SIEMENS

SIMATIC TI560T/TI565T

Technical Product Description

Order Number: PPX:560/565–8107–3
Manual Assembly Number: 2597773–0033
Third Edition

MANUAL PUBLICATION HISTORY

SIMATIC TI560T/TI565T Technical Product Description
Order Manual Number: PPX:560/565–8107–3

Refer to this history in all correspondence and/or discussion about this manual.

Event	Date	Description
Original Issue	04/88	Original Issue (2601337–0001)
Second Edition	02/90	Second Edition (2601337–0002)
Third Edition	11/93	Third Edition (2601337–0003)

LIST OF EFFECTIVE PAGES

Pages	Description	Pages	Description
Cover/Copyright	Third		
History/Effective Pages	Third		
iii — ix	Third		
1-1 — 1-12	Third		
2-1 — 2-20	Third		
3-1 — 3-19	Third		
4-1 — 4-17	Third		
5-1 — 5-14	Third		
A-1 — A-7	Third		
B-1 — B-19	Third		
Registration	Third		

Contents

Chapter 4 Hot Backup and System Diagnostics

Chapter 5 System Configuration

List of Figures

List of Tables

Preface

This Technical Product Description provides detailed information on the SIMATIC® TI565T™ programmable controller family CPU models SIMATIC® TI560T™ and TI565T. Topics include:

- TI560T overview
- TI565T overview
- Configuration system
- Component descriptions
- System operation and performance
- Physical dimensions

This document is intended to be used as a reference for technical product information. It is not intended to provide detailed product information required during installation or operation of the system. Use the *SIMATIC TI560T/TI565T System Manual* (PPX:560/565–8105–x) to set up and operate your TI560T/TI565T.

Agency Approvals

The TI560T/TI565T I/O modules meet the standards of the following regulatory agencies.

- Underwriters Laboratories Inc.®: UL Listed (Industrial Control Equipment)
- Canadian Standards Association: CSA Certified (Process Control Equipment)
- Factory Mutual Approved: approved for Class I, Div 2 Hazardous Locations

TI560T/TI565T products have been developed with consideration of the draft standard of the International Electrotechnical Commission Committee proposed standard (IEC-65A/WG6) for programmable controllers. Contact Siemens Industrial Automation, Inc., for a listing of the standards to which Series 505™ complies.

Telephoning for Assistance

If you need information that is not included in this manual, or if you have problems using the SIMATIC TI560T/TI565T modules, contact your Siemens Industrial Automation, Inc. distributor or sales office. If you need assistance in contacting your U.S. sales office, call 1–800–964–4114.

Chapter 1
Overview

1.1 System Overview

The TI560T and TI565T, along with the other members of the Siemens Industrial Automation, Inc. control family, provide a cost effective and practical approach to designing a control system for your application, regardless of size or type.

You can use the SIMATIC® TI500™/TI505™ communication modules to link your TI560Ts, TI565Ts, or any of the TI500 or TI505 controllers. The CVU™ operator interface product line can provide high quality monitoring and control of single operating units. The TISTAR™ system provides the high-level management to coordinate and control your automated site. Finally, to program and monitor your application, Siemens offers a complete line of programming software and operator interfaces.

The following illustration highlights the compatibility and versatility of the Siemens product line for customizing to fit your application, using controllers such as:

- SIMATIC® TI530C™
- SIMATIC® TI525™
- SIMATIC® TI535™
- SIMATIC® TI520C™
- SIMATIC® TI560T™
- SIMATIC® TI565T™
- SIMATIC® 5TI™

1.2 System Diagram

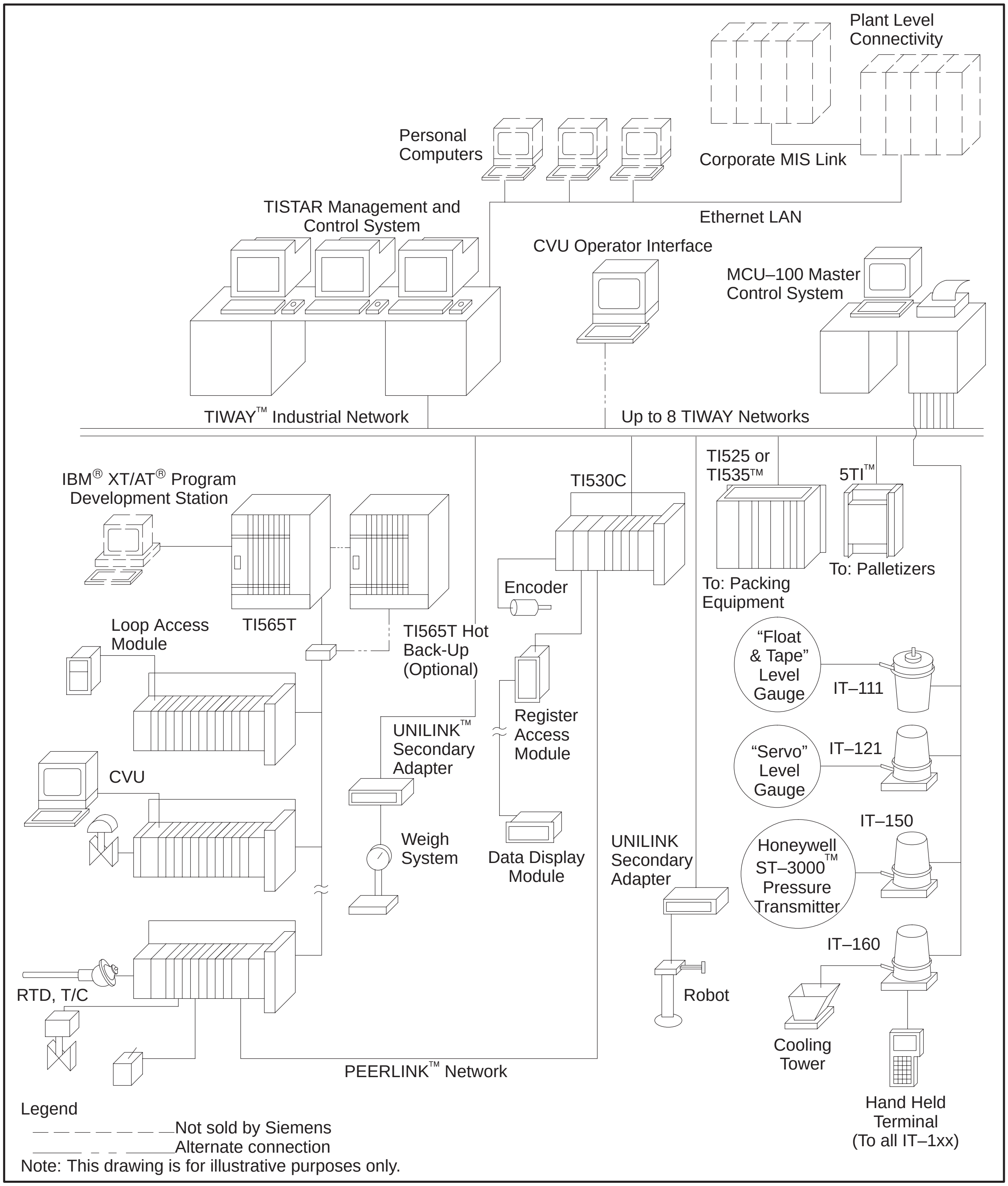


Figure 1-1 TI560T/TI565T System Layout

1.3 TI560T and TI565T

The TI560T/TI565T product family offers features to handle a wide variety of application needs. For both discrete and batch processes, the TI560T/TI565T performance meets the requirements.

The TI560T/TI565T Programmable Controller has MC68000 microprocessor based, card-and-rack architecture. The system can be expanded with the addition of modular plug-in cards to add more processing power, memory, and I/O capabilities to the system. A full-up TI565T system with Hot Backup has seven microprocessors operating in parallel.

A basic TI560T system is composed of the chassis, a chassis power supply, a main CPU card, and one Remote Channel Controller (RCC) card. Memory expansion cards and additional RCC cards can be added to the system as needed. An optional Hot Backup system can be added to provide redundancy of CPU, memory, power supply, and RCCs.

The TI560T can be upgraded to the TI565T with the addition of the Special Function (SF) CPU card. The two processors execute multiple control programs in parallel. The Main CPU executes the relay ladder logic program while the Special Function CPU executes PID loops, analog alarms, and Special Function programs and subroutines. Up to 64 PID loops with advanced process control capabilities are available, along with 128 analog alarms. The loops and analog alarms are programmed with fill-in-the-blank menus. Function programming is a high-level programming language that supplies advanced math and reporting capabilities. Hot Backup is also available with the TI565T; and all TI500/TI505 programs, I/O modules, programmers, and peripherals are compatible.

A TI560T/TI565T controller fits in a 19″ chassis. The components needed for a TI560T/TI565T consist of a power supply and from three to eight cards, depending on your application.

The TI560T and TI565T modular hardware lends itself to custom system design. You may select from the following SIMATIC® TI560™/TI565™ cards to build your controller.

TI560/TI565 Cards

Main CPU

Special Function CPU

Memory

Hot Backup

Remote Channel Controller

To complete the system are the TI500/505 I/O modules which communicate with the controller through the Remote Channel and Remote Base controllers. This allows an I/O connect distance of up to 15,000 feet from the controller.

I/O

NSC800 Microprocessor-based RBC

Fast Update Times

Up to 8,192 I/O Points

I/O Configurable from Programming Device

15,000-foot Range with RF Link

3,000 feet with Twisted Pair Media

Coaxial Cable Communication Media

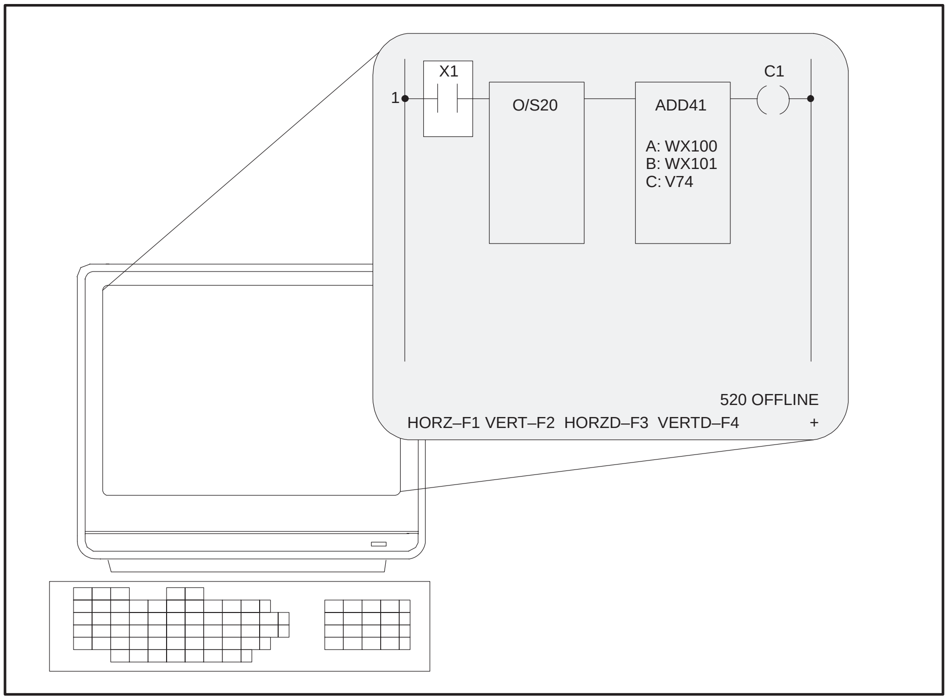
Supports all TI500/TI505 I/O Modules

RS-232 Ports available on RBCs

1.4 Programming

TISOFT

TISOFT™ is the factory-floor programming software for Siemens Industrial Automation, Inc. controllers. TISOFT provides the means for directly entering, debugging, and documenting your TI560T/TI565T ladder logic or special function program. TISOFT is menu-driven, provides block copy and edit functions, and allows you to add comments to instructions.



Hardware

You have the option of using TISOFT on the Siemens VPU or on an IBM or IBM-compatible Personal Computer for designing and entering your program.

1.5 Control System Design

APT

Siemens Application Productivity Tool—APT™ provides a structured, graphical design environment for complex control problems. It supports a top-down, object-oriented approach to the development, integration and documentation of advanced control strategies. A control problem is described and solved in two major working environments:

- Sequential function charts (Figure 1-2) define in flowchart form the control sequence and transitions from one operating state to another.
- Continuous function charts (Figure 1-3) define the regulatory control strategies which are to be enabled in each of the various states.

APT allows for independent development of control modules in a team design environment and provides intermediate design review documentation all along the way.

Figure 1-2 Sequential Function Charts

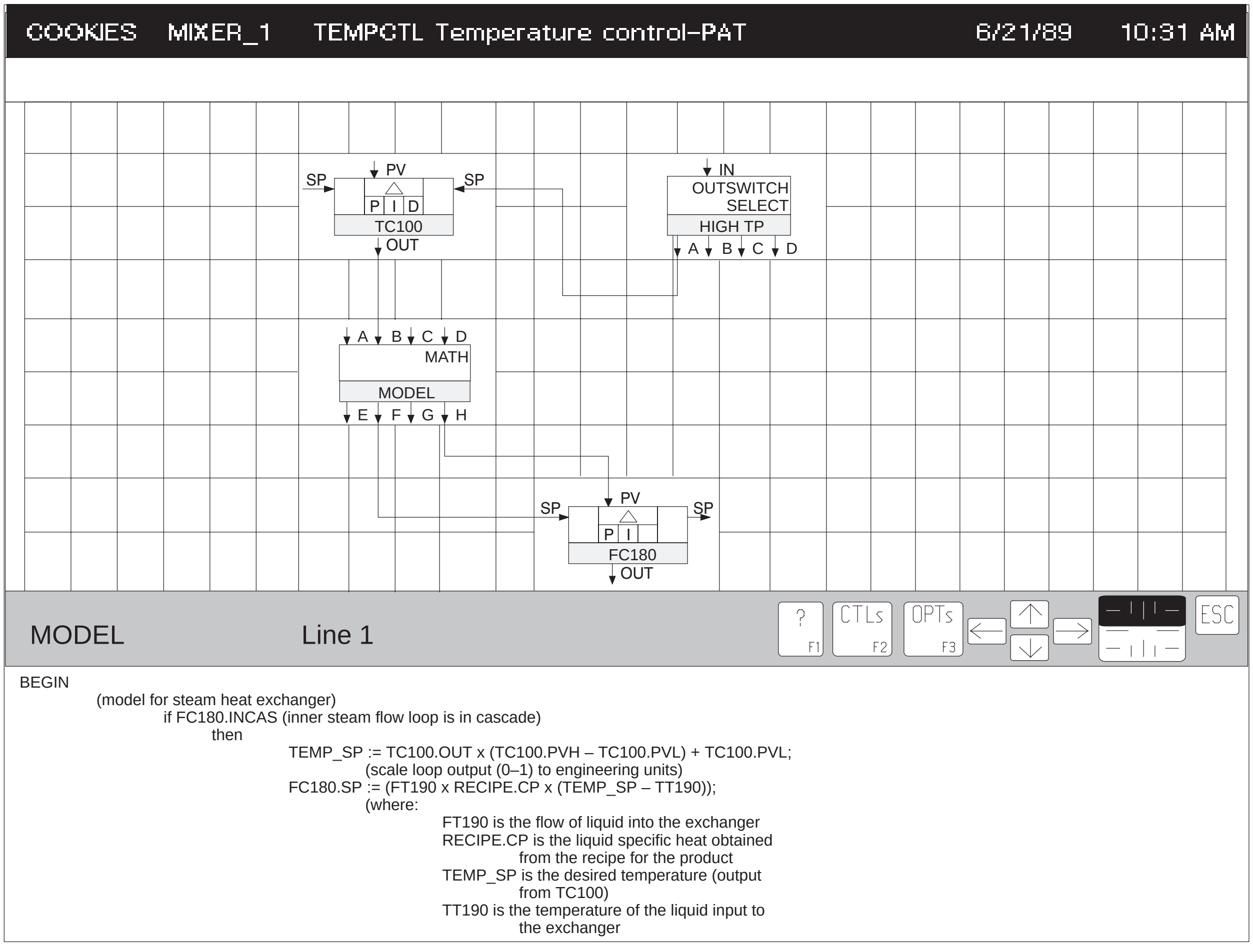


Figure 1-3 Continuous Function Chart

Hardware

APT operates on the following hardware platforms:

- TISTAR operator station
- IBM Personal Computer AT compatible with an EGA, VGA or CVU graphics card
- IBM PS/2™ Model 50, 60, 70 or 80 VGA graphics

1.6 Networking

To link your control system, Siemens offers three communication interface options. (See Figure 1-4.)

Network Communication

Network Interface Module (NIM)

PEERLINK Module

UNILINK Module

Each of the communication modules lends itself to a particular networking need, and all three can be combined to fit a variety of needs within a single application.

Network Interface Module

The NIM connects TI500/TI505 controllers with the TIWAY network. Each NIM contains a microprocessor and two media interface ports for redundant media protection.

Peerlink Module

The PEERLINK module provides high speed transmission of relatively small amounts of data from TI500/TI505 controllers on the network. Peerlink can be configured for redundancy also.

UniLink Module

The UNILINK module is available for applications requiring data transfer and control between third-party equipment and Siemens controllers. The third-party equipment may be either other secondaries or host computers.

Networking (continued)

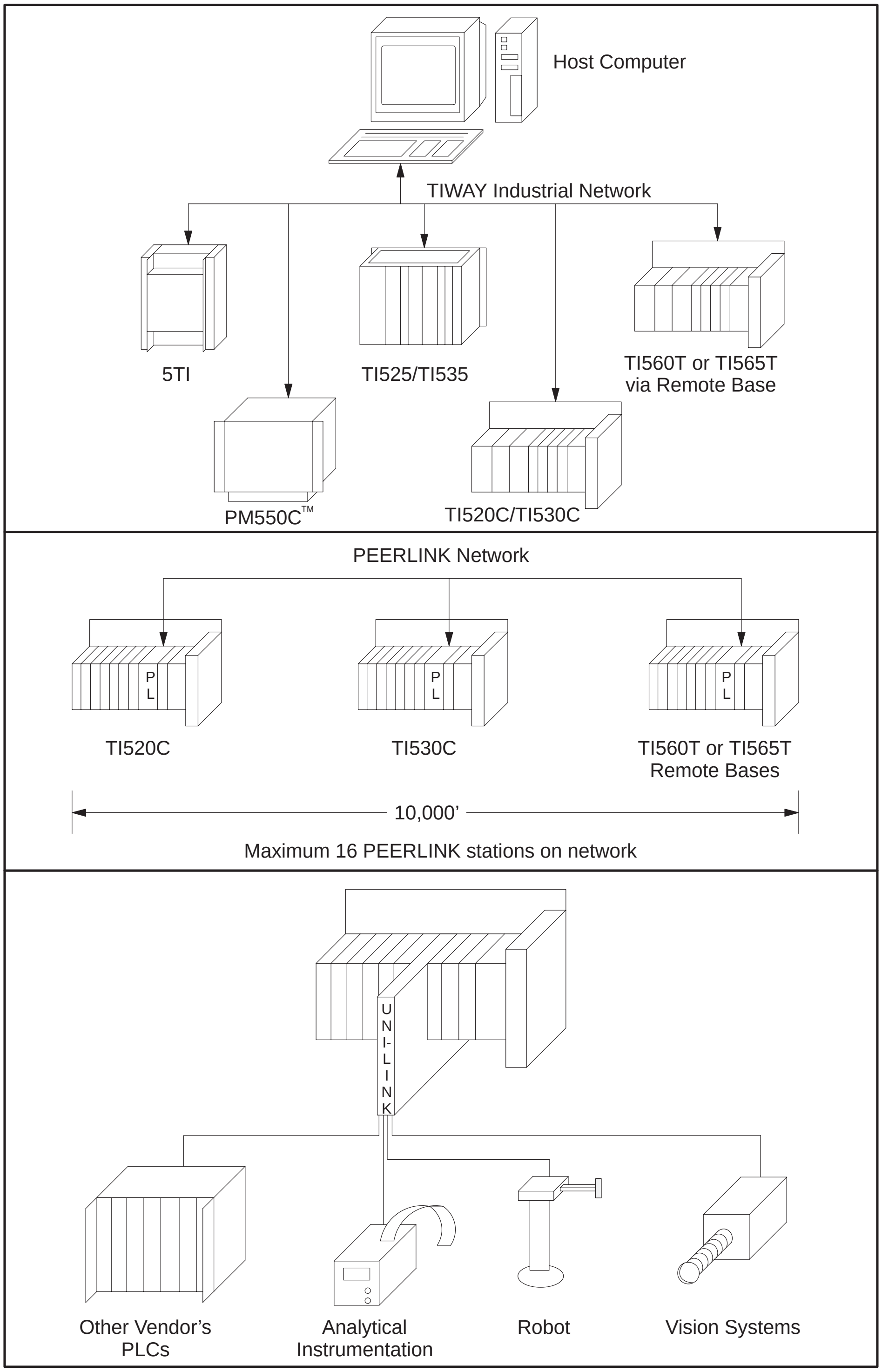


Figure 1-4 Networking Configurations

1.7 Operator Interfaces

Siemens provides a choice of operator interfaces for monitoring and controlling TI560T/TI565T programs. Depending upon your application, select from the CVU or the Micropanel family of operator interfaces. If your operation is large enough to require different types of control at various points, then you can choose the particular operator interface you need for the different areas.

CVU

The CVU (Figure 1-5) is a control and monitoring device that allows the operator to supervise automated processes driven by Siemens controllers. The CVU offers the following capabilities.

- Mimicking the Process
- Displaying Data
- Tuning Continuous Control Loops
- Reporting Alarm Conditions
- Trending Process Data
- Generating Reports
- Password Protection
- Downloading Recipes

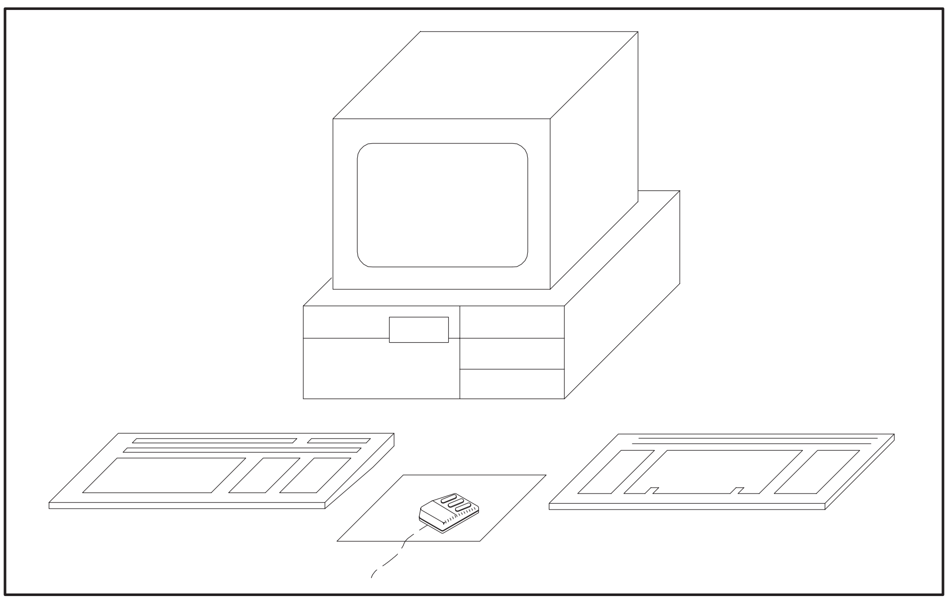


Figure 1-5 CVU

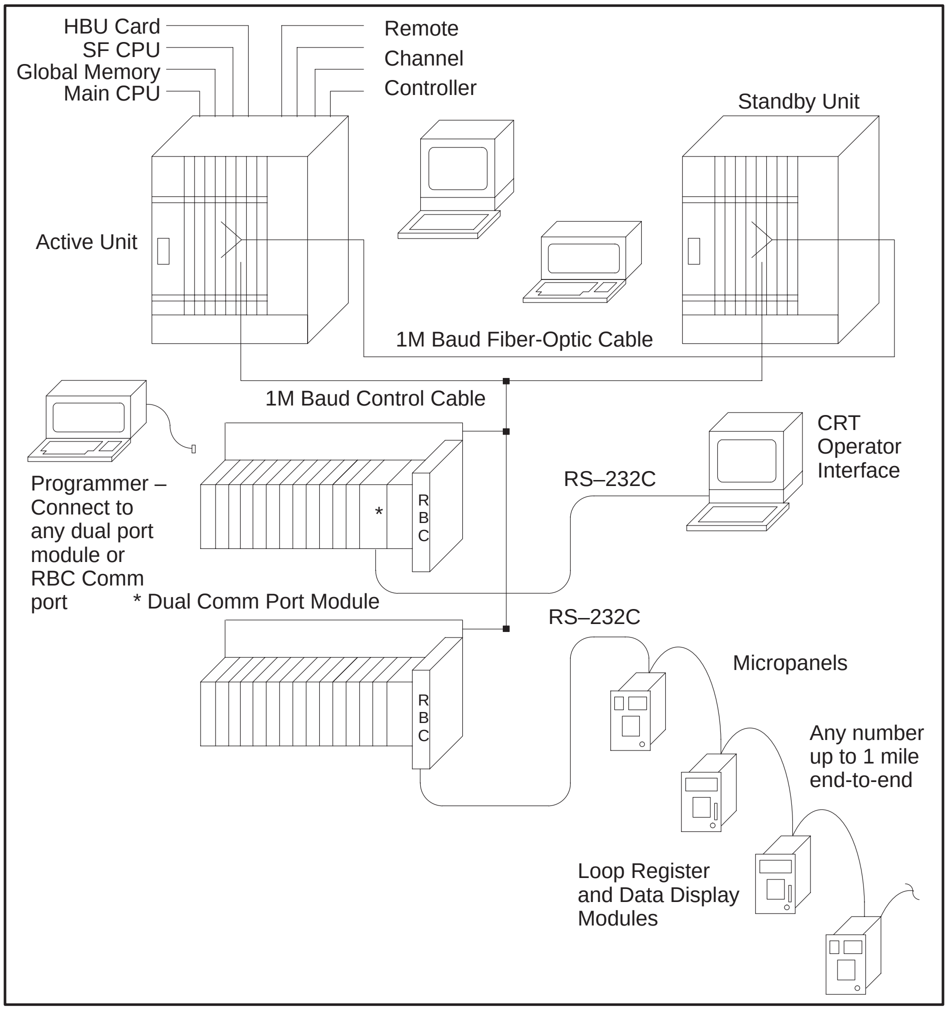


Figure 1-6 Micropanel Operator Interfaces with TI560T/TI565T

The Micropanel family of operator interfaces includes the Random Access Module (RAM), the Loop Access Module (LAM), and the Data Display Module (DDM). (See Figure 1-6.)

RAM

The RAM is useful for displaying and changing most unprotected memory locations in a TI500/TI505 controller.

LAM

Using the LAM enables an operator to display PID loop parameters or tune PID constants.

DDM

The DDM is a read-only device for displaying process information contained in the controller.

Familiarize yourself with the features of the TI560T/TI565T as they are detailed in the following pages, and decide which configuration best meets your needs.

Chapter 2
TI560T

2.1 TI560T

Advantages

The TI560T satisfies your application requirements for a high number of I/O points (8192) with quick, reliable performance. Additionally, the TI560T is compatible with Siemens' other TI500/TI505 products, which enables you to select from a full product line to customize your application.

Features

The TI560T offers high-level instruction boxes, such as multiplication and division, timers and counters, and word moves. Using the instruction boxes for these functions both eliminates the unnecessary use of memory to program individual steps and saves valuable programming time.

You can select the memory size you need, from the 176K words resident on the CPU to a total of 512K words, with the addition of memory cards.

The remote I/O provides a connect distance of up to 15,000 feet from the controller.

The Hot Backup option is available for critical operations requiring minimum unscheduled process down time.

The TI560T or TI565T chassis is a 19-inch, rack-mountable housing which can accommodate up to eight cards. The chassis contains a backplane to which the system cards are physically connected.

The system power supply is mounted in the far left side of the chassis and does not occupy any of the 8 slots available for cards. The first slot next to the power supply is reserved for the Main CPU card. The remaining slots can be used for memory expansion, I/O expansion, SF CPU option, and the Hot Backup option.

2.2 Chassis

The TI560T or TI565T chassis (Figure 2-1) is a 19-inch, rack-mountable housing which can accommodate up to eight cards. The chassis contains a backplane to which the system cards are physically connected.

The system power supply is mounted in the far left side of the chassis and does not occupy any of the 8 slots available for cards. The first slot next to the power supply is reserved for the Main CPU card. The remaining slots can be used for memory expansion, I/O expansion, SF CPU option, and the Hot Backup option.

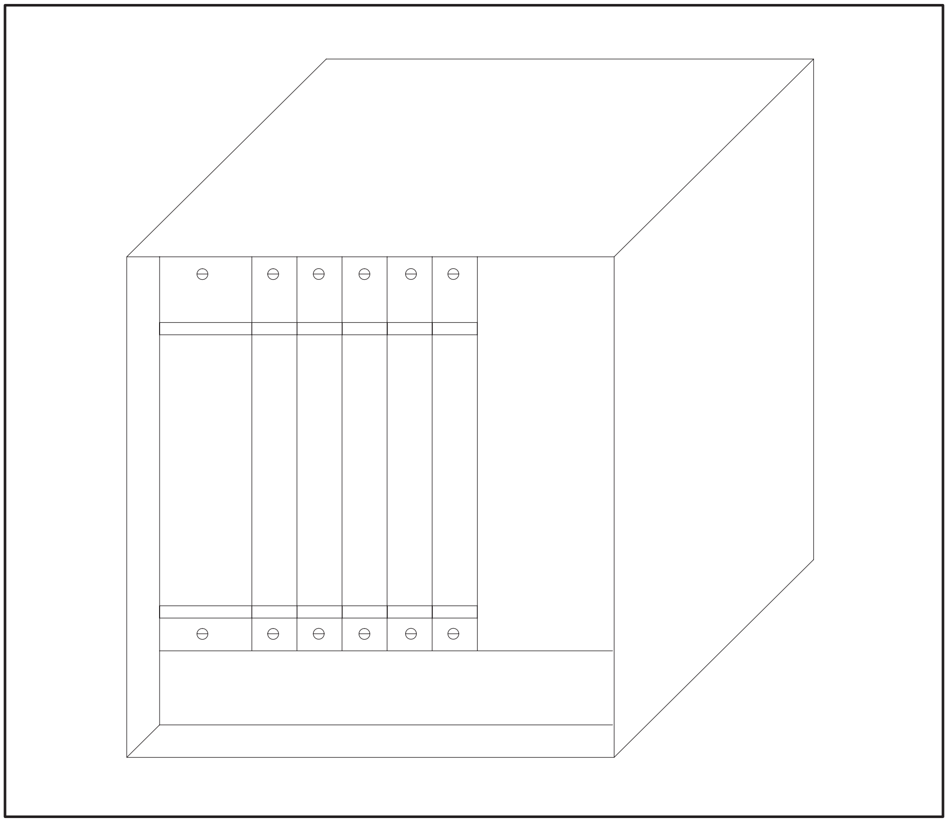


Figure 2-1 TI560T/TI565T Chassis

2.3 Chassis Power Supply

The TI560T or TI565T chassis power supply is a plug-in module which resides in the left side of the chassis. (See Figure 2-2.) You may select either a 110/220 VAC or a 24 VDC power supply. This module supplies power to all the cards in the chassis. Use the PPX:560T–2125 chassis with the 24 volt power supply and the PPX:560T–2124 chassis with the 110/220 volt power supply.

The AC power supply provides 112.5 watts of total power to the system and up to 1 amp of battery backup power. Individual cards do not have battery backup. The DC power supply affords 147.5 watts of total power to the system.

A memory-protect lock, power LED, and battery status LED are located on the front panel of the power supply. The battery on/off switch, battery, AC power connection, and fuse are also accessible through the front panel. Selection of 110 VAC or 220 VAC is done via an internal switch.

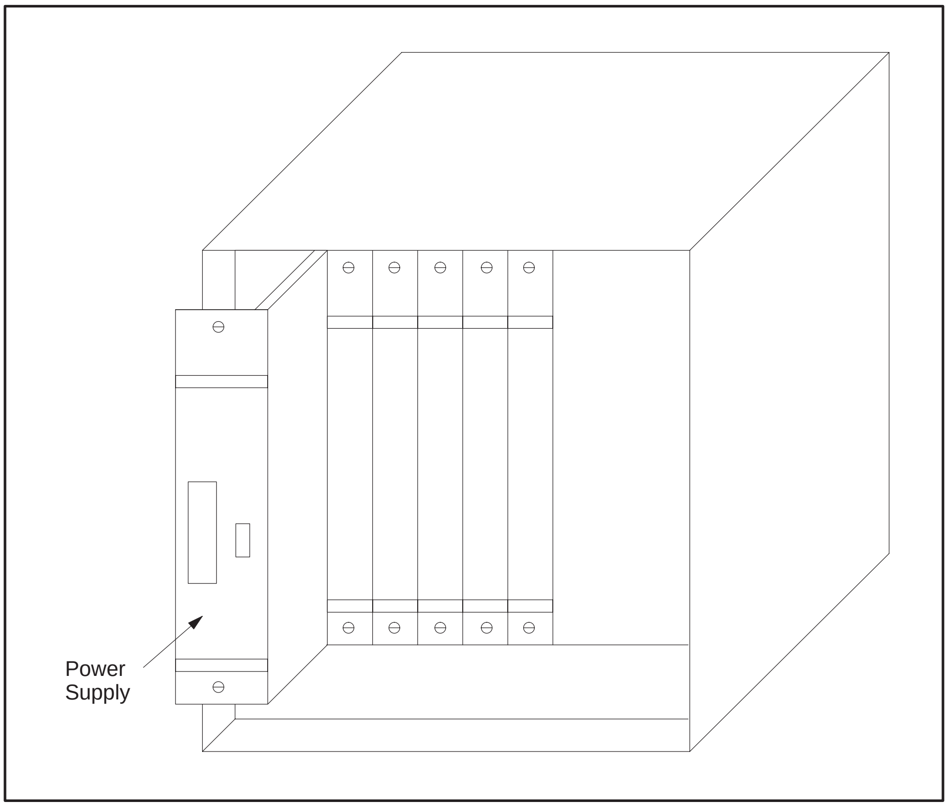


Figure 2-2 TI560T/TI565T Power Supply Location

2.4 Main CPU Card

The Main CPU card is MC68000 microprocessor-based and contains 176K words of system memory along with 49,152 local control relay bits. This CPU acts as the central coordinator for the system. It executes relay ladder logic (RLL), schedules I/O updates, processes intelligent I/O module requests, SF CPU card communications, Hot Backup card communications, and the local RS–232–C and RS–422 port communications. The CPU card must be placed in the far left slot adjacent to the power supply (Figure 2-3). The edge connectors are reverse-keyed to prevent the card from being inserted into another slot.

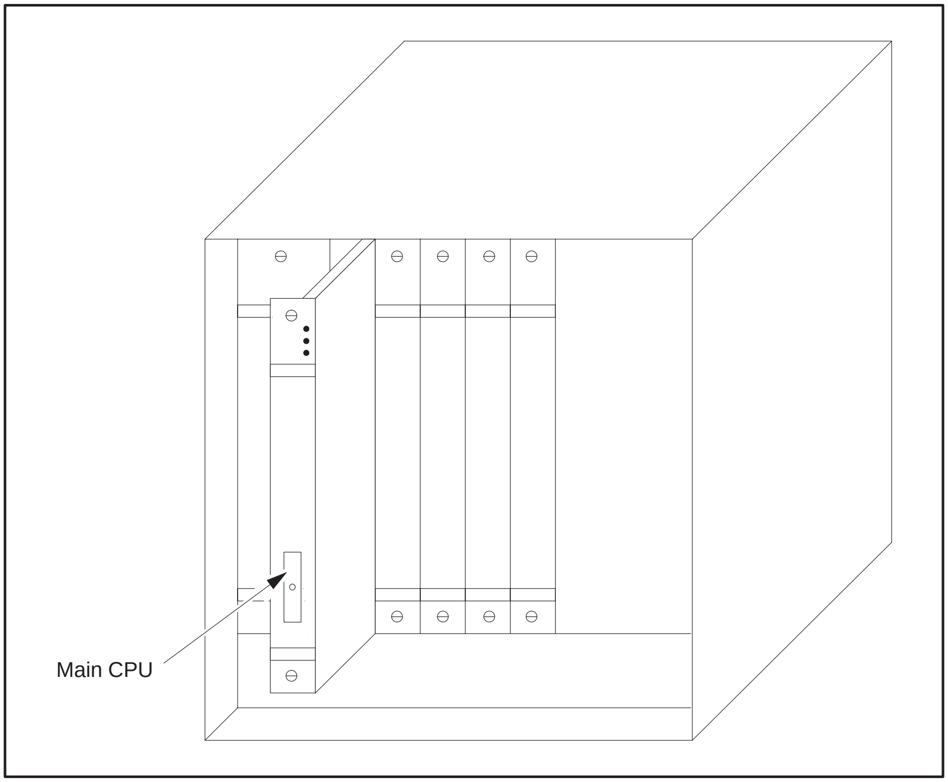


Figure 2-3 TI560T Main CPU Card Placement

2.5 CPU Functions

Figure 2-4 shows the relationship of the sequence of operations performed during a scan.

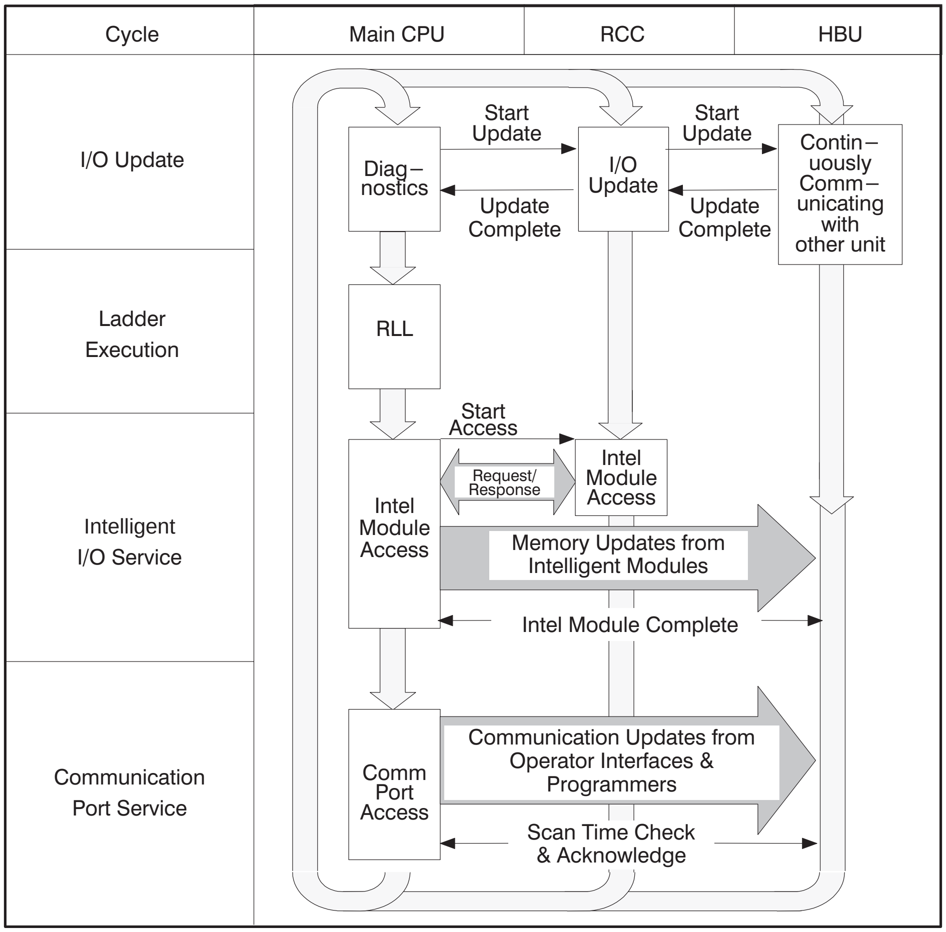


Figure 2-4 CPU Scan

TI560T CPU

The TI560T CPU requires approximately 11 ms for overhead tasks. This time is distributed throughout each scan.

I/O Cycle

During the I/O cycle update, the Main CPU card runs board-level diagnostics. The RCC card(s) update all output points (discrete, word, and analog) with new values from the last program scan. All input modules are read and transferred to the image register for the next RLL execution. The length of the I/O update cycle corresponds to the RCC card that requires the longest update time. This is primarily dependent upon the number of bases and types of modules (analog, discrete or intelligent) on each channel. Each RCC has two channels which update in parallel. All I/O points are fully updated each scan.

Ladder Logic Cycle

Upon completion of the I/O update, the Main CPU card starts the execution of the relay ladder logic program. While the program is being executed, the RCCs run background tasks (polling for unconfigured bases and servicing operator interfaces connected at the I/O bases if that RBC has requested service).

The Main CPU executes typical relay ladder logic programs in 1.5 milliseconds per K words of program instructions. The entire ladder logic program is fully executed each scan. Execution times for the full range of instructions are given with the Instruction Set listed in Appendix A.

Intelligent Module

Upon completion of the ladder logic scan, the intelligent module communications begin. The Main CPU executes task codes gathered from the intelligent modules by the RCC. This information is made available to the RCCs for transfer back to the intelligent modules in the I/O bases. The RCCs start diagnostics when the transfer of SF module information is complete.

Each intelligent 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 task code requests, block transfers, or store-and-forward operations per scan. Once these are completed, the SF cycle is terminated by the RCC card.

Communication Port Service

The communication period consists of three 2-millisecond time slots to service the two communications ports on the Main CPU and all communication ports located on all RBCs. Each time slot is allocated approximately 2 milliseconds, with a maximum of 6.0 milliseconds available. Four communication tasks are serviced in the three time slots such that at any given time two of the four communication tasks being performed are from the local ports. The other two tasks being performed are from the remote ports on a time-share basis. If only one port has activity, then all three time slots will be allocated to that port.

Hot Backup Unit Communications

During all of the above periods, the HBU transmits messages between the TI560T or TI565T systems (with standby unit on-line). The HBU adds approximately 9.0 milliseconds for TI560T operation and approximately 1 to 4 ms additional for TI565T updates.

2.6 Memory Cards

The TI560T system comes with 176K words of memory resident on the Main CPU card. Additional memory cards can be installed. (See Figure 2-5.)

Memory cards may be placed anywhere in the TI560T or TI565T chassis, excluding the slot adjacent to the power supply where the Main CPU must be positioned.

The TI560T or TI565T system is capable of storing 512K words of user memory. Additional memory can be purchased in card sizes of 64K or 256K words.

Features

Main CPU

User-Configurable

176K words memory resident on main CPU card

Three additional memory cards can be installed

System memory 512K words maximum

Ladder logic up to 128K words

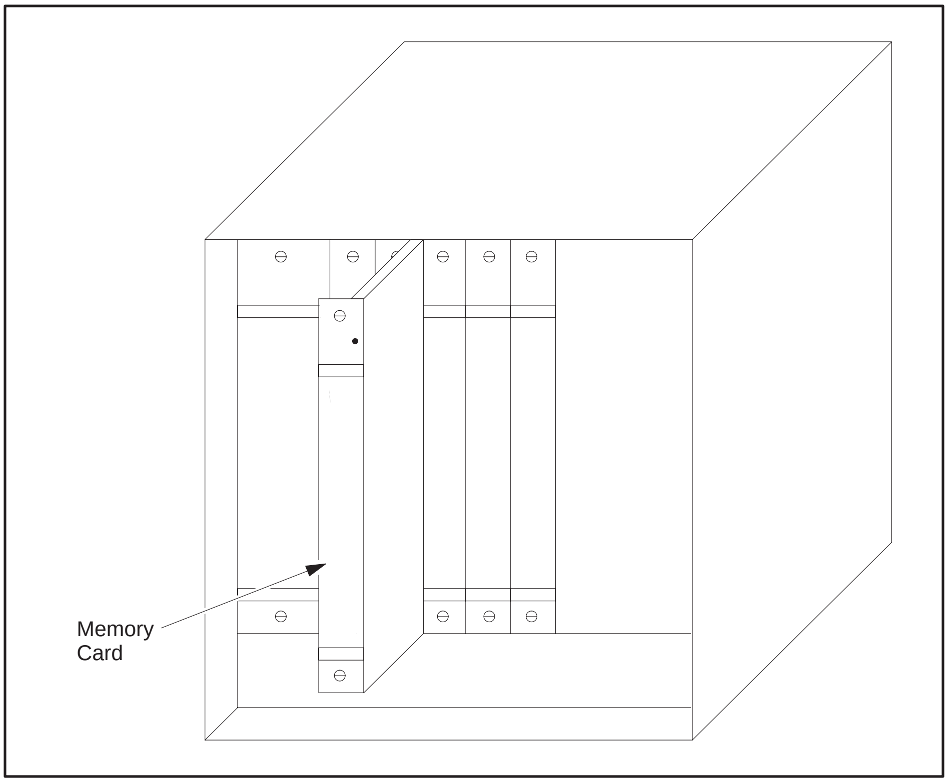


Figure 2-5 TI560T/TI565T with Additional Memory Card

Memory Types	
L-Memory	V-Memory
K-Memory	S-Memory

L-memory is used for ladder logic, V-memory is used for variable memory (changeable from ladder logic), K-memory is used for constant memory (changeable only from task codes), and S-memory is used with the SF CPU for PID loops, analog alarms, special function programs, and special function subroutines.

Memory Allocation

System memory is user-configurable. L-, V-, K-, and S-memory allocations can be set or altered with a TISOFT programming device. This allows an optimum memory configuration for any application. And, the user can expand memory type capacities without affecting program integrity. Additional memory space is allocated without damage to existing programs.

Ladder logic program code is stored in both source code and compiled code. For this reason, the total amount of L-memory required will be 3X the amount of ladder logic program code. Also note in the tables that memory allocation is displayed in K bytes rather than words (2K bytes = 1K word).

Table 2-1 lists the minimum and maximum memory configuration for the TI560T/TI565T controllers.

Table 2-1 CPU Card Memory Configuration

Memory Type	Allocation Size	Required per Block	Minimum Size	Maximum Size	Required for Maximum
Ladder (L)	4K bytes	12K bytes	16K bytes	256K bytes	768K bytes
Variable (V)	2K bytes	2K bytes	4K bytes	958K bytes	958K bytes
Constant (K)	2K bytes	2K bytes	0K bytes	954K bytes	954K bytes
Special (S)	2K bytes	2K bytes	0K bytes	954K bytes	954K bytes
TMR/CTR/ DCATS/MCAT	1024*	5K bytes	1024	20480	100K bytes
DRUM/EDRUMS/ MDRMW/MDRMD	128*	6K bytes	128	2304	108K bytes
Shift Registers	2048*	2K bytes	2048	16384	16K bytes
MOVE TO/FROM Tables	1024*	2K bytes	1024	14336	28K bytes
One Shots	2048*	2K bytes	2048	32768	32K bytes

* Number per block

2.7 Remote Channel Controller Cards

The TI560T or TI565T I/O system serves as the interface between field sensors and the logical control performed by the system processors. Remote Channel Controller (RCC) Cards, which reside in the TI560T or TI565T chassis (Figure 2-6), serve as the communication masters for the I/O system. These MC68000 microprocessor-based cards perform I/O reads and writes to gather information from remote I/O bases. Each remote I/O base is controlled by a Remote Base Controller (RBC) which communicates with its respective Remote Channel Controller through CATV cables, fittings, and taps. RS-485 media allows communication up to 3,300 ft. depending on the number of taps.

Each Remote I/O Base contains a remote base power supply which supplies logic power to the RBC and TI500/TI505 I/O modules installed in the base. The TI500/TI505 controller family allows you to install any I/O module in any slot of any base: the I/O map is fully user-configurable. Any TI500/TI505 I/O base may be used with the TI560T or TI565T system. Some TI500 I/O bases require adapter bases. See the *SIMATIC TI560T / TI565T System Manual* (PPX:560/565–8105–x) for details.

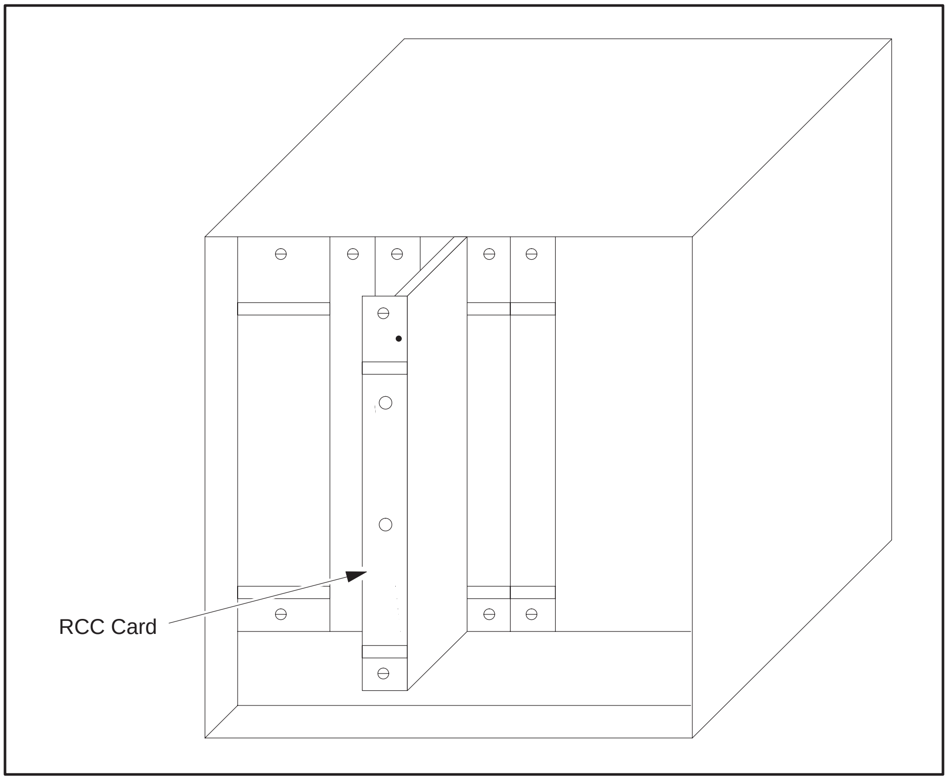


Figure 2-6 TI560T / 565T RCC Card

Operation

The Remote Channel Controller (RCC) controls all communications between the TI560T or TI565T system and the Remote Base Controllers (RBC). The RCC sends and receives I/O updates to and from the RBCs located on the remote I/O base.

The TI560T or TI565T system can accommodate up to four RCC cards, each card addressing up to 2048 I/O points for a system total of 8,192 I/O points. Each RCC can address 2 I/O channels; each I/O channel can consist of up to 1024 I/O points of any mix of analog or discrete. In addition, each RCC can address 2048 Control Relays (Cs) which are used as internal programming control elements. A system with 4 RCCs has 8,192 Control Relays available. The last 256 Cs on each RCC channel are retentive i.e., C769 – C1024 on channel 1; C1793 – C2048 on channel 2; etc.

These 8192 control relays on the RCCs are in addition to the 49,152 local control relays provided on the TI560 CPU card. A system with all 4 RCCs installed then would have a total of 57,344 control relays.

Using CATV media, the RCC can communicate with up to 16 RBCs on each channel and the RBCs can be located up to 15,000 ft. or further from the TI560T chassis. There should be less than 55 dB loss from the RCC card to any RBC. RS-485 media allows communication up to 3,300 ft. depending on the number of taps.

Each RCC is controlled by a dedicated MC68000 microprocessor. All I/O channels (8 max.) are updated in parallel to minimize system I/O update times. The I/O communications channel operates at a data rate of 1 MBaud on a 35.75 MHz carrier frequency. A Miller encoding technique is utilized with FSK data transfer with an extended message preamble. Error checking includes CRCC–CCITT and field-length checks.

2.8 Ladder Logic Instruction Set

Instruction Types

The Main CPU card executes an advanced ladder logic instruction set, which includes discretes, ladder logic subroutines, and high-level instructions. This instruction set is a superset of the TI520C and TI530C instruction sets. As a result, any program that operates on these models can be upgraded to a TI560T or TI565T.

Discrete Instructions

Inputs — The input instruction (X) represents a discrete (on or off) input device. Up to 8192 inputs are available.

Outputs — The output instruction (Y) controls the status (on or off) of the output module point that corresponds to the instruction reference number. Up to 8192 outputs are available.

Control Relays — Each channel of the RCC card provides 1024 control relays. The first 768 relays on each channel are nonretentive, the next 256 are retentive. A total of 8192 control relays are available on a full-up (4 RCCs) system.

Conditional/Unconditional Operations

Jump — The Jump instruction freezes all outputs in its field of control. The end jump instruction may be either conditional or unconditional.

Master Control Relay — The MCR instruction turns off all 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.

Scan Sync Inhibit — The Scan Synchronization Inhibit instruction is used to control bringing a standby unit on-line to run in synchronization with the active unit in a hot backup configuration.

Bit Instructions

Bit Clear — The Bit Clear instruction clears a selected bit of a word to 0.

Bit Pick — The Bit Pick instruction tests a selected bit of a specified word to determine its status.

Bit Set — The Bit Set instruction sets a selected bit of a word to 1.

Bit Shift Register — The Bit Shift Register instruction creates a bit shift register containing up to 1023 bits.

Counter/Timer Instructions

Counter The Counter instruction counts recurring events up to 32,767.

Up/Down Counter The Up-down Counter instruction counts the number of events (up or down) that have occurred between zero and 32,767.

Timer The Timer instruction decrements in .1-second or 1-millisecond steps for timing of events.

Discrete Control Alarm Timer The Discrete Control Alarm Timer is designed for use with a 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 The Motor Control Alarm Timer is 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 The Drum instruction simulates operation of an electromechanical drum, and provides 15 output coils with 16 steps. The steps are indexed on multiples of the time base set up for the drum.

Event Drum The Event Drum instruction simulates operation of an electromechanical drum and provides 15 output coils with 16 steps. The steps can be indexed by a timer only, and event contact only, or a combination of an event contact and a timer.

Maskable Event Drum-Discrete The Maskable Event Drum-Discrete functions in the same manner as the event drum with the addition of the capability of specifying a mask for each step to allow selection of the outputs to be controlled by the step.

Maskable Event Drum-Word The 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

Scan Matrix Compare

The Scan Matrix Compare instruction 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

The Indexed Matrix Compare instruction 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

Add

The Add instruction adds a signed integer (positive or negative) in memory location A to a signed integer in memory location B 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

The Subtract instruction subtracts a signed integer (positive or negative) in memory location B 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

The Divide instruction 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

The Multiply instruction 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

The Compare instruction 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

The Square Root instruction takes the square root of a positive integer and stores the result in a specified memory location. The integer retains its original value.

Move Instructions

Load Data Constant
The Load Data Constant instruction loads the specified into the A memory location.

Move Discrete IR to Word
The 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
The Move Word to Image Register instruction 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)
The Move Word instruction moves up to 256 words from a designated memory location to another designated location.

Move Word from Table
The Move Word from Table instruction 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
The Move Word to table instruction 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
The Indirect Move Word instruction is actually a table-to-table block move operation. In a single scan it move a block of up to 255 words from within a source table to within a destination table, then increments both tble pointers by the number of words moved. In batch appliations this instruction greatly simplifies the management of large tables of recipe parameters.

Move Image Register to/from Table
These instructions move blocks of image register bits (up to 255 words or 4080 bits per scan) into or out of tables in V-memory 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 Instructions

Convert Binary to BCD — The Convert Binary to BCD instruction 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 — The Convert Decimal to Binary instruction converts BCD inputs to equivalent binary integer values. BCD inputs up to 9999 are converted to their binary integer equivalents.

Word AND — The Word AND instruction 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 — The Word OR instruction 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 — The Word Exclusive OR exclusively ORs a word in memory location A with a word in a second location B. The result of the WXOR instruction is placed in memory location C.

Word Rotate — The Word Rotate instruction rotates to the right 4-bit segments of the word location specified.

Word Shift Register — The Word Shift Register instruction shifts from 1 to 1023 words from memory location A to V-memory, beginning at location B.

One Shot — The One-Shot instruction provides an output for one memory scan.

Ladder Logic Subroutines

Go to Subroutine — The Go to Subroutine instruction enables you to write ladder logic programs preceded by a subroutine number and to call them as needed.

Subroutine — The Subroutine instruction is placed before a set of ladder logic instructions that are to be executed only when called with the Go to Subroutine instruction.

Return — The Return instruction brings execution of the ladder logic program back to the network. The Return may be either conditional or unconditional.

Table Instructions

Table to Table AND The Table to Table AND instruction ANDs the corresponding bits in two tables and places the results in a specified third table.

Table to Table OR The Table to Table OR instruction ORs the corresponding bits in two tables and places the results in a specified third table.

Table to Table EXCLUSIVE OR The Table Exclusive OR instruction exclusively ORs the corresponding bits in two tables and places the results in a designated third table.

Table Complement The Table Complement instruction 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 The Word to Table instruction places a duplicate of a word at the address specified in the destination table.

Table to Word The Table to Word instruction duplicates a specified word in a table to another word location.

Word to Table AND The Word to Table AND instruction 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 The Word to Table OR instruction 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 EXCLUSIVE OR The Word to Table EXCLUSIVE OR instruction exclusively ORs the corresponding bits in a specified word and a word from a source table. The resultant word is placed in a destination table.

Table Search for Equal The Table Search for Equal instruction 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 The Table Search for not Equal instruction 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.

Clock Instructions

Date Compare The Date Compare instruction compares the current date of the real-time clock with the value contained in the designated V locations.

Time Compare The Time Compare instruction compares the time of the real-time clock with the values contained in the designated V-memory locations.

Time Set The Time Set instruction sets the time portion of the real-time clock to the values contained in designated V-memory locations.

Date Set The Date Set instruction sets the date portion of the real-time clock to the values contained in designated V-memory locations.

2.9 TI560T Controller Component List

The following listing provides the part numbers for components required to assemble a TI560T controller to fit your application.

PPX:560T–1101	Basic TI560T System, 120/240 VAC Includes: 1–Power Supply, 120/240 VAC (PPX:560T–2122) 1–Main CPU (PPX:560T–2820) w/176K Words of Memory 1–Chassis, 120/240 VAC (PPX:560T–2124)
PPX:560T–1102	Basic TI560T System, 24 VDC Includes: 1–Power Supply, 24 VDC (500–2123) 1–Main CPU (PPX:560T–2820) w/176K Words of Memory 1–RCC (PPX:560T–2126) (2048 I/O Capacity) 1–Chassis,-24 VDC (560T–2125)
PPX:560–2120	Main CPU w/48K Words of Memory
PPX:560–2122	120/240 VAC Power Supply
PPX:560–2123	24 VDC Power Supply
PPX:560–2124	Chassis, 120/240 VAC
PPX:560–2125	Chassis, 24 VDC
PPX:560–2126A	RF Remote Channel Controller (RCC) Card 2048 I/O Point Capacity per Card 2048 Control Relays per Card 15000-ft Channel Distance
PPX:560–2127	Twisted Pair Remote Channel Controller (RCC) Card
	2048 I/O Point Capacity per Card
	2048 Control Relays per Card
	3000-ft Channel Distance
PPX:560–2128	Hot Backup Card (Need Qty. of 2)
PPX:560–2129	Hot Backup Upgrade Kit (contains 2 PPX:560–2128 cards)
PPX:560–2130	64K Word Memory Expansion Card

Chapter 3

TI565T

3.1 TI565T

Advantages

The TI565T provides the application solutions for hybrid processes. With the Main CPU executing ladder programs and coordinating communications while the SF CPU handles loops, special function programs, and analog alarms, the TI565T is your choice for a controller that can handle your entire application.

Features

The Special Function programming statements enable you to use complex math computations, conditional statements, data tables, and other needed functions simply by filling in the memory locations to be used. The loops and analog alarms are also easy to program by using the same fill-in-the-blanks menu concept.

The TI565T SF CPU automatically allocates memory for loops, analog alarms, and special function programs. No tedious forms are required to track the SFs in memory.

You can program up to 64 PID loops with a choice of options that include the following features.

- Position/Velocity Algorithm
- Ramp/Soak Setpoint
- Lock Setpoint
- Derivative Gain-Limiting Algorithm
- Freeze Bias if Output Out of Range
- Rate-of-Change Alarm

In addition, the TI565T provides up to 128 analog alarms for handling analog signals with the precision you use for loop alarms.

3.2 Chassis

The TI560T or TI565T chassis (Figure 3-1) is a 19-inch, rack-mountable housing which can accommodate up to eight cards. The chassis contains a backplane to which the system cards are physically connected.

The system power supply is mounted in the far left side of the chassis and does not occupy any of the 8 slots available for cards. The first slot next to the power supply is reserved for the Main CPU card. The remaining slots can be used for memory expansion, I/O expansion, SF CPU option, and the Hot Backup option. These cards can be installed in any order; however, there must be no blank slots between cards.

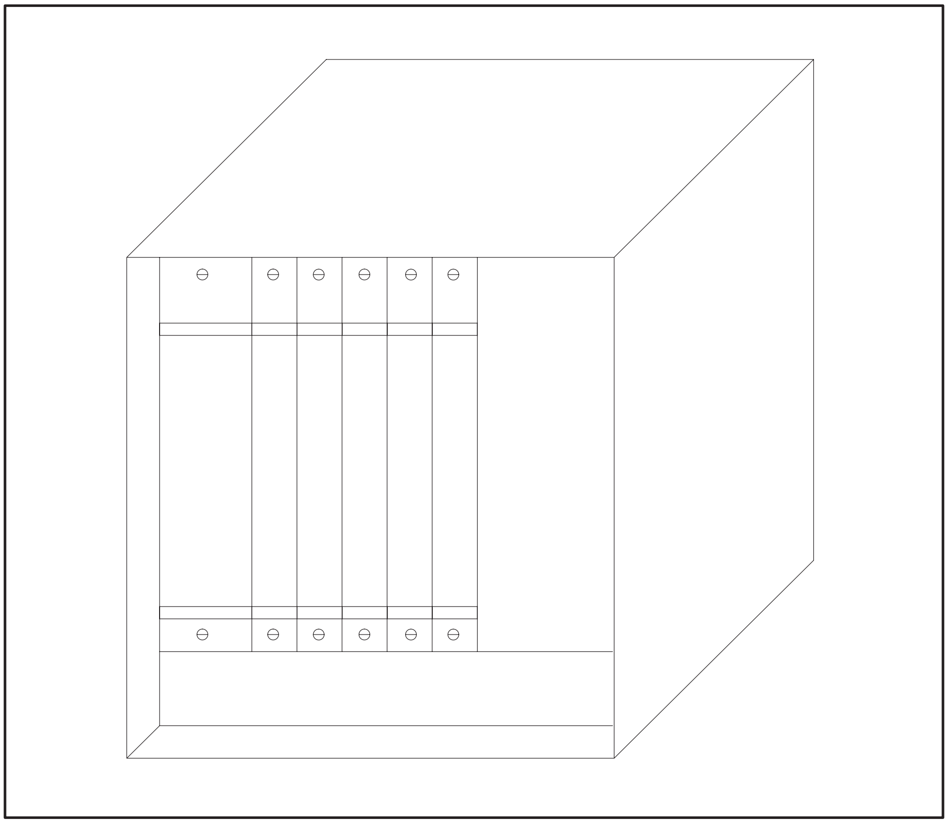


Figure 3-1 TI565T Chassis

3.3 Chassis Power Supply

The TI560T or TI565T chassis power supply (Figure 3-2) is a plug-in module which resides in the left side of the chassis. You may select either a 110/220 VAC or a 24 VDC power supply. This module supplies power to all the cards in the chassis. Use the PPX:560T–2125 chassis with the 24 volt power supply and the PPX:560T–2124 chassis with the 110/220 volt power supply.

The AC power supply provides 112.5 watts of 5 V power to the system and 1 amp of 5.10 volt battery backup power. Individual cards do not have battery backup. The DC power supply affords 147.5 watts of 5 V power to the system.

A memory-protect lock, power LED, and battery status LED are located on the front panel of the power supply. The battery on-off switch, battery, AC power connection, and fuse are also accessible through the front panel. Selection of 110 VAC or 220 VAC is done using an internal switch.

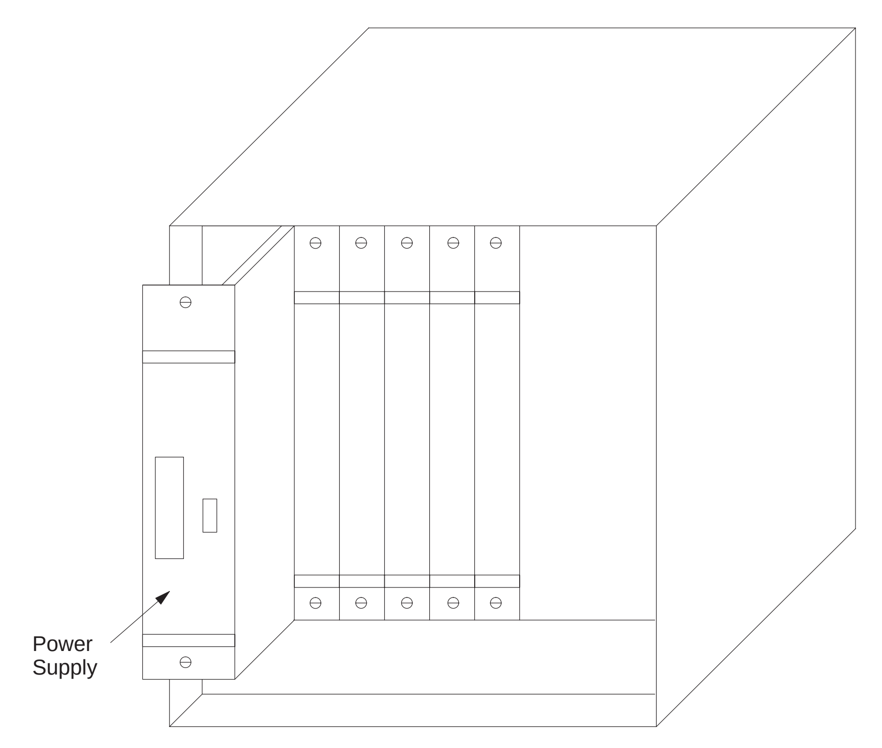


Figure 3-2 TI565T Power Supply Location

3.4 Special Function CPU Card

The TI565T requires the addition of a SF CPU (Figure 3-3) into an open chassis slot of a TI560T system. The chassis slot adjacent to the power supply is reserved for the Main CPU. The SF CPU card is an MC68000 microprocessor-based device which runs in parallel with the Main CPU. The SF CPU card provides two additional communications ports. Both ports are RS-232–C and provide for ASCII message output only.

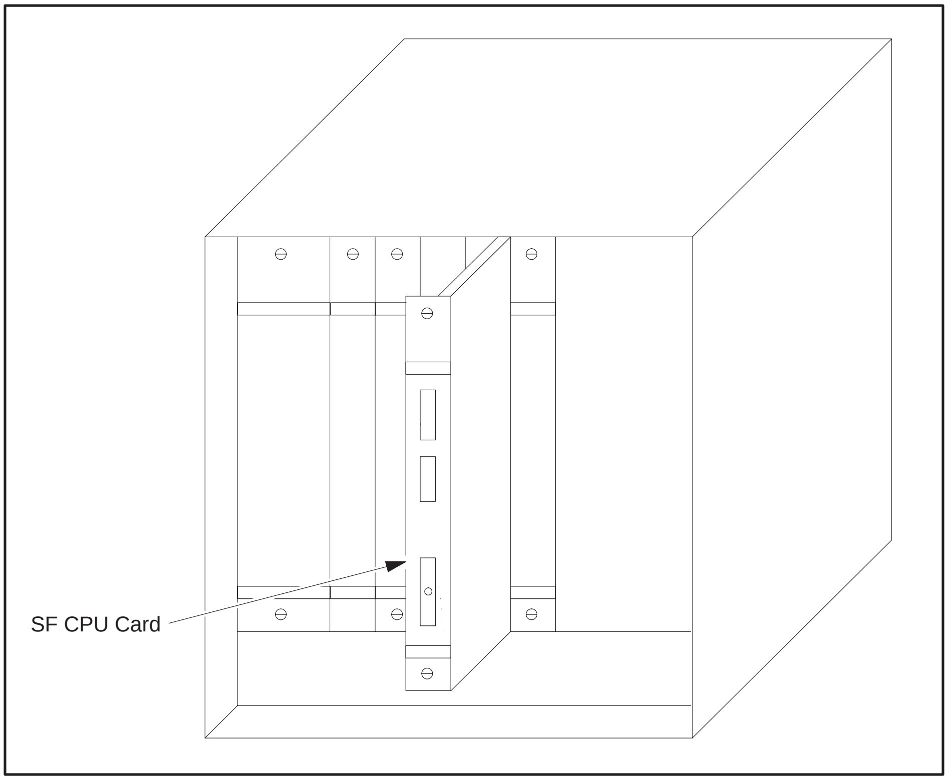


Figure 3-3 TI565T

3.5 Special Function CPU Operation

SF CPU Functions

The SF CPU executes processes termed Loops, Analog Alarms, and Special Function programs asynchronous to the controller timeline. During the I/O cycle, if the SF CPU is running a process that needs to read an I/O point, this process is stopped and the next priority process is started. Once the I/O cycle completes, the SF CPU resumes running the previous higher priority process.

SF CPU Communications

The Main CPU is idle during SF CPU communications and gives the SF CPU full access to the bus. This time period typically is less than 1.0 millisecond, but is dependent on the number of Special Function programs being called from ladder logic and other requests queued for processing by the SF CPU. Figure 3-4 shows the relationship among the cards.

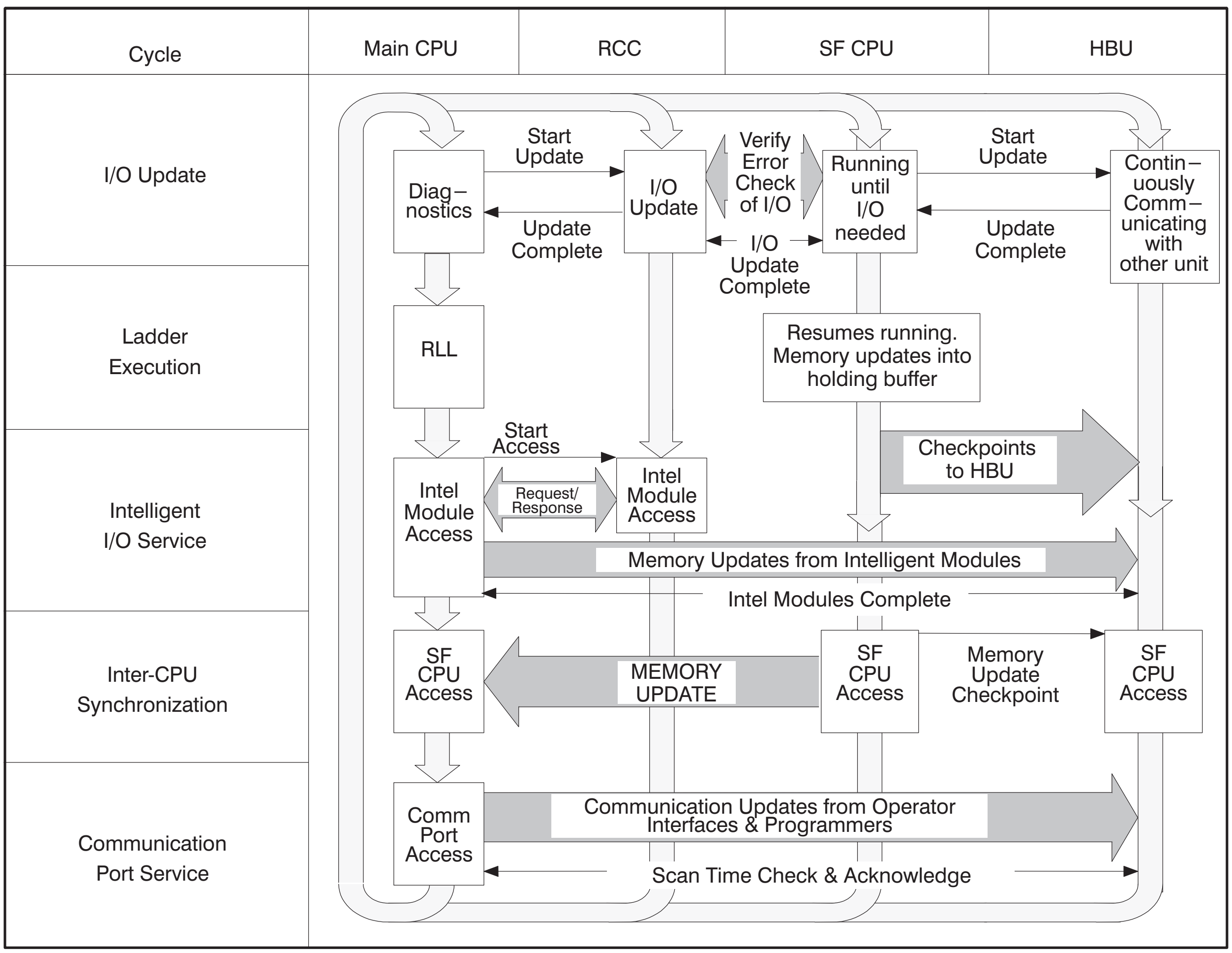


Figure 3-4 SF CPU Functions

3.6 Special Function Programming

SF Programs

The TI565T offers application solutions with Special Function programs and subroutines.

Special Function programs can be called on a timed basis (Cyclic), from ladder logic, from PID loops, or from Analog Alarms. Additionally SF programs can be identified as Priority or Non-priority. Priority SF programs are allocated twice as much execution time as non-priority programs.

In the TI565T, user-defined Special Function Programs consist of Special Function statements. In addition, SF Programs may call SF Subroutines. Each Subroutine is assigned a unique number and can be called from one or more Special Function Programs. SF subroutines can also call other subroutines up to a total of four levels deep.

TI565T SF PROGRAMMING SET

COMMENT	Provides comment capabilities.
BINARY-BCD	Performs a Binary to BCD conversion.
BCD-BINARY	Performs a BCD to Binary conversion.
SCALE	Converts Binary to Engineering units.
UNSCALE	Converts Engineering units to Binary.
MATH	Allows the user to enter math equations in a natural algebraic manner using the usual 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. Math statements support the following operators.

**	exponentiation
*	multiplication
/	division
+	addition
−	subtraction/unary minus
:=	assignment

In addition to the above operators, the Math SF provides the following functions:

ABS	absolute value
TRUNC	truncate, return integer
FRAC	return fraction
ROUND	round, return integer
CEIL	return smallest integer greater than or equal to x
FLOOR	return largest integer less than or equal to x
SQRT	square root
EXP	exponential
LN (Base e)	natural logarithm
LOG (Base 10)	common logarithm
SIN	sine—in radians
COS	cosine—in radians
TAN	tangent—in radians
ARCSIN	inverse sine—in radians
ARCCOS	inverse cosine—in radians
ARCTAN	inverse tangent—in radians

TI565T SF PROGRAMMING SET (continued)

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.

SDT Sequential Data Table

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 output. When the end of the table is reached, the index is reset to point 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.

TI565T SF PROGRAMMING SET (continued)

SSR Sequential Shift Register	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. The data shifted out of the last position in the table 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 synchronized shift register is especially suited to following parts down an assembly line.
FTSR IN	Fall Thru Shift Register Input. 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 output a message through the ASCII message ports. This statement may be used to print both text and the contents of integer and real variables.
CALL	The CALL Special Function is used to call an SF subroutine. Up to 1023 subroutines can be created in the TI565T. Each subroutine is given a unique number, once the SF 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	The PACK SF provides a means to easily pack data into contiguous blocks. This SF can pack Xs, Ys, Cs, WXs, WYs, or K or V words into blocks. This SF is most useful for transmission of data on a communications network such as TIWAY.
LEAD/LAG	The Lead/Lag SF allows the user to perform a Lead/Lag function on any analog variable. Typically used in a process control Loop as a compensator in dynamic Feed-Forward control. This SF may only be invoked from Loops, Analog Alarms, or Cyclic SF programs. In specifying the Lead/Lag SF the Input and Output words are defined along with the lead-time in minutes and the lag-time in minutes plus a gain.

TI565T SF PROGRAMMING SET (continued)

RETURN The Return SF returns from a subroutine. If invoked from an SF program, the SF program is terminated.

PACK LOOP The Pack Loop data packs loop data from a loop into contiguous blocks.

PACK ALARM The Pack Analog Alarm SF packs Analog Alarm data from an analog alarm into contiguous blocks.

3.8 SF Program Example

Programming a Special Function (SF) in the TI565T begins with calling up the directory that lists the programs by title and number as shown in Figure 3-5.

SPECIAL FUNCTION PROGRAM DIRECTORY

SFPGM	PROG	TITLE			
1	YES	MTR CTRL	17		
2	NO		18		
3	YES	BURN #2 :	19		
4			20		
5			21		
6			22		
7			23		
8			24		
9			25		
10			26		
11			27		
12			28		
13			29		
14			30		
15			31		
16			32	YES	VLV P34

Figure 3-5 Special Function Program Directory

To program a Special Function, fill in a menu as shown in Figure 3-6.

```
SF PROGRAM # 5
                TITLE =                                        BOILER 2
                CONTINUE ON ERROR (YES/NO):                    Y
                PROGRAM TYPE (N,P,C,R) =                       C
                ERROR STATUS ADD:                              V500
                CYCLE TIME (secs) =                            +0.5000

0001  *       CALCULATE FLOW AND MASS FLOW RATE TO BOILER 2. GET
0002  *       SIGNAL, TAKE SQUARE ROOT, ADD CORRECTION AND SAVE.
0003  MATH    V200 := SQRT (WX100) + V250
0004  *       THE CORRECT FLOWRATE IS NOW SAVED IN CUBIC
0005  *       FEET/MIN). CALL SUBROUTINE 2 TO CALCULATE
0006  *       MASS FLOWRATE FROM TEMP AND FLOWRATE
0007  CALL    SFSUB: 2 P1: V200  P2: V110  P3: V202  P4:  P5:
0008  *       THE MASS FLOWRATE IS STORED IN LOCATION V202, IN (LBS/MIN)
              IF THERE IS A FLOWRATE MISMATCH WITH BOILER 1, SET FLAG.
0009  IF      (V202–V102) > 150
0010  IMATH   V100 := 1
0011  ELSE
0012  *       ELSE RESET FLAG
0013  IMATH   V100 := 0
****END****
EXIT–F1  UP–F2  DOWN–F3  FIND–F4  EDIT–F5  INS–F6  DEL–F7
```

Figure 3-6 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 found. The first word in the three word address will contain an error code that determines what type error was encountered. The second word will contain the number of the SF program or SF subroutine where the error was encountered. And the final word will contain the last fully executable line before encountering the error. Now, the error can be quickly identified and corrected.

3.9 PID Loops

The TI565T PID equation has available two operating modes which may be selected by the user for the best fit to the application.

PID Algorithm Types

Position Velocity

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 is thus known as the Position equation and is the implementation which should be used in most cases.

The velocity form of the PID equation is provided as an option on the TI565T. It can be derived by taking the first derivative of the Position equation; and 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.

A loop flag address can be configured to contain status information about the loop. For example, this address contains information that tells the ladder logic if the loop is in manual, auto, or cascade, or if the loop is in an alarm condition.

A Ramp/Soak option is also available. This option allows automatic setpoint changing as a function of time. The user defines the setpoints and the ramp rates. A total of 256 steps are allowed per setpoint. 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.

A derivative gain limiting algorithm is present to limit the derivative contributor to the gain expression. Since the coefficient of the derivative term in the PID difference equation is inversely proportional to the sample rate, when the sample rate is very small, noise disturbances to the process variable (PV) can cause the loop to over-react. Derivative gain limiting is a filter applied to the PV in the derivative term to limit this disturbance.

A Special Function Program may be called from any loop to operate on the setpoint (SP) or the Process Variable (PV). The SF program will be called when the SP or PV is accessed; it does not necessarily have to operate on either. The SP is accessed each time the loop PID calculation is done as required by the sample rate. The PV is accessed every 200 msec (minimum) regardless of the operating mode of the loop. Here SFs 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). Temporary location T2 within the program indicates which option is running.

A clamp setpoint limits option is available to prevent the setpoint from being changed by an operator interface.

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 by setting the coefficients not required to zero for gain and rate, or setting them to infinity for reset.

Reset windup protection is inherent to the design of the TI565T even though it is not mentioned in the menu.

A freeze bias option is provided that will freeze the bias when the output becomes greater than 100%. This operation is used to determine how the loop performs reset-windup protection. Normally, reset-windup protection is performed by back-calculating the bias term. With the freeze bias option "on", the bias term is simply "frozen". Therefore, the controller will not continue to instruct a valve to open past its effective limit, which would be 100% open.

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 deadband is selectable from 0.2% of span to 5.0% of span. A Lead/Lag Special Function allows the user to easily implement dynamic feed-forward control. The lead/lag function is discussed in the section under Special Functions.

For details on the operation of the TI565T PID algorithm, see Appendix B.

PID Loop Entry

Programming a PID loop in a TI565T begins with a Loop Directory that identifies the loops by number and title. After selecting a loop to program, fill in a menu with the required information as shown in Figure 3-7.

```
        PID LOOP 64     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: 384                                   ENABLED

                                               565T          OFFLINE

   EXIT–F1     EDIT–F2                    COMMNT–F7   EN/DIS–F8
```

```
           ALARM DEADBAND:  +0.00000            MONITOR DEVIATION:  NO
                                     DEVIATION ALARM: YELLOW =  +100.000
    MONITOR LOW-LOW/HI-HI:  NO                          ORANGE =  +100.000
       MONITOR LOW/HIGH:    NO
 PV ALARMS:      LOW-LOW =  +0.00000      MONITOR RATE OF CHANGE:  NO
                     LOW =  +0.00000         RATE OF CHANGE ALARM:  +0.00000
                    HIGH =  +100.000
               HIGH-HIGH =  +100.000      MONITOR BROKEN XMITTER:  NO

S-MEMORY AVAILABLE: 384                                   ENABLED

                                               565T          OFFLINE

   EXIT–F1     EDIT–F2                    COMMNT–F7   EN/DIS–F8
```

Figure 3-7 PID Loop Menu Example

3.10 Analog Alarms

The Analog Alarm function in the TI565T allows monitoring of analog data as if it were in a Loop. The alarm address is specified and the alarm setpoints are selected. Eight alarms are available:

TI565T Analog Alarms

High-High

High

Low

Low-Low

Yellow Deviation

Orange Deviation

Rate of Change Engineering Units/Min.

Broken Transmitter

Alarm Limits and Configurations in Engineering Units

Alarm limits are configured in Engineering units. In addition, the PV alarms deadband is adjustable from 0.2% to 5% of the span.

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 TI565T begins with an analog alarm directory which lists the Analog Alarms by number and title.

After the analog alarm is selected for programming, the user provides the information to complete a menu as shown in Figure 3-8.

```
ANALOG ALARM #(1–128)                TITLE:   XXXXXXXX
SAMPLE RATE (secs)           .............  +2.00000   ALARM FLAG ADDRESS:..........          XXXX
PROCESS VARIABLE ADDRESS             .....  XXXXXX     PV IS BIPOLAR              ..........................N
SQUARE ROOT OF PV            .............  N          20% OFFSET ON PV                  .................N
PV RANGE    ..................      HIGH =  +1.00000
                                     LOW =  +0.00000
SPECIAL FUNCTION             .................  N      SFPGM NUMBER:                   .................XXXX
SP VALUE OR REMOTE SP                ........  SP VALUE  VALUE OR ADDRESS:                       +0.00000
CLAMP SETPOINT LIMITS: HIGH =               +0.00000
                                     LOW =  +0.00000
ALARM DEADBAND               ................. =  +0.00000  (ENGR. UNITS)
PV ALARMS      .........          LOW-LOW =  +0.00000  (ENGR. UNITS)
                         ................LOW =  +0.00000  (ENGR. UNITS)
                        .................HIGH =  +0.00000  (ENGR. UNITS)
                        ..........HIGH-HIGH =  +0.00000  (ENGR. UNITS)
MONITOR L-L/H-H ALARM              ...........  N      MONITOR L/H ALARM                 ...............N
DEVIATION ALARMS........          YELLOW=  +0.00000  (ENGR. UNITS)
                         ........ ORANGE=  +0.00000  (ENGR. UNITS)
MONITOR DEVIATION ALARMS            ......  N
RATE OF CHANGE ALARM              ......... =  +0.00000  (ENGR. UNITS/MIN)
MONITOR RATE OF                                        MONITOR BROKEN
   CHANGE ALARM              ................  N          TRANSMITTER           ......................N
ABORT–F1   UP–F2   DOWN–F3   ENTER–F4
```

Figure 3-8   Analog Alarm Menu

3.11 TI565T Controller Component List

The following list provides the part numbers for components required to assemble a TI565T controller to fit your application.

Part Number	Description
PPX:565T–1101	Basic TI565T System, 120/240 VAC. Consists of complete PPX:TI560T–1101 system described earlier, plus: 1 SF CPU Card (PPX:565–2120) w/64 Menu-Driven PID Loops, Special Function Programming and Subroutines, and 128 Menu-Driven Analog Alarms
PPX:565T–1102	Basic TI565T System, 24 VDC. Consists of complete PPX:TI560T–1102 system described earlier, plus: 1 SF CPU Card (PPX:565–2120) w/64 Menu-Driven PID Loops, Special Function Programming and Subroutines, and 128 Menu-Driven Analog Alarms
PPX:565–2120	SF CPU Card

Chapter 4

Hot Backup and System Diagnostics

4.1 Hot Backup Card

The TI560T/TI565T can be configured in a Hot Backup (HBU) system. The Hot Backup option provides controller redundancy for critical applications that require minimum unscheduled process downtime.

An HBU system consists of two identical TI560Ts or TI565Ts, each with an HBU card. One unit is designated Active and the other, Standby. The HBU card automatically programs and transfers control to the Standby unit upon detection of defined fatal errors in the Active unit.

The TI560T/TI565T HBU system provides controller redundancy with a minimum of additional effort. As shown in the feature list below, the HBU unit requires no third box nor additional programs to operate.

Features

MC68000 Microprocessor-based

Communications
- 1 MHz serial fiber-optic link
- 1 transmit, 1 receive

Performance
- Degrades Controller performance less than 10 msec per scan

Displays
- HBU Good, Active, Standby, and Comm Good LEDs

No Additional User Programming Required

No Effect on Active Unit if Standby Unit is Disconnected

No Third Box Necessary

Automatic Program Transfer from Active to Standby Unit

Automatic Program Updating

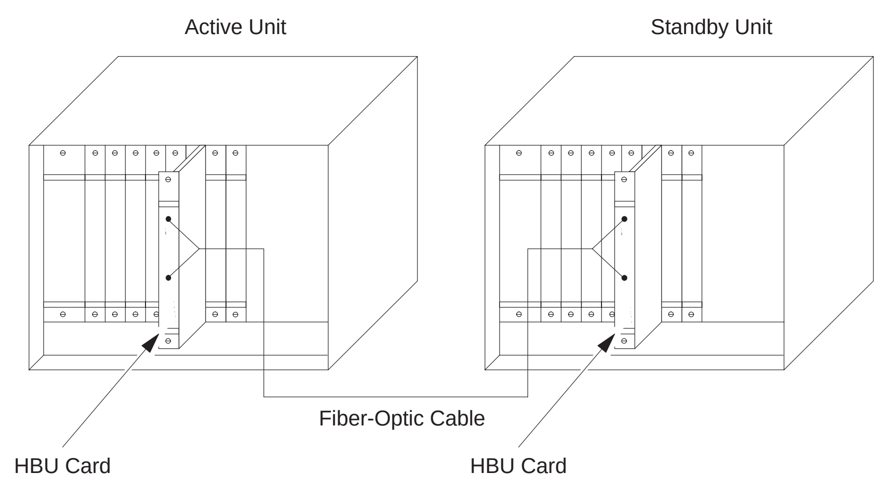


Figure 4-1 Hot Backup Units

Hot Backup Configuration

The TI560T and TI565T Hot Backup schemes provide the capability to transfer control to a standby TI560T or TI565T controller in the event of a system fatal error in the active unit. This type of redundant configuration requires two identical TI560T or TI565T systems and two Hot Backup cards. See Figure 4-1.

The HBU card may be inserted anywhere in the TI560T or TI565T chassis, with the exception of the discrete CPU slot which is adjacent to the power supply.

The HBU card is an MC68000 microprocessor-based card that communicates with the Main CPU, RCC cards, and another HBU card. The communication to the Main CPU is done through the chassis internal bus. The communication to the other HBU card is done through an optical cable over a 1 MHz serial link.

Both TI560T or TI565T units are connected to the I/O chain through a combination of 3 dB line splitters. Figure 4-2 illustrates the communication cycle between the active and standby units.

4.2 Hot Backup Execution

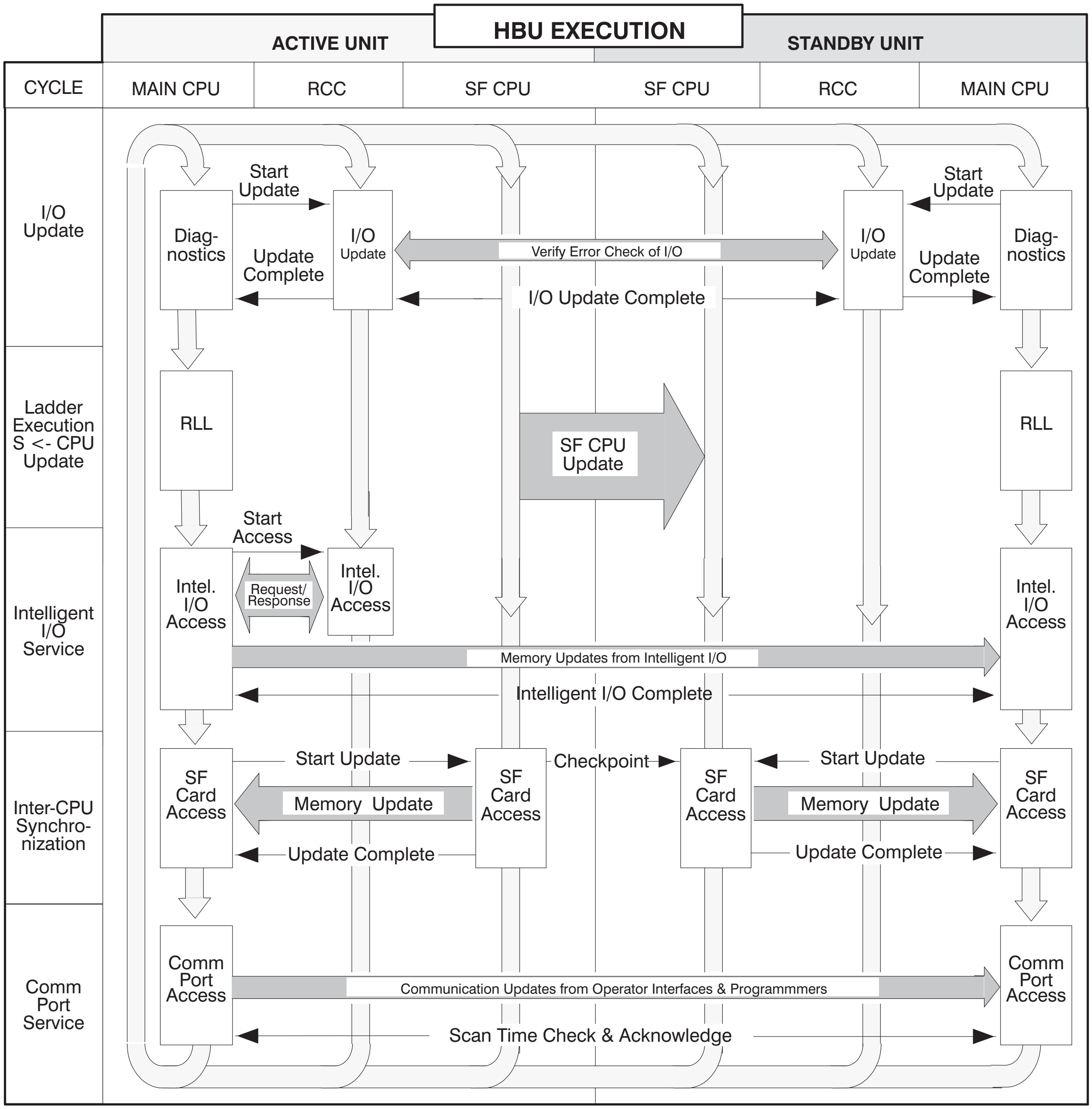


Figure 4-2 Hot Backup Updates

Operation

Following installation of the two HBU cards and connection of the cable, the standby unit will power up in the off-line mode. On power-up, the standby unit is downloaded automatically by the active unit unless the Scan Synch Inhibit instruction is activated. I/O activity is frozen during the download.

During the download (five seconds or less, depending on memory size) the active unit sends all transmissions to the standby unit enabling the standby unit to copy the active unit's program. When the transmission is complete, the standby unit will be online and both units will be fully synchronized. Also, there is no additional user programming necessary to achieve HBU functionality; it is all in the firmware.

The standby unit monitors the I/O channel under control of the active unit, in a listen-only mode. If an I/O message is missed due to transmission faults, the standby unit will request that the active unit do a re-transmission. The standby unit may not transmit on the shared I/O system. The standby unit receives input data for storage into its image register and uses it in the next scan.

Both units execute identical ladder logic programs operating on identical image register (IR) and user memory.

The standby unit uses a number provided by the active unit to decrement timer and drum current counts (Delta T).

Any operations that change the IR or user word memory, asynchronous to the execution of the ladder program and I/O cycle, are provided to the standby unit during the HBU time slice. In the event of a detected active unit failure, the HBU communication goes dead, an I/O listen takes place, and the standby unit assumes control.

4.3 Hot Backup Synchronization

With an SF CPU card installed, check-pointing is used for synchronization. Only the active unit executes loops, analog alarms, and SF programs. During each scan the active unit passes the following to the standby unit:

- IR updates and checkpoint information to provide an indication where loops, analog alarms, and SFs are in the processing sequence.
- S-memory edit task code requests.

At the end of a scan the active SF CPU card sends an “update memory” message to the standby SF CPU card; after this, both units will dump all image register (IR) updates queued during the scan to memory.

In the event of an HBU failure in the standby unit, control will stay with the active unit. If a failure is detected in the active unit, control is passed to the standby unit. The standby unit then becomes the active unit.

Operator interface devices or programming devices may be connected to the standby unit. However, if the standby is ON-LINE, any devices attached to the standby unit will be in a read-only mode. In OFF-LINE mode, other functions are available.

Control switchover to the standby unit occurs only in the event of a detected system fatal error or power loss in the active unit. The controller uses normal system diagnostics to verify system integrity; no special diagnostics are executed with the hot backup option. However, there are special hot backup routines to determine if the active unit is talking to the I/O channel before the standby takes control.

In hot backup mode, the two systems check each other four times per scan to ensure system synchronization. Since the standby fully updates its image register each scan, it is ready to take control at any time. The standby unit executes the ladder logic program and places these results in its image register. The SF CPU constantly sends its updates to the standby. Both systems run diagnostics each scan, so system status is constantly monitored.

4.4 Diagnostics

To help obtain maximum uptime with a minimum of effort, the TI560T and TI565T provide diagnostic routines to check system functionality. Using these routines, you can determine the operation of hardware and the status of the software. With a few simple commands and visual checks, you can obtain information that could take hours to assemble without the diagnostic routines.

Three types of system diagnostics are available.

- Power-up: performed automatically at system start-up.
- Run-time: performed automatically during controller operation.
- User-initiated: provide system information upon request.

4.5 Power-Up Diagnostics

Checks Performed

Upon power-up, the TI560T or TI565T tests operation of the microprocessors, checks that RAM and ROM memory are good, verifies communications among the cards, and runs communication port tests. If all tests are passed, the PC GOOD LED on the front of the Main CPU card comes on. If an error is found during power-up diagnostics, the PC GOOD LED does not come on. Use the programming device to call up a display to indicate which error exists. The lists on the following page give the errors that may appear according to controller and error-type.

4.6 Error Messages

TI560T/TI565T Fatal Errors

O. S. RAM Parity

Program RAM Parity

Operating System Fatal Error

ROM Diagnostic Failure

I/O Fatal Error

Abnormal Power Loss

Unidentified Failed Card Present

Watchdog Timeout

Dynamic Program Memory Diagnostic

Illegal Op Code

RAM Diagnostic Failure

Fatal Errors TI565T only

SF ROM Error

SF RAM Error

SF O. S. Error

Invalid Control Block Encountered

Diagnostic Failure

Memory Parity Error

S Memory is Inconsistent

SF Program Number Received from Ladder Logic is Invalid

TI560T/TI565T Non-fatal Errors

Scan Overrun

I/O Base Failure

Special Function Port Failure

I/O Module Failure

Over Temperature

Board Marked Non-fatal has Failed

Non-fatal Errors TI565T only

Loops are Overrunning

Analog Alarms are Overrunning

Cyclic SF Programs are Overrunning

Normal SF Program Queue is Full

Priority SF Program Queue is Full

Cyclic SF Program Queue is Full

Error Occurred During Loop Calculations

Error Occurred During Analog Alarm Calculations

A Control Block is Disabled

Attempt to Execute an Undefined SF Program or SF Subroutine

Attempt to Invoke a Restricted SF Program from ladder logic

SF Comm. Port 1 Comm. Error

SF Comm. Port 2 Comm. Error

Memory in Standby unit does not match Active

4.7 Run-Time Diagnostics

The following diagnostic routines are run during operation of the controller.

- ROM “Fletcher” Checksum Test
- RAM Checksum
- RAM Parity on each RAM Access
- Watchdog Timer on Inter-Processor Communication and Acknowledgments, as well as major functions of controller timeline
- Various Sanity Checks on the Loop, Analog Alarm, and SF Program Variables
- Consistency checks on S memory Control Block Structures
- Invalid Instruction and Spurious Interrupt Traps
- CRC Checks on all Remote I/O Communication
- Early Power-Fail Detection and Controlled Switchover to battery Backup

If a fatal error is encountered, the controller halts operation and the PC GOOD LED is turned off.

4.8 User-Initiated Diagnostics

User-Initiated Diagnostics are available via a programming device and are accessed through the Auxiliary Functions Menu. These user-initiated diagnostics are used for troubleshooting the TI560T and TI565T. The diagnostics routines available from the Auxiliary Functions Menu are outlined as follows.

Run controller Diagnostics

This routine runs a set of diagnostics on each of the cards installed in the system chassis. When the diagnostics are complete, the message “Diagnostics passed” is displayed if no errors were found. Otherwise, one or more of the following error messages is displayed for each slot with a board installed.

Error Messages

O/S RAM FAILED

PROGRAM RAM FAILED

IR RAM FAILED

ROM FAILED

SIMULATED SCAN FAILED

ENABLE RAM FAILED

Run Remote Base Diagnostics

This routine runs diagnostic checks on each remote base controller configured in the system. When the diagnostics are complete, one or more of the following messages is displayed.

Remote Base Messages

DIAGNOSTICS PASSED

O/S RAM FAILURE

PROGRAM RAM FAILURE

IR RAM FAILURE

ENABLE RAM FAILURE

ROM FAILURE

SIMULATED SCAN FAILURE

WATCHDOG TIMER FAILURE

COMMUNICATIONS PORT FAILURE

NON-VOLATILE RAM FAILURE

REQUESTED DEVICE NOT FOUND

Display Failed I/O

This routine initiates a status check of the configured I/O modules by the Main CPU. When the check is complete, the location of failed I/O modules is displayed in chart form on the programming device. The failures are identified by Channel, Base, and Slot.

Show Controller Diagnostics Cell

This routine calls a display which supplies information for the items shown in Figure 4-3.

TI560T OPERATIONAL STATUS

SCAN = (TIME, FIXED OR V)	BATTERY = (GOOD, LOW)
KEY = (LOCKED OR UNLOCKED)	DOWNLOAD = (ON OR OFF)
MODE = (RUN OR PROGRAM)	PROGRAM IN RAM
SF/LOOP MODE = (RUN OR PROGRAM)	
SF/LOOP FOLLOWS P/C TO PROGRAM MODE	
FATAL ERROR	= (NONE OR ERROR TYPE)
NON FATAL ERROR	= (NONE OR ERROR TYPE)
LOOP FATAL ERROR	= (NONE OR ERROR TYPE)
LOOP NON FATAL ERROR	= (NONE OR ERROR TYPE)

Figure 4-3 TI560T Operational Status Screen

If errors are encountered, they will be listed by classification as fatal or non-fatal as indicated above. The type of error appears in the space next to the fatal or non-fatal error prompt.

4.9 Status Words

Status words are available in the TI560T and TI565T to indicate system status. This status information words can be used in either ladder logic or Special Function programs to take corrective action when unfavorable conditions occur. These status words are useful in troubleshooting situations.

TI560T

For the TI560T, status words provide both system and I/O status information as outlined in the following table.

TI560T Status Words

Word	Error Condition
1	Specific bits provide information on power supply, battery, ports, scan time, move insts.
2-9	Bases on Channels 1-8
11-26	Modules on Channel 1
27-42	Modules on Channel 2
43-58	Modules on Channel 3
59-74	Modules on Channel 4
75-90	Modules on Channel 5
91-106	Modules on Channel 6
107-122	Modules on Channel 7
123-138	Modules on Channel 8

Status Words (continued)

TI565T

Status Words 161 and 162 enable you to read fatal and non-fatal error status in the TI565T. Access of the error data through status words provides a means for incorporating alarms into ladder logic programs to alert operators when an error occurs. The errors designated by the bits in the status words are given in the following tables.

Status Word 161

Bits 1-8 set to 1 indicate errors as follows:

1...............	ROM Error.
.1..............	RAM Error.
..1.............	Operating System Error.
...1............	Invalid Control Block Encountered.
....1...........	Diagnostic Failure.
.....1..........	Memory Parity Error.
......1.........	S Memory is inconsistent.
.......1........	SF Program Number received from ladder logic is invalid.

Bits 9-11 are unused, and are always 0.

...........1....	Alarm Enabled. (If a word is selected for the analog flags, Bit 12 is written. If a discrete is selected for the analog flags, Bit 12 is not written.)

Bits 13–16 are unused, and are always 0.

Status Word 162

1..............	Port 1 Communications Error. This error is logged if the TI565T board has a character queued for printing to Port 1 for more than 30 seconds.
.1.............	Port 2 Communications Error. This error is logged if the TI565T board has a character queued for printing to Port 2 for more than 30 seconds.
..1............	Loops are overrunning.
...1...........	Analog Alarms are overrunning.
....1..........	Cyclic SF Programs are overruning.
.....1.........	Normal SF Program Queue is full.
......1........	Priority SF Program Queue is full.
.......1.......	Cyclic SF Program Queue is full
........1......	Error occurred during Loop calculations.
.........1.....	Error occurred during Analog Alarm calculations.
..........1....	A control block is disabled.
...........1...	Attempt to execute an undefined SF Program or SF Subroutine.
............1...	Attempt to invoke a restricted SF Program from ladder logic.
....	Bit 14 is unused.
....	Bit 15 is unused.
....	Bit 16 is unused.

4.10 LEDs

Using LEDs as Diagnostic Tools

LED displays on the cards are a simple means for beginning troubleshooting. Each card has LED indicators that supply information on the status of that particular card. Figure 4-4 illustrates the location of the LEDs on the cards.

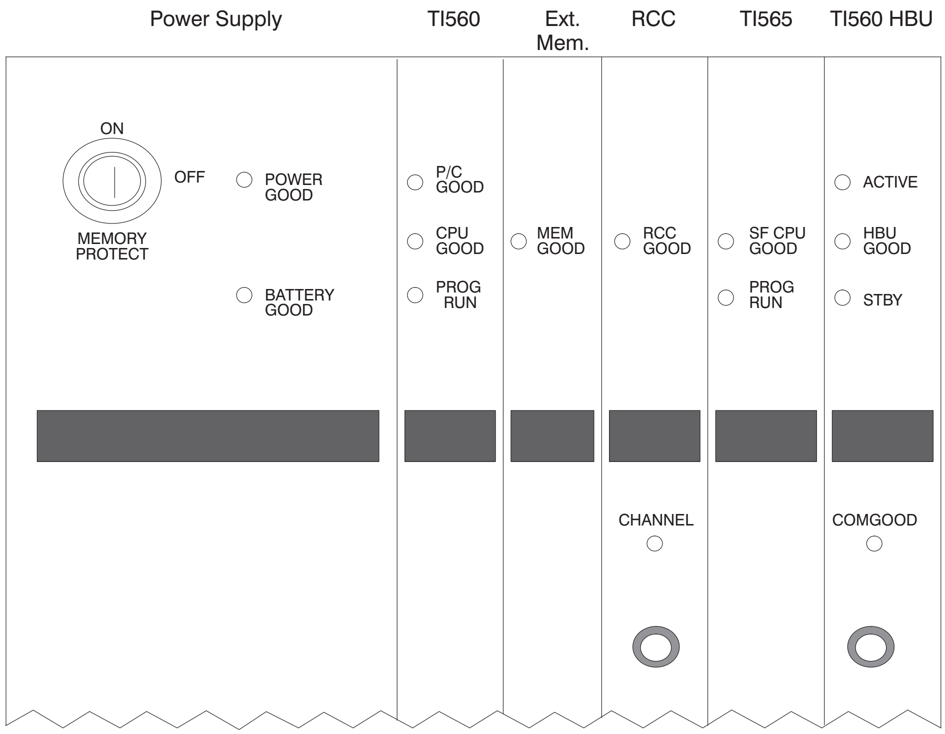


Figure 4-4 LED Locations

The LEDs shown above provide status information at a glance for each of the cards. Troubleshooting is expedited by starting from the primary controller components and working through them to the I/O modules and wiring.

Table 4-1 lists the indications provided by the LEDs on each card.

Table 4-1 LED Indications

LEDs	Indications
Power Good	Remains on as long as the power supply DC output voltage remains above the operating level..
Battery Good	Stays on as long as the memory back-up battery voltage remains above 2.4 volts, and system power is applied.
TI560T PC Good	When on, overall indicator of chassis operating integrity.
TI560T CPU Good	On indicates the CPU card is good.
TI560T PROG/RUN	LED on indicates the system is in the run mode.
TI565T CPU Good	On indicates the TI565 CPU card is good.
TI565T PROG/RUN	On indicates the system is executing Loops, Analog Alarms, and Special Function Programs.
RCC Good	On indicates the Remote Channel Controller has passed self-diagnostics.
Channel (top)	When on (may be flashing), indicates channel A is transmitting.
Channel (bottom)	When on (may be flashing), indicates channel B is transmitting.
MEM Good	On indicates the memory card is good.
HBU Good	On indicates integrity of card is good.
Active	On indicates this is Active Unit.
Standby	On indicates this is Standby Unit.
COMM Good	On indicates HBU Communications good.

Chapter 5

System Configuration

5.1 Overview

In this chapter, you will find dimensional drawings and component listings for the TI560T/TI565T to help you in planning your installation. Figure 5-1 shows a typical layout of the TI560T/TI565T with the Hot Backup option.

The TI560T/TI565T hardware is modular to make it more easily adapted to your specific needs. From the compact, 19-inch chassis that houses the power supply and cards to the I/O racks and modules, each component of a TI560T/TI565T system may be selected to conform to your particular application.

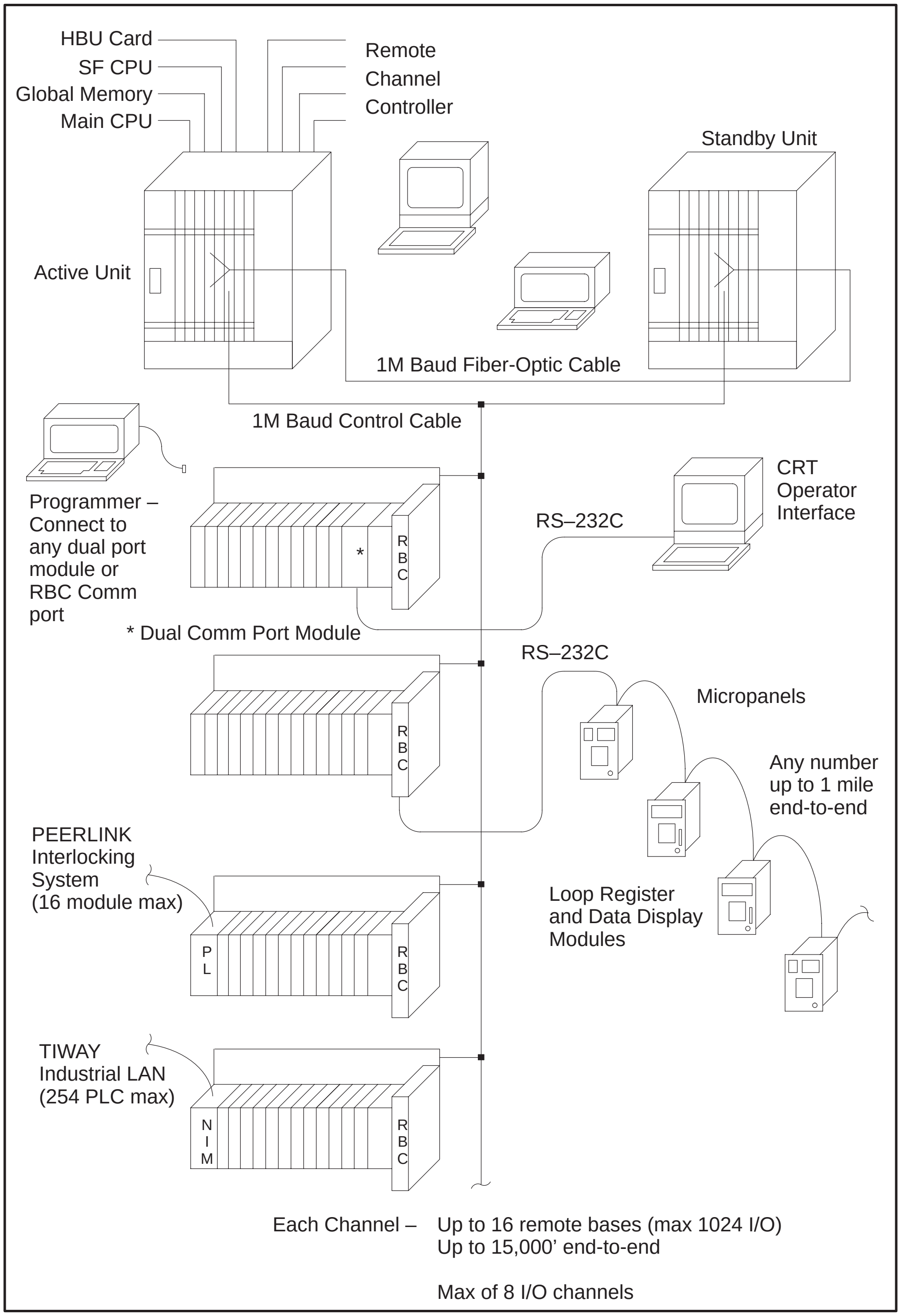


Figure 5-1 TI560T/TI565T with Hot Backup Option

5.2 TI560T/TI565T Chassis

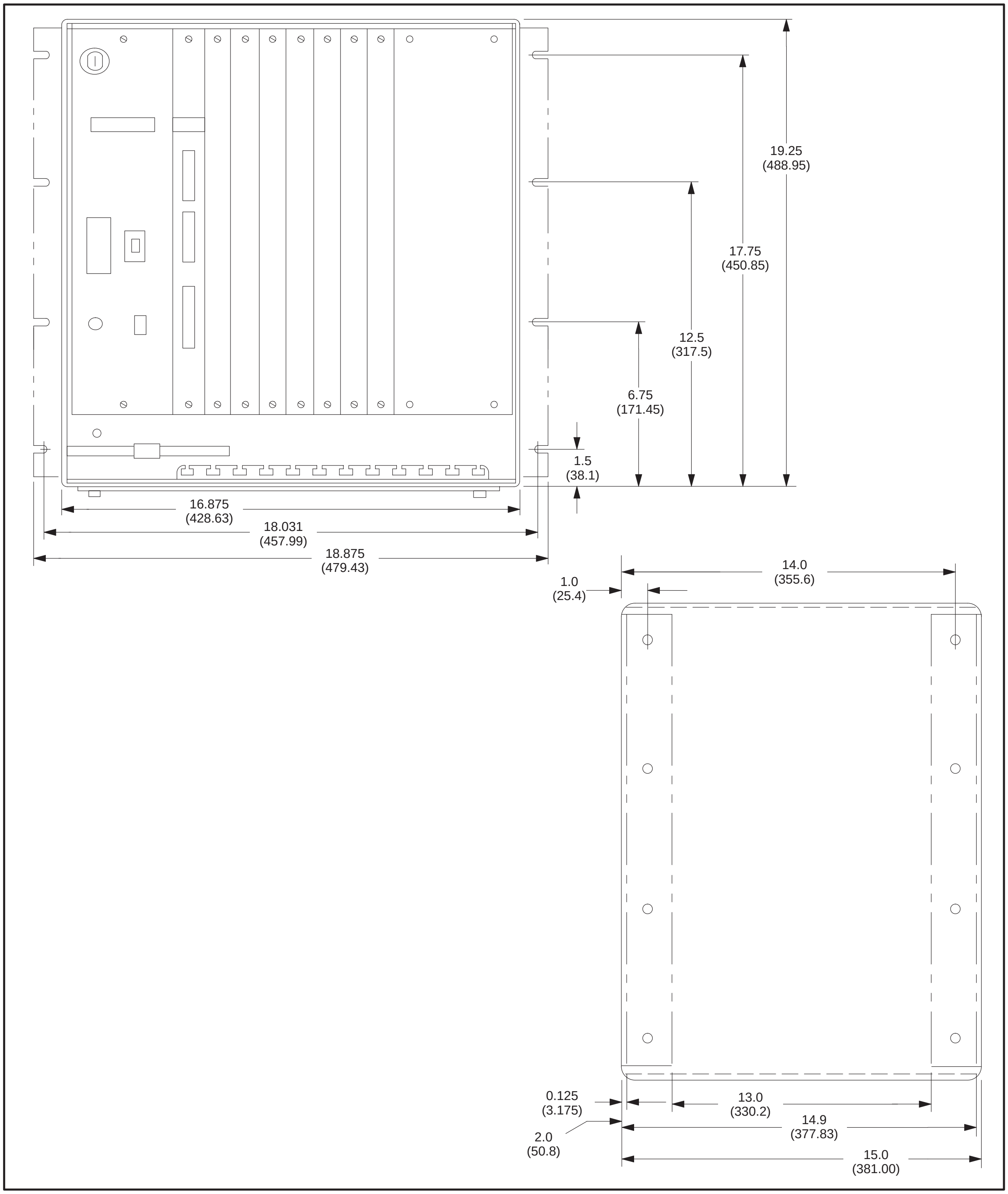


Figure 5-2 Mounting Dimensions for TI560T/TI565T

5.3 Remote I/O Components

Components

Table 5-1 provides the part numbers for I/O components to assemble a TI565T remote I/O system to fit your application.

Table 5-1 I/O Components

Part Number	Description
PPX:500–2114	Remote Base Controller (RBC) with RS–232 Communication Port
PPX:500–2151–A	Remote Base Power Supply (110/220 VAC)
PPX:500–5840	Adapter Base Kit for using existing 8- and 16-slot SIMATIC® TI520™/TI530™ Bases
PPX:500–5848	14-Slot I/O Base
PPX:500–5884	12-Slot I/O Base (mounts in 19-rack)
PPX:500–5892	6-Slot Base

Remote Base Controller

The RBC is an intelligent interface between a remote base and a Remote Channel Controller card in the TI560T/TI565T chassis.

Remote Base Power Supply

The remote base power supply converts user-supplied 110/220 VAC power into +5.1 and -5 VDC to power the base controller and I/O modules installed in the base.

Adapter Base Kit

The adapter base enables the 8- and 16-slot bases to be used with the TI560T/TI565T.

6-Slot I/O Base
14-Slot I/O Base

The 6- and 14-slot I/O bases contain a controller slot in addition to the I/O slots.

12-Slot I/O Base

The 12-slot base is sized for mounting in a standard 19-inch rack.

Redundant Base Controller Options

Table 5-2 shows the Redundant Base controller options.

Table 5-2 I/O Components

Part Number	Description
PPX:500–9914	Redundant RBC
PPX:500–9940	Redundant Adapter Base
PPX:500–5114	Twisted-pair RBC
PPX:500–6850	RF RBC
PPX:500–6851	Twisted-pair RBC
PPX:500–6660	P/S
PPX:500–6505, –6508, 6516	I/O Bases

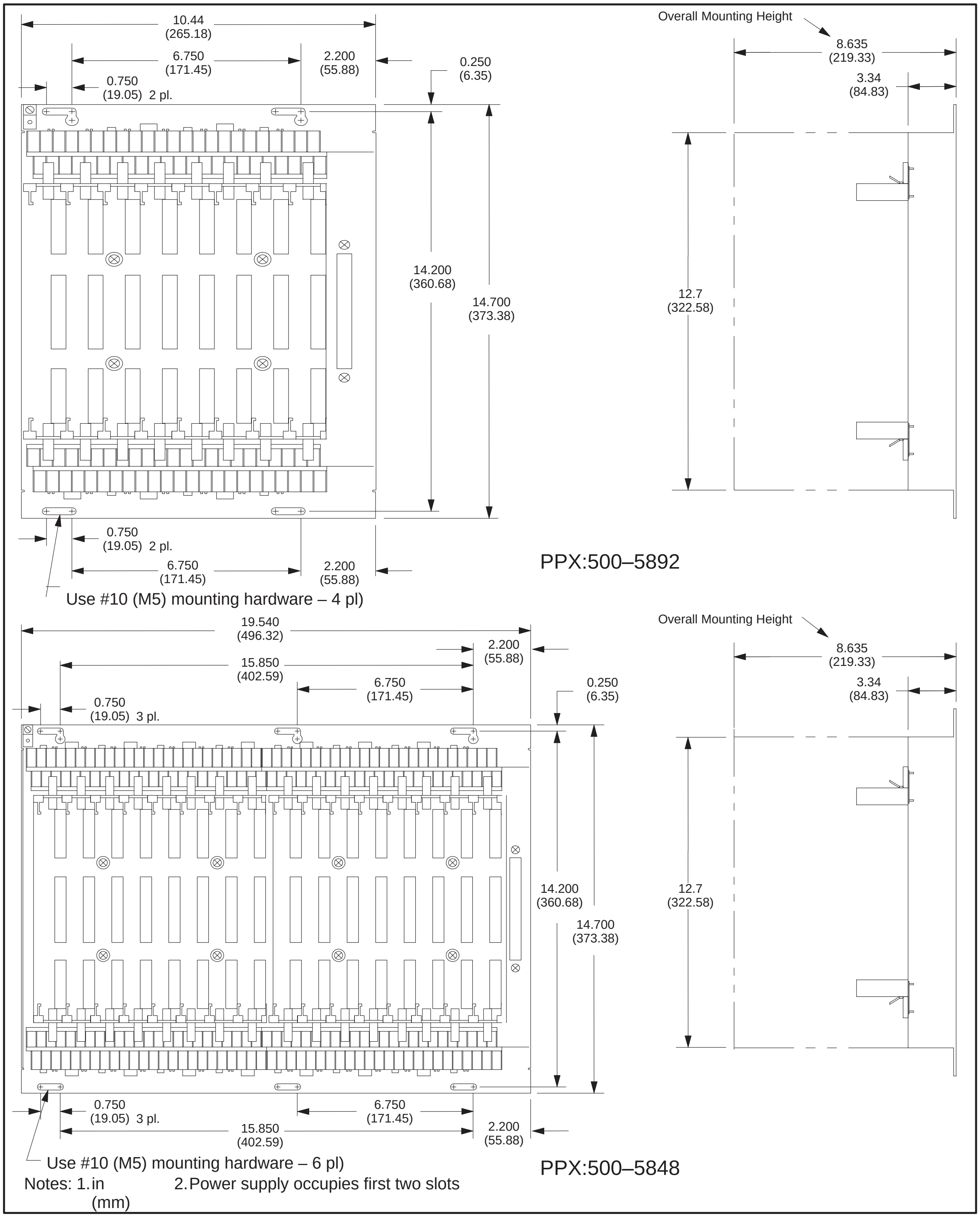


Figure 5-3 6-Slot and 14-Slot I/O Bases

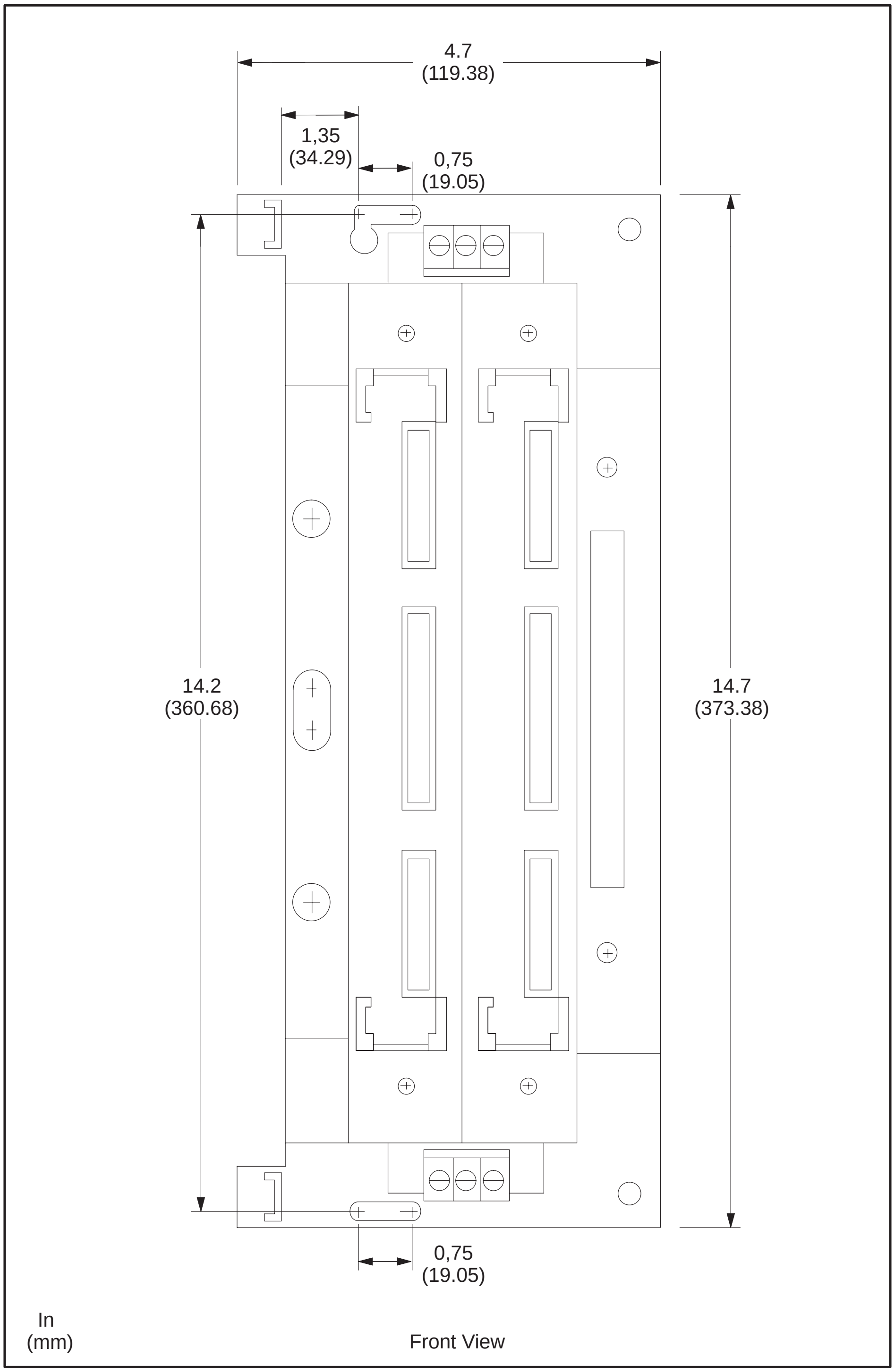


Figure 5-4   PPX:500–5840 Adapter Base

5.4 TI505 I/O

I/O Modules

The TI505 I/O modules are used with the TI560T/TI565T. Table 5-3 lists the I/O modules available and their part number and description.

Table 5-3 TI505 I/O Modules

Part Number	Description	DC Consumption (watts) +5 V	–5 V
PPX:505-4008	24 VAC Input (8 point)	2.0	-
PPX:505-4016	24 VAC Input (16 point)	2.0	-
PPX:505-4032	24 VAC Input (32 point)	2.0	-
PPX:505-4108	LVDC/TTL (8 point)	2.0	-
PPX:505-4116	LVDC/TTL (16 point)	2.0	-
PPX:505-4132	LVDC/TTL (32 point)	2.0	-
PPX:505-4208	110 VAC Input (8 point)	2.0	-
PPX:505-4216	110 VAC Input (16 point)	2.0	-
PPX:505-4232	110 VAC Input (32 point)	2.0	-
PPX:505-4308	24 VDC Input (8 point)	2.0	-
PPX:505-4316	24 VDC Input (16 point)	2.0	-
PPX:505-4332	24 VDC Input (32 point)	2.0	-
PPX:505-4408	220 VAC Input (8 point)	2.0	-
PPX:505-4416	220 VAC Input (16 point)	2.0	-
PPX:505-4432	220 VAC Input (32 point)	2.0	-
PPX:505-4508	24 VDC Output (8 point)	2.5	-
PPX:505-4516	24 VDC Output (16 point)	2.5	-
PPX:505-4532	24 VDC Output (32 point)	2.5	-
PPX:505-4608	110 VAC Output (8 point)	2.5	-
PPX:505-4616	110 VAC Output (16 point)	2.5	-
PPX:505-4632	110 VAC Output (32 point)	2.5	-
PPX:505-4708	24 VDC Output (8 point)	5.0	-
PPX:505-4716	24 VDC Output (16 point)	5.0	-
PPX:505-4732	24 VDC Output (32 point)	5.0	-
PPX:505-4808	220 VAC Output (8 point)	5.0	-
PPX:505-4816	220 VAC Output (16 point)	5.0	-
PPX:505-4832	220 VAC Output (32 point)	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 5-3 TI505 I/O Modules (continued)

Part Number	Description	DC Consumption (watts) +5 V	–5 V
PPX:505-4932	Relay Output Form A (32-point)	2.5	-
PPX:505-4932	Relay Output Form A (32-point)	2.5	-
PPX:505-5100	TurboPlastic™	7.0	0
PPX:505-5103	TurboParison™	7.0	0
PPX:505-5184	MODNIM 〈Modbus NIM)	8.0	0
PPX:505-5190	6MTCC: TI505 to 6MT I/F	4.0	0
PPX:505-5417	16 Output Relay, 115 VDC	3.0	0
PPX:505-6010	Input Simulator	2.0	-
PPX:505-6011	Output Simulator	2.5	-
PPX:505-6108	Analog Input	4.0	-
PPX:505-6108–A	Analog Input	4.0	-
PPX:505-6202	Analog Output (2 point)	2.5	-
PPX:505-6204	Analog Output (4 point)	5.0	-
PPX:505-6208	Analog Output (8 point)	5.0	-
PPX:505-6208–A	Analog Output (8 point)	2.0	-
PPX:505-6308	Word Input	4.0	-
PPX:505-6408	Word Output	5.0	-
PPX:505-6851–A	Remote Base Controller	5.0	0.2
PPX:505–6830	IOCC	5.0	0.1
PPX:505–6840	Distributed Base Controller	5.0	0.1
PPX:505–7002	High Speed Counter & Encoder	2.5	-
PPX:505–7012	8 In/4 Out Analog	3.0	-
PPX:505–7016	Bipolar 8 In/4 Out Analog	6.5	-
PPX:505–7028	Thermocouple	6.5	0.01
PPX:505–7038	RTD	2.2	0
PPX:505–7101	BASIC Module	6.0	0.125
PPX:505-7339	NIM (Network Interface Module)	8.0	-
PPX:505-7340	Dual Media NIM	8.0	-
PPX:505-7354	Peerlink	8.0	-

Table 5-3 TI505 I/O Modules (continued)

Part Number	Description	DC Consumption (watts) +5 V	–5 V
PPX:505–ATM	IBM AT Compatible Coprocessor	11.0	0.2
PPX:505–0220	IBM AT Compatible Coprocessor	11.0	0.2
PPX:505–0440	IBM AT Compatible Coprocessor	11.0	0.2
PPX:505–4120	IBM AT Compatible Coprocessor	11.0	0.2

5.5 TI500 I/O

I/O Modules

The TI500 I/O modules are used with the TI560T/TI565T. Table 5-4 lists the I/O modules available, and gives the part number and description of each module.

Table 5-4 TI500 I/O Modules

Part Number	Description	DC Consumption (watts) +5 V	–5 V
PPX:500-2108	IOCC	5.0	0.1
PPX:500-2109	Distributed Base Controller	5.0	0.1
PPX:500-5002	220 VAC	1.0	-
PPX:500-5003	110 VDC Input	1.0	-
PPX:500-5004	220 VDC Input	1.0	-
PPX:500-5005	24V AC/DC Input	1.0	-
PPX:500-5006	48V AC/DC Input	1.0	-
PPX:500-5007	TTL Input	1.0	-
PPX:500-5008	14 VDC Input	1.0	-
PPX:500-5009	Analog Input	3.0	-
PPX:500-5010	24/48 VAC Output	1.4	-
PPX:500-5011	110 VAC Output	1.4	-
PPX:500-5012	220 VAC Output	1.4	-
PPX:500-5013	24/48 VDC Output	1.4	-
PPX:500-5016	Analog Output	3.0	-
PPX:500-5018	Word Input	2.0	-
PPX:500-5019	Word Output	2.0	-
PPX:500-5020	Input Simulator	1.0	-
PPX:500-5021	Output Simulator	1.4	-
PPX:500-5022	ASCII Module	8.5	0.125
PPX:500-5023	High Speed Pulse Input	6.0	0.050
PPX:500-5024	Servo Axis Controller	7.5	0.150
PPX:500-5026	24 VDC Latching Input Rapid Response Output	2.0	-
PPX:500-5027	110 VAC Latching Input Rapid Response Output	2.0	-
PPX:500-5029	Dual Communication Port	8.5	0.125
PPX:500-5030	32-point 15-30 VDC Input	3.0	-
PPX:500-5031	32-point 20-30 VDC Output	3.0	-
PPX:500-5032	6-point Isolated Input, 85-132 VAC	1.0	-

Table 5-4 TI500 I/O Modules (continued)

Part Number	Description	DC Consumption (watts) +5 V	–5 V
PPX:500-5033	4-point Isolated output, 85-132 VAC	1.4	-
PPX:500-5035	BASIC Module	8.5	0.125
PPX:500-5037−A	8 In, Analog	2	-
PPX:500-5038	Network Interface RS-232 & LLP	8.0	0.625
PPX:500-5039	Network Interface Dual LLP	8.0	1.250
PPX:500-5040	Network Interface Dual RS-232	8.0	0.050
PPX:500-5114	Remote Base Controller	6.0	0.200
PPX:500-5041	110 VAC Redundant Output	2.5	-
PPX:500-5047	8-Channel Analog Output	2.0	-
PPX:500-5047-A	8 Out, Analog	2.0	-
PPX:500-5048	24 VDC Isolated Input	1.0	-
PPX:500-5049	24 VDC Isolated Output	1.4	-
PPX:500-5051	Thermocouple	4.0	-
PPX:500-5052	Resistance Temp. Detector (RTD)	3.5	-
PPX:500-5053	TIWAY Peerlink	6.5	-
PPX:500-5054	Redundant TIWAY PEERLINK	6.9	1.25
PPX:500-5055	32-point, 110 VAC Input	3.0	-
PPX:500-5056	32-point,110 VAC Output	3.0 *	-
PPX:500-5061	High-current Relay	3.0	-
PPX:500-5062	Low-current Relay	3.0	-
PPX:500-5192	48 VDC Input	3.0	-
PPX:500-9912	Redundant Remote Base Controller	6.0	0.200

* Some versions of the PPX:500-5056 consume 3.3 watts of +5 VDC power and are identified with a label.

5.6 I/O Installation Considerations

CATV cable is the medium used to transmit I/O updates. The CATV cable, taps and fittings are user-supplied; consult the manufacturer for detailed specifications.

The TI560T or TI565T I/O system allows remote I/O bases to be located as much as 15,000 ft. away from the main system processor. The bases are distributed in a multidrop configuration, and there should be less than 55 dB signal loss @25°C between the RCC and any I/O base. RS–485 media allows communication up to 3,300 ft. depending on the number of taps.

Siemens recommends no less than quad-shield RG-6 for all drop lines. The CATV should not have any other communications traffic on it.

5.7 Specifications for TI560T/TI565T

Table 5-5 Environmental Specifications

Operating temperature	0 to 60°C (32 to 140°F)
Storage temperature	–40 to +85°C (–40 to 185°F)
Relative humidity	5% to 95% noncondensing
Vibration	Random profile based on NAVMAT P–9492
Noise immunity, conducted	MIL STD 461A CS01, CS02 and CS06 IEEE472 Transient Surge NEMA ICS 2–230.45 Showering Arc
Noise immunity, radiated	MIL STD 461A RS01, RS02 and RS03 FCC Docket 20780 (Class A)
Agency Approvals	UL Listed CSA Certified FM Approved (Class I, Div. 2, Haz. Loc.)
Corrosion Protection	All parts of corrosion-resistant material or plated or painted as corrosion protection

Table 5-6 Electrical Specifications for Power Supply

Input voltage selection	110 VAC/220 VAC, or 24 VDC
Voltage Range	85 to 132 VAC (110 VAC) 170 to 260 VAC (220 VAC) 20.4 to 30 VDC (24 VDC)
Maximum Power	300 VA
Frequency Range	47 to 63 Hz
Fuse	5 A/250 VAC
Output	5.1 V at 23 A for 110 VAC/220 VAC, 30 A for 24 VDC 12.0 V at 1 A -12.0 V at 1 A
Battery Shelf Life	3 years typical storage –40°C to +80°C
Battery Type	Lithium 1/2 gram (non-rechargeable)
Battery	6 months typical memory backup 0°C to 60°C
Backup Features	Memory Backup Displays: POWER GOOD and BATTERY GOOD LEDs Memory Protect Keylock
Agency Approvals	UL Listed CSA Certified FM Approved (Class I, Div. 2, Haz. Loc.)

Appendix A
Performance

A.1 TI560T Performance

Scan Times

Scan times for the main CPU can be estimated using the following approximations.

Main CPU	Scan Times
Central Processing Unit Overhead	11 ms
I/O Update Cycle	time of channel with most bases (3.0 ms for first and 1.25−1.5 ms for each subsequent base) + transmission time for word modules. (16 µs per word)
Ladder Logic Execution	1.5 ms/K words of ladder logic Boolean elements + time for box instructions as shown in the following tables
Special Function I/O Modules	varies from 8−12 ms per module (4−25 ms for Peerlink dependent on amount of data)
SF CPU Comm	nominally 1−4 ms; depends on number of loops, SFs, and analog alarms)
Port Comm	6 ms maximum if both local and two remote ports are active
Hot Backup Comm	9 ms maximum (on-line mode only)

NOTE: If feature is not present, no time will be added to the scan. These are worst-case figures.

Fixing scan time to a value longer than needed only allows more processing time for the Main CPU communication ports. Other functions consume only the amount of time required to execute. If the scan time is fixed at less than required, the controller will use the amount of time required to perform all tasks in the timeline and set the “scan overrun” flag. Nothing is left out of a scan due to a fixed scan time.

A.2 Ladder Logic Execution Times

Instruction Set

Instruction	Mnemonic	Execution Time μ secs †	Words of Ladder Memory *	Reference No.		Comments
DISCRETE INSTRUCTIONS:						
Contacts	X, Y, C	1.5	1–2	1–8192		Inputs and Outputs should not have the same #.
Outputs	Y	1.5	1–2	1–8192		
Control Relays	C	1.5	1–2	1–8192		
CONDITIONAL OPERATIONS:						
Jump	JMP	8	1	1–8		
End Jump, Conditional	JMP(E)	19	2	1–8		
Master Control Relay	MCR	9	1	1–8		
End Master Control Relay, Cond.	MCR(E)	19	2	1–8		
End, Conditional	END(C)	0	1			
Skip Forward to LBL	SKP	4	1	1–255		
Label	LBL	0	1	1–255		
Scan Sync Inhibit	SSI	10	1			
UNCONDITIONAL OPERATIONS:						
End Jump	JMP(E)	19	1			
End Master Control Relay	MCR(E)	19	1			
End, Unconditional	END	0	1			
BIT INSTRUCTIONS:						
Bit Clear	BITC	19	3–5	0–32767	***	
Bit Pick	BITP	17	3–5	0–32767	***	
Bit Set	BITS	19	3–5	0–32767	***	
Bit Shift Register	SHRB	15(4.7)	3–5	**		No two SHRB or SHRW should have the same #.
COUNTER/TIMER INSTRUCTIONS:						
Counter	CTR	28	2–3	**		Counters, Timers, and Control Alarms should not be assigned the same #.
Timer (bases of a and 100 msec)	TMR	23	2–3	**		
Up/Down Counter	UDC	56	3–5	**		
Discrete Control Alarm Timer	DCAT	49	6–11	**		
Motor Control Alarm Timer	MCAT	83	9–15	**		
DRUM INSTRUCTIONS:						
Drum	DRM	191	50–66	**		Drums and Event Drums should not be assigned the same #.
Event Drum	EDRM	218	66–98	**		
Maskable Event Drum – Discrete	MDRMD	251	68–100	1–1152		
Maskable Event Drum – Word	MDRMW	108	54–72	1–1152		
MATRIX INSTRUCTIONS:						
Scan Matrix Compare	SMC1	142(8.9)	34–51	0–32767	***	
Index Matrix Compare	IMC	135	33–50	0–32767	***	

† Figures in parentheses are execution times for each additional word after the first.

* Two numbers in this column indicate the minimum and maximum number of words required in L memory for for these instructions.

** The maximum reference number for these instructions is dependent upon memory size and allocation.

*** Instructions requiring numbers must start with 1; where numbering is optional, 0 may be used. (Programming devices may restrict entry of 0). On the instructions with optional numbering, the number entered is for identification only.

Ladder Logic Execution Times (continued)

Instruction	Mnemonic	Execution Time µsecs †	Words of Ladder Memory *	Reference No.		Comments
MATH INSTRUCTIONS:						
Add	ADD	22	4–8	0–32767	***	
Subtract	SUB	22	4–8	0–32767	***	
Multiply	MULT	24	4–8	0–32767	***	
Divide	DIV	20	4–8	0–32767	***	
Compare	CMP	32	5–10	0–32767	***	
Square Root	SQRT	190	3–6	0–32767	***	
MOVE INSTRUCTIONS:						
Load Data Constant	LDC	14	3–5	0–32767	***	
Move Discrete IR to Word	MIRW1	29(3.8)	4–7	0–32767	***	
Move Word to Discrete IR	MWIR	30(4.9)	4–7	0–32767	***	
Move Word (memory)	MOVW1	18(3.3)	4–7	0–32767	***	
Move Word from Table	MWFT	39	5–9	**		MWFT and MWTT should not be assigned the same #.
Move Word to Table	MWTT	43	5–9	**		
WORD INSTRUCTIONS:						
Convert Binary to BCD	CBD	44	3–6	0–32767	***	
Convert BCD to Binary	CDB1	39(11)	4–7	0–32767	***	
Word AND	WAND	22	4–8	0–32767	***	
Word OR	WOR	22	4–8	0–32767	***	
Word Exclusive OR	WXOR	22	4–8	0–32767	***	
Word Rotate Right	WROT1	22	3–5	0–32767	***	
Word Shift Register	SHRW	28(5.6)	4–7	**		No two SHRW and SHRB should be given the same #.
One Shot	O/S	15	2	**		
SUBROUTINE INSTRUCTIONS:						
Go to Subroutine	GTS	17	2	1–255		
Subroutine	SBR	0	2	1–255		
Return (Cond. or Uncond.)	RTN	5	2			
TABLE INSTRUCTIONS:						
Table to Table AND	TAND1	31(5)	6–9	0–32767	***	
Table to Table OR	TOR	31(5)	6–9	0–32767	***	
Table to Table EXCLUSIVE OR	TXOR	30	6–9	0–32767	***	
Table Complement	TCPL1	29(4.2)	5–7	0–32767	***	
Word to Table	WTOT1	21	6–9	0–32767	***	
Table to Word	TTOW1	21	6–9	0–32767	***	
Word to Table AND	WTTA1	25	7–11	0–32767	***	
Word to Table OR	WTTO1	25	7–11	0–32767	***	
Word to Table EXCLUSIVE OR	WTTXO1	25	7–11	0–32767	***	
Table Search for Equal	STFE1	23(0)	6–9	0–32767	***	
Table Search for Not Equal	STFN1	27(0)	7–11	0–32767	***	
CLOCK INSTRUCTIONS:						
Date Compare	DCMP	67	3–4	0–32767	***	
Time Compare	TCMP	87	5–8	0–32767	***	
Time Set	TSET		3–4	0–32767	***	
Date Set	DSET	21	3–4	0–32767	***	

† Figures in parentheses are execution times for each additional word after the first.

* Two numbers in this column indicate the minimum and maximum number of words required in L memory for for these instructions.

** The maximum reference number for these instructions is dependent upon memory size and allocation.

*** Instructions requiring numbers must start with 1; where numbering is optional, 0 may be used. (Programming devices may restrict entry of 0). On the instructions with optional numbering, the number entered is for identification only.

A.3 TI565T Performance

The TI565T can update all 64 menu driven loops in 2 seconds, and 32 loops in 1 second with a typical floating point load. The TI565T can perform 32 loop calculations, update 16 analog alarms, and execute additional floating point calculations as called from SFs all in one second. This assumes the scan time is equal to, or greater than, 50 ms to allow the TI565T to complete tasks without having to process interrupts from the TI560T.

With *no* floating point load (no analog alarms, no special functions programs) and a ladder logic scan time over 50 in S, the TI565T can process and update as many as 99 loops per second. The minimum time to update a single loop, with processor overhead, is 100 milliseconds.

The TI565T Floating Point Math Package is IEEE format.

The TI565T can also recognize subscripted variables as addresses for indirect addressing. Variables may also be used as subscripts for repetitive calculations.

A.4 RCC Performance

I/O Update

The I/O update time is dependent on the number of bases and the amount of analog I/O. The following numbers show representative variance between discrete and analog updates, assuming that 10% of the modules are high density.

I/O Updates	
All 8K discrete	10.0 ms
Mix 87.5% discrete/12.5% Analog	12.0 ms
All 8K analog	28.0 ms

To calculate typical update times, allow 3 ms for the first base and 1.25 ms for additional bases. Add to this 16 μs for each word I/O point.

The I/O update time can be decreased by the addition of RCC cards or by arranging the analog to discrete I/O mix to reflect a minimum amount of analog I/O per channel of each RCC. Since all the RCCs are updated in parallel, the I/O update time is most dependent on the RCC channel with the largest number of bases or the heaviest percentage of analog I/O per channel.

As an example, if one RCC is installed with all the analog I/O on one channel and all the discrete I/O on the other channel, the I/O update time will be 28 milliseconds (which is the maximum time for updating one channel with all analog I/O).

Better performance can be achieved by placing analog modules on the lowest numbered bases. With one broadcast message at the beginning of the I/O cycle, the RCC instructs all Remote Base Controllers to prepare inputs. It then polls each base one at a time for these inputs. Since the highest numbered bases (base 14 or 15) are polled for information and updated first, the lower numbered bases (base 0 or 1) have more time to prepare information for transmission. Because the Remote Base Controller takes longer to prepare and transmit analog information, placing analog modules in the lower numbered bases, which are updated last, allows the RBC more time to prepare the data for transmission. If analogs are placed on the higher numbered bases the preparation time may delay updates on the lower numbered bases.

A.5 Hot Backup Performance

As long as the standby unit is in the off-line mode, there will be no effect on performance of the active unit. However, when the standby unit is in the on-line mode, the scan time will be approximately 9 ms longer. With the TI565T, further scan variation results from interaction between the Main CPU and the SF CPU. Since PACK SFs must complete in the scan on which they are initialized, the scan can be extended considerably. Use of long PACK statements should be avoided if this is of concern for your process. The same is true for changes to PV range on loops with many RAMP/SOAK steps.

Switchover times vary depending on the type of fatal error encountered and the time lapse before the fatal error is logged and system shutdown occurs (some errors take longer to be recognized than others). Typical switchover times after a system fatal error has been detected can range from 130 ms to 200 ms. The best case is 130 ms required for the standby to verify that the active unit is no longer communicating over the I/O channel (possible if the active unit is good but the HBU fiber optic link is lost). A watchdog timer fatal error would take the most time for the system to detect. The watchdog timer times out after 500 ms have elapsed. So, worst case switchover time for this type error would be 500 ms for error detection and 130 ms for switchover, or 630 ms total.

Appendix B
Loop Operation

B.1 TI565T Loop Algorithms

Overview

This appendix describes the implementation of PID control loops as done at Texas Instruments in the TI565T controller. The discussion includes algorithms used, options that may be selected, and performance.

This introductory section gives the basics of PID control as implemented by the TI565T. A detailed description of the actual algorithms used is contained in the next sections.

Analog Variables and Scaling

The Series 500 and Series 505 have 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 wide words, the interpretation of the bits being dependent on the type of I/O module. Of primary interest to our discussion are analog I/O points. 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

In TI565T loops, 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 TI565T allows you to specify the location of the variable and the manner in which it is scaled.

When the TI565T 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 TI565T 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 internal to the TI565T for the purpose of loop calculations.

PID Control

The TI565T 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

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

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

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 TI565T 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 = *Sample rate*

K_c = *Proportional gain*

K_i = *Coefficient of the integral term*

= $K_c(T_s/T_i)$

K_r = *Coefficient of the integral term*

= $K_c(T_d/T_s)$

T_i = *Reset or integral time*

T_d = *Derivative time or rate*

SP = *Setpoint*

PV_n = *Process Variable at nth sample*

e_n = *Error at nth sample given by*

$e_n = SP - PV_n$

M_0 = *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 TI565T 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 TI565T 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 TI565T also combines the integral sum and the initial output into a single term called the bias (denoted in this document by M_X). 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 TI565T 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 M_x is performed exactly as shown in the equation below.

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

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 TI565T 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 TI565T 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 adjusting the bias term.

Freeze Bias When Output Goes Out of Range

If the "Freeze Bias When Output Goes Out of Range" option is selected, the TI565T simply stops changing the bias Mx whenever the 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
$$M_n = 0.0 \quad \text{if} \quad \overline{M} < 0.0 \, ,$$
$$M_n = \overline{M} \quad \text{if} \quad 0.0 \leq \overline{M} \leq 1.0 \, ,$$
$$M_n = 1.0 \quad \text{if} \quad \overline{M} > 1.0 \, .$$

New bias term
$$Mx_n = \overline{Mx}_n \quad \text{if} \quad 0.0 \leq \overline{M} \leq 1.0 \, ,$$
$$M_n = Mx_{n-1} \quad \text{otherwise} \, .$$

Thus in our example, the bias will probably not go all the way to zero so that, 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 TI565T 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 $$M_n = 0.0 \quad \text{if} \quad \overline{M} < 0.0$$

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

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

New bias term $$Mx_n = \overline{Mx} \quad \text{if} \quad 0.0 \leq \overline{M} \leq 1.0 \quad \textit{no windup)}$$

$$Mx_n = 1 - (K_c e_u - K_r(PV_u - PV_{u-1})) \quad \text{if output} \quad M_n \geq 1.0$$

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 we 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, then 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 slugglishness 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 *or* by entering a value of zero (0.0). The controller tests for the special case of zero integral time and simply turns off the integral calculations in that case.

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 TI565T makes the computation of these values conditional on the value of *(K_c)* as follows:

$$
\begin{aligned}
K_i &= K_c(T_s/T_i) \quad &\text{if} \quad K_c \neq 0\ , \\
&= T_s/T_i \quad &\text{if} \quad K_c = 0\ , \quad &\textit{(for I or for ID control)} \\
K_r &= K_c(T_d/T_s) \quad &\text{if} \quad K_c \neq 0\ , \\
&= T_d/T_i \quad &\text{if} \quad K_c = 0\ , \quad &\textit{(for ID or D control)}
\end{aligned}
$$

Velocity PID Equation

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” from the equation at time “n–1”.

The velocity equation is given by

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

$$\Delta M_n = K_c(e_n - e_{n-1}) + K_i e_n - K_r(PV_n - 2PV_{n-1} + PV_{n-2})$$

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.

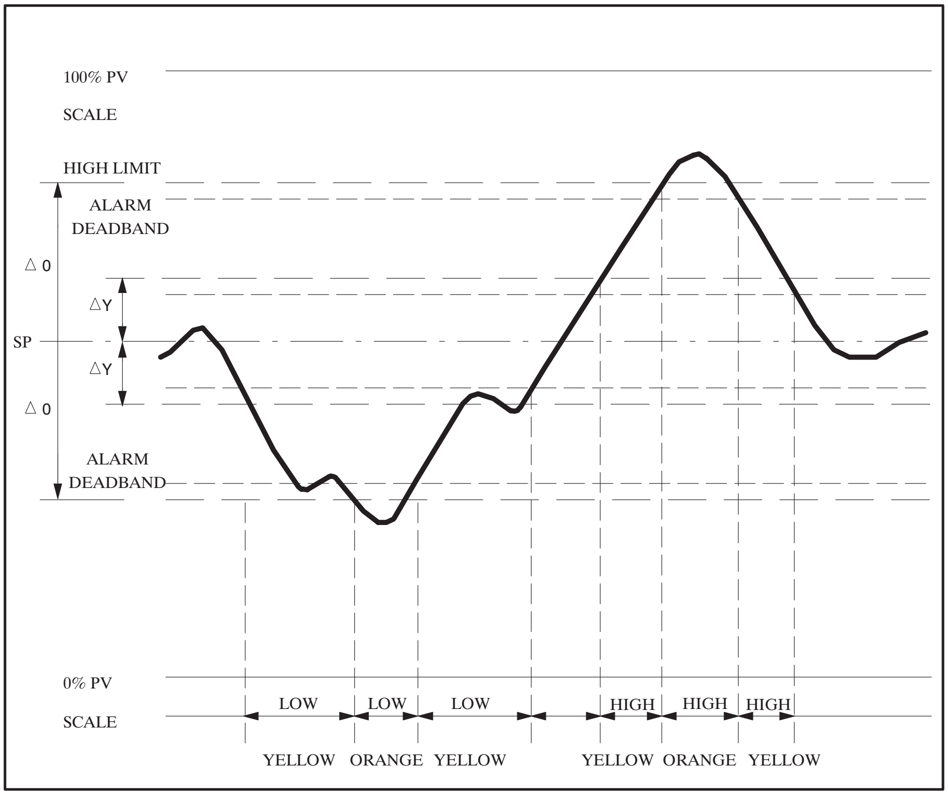
B.2 Loop Alarms

The TI565T 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.

Loop Alarms (continued)

The TI565T 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 the following diagram.



All alarm conditions are available to TI565T SF programs, to ladder logic, and to operator interface devices through task codes.

The TI565T has three modes in which loops operate. (Actually there is a fourth state—loop is not operating—which the loop is in when the TI565T is in PROGRAM mode.) These modes are descri ed below.

Manual

In this mode, the loop output is not calculated by the TI565T, but comes from the operator. While a loop is in MANUAL, the TI565T will still monitor the Broken Transmitter, High-High, High, Low, Low-Low, and Rate-of-Change alarms. The Yellow and Orange deviation alarms are not monitored.

Auto

In this mode, the TI565T computes the loop output. The setpoint (SP) for the loop comes from either an operator, SF Program, or from a RAMP/SOAK Table. RAMP/SOAK is discussed later in this document. All alarms are monitored.

Cascade

In this mode, the TI565T 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 TI565T also allows the remote setpoint to be some other variable in the TI565T. (Such loops are not truly cascaded but the same terminology is used. The TI565T 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. It is possible to cascade loops more than two levels deep.

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 TI565T. Briefly, this is done 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 TI565T allows the loop mode to be changed by an SF 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 TI565T 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 TI565T provides for bumpless mode changes (this means that the controller output does not change immediately after the mode change). This is 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.3 Special Computations on Output, PV, or Error

Forward and Reverse Acting Loop

The TI565T 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 TI565T 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 Set Point. A TI565T 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 Set Point is increased.

Square Root of the Process Variable

The TI565T 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:

$$e_n = (SP - PV_n)ABS(SP - PV_n)$$

Since $e_n =< \overline{e}$, 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:

$$\bar{e} = SP - PV_n$$

$$
\begin{aligned}
e_n &= 0 && \text{if} \quad abs(\bar{e}) < YDEV \\
&= \bar{e} - YDEV && \text{if} \quad \bar{e} > YDEV \\
&= \bar{e} + YDEV && \text{if} \quad \bar{e} < -YDEV
\end{aligned}
$$

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. The applications are prevented from using the 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 TI565T 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, the TI565T loop calculation is modified as follows:

Position PID 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 PID 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, 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 probably good for almost all applications requiring derivative control.

Ramp/Soak

For each loop, the TI565T allows the user 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, the user may 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.

SF Programming

At various points in the execution of a loop, the TI565T will allow the user to break out of the normal loop execution sequence to perform his own processing. The mechanism provided is through SF Programs. An SF Program is a sequence of BASIC-like statements that allow the user to manipulate TI560T or TI565T variables, including all of the parameters to a loop.

The points in the execution of a loop in which an SF Program may be invoked

1. After reading and performing any indicated conditioning on the Process Variable. This type of SF Program would typically be used to perform filtering on the Process Variable. (SF on PV).

2. Before using the Setpoint to compute the error. This type of SF Program would typically be used in a ratio-control application. (SF on SP).

3. Before storing the output. (SF on output).

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 TI560T/TI565T Technical Product Description **Edition:** Third
Manual Assembly Number: 2597773–0033 **Date:** 11/93
Order Number: PPX:560/565–8107–3

FOLD

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 3519
SIEMENS INDUSTRIAL AUTOMATION INC.
3000 BILL GARLAND RD
P O BOX 1255
JOHNSON CITY TN 37605–1255

FOLD