ms. This turn-around time is allowed to ensure that the transmitting device has had time to set up the receiver. The same delay applies between receipt of the response and the beginning of a new message.

Minimum Time Between Requests

After a request message has been sent, the requester waits a minimum of one message timeout plus one turn around delay before attempting another transmission. This allows the receiver to timeout the first message and re-initialize to receive. The time required to execute the request is determined separately and may be much longer.

Application Note for Parity Checking NITP Message

Since a device that supports TBP must also support NITP, lookup tables can be used to avoid having to reprogram the UART to send and receive 7 bits of data with odd parity. For transmission, the hexadecimal "nibble" value can be used as an index into the table which contains the hexadecimal representation of the ASCII character. The values of the ASCII characters used in NITP are listed in Table D-4.

Table D-4 ASCII Character Values for NITP

ASCII Character (or "Nibble")	Hexadecimal Representation With Parity
0	B0
1	31
2	32
3	B3
4	34
5	B5
6	B6
7	37
8	38
9	B9
A	C1
B	C2
C	43
D	C4
E	45
F	46
:	BA
;	3B
<CR>	0D
<LF>	8A

For reception, make a copy of the received character, mask the parity bit, and convert it to a hexadecimal nibble. Using this nibble value as an index into the transmit table, compare the received character with the table value. If they do not match, an error has occurred.

Customer Registration

We would like to know what you think about our user manuals so that we can serve you better. How would you rate the quality of our manuals?

	Excellent	Good	Fair	Poor
Accuracy	_______	_______	_______	_______
Organization	_______	_______	_______	_______
Clarity	_______	_______	_______	_______
Completeness	_______	_______	_______	_______
Overall design	_______	_______	_______	_______
Size	_______	_______	_______	_______
Index	_______	_______	_______	_______

Would you be interested in giving us more detailed comments about our manuals?

☐ **Yes!** Please send me a questionnaire.

☐ **No.** Thanks anyway.

Your Name: ______________________________

Title: ______________________________

Telephone Number: (_____) ______________________________

Company Name: ______________________________

Company Address: ______________________________

____

____

Manual Name: SIMATIC 545 Technical Product Description **Edition:** Third
Text Assembly Number: 2592506–0003 **Date:** 09/92
Order Number: PPX:545–8103–3

FOLD

SIEMENS ENERGY & AUTOMATION INC
3000 BILL GARLAND ROAD
PO BOX 1255
JOHNSON CITY TN 37605–1255

NO POSTAGE
NECESSARY
IF MAILED
IN THE
UNITED STATES

BUSINESS REPLY MAIL

FIRST CLASS PERMIT NO.3 JOHNSON CITY, TN

POSTAGE WILL BE PAID BY ADDRESSEE

ATTN TECHNICAL COMMUNICATIONS M/S 519
SIEMENS ENERGY & AUTOMATION INC
PO BOX 1255
JOHNSON CITY TN 37605–1255

FOLD

SIMATIC is a registered trademark of Siemens AG.

APT, CVU, Peerlink, Series 305, Series 405, Series 505, Stage, TISOFT, TISTAR, TIWAY, and Unilink are trademarks of Siemens Energy & Automation, Inc.

Texas Instruments is a registered trademark of Texas Instruments Incorporated.

IBM, PC/AT, and PC/XT are registered trademarks of International Business Machines Incorporated.

Modbus is a registered trademark of Modicon Inc.

MS-DOS and MS-BASIC are registered trademarks of Microsoft Corporation.

80C386SX is a trademark of Intel Corporation.

Local Line is a trademark of Siemens Industrial Automation.

Foxboro Foxnet is a registered trademark of the Foxboro Company. SPECTRUM is a trademark of the Foxboro Company.

Honeywell TDC300 and TDC2000 DHP are registered trademarks of Honeywell, Inc.

Fisher PROVOX and PCIU are registered trademarks of Fisher Controls International, Inc.

DEC, VAX, MicroVAX, VMS, and RSX–11 are trademarks of Digital Equipment Corporation.

LOTUS is a registered trademark of Lotus Development Corporation.

HP Deskjet, HP Paintjet, and HP Laserjet are trademarks of Hewlett-Packard Company.