Logipam

SFT2885 Programming Software for Sepam Series 80 User's Manual

01/2013

SEPED303004EN

www.schneider-electric.com

Schneider Electric

The information provided in this documentation contains general descriptions and/or technical characteristics of the performance of the products contained herein. This documentation is not intended as a substitute for and is not to be used for determining suitability or reliability of these products for specific user applications. It is the duty of any such user or integrator to perform the appropriate and complete risk analysis, evaluation and testing of the products with respect to the relevant specific application or use thereof. Neither Schneider Electric nor any of its affiliates or subsidiaries shall be responsible or liable for misuse of the information that is contained herein. If you have any suggestions for improvements or amendments or have found errors in this publication, please notify us.

All pertinent state, regional, and local safety regulations must be observed when installing and using this product. For reasons of safety and to help ensure compliance with documented system data, only the manufacturer should perform repairs to components.

When devices are used for applications with technical safety requirements, the relevant instructions must be followed.

Failure to use Schneider Electric software or approved software with our hardware products may result in injury, harm, or improper operating results.

Failure to observe this information can result in injury or equipment damage.

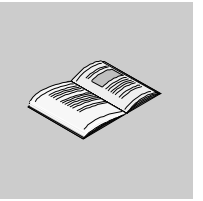
Document Set

Structure of this Manual

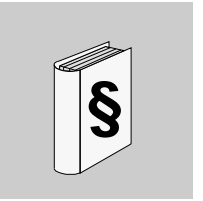
This manual consists of 7 chapters and 2 appendices:

- **Chapter 1: Presentation of the Logipam Software**
  This chapter describes the general principles of the Logipam software. It contains a general description of the Logipam software and an introduction to the method of creating and running control system programs.
- **Chapter 2: Description of Logipam Variables and Objects**
  This chapter describes the variables and objects used by the language, their states, their function and their use.
- **Chapter 3: Ladder Language**
  This chapter describes the Ladder language used by Logipam and the execution mode for a control system program written in this language.
- **Chapter 4: Familiarization With the Logipam Software**
  This chapter describes the Logipam software interface and the options for customizing the working environment.
- **Chapter 5: Creating a Program**
  This chapter introduces the specific edit mode functions. It describes the procedures for creating a control system program: the program structure, configuration of the variables and objects, printing the programming file and the consistency check.
- **Chapter 6: Simulating a Program**
  This chapter introduces the specific simulation mode functions. It describes the procedures for simulating program execution.
- **Chapter 7: Operating a Program**
  This chapter describes the steps involved in operating the Logipam control system program and maintaining it on Sepam.
- **Appendices**:
  List of Predefined Variables.
  List of Logipam Keyboard Shortcuts and Function Keys.

Table of Contents



Safety Information



Important Information

NOTICE

Read these instructions carefully, and look at the equipment to become familiar with the device before trying to install, operate, or maintain it. The following special messages may appear throughout this documentation or on the equipment to warn of potential hazards or to call attention to information that clarifies or simplifies a procedure.

The addition of this symbol to a Danger safety label indicates that an electrical hazard exists, which will result in personal injury if the instructions are not followed.

This is the safety alert symbol. It is used to alert you to potential personal injury hazards. Obey all safety messages that follow this symbol to avoid possible injury or death.

⚠ DANGER

DANGER indicates an imminently hazardous situation which, if not avoided, **will result in** death or serious injury.

⚠ WARNING

WARNING indicates a potentially hazardous situation which, if not avoided, **can result in** death or serious injury.

⚠ CAUTION

CAUTION indicates a potentially hazardous situation which, if not avoided, **can result in** minor or moderate injury.

NOTICE

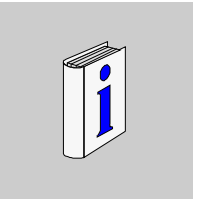
NOTICE is used to address practices not related to physical injury.

PLEASE NOTE

Electrical equipment should be installed, operated, serviced, and maintained only by qualified personnel. No responsibility is assumed by Schneider Electric for any consequences arising out of the use of this material.

A qualified person is one who has skills and knowledge related to the construction and operation of electrical equipment and its installation, and has received safety training to recognize and avoid the hazards involved.

About the Book



At a Glance

Document Scope

This manual describes how to use Logipam in MS Windows to create control system programs for Sepam series 80 protection relays.

It explains how the software works, its programming language, programming elements, creation and simulation of programs and also program operation.

Who is This Manual Aimed at?

This document is aimed at design office personnel who wish to create a control system program, and at anyone involved in setting up control system programs and maintaining them on Sepam series 80 protection relays.

Knowledge Required

The creation and setup of a Logipam control system program requires knowledge of the MS Windows environment and applied control system projects.

Validity Note

Conditions of Use for the Logipam Software

- Logipam can only be used to develop control system programs for Sepam series 80.
- A cartridge with the Logipam SFT080 option must be used on the Sepam when executing control system programs developed with Logipam.
- Functions programmed using another editor are not compatible with Logipam programmed functions.

Environment Supported

Logipam runs on PCs with the following operating systems:
- Windows 2000
- Windows XP

Only Sepam series 80 protection relays with a cartridge including the Logipam SFT080 option are capable of executing the control and monitoring functions with the Logipam SFT2885 software.

Related Documents

Title of Documentation	Reference Number
Sepam Series 80 - Metering, Protection, Control and Monitoring User's Manual (IEC)	SEPED303001FR SEPED303001EN
Sepam Series 80 - Modbus Communication User's Manual (IEC)	SEPED303002FR SEPED303002EN
Sepam Series 80 - Operation Manual (IEC)	SEPED303003FR SEPED303003EN
Sepam Series 80 Digital Relay - Reference Guide (ANSI)	63230-216-230
Sepam Series 80 Digital Relay - Modbus Communication Guide (ANSI)	63230-216-231
Sepam Series 80 Digital Relay - Installation Guide (ANSI)	63230-216-229

You can download these technical publications and other technical information from our website at www.schneider-electric.com.

User Comments

We welcome your comments about this document. You can reach us by e-mail at techcomm@schneider-electric.com.

Introduction to the Logipam Software

1

Scope of This Chapter

This chapter describes the general principles of the Logipam software.

What Is in This Chapter?

This chapter contains the following topics:

Introduction

General

Logipam (SFT2885 software) is used to program control system functions for Sepam series 80.

Logipam runs in MS Windows.

Programming Language

The programs are created in Ladder language.

Ladder language is a graphic language used to transcribe ladder diagrams. It uses the basic graphic symbols of a diagram: contacts, coils and connections.

When Should Logipam Be Used?

Logipam is used to create specific functions or to complement Sepam predefined functions.

NOTE: Each Sepam contains the appropriate predefined functions for the chosen application.

Composition of the Logipam Software

Logipam consists of:

- A program editor in Ladder language
- A simulator for checking that the program is working correctly
- A code generator for executing the Sepam program

Registering the Logipam Software

Introduction

On installation, you are invited to register your software. In order to register, you will need a license number.

If you do not register, the program will run in demonstration mode, for a 30-day trial period.

Demonstration Mode

In demonstration mode, all program functions are available without any restriction on their use. Each time the software is launched, you are invited to register.

At the end of the 30-day trial period, the software will no longer run and you must register.

Registration Methods

During registration, a wizard offers you the following registration methods:

- Requesting a license number
- Transferring license rights
- Entering a license number (code obtained by fax or email)

Requesting a License Number

To request a license number, use one of the following 5 methods:

- Request a license number via the Internet from the PC it is installed on
- Request a license number via the Internet from another PC
- Request a license number by telephone using the number given by the installation wizard
- Request a license number by email, using the address given by the installation wizard
- Request a license number by fax, sent to the number given by the installation wizard

Transferring the License Rights

The rights for an existing license can be transferred via:

- A USB memory stick
- Another connected computer

It is possible for the license rights to be transferred to another user from a different company.

Architecture

Introduction

Logipam is used to create, simulate and generate control system programs. The programs are then set up and operated with the aid of SFT2841 and Sepam.

Hardware and Software Architecture

The hardware and software architecture requires the creation of a control system program that consists of:

- The Logipam SFT2885 software installed on a PC running MS Windows
- The SFT2841 parameter-setting and operating software installed on a PC running MS Windows
- A Sepam series 80 that includes the Logipam SFT080 option, connected to the PC

Method of Creating and Operating a Logipam Program

Introduction

For a control system program to be implemented in the optimum conditions, it is important to:

- Adhere to the preliminary consultancy project
- Follow the creation and setup principles described in the paragraphs below.

Method of Creating and Operating a Logipam Program

The table below lists the steps involved in creating and operating a Logipam program:

Step	Description
1	Preliminary study of the control system project
2	Creation of the program using SFT2885
3	Setup and operation of the program: ● **Using SFT2841** To configure Sepam and the Logipam program, to set the program parameters and monitor program execution. ● **On Sepam** To set the program parameters and monitor program execution.

Study of the Control System Project

A study phase should be conducted before a control system program is created. This is divided into two parts:

- Study of the protection plan
- Definition of the control and monitoring functions required, making it possible to determine:
  - the predefined functions in Sepam to be used
  - whether the Logipam software should be used to adapt the predefined functions or to complement them with special functions
  - the list of I/O required

Creating the Logipam Program

The operations for creating a control system program are as follows:

- Creation of the control system program
- Program entry: Consists of entering the ladder diagram and configuring the variables and objects used.

- Program simulation: Consists of checking program entry and testing program operation before it is exported.
- Export: Consists of generating a `.bin` file for exporting the program to SFT2841. This file is automatically placed in a directory common to both SFT2841 and Logipam.

Setting up and Operating the Program

The created program is set up and operated as follows:

- With SFT2841:
  - in disconnected mode to configure Sepam with the Logipam program
  - in connected mode to configure Sepam or to monitor execution of the Logipam program

  The parameters of the objects used in the program and the configuration bits can be set in both modes.
- From the Sepam interface during operation, to monitor program execution

Description of Logipam Variables and Objects

2

Scope of This Chapter

This chapter describes the variables and objects used by the Logipam software.

What Is in This Chapter?

This chapter contains the following topics:

Introduction

General

Ladder language consists of Boolean variables and Logipam objects.

The three types of Boolean variable are:
- Logipam inputs
- Logipam internal variables
- Logipam outputs

The 3 types of Logipam object are:
- Timers
- Counters
- Clocks

Logipam Inputs

The 4 types of Logipam input variable are:
- Logic inputs
- Remote controls
- Predefined inputs
- Protection outputs

Logipam Internal Variables

The 3 types of Logipam internal variable are:
- Memorized bits
- Non-memorized bits
- Configuration bits

Logipam Outputs

The 5 types of Logipam output variable are:
- Logic outputs
- Remote indications
- Outputs to matrix
- Predefined outputs
- Protection inputs

Logipam Objects

Objects are preprogrammed control system functions that have specific variables, control instructions and parameters.

Identifying Variables and Objects

General

All Logipam variables and objects are identified by:

- An invariable identifier
- A name
- A comment

Names and Comments

Names and comments can be one of two types:

- Predefined
  Predefined variables (predefined inputs and outputs as well as protection inputs and outputs) have a predefined invariable name and comment.
- Free entry
  Each unpredefined variable or each object can be identified by a name and a comment in free text, to make the program easier to understand. It is not compulsory to enter a name or comment. It is entered in the variable or object parameter-setting window. For further information on inserting comments, see *Defining Variables and Objects, page 87*.

Using the Identifier and the Name

All variables and all objects can be represented either by their identifier, or by their name, in all Logipam screens.

Viewing Comments

A variable or object comment can be viewed:

- On the **Parameters** and **Variables** screens
- From the **Diagram** screen, it is displayed:
  - As an info balloon when the variable or object is placed on the diagram entry grid
  - As an info balloon in the variable or object selection window

Logic Inputs

Definition

"Logic input" variables are Logipam input variables; they are divided into two categories:

- "Logic images" of the electrical state of the Sepam MES120 module inputs (IXYY)
- GOOSE (GXYY) logic inputs used by the protocol IEC 61850

They can only be used in the program contact zone.

The logic inputs are updated at the start of each program execution cycle.

Identification

The logic inputs are identified by:

- An identifier:
  - IXYY for the MES120 module logic inputs, for example I105
  - GXYY for the GOOSE logic inputs, for example G403
- A name
- A comment

The name and the comment can be modified.

Logic inputs identifier IXYY

The logic inputs identifier IXYY for the MES120 modules depends:

- On the physical position of the MES120 module mounted on the Sepam base unit
- On the module input number

It is defined as follows:

- X = 1 for the 14 inputs of the MES120 module mounted in H1
- X = 2 for the 14 inputs of the MES120 module mounted in H2
- X = 3 for the 14 inputs of the MES120 module mounted in H3
- YY = 01 to 14 for the MES120 module input number

GOOSE logic inputs identifier GXYY

The GOOSE logic inputs are divided between the logic input “virtual” modules: the GSE modules.

The GXYY identifier is defined as follows:

- X = 4 for the 16 GOOSE inputs in the first virtual GSE module
- X = 5 for the 16 GOOSE inputs in the second virtual GSE module
- YY = 01 to 16 for the GOOSE input number

Use

A logic input that is not present or is declared "unused" in SFT2841 is always at 0.

The logic inputs used only in Logipam should be assigned as "Other use" in SFT2841.

If a name has been allocated to the input in Logipam, this name appears in the input assignment table in SFT2841, instead of "Other use".

Remote Controls

Definition

"Remote control" variables are Logipam input variables. These are logic orders sent by a remote system (a supervisor, for example) to Sepam by the communication system.

The 96 available remote controls are divided into two categories:

- 64 pulse-type remote controls
- 32 maintained remote controls

They can only be used in the program contact zone.

Identification

Remote controls are identified by:

- An identifier
  - TC01 to TC64 for pulse-type remote controls
  - TCM01 to TCM32 for maintained remote controls
- A name
- A comment

The name and the comment can be modified.

Using Pulse-Type Remote Controls

Pulse-type remote controls are used:

- By predefined control and monitoring functions: each remote control is preassigned to a defined logic instruction that is taken into account in processing a predefined function.
- Or by Logipam: any logic order can be assigned to each remote control

Once a remote control is used by Logipam, it is no longer taken into account in the processing of the predefined function to which it has been assigned.

Using Maintained Remote Controls

Maintained remote controls are only used by Logipam and any logic order can be assigned to each remote control.

They are reset to 0 in the event of a power failure.

Predefined Inputs

Definition

"Predefined input" variables are Logipam input variables. These derive from the Sepam predefined functions.

They can only be used in the program contact zone.

Identification

The predefined inputs are identified by:
- An identifier, SIxxx
- A name
- A comment

The name and the comment are predefined and cannot be modified.

The exhaustive list of predefined inputs is provided in the appendix and gives a complete identification of each predefined input (identifier, name and comment) together with the predefined function from which it derives.

For example:
- Identifier: SI087
- Name: V_OPG_ON
- Comment: Disturbance recording on
- Derives from the function: Disturbance recording

Use

Predefined functions are available in each Sepam depending on the chosen application: not all the predefined functions are processed by all Sepams.

If a predefined input is used in the Logipam program even though the associated function is not processed by Sepam, the value of this input is always 0.

Protection Outputs

Definition

"Protection output" variables are Logipam input variables.

They can only be used in the program contact zone.

Identification

Protection outputs are identified by:

- An identifier PXX_Y_ZZ, where:
  - XX is the ANSI protection code
  - Y is the number of the protection unit
  - ZZ is the variable sequence number, between 01 and 99
- A name, identical to the identifier
- A comment

The name and the comment are predefined and cannot be modified.

The exhaustive list of protection outputs is provided in the appendix.

For example:

- Identifier: P50/51_2_7
- Name: P50/51_2_7
- Comment: Protection 50/51 Unit 2 Phase 1 fault.

Use

Protection functions are available in each Sepam depending on the chosen application: not all the protection functions are processed by all Sepams.

If a protection output is used in the Logipam program although the associated function is not processed by Sepam, the value of this output is always 0.

Memorized Bits

Definition

The 128 "Memorized bit" variables are Logipam internal variables. The state of a memorized bit is saved on loss of the Sepam auxiliary power supply.

They can be used in both the program contact zone and coil zone.

Identification

Memorized bits are identified by:
- An identifier, MS001 to MS128
- A name
- A comment

The name and the comment can be modified.

Use

Memorized bits can be viewed during operation with the aid of SFT2841.

Non-memorized Bits

Definition

The 512 "Non-memorized bit" variables are Logipam internal variables. The state of a non-memorized bit is not saved on loss of the Sepam auxiliary power supply and is 0 on each energization.

They can be used in both the program contact zone and coil zone.

Identification

Non-memorized bits are identified by:
- An identifier, MX001 to MX512
- A name
- A comment

The name and the comment can be modified.

Use

Non-memorized bits can be viewed during operation with the aid of SFT2841.

Configuration Bits

Definition

The 64 "configuration bit" variables are Logipam internal variables. The state of a configuration bit is saved on loss of the Sepam auxiliary power supply.

They can be used in both the program contact zone and coil zone.

Identification

Configuration bits are identified by:
- An identifier, MP01 to MP64
- A name
- A comment

The name and the comment can be modified.

Use

Configuration bits are used to configure the Logipam program processing.

For example, configuration bit MP01 can be used to set the parameters of the switchgear control to the type of breaking device where:

- MP01=0 to control a circuit breaker with shunt trip coil
- MP01=1 to control a circuit breaker with undervoltage trip coil

Configuration bits are parameters, similarly to the Sepam general parameters.

They can be accessed from SFT2841 and are saved in the settings file with all the parameters.

They can also be viewed and modified during operation from the Sepam user-machine interface.

Logic Outputs

Definition

"Logic output" variables are Logipam output variables.

They can be used in both the program contact zone and coil zone.

Identification

Logic outputs are identified by:
- An identifier, OXYY
- A name
- A comment

The name and the comment can be modified.

Identifier

The OXYY logic outputs identifier depends:
- On the physical position of the MES120 module mounted on the Sepam base unit
- On the module output number

It is defined as follows:
- X = 0 for the 5 base unit outputs
- X = 1 for the 6 outputs of the MES120 module mounted in H1
- X = 2 for the 6 outputs of the MES120 module mounted in H2
- X = 3 for the 6 outputs of the MES120 module mounted in H3
- YY = 01 to 06 for the output number

Use

Controlling an output that is not physically present on Sepam or which has not been declared "Used" on SFT2841 has no effect.

A single Sepam logic output can be controlled via:

- The Logipam program
- The control matrix
- The "local control" function, from the mimic-based UMI

Sepam performs a logical OR on these three orders from different sources to control the output relay.

NOTE: A single logic output should never be controlled from these different functions.

We recommend that each logic output is controlled from a single function, and that predefined I/O are used to interface data sent by other functions.

The state of "Logic output" variables is only determined by their processing in the Logipam program.

The "Logic output" variables that are not controlled by Logipam have state 0.

The state of a "Logic output" variable is not necessarily the "logic image" of the electrical state of the Sepam output.

Example: O101 is controlled by Logipam and the control matrix.

- If O101=1 for Logipam and O101=0 for the control matrix, then the Sepam O101 output is controlled and logic output variable O101=1 for Logipam
- If O101=0 for Logipam and O101=1 for the control matrix, then the Sepam O101 output is controlled but logic output variable O101=0 for Logipam

Predefined Outputs

Definition

"Predefined output" variables are Logipam output variables.

Logipam predefined outputs are Sepam predefined function inputs.

They can be used in both the program contact zone and coil zone.

Identification

Predefined outputs are identified by:
- An identifier, SOxxx
- A name
- A comment

The name and the comment are predefined and cannot be modified.

The exhaustive list of predefined outputs is provided in the appendix and gives a complete identification of each predefined output (identifier, name and comment) as well as the function to which it is assigned.

For example:
- Identifier: SO057
- Name: V_CLOSECB
- Comment: circuit breaker closing order - Switchgear control function input
- Function input: Switchgear control

Use

Predefined functions are available in each Sepam depending on the chosen application: not all the predefined functions are processed by all Sepams.

If a predefined output is used in the Logipam program even though the associated function is not processed by Sepam, the value of this output is not processed.

Outputs to Matrix

Definition

The 128 "Output to matrix" variables are Logipam output variables to be linked to the control matrix inputs.

They can be used in both the program contact zone and coil zone.

Identification

Outputs to matrix are identified by:
- An identifier, MAT001 to MAT128
- A name
- A comment

The name and the comment can be modified.

Use

Outputs to matrix can be accessed in the SFT2841 "Control matrix" screen with the Logipam button.

They can be used to control:

- The Sepam logic outputs
- The indication LEDs on the front panel
- The display messages on the display unit
- Triggering of disturbance recording data

Protection Inputs

Definition

"Protection input" variables are Logipam output variables.

They can be used in both the program contact zone and coil zone.

Identification

Protection inputs are identified by:

- An identifier PXX_Y_ZZZ, where:
  - XX is the ANSI protection code
  - Y is the number of the protection unit
  - ZZZ is the variable sequence number, between 100 and 199
- A name, identical to the identifier
- A comment

The name and the comment are predefined and cannot be modified.

The exhaustive list of protection inputs is provided in the appendix.

For example:

- Identifier: P12_1_101
- Name: P12_1_101
- Comment: Protection 12 Unit 1 Reset

Use

Protection functions are available in each Sepam depending on the chosen application: not all protection functions are processed by all Sepams.

If a protection input is used in the Logipam program even though the associated function is not processed by Sepam, the value of this input is not processed.

Remote indications

Definition

The 240 "Remote indication" variables are Logipam output variables. These are logic data items sent by Sepam to a remote system (a supervisor, for example) by the communication system.

They can be used in both the program contact zone and coil zone.

Identification

Remote indications are identified by:
- An identifier, TS001 to TS240
- A name
- A comment

The name and the comment can be modified.

Use

Remote indications are used:
- By predefined control and monitoring functions: each remote indication is preassigned to a logic data item resulting from the processing of a predefined function.
- Or by Logipam: any logic data item can be assigned to each remote indication

Once a remote indication is used by Logipam, it is no longer generated by the predefined function to which it has been assigned.

Timers

Definition

60 "Timer" objects are available in Logipam.

They can be used:

- As an on-timer (TON), to delay the appearance of a data item for a time T
- As an off-timer (TOF), to delay the disappearance of a data item for a time T

Identification

Timers are identified by:

- A predefined identifier, T01 to T60
- A name
- A comment

The name and the comment can be modified.

Variables and Parameters

Each timer is linked to:

- A status variable
- A control instruction (TON or TOF)
- The "duration" parameter
- The duration setting range

State of a Timer

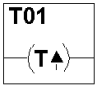
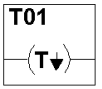
The status variable follows the state at the start of the timer with an on-delay or off-delay.

The status variable is used in the contact zone, and is identified as the object.

A Timer TON or TOF Control Instruction

Two special instructions can be used to control a timer to be used in the coil zone. However, a timer can only have one coil on the whole diagram.

These two instructions are as follows:

Coil	Control
	TON timer Start of an on-timer, on the rising edge of the associated condition programmed in the contact zone.
	TOF timer Start of an off-timer, on the falling edge of the associated condition programmed in the contact zone.

Duration of a Timer

The duration of each timer can be set between 0 and 650000 ms in 10 ms steps:

- With Logipam, from the object entry zone
- With SFT2841, in the **Timers** tab of the Logipam screen.

Setting Range

The timer duration setting range is used to fix the maximum and the minimum value of this duration.

This defined setting range is used to limit the values allowed for the timer duration during setting with SFT2841.

Counters

Definition

24 "Counter" objects are available in Logipam.

Identification

Counters are identified by:
- An identifier, C01 to C24
- A name
- A comment

The name and the comment can be modified.

Variables and Parameters

Each counter is linked to:

- A status variable
- A current value
- Two control instructions
- The "set point" parameter

State of a Counter

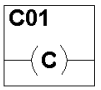
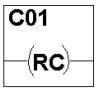
The status variable is:

- At 1 when the current counter value equals the set point
- At 0 when the current counter value is different from the set point

The status variable is used in the contact zone.

Counter Control Instructions

Two instructions can be used to control a counter to be used in the coil zone:

Coil	Control
	Increments the counter by one unit.
	Resets the counter to zero.

Counter Set Point

The set point can be set for each counter:

- With Logipam, from the object entry zone
- With SFT2841, in the **Counters** tab of the Logipam screen.

The set point of each counter must be set between 0 and 65535.

Viewing the Current Value

The current value of the counters can be viewed:

- With SFT2841, in the **Counters** tab of the Logipam screen
- On the Sepam display unit
- Remotely, via the communication network

Clocks

Definition

4 "Clock" objects are available in Logipam.

Clocks are used to generate periodic pulses with a weekly cycle to trigger periodic actions.

Identification

Clocks are identified by:
- An identifier, H01 to H04
- A name
- A comment

The name and the comment can be modified.

Variables and Parameters

Each clock is linked to:

- A status variable
- The following parameters:
  - A start time, configurable in hours/minutes/seconds
  - A period, configurable in hours/minutes/seconds
  - A number of pulses per day, less than 1440
  - The days of the week on which the pulses will be generated

For example, to generate 4 pulses every Monday and Wednesday at 08.00, 11.00, 14.00 and 17.00, a clock must be set with the following parameters:

- Start time: 08.00
- Period: 3 hours
- Number of pulses: 3 (the first pulse 08.00 is generated automatically by Logipam)
- Days: Monday and Wednesday

State of a Clock

On each pulse, the status variable is at 1 for one cycle duration.

The status variable is used in the contact zone.

Setting a Clock

The parameters can be set for each clock:

- With Logipam, from the object entry zone
- With SFT2841, in the **Clocks** tab of the Logipam screen

Ladder Language

3

Scope of This Chapter

This chapter describes the general principles of Ladder language.

What Is in This Chapter?

This chapter contains the following topics:

General Presentation of Ladder Language

Introduction

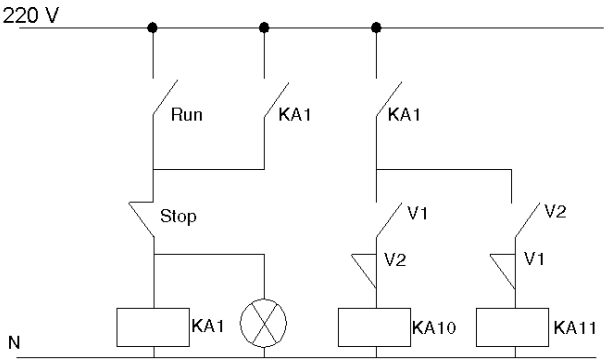
Ladder language is the Logipam programming language.

This graphic language, adapted to combinational processing, is used to transcribe ladder diagrams into a software environment.

A program is a diagram consisting of a set of instructions executed sequentially by Sepam.

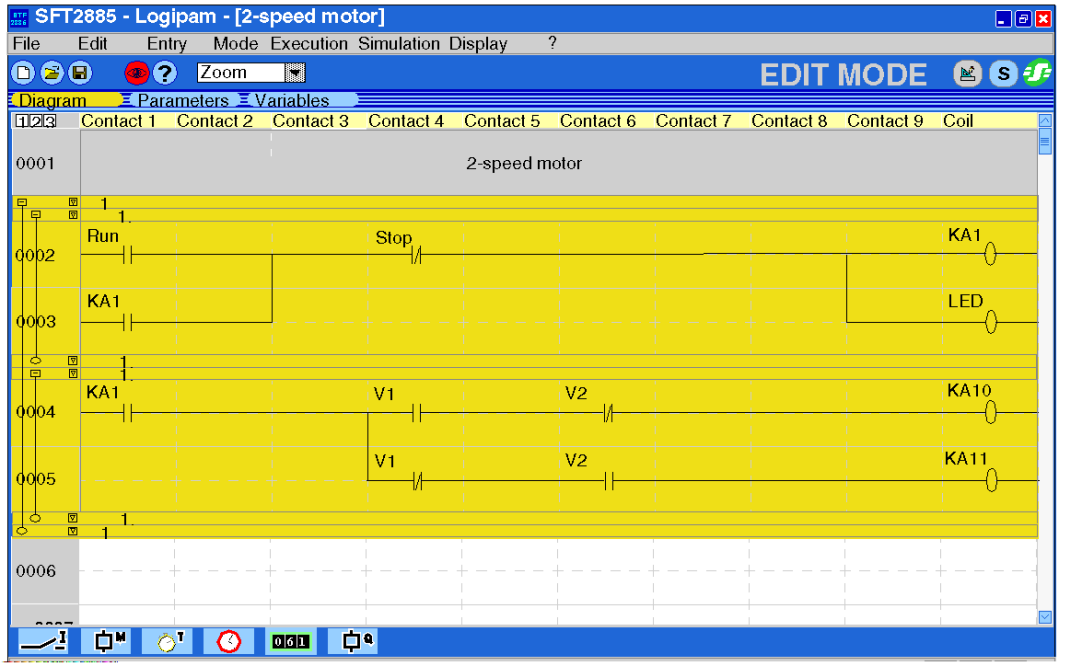
Example of Diagram

The figure below represents an electrical diagram.



Transcription Into a Ladder Diagram

The figure below illustrates how to transcribe the electrical diagram into a Logipam ladder diagram.



By analogy with an electrical diagram, the coils are executed if they are linked by a passing connection on the left-hand potential bar.

Ladder Language Instructions

Logipam Ladder language has three types of instruction:

- Contacts
- Connections
- Coils

Ladder diagrams can contain a maximum of 5000 contacts and coils.

Basic Principles of Ladder Language Programming

The programming principles are as follows:

- Connections are always passing
- The different types of contact are passing, depending on the value of the associated variable
- The combination of series and parallel contacts via connections is used to program the logic conditions for executing coils

Diagram Structure

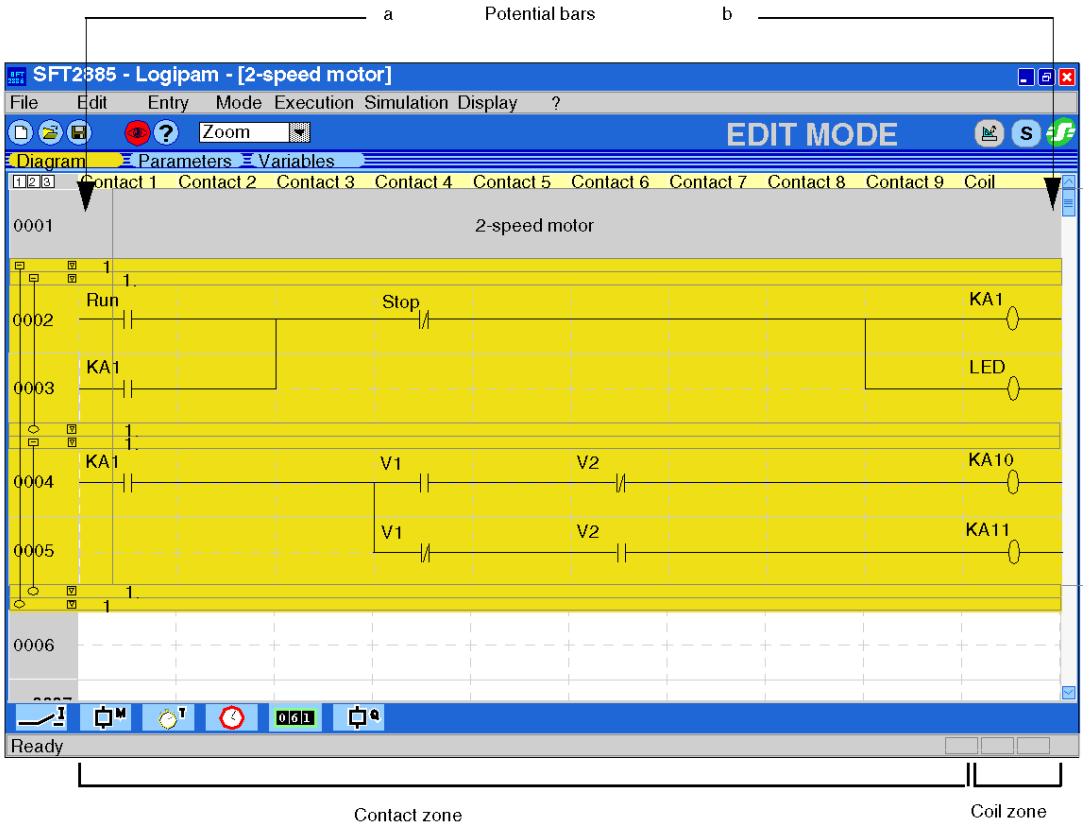
General

A Ladder diagram is drawn between two potential bars (a) and (b).

The current flows from the left-hand potential bar (a) to the right-hand potential bar (b).

Illustration

The figure below illustrates the representation of the Ladder diagram between the potential bars.



Description of the Ladder Diagram

A Ladder diagram consists of a set of graphic elements arranged on a data entry grid of:

- 10 columns
  - 9 columns representing the contact zone in which the conditions necessary for an action appear
  - 1 column representing the zone for the coil that performs the action following a sequence of tests in the contact zone
- 1400 numbered lines
  - 1000 programming lines
  - 400 comment lines

Cells

The intersections of lines and columns are cells.

A cell can contain a contact, a connection or a coil.

Sections

To structure a program and make it easy to read, the lines can be grouped in sections and sub-sections.

Lines can be grouped in a maximum of 100 sections.

A section can contain two levels of sub-section.

Sections and sub-sections are numbered, and can be allocated a title.

Execution of a section may be conditioned by a variable. The section will or will not be executed according to the value of the selected variable, as follows:

- Variable=0, the section is not executed in the program
- Variable=1, the section is executed in the program

Contacts

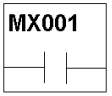
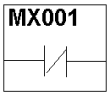
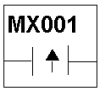
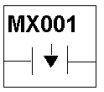
Definition

Contact type instructions are positioned on a cell, in the contact zone.

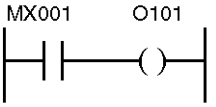
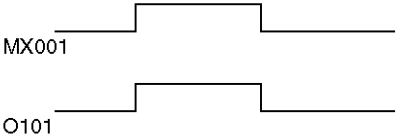
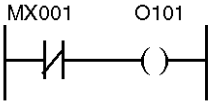
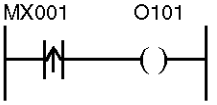
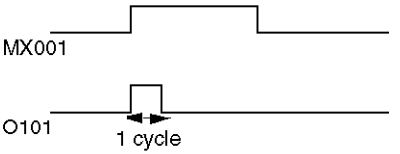
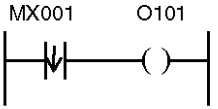
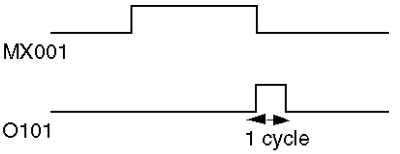
Contacts are used to test the state of variables.

Description

The 4 types of contact and their functions are indicated in the table below.

Type	Illustration	Function
Normally open contact		The contact is passing when the variable that controls it is at state 1.
Normally passing contact		The contact is passing when the variable that controls it is at state 0.
Rising edge contact		The contact is passing for one cycle on detection of the change from 0 to 1 of the variable that controls it.
Falling edge contact		The contact is passing for one cycle on detection of the change from 1 to 0 of the variable that controls it.

Examples

Logipam program	Equivalent timing diagram
	
	
	
	

Connections

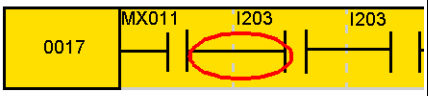
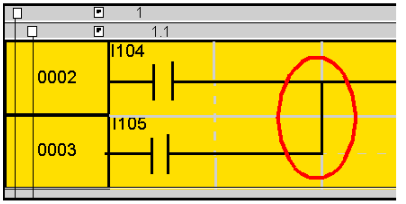
Definition

Connections are used to link up the contacts and coils.

They are represented by lines.

Description

The types of connection and their functions are indicated in the table below.

Type	Illustration	Function
Horizontal connection		Used to link instructions in series between the two potential bars.
Vertical connection		Used to link instructions in parallel.

Coils

Definition

Coil-type instructions are positioned on a cell, in the coil zone.

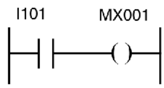
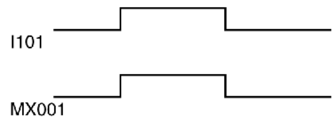
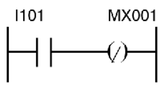
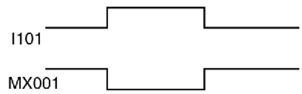
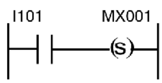
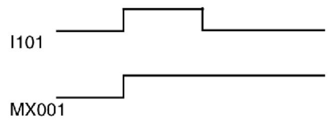
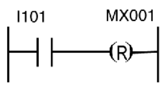
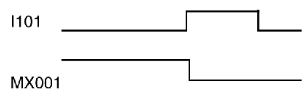
A coil is active when at least one closed string of contacts and connections links it to the left-hand potential bar.

Coils for Action on Variables

The types of coil and their functions are indicated in the table below.

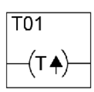
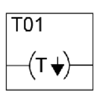
Type	Illustration	Function
Direct coil		The associated variable takes the value of the result of the contact zone.
Inverse coil		The associated variable takes the inverse value of the result of the contact zone.
Set coil		The associated variable is set to 1 when the result of the contact zone is at 1.
Reset coil		The associated variable is set to 0 when the result of the contact zone is at 1.

Examples

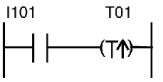
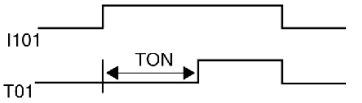
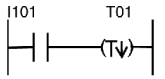
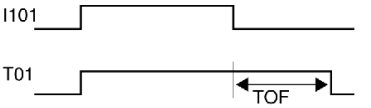
Logipam program	Equivalent timing diagram
	
	
	
	

Coils for Action on Timers

The types of coil on timers and their functions are indicated in the table below.

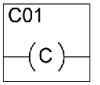
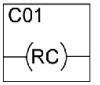
Type	Illustration	Function
TON timer		An on-timer is used to delay the pick-up of a data item for a time T.
TOF timer		An off-timer is used to delay the drop-out of a data item for a time T.

Examples

Logipam program	Equivalent timing diagram
	
	

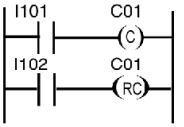
Coils for Action on Counters

The types of coil on counters and their functions are indicated in the table below.

Type	Illustration	Function
Counter		Increments the counter on a rising edge.
Reset counter		Resets the counter

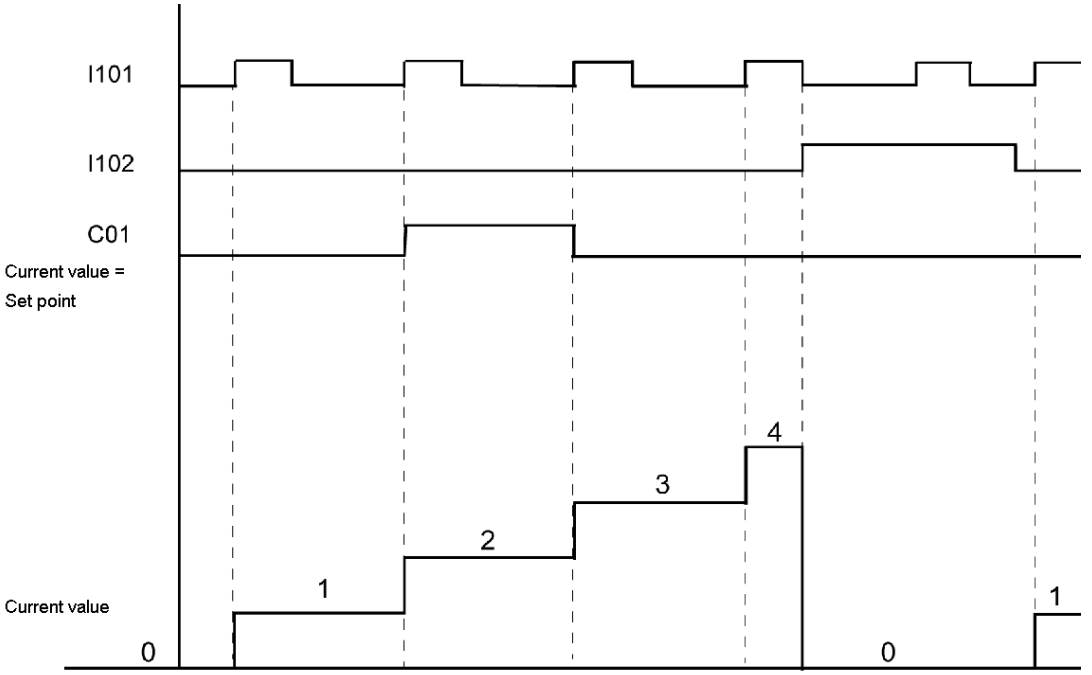
Example

Logipam program



Equivalent timing diagram

In this example, the set point equals 2



Comment Lines

General

Comment lines make it easier to read and interpret the diagram, but they are not compulsory.

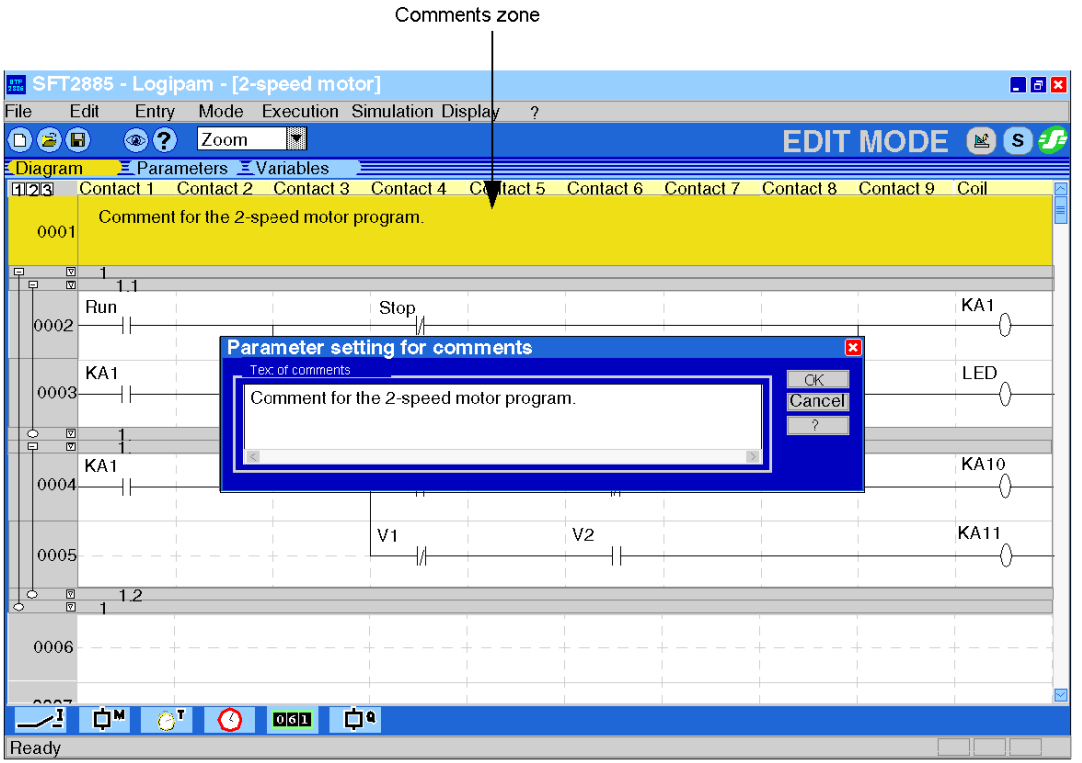
Description

The comment is integrated in the diagram. It consists of a maximum of 3 lines of 60 characters each.

400 comment lines are available.

Illustration

The figure below shows the position of the comment.



Rules for Executing the Diagram

Sepam Execution Cycle

A Sepam execution cycle includes the following operations:

- Acquisition of the logic inputs
- Processing of the protection functions
- Processing of the Logipam program
- Processing of the predefined functions
- Processing of the control matrix
- Control of the logic outputs, LEDs and messages

Sepam executes a cycle every 14 ms.

Rule for Executing the Diagram

A set of consecutive lines linked by vertical connections is a rung.

A program is executed sequentially, rung and lines in the order of the diagram.

The data is evaluated starting at the coil and working towards the left-hand potential bar. The variable associated with the coil takes the value resulting from operations performed and tested on the contacts.

Executing Elements in a Rung

The elements are executed in order of appearance of the rung coils on the diagram.

The contacts linked to the coil are evaluated according to the associated variables. Contacts that have already been evaluated during processing of a coil are not re-evaluated.

If there is a succession of closed contacts and connections, then the coil is executed and the variable associated with the coil is updated.

Conditioned Execution of a Section

Execution of a section may be conditioned by a variable. The section will or will not be executed according to the value of the selected variable, as follows:

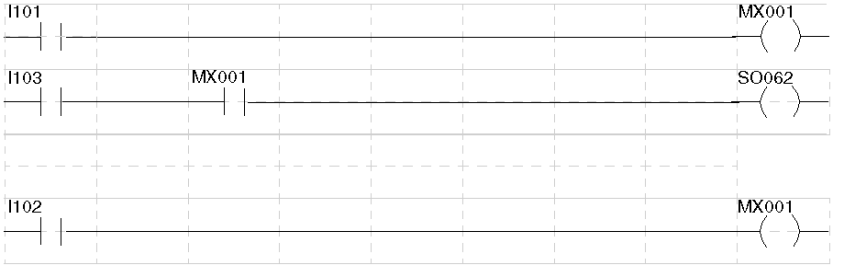
- Variable=0, the section is not executed in the program, the variables that could have been modified inside the section retain their value
- Variable=1, the section is executed in the program

Programming Tips

Using Coils

All coils encountered are executed, so a single counter used twice is interpreted twice.

A single coil can take several values during an execution cycle. However, only the value at the end of execution is retained, as the intermediate values are replaced each time there is a new coil. In the example below, MX001 takes the value I101 during the cycle and the value I102 at the end of execution.



Detection of the First Cycle

With Logipam, it is possible to use a line without a contact. This is used mainly to indicate the first execution cycle in order to perform initializations.

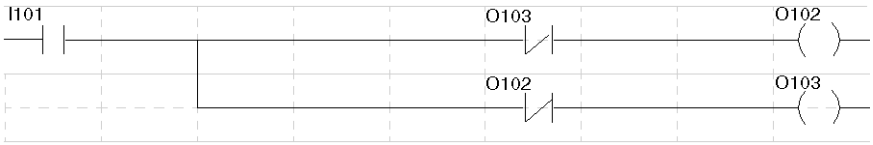
Example with MX002 at the end of the diagram

- MX002=0, it is the first execution cycle
- MX002=1, it is no longer the first execution cycle

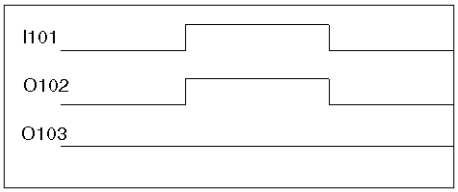
Order of Coils

The order of the coils is important. The examples below illustrate the differences in evaluation according to the order of the coils.

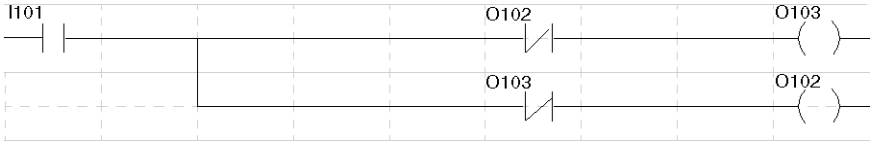
Example 1



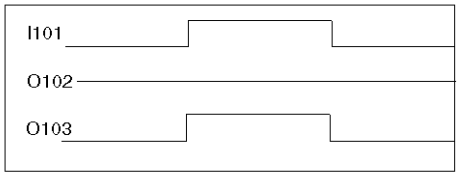
Timing diagram associated with example 1



Example 2



Timing diagram associated with example 2



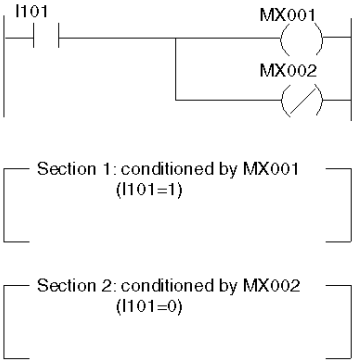
Order of Coils for the Set/Reset Function

For the Set/Reset function, the order of the coils conditions the order of priority between the Set and the Reset. The result of the last coil executed will be retained. For the Set to have priority, the Set coil must come last, and vice versa for the Reset to have priority.

Using Sections to Perform Exclusive Processing

Conditional sections simplify organization of two mutually-exclusive parts of a program.

Example: Section 1 is performed when I101=1 and section 2 when I101=0



Familiarization With the Software

4

Scope of This Chapter

This chapter describes the Logipam software and the initial operations for getting started.

What Is in This Chapter?

This chapter contains the following topics:

Starting the Software

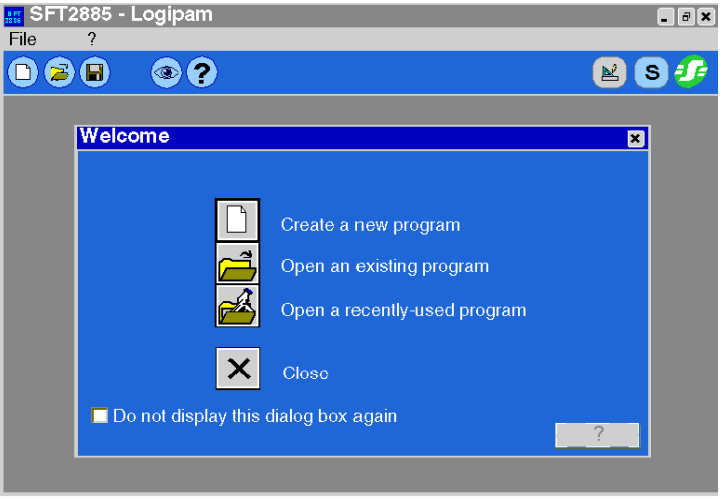
Introduction

To start Logipam, you can:

- Use the Windows menu
- Or use the shortcut available on your workstation desktop

Opening Logipam

To open Logipam from the Windows menu, proceed as follows:

Step	Action
1	In Windows Explorer, click on **Start** → **Programs** → **Schneider** → **SFT2885**. The **Welcome** dialog box opens. 
2	Select one of the options to open a programming project.

The Start-up Wizard

The start-up wizard guides you by offering shortcuts for the following routine actions:

- Creating a new program
- Opening an existing program
- Opening a recently-opened program
- Exiting the application

The wizard always opens by default when Logipam is started.

To disable it, you can:

- Check the **Do not display this dialog box again** box in the **Welcome** dialog box.
- Or open the **Options** window from the **File** → **Options** menu, and deselect the **Display welcome box** box.

To re-enable it, open the **Options** window from the **File** → **Options** menu, and check the **Display welcome box** box.

Creating a New Program

To create a new program, proceed as follows:

Step	Action
1	In the **Welcome** dialog box, click the **Create a new program** button. The **Choice of editing language** dialog box opens.
2	Select one of the suggested languages. The selected language corresponds to the diagram language. The Logipam main window opens.

Opening an Existing Program

To open an existing program, proceed as follows:

Step	Action
1	In the **Welcome** dialog box, click the **Open an existing program** button. The **Open** dialog box is displayed so you can select a program that has already been created.
2	Select a program (`.bft file`) and confirm. The Logipam main window opens.

Opening a Recently-Used Program

To open a recently-used program, proceed as follows:

Step	Action
1	In the **Welcome** dialog box, click on the **Open a recently-used program** button. The list of most recently-used programs opens.
2	Select a `.bft` file in the proposed list.
3	Click **OK**. The Logipam main window opens.

Importing a Program

Introduction

The Logipam program import function is used to retrieve a program from Sepam.

Functionality

During the import, the imported file is converted so that it can be used by Logipam. The program is validated automatically.

Procedure

To import a program, proceed as follows:

Step	Action
1	Open Logipam.
2	Click on **File** → **Import from SFT2841.** The **Open** dialog box is displayed.
3	Select the program.
4	Click **OK**. A dialog box is displayed to tell you the result of the import.

Software Options

Introduction

The options available from the **File** → **Options** menu can be used to customize use of Logipam.

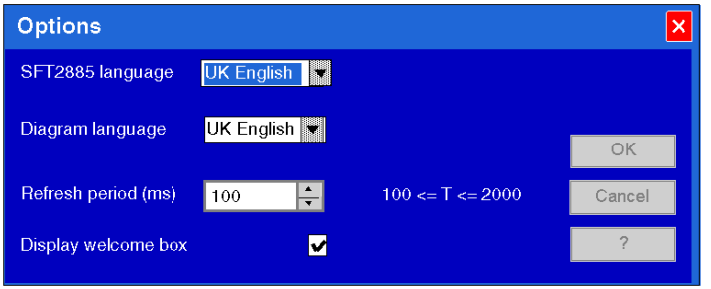
Description

The software options are as follows:

- Logipam interface language
  You can modify the software interface language (Logipam menus, windows, messages). The interface is available in UK English, US English, French and Spanish.
- Predefined text language
  You can modify the language of comments and names of predefined variables. The information is available in UK English, US English, French and Spanish.
- Refresh period
  The refresh period corresponds to the time that elapses between each updating of the diagram and the variables in simulation mode.
  By default, the refresh period is 100 ms. This value can be increased if the simulations run too slowly.
- Activate or deactivate display of the **Welcome** box on start-up.

Selecting the Interface Language

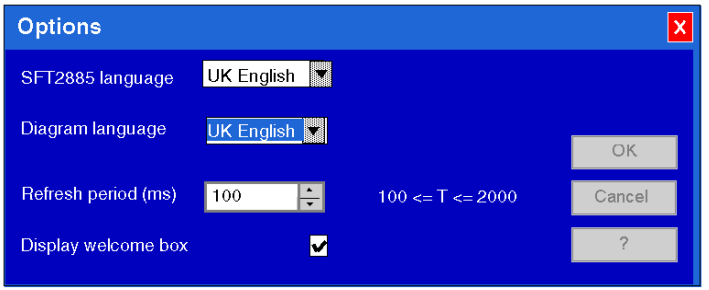
To select the interface language, proceed as follows:

Step	Action
1	Open Logipam. The Logipam main window opens.
2	Click on **File** → **Options**. The **Options** dialog box opens. 

Step	Action
3	In the **SFT2885 language** drop-down menu, select the software interface language and click **OK**. The selected language corresponds to the Logipam interface language. It will not become active until the next time Logipam is started.
4	Restart the application to activate the selected language.

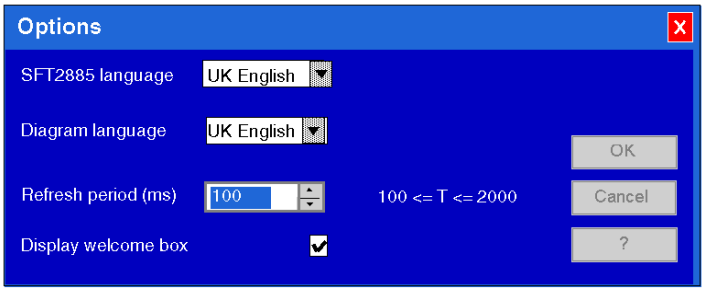
Selecting the Predefined Text Language

You can also modify the language of comments and names of predefined variables in Logipam. To do this, proceed as follows:

Step	Action
1	Open Logipam. The Logipam main window opens.
2	Click on **File** → **Options**. The **Options** dialog box opens. 
3	In the **Diagram language** drop-down menu, select the desired language and click **OK**. **Important**: Only information in the requested editing language is displayed. If you select French as the diagram language but have previously entered text with UK English as the diagram language, the UK English text will not be displayed.

Modifying the Refresh Period

To modify the refresh period, proceed as follows:

Step	Action
1	Open Logipam. The Logipam main window opens.
2	Click on **File** → **Options** The **Options** dialog box opens. 
3	In the **Refresh period** field, enter the desired value and click on **OK** to close again.

Activating/Deactivating the Startup Wizard

To activate or deactivate the startup wizard, proceed as follows:

- Select the **Display welcome box** box to activate the wizard
- Deselect the **Display welcome box** box to deactivate the wizard

Password Protection

Introduction

Logipam allows a program to be protected with a password.

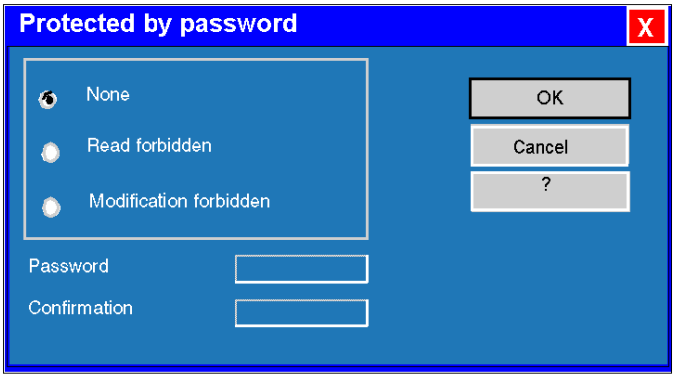
Description

The three levels of protection available are:

- None
  (Default protection level)
  The program is not protected.
- Read forbidden
  The program is read-protected. You must enter a password to open it.
- Modification forbidden
  The program is write-protected. You can view it but you must enter a password in order to save any modifications.

Procedure

To protect a program with a password, proceed as follows:

Step	Action
1	Open Logipam.
2	Click on **File** → **Password**. The **Protected by password** dialog box opens. 

Step	Action
3	Select one of the protection options: **Read forbidden** or **Modification forbidden**, and enter a password. The password must include between 4 and 10 characters. The permitted characters are: a-z, A-Z and 0-9. The password must be confirmed by a second identical entry. **Remark**: The password is case-sensitive.
4	Click **OK**. If you have selected the **Read forbidden** option, Logipam will ask you to enter your password the next time the program is opened. If you have selected the **Modification forbidden** option, Logipam will ask you to enter your password the next time the program is saved. **Remark**: To modify these options in the future, Logipam will ask you to enter your password again.

Description of the Main Screen

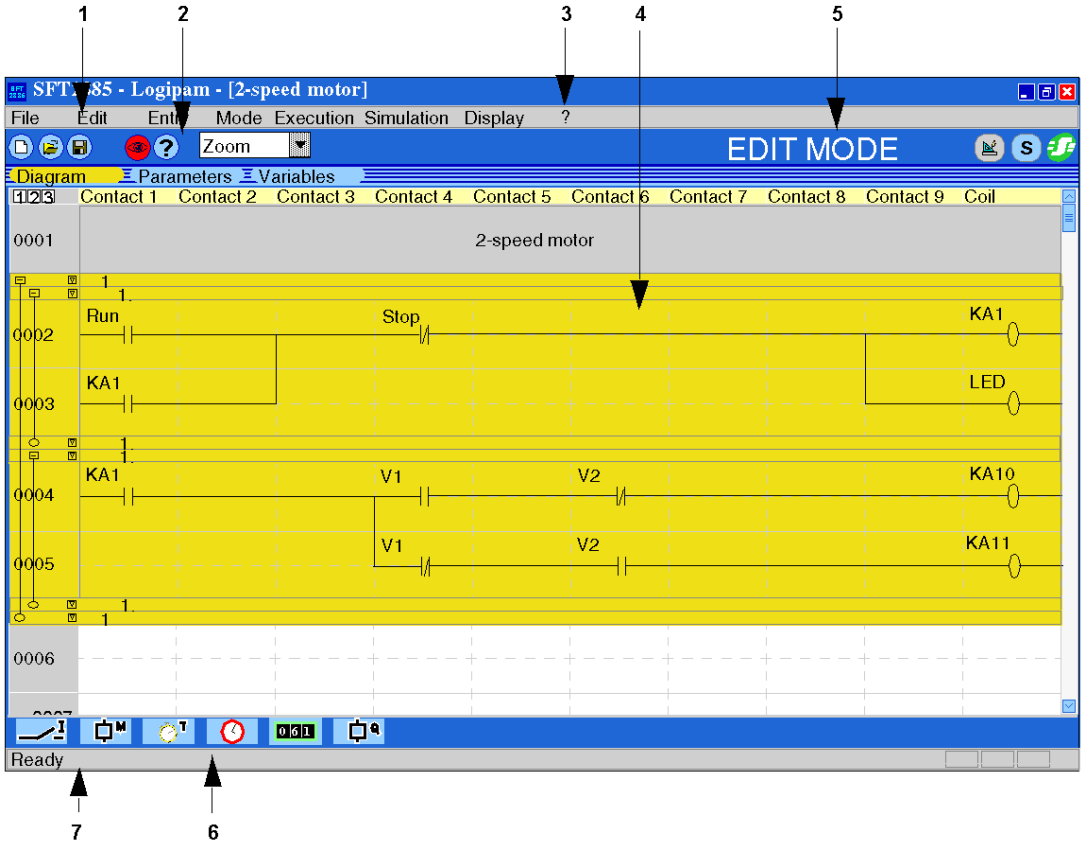
General

The Logipam main screen offers the general MS Windows functions:

- Contextual and drop-down menus
- Toolbars with icons
- Keyboard shortcuts and function keys
- Online help and info balloons

Illustration

The figure below shows the Logipam main screen:



Description

The table below describes the Logipam main screen:

Label	Element	Function
1	Menu bar	Provides access to the software functions.
2	Toolbar	Provides quick access to the standard mode functions during use.
3	On-line help	Supplies information about the software.
4	Programming zone	Main working zone. This main screen offers two operating modes: • Edit mode for entering program elements • Simulation mode for validating program operation
5	Operating mode	Indicates the active mode (edit or simulation) and is used to switch from one mode to the other.
6	Variables and objects bar	Provides access to the Logipam variables and objects.
7	Status bar	Supplies information relating to the software.

The Main Toolbar

Presentation

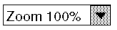
The toolbar provides quick access to the standard software functions.

Illustration

The main toolbar looks like this:

Description

The table below describes the icons available from the toolbar:

Icon	Function
	Create a new project
	Open an existing project
	Save the current project
	Consistency check
	Open the online contextual help
	Adjust zoom
	Switch to edit mode
	Switch to simulation mode

NOTE: These functions can be accessed via the menus in both edit mode and simulation mode.

Online Help

Presentation

The online help supplies information about using Logipam from all screens. It consists of this document.

Access to the Online Help

There are two suggested access methods:

Icon	Access
	Access from the icon on the menu bar to open the entire help file.
	Contextual access from each screen, by right-clicking the mouse.

The Variables and Objects Bar

Presentation

The "Variables and objects" toolbar located at the bottom of the screen shows all the categories of variables and objects.

This variables and objects bar can be used:

- In edit mode: to access all the variables and objects available in order to place them on the data entry grid
- In simulation mode: simply to view the variables and objects used in the program

Description

The figure below shows the variables and objects bar:

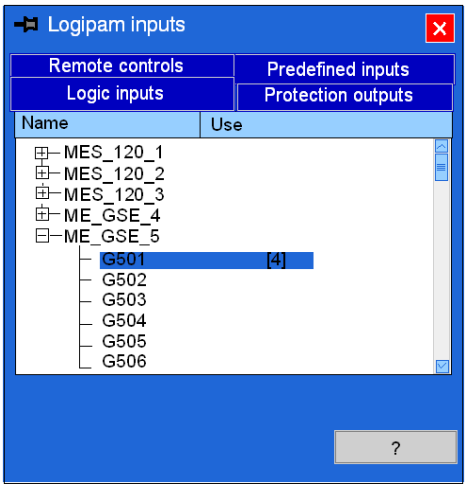
Each category of variables and objects is represented by an icon:

Icon	Description
	Logipam inputs
	Internal bits
	Timer objects
	Clock objects
	Counter objects
	Logipam outputs

Use

When the cursor passes over an icon, a window appears presenting all the variables or objects in the requested category.

For example, when the cursor passes over the Logipam inputs icon, the following window opens:



Nailing Windows Open with a Thumbtack

Several windows have a thumbtack in the top left-hand corner of the frame:

This window can be displayed permanently (nailed) by clicking on the thumbtack.

Edit and Simulation Operating Modes

Presentation

Logipam offers the following two operating modes for programming:

- Edit Mode
  Edit mode is used to enter the program (default mode).
- Simulation Mode
  Simulation mode is used to simulate program execution, to check its operation, and to make modifications to variables and objects.

Switching from One Mode to the Other

To switch from one mode to the other:

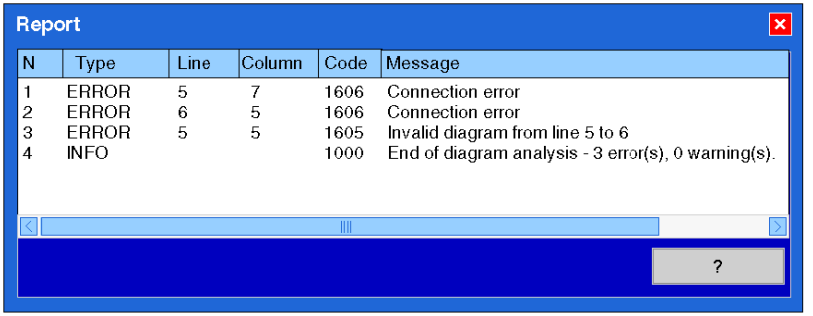
Mode	Action
Simulation	To switch to simulation mode, click on **Mode** → **Simulation** or on the following icon in the toolbar:  One of the two following scenarios may occur: ● The switch to simulation mode happens immediately. ● The switch to simulation mode does not happen. The **Report** window opens and gives information on analyzing the diagram and generating the executable program. Correct any errors you find, and continue. For further information on the report, see *Program Checking, page 117*.
Edit	To return to edit mode, click on **Mode** → **Edit** or on the following icon in the toolbar: 

Conditions for Switching to Simulation Mode

The switch to simulation mode should only be made on a project which is not undergoing modification; it should have been saved first.

If the current project has not already been saved, a window opens with the option of saving the project.

The switch to simulation mode is accompanied by the diagram analysis and executable code generation phases. The result of these phases is displayed and any errors are listed, as shown in the figure below:



If an error has been made, simulation mode cannot be accessed.

In order to continue, correct the errors.

Description of the Screen in Edit Mode

Presentation

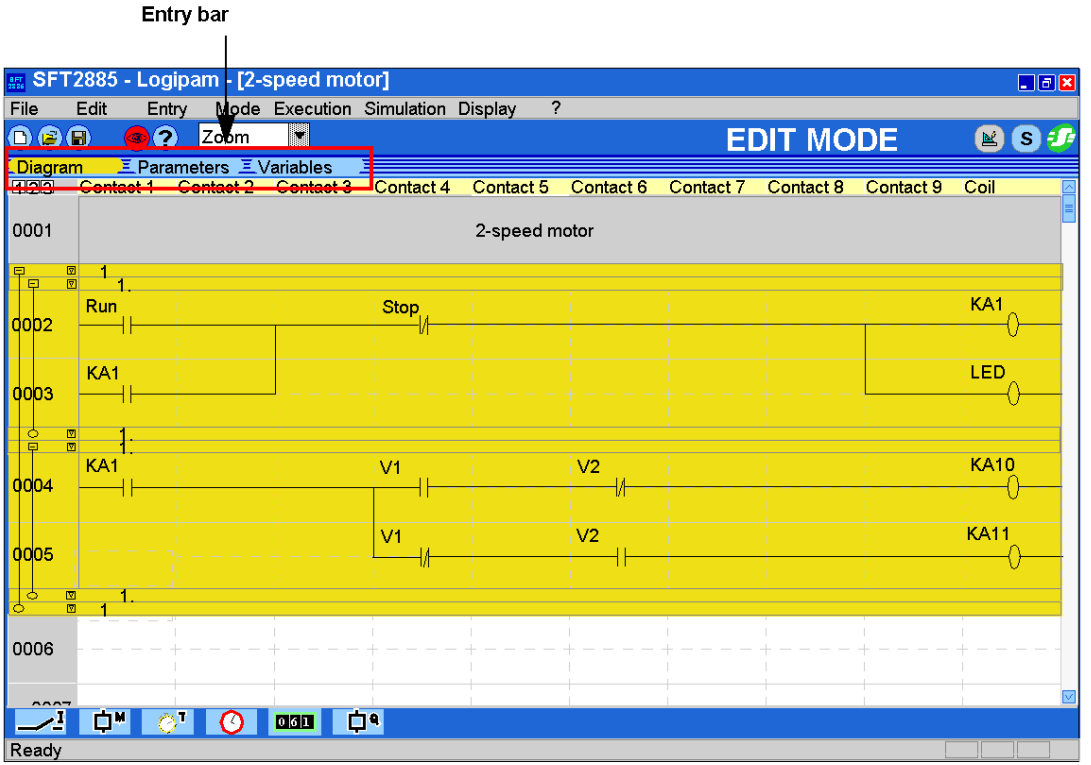
Edit mode is used to enter programming elements and structure them on the edit grid.

In edit mode the screen contains:

- The main menu bars and toolbars.
- The "Entry bar" toolbar containing three buttons. Each button provides access to a data entry mode. These modes can also be accessed from the **Entry** menu. The current data entry mode appears highlighted in yellow.

Description

The figure below shows the screen in edit mode and the entry bar:



The buttons available are as follows:

Button	Use
Diagram	Used for free entry of the program instructions (default data entry mode).
Parameters	Used to define the parameters for the timers, counters and clocks without positioning them on the program edit grid.
Variables	Used to enter the name and comment for all the variables.

"Diagram" Entry Mode

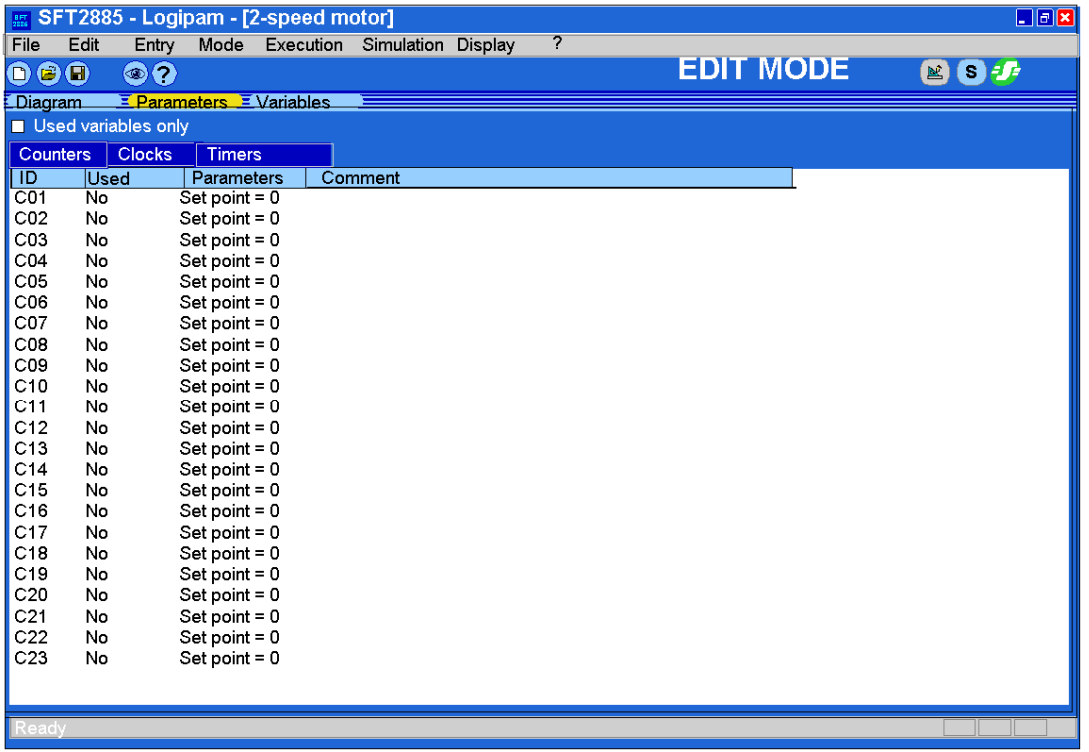
The **Diagram** mode is used to enter the ladder diagram instructions in order to create the program. It displays the program entry grid, as shown in the figure above.

In full-screen mode, 10 lines are visible but it is possible to display more lines by modifying the zoom.

"Parameters" Entry Mode

Parameters mode displays all the available objects whose parameters can be set. It is used to enter their parameters before placing them on the entry grid.

It consists of three tabs, one tab for each type of object, as shown in the figure below:



The information available for each tab is:

Column	Description
ID/Name	Indicates: ● The identifier (ID) for the object, if you have selected display of variables by identifier (menu **Display** → **Variables** → **By identifier (ID)**). ● The name of the object, if you have selected display of variables by name (menu **Display** → **Variables** → **By name**).
Used	Indicates whether the object is used in the diagram.
Parameters	Provides access to the parameters of each object.
Comments	Used to enter a text to add a comment to the object, by double-clicking on the line.
Type	Only in the **Timers** tab: this information indicates whether the timer is a TON or TOF type.

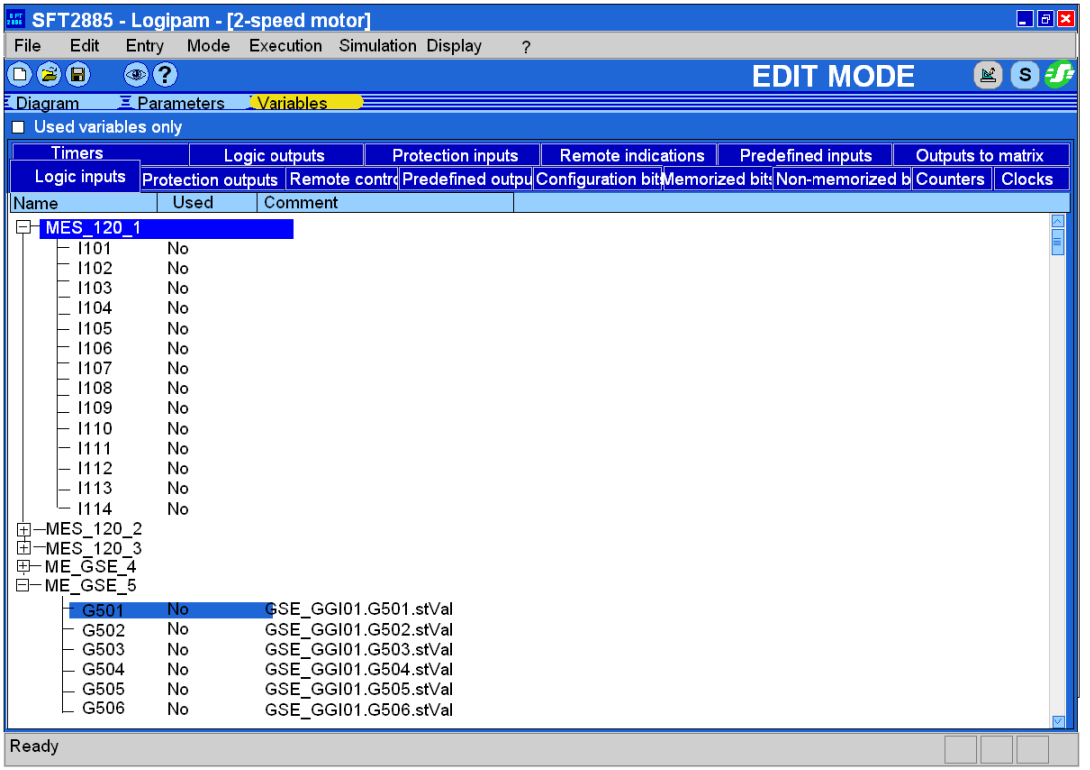
Double-clicking on the line of an object brings up its parameter-setting window.

The **Used variables only** check box is used to select and display only the objects used by the current program.

"Variables" Entry Mode

Variables mode displays all the variables and available objects. It is used to enter their name and a comment when they have not been predefined in Logipam.

It consists of one tab for each type of variable or object, as shown in the figure below:



The information available for each tab is:

Column	Description
ID/Name	Indicates: ● The identifier (ID) for the object, if you have selected display of variables by identifier (menu **Display** → **Variables** → **By identifier (ID)**). ● The name of the object, if you have selected display of variables by name (menu **Display** → **Variables** → **By name**).
Used	Indicates whether the variable is used in the diagram.
Comments	Used to enter a text to add a comment to the variable, by double-clicking on the line.

Click on the data entry field to enter the text.

The **Used variables only** check box is used to select and display only the variables and objects used by the current program.

Description of the Screen in Simulation Mode

Presentation

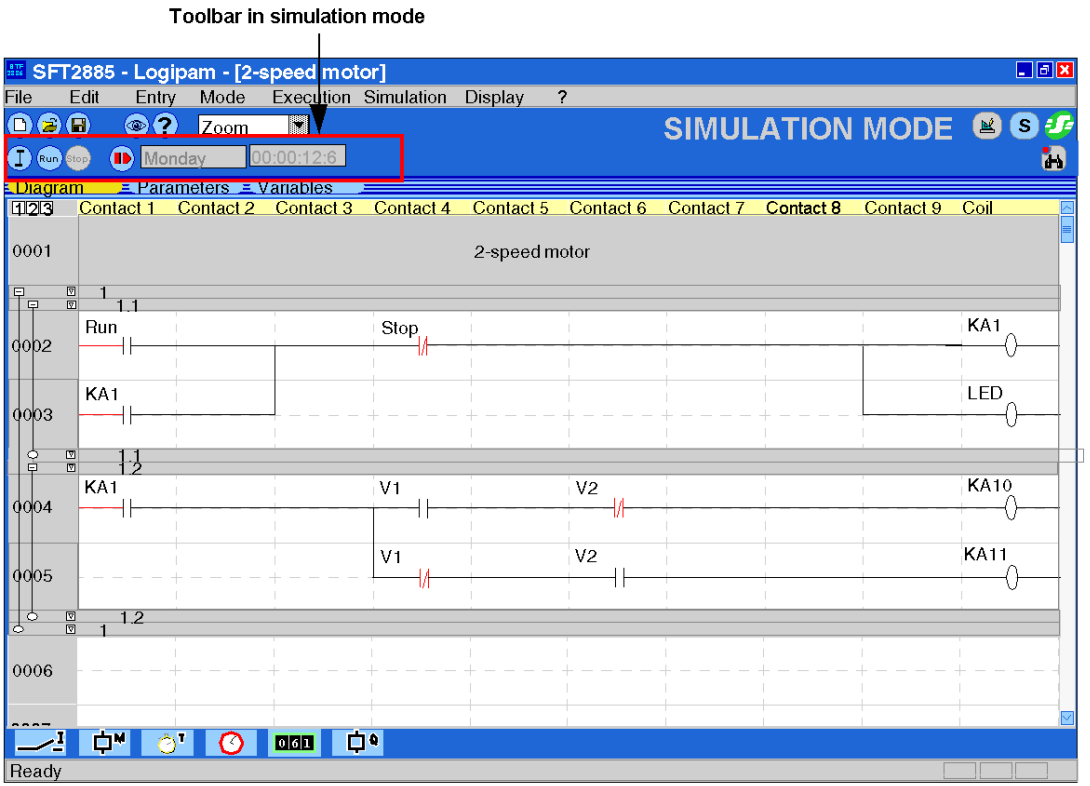
Simulation mode is used to check the program operation.

In simulation mode the screen contains:

- The main menu bars and toolbars.
- A special toolbar for accessing the simulation functions.

Illustration

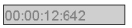
The figure below illustrates the program in simulation mode :



Simulation Toolbar

The simulation toolbar looks like this:

The table below describes the available icons:

Icon	Function
	Initialize variables
	Execute variables
	Stop simulation
	Execute one step
	Simulation day
	Simulation time
	Variable status window

Customizing the Display

Presentation

Logipam can be used to customize the working environment and to modify its display options.

Window Layout

Logipam stores the layout of the open windows before closing the current project.

At the next session, it opens in the same working environment with the same window layout on the screen.

Adjusting the Zoom

By default, the data entry grid in edit mode displays 10 lines for full-screen entry.

In Logipam, the zoom is a vertical zoom that allows a larger number of lines to be displayed.

Creating a Program

5

Scope of This Chapter

This chapter describes all the operations involved in creating a program.

What Is in This Chapter?

This chapter contains the following topics:

Defining the Properties of a Program

Introduction

Logipam is used to define the properties of a program such as the author, comments or even a version number.

This information is useful:

- When building the printer file
- When importing the program into SFT241 to identify the program and its version

List of Properties

The properties that can be assigned are as follows:

- The program name
- The program author
- The type of application
- The customer
- The project
- The switchboard
- The cubicle
- Comments

These properties appear when the cover page is printed. For reasons of traceability, it is a good idea to enter them all.

The properties automatically assigned by Logipam are as follows:

- The program version
- The SFT2885 software version

This information cannot be modified. The program version is incremented automatically each time it is saved after a program modification.

Procedure

To define the properties of a program, proceed as follows:

Step	Action
1	Open Logipam.
2	Click on **Edit** → **Program properties**. The **Program properties** dialog box opens.
3	Enter each of the available properties.
4	Click **OK**.

Defining Variables and Objects

Introduction

The variables and objects that are used in the program can be defined:

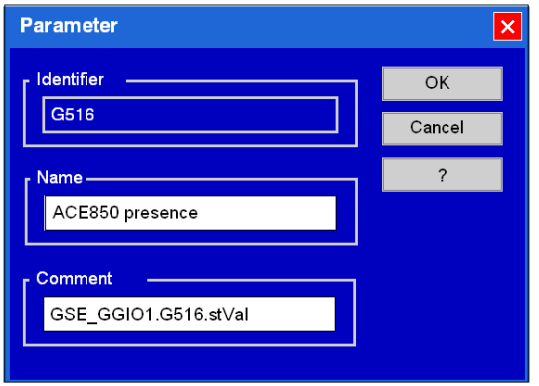
- Before being placed on the data entry grid, as follows:
  - using the **Variables** mode to enter the names and comments of the variables and objects
  - using the **Parameters** mode to enter the object parameters
- During entry of the elements in **Diagram** mode, using the parameter-setting window available for each variable and each object.

Definition in Variables and Parameters Mode

For further information on these data entry modes, see *Description of the Screen in Edit Mode, page 76*.

Defining Comments and Names of Variables in Diagram Mode

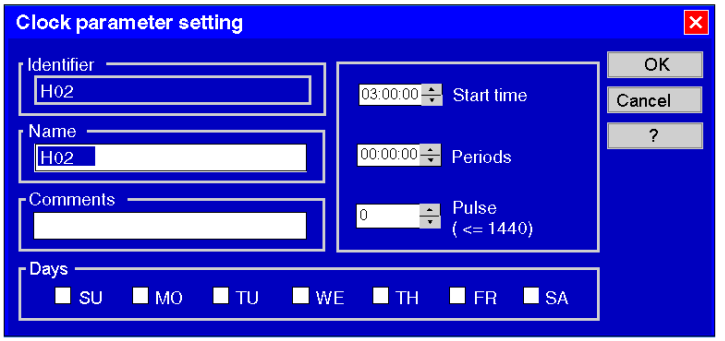
To enter the name and comment for a variable in **Diagram** mode, proceed as follows:

Step	Action
1	On the diagram grid, double-click on the variable to be defined. A **Parameter settings** dialog box opens, as shown in the figure below for a variable which has not been predefined.  There are two possible scenarios: ● The variable is a predefined variable in Logipam: the identifier, the name and the comment are already shown and cannot be modified. ● The variable has not been predefined in Logipam: the identifier is shown by default, the name and the comment can be modified. **NOTE:** When the variable corresponds to an IEC 61850 data item, its IEC 61850 identifier (LN.DO.DA) is indicated in the Comments zone.

Step	Action
2	Enter the information and click **OK**.

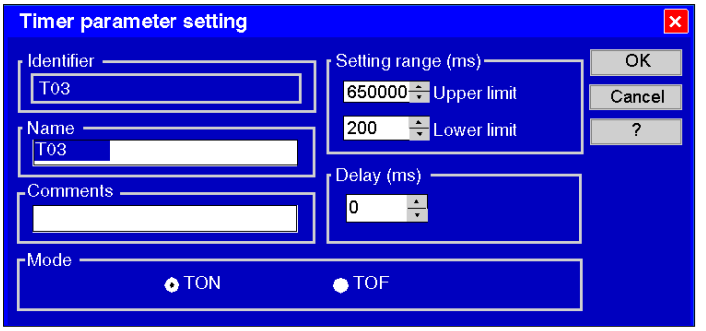
Defining and Setting Clock Parameters in Diagram Mode

To define the clock parameters, proceed as follows:

Step	Action
1	In **Diagram** mode, double-click on the clock whose parameters are to be set. The **Clock parameter setting** dialog box opens. 
2	Enter the information and click **OK**.

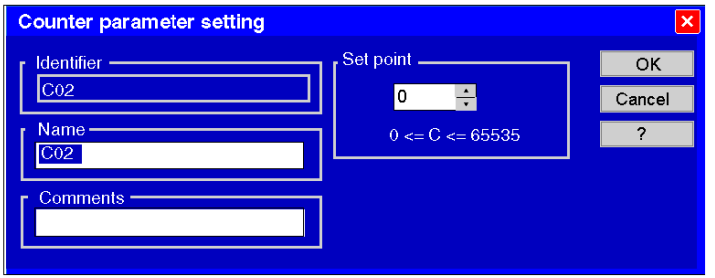
Defining and Setting Timer Parameters in Diagram Mode

To define the timer parameters, proceed as follows:

Step	Action
1	In **Diagram** mode, double-click on the timer whose parameters are to be set. The **Timer parameter setting** dialog box opens. 
2	Enter the information and click **OK**.

Defining and Setting Counter Parameters in Diagram Mode

To define the counter parameters, proceed as follows:

Step	Action
1	In **Diagram** mode, double-click on the counter whose parameters are to be set. The **Counter parameter setting** dialog box opens. 
2	Enter the information and click **OK**.

Note

You can also open the Parameter-setting window as a contextual menu by right-clicking on the cell of the variable or the object.

Creating a Line

Introduction

Creating a program line consists of:

- Inserting elements
  Select the variables or objects and place their graphic representation on a cell of the grid as a contact or a coil.
  For further information on graphic representations, see *Ladder Language, page 41*.
- Drawing the connections to link up the contacts and coils
  In a diagram you can create horizontal or vertical connections between the elements. The connections are represented by lines.
  To link two elements, you should draw as many lines as there are blank cells between the elements.
- Modifying the operations on the contacts and coils.

Inserting Variables and Objects

To place an element in a diagram, proceed as follows:

Step	Action
1	The **Diagram** tab is activated by default. If it is not, click on **Entry** → **Diagram**.
2	In the variables and objects bar, position the mouse cursor on the **Logipam inputs** button, for example. The dialog box corresponding to the **Logipam inputs** variables category opens.
3	Select the type of variable, the **Logic inputs** tab for example. A tree structure representing the logic inputs expands.
4	In the tree structure, select a logic input, for example input `I102` of the `MES_120_1` module.
5	Click and drag input `I102` onto a cell of the diagram, for example `Contact 4` of line `0002`. The activated cell appears as a yellow rectangle in the diagram.
6	Release the cursor. Input `I102` appears on the cell. In the dialog box corresponding to the "Inputs" variable, you can see that the position of the variable is displayed in the **Contact** column of input `I102` in form [2].

Creating Connections

To create a connection, see *Creating and Deleting Connections, page 110*.

Modifying Functions

To modify the functions of elements placed as a contact or a coil, proceed as follows:

Step	Action
1	In the diagram, select the graphic element whose function you wish to modify.
2	Right-click to access the contextual menu. The operations possible as a contact or as a coil appear according to the variable or object selected.
3	Select the operation. The graphic representation of the element is modified on the diagram grid.

Adding and Deleting Lines

Introduction

A program grid consists of 1400 lines.

On opening Logipam, 100 lines are available. You can add or delete lines as the program requires.

Adding a Line

To add a line to a program, proceed as follows:

Step	Action
1	In Logipam, make sure you are in edit mode. If not, click on **Mode** → **Edit**.
2	Select a cell or line in the program.
3	Click on **Edit** → **Add a line after the selection** or **Add a line before the selection**, as appropriate. A new line appears in the program. The number of the new line appears in the line header in the grayed-out area to the left of the screen. **Notes**: 1. To add a line, you can also right-click on the cell or the line, then select **Add a line after the selection** or **Add a line before the selection**. 2. To add a line at the end of the program, you should scroll through the program until you find the last line. Click on the icon representing a down arrow below the last line of the program: a new line appears.

Adding Several Lines

To add several lines to a program, proceed as follows:

Step	Action
1	In Logipam, make sure you are in edit mode. If not, click on **Mode** → **Edit**.
2	Select a cell or line in the program.
3	Right-click on the cell or line.
4	Select the **Insert n lines** option. The **Insert of n lines** dialog box opens.
5	Enter the number of lines you wish to insert (10 lines by default). The new lines appear in the program before the selected line.

Deleting Lines

To delete lines from a program, proceed as follows:

Step	Action
1	In Logipam, make sure you are in edit mode. If not, click on **Mode** → **Edit**.
2	Select the lines you wish to delete from the program.
3	Click on **Edit** → **Delete**, or right-click on the lines, then select the **Delete** option, or use the **<Del>** key on your keyboard.

Defining, Canceling and Deleting Sections

Introduction

To structure a program and make it easy to read, the lines can be grouped in sections.

Section Characteristics

A section is defined with a sequence number in the diagram.

You can attribute a title as well as an execution variable to the section.

You can create up to 100 sections per diagram, and a section can contain two sub-section levels.

Creation Conditions

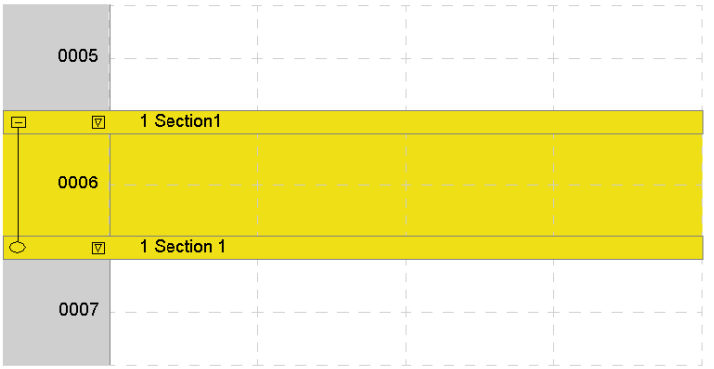
The definition of a section is rejected if :

- The selected lines contain more than 2 section sub-levels.
- The selected lines already belong to part of a section.
- Vertical connections have been made at the limits of the selected lines where you wish to create a section.

Defining a New Section

To define a new section, proceed as follows:

Step	Action
1	Make sure you are in edit mode. If not, click on **Mode** → **Edit**.
2	Select the line where you wish to create a section.

Step	Action
3	Right-click on the line, then select the **Define a section** option. The section appears on the line you have just selected. A section expansion icon represented by a square appears in the line header as shown in the figure below: 

Defining a Section from Sections or Lines Already Created

To define a section from existing sections or lines, proceed as follows:

Step	Action
1	Make sure you are in edit mode. If not, click on **Mode** → **Edit**.
2	Select the sections or the sections and lines you wish to form into one section.
3	Right-click on the sections and lines, then select the **Define a section** option. The section appears on the sections or lines you have just selected.

Canceling a Section

Canceling a section or a sub-section deletes the section definition. The grouping of lines by section is thus canceled, and the lines retain their content. To cancel a section, proceed as follows:

Step	Action
1	Make sure you are in edit mode. If not, click on **Mode** → **Edit**.
2	Select the section or sub-section you wish to cancel.
3	Right-click and select the **Cancel a section** option. The grouping of lines in sections is canceled, and the section limits disappear.

Conditions for Deleting Sections

Caution: Deleting a section deletes the lines and their content. Deletion of lines is irreversible. This action cannot be canceled.

Deleting sub-sections must not generate an empty section. Therefore, make sure that there is at least one line or section in the section resulting from the deletion of sub-sections or lines.

Deleting Sections

To delete sections, proceed as follows:

Step	Action
1	Make sure you are in edit mode. If not, click on **Mode** → **Edit**.
2	Select the sections you wish to delete from the diagram.
3	Click on **Edit** → **Delete** or right-click on the sections to be deleted, then select the **Delete** option.

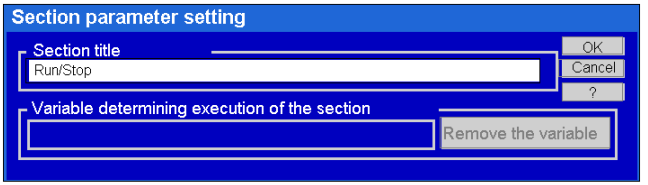
Section Title

Introduction

Once a section exists in a diagram, you can allocate it a title.

Procedure

To allocate a title to a section, proceed as follows:

Step	Action
1	Make sure you are in edit mode. If not, click on **Mode** → **Edit**.
2	Select the section to which you wish to allocate a title.
3	Double-click on the section header The **Section parameter setting** dialog box opens.  You can also right-click on the section, then select the **Parameter-setting window** option.
4	Enter the section title. The title is displayed at the start and end of the section.

Creating and Deleting an Execution Condition on a Section

Introduction

A section may be subject to an execution condition defined by a variable.

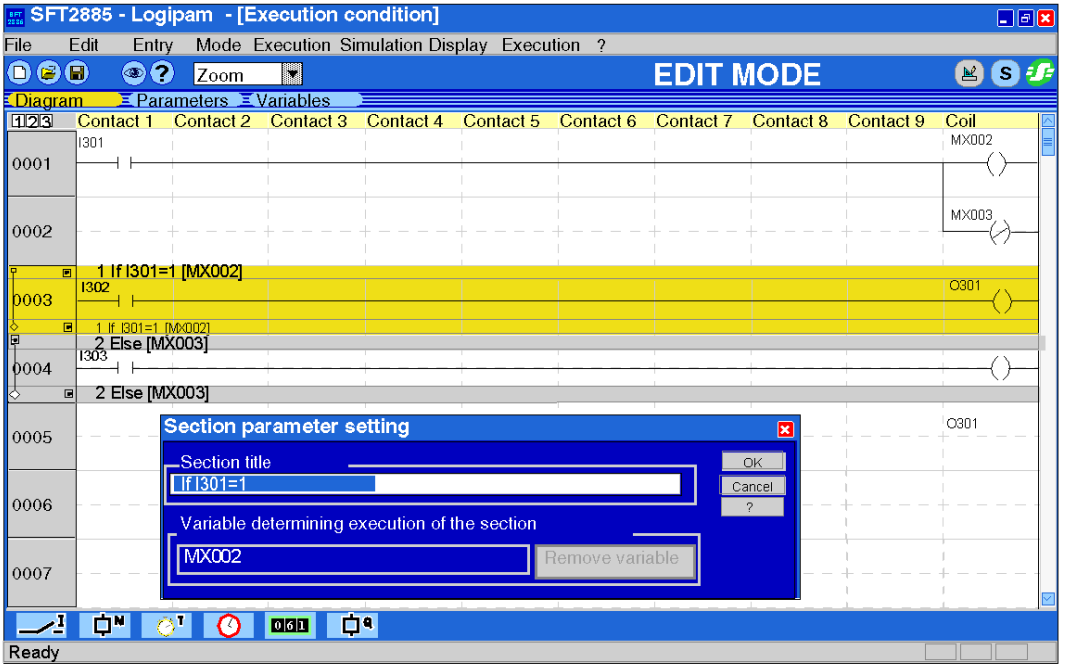
Functionality

To create an execution condition, a variable is linked to the section:

- If the variable is at 0, the section is not executed.
- If the variable is at 1, the section is executed.

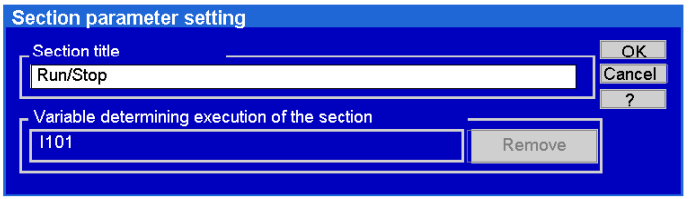
Illustration

The figure below illustrates insertion of an execution condition:



Creating a Condition

To allocate an execution condition to a section, proceed as follows:

Step	Action
1	Make sure you are in edit mode. If not, click on **Mode** → **Edit**.
2	Select the variable which will determine the section and drag it onto the section header. The identifier or the name of the variable appears in the section title bar (I101 in the example below).  The Run/Stop section is executed if the value of variable I101 is 1.

Deleting a Condition

To delete the execution properties from a section, proceed as follows:

Step	Action
1	Make sure you are in edit mode. If not, click on **Mode** → **Edit**.
2	Select the section from which you wish to delete the execution condition.
3	Double-click on the section header The **Section parameter setting** dialog box opens. You can also right-click on the section, then select the **Parameter-setting window** option.
4	Click on the **Remove variable** button to delete the execution condition.

Expanding and Reducing the Content of a Section

Introduction

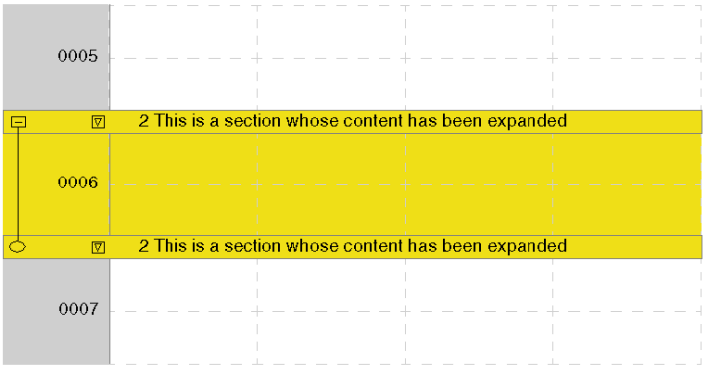
You can expand or reduce the content of a section in a diagram.

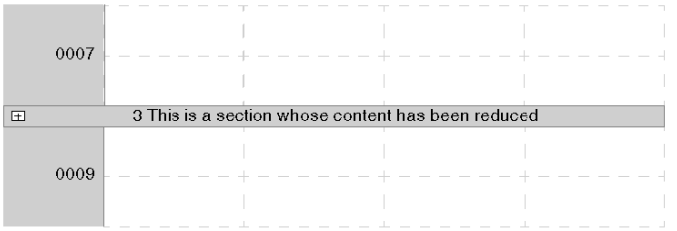
Functionality

By default, a section appears in expanded form.

Procedure

To expand or reduce the content of a section, proceed as follows:

Step	Action
1	Make sure you are in edit mode. If not, click on **Mode** → **Edit**.
2	Create a section. An expansion icon, represented by a white square, appears as illustrated below: 

Step	Action
3	Click on the section expansion icon. The section content is reduced. You can tell the section has been reduced when a "+" sign is visible in the section expansion icon, as illustrated below: 

Displaying a Section Level

Introduction

You can display up to three section levels in a program:

- By expanding the section levels one after another
- Or by manipulating the tree structure levels

Functionality

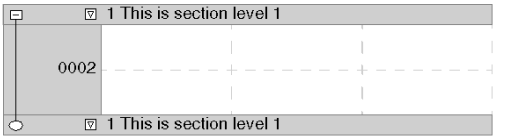
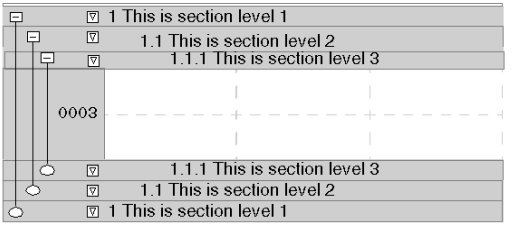
The sections and sub-sections define a tree structure with three levels.

The hierarchical level of a section is indicated by means of automatic numbering in the section header and end.

The tree structure levels are available in the grayed-out part above the first program line in the form of three numbered icons. They are used to control the expansion or reduction of sections.

Defining a Section Tree Structure

To display the section levels in a program, proceed as follows:

Step	Action
1	Make sure you are in edit mode. If not, click on **Mode** → **Edit**.
2	Define a section. The first level of the section tree structure appears in the header. 
3	Define two other section levels. The three levels of the section tree structure appear in their respective headers. 

Displaying the Tree Structure Levels

To display the tree structure levels in a program, proceed as follows:

Step	Action
1	Create several sections and sub-sections in a program.
2	Click on icon **1** for example. The content of the level 1 sections expands (or reduces). The – (or +) sign appears against the top line of each expanded (or reduced) section.  In the above example, the level 1 sections include a level 2 section, and two sections of level 2 and 3 respectively. You can only pass from one tree structure level to another if the higher (or lower) level of the relevant section level has been expanded (or reduced).

Inserting and Deleting Comments

Introduction

Any program line or section can have a comment added to it.

The comment appears on one line.

The text can contain 3 lines of 60 characters maximum.

You can create up to 400 comment lines per diagram.

Inserting a Comment

To insert a comment, proceed as follows:

Step	Action
1	Make sure you are in edit mode. If not, click on **Mode** → **Edit**.
2	Select the line or section where you wish to insert a comment.
3	Right-click and select the **Insert a comment** option. A comment line appears in gray above your selection. Double-click on the comment line to open the **Parameter setting for comments** text entry window.
4	Enter the comment text. The comment is displayed in the center of the line.

Deleting a Comment

To delete a comment line, proceed as follows:

Step	Action
1	Make sure you are in edit mode. If not, click on **Mode** → **Edit**.
2	Select the comment line to be deleted.
3	Click on **Edit** → **Delete**, or right-click on the line, then select the **Delete** option, or use the **<Del>** key on your keyboard.

Copy/Paste Function

Introduction

You can perform copy/paste operations on the cells, lines or even sections of a program.

NOTE: The MS Windows keyboard shortcuts **<Ctrl> + C** (to copy) and **<Ctrl> + V** (to paste) are also available.

Copying/Pasting Cells

It is possible to copy/paste cells, or a selection of cells. The content of copied cells overwrites the content of the cells into which the data is pasted.

Copy/paste cell operations are performed in the same zone: from contact to contact and coil to coil.

When copying a selection of linked cells, you must take account of the size of the selection and the type of instructions they contain. A selection of 3 cells on 3 lines cannot be copied from column 9, for example.

If a mistake is made, Logipam issues the following messages:

- `Paste operation could not be carried out because it would have resulted in a contact/coil inconsistency.`
  In this case, make sure that you are pasting a coil onto a coil or a contact onto a contact.
- `Paste operation could not be carried out because the targeted selection is not compatible with the source.`
  In this case, make sure that the target is a cell on the edit grid and not a line or section.
- `Paste operation could not be carried out because it would have resulted in overrunning the limits of the diagram.`
  In this case, make sure that the selection is compatible with the program zone where it is to be copied.

Procedure

To copy and/or paste a variable of the same type, proceed as follows:

Step	Action
1	Make sure you are in edit mode. If not, click on **Mode** → **Edit**.
2	In the program, select a cell containing, for example, the memorized bit `MS005`.
3	Click on **Edit** → **Copy**.
4	Point to another cell.

Step	Action
5	Click on **Edit** → **Paste**. The resource, `MS005`, is displayed in the cell. You can also right-click on the resource `MS005`, then select the **Copy** option then **Paste**.

Copying/Pasting Lines

It is possible to copy/paste a line, or a selection of lines. The content of the copied line is inserted before the line where the data is pasted.

To copy/paste lines, make sure that the maximum number of lines does not exceed the 1400 lines allowed in a program.

If a mistake is made, Logipam issues the following messages:

- `Paste operation could not be carried out because it would have resulted in overrunning the maximum number of lines.` In this case, reorganize the program.
- `Paste operation could not be carried out because it would have resulted in overrunning the maximum number of comment lines.`
  The number of comment lines must not exceed 400 lines: reorganize the comments.

Procedure

To copy and/or paste lines, proceed as follows:

Step	Action
1	Make sure you are in edit mode. If not, click on **Mode** → **Edit**.
2	In the program, select one or more lines.
3	Click on **Edit** → **Copy**.
4	Point to another line.
5	Click on **Edit** → **Paste**. The copied line is inserted before the selected line.

Copying/Pasting Sections

It is possible to copy/paste a section, or a selection of sections, or a selection of lines and sections. The content of the copied selection is inserted before the line where the data is pasted.

To copy/paste sections, make sure that the maximum number of sections does not exceed the 100 sections allowed and that the section nesting level does not exceed the 3 levels allowed.

If a mistake is made, Logipam issues the following messages:

- `Paste operation could not be carried out because it would have resulted in overrunning the maximum number of sections.`
  In this case, reorganize the program.
- `Paste operation could not be carried out because it would have resulted in exceeding the authorized depth in the sections.`
  The number of nesting levels is more than three: reorganize the program.

Procedure

To copy and/or paste sections, proceed as follows:

Step	Action
1	Make sure you are in edit mode. If not, click on **Mode** → **Edit**.
2	In the program, select one or more sections. **Note**: If the section has been reduced, expand it so that you are sure of the nesting level and the number of lines.
3	Click on **Edit** → **Copy**.
4	Point to another section: either after a section if you wish to insert the selected section(s), or selecting the section(s) you wish to overwrite.
5	Click on **Edit** → **Paste**. The content of the selection is inserted before the selected line.

Search/Replace Function

Introduction

You can search for occurrences of a variable or object in a program and replace them.

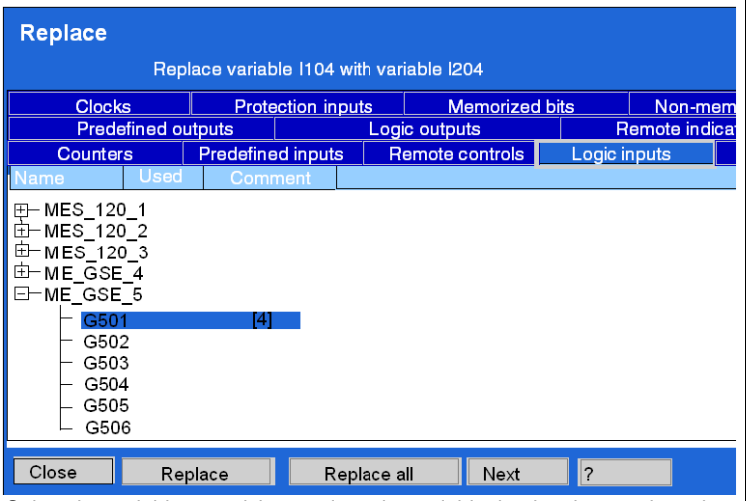
Searching for a Variable

To search for occurrences of a variable or an object, proceed as follows:

Step	Action
1	In the program, select the variable you wish to search for.
2	• Press **F3** to search for the next occurrence of the variable • Press **<Ctrl> + F3** to search for the previous occurrence of the variable

Replacing a Variable

To replace one variable with another, proceed as follows:

Step	Description
1	Select the variable to be modified in the program (I104, for example). The selected variable cell turns yellow.
2	Select **Edit** → **Replace** to access the replace window or use the contextual menu. The **Replace** window opens.  Select the variable you wish to replace the variable that has been selected (I204, for example). The replacement to be made is indicated under the dialog box title. In the illustration, variable I104 that has already been selected in the program is to be replaced with variable I204.
3	Click on **Replace** to replace the selected occurrence or on **Replace all** to replace all occurrences in the program. Click on **Next** to display the next occurrence. Click on **Close** to close the dialog box again.

Creating and Deleting Connections

Introduction

The variables and objects are linked by horizontal or vertical connections.

Creating a Connection

To create a connection between two variables, proceed as follows:

Step	Action
1	Make sure you are in edit mode. If not, click on **Mode** → **Edit**.
2	Create two elements in the program; for example, internal bits `MS006` and `MX006` in the contact 2 and 4 columns of line 3.
3	Position the mouse cursor between the two elements, i.e. in the cell at the intersection of line 3 and column 3, on the dotted line.
4	Click on the cell dotted line. A horizontal connection appears as a solid line in the cell. **Remark**: If the connection affects several cells, click on the left-hand mouse button and hold it down as you move the cursor over the dotted lines.

Deleting a Connection

To delete a connection between two variables, proceed as follows:

Step	Action
1	Make sure you are in edit mode. If not, click on **Mode** → **Edit**.
2	Select a connection; for example, the solid line of the horizontal connection created on line `3` of the contact `3` column.
3	Position the mouse cursor on the solid line of the connection.
4	Click on the line. The connection disappears from the cell and the line becomes a dotted line.

Displaying Variables and Objects by Identifier or by Name

Introduction

You can display a variable or an object in a program by its name if you have allocated it one, or by its identifier.

The selected display mode is active in all Logipam windows.

Displaying by Identifier

To display variables by their identifier, proceed as follows:

Step	Action
1	Go to the Logipam startup window.
2	Click on the **Display** → **Variables** → **By identifier** tab. The option is checked. The variable appears in the program or on the different views with its identifier (I101, for example).

Displaying by Name

To display variables by their name, proceed as follows:

Step	Action
1	Go to the Logipam startup window.
2	Click on the **Display** → **Variables** → **By name** tab. The **By name** option is checked. The variable appears in the program or on the different views with the name allocated earlier using the parameter-setting window.

Viewing a Variable Comment

Introduction

You can view the comment associated with a variable or an object in a program from several screens.

Description

The comments viewing modes are as follows:

- From the **Variables** tab in the **Entry** menu
- From the variable or object parameter-setting window
- As an info balloon from the cell where the variable is positioned on the diagram
  or
  As an info balloon from the variables and objects bar selection window

Saving a Program

Introduction

To save a program, use:

- The **Save** option to save a program that has already been saved.
- The **Save as** option to save a new program or to save a program with another name.

Description

When saving a programming project, Logipam creates:

- A Logipam file (`.bft`).
- A file containing the modifiable program text for each language (`.fr`, `.sp` and `.uk`)

Modification Not Saved (*)

The asterisk "*", placed after the file name in the title bar, indicates that you have modified a program but you have not saved the modifications.

Checking Program Entry

Introduction

Logipam continuously validates the program lines to check that they have been entered correctly and that all the elements are linked.

Logipam provides a report of the consistency errors detected.

Functionality

In both edit mode and simulation mode, the toolbar contains a consistency button. This button is:

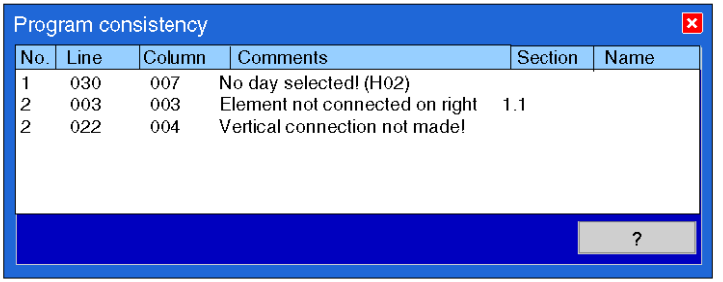
- **Green** when the program contains no errors
- **Red** when the program contains errors

Clicking on this button or using the **List of inconsistencies** option of the **Display** menu accesses the data entry error report.

You can switch to **Simulation** mode without correcting the errors. However, it is advisable to run this check and to correct the errors to avoid any problem during simulation.

Accessing the Consistency Report

To display the result of the consistency check, proceed as follows:

Step	Action
1	Select **Display** → **List of inconsistencies** to start program validation or click on the following icon which can be found on the toolbar:  The **Program consistency** window opens. The **Nothing to report** message appears if no error is detected; otherwise, the errors are listed as shown in the figure below:  For each error, the report shows the following information: ● **No.**: Indicates the message number. ● **Line**: Indicates the number of the line where the error is located. ● **Column**: Indicates the number of the column where the error is located. ● **Comments**: Provides a description of the error. ● **Section**: Indicates the number of the section where the error is located. ● **Name**: Indicates the name of the section where the error is located if you have allocated a name to the section.
2	In the window, double-click on the line with the error. Logipam goes directly to the zone containing the error in the program.
3	Correct the errors. **Note**: The error message is automatically deleted from the list once the error has been corrected.

Error and Resolution Messages

The table below lists the possible errors and the actions to be taken to resolve them.

Message	Description
Element not connected on left	A connection or an element is not connected on the left-hand side. Check the program on the line indicated.
Element not connected on right	A connection or an element is not connected on the right-hand side. Check the program on the line indicated.
Vertical connection not made	A vertical connection is not made to a neighboring cell. Check the program on the line indicated.
Time delay equal to zero	No value has been assigned to the timer indicated. Double-click on the timer to access its parameter-setting window, enter the appropriate value and save the modifications.
Timer used several times	The same timer has been activated several times. Check the object values.
Period equal to zero	The indicated clock period value is zero. Double-click on the clock to access its parameter-setting window, enter the appropriate value and save the modifications.
Number of pulses equal to zero	The indicated number of clock pulses value is zero. Double-click on the clock to access its parameter-setting window, enter the appropriate value and save the modifications.
No day selected	No day has been selected for the indicated clock. Double-click on the clock to access its parameter-setting window, enter the appropriate value and save the modifications.

Program Checking

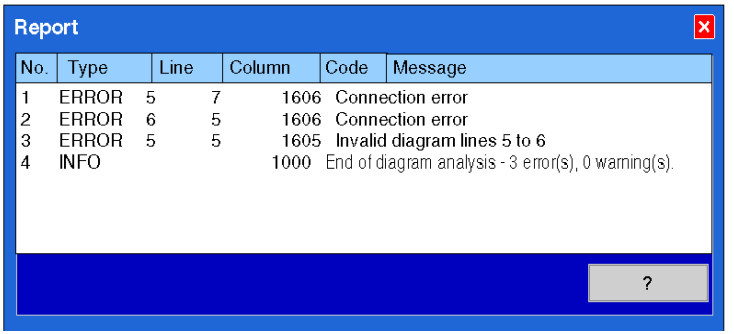
Introduction

When switching to **Simulation** mode or exporting to SFT2841, Logipam generates the executable program and checks the program, to ensure that it does not contain any errors.

The result of this checking can also be viewed at any time using the **Report** option of the **Display** menu.

Accessing the Report

To access the report, proceed as follows:

Step	Action
1	Select **Display** → **Report**. The **Report** window opens.  For each error, the report shows the following information: • **No.**: Indicates the message number. • **Type**: Determines whether it is an error, a warning or a data item. • **Line**: Indicates the number of the line where the error is located. • **Column**: Indicates the number of the column where the error is located. • **Code**: Indicates the line of code where the error is located. • **Message**: Describes the error type.
2	To correct the errors, double-click on the line with the error. Logipam goes directly to the zone containing the error in the program.

Error and Resolution Messages

The table below lists the possible errors and the actions to be taken to resolve them.

Type	Message	Description
Error	Too big Logipam source file	Indicates that the number of comments is too high.
	Invalid file	Indicates that the imported file is invalid.
	Timer x used several times	Indicates that the same timer is used in several cells.
	Not enough memory	Indicates that the available memory has been exceeded; the program is too large.
	Too many contacts	Indicates that the maximum number of contacts and coils allowed has been exceeded (5000 contacts).
	Invalid diagram line n	An error has been detected on line **n** of the program.
	Invalid diagram lines n to z	An error has been detected on a zone from line **n** to line **z** of the program.
	Connection error	A connection error has not been corrected on a program zone.
Warning	Assigned variable x never used.	A value has been defined for a variable but it is not used in the program. Check the program.
	Used variable x never assigned.	A variable is used in the program but no value has been defined. Check the program.
Information	Memory used x.x%	Information on the amount of memory used by the program compared to the amount available.
	Contacts and coils used	Information on the number of contacts and coils used in the diagram compared to the 5000 available.

Printing the Logipam File

Introduction

All or part of the program can be printed to form or add to the study file.

Selecting the Elements to be Printed

From Logipam, you can print the following elements:

- The cover page
  The cover page presents the general program characteristics such as the date, the name and the functional description.
- The program listing in Ladder language
  This listing corresponds to the diagram as it appears on the data entry grid in **Edit** mode.
- The summary list of variables and objects
  This list consists of:
  - Timers
  - Clocks
  - Counters
  - Logic I/O
  - Protection I/O
  - Remote controls
  - Predefined I/O
  - Internal bits
  - Remote indications
  - Outputs to matrix
- Page headers and footers
  The page headers and footers contain three print zones for each page header and footer (left, central and right).
  You can select the following options:
  - A logo
  - The page number
  - The number of pages
  - The date
  - The time
  - The document name

Printing the Selected Elements

To print the selected elements, proceed as follows:

Step	Action
1	Click on **File** → **Printer configuration**. The **Printer configuration** dialog box opens.
2	Check the box(es) corresponding to the elements you wish to print.
3	Click on **Headings and foot of pages** to define the elements to be printed in the page header and footer.
4	Click on **Pagination** to define the page setup elements. Select the page size and orientation. The diagram size adapts so that it fits within the width of a page whichever orientation is selected. The cover page is always printed in "portrait" format.
5	Click **OK** to confirm the selection.
6	Select**File** → **Print** to print the file.

Simulating a Program

6

Scope of This Chapter

This chapter describes how simulation works and the operations to be performed to simulate a program.

What Is in This Chapter?

This chapter contains the following topics:

Simulation Operating Principles

Introduction

Simulation mode is used to simulate a program to check that it is operating correctly.

Description of the Operating Principle

The simulator executes the program in continuous or step-by-step mode. One step represents a Sepam cycle time, i.e. 14 ms.

The program is executed in the same way as on Sepam. On each cycle, the input variables are read, the program logic is executed and the output variables are modified.

You need to modify the variables in order to simulate the environment external to the program (logic inputs, protection outputs, predefined inputs, etc.).

Interface Refresh Time

The refresh time corresponds to the time that elapses between two updates of the screen when the program is simulated continuously.

The refresh time is between 100 ms and 2000 ms. You can modify it from the options window by clicking on **File** → **Options**.

Executing Simulation

Introduction

The operations for executing simulation are as follows:

- Initialize variables (**Init)**
  Resetting of variables that have not been saved. Saved variables are not modified. This corresponds to Sepam operation on energization.
- Execute in continuous mode (**Run**)
  Performing simulation in continuous mode. The variables can be modified.
- Stop simulation (**Stop**)
  The inputs can be activated. In the event of modification, a step must be executed again so that they are taken into account.
- Execute one step (**One step**)
  Executes a step of a 14 ms cycle.
- Set the clock (**Date and time**)

Each function can be activated from an icon in the toolbar. For more information on the simulation mode toolbar, see *Familiarization With the Software, page 59*.

Displaying States and Behaviors

During simulation, the screen display is automatically refreshed so that it constantly reflects the state of the variables and objects.

During simulation, the contacts, coils and connections take on the following colors:

- Red: represents passing state
- Black: represents non-passing state

The name of the variables and objects takes the following colors:

- Red: represents state 1
- Black: represents state 0

Performing Simulation

Switching to simulation mode automatically places the simulator in **Stop** state and initializes simulation.

Run starts simulation in continuous mode. Execution in continuous mode corresponds to successive execution of several simulation steps.

Stop interrupts simulation immediately, this state can only be exited with the aid of the **Run** command.

When simulation is in **Run** mode, you must stop it using the **Stop** button before initializing the variables.

Execution of one simulation step corresponds to actual execution of one Sepam cycle.

Exiting simulation mode causes simulation to stop automatically.

Modifying the Status of Variables During Simulation

Introduction

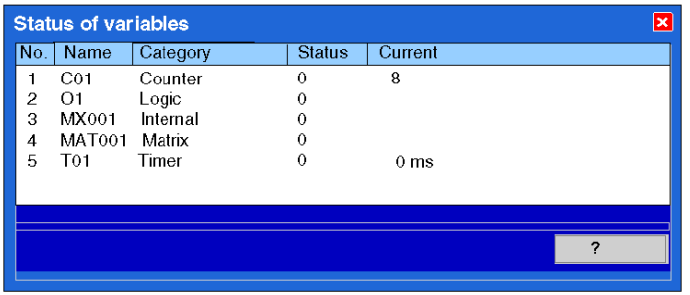
In **Simulation** mode, Logipam can be used to modify variables and monitor their status using the **Status of variables** window.

Description of the Status of Variables Window

To access the **Status of variables** window, click on the following icon in the toolbar:

The window opens. By default this window is empty, and you have to select the variables you wish to modify.

The **Status of variables** window looks like this:



It shows the following information:

- **No.**: Indicates the number of the variable in the list.
- **Name**: Indicates the name of the variable (or the identifier, if this display mode has been chosen).
- **Category**: Indicates the type of variable.
- **Status**: Indicates the current value of the variable, 0 or 1.
- **Current**: Indicates the value of the counters and timers.

Selecting and Deleting the Variables to be Monitored

To list variables in the **Status of variables** window, click on the variable from the simulation screen and drag it onto the window. Repeat this operation for all variables you wish to monitor or modify.

To delete variables in the **Status of variables** window, click on the variable and then right-click. The delete option appears; click to confirm deletion.

Modifying Variables and Objects

To modify variables and objects from the **Status of variables** window, proceed as follows:

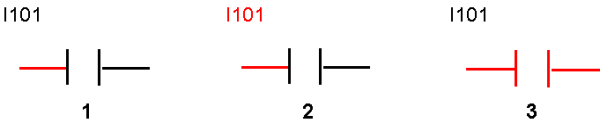
- For variables
  - Double-click on the status value to change it to the opposite status.
  - Double-click on the variable in the **No., Name** or **Category** column to open the corresponding parameter-setting window.
- For objects:
  - Double-click on the current value of the object to modify it.
  - Double-click on the object in the **No.**, **Name** or **Category** column to open the corresponding parameter-setting window.

It is also possible to toggle the status of a variable by double-clicking on the variable from the programming grid when the cursor takes the form of a red cross-hair.

The status of the contacts associated with a variable is not evaluated at the time of clicking: only executing simulation updates the contact. In continuous mode, the contacts are evaluated immediately. When simulation is inactive, the contacts are only updated when a step is executed or continuous mode is restarted.

Examples of Behavior of Variables During Modification

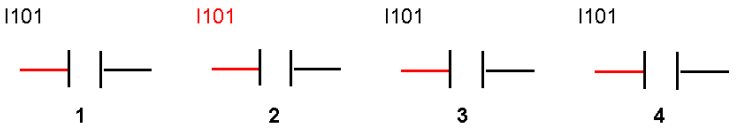
The following example illustrates the change from status 0 to 1 of a contact while simulation is stopped.



This modification is made as follows:

Ref. No.	Description
1	I101=0, the contact is open.
2	If you double-click, the variable identifier turns red. I101=1, the contact is closed. When the variable has been modified but its status has not been updated on the diagram because simulation is stopped, it is highlighted in yellow.
3	If you launch an execution step, the current flows and the whole contact turns red.

The example below illustrates the change of status from 0 to 1 then back to 0.



This modification is made as follows:

Ref. No.	Description
1	I101=0, the contact is open.
2	If you double-click, the variable identifier turns red. I101=1, the contact is closed. When the variable has been modified but its status has not been updated on the diagram because simulation is stopped, it is highlighted in yellow.
3	If you double-click, the variable identifier turns black. I101=0, the contact is open again.
4	If you launch an execution step, the current does not flow.

Setting the Simulation Clock

Introduction

The simulation date allows you to modify the current time of the simulation as well as the time at initialization.

- Current time
  The current time is the present value of the simulation clock, which can be modified during simulation or with the simulation stopped.
- Time at initialization
  The time at initialization is the value that the current time takes on initialization, which can only be modified with the simulation stopped.

You need to set the simulation initialization time. This information is saved and then positioned each time simulation is initialized.

Simulation Time Operation

The simulation day and time change as follows:

- The time display format shows the day of the week, hour, minute, second and millisecond.
- A week starts on Monday at 00h00m00s000ms and ends on Sunday at 23h59m59s999ms and continues cyclically.

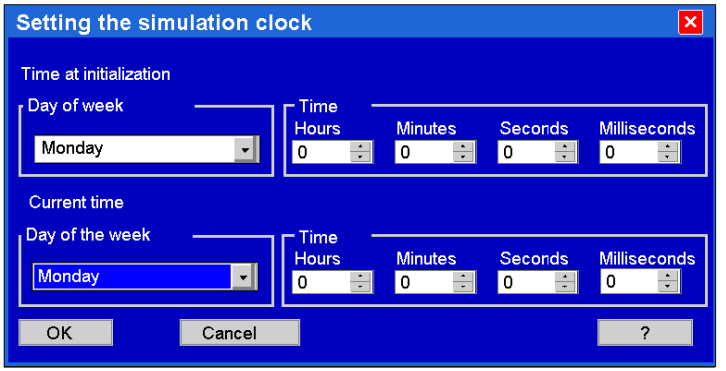
The time changes according to the simulation execution; if the simulation executes 40 Sepam cycles of 14 ms, the time changes by 560 ms.

The initial simulation time is set by default at Monday 00h00h00s000ms. The simulation time only acts as a reference for clocks that send their pulse on defined days and at defined times.

The simulation time does not affect the timers; the latter base their behavior on the simulation execution.

Setting the Simulation Initialization Time and the Current Time

To define the simulation initialization time, proceed as follows:

Step	Action
1	Make sure you are in Simulation mode. If not, click on **Mode** → **Simulation**.
2	Click on **Simulation** → **Date and time**. The **Setting the simulation clock** dialog box opens. 
3	Select the **Time at initialization** or **Current time** drop-down menu, and choose the **Day of the week**.
4	Define the time and click **OK**.

Operating a Program

7

Scope of This Chapter

This chapter describes all the operations involved in loading a Logipam program into Sepam, operating and maintaining it.

What Is in This Chapter?

This chapter contains the following topics:

Exporting a Program for SFT2841

Introduction

In Logipam, the **Export for SFT2841** function in the **Execution** menu can be used to generate the file which will then be used by SFT2841 and loaded onto Sepam.

Functionality

The tested program is analyzed during the export phase. A `.bin` file is generated.

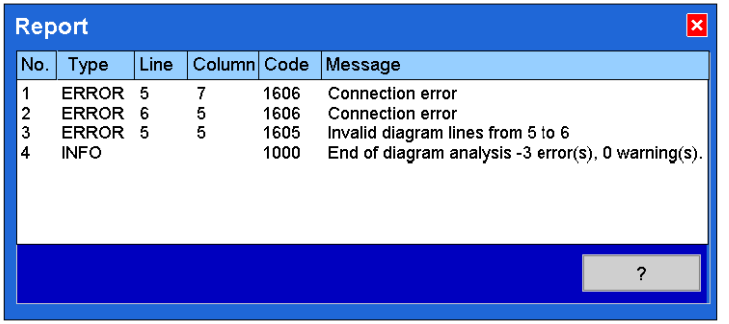
The generation process performs the following operations:

- Conversion of the program to binary data that can be executed on Sepam
- Generation of the interface for SFT2841

The program can only be exported to a saved project.

Procedure

To export a project, proceed as follows:

Step	Action
1	Select **Execution** → **Export for SFT2841**. **If the program contains no errors**: the export occurs automatically and the `xxxx.bin` program file is created in the `C:\Program Files\SCHNEIDER\SFT2841\Logipam` directory. **If the program contains errors**: the **Report** window opens to indicate the errors, as illustrated in the figure below:  The report shows the following information: ● **N°**: Message number ● **Type**: Error or warning ● **Line**: Number of the line where the error is located ● **Column**: Number of the column where the error is located ● **Code**: Line of the diagram where the error is located ● **Message**: Description of the error type
2	To correct the errors, double-click on the line with the error. Logipam goes directly to the zone in the diagram containing the error. When the errors are corrected, Logipam goes ahead with exporting the file. The `xxxx.bin` program file is created in the `C:\Program Files\SCHNEIDER\SFT2841\Logipam` directory.

Export Error Messages

The error messages that appear during export are the same as those that appear when switching to **Simulation** mode. For more information, see *Edit and Simulation Operating Modes, page 74*.

Operating Principles for a Logipam Program

At a Glance

A Logipam program can run in three different ways:

- With SFT2841 in disconnected mode to configure Sepam and its Logipam program
- With SFT2841 in connected mode to configure Sepam or to check that the Logipam program is running correctly
- With Sepam in operating mode

The information provided in this document relates only to the use of Logipam programs in SFT2841 with a view to their setup. For more information on the functions of SFT2841, see the documentation for the SFT2841 software.

Configuring Sepam and a Logipam Program

Sepam is configured with its new Logipam program in connected or disconnected mode with SFT2841.

The steps involved in setup are as follows:

- Select the exported Logipam program for SFT2841
  Remark: The logic inputs must first have been assigned to SFT2841 in order for correct operation.
- Activate the Logipam program
- Set the configuration bits and the object parameters of the program
- Configure the logic outputs, LEDs, and messages with the SFT2841 software, using the control matrix.
- Load the Sepam configuration containing the Logipam program onto Sepam.

Checking Execution of the Logipam Program in Connected Mode

It is possible to check execution of the Logipam program in connected mode.

To do this, the following conditions must be fulfilled:

- The Logipam file and the SFT2841 parameters and protection settings file are available.
- SFT2841 is connected to Sepam.

The displayed state of all the internal variables is then used to check execution of the program.

Operation from Sepam

The Sepam user-machine interface provides access to Sepam and Logipam information that can be viewed and/or modified. This information is as follows:

- Counters
  It is possible to view the current value of the counters.
- Configuration bits
  It is possible to display their state and toggle them.

For more information on the operation of Sepam, see the Sepam series 80 documentation.

Configuring Sepam and the Logipam Program on SFT2841

Introduction

The program must be selected on SFT2841 so that it can be activated and the configuration can be adjusted.

NOTE: Before using the Logipam program on SFT2841, you should ideally have created and saved the Sepam configuration and assigned the required inputs and outputs.

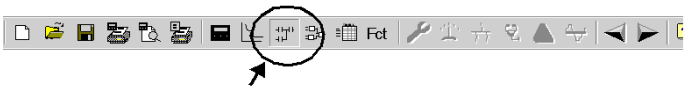
Configuration Steps

The steps involved in configuring Sepam and its Logipam program on SFT2841 are as follows:

Step	Description
1	Selection of the Logipam program.
2	Turning ON or OFF according to whether or not the program is to be executed.
3	Setting the parameters, if necessary.
4	Application of modifications: application of modifications involves loading into Sepam (in connected mode) or storing in the configuration file (in disconnected mode). In the latter case, the Logipam program will be loaded into Sepam when the configuration is loaded.

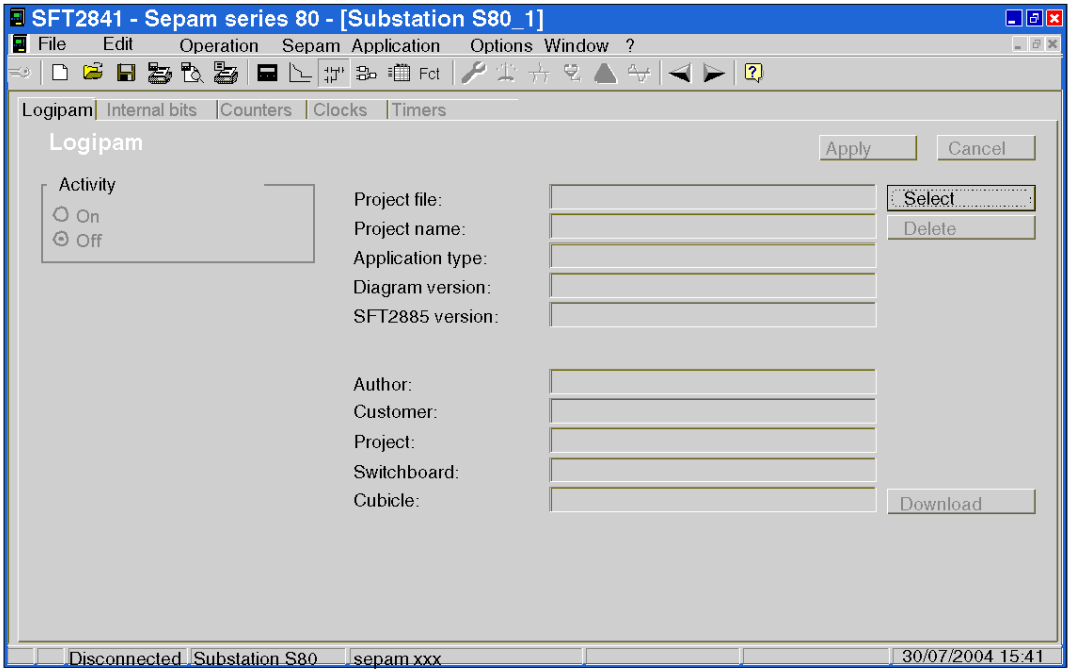
Opening the Logipam Interface in SFT2841

To open a Logipam program, proceed as follows:

Step	Action
1	In Windows Explorer, click on **Start** → **Programs** → **Schneider** → **SFT2841**. The **Connection to Sepam** dialog box opens and asks you whether you wish to: ● Connect to Sepam ● Modify the file of an application that has already been created ● Create a file
2	Select one of the options to open a Sepam series 80 project. SFT2841 opens on the **Sepam hardware** screen.
3	In the SFT2841 menu bar, click on **Application** → **Logipam settings** or select the icon indicated below in the toolbar:  **Fast access to the Logipam setting screen icon** The Logipam setting screen opens.

Description of the Logipam Setting Screen

The Logipam setting screen looks like this:



The screen contains the following options:

- Logipam tab
  Used to select and set up a Logipam program.
  The information displayed on this screen indicates the program properties defined at the time of its creation using Logipam.
  The following operations are possible from this screen:
  - **Select**
    Used to select the Logipam program for Sepam.
    SFT2841 displays the properties of the selected Logipam program.
  - **Activity**
    Used to activate or deactivate the Logipam program.
  - **Delete**
    Deletes the Logipam program from Sepam.
  - **Apply**
    Confirms the operation performed.
  - **Cancel**
    Cancels the operation performed.
  - **Download**
    Used to download the Sepam file in a format that can be read by Logipam.

- Internal Bits/Counters/Clocks/Timers tabs
  Used to view and adjust the values defined during programming with Logipam.

Selecting a Logipam Program

To select a Logipam program, proceed as follows:

Step	Action
1	In the **Logipam settings** window, click on **Select** to open the **Select a Logipam project** window. By default, SFT2841 browses the directory common to both software programs and suggests programs that have already been exported to `C:\Program Files\SCHNEIDER\SFT2841\Logipam`.
2	Select the file and click on **Open**. In the **Logipam** tab, the Logipam program properties are displayed and the function keys become active.
3	Click on **Apply** to confirm acceptance of the program. The program is now part of the Sepam configuration. The variables settings are retrieved by SFT2841 for operation, the variables tabs are active and the program can be turned on/off. All the program information allowing modification at a later date is stored in the Sepam cartridge. **Important**: If you change any settings, then return to the **Logipam** tab to apply the program again, SFT2841 asks you if you wish to return to the default settings, i.e. to the values in the Logipam program. If you answer **Yes** to this question, the settings entered on SFT2841 are replaced by the settings defined in the Logipam program.

Program Activity

To turn on the selected program and make it active, click on the **On** button in the **Activity** zone.

To turn off the selected program and make it inactive, click on the **Off** button in the **Activity** zone.

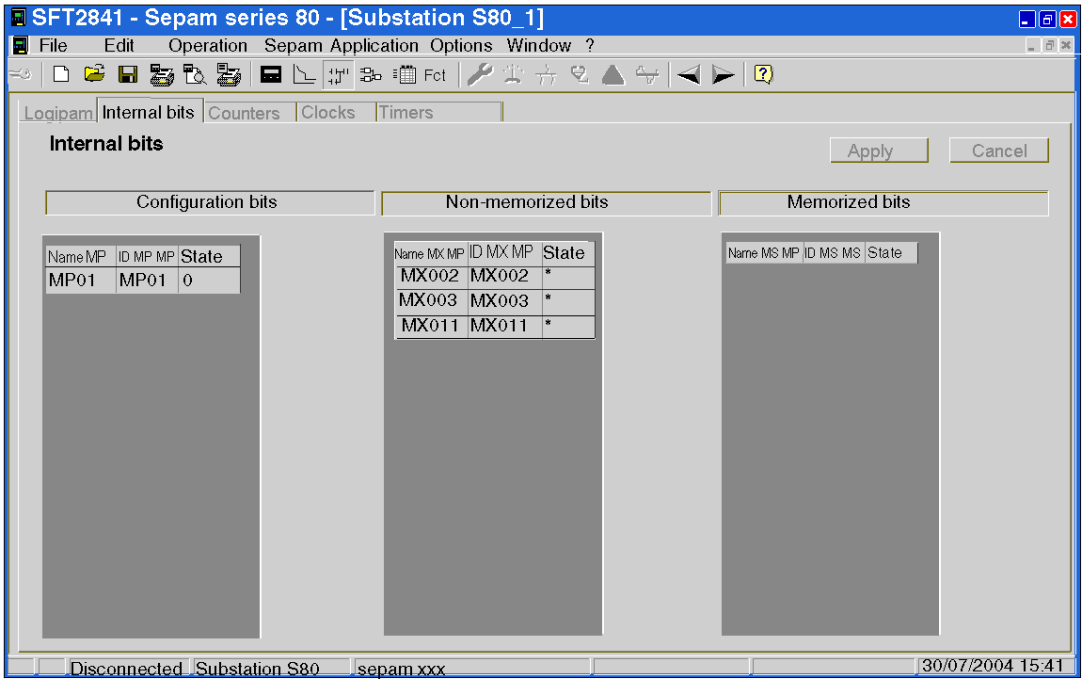
Setting Variables

The SFT2841 can be used to view internal bits and programmed objects and to set them. To do this, select the corresponding tab.

Each tab displays the current value of the parameters initialized with the default value contained in the program during selection. The default value is the one chosen when configuring the program on the SFT2885.

Setting Internal Bits

The **Internal bits** tab is used to display the bits used by the program and to toggle the state of the configuration bits if necessary.

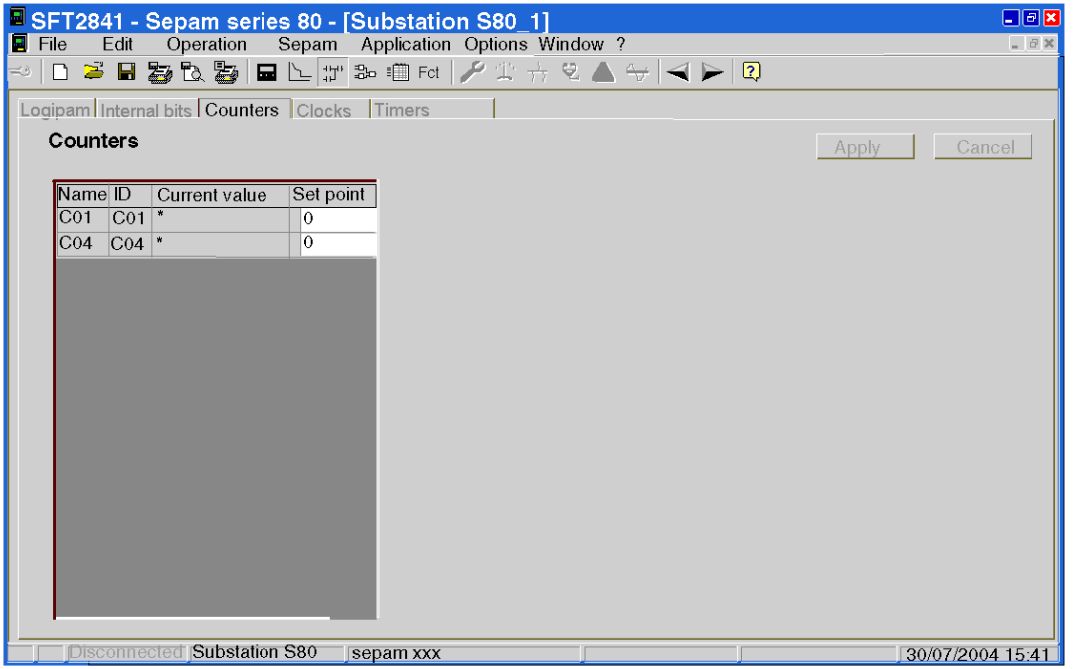


Only the state of configuration bits can be modified.

Apply the modifications before going to the next settings.

Setting Counters

The **Counters** tab is used to display the counters used by the program and to set their parameters if necessary.

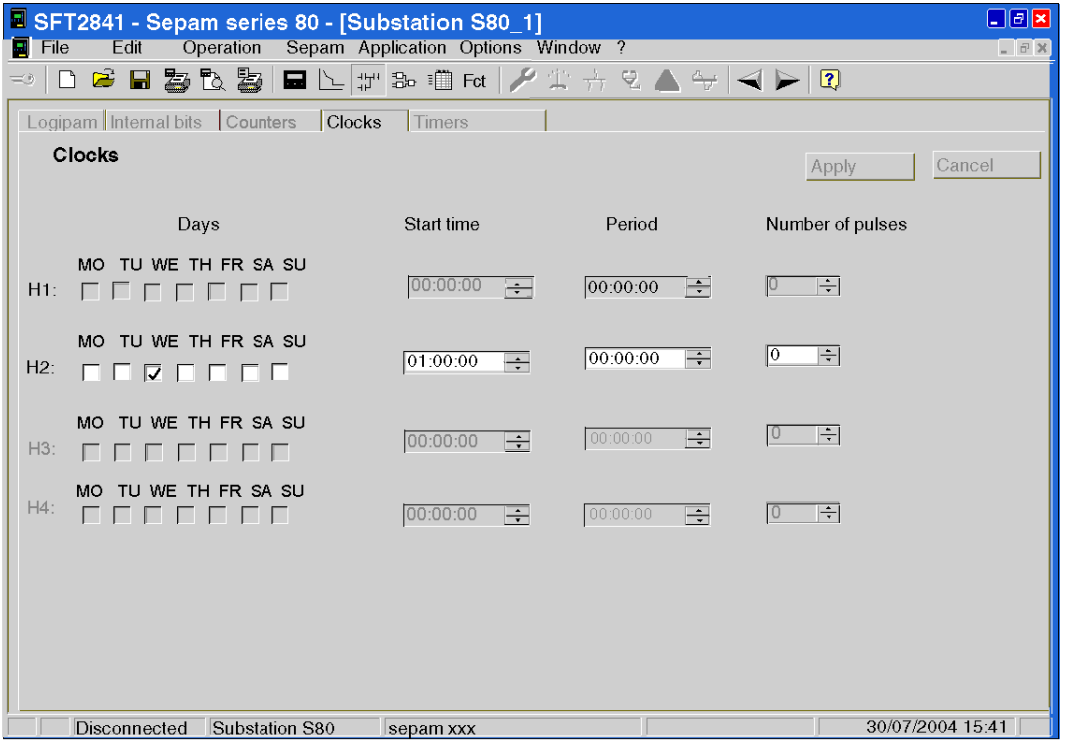


Only the set point can be modified by double-clicking on it. It must however remain within the range [0-65535].

Apply the modifications before going to the next settings.

Setting Clocks

The **Clocks** tab is used to display the clocks used by the program and to set their parameters if necessary.

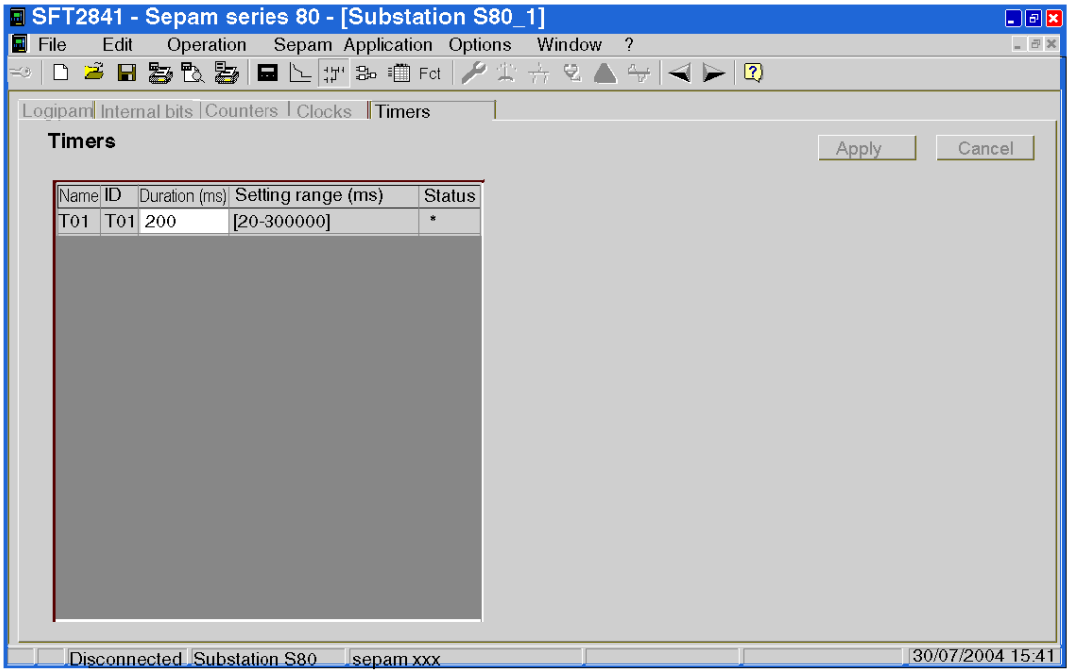


You can adjust the clock setting by clicking in the boxes relating to the days or by entering a value in the fields reserved for times, pulses and periods.

Apply the modifications before going to the next settings.

Setting Timers

The **Timers** tab is used to display the timers used by the program and to set their parameters if necessary.



Only the duration can be modified by double-clicking on it. It must however remain within the defined range.

Apply the modifications before going to the next settings.

Operation of Outputs, LEDs, and Messages Using the Control Matrix.

Introduction

The outputs, LEDs and messages are operated by means of the control matrix available in the **Application** → **Set control matrix** menu.

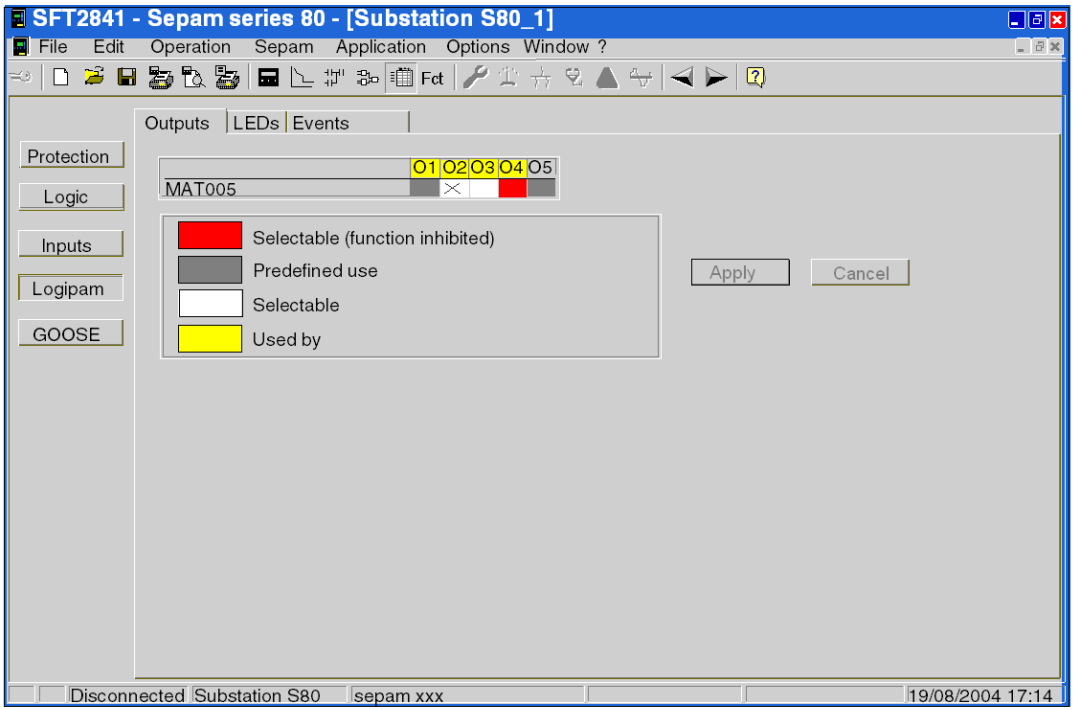
Description

The matrix is used to assign "Output to matrix" type Logipam variables to the logic outputs, to messages, to LEDs and to triggering disturbance recording.

To display the assignments linked only to Logipam variables, click on the **Logipam** button.

To assign an "Output to matrix" type Logipam variable to a Sepam logic output, click on the intersection of the variable line and the output column. A cross appears.

In the example below, variable MAT005 is assigned to logic output O2:



NOTE: The Logipam button is only available if the Logipam program has been configured, otherwise the **Equations** button appears for logic equations.

By default, the matrix displays all the possible assignments.

Each column performs an OR between all the selected lines.

NOTE: A logic output controlled by the Logipam program should not be used in the matrix. If it is, Sepam performs an OR between the value set by the matrix and the value set by Logipam before controlling the output.

The colors of the matrix boxes have the following meaning:

- **Yellow** means that the output is used by Logipam
- **Red** indicates that the output is not configured as being used
- **White** indicates that the output is configured as being used. For more information on this assignment, view the **Logic I/O** tab of the **Sepam** → **General settings** option.

Checking the Program Execution in Connected Mode

Introduction

The program execution is checked from SFT2841.

Having first connected SFT2841 to Sepam, you can check the program execution with the aid of the Logipam programming file and the SFT2841 settings file.

Principles

This check consists of:

- Checking the state of the program variables
- Adjusting the setting of the program objects if necessary, with or without any interruption of Sepam

Checking the State of Variables

Checking the state of the internal program variables is a way of checking the program execution or identifying the cause of non-operation of a function programmed with Logipam.

To display the name of the variables and objects, and also the program properties in the tabs, SFT2841 needs the Logipam program file (`.bin` file) in the `C:\Windows\Program Files\Schneider\Logipam` directory. If the file is missing, it can be downloaded by clicking on the **Download** button on the Logipam screen.

Program Object Setting Mode

The objects can be set in two ways:

- Without interruption of Sepam
  The values are modified directly in Sepam.
  Download the SFT2841 parameters and settings file to the PC to keep an up-to-date file on the PC.
- With interruption of Sepam
  The values are modified in the SFT2841 parameters and settings file.
  The SFT2841 file is then loaded into Sepam.

Appendices

Appendices

A

Scope of This Chapter

This chapter lists the predefined variables, grouped by function, and the keyboard shortcuts.

What Is in This Chapter?

This chapter contains the following topics:

List of Predefined Inputs

Predefined Inputs

The predefined inputs available in Logipam are as follows:

Function	Description	Name	Identifier
Latching acknowledgement	Reset order given	V_RESET_ORD	SI068
	Reset key on advanced UMI or mimic-based UMI	V_KEY_RESET	SI074
Alarm	Clear order given	V_CLEAR_ORD	SI069
	Clear key on advanced UMI or mimic-based UMI	V_KEY_CLEAR	SI075
Genset shutdown	Genset shutdown order given - Genset shutdown function output	V_SHUTDN_ORD	SI035
	Genset shutdown enabled	V_SHUTDN_ON	SI082
Automatic transfer	Coupling synchronization failure	V_TIESYNCFAIL	SI053
	Tripping order given via automatic transfer - Automatic transfer output	V_AT_TRIPPING	SI054
	Tripping order given via 2/3, 1/2 logic - Automatic transfer output	V_2/3_TRIPPING	SI055
	Circuit breaker closing ready - Automatic transfer output	V_CLOSE_EN	SI056
	Circuit breaker closing order given - Automatic transfer output	V_CLOSE_NO_ORD	SI057
	Coupling closing ready - Automatic transfer output	V_TIE_CLOSE_EN	SI058
	Coupling closing order given - Automatic transfer output	V_TIE_CLOSING	SI059
	Coupling tripping order given - Automatic transfer output	V_TIE_OPENING	SI060
	Automatic transfer enabled	V_TRANSF_ON	SI086

Function	Description	Name	Identifier
Switchgear control	Tripping order given - Switchgear control function output	V_TRIPPED	SI017
	Closing order given - Switchgear control function output	V_CLOSED	SI018
	Closing inhibited - Switchgear control function output	V_CLOSE_INHIBITED	SI019
	Contactor order - Switchgear control function output	V_CONTACTOR	SI020
	Request to close with synchronism check in progress	V_SYNC_INPROC	SI039
	Closing with synchronism check failed - Voltage difference too high	V_NOSYNC_DU	SI040
	Closing with synchronism check failed - Phase difference too high	V_NOSYNC_DPHI	SI041
	Closing with synchronism check failed - Frequency difference too high	V_NOSYNC_DF	SI042
	Stop closing with synchronism check	V_SYNC_STOP	SI043
	Switchgear control enabled	V_SWCTRL_ON	SI078
	Synchro-check enabled	V_SYNC_ON	SI079
	Closing with synchronism check failed - Synchronism duration too short	V_NOSYNC	SI089
	Closing with synchro-check successful	V_SYNC_OK	SI090
Capacitor step control	Step tripping order given - Step tripping function output	V_STP1_TRIPPING to V_STP4_TRIPPING	SI044 to SI047
	Step matching fault	V_STP1_CTRLFLT to V_STP4_CTRLFLT	SI048 to SI051
	Step closing order given - Step control function output	V_STP1_CLOSING to V_STP4_CLOSING	SI091 to SI094
	Step control enabled	V_BANK_ON	SI080
Communication	Remote protection settings enabled	V_REM_SETTING	SI066
Phase rotation direction check	Reverse phase rotation, main channels	V_REVERSEPHASE	SI023
	Reverse phase rotation, additional channels	V_REVERSEPHASE2	SI098
Load shedding	Load shedding order given - Load shed function output	V_LOADSH_ORD	SI034
	Load shedding enabled	V_LOADSH_ON	SI084
De-excitation	De-excitation order given - De-excitation function output	V_DE-EXCIT_ORD	SI036
	De-excitation enabled	V_DE-EXICT_ON	SI083

Function	Description	Name	Identifier
TC/switchgear position discrepancy	TC/switchgear position discrepancy	V_TC/CBDISCR	SI021
Data log	Data log function active	V_DLG_ACTIVED	SI104
Power factor	Inductive or resistive power factor	V_PFACTOR	SI076
	Positive power factor	V_PFACTOR_SIGN	SI077
Mimic-based UMI	Mimic-based UMI output	V_MIMIC_OUT_1 to V_MIMIC_OUT_16	SI001 to SI016
	Key in Local position on mimic-based UMI	V_MIMIC_LOCA	SI070
	Key in Remote position on mimic-based UMI	V_MIMIC_REMOTE	SI071
	Key in Test position on mimic-based UMI	V_MIMIC_TEST	SI072
Temperature measurement	MET 1 module RTD fault	V_RTD2FAULT	SI025
	MET 2 module RTD fault	V_RTD2FAULT	SI026
Disturbance recording	Disturbance recording in progress	V_OPG_TRIGGED	SI097
	Disturbance recording enabled	V_OPG_ON	SI087
	Disturbance recording function inhibited	V_OPG_INHIBITED	SI033
Protections	Pick-up	V_PICKUP	SI027
	Summary of pick-up of overcurrent protections	V_PICKUP_OC	SI028
	Drop-out	V_DROPOUT	SI029
Restart	Restart command given - Restart function output	V_RESTARTING	SI052
	Restart enabled	V_RESTART_ON	SI085
Motor start report	Motor start report in progress	V_MSR_TRIGGED	SI103
Logic discrimination	Blocking No. sent - Logic discrimination function output	V_LOGDSC_BL1 to V_LOGDSC_BL2	SI030 to SI031
	Tripping command given by logic discrimination - Logic discrimination function output	V_LOGDSC_TRIP	SI032
	Logic discrimination enabled	V_LOGDSC_ON	SI081
Phase rotation direction	Rotation direction command 123 active	V_PHASE_DIR	SI099
	Rotation direction command 132 active	V_PHASE_INV	SI100
	Phase rotation direction command matching fault for longer than 2 seconds	V_PHASE_DISC	SI101
	Phase rotation direction 123 (0) or 132 (1)	V_PHASE_ACTIVE	SI102
Power supply monitoring	Auxiliary voltage low threshold overshoot	V_VAUX_LOW	SI037
	Auxiliary voltage high threshold overshoot	V_VAUX_HIGH	SI038
	Auxiliary voltage monitoring enabled	V_VAUX_ON	SI067

Function	Description	Name	Identifier
Cumulative breaking current monitoring	Cumulative breaking current threshold overshoot	V_MAXBRKCUR	SI024
Trip circuit supervision	Closing circuit fault - Closing circuit supervision function output	V_CCS	SI095
Trip circuit supervision	Trip circuit fault - Trip circuit supervision (TCS) function output	V_TCS	SI096
Open/close order supervision	Control fault (a circuit breaker activation order has not been executed)	V_CTRLFAULT	SI022
System	First Sepam cycle following energization	V_INIT	SI061
	Sepam in settings or parameter-setting mode	V_ADJUST	SI062
	Sepam partial fault	V_PARTIALFAULT	SI063
	Group A enabled	V_GROUPA	SI064
	Group B enabled	V_GROUPB	SI065
	Square wave period 1 minute	V_ONEMINUT	SI073
	The SFT2841 is connected to Sepam	V_SFT2841	SI088

List of Predefined Outputs

Predefined Outputs

The predefined outputs available in Logipam are as follows:

Function	Comments	Name	Identifier
Latching acknowledgement	Reset Sepam	V_RESET	SO045
	Inhibition of Sepam local reset	V_INHIB_RESET_LOCAL	SO047
Alarm	Clear Sepam messages	V_CLEAR	SO046
Genset shutdown	Genset shutdown order - Genset shutdown function input	V_SHUTDOWN	SO048
Automatic transfer	Transfer on fault order - Automatic transfer input	V_TRANS_ON_FLT	SO059
	Transfer stop order - Automatic transfer input	V_TRANS_STOP	SO060
	Upstream voltage present - Automatic transfer input	V_TRANS_V_EN	SO077
Switchgear control	Circuit breaker tripping order - Switchgear control function input	V_TRIPCB	SO042
	Circuit breaker closing order - Switchgear control function input	V_CLOSECB	SO043
	Inhibit circuit breaker closing order - Switchgear control function input	V_INHIBCLOSE	SO044
	Close without synchro-check authorization - Switchgear control function input	V_CLOSE_NOCTR	SO050
Step x closing control	Step control function input	V_CLOSE_STP1 to V_CLOSE_STP4	SO055 to SO058
Capacitor step control	Step tripping order - Step control function input	V_TRIP_STP1 to V_TRIP_STP4	SO051 to SO054
De-excitation	De-excitation order - De-excitation function input	V_DE-EXCITATION	SO049
Data log (DLG)	Starting a Data log	V_DLG_START	SO079
Mimic-based UMI	Mimic-based UMI inputs	V_MIMIC_IN_1 to V_MIMIC_IN_16	SO061 to SO076
Peak demand current	Reset peak demand and demand phase current	V_RST_CURRENTS	SO005
Peak demand power	Reset peak demand active and reactive power	V_RST_POWERS	SO006

Function	Comments	Name	Identifier
Disturbance recording	Inhibition of disturbance recording function	V_OPG_INHIBIT	SO004
	Validation of disturbance recording function	V_OPG_VALID	SO001
	Manual triggering of disturbance recording data	V_OPG_MANUAL	SO002
	Recording a logic variable in the disturbance recording	V_FLAGREC2 to FLAGREC15	SO007 to SO020
	Recording a logic variable in the disturbance recording	V_FLAGREC	SO041
Motor start report (MSR)	Starting a Motor start report	V_MSR_START	SO078
Thermal overload protection (49RMS)	Thermal capacity used reset	V_RST_THERMAL	SO003

List of Protection Inputs

Protection Inputs

The protection inputs are as follows:

Protection	Identifier	X	Associated Information
12	P12_X_101 P12_X_113	1 to 2	Reset Inhibition
14	P14_X_101 P14_X_113	1 to 2	Reset Inhibition
21B	P21B_1_101 P21B_1_113		Reset Inhibition
24	P24_X_101 P24_X_113	1 to 2	Reset Inhibition
27	P27_X_101 P27_X_113	1 to 4	Reset Inhibition
27D	P27D_X_101 P27D_X_103	1 to 2	Reset Inhibition
27R	P27R_X_101 P27R_X_103	1 to 2	Reset Inhibition
27TN/64G2	P27TN/64G2_X_101 P27TN/64G2_X_113	1 to 2	Reset Inhibition
32P	P32P_X_101 P32P_X_113	1 to 2	Reset Inhibition
32Q	P32Q_1_101 P32Q_1_113		Reset Inhibition
37	P37_1_101 P37_1_113		Reset Inhibition
37P	P37P_X_101 P37P_X_113	1 to 2	Reset Inhibition
38/49T	P38/49T_X_101 P38/49T_X_113	1 to 16	Reset Inhibition
40	P40_1_101 P40_1_113		Reset Inhibition
46	P46_X_101 P46_X_113	1 to 2	Reset Inhibition
47	P47_X_101 P47_X_113	1 to 2	Reset Inhibition
48/51LR	P48/51LR_1_101 P48/51LR_1_102 P48/51LR_1_113		Reset Motor re-acceleration Inhibition

Protection	Identifier	X	Associated Information
49RMS	P49RMS_1_101 P49RMS_1_113		Reset Inhibition
50/27	P50/27_1_101 P50/27_1_113		Reset Inhibition
50/51	P50/51_X_101 P50/51_X_113	1 to 8	Reset Inhibition
50BF	P50BF_1_101 P50BF_1_107 P50BF_1_113 P50BF_1_119		Reset Start 50BF Inhibition Circuit breaker closed
50N/51N	P50N/51N_X_101 P50N/51N_X_113	1 to 8	Reset Inhibition
51C	P51C_X_101 P51C_X_113	1 to 8	Reset Inhibition
50V/51V	P50V/51V_X_101 P50V/51V_X_113	1 to 2	Reset Inhibition
59	P59_X_101 P59_X_113	1 to 4	Reset Inhibition
59N	P59N_X_101 P59N_X_113	1 to 2	Reset Inhibition
64REF	P64REF_X_101 P64REF_X_113	1 to 2	Reset Inhibition
66	P66_1_101 P66_1_102 P66_1_113		Reset Motor re-acceleration Inhibition
67	P67_X_101 P67_X_113	1 to 2	Reset Inhibition
67N	P67N_X_101 P67N_X_113	1 to 2	Reset Inhibition
78PS	P78PS_1_101 P78PS_1_113		Reset Inhibition
79	P79_1_113		Inhibition
81H	P81H_X_101 P81H_X_113	1 to 2	Reset Inhibition
81L	P81L_X_101 P81L_X_113	1 to 4	Reset Inhibition
81R	P81R_X_101 P81R_X_113	1 to 2	Reset Inhibition
87M	P87M_1_101 P87M_1_113		Reset Inhibition

Protection	Identifier	X	Associated Information
87T	P87T_1_101 P87T_1_113 P87T_1_118		Reset Inhibition Restraint on closing
CTS	PCTS_X_113	1 to 2	Inhibition
VTS	PVTS_X_101 PVTS_X_103 PVTS_X_113 PVTS_X_117	1 to 2	Reset Phase VT fault Inhibition Voltage presence

List of Protection Outputs

Protection Outputs

The protection outputs are as follows:

Protection	Identifier	X	Associated Information
12	P12_X_1 P12_X_3 P12_X_16	1 to 2	Pick-up Protection output Protection inhibited
14	P14_X_1 P14_X_3 P14_X_16 P14_X_36	1 to 2	Pick-up Protection output Protection inhibited Zero speed
21B	P21B_1_1 P21B_1_3 P21B_1_16		Pick-up Protection output Protection inhibited
24	P24_X_1 P24_X_3 P24_X_16	1 to 2	Pick-up Protection output Protection inhibited
25	P25_1_46 P25_1_47 P25_1_49 P25_1_50 P25_1_51 P25_1_52 P25_1_53		Synchro-check No voltage check Phase difference check Frequency difference check Voltage difference check No Usync1 No Usync2
27	P27_X_1 P27_X_3 P27_X_7 P27_X_8 P27_X_9 P27_X_16 P27_X_23 P27_X_24 P27_X_25 P27_X_26 P27_X_27 P27_X_28	1 to 4	Pick-up Protection output Phase 1 fault Phase 2 fault Phase 3 fault Protection inhibited Phase 1 instantaneous Phase 2 instantaneous Phase 3 instantaneous Phase 1 delayed Phase 2 delayed Phase 3 delayed
27D	P27D_1_1 P27D_1_3 P27D_1_16	1 to 2	Pick-up Protection output Protection inhibited
27R	P27R_1_1 P27R_1_3 P27R_1_16	1 to 2	Pick-up Protection output Protection inhibited

Protection	Identifier	X	Associated Information
27TN/64G2	P27TN/64G2_X_3 P27TN/64G2_X_16 P27TN/64G2_X_23	1 to 2	Protection output Protection inhibited Instantaneous
32P	P32P_X_1 P32P_X_3 P32P_X_16 P32P_X_19 P32P_X_20	1 to 2	Pick-up Protection output Protection inhibited Positive reactive power Negative reactive power
32Q	P32Q_1_1 P32Q_1_3 P32Q_1_16 P32Q_1_54 P32Q_1_55		Pick-up Protection output Protection inhibited Positive reactive power Negative reactive power
37	P37_1_1 P37_1_3 P37_1_16		Pick-up Protection output Protection inhibited
37P	P37P_1_1 P37P_1_3 P37P_1_16	1 to 2	Pick-up Protection output Protection inhibited
38/49T	P38/49T_X_3 P38/49T_X_10 P38/49T_X_12 P38/49T_X_16	1 to 16	Protection output Alarm RTD fault Protection inhibited
40	P40_1_1 P40_1_3 P40_1_16 P40_1_23		Pick-up Protection output Protection inhibited Circle 1
46	P46_X_1 P46_X_3 P46_X_16	1 to 2	Pick-up Protection output Protection inhibited
47	P47_X_1 P47_X_3 P47_X_16	1 to 2	Pick-up Protection output Protection inhibited
48/51LR	P48/51LR_1_3 P48/51LR_1_13 P48/51LR_1_14 P48/51LR_1_15 P48/51LR_1_16 P48/51LR_1_22		Protection output Locked rotor Excessive starting time Locked rotor on start Protection inhibited Starting in progress

Protection	Identifier	X	Associated Information
49RMS	P49RMS_1_3 P49RMS_1_10 P49RMS_1_11 P49RMS_1_16 P49RMS_1_18 P49RMS_1_22 P49RMS_1_38		Protection output Alarm Inhibit closing Protection inhibited Hot state Inhibit thermal overload Zero rotor speed
50/27	P50/27_1_1 P50/27_1_16 P50/27_1_35		Protection output Protection inhibited 50-27 protection ready
50/51	P50/51_X_1 P50/51_X_3 P50/51_X_4 P50/51_X_7 P50/51_X_8 P50/51_X_9 P50/51_X_16	1 to 8	Pick-up Protection output Drop-out Phase 1 fault Phase 2 fault Phase 3 fault Protection inhibited
50BF	P50BF_1_1 P50BF_1_3 P50BF_1_16		Pick-up Protection output Protection inhibited
50N/51N	P50N/51N_X_1 P50N/51N_X_3 P50N/51N_X_4 P50N/51N_X_16 P50N/51N_3_56	1 to 8	Pick-up Protection output Drop-out Protection inhibited 15 A set point
51C	P51C_X_1 P51C_X_3 P51C_X_16	1 to 8	Pick-up Protection output Protection inhibited
50V/51V	P50V/51V_X_1 P50V/51V_X_3 P50V/51V_X_4 P50V/51V_X_7 P50V/51V_X_8 P50V/51V_X_9 P50V/51V_X_16	1 to 2	Pick-up Protection output Drop-out Phase 1 fault Phase 2 fault Phase 3 fault Protection inhibited

Protection	Identifier	X	Associated Information
59	P59_X_1 P59_X_3 P59_X_7 P59_X_8 P59_X_9 P59_X_16 P59_X_23 P59_X_24 P59_X_25 P59_X_26 P59_X_27 P59_X_28	1 to 4	Pick-up Protection output Phase 1 fault Phase 2 fault Phase 3 fault Protection inhibited Phase 1 instantaneous Phase 2 instantaneous Phase 3 instantaneous Phase 1 delayed Phase 2 delayed Phase 3 delayed
59N	P59N_X_1 P59N_X_3 P59N_X_16	1 to 2	Pick-up Protection output Protection inhibited
64REF	P64REF_X_1 P64REF_X_3 P64REF_X_16	1 to 2	Pick-up Protection output Protection inhibited
66	P66_1_3 P66_1_16 P66_1_29 P66_1_30 P66_1_31		Protection output Protection inhibited Stop/start inhibit Startup total reached Consecutive startups reached
67	P67_X_1 P67_X_3 P67_X_4 P67_X_6 P67_X_7 P67_X_8 P67_X_9 P67_X_16 P67_X_21 P67_X_36 P67_X_37	1 to 2	Pick-up Protection output Drop-out Instantaneous output reverse zone Phase 1 fault Phase 2 fault Phase 3 fault Protection inhibited Instantaneous output at 0.8 Is Delayed 1/3 before latching Delayed 2/3 before latching
67N	P67N_X_1 P67N_X_3 P67N_X_4 P67N_X_6 P67N_X_16 P67N_X_21	1 to 2	Pick-up Protection output Drop-out Instantaneous output reverse zone Protection inhibited Instantaneous output at 0.8 Is0
78PS	P78PS_1_1 P78PS_1_3 P78PS_1_16		Pick-up Protection output Protection inhibited

Protection	Identifier	X	Associated Information
79	P79_1_201 P79_1_202 P79_1_203 P79_1_204 P79_1_205 P79_1_211 P79_1_212 P79_1_213 P79_1_214		Recloser in service Recloser ready Recloser successful Permanent trip Closing by reclosing indication Recloser cycle 1 Recloser cycle 2 Recloser cycle 3 Recloser cycle 4
81H	P81H_X_1 P81H_X_3 P81H_X_16	1 to 2	Pick-up Protection output Protection inhibited
81L	P81L_X_1 P81L_X_3 P81L_X_16	1 to 4	Pick-up Protection output Protection inhibited
81R	P81R_X_1 P81R_X_3 P81R_X_16 P81R_X_42 P81R_X_43 P81R_X_44 P81R_X_45	1 to 2	Pick-up Protection output Protection inhibited Voltage invalid Frequency invalid Positive rate of change Negative rate of change
87M	P87M_1_3 P87M_1_7 P87M_1_8 P87M_1_9 P87M_1_16 P87M_1_33 P87M_1_34 P87M_1_39		Protection output Phase 1 fault Phase 2 fault Phase 3 fault Protection inhibited High threshold Threshold % Sensor loss
87T	P87T_1_3 P87T_1_16 P87T_1_33 P87T_1_34 P87T_1_39 P87T_1_41		Protection output Protection inhibited High threshold Percentage-based set point Sensor loss Test mode
CTS	PCTS_X_3 PCTS_X_7 PCTS_X_8 PCTS_X_9 PCTS_X_16	1 to 2	Protection output Phase 1 fault Phase 2 fault Phase 3 fault Protection inhibited
VTS	PVTS_X_3 PVTS_X_16	1 to 2	Protection output Protection inhibited

Keyboard Shortcuts

Navigation Shortcuts in a Diagram

The navigation shortcuts available in a Logipam diagram are as follows:

Task	Keyboard shortcut
Open a new diagram	Ctrl+N
Open an existing diagram	Ctrl+O
Save a diagram	Ctrl+S
Print a diagram	Ctrl+P

Editing Shortcuts in a Diagram

The editing shortcuts available in a Logipam diagram are as follows:

Task	Keyboard shortcut
Cut a line, a section, a variable or a selection of variables from a diagram.	Ctrl+X
Copy a line, a section, a variable or a selection of variables in a diagram.	Ctrl+C
Paste a line, a section, a variable or a selection of variables in a diagram.	Ctrl+V
Delete a line, a section, a variable or a selection of variables from a diagram.	Del
Select the whole diagram.	Ctrl+A
Search for a variable or the next occurrence of a variable in a diagram.	F3
Search for a variable or the previous occurrence of a variable in a diagram.	Ctrl+F3
Replace a variable in a diagram.	Ctrl+H