PLC TERMINOLOGY AND APPLICATIONS

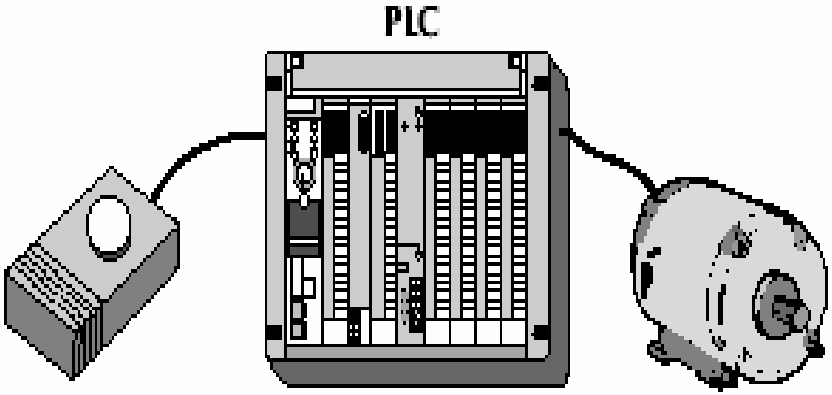
Programmable Logic Controller, also called a ***PLC*** or ***programmable controller***, is a computer-type device used to control equipment in an industrial facility.

PLC Applications: Process Control, HVAC Control, Conveyor Systems, Food Processing Machinery, Auto Assembly Lines, etc. (Regulatory Control as well as Sequential Control Systems)

SYSTEM WIRING COMPARISON

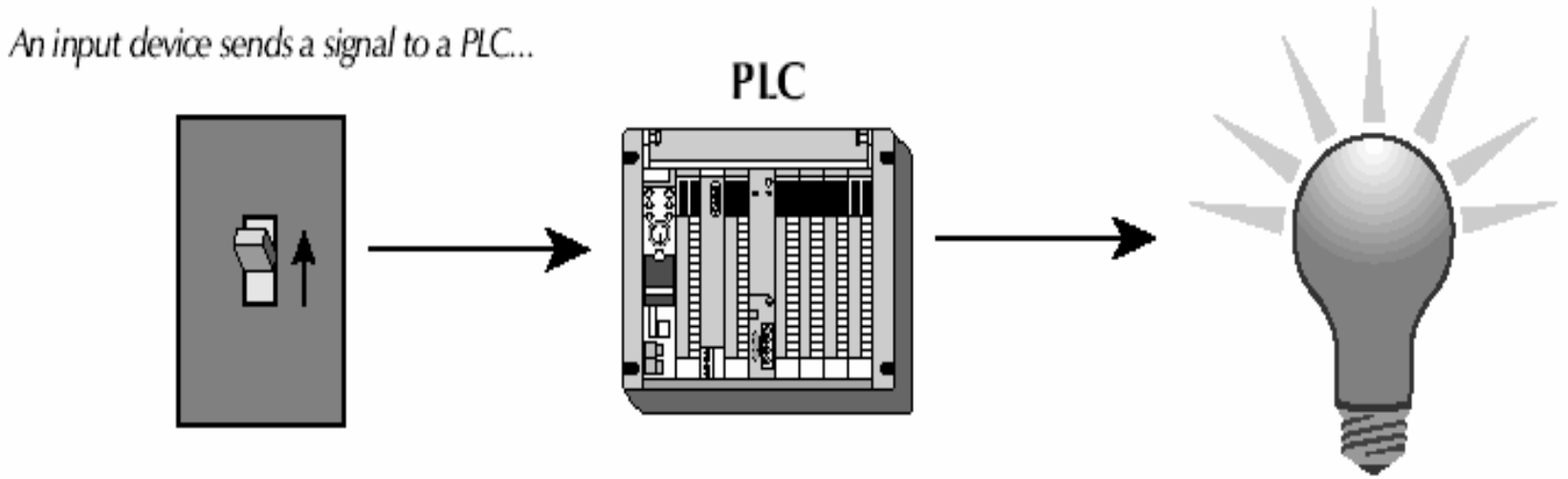


Traditionally hardwired system

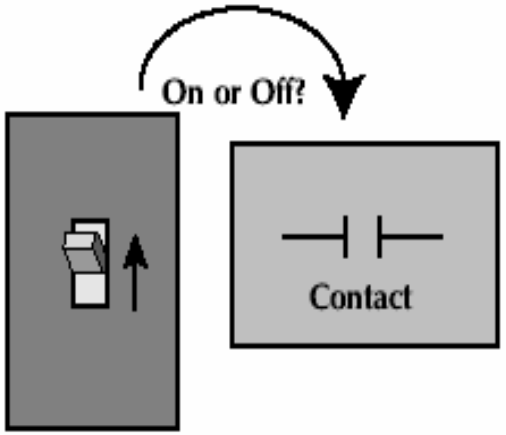


PLC System Wiring

PLC SYSTEM INPUT/OUTPUT

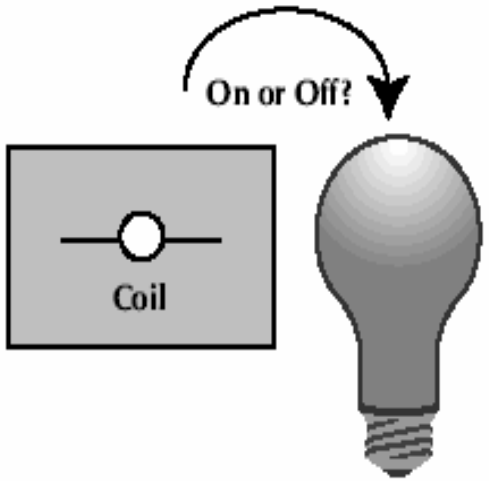


RELAY AND COIL

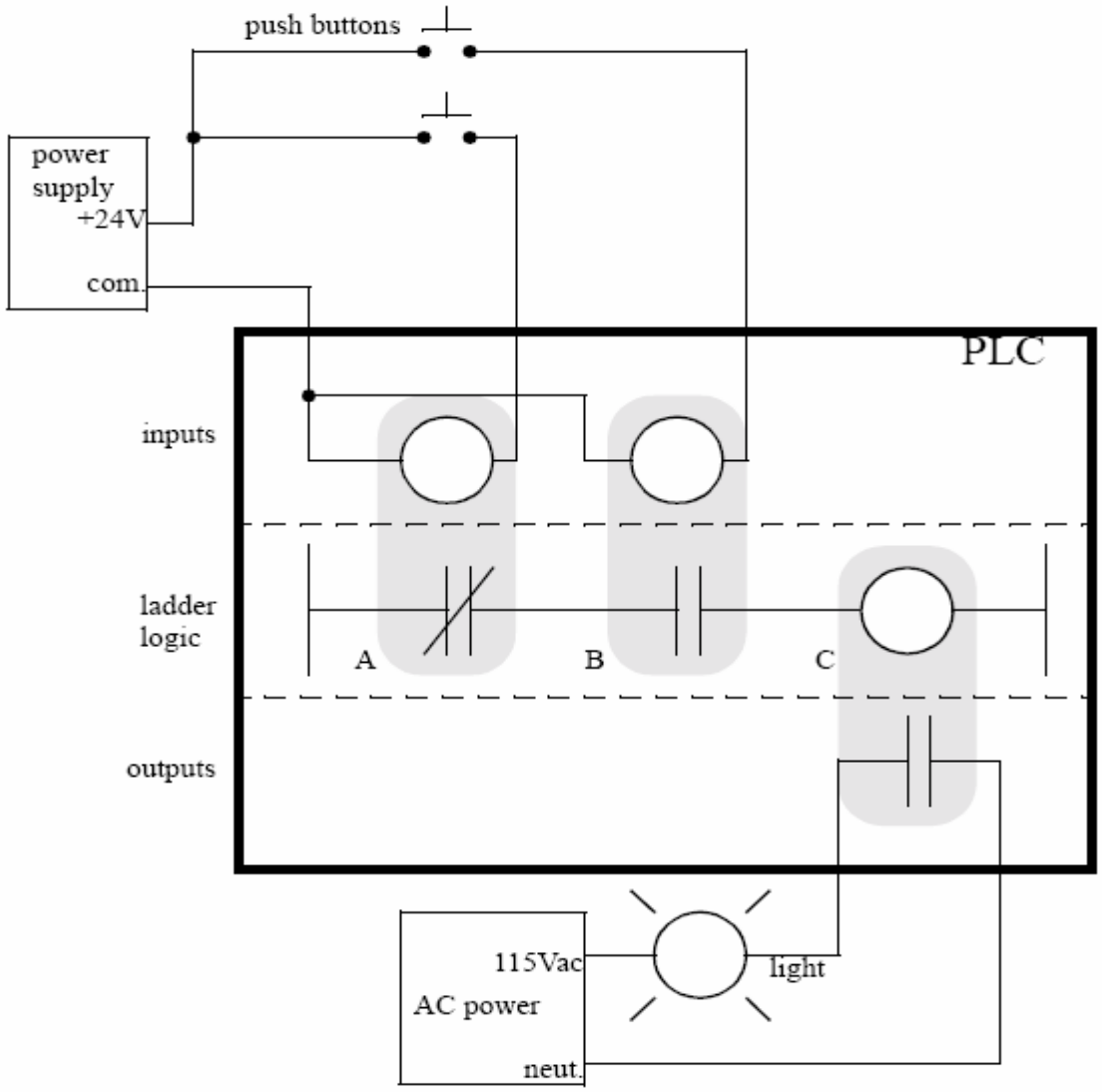


A contact is a computer code that monitors the status of an input…

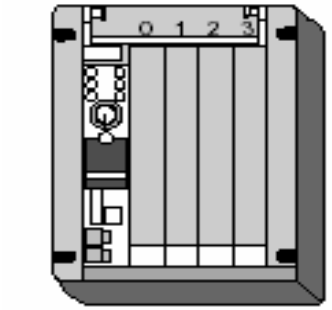
…A coil is a computer code that monitors the status of an output.



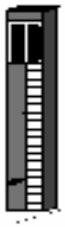
PLC SYSTEM WITH RELAYS



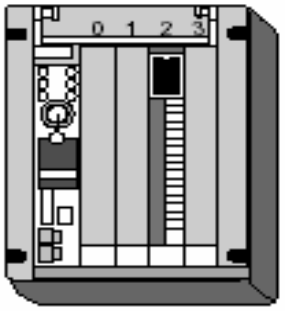
SYSTEM RACK COMPONENTS



A rack is an enclosure with slots…



…into which I/O modules…



…are installed.

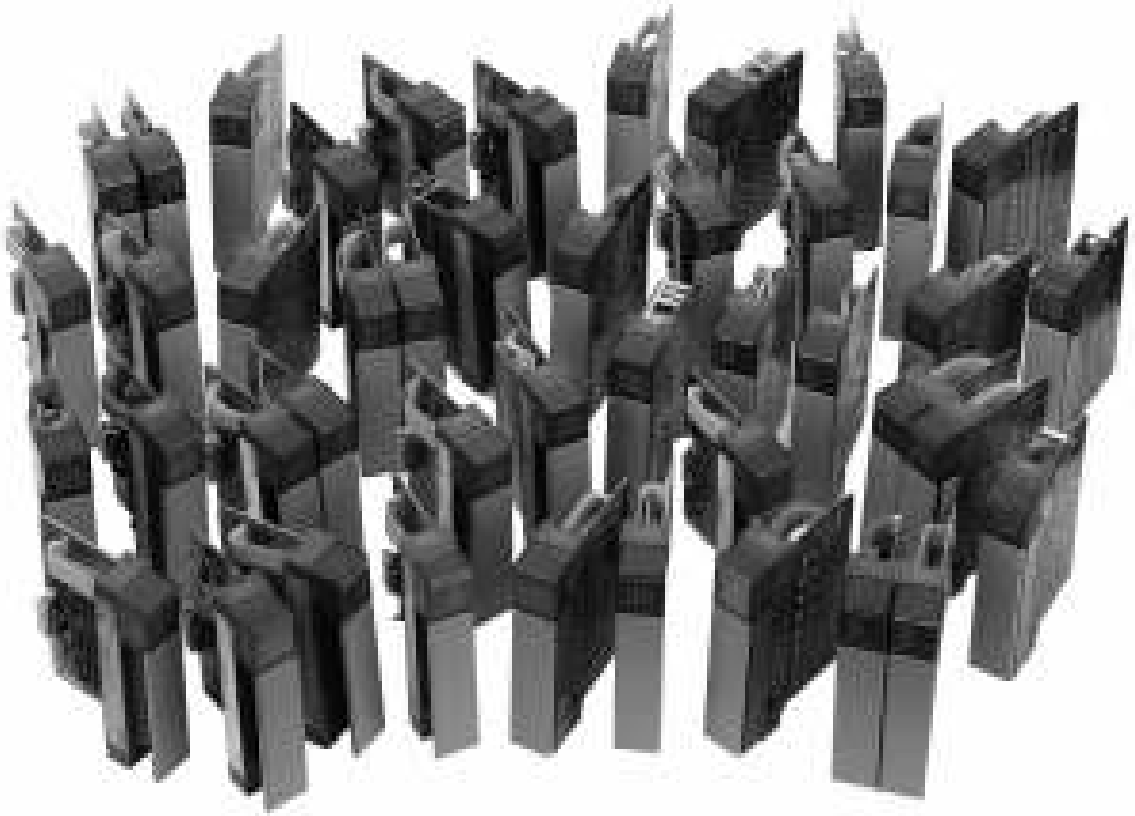


PLC MODULES

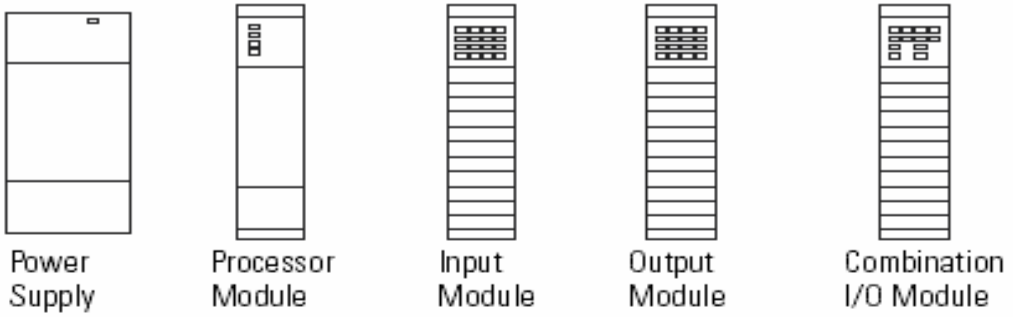
I/O MODULES

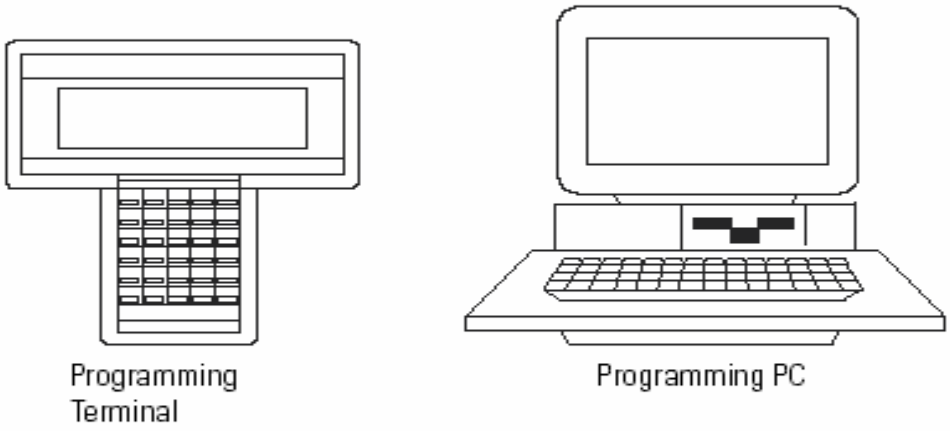
PROCESSOR

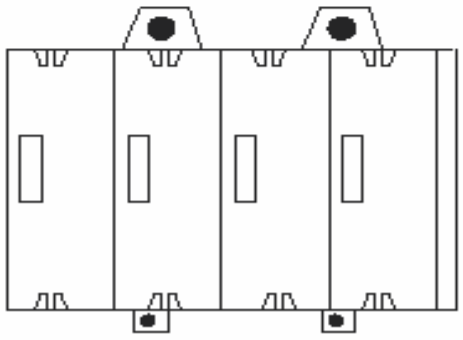


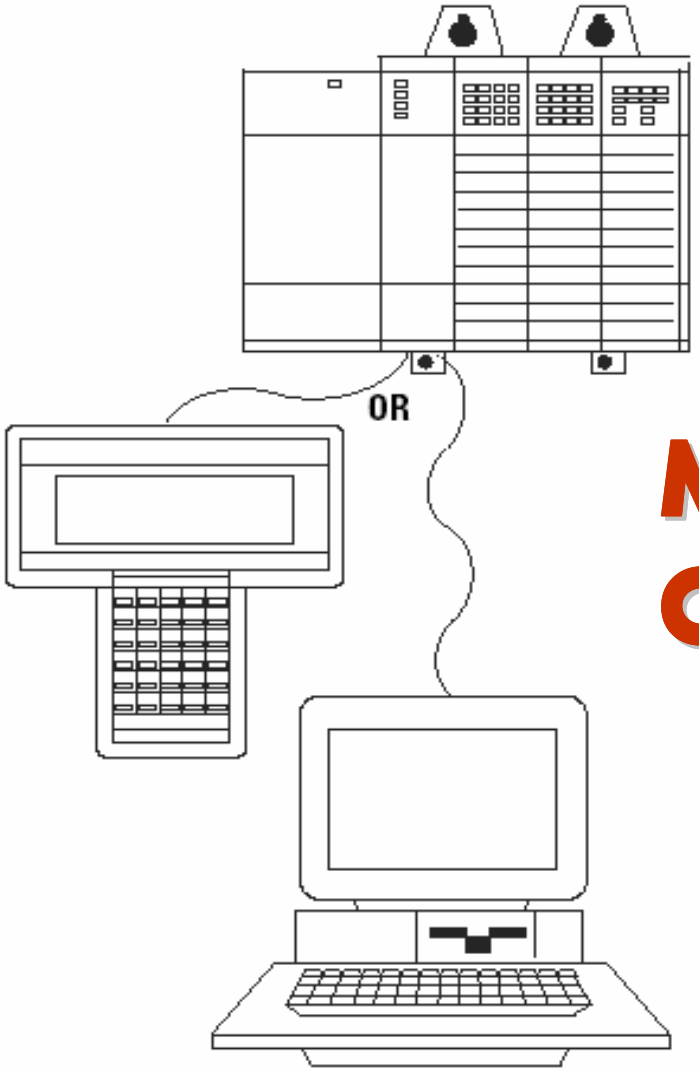


PLC MODULES


Power Supply
Processor Module
Input Module
Output Module
Combination I/O Module

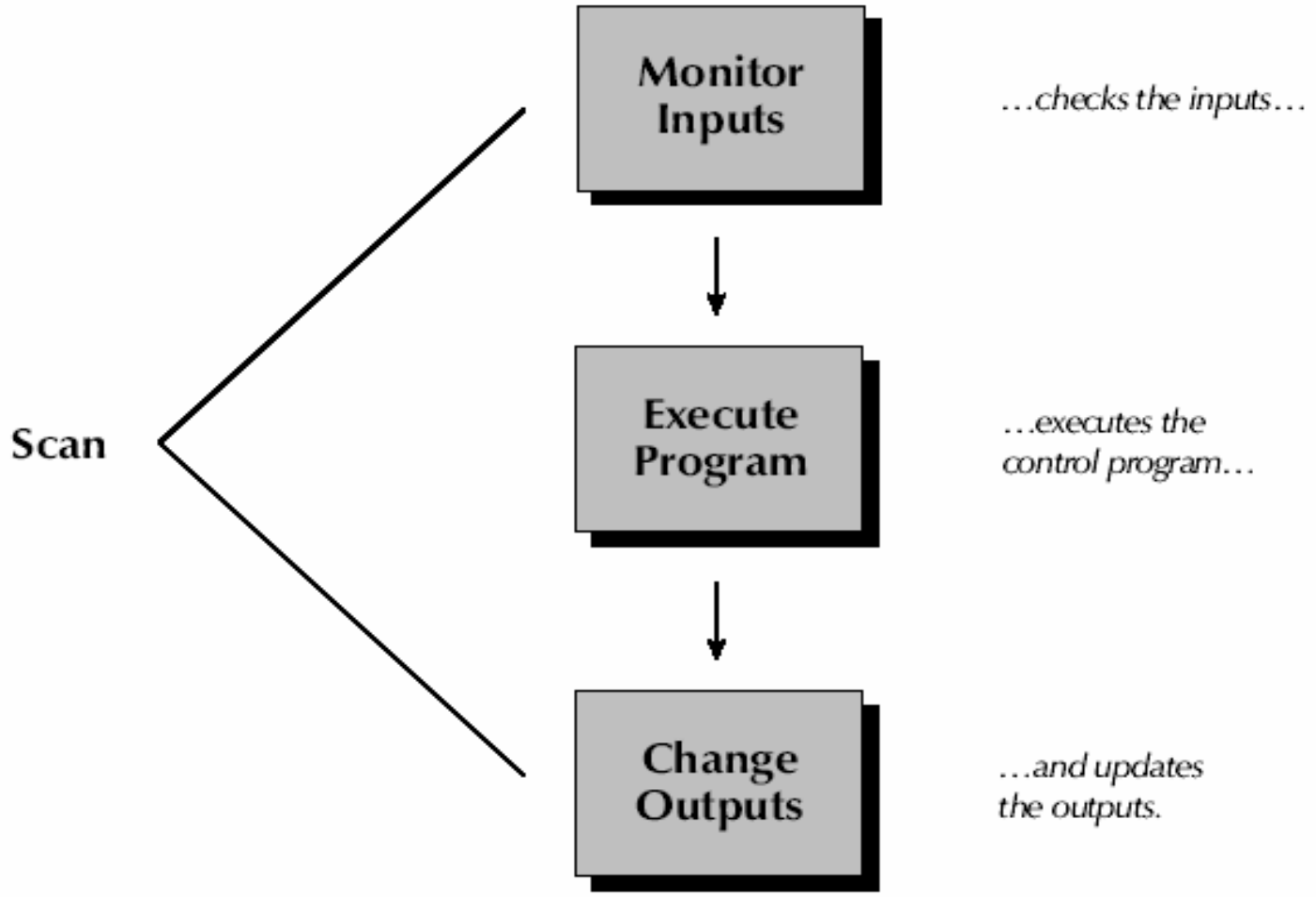

Programming Terminal
Programming PC


Chassis

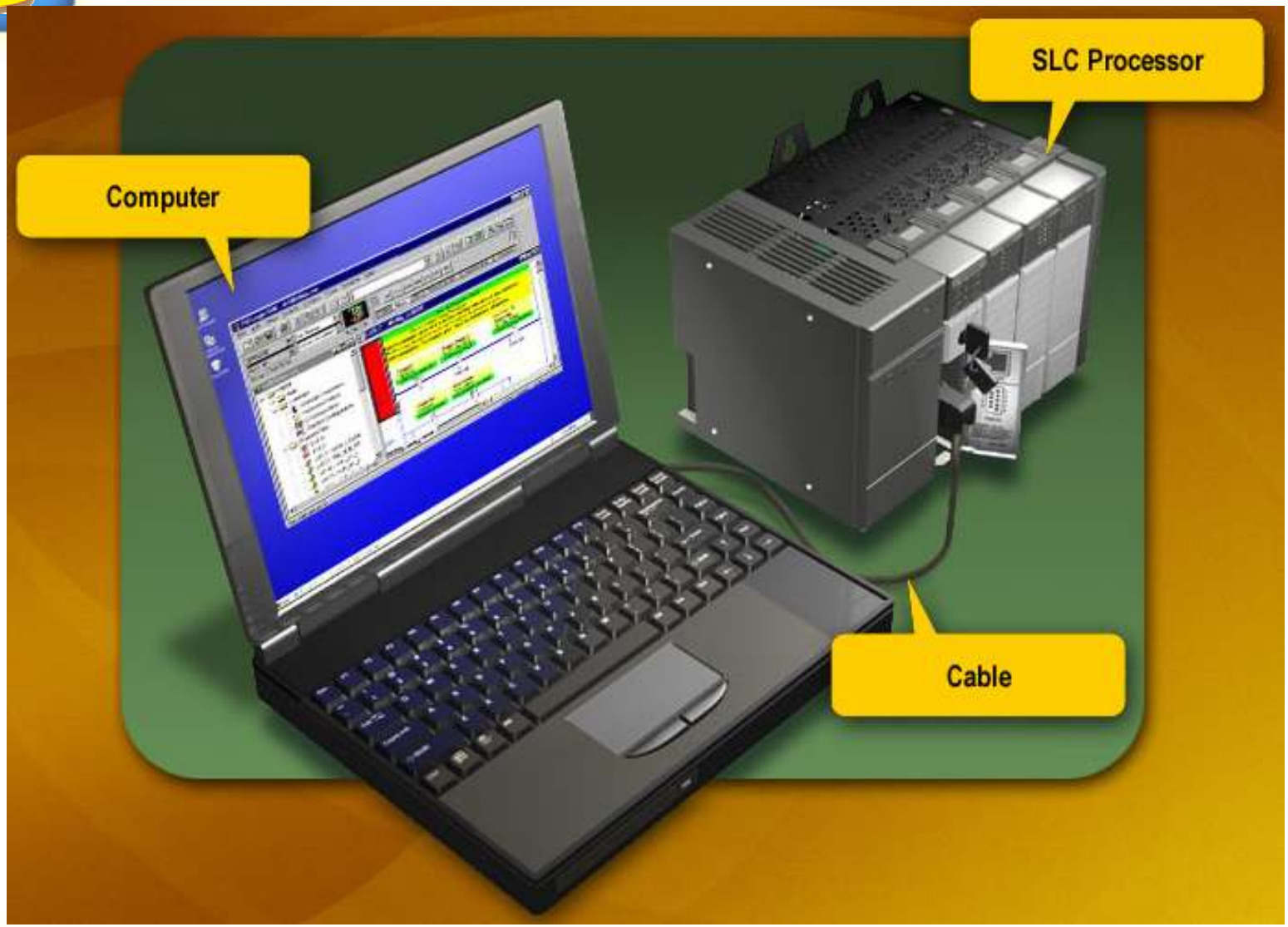


MODULAR CONTROLLER

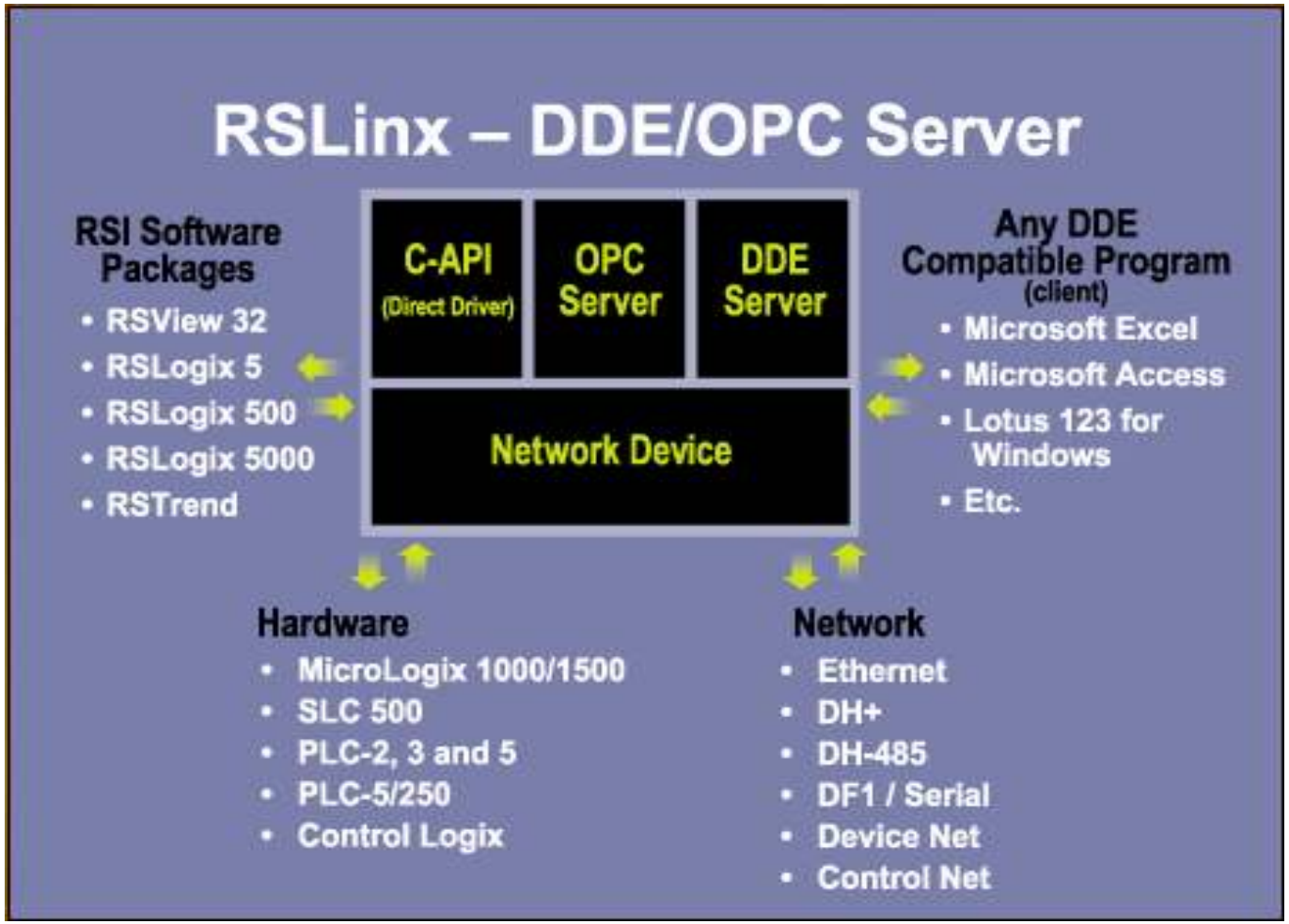
PLC PROGRAM FLOW



DIRECT PC-PLC LINK



PC-PLC LINK SOFTWARE TOOL



PC-PLC LINK SOFTWARE TOOL



PLC PROGRAM UP-LOAD



PLC PROGRAM DOWN-LOAD



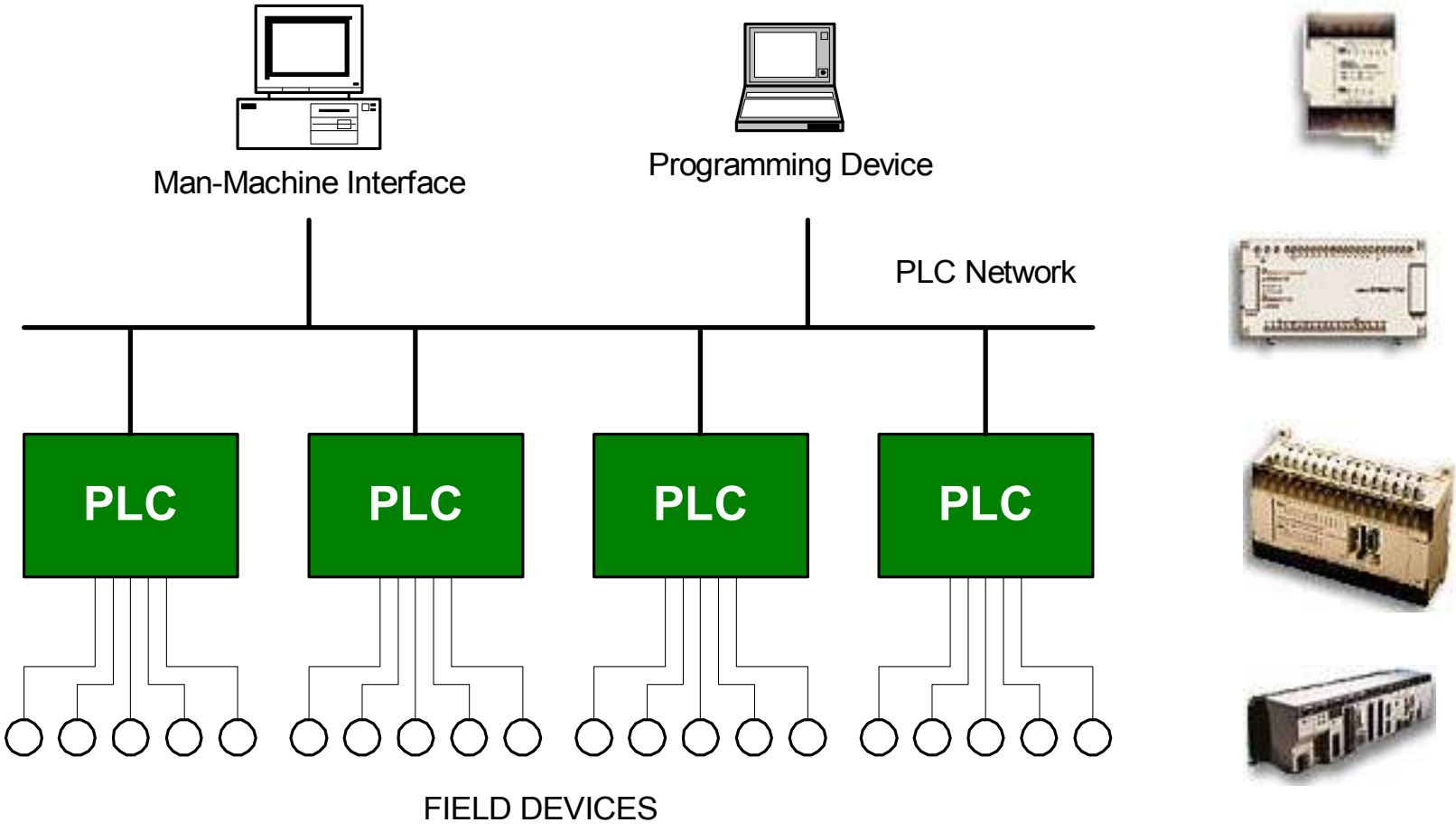
ON-LINE EDITING



ON-LINE EDITING FUNCTIONS

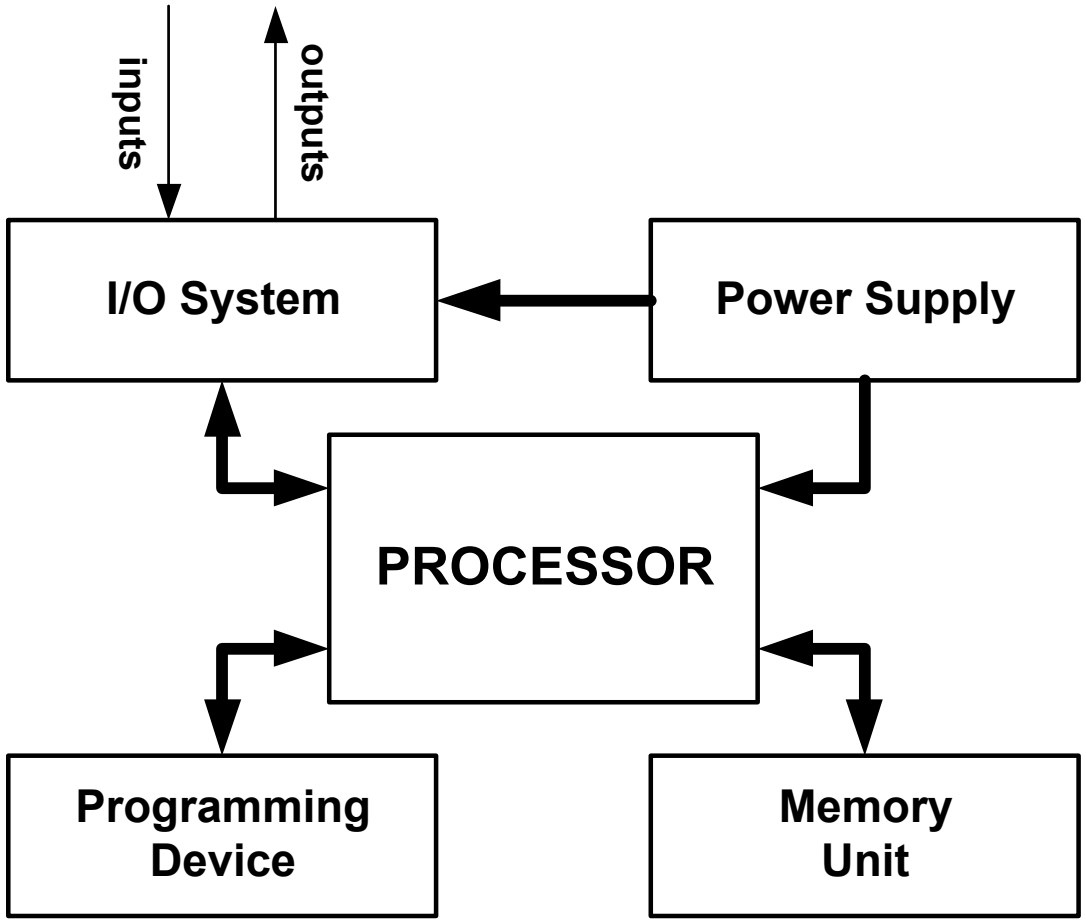
- Inserting Rungs
- Replacing Rungs
- Deleting Rungs
- Documenting Program
- Modifying Addresses or Instruction Types

A Picture of PLC System



Basic Components of PLC

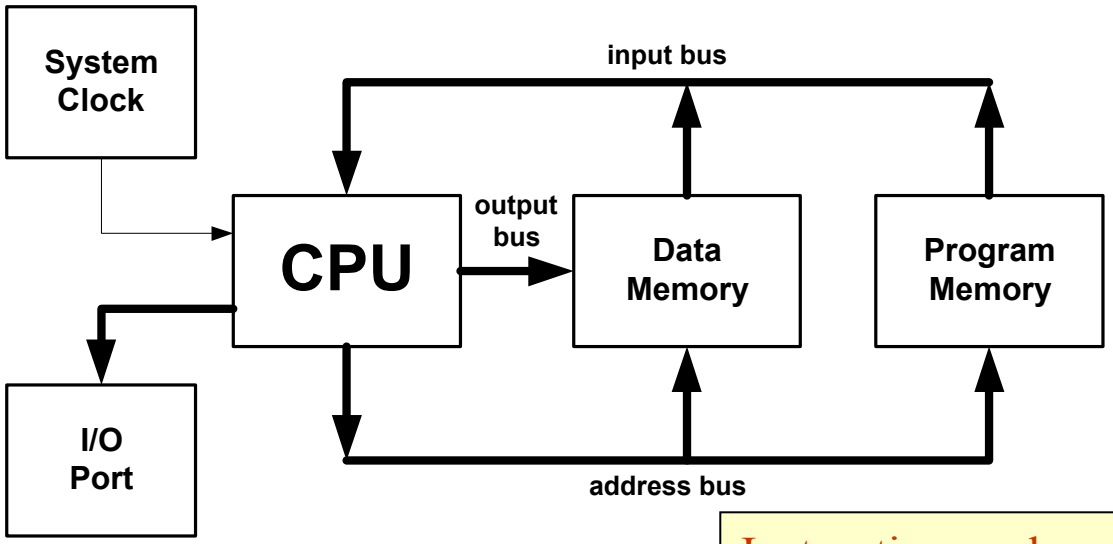
- Processor
- Memory
- I/O Interface
- Programming Device & Languages
- Power Supply



How does a PLC work ?

- Examine the status of inputs and outputs
- Controls some process or machine through outputs using some control logic
- This control logic is executed periodically by the processor in a predetermined sequential order
- User can change the control logic using a programming language and it is stored in the program memory

Processor & Memory



- 80386/80486 microprocessor based
- Random Access Memory (RAM)
- Electrically Erasable Programmer Read Only Memory (EEPROM)

Instruction cycle :

Fetch the next instruction from program memory

Place it in the instruction register in the CPU

Increment the program counter by one

Execute the instruction

Input/Output Systems

- Discrete Inputs/Outputs
- Analog Inputs/Outputs
- Special Purpose Modules
- Intelligent Modules
- Communication Modules

Discrete Signal Types

- Selector switches
- Temperature switches
- Flow switches
- Level switches
- Valve position switches
- Starter auxiliary contacts
- Pushbuttons
- Motor starter contacts
- Limit switches
- Pressure switches
- Hand switches
- Proximity switches
- Relay contacts
- Photoelectric sensors
- Anunciators
- Alarm lights
- Electric control relays
- Electric fans
- Indicating lights
- Electric valves
- Alarm horns
- Solenoid valves
- Motor starters
- Heater starters

Discrete Input Modules

- **AC and DC Discrete Input Module**
  - Electrical isolation between the field device (power) and the controller (logic)
  - Common return line connection
  - ACI-XX where XX is voltage (120 and 220 VAC)
  - DCI-XX where XX is voltage (12, 24, and 48 VDC)
- **TTL**
  - TTL-compatible devices (solid-state controller and sensing instruments)
  - external +5 V dc power supply
- **Isolated Input Module**
  - Separated return lines
  - IACI-XX or IDCI-XX

Discrete Output Module

- **AC Output Module**
  - AC loads (using Triac/ SCR as switch)
  - RC snubber protection
  - Peak voltage limiter (metal oxyde varistor - MOV)
  - Fuse protection
  - External switching voltage
- **DC Output Module**
  - DC loads (using power transistors)
  - Fuse protection
- **TTL Output Module**
  - TTL-compatible output devices (seven segment LED, IC, and +5 Vdc logic-based devices)
  - External +5 Vdc power supply
- **Isolated AC Output Module**
- **Dry Contact**
  - Normally open (NO)
  - Normally closed (NC)

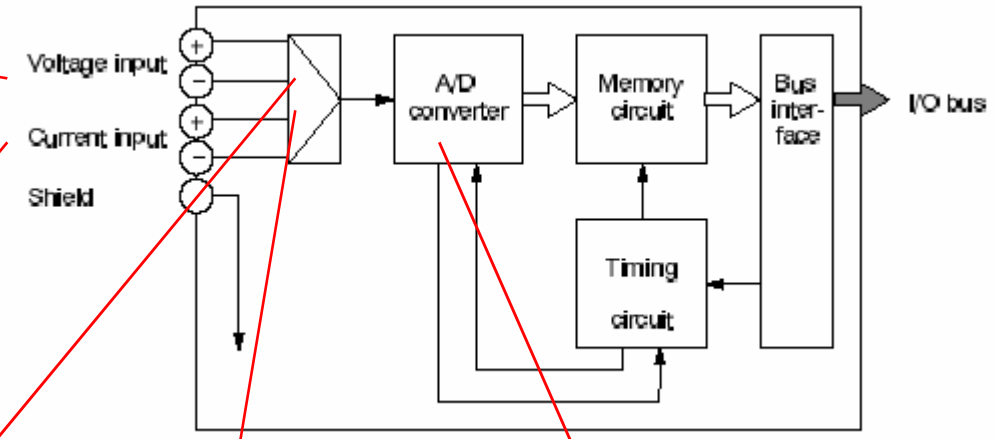
Analog Devices

- Flow transmitters
- Pressure transmitters
- Temperature transmitters
- Analytical transmitters
- Position transmitters
- Potentiometers
- Level transmitters
- Speed instruments
- Electric motor drives
- Analog meters
- Chart data recorders
- Process controllers
- Current-to-pneumatic transducers
- Electrical-operated valve
- Variable speed drives

Analog Input Module

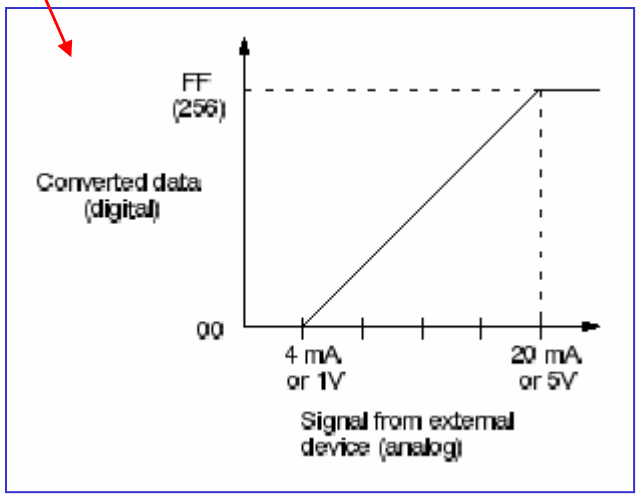
Single-ended
Differential

Current (0-20 mA)
Voltage (1-5 V)



Very high input impedance to prevent signal loading

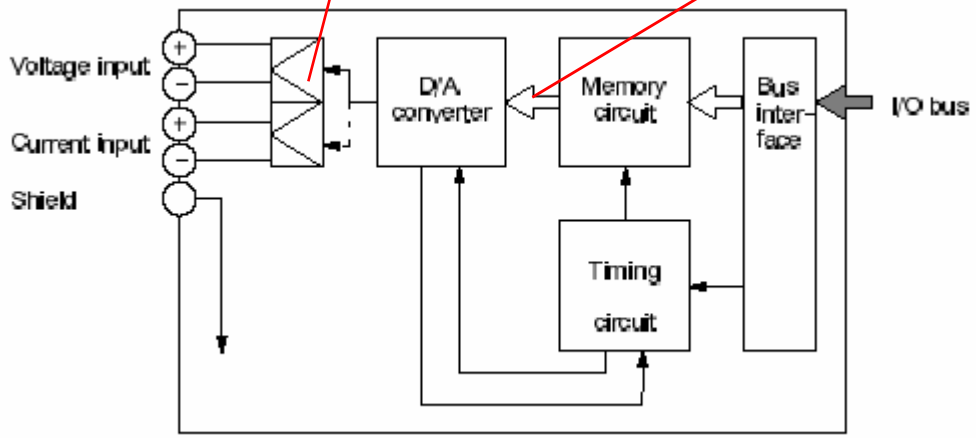
Filtering and isolation circuits

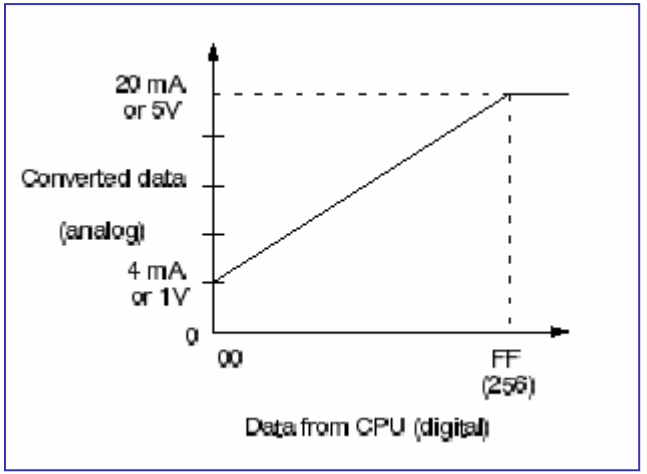


Analog Output Module

Isolation between the output circuit and the logic circuit is provided through optical couplers

Receive data from the CPU which is translated into a proportional voltage/current to control an analog field device





Special Purpose Modules

- Binary Coded Decimal (BCD) Modules
  - Provide parallel communication between the processor and input/output devices
- Encoder/Counter Input Module
  - Used for operations that require direct high speed encoder input into a counter
- Pulse Counter Input Module
  - Used to interface with field instruments that generate pulse such as positive displacement (PD) flowmeters and turbine type flowmeters

Intelligent Module

- Can perform complete processing functions, independent of the CPU and the control program scan
- Thermocouple Input Module
  - Designed to accept inputs directly from a thermocouple
  - Provides cold junction compensation
- Stepping Motor Module
  - Generates a pulse train that is compatible with stepping motor translators that represent distance, speed, and direction commands
- Control Loop Module (PID module)
  - Used in continuous closed-loop where the proportional-integral-derivative (PID) control algorithm is required

Communication Module

- **ASCII**
  - Used to send and receive alphanumeric data between peripheral equipment and the controller
- **Universal Remote I/O Link**
  - Allows I/O subsystems to be remotely located from the processor (1000 ft to several miles)
- **Serial Communications Module**
  - Used to communicate between the programmable controller and an intelligent instrument with a serial output
  - RS-232C, RS-422, or RS-485 communication link
- **PCMCIA Interface Card**
  - Allows communications between PLC or data highway and notebook PC
- **Ethernet**
  - Designed to allow a number of PLC and other computer-based devices to communicate over a high speed plant local area network
- **Fiber Optic Converter**
  - Transform electrical signals and transmit these signals through fiber optic cables

Designing I/O Systems : Electrical

- **Input voltage rating**
  - Lists the magnitude and type of signal the module will accept
- **Input current rating**
  - The minimum input current required at the module's rated voltage that the field device must be capable of supplying to operate the input module circuit
- **Input threshold voltage**
  - The voltage at which the input signal is recognized as being ON/true
- **Output voltage rating**
  - The magnitude and type of voltage that can be controlled within a stated tolerance
- **Output current rating**
  - The maximum current that a single output circuit in a module can safely carry under load
- **Output power rating**
  - The maximum total power that an output module can dissipate with all output energized
- **Backplane current requirements**
  - Lists the current demand that a particular I/O module internal circuitry places on the rack power supply

Designing I/O Systems : Mechanical

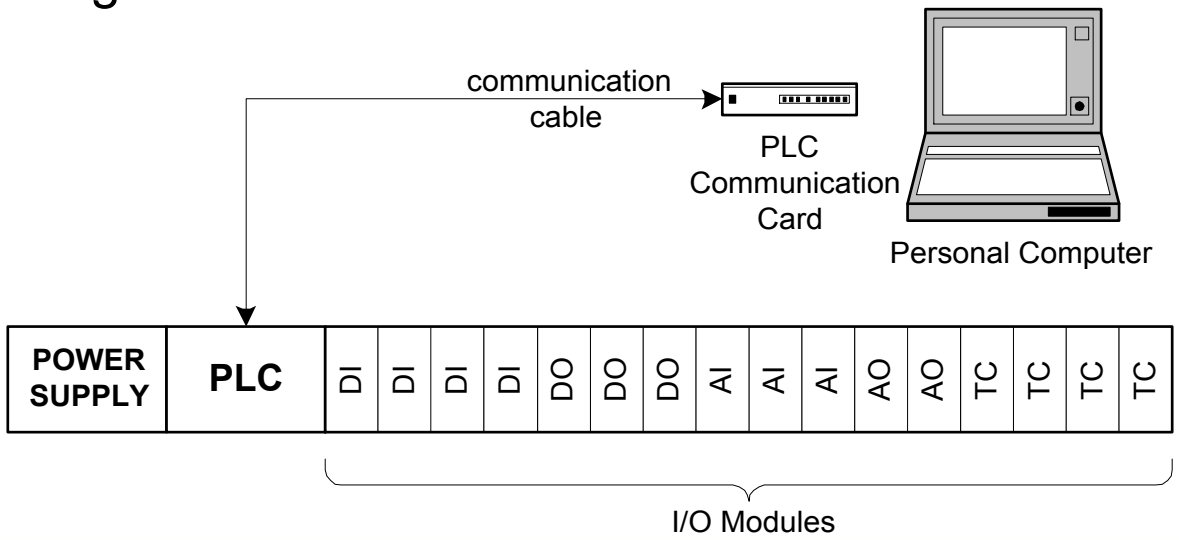
- Number of I/O points per module
- Number of wires
- Wire size specification
  - Number of conductors
  - Largest wire gage that the I/O terminal points will accept
- Ambient temperature rating
  - Based on the heat dissipation characteristics of the circuit components inside the I/O module
  - 0 - 60 C typical
- Humidity rating
  - 5 - 95 % typical relative humidity

IEC1131-3 Standard Languages

- Ladder Diagram (LD)
  - Uses a standardized set of ladder logic symbols
- Sequential Function Charts (SFC)
  - Blocks connected together like a circuit diagram
- Functional Block Diagram (FBD)
  - A graphical language used to describe sequential operations
- Instruction List (IL)
  - Low level language
- Structured Text (ST)
  - High level structured language designed for automation processes

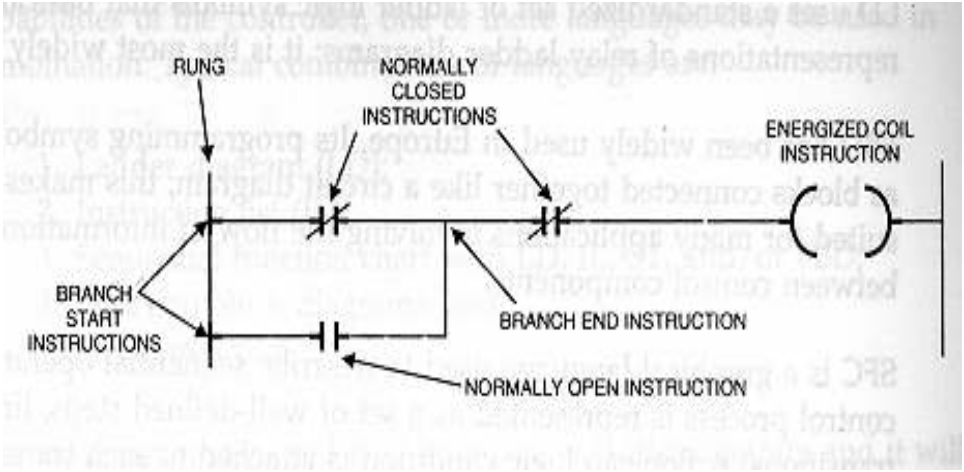
Programming Devices

- Used to enter, store, and monitor the PLC software
- PC-based system
- Connected to the PLC only during :
  - programming & testing
  - startup
  - troubleshooting



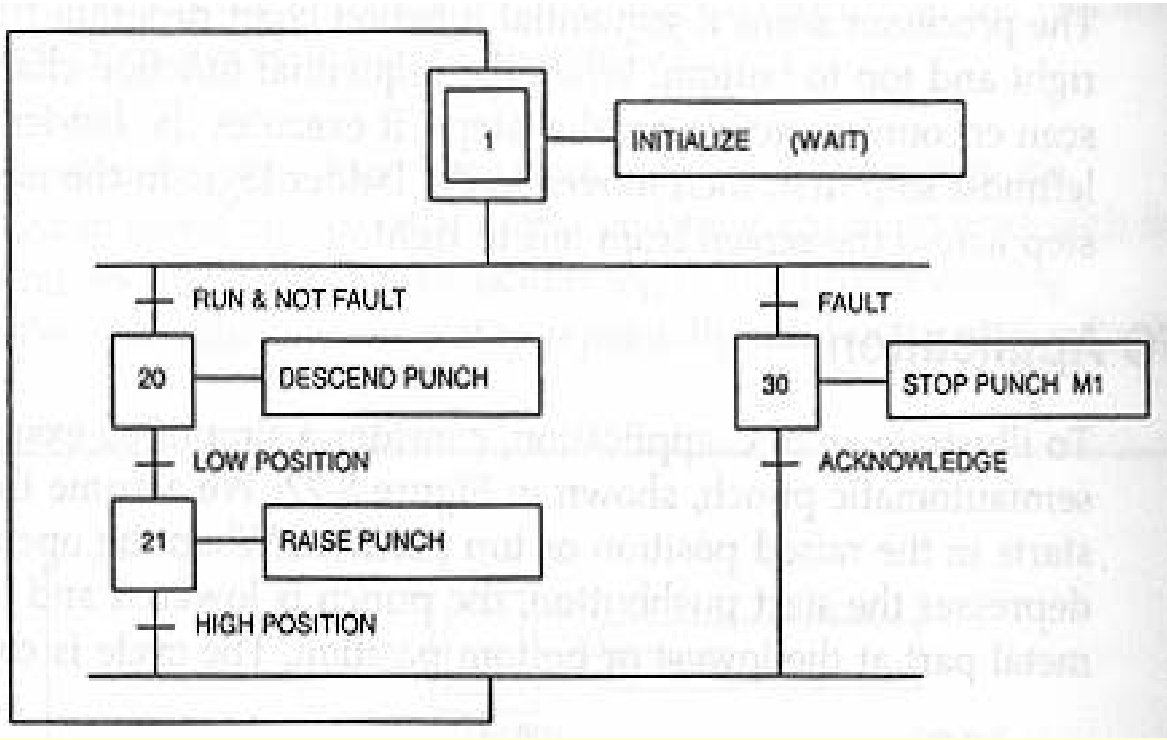
Ladder Diagram (LD)

- Representations of relay ladder diagrams
- The most widely used
- Composed of six categories of instructions
  - Relay type
  - Timer/counter
  - Data manipulation
  - Arithmetic
  - Data transfer
  - Program control



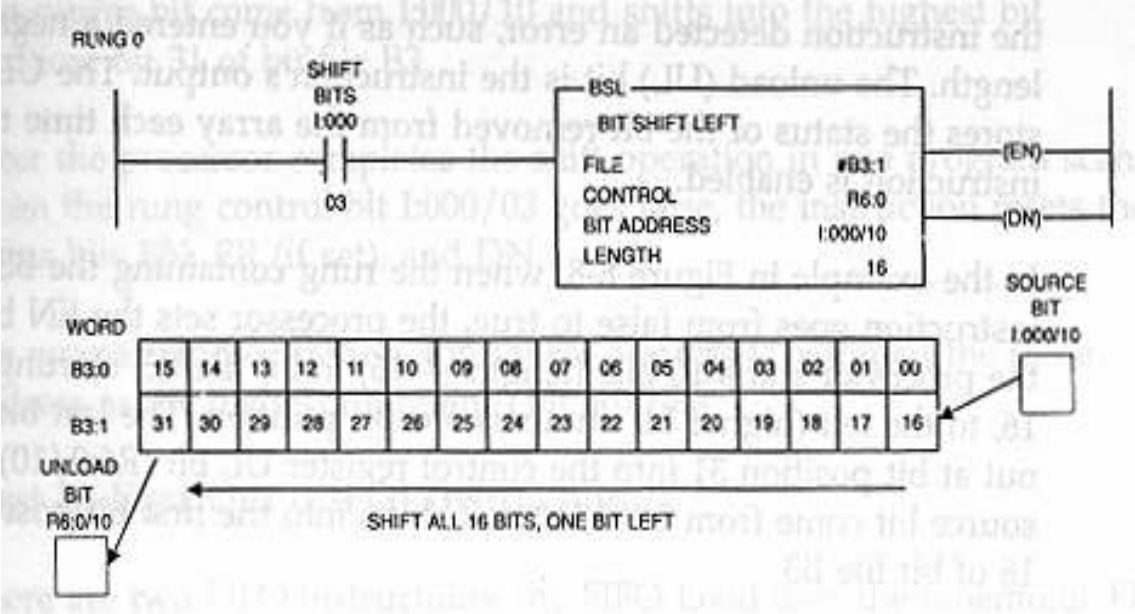
Sequential Function Charts (SFC)

- Represented as a set of well-defined steps, linked by transitions
- Actions within the steps are detailed by using the Structured Text language (ST)



Functional Block Diagram (FBD)

- Describes a relationship or function between input and output variables
- Well suited for many applications involving the flow of information or data between control components
- Allows the programmer to build complex control procedures
- Extensive function blocks library is provided



Instruction List (IL)

- Very effective for small simple applications or for optimizing parts of an application
- A list of low level instructions
- Consists of standardized operators
  - Data manipulation
  - Arithmetic
  - Branch instruction

Example :

```
Start :   LD     IX1     (* load input IX1, start pushbutton *)
          ANDN   MX5     (* AND with NOT of MX5)
          ST     QX2     (* store output QX2 to start motor *)
```

Structured Text (ST)

- Used mainly to implement complex procedures
- Default language for the description of the actions within steps and conditions attached to the transitions of the SFC
- Statement types :
  - assignment
  - subprogram or function call
  - “C” function block call
  - Selection (if, then, else, case, etc)
  - Iteration (for, while, repeat, etc)
  - Control (return, exit, etc)
  - Special

```
imax:=max_ite;
cond:=X12;
if not(cond(*alarm*))
then return;
end_if;

for i:=1 to max_ite do
if i<>2 then
SPcall();
end_if;
end_for;
```

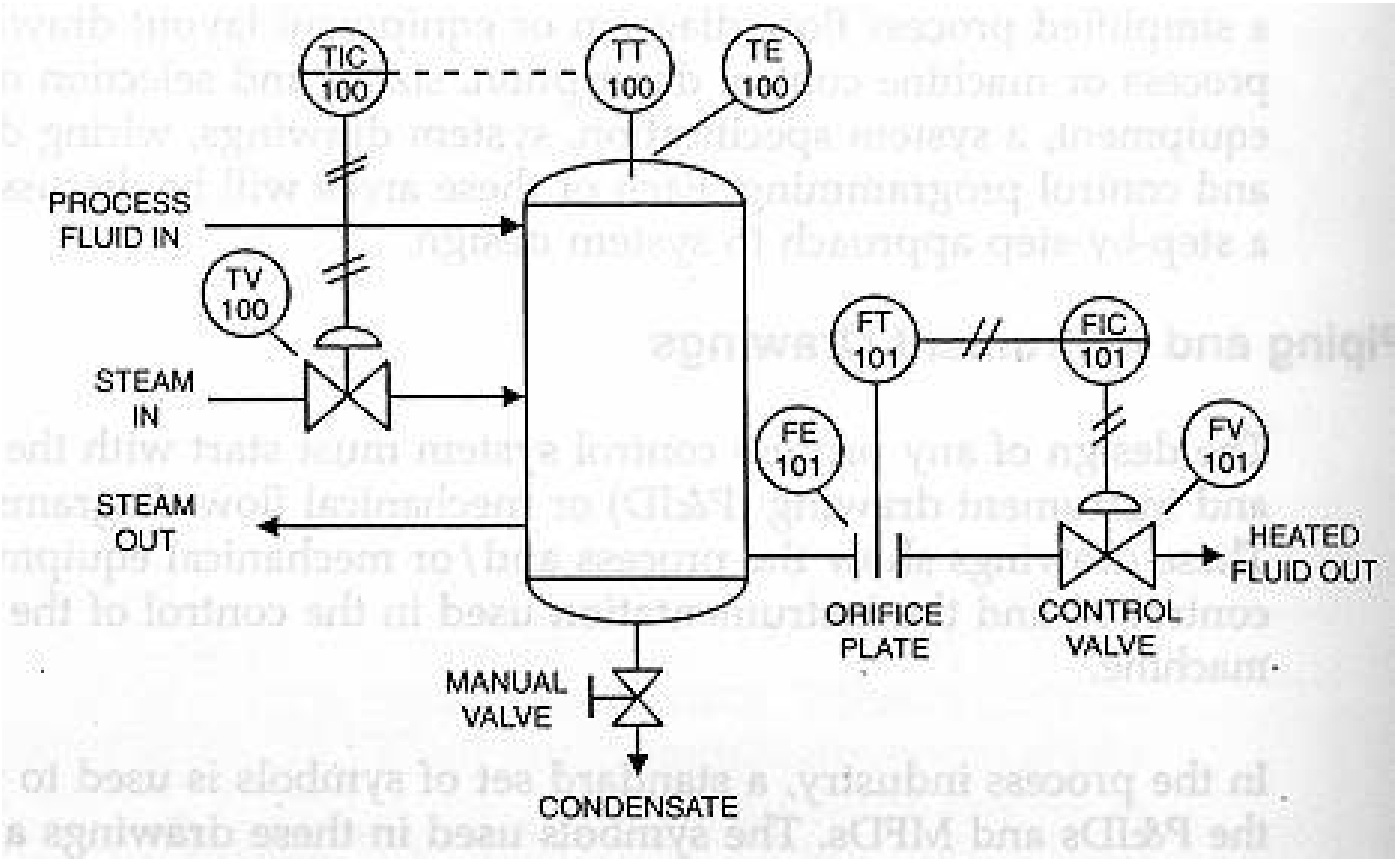
PLC System Design

1. Process Description
2. I/O Sizing
3. Memory Sizing
4. Selecting programming language
5. Peripheral requirements
6. System drawing and I/O wiring diagrams
7. System programming

Process Description

- States the purpose and the steps of the process/plant operation
- Process Description is the most important step in the design process
- Bridge of communications between the user and the designer
  - Piping and Instrument Diagram (P&ID) and Mechanical Flow Diagrams (MFDs)
  - Simplified drawing that shows only the equipment and instrumentation controlled or measured by the PLC is required
  - This drawing will be used to show the status of the process in each step or state to aid in the programming of the system

Piping & Instrument Diagram (P&ID)



I/O Sizing (1)

- Estimate the number of input/output (I/O) required to control the process
  - Obtain the number of device from P&ID diagram
  - Add the number of I/O points from each devices to obtain total I/O points
- Consider different types of I/O :
  - Discrete AC/DC : Limit switches, push buttons, selector switches, solenoid, etc.
  - TTL : solid state displays and electronic instrumentation
  - Analog : Level transmitter, pressure transmitter, etc.
  - Encoders

I/O Sizing (2)

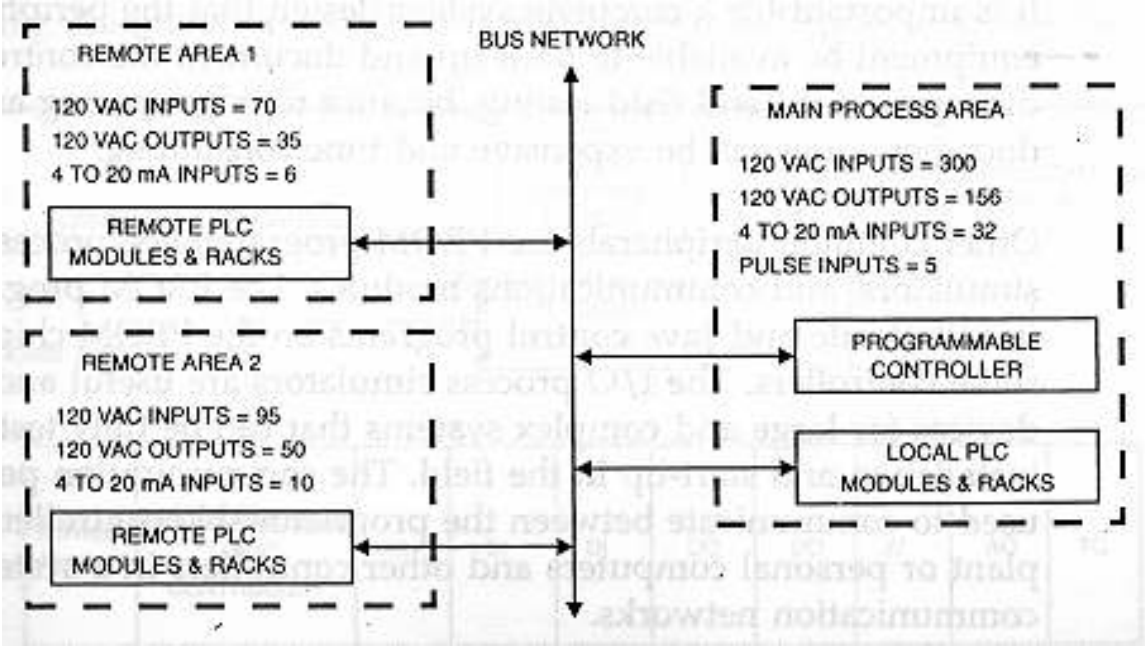
- Select PLC size :
  - *Micro* : up to 32 I/O points
  - *Small* : 32 - 256 I/O points
  - *Medium* : 256 - 1024 I/O points
  - *Large* : > 1024 I/O points
- Estimate the number of I/O module
  - Classify each type of I/O points (discrete, analog, isolated, TTL, dry contact)
  - Select suitable I/O modules from PLC manual and obtain the number of I/O points per module
  - The number of each I/O module type required = *[Total I/O points] / [Number of I/O points per module]*
- Add spare and future expansions (10-20% spare capacity)

Memory Sizing

- The amount of memory required depends on :
  - control program complexity
  - the number of I/O points
- Precise (*almost impossible*) method to determine memory size :
  - Write out the control program
  - Count the number of instructions used
  - Multiply this count by the number of words used per instructions (obtained from PLC programming manual)
  - Add the amounts of memor used by executive programs and the processor overhead
- **Practical method** :

  Total memory = 10 x [the number of I/O points]

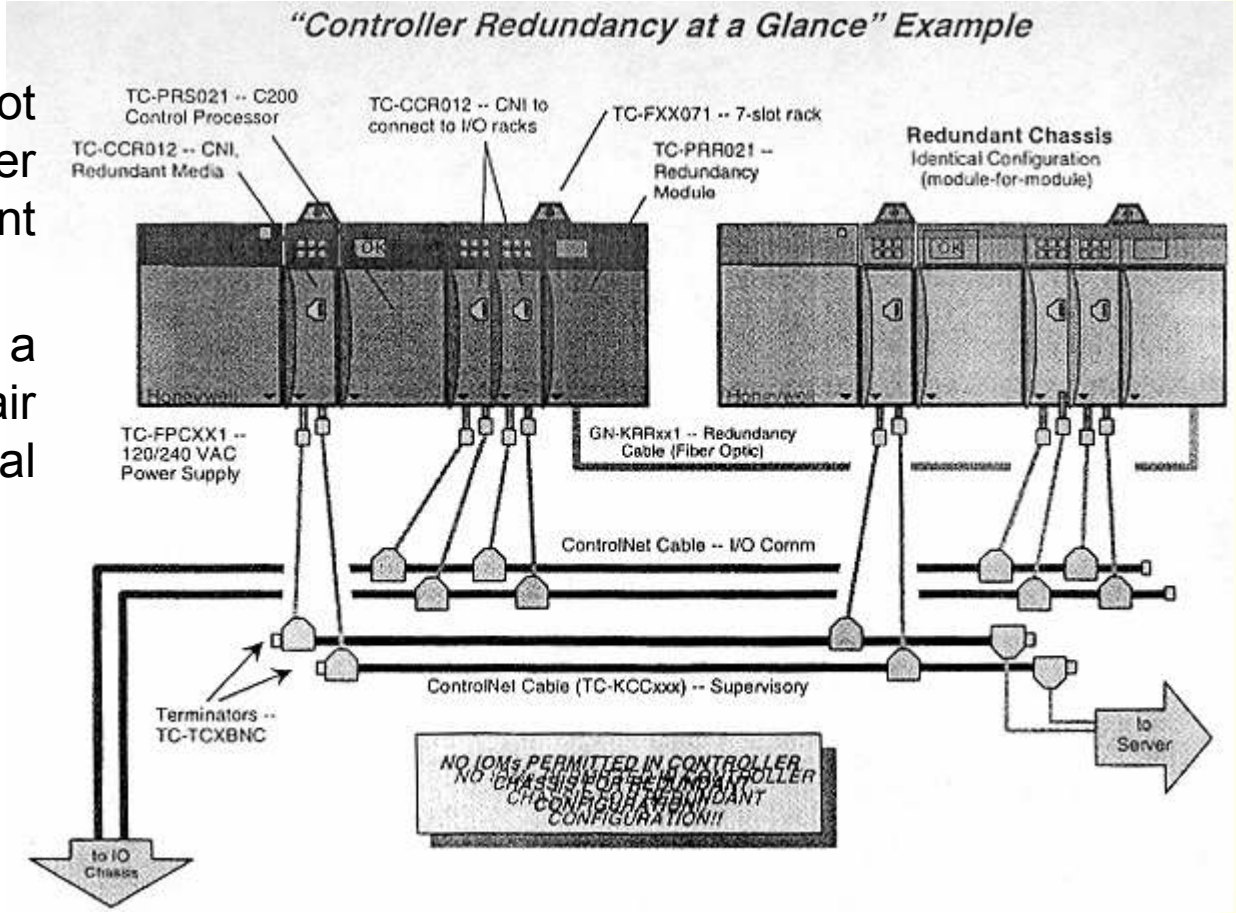
I/O and Memory Sizing Example



- I/O Points
  - Remote Area 1 : I/O = 70 + 35 + 6 = 111
  - Remote Area 2 : I/O = 95 + 50 + 10 = 155
  - Main Process Area : I/O = 300 + 156 + 32 + 5 = 493
  - Total I/O points = 759
  - Spare points = 10% x 759 = 76
- PLC size = Medium-sized (1024 I/O points max.)
- Memory size = 10 x 759 = 7590 or 8K

Redundant Configuration

- I/O modules are not permitted in either chassis of a redundant chassis pair
- Both chassis of a redundant chassis pair must have identical slot-for-slot configurations



Selecting Programming Language

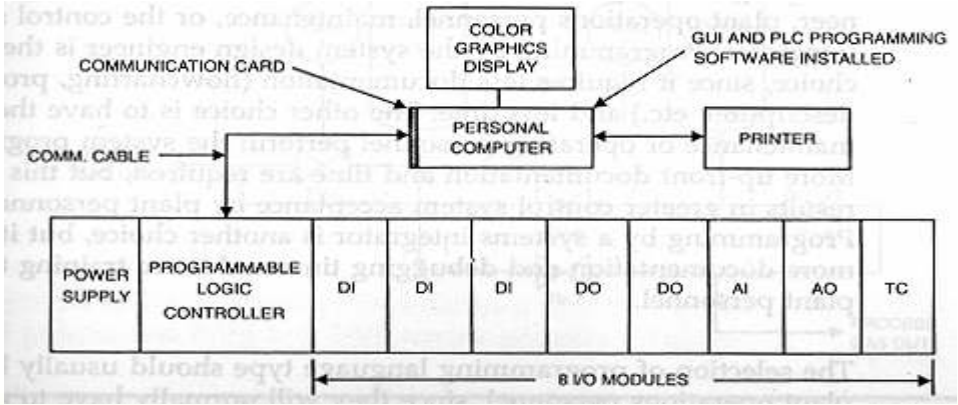
- Most PLCs offer the basic ladder logic instructions plus a combination of the other types of languages
- Programming language selections is depends on :
  - Complexity of the control system
  - Background knowledge of the control system programmers and operators

Peripheral Requirements

- Peripheral = other equipment in the PLC system that is not directly connected to field I/O devices
  - Compact portable programming device from PLC manufacturer
  - Portable PC with PLC software
  - Magnetic tape storage unit to store control program
  - PROM Programmer
  - Process I/O simulators
  - Communications modules
    - Depends on plant network design
    - Extra modules is required for integration within different brands
  - Operator interfaces
    - Hard-wired local and main control panels
    - GUI software runs on a personal computer
    - Intelligent peripheral devices such as touch screen
    - Industrial PC with function keys and GUI software

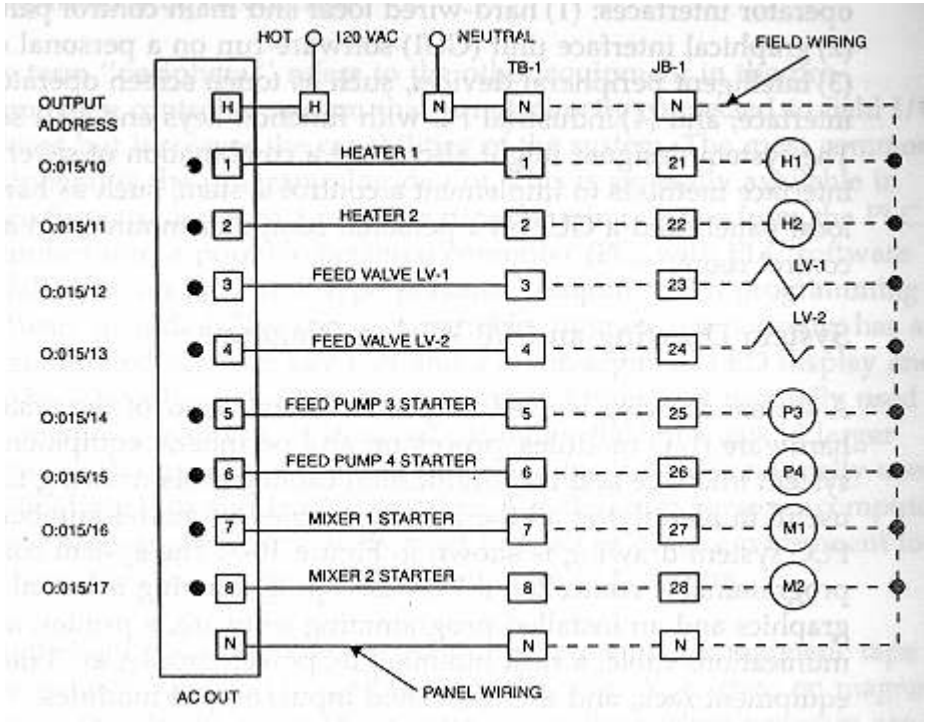
System Drawing

- Gives an overall view of the system hardware
  - I/O modules
  - processor
  - peripheral equipment
  - system interface
  - communication cabling
- Useful in identifying all the interface cables by model number



Example of I/O Wiring Diagrams

- Field wiring is normally indicated by a dashed line
- PLC output addresses are given on the left-hand side
- TB = Terminal Box
- JB = Junction Box

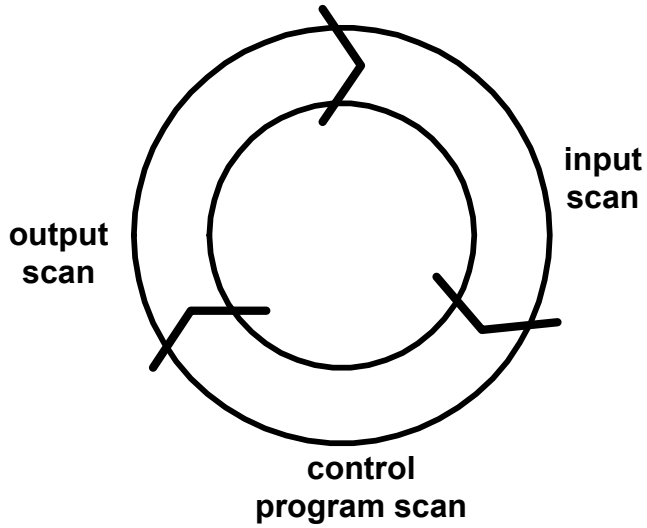


System Programming

- Person involved in system programming :
  - System design engineer
  - Plant operations personnel
  - Maintenance personnel
  - Control system integrator
- Programming by system design engineer takes less time and require less documentation (flowcharting, process description, etc.)
- Selection of programming language type should usually be left to plant operations personnel for easier maintenance and troubleshooting

Real Time Consideration

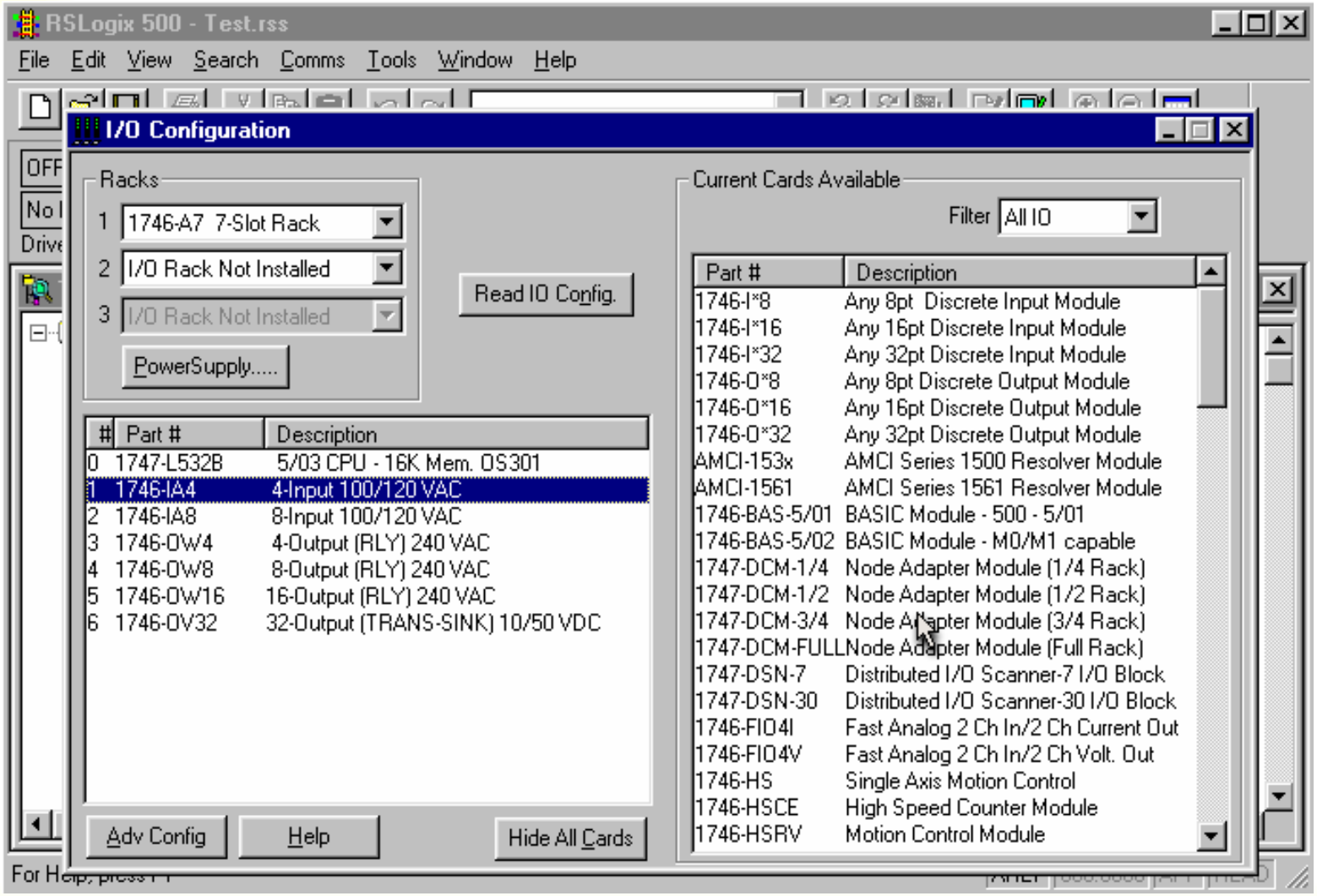
- The period required to examine inputs, perform the control logic, and execute the outputs is called "*scan time*"
- Scan time must be determined correctly to achieve sufficient real-time performance
- The numbers of I/O points and complex control algorithms affects scan time



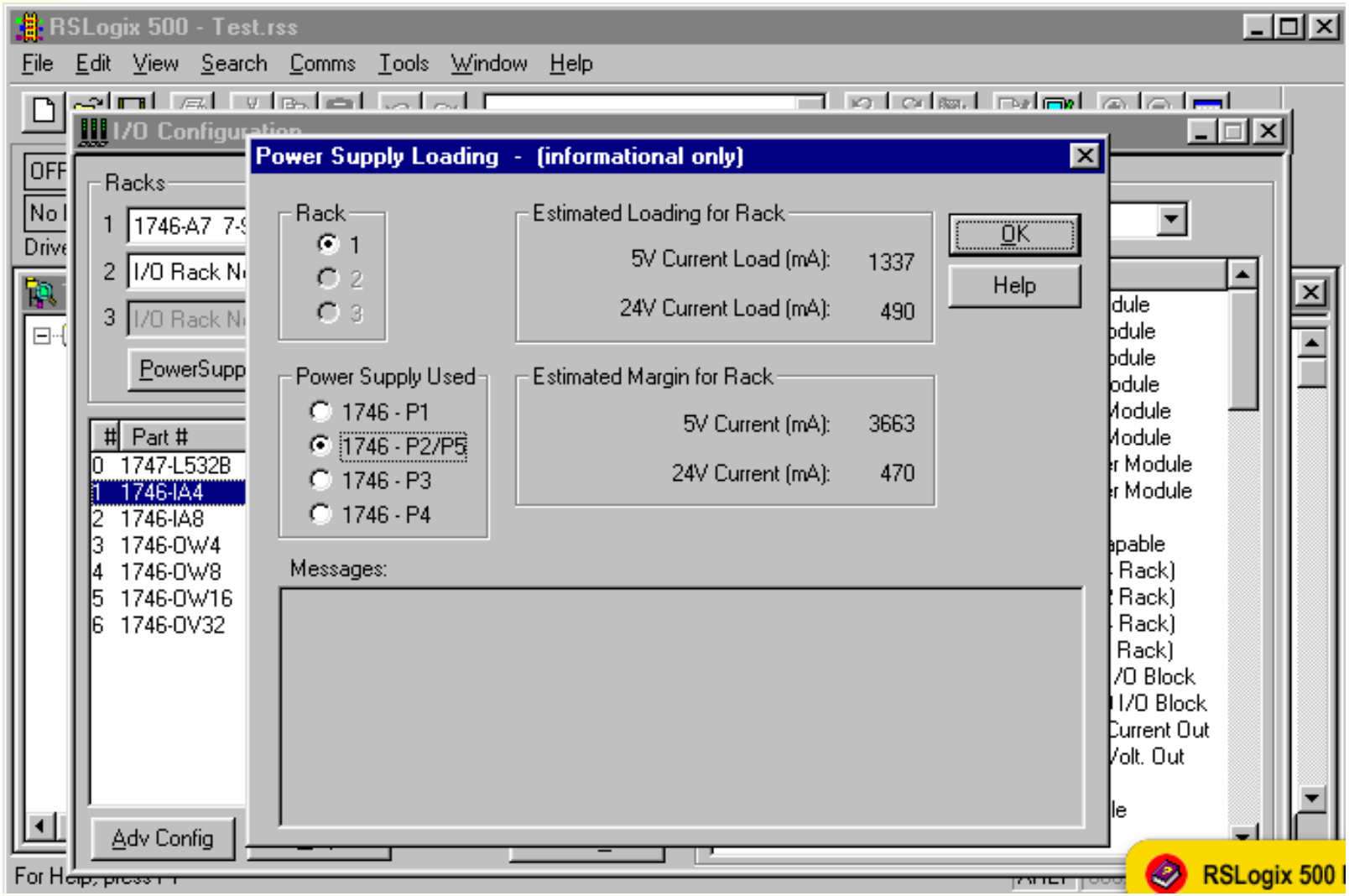
Installation

- Control Panel Design
  - Layout
  - Heating Consideration
  - Enclosure Standards (NEMA)
- Maintenance Features
- Panel Duct and Wiring Design
- Power Distribution Design
- Grounding Considerations
- Electrical Interference Considerations
- I/O Module Installation and Wiring
- Equipment Layout Design

I/O Configuration



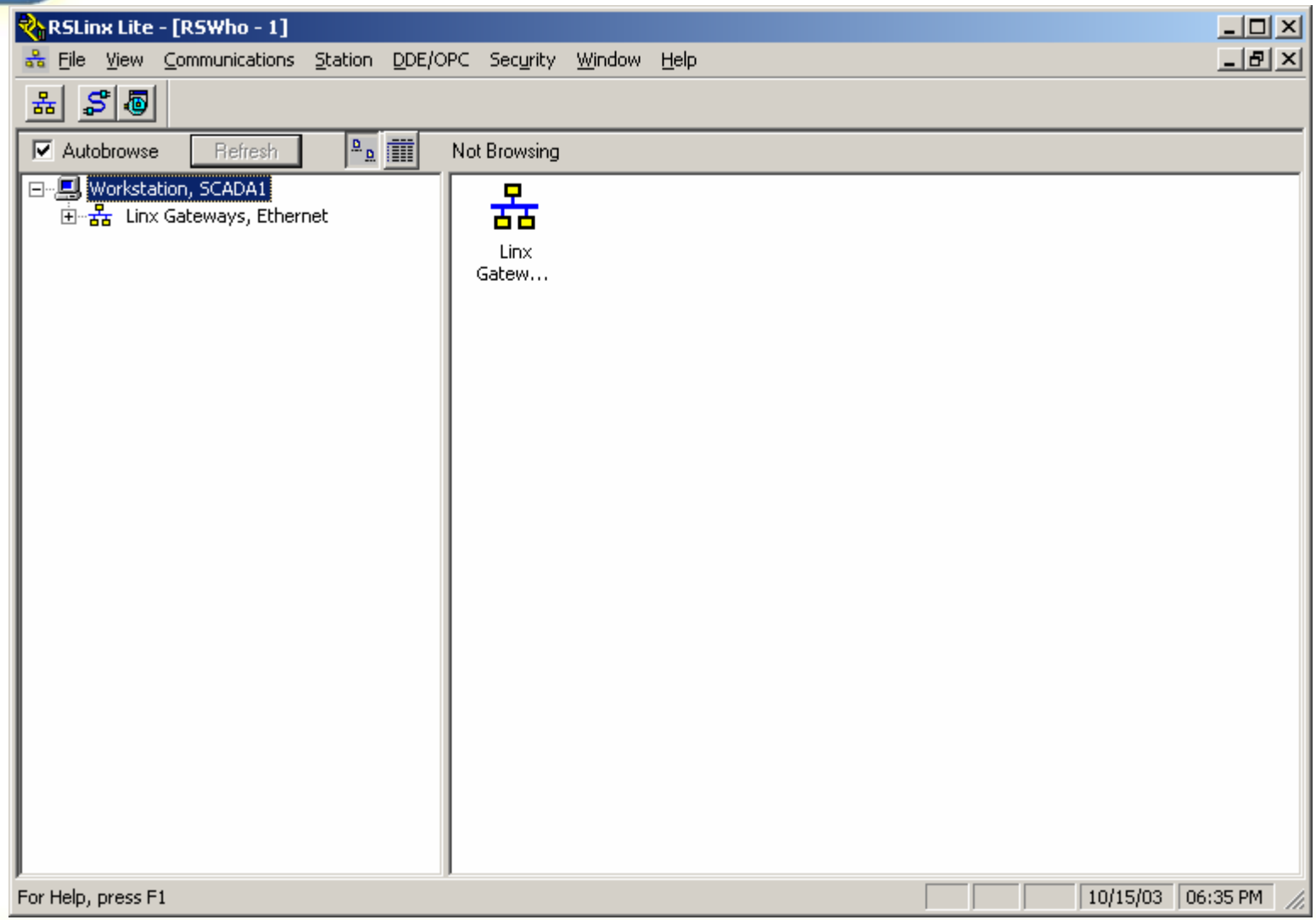
POWER SUPPLY OVERLOADING



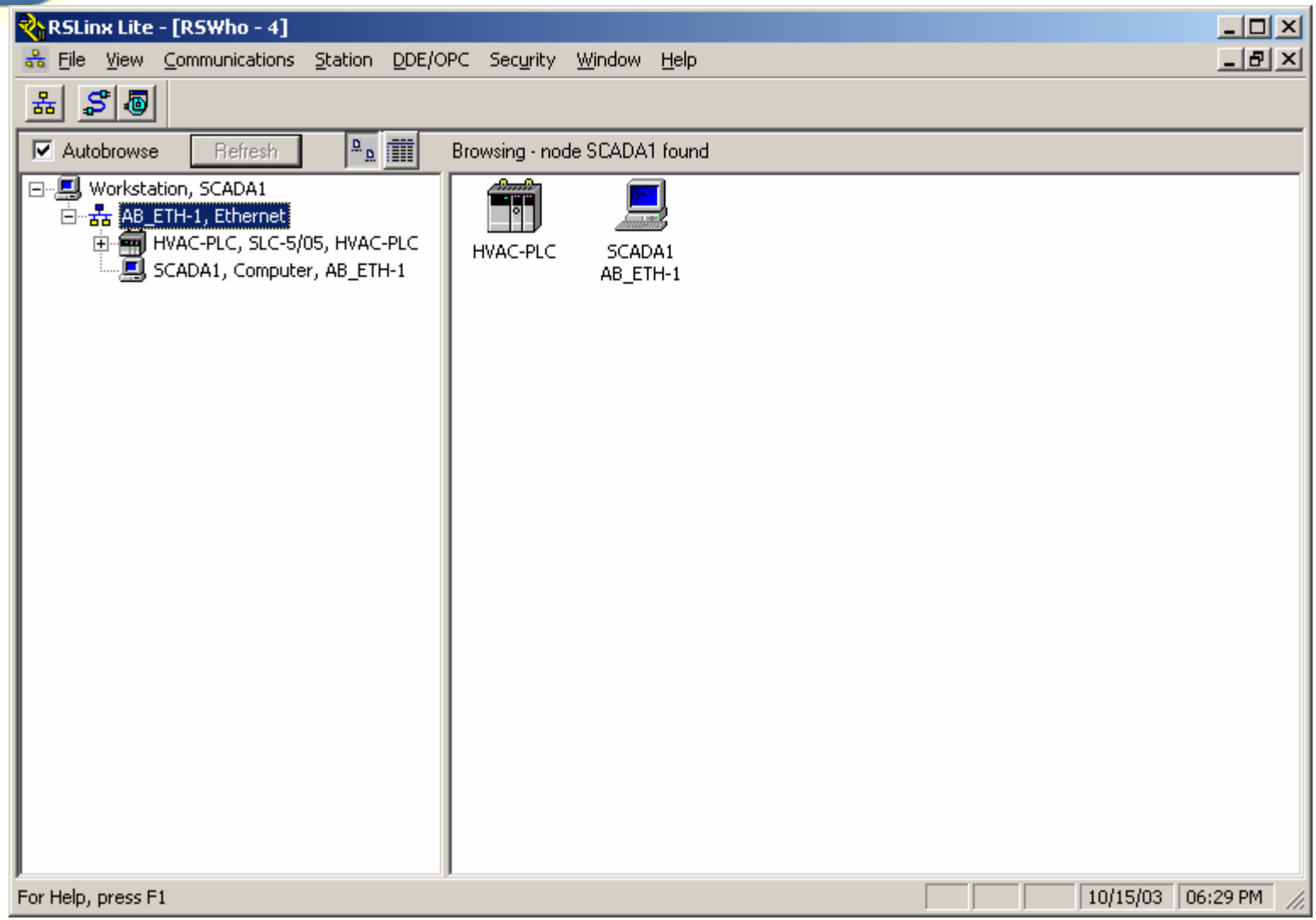
SUPPORTED COMMUNICATION PROTOCOL (SLC-505)

- Ethernet : Client/Server and Peer-to-Peer
- DF1 (RS-232): Full-Duplex (Point-to-Point), Half-Duplex (Slave) & Half-Duplex (Limited Master).
- DH-485: Max. 32 Nodes for PLCs, Computers & Data Terminals.
- ASCII: Display Device or Printer

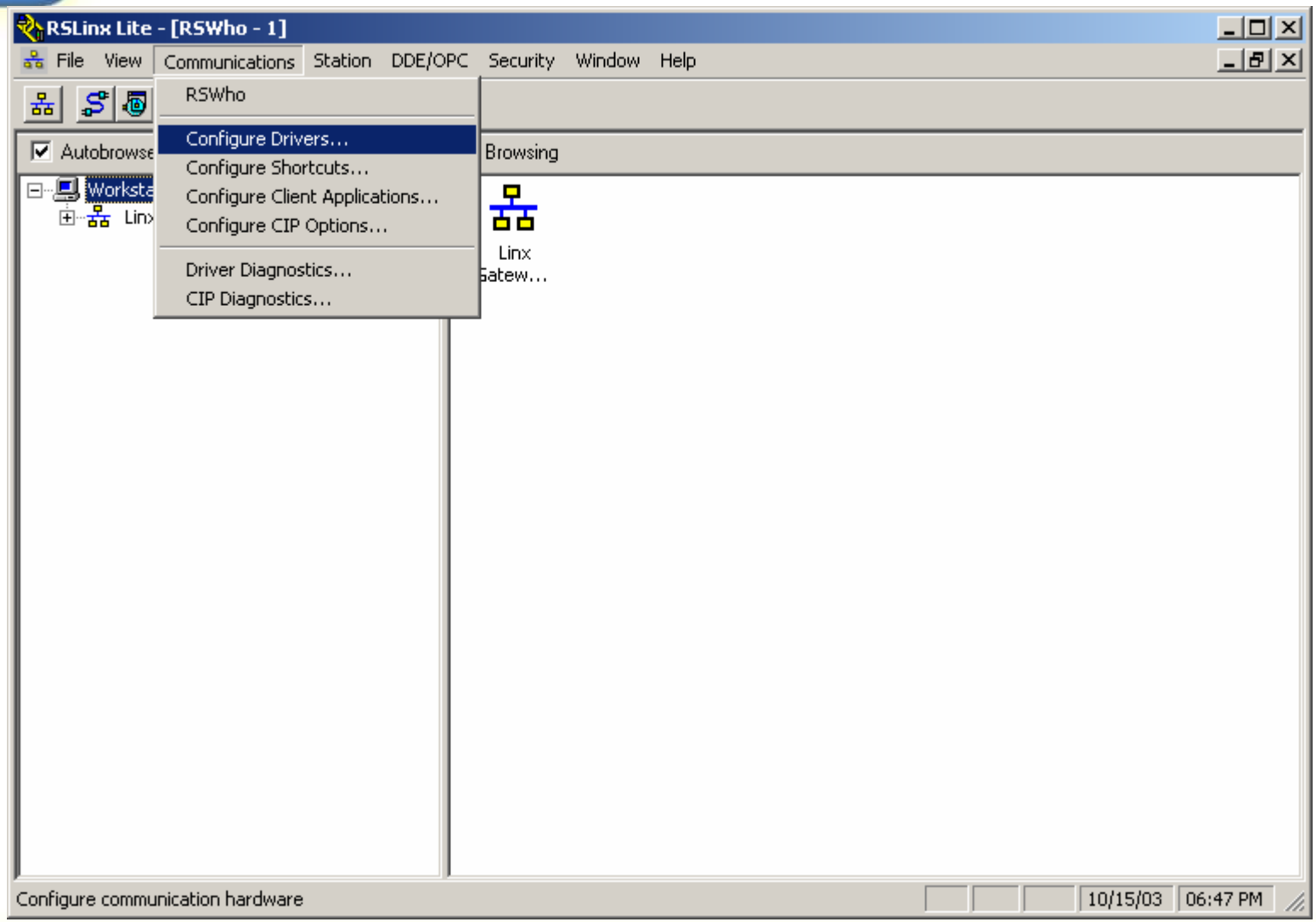
TOOLS TO CONNECT PLC



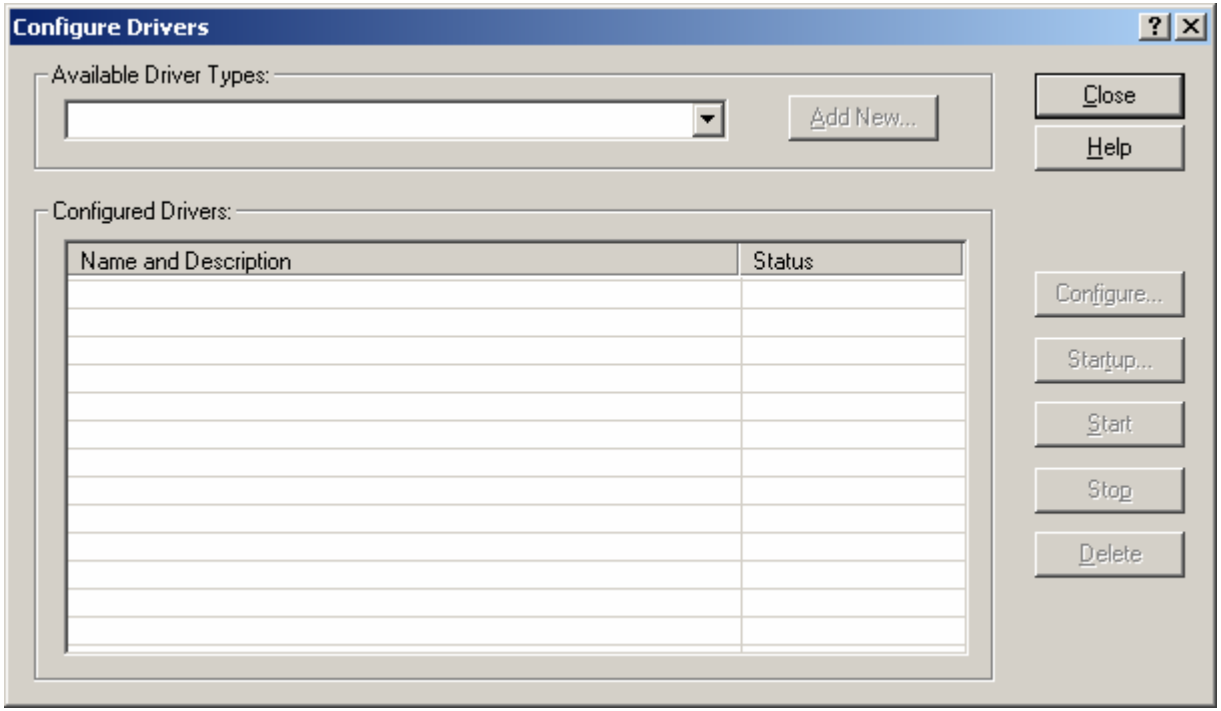
AUTOMATIC NODE BROWSING



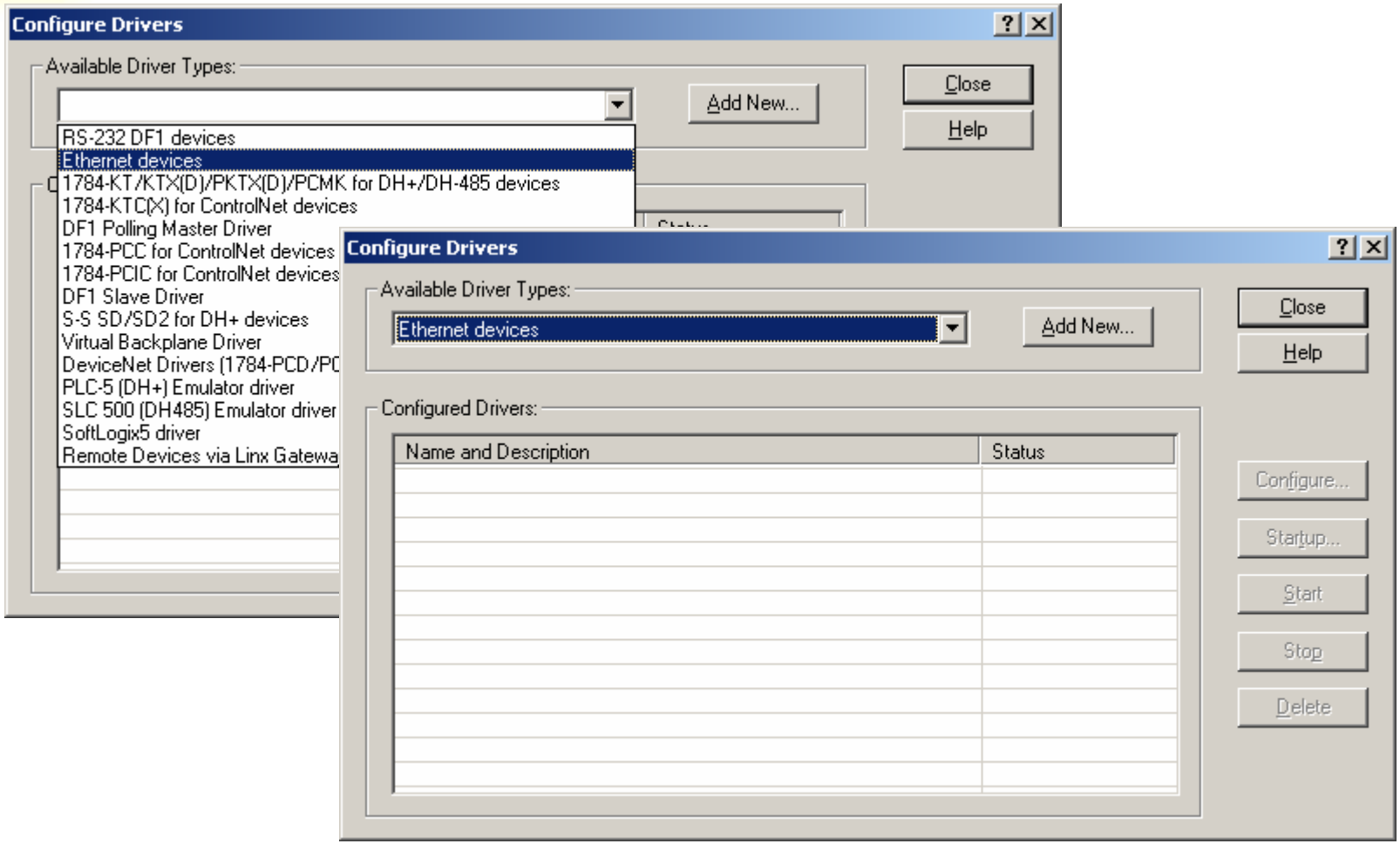
DRIVER CONFIGURATION



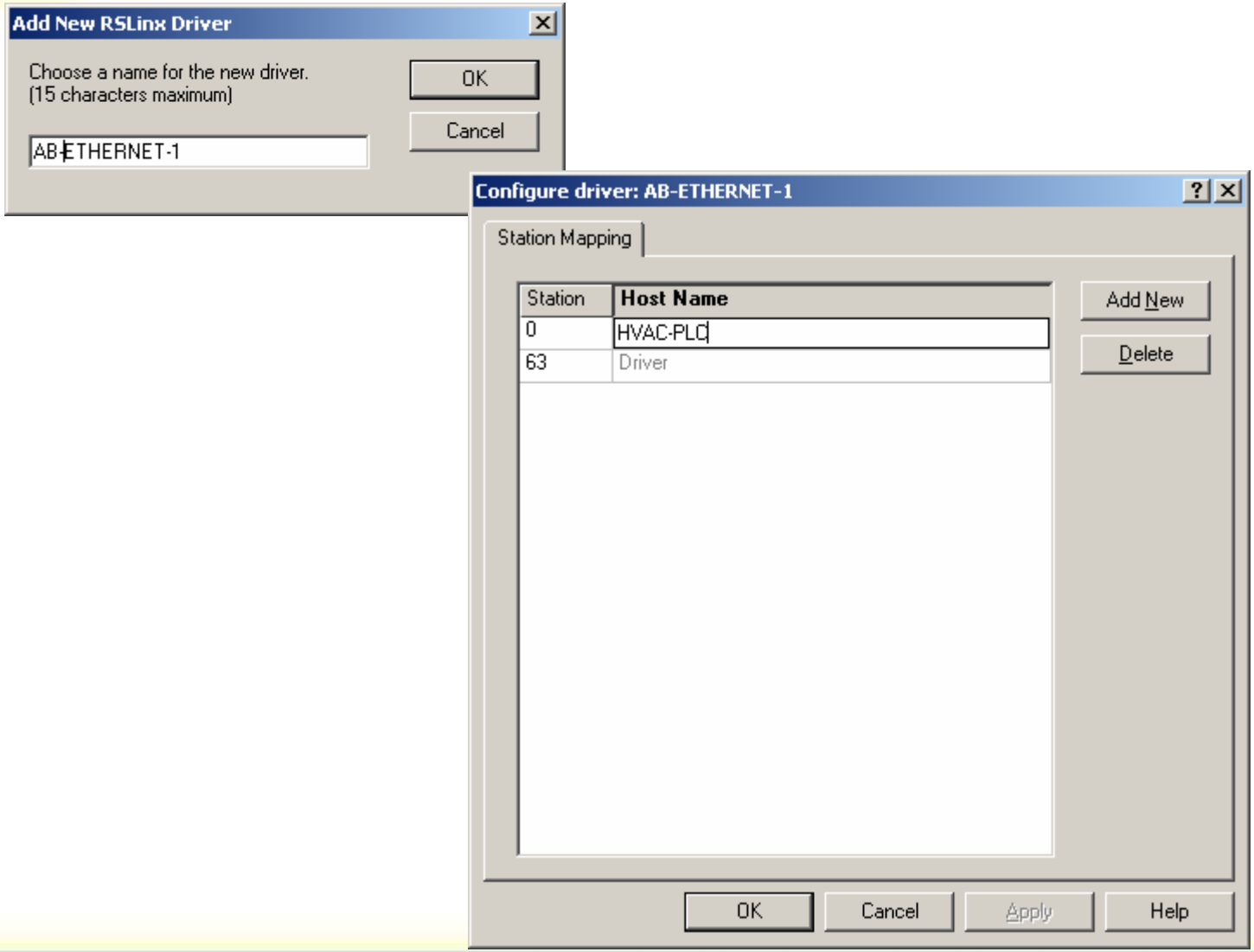
DRIVER CONFIGURATION



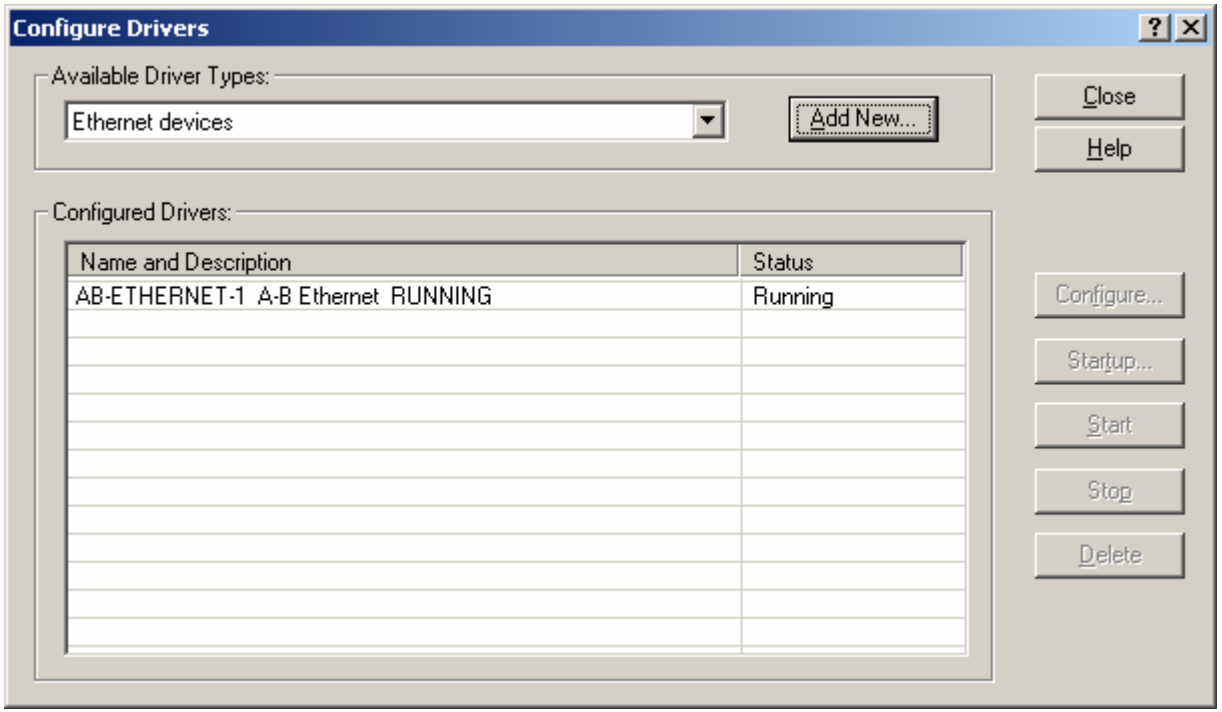
DRIVER SELECTION



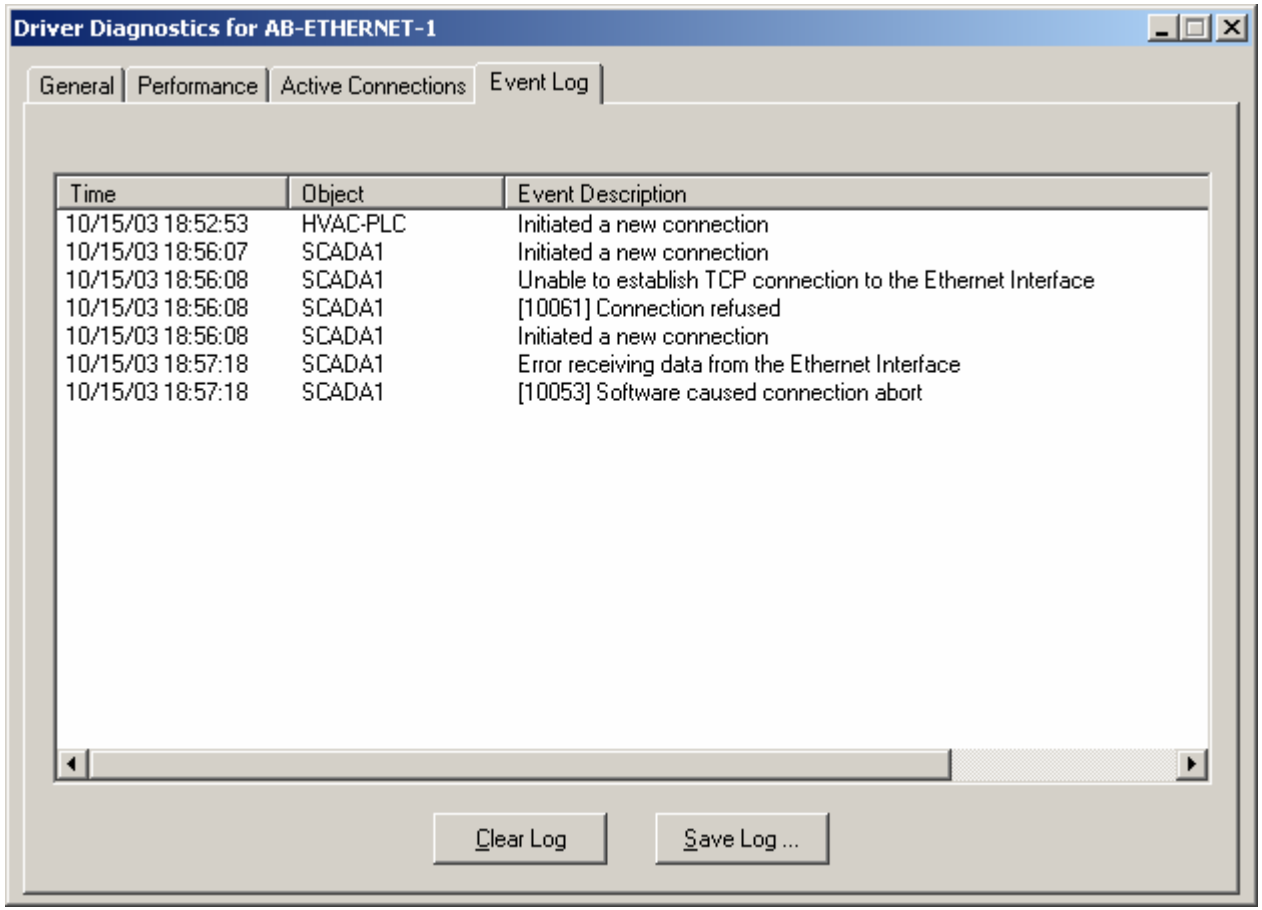
DRIVER CONFIGURATION



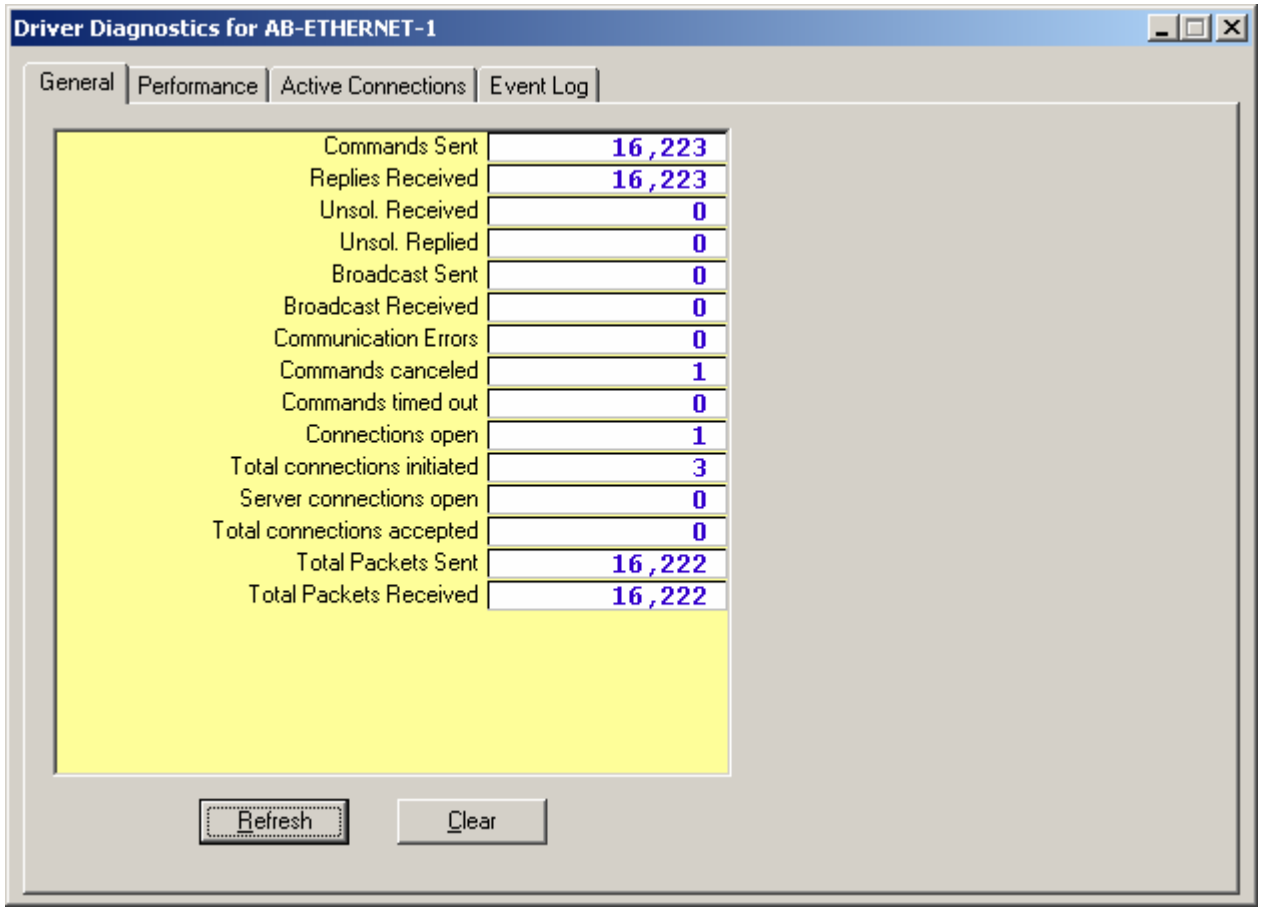
DRIVER CONFIGURATION



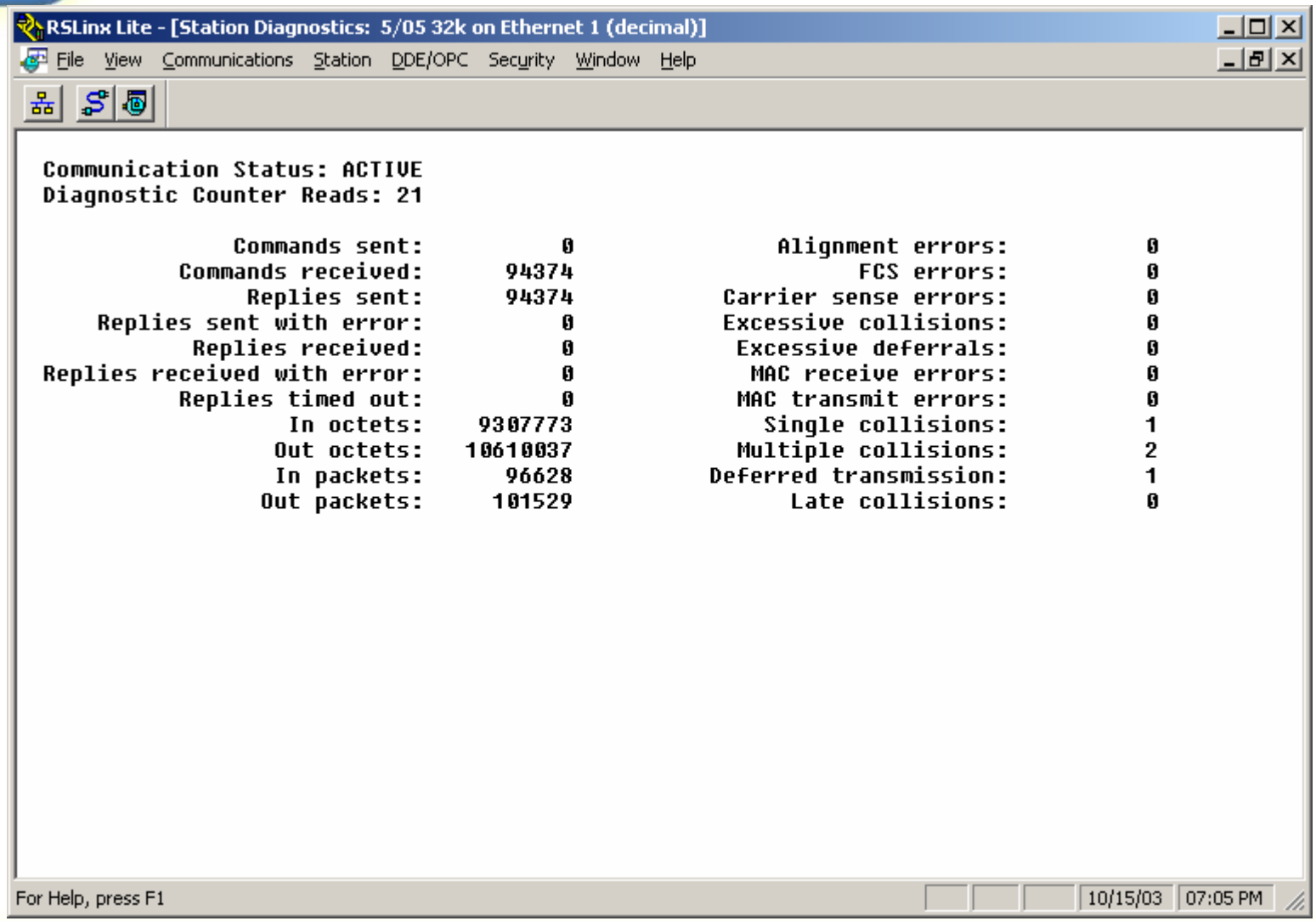
DRIVER DIAGNOSTICS



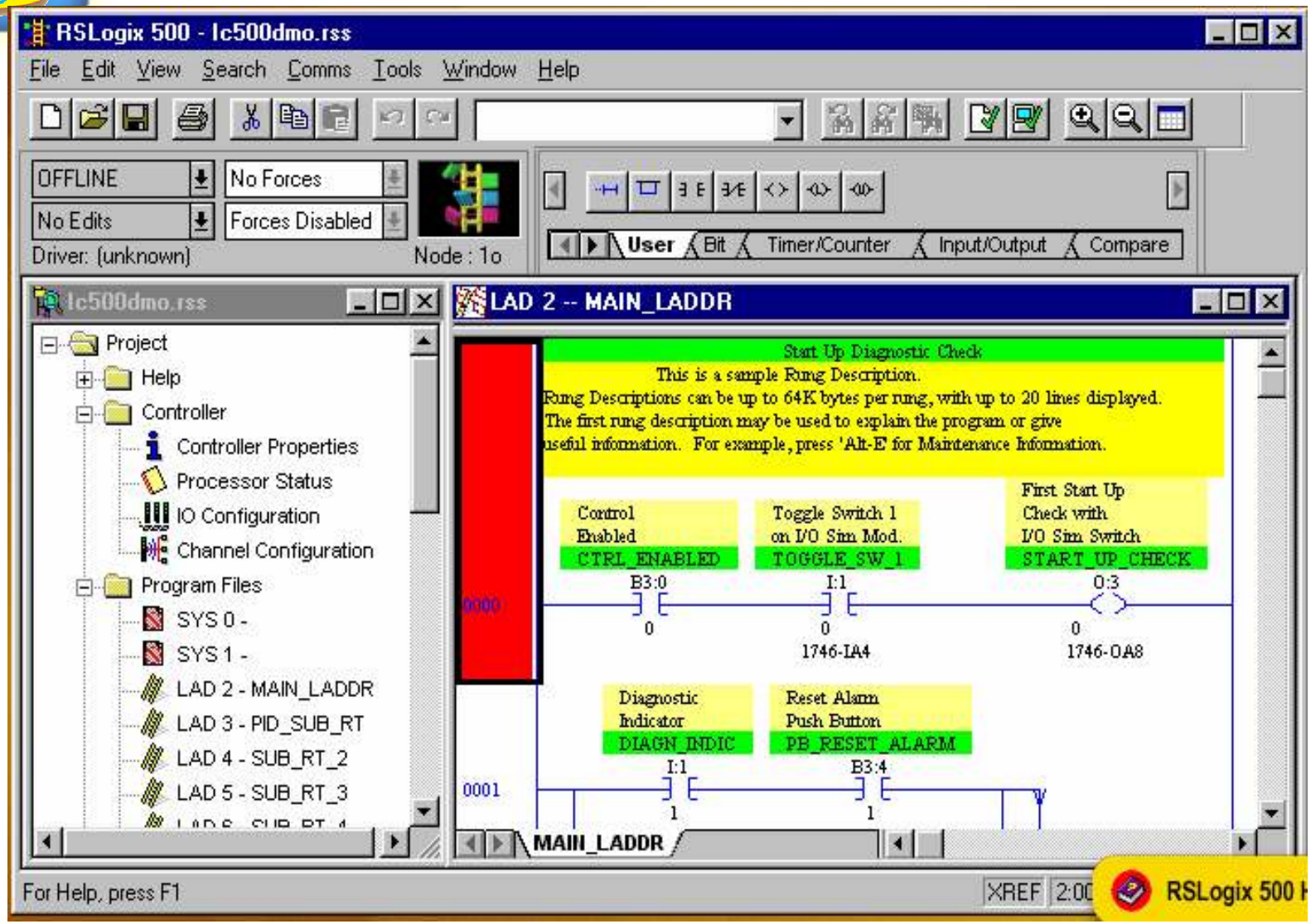
DRIVER DIAGNOSTICS



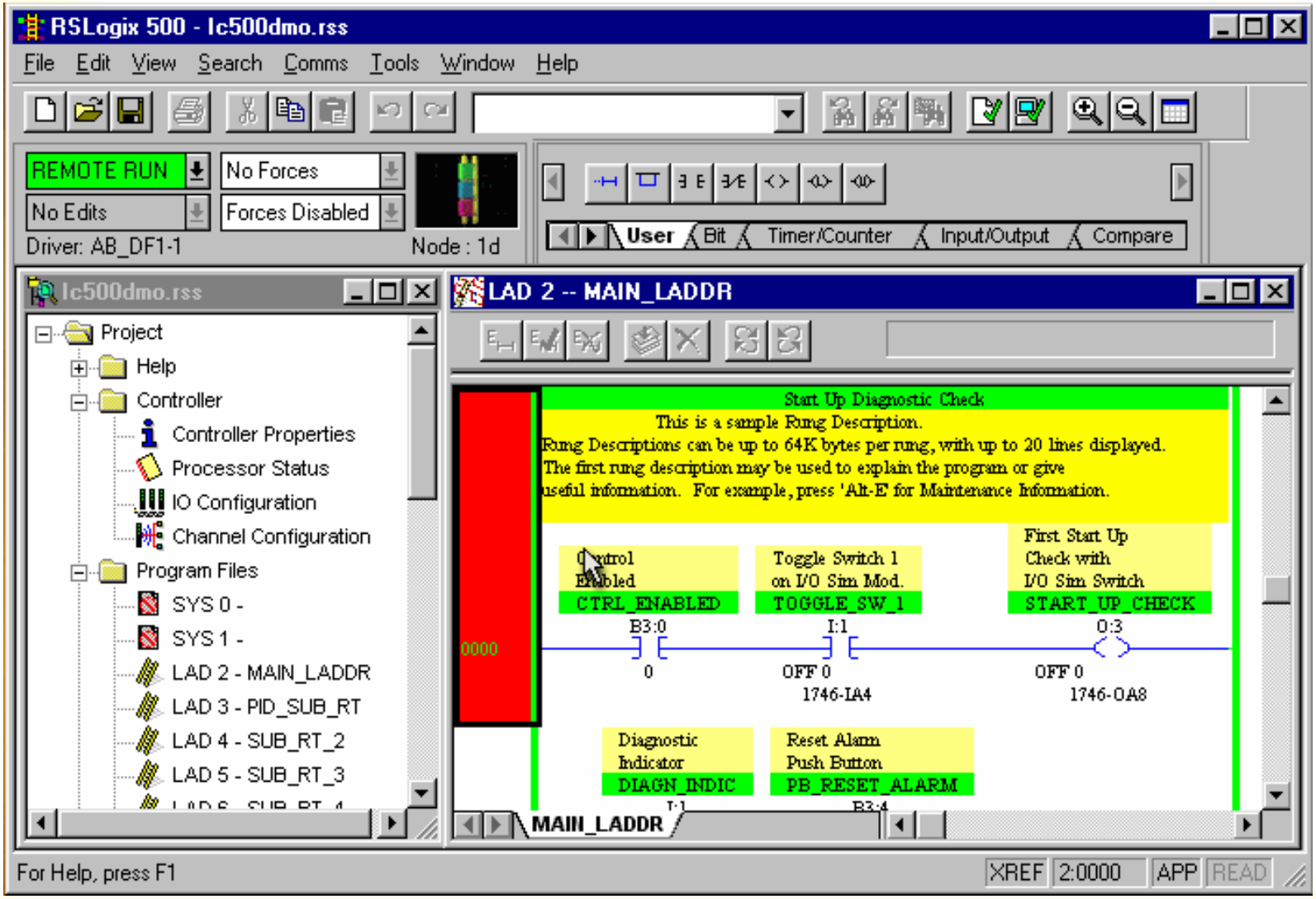
DRIVER DIAGNOSTICS



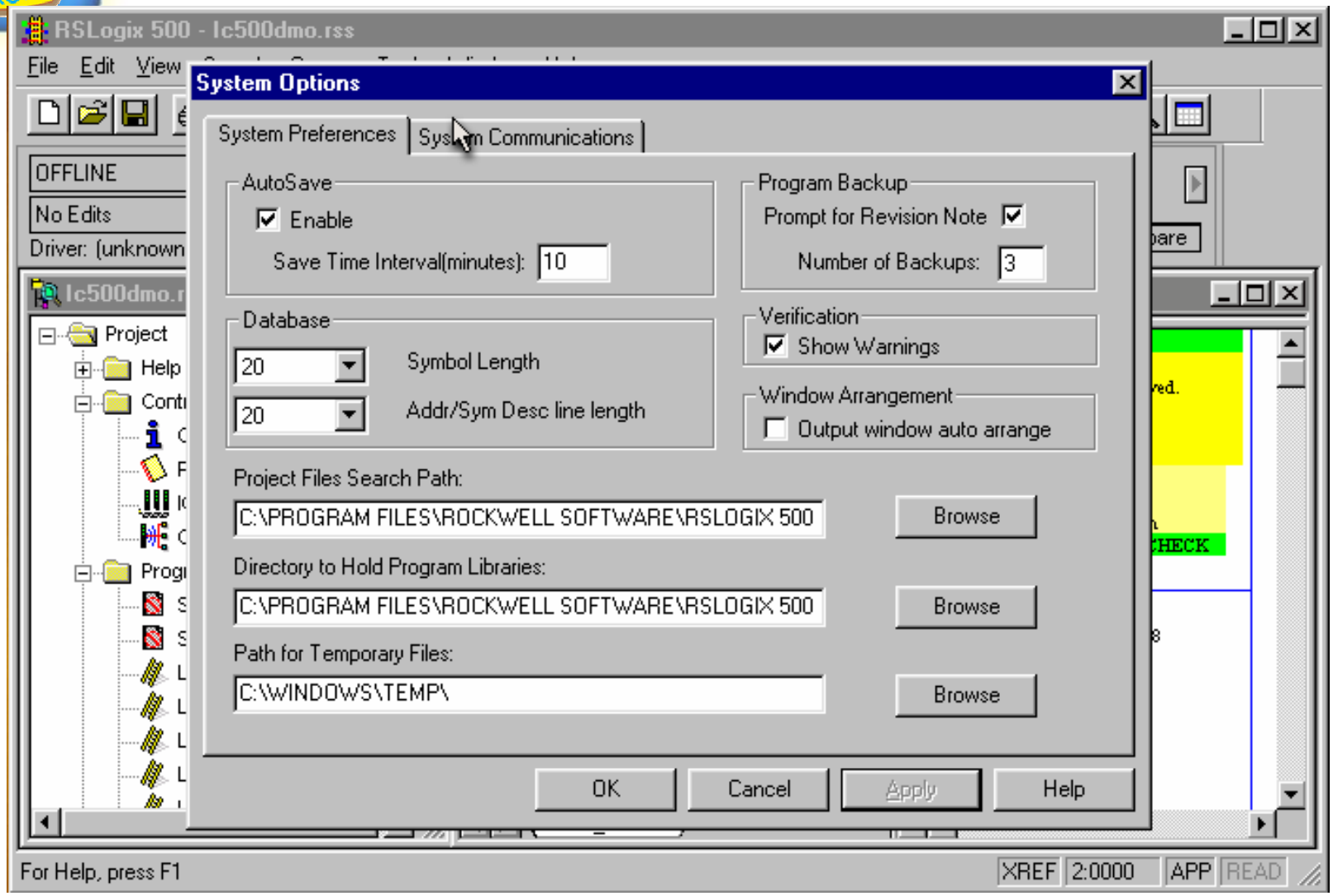
PLC PROGRAMMING SOFTWARE



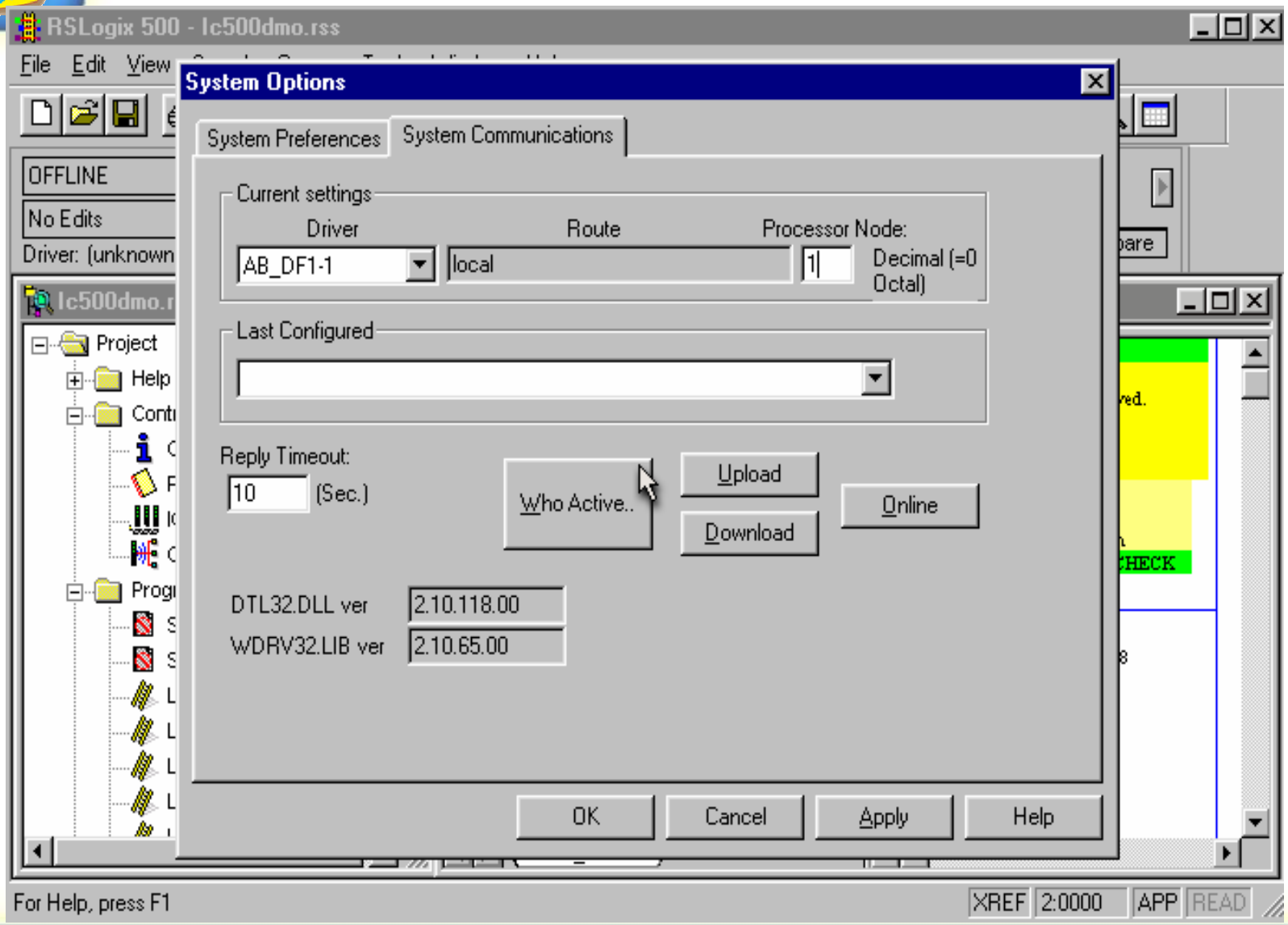
PLC ON-LINE PROGRAM



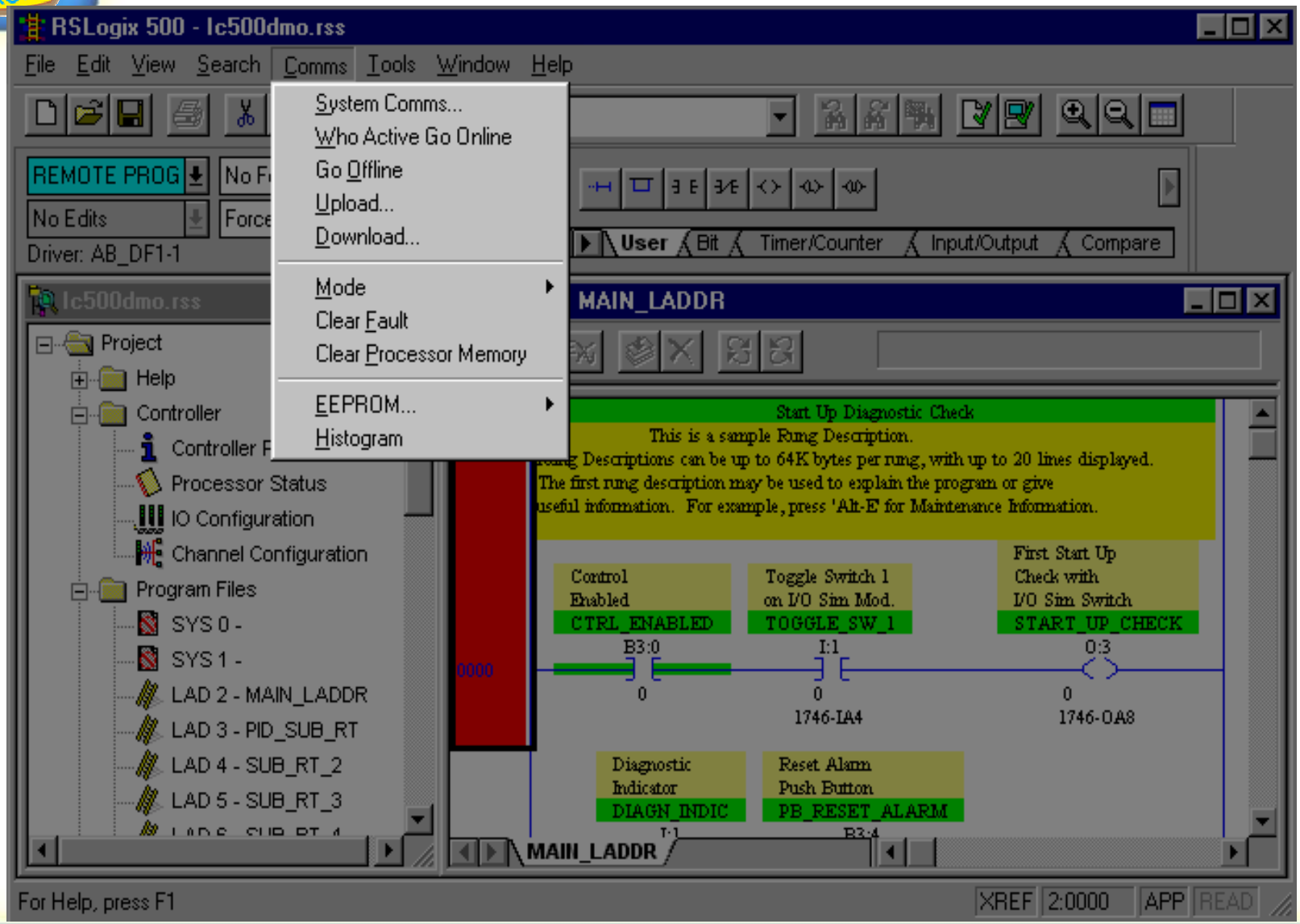
DOWN-LOAD OPTION



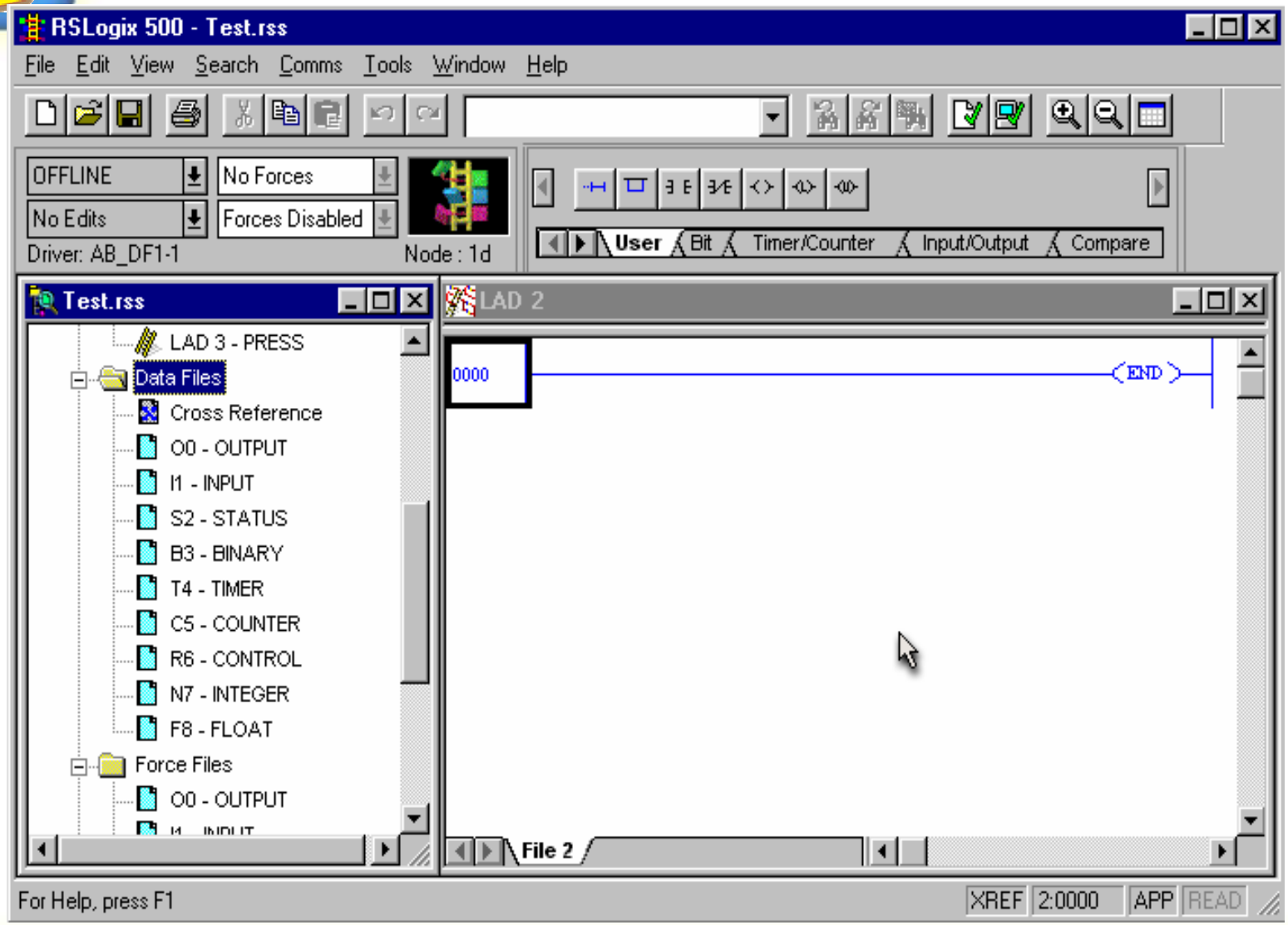
PLC COMMUNICATION SET-UP



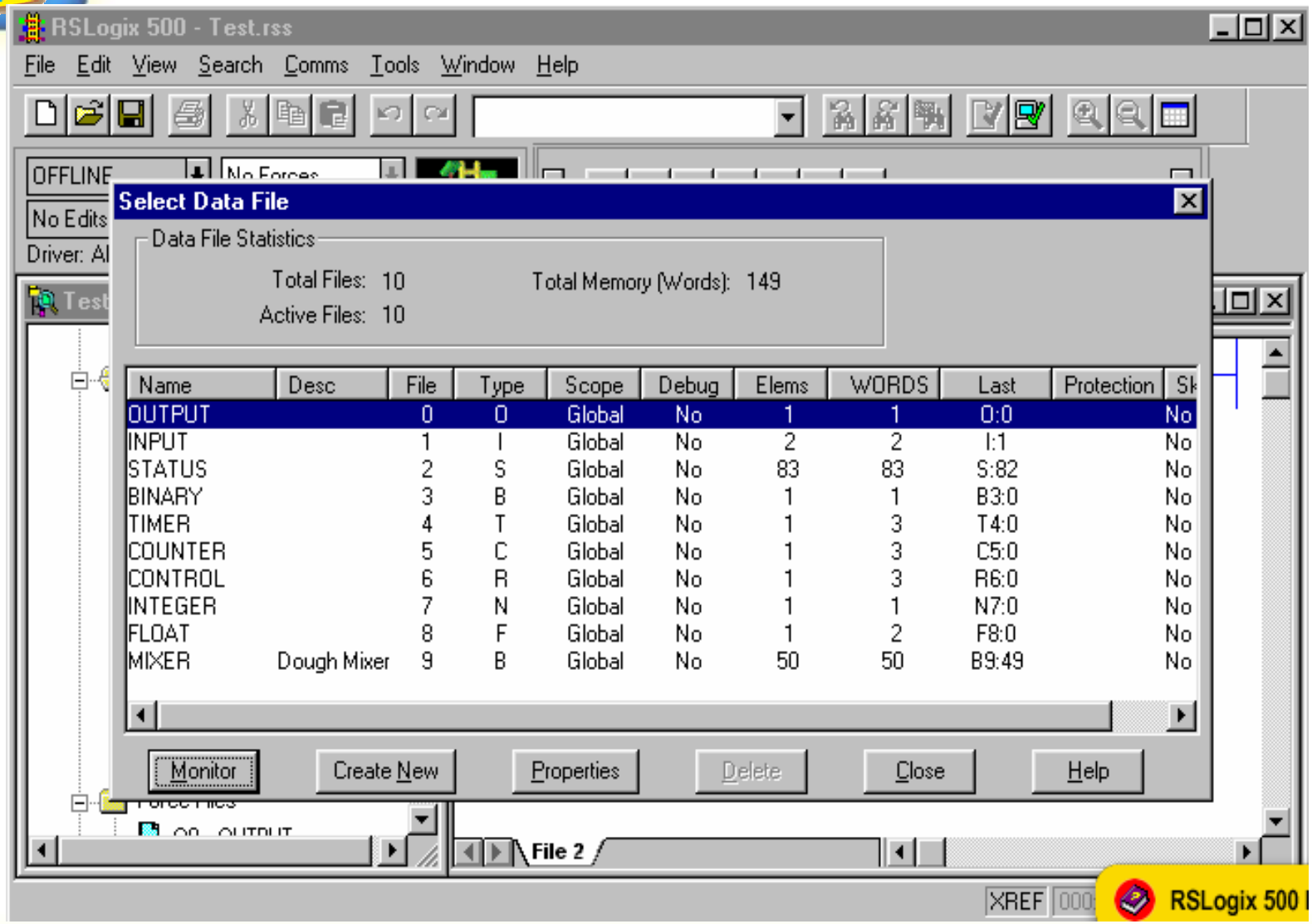
PLC COMMUNICATION OPTION



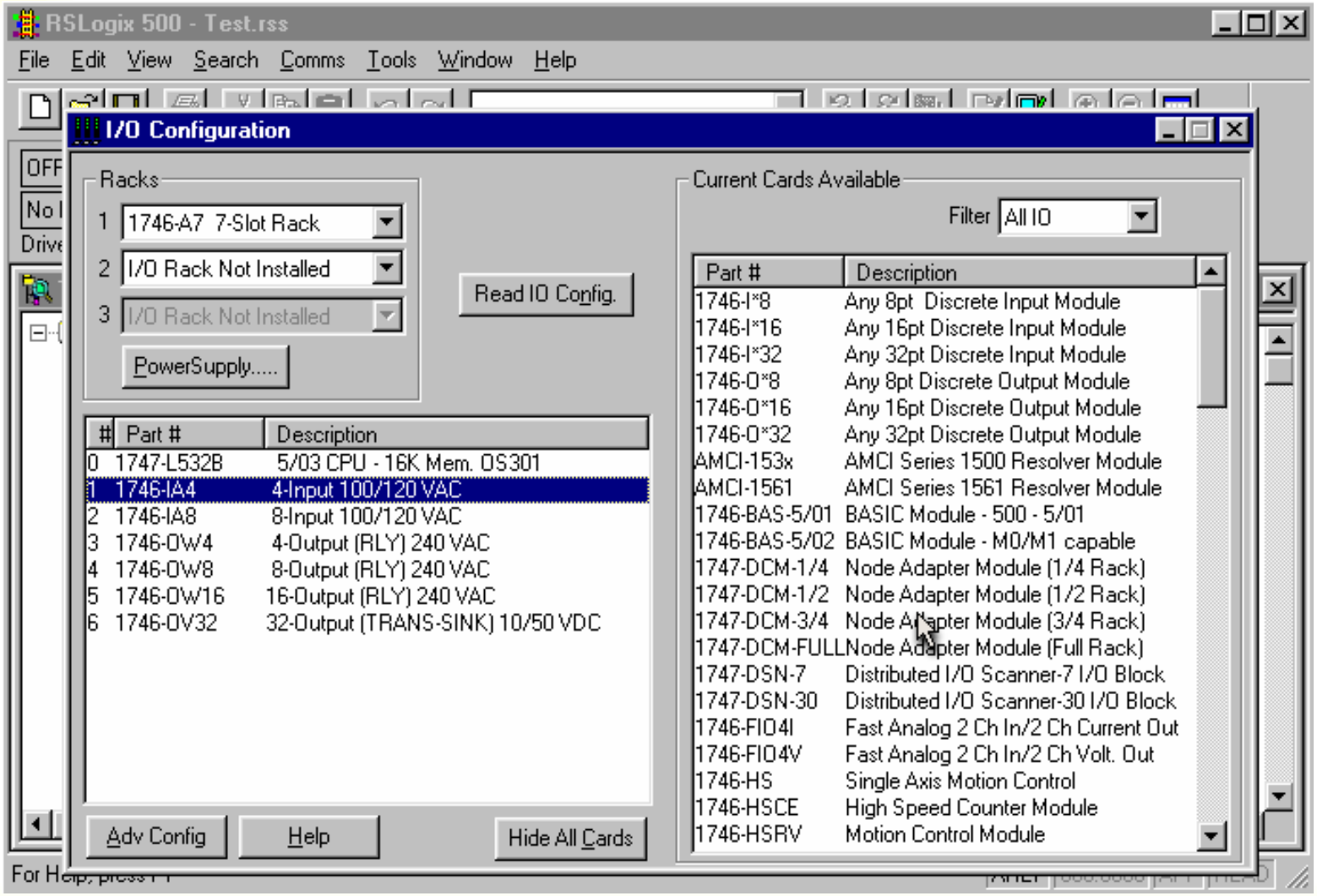
PLC DATA FILES



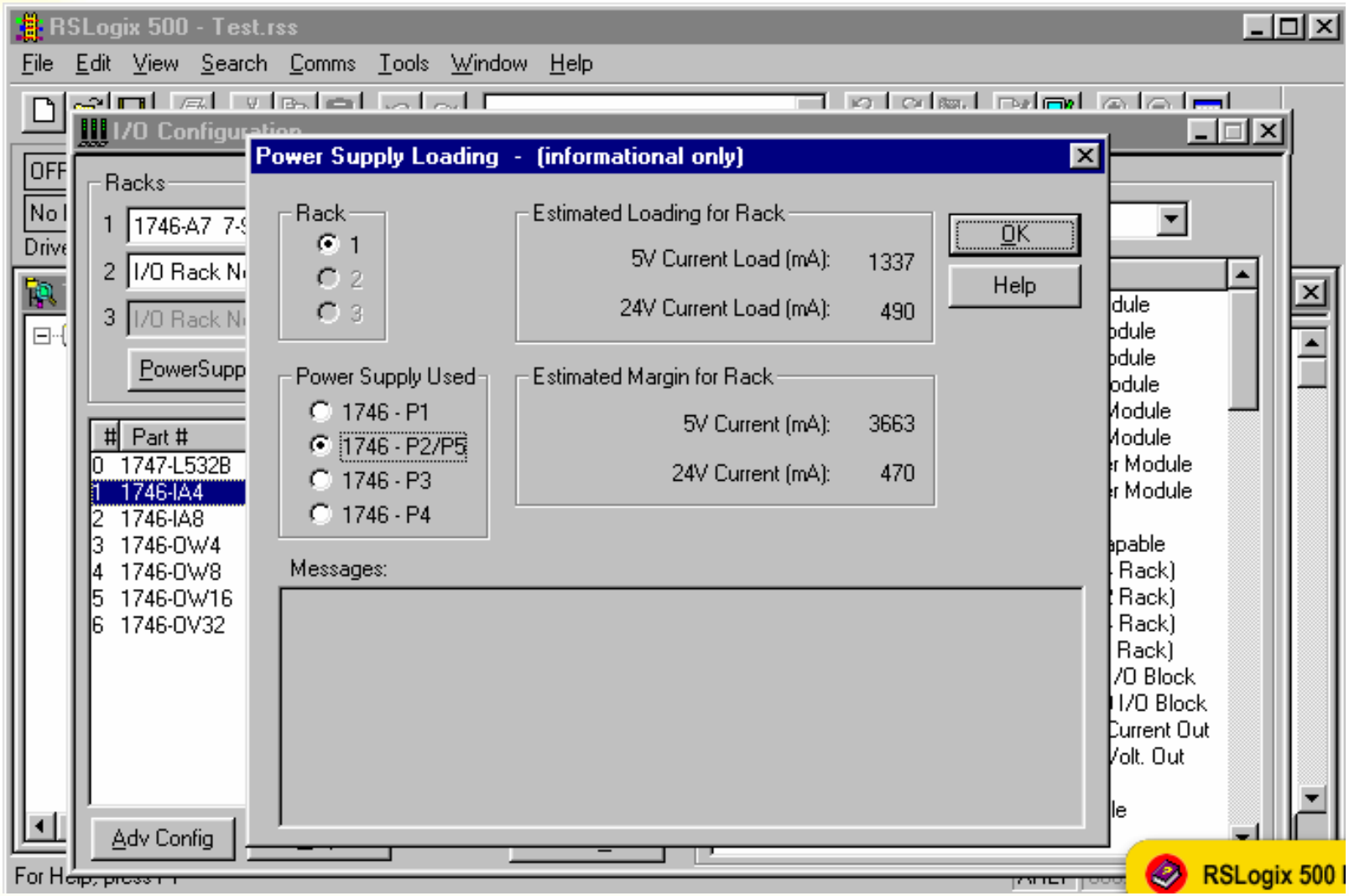
PLC DATA FILE DISPLAY



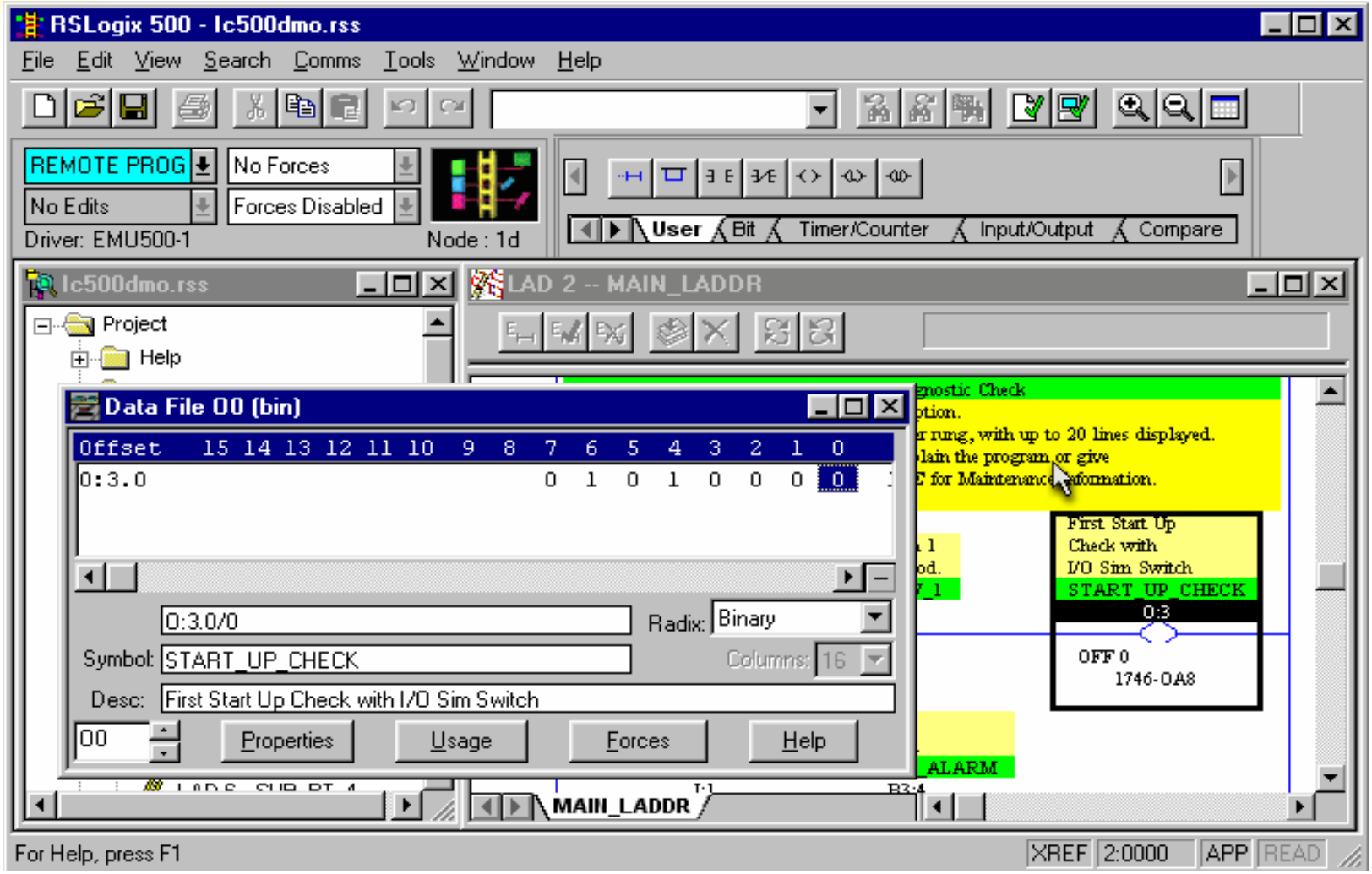
I/O Configuration



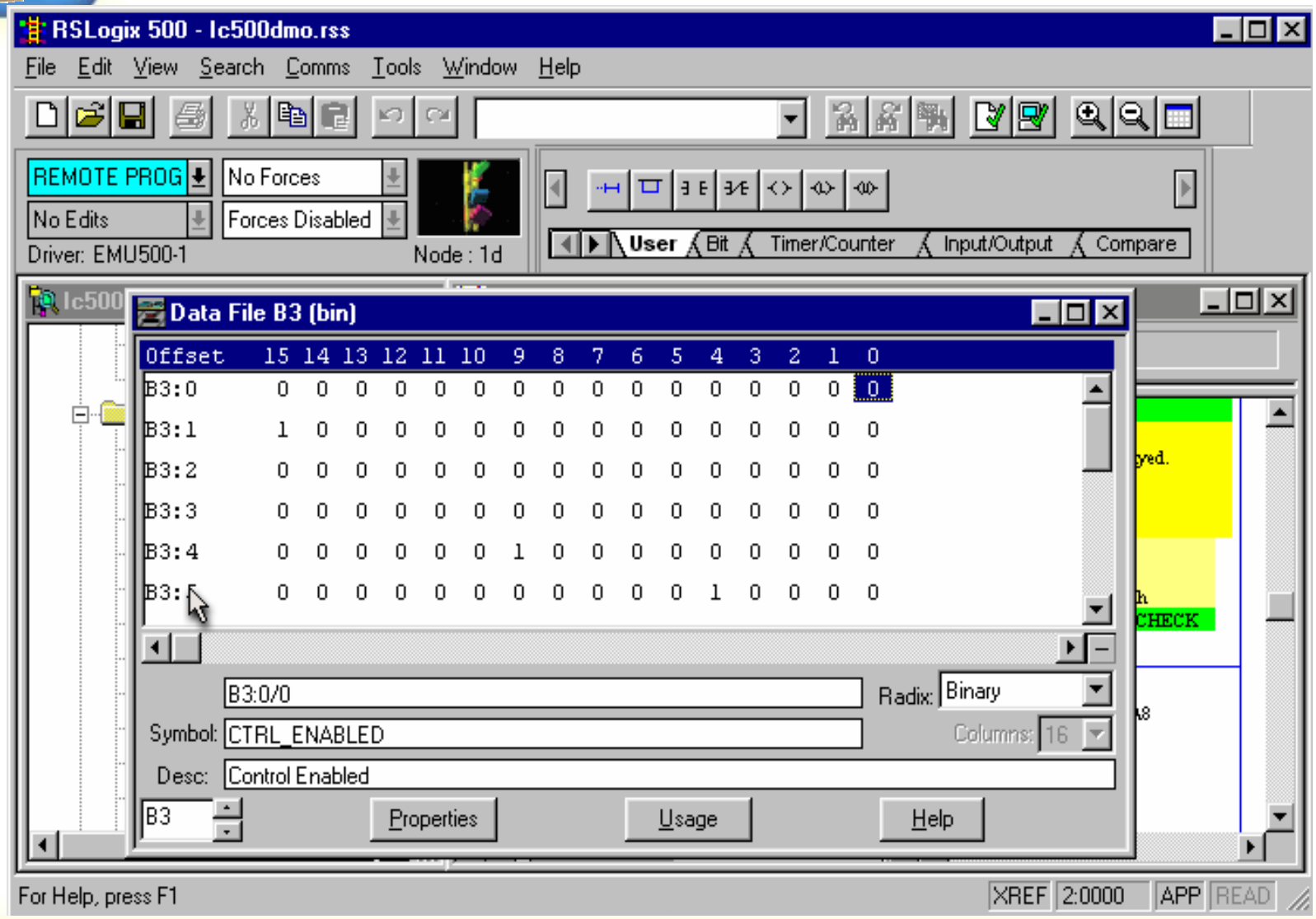
POWER SUPPLY OVERLOADING



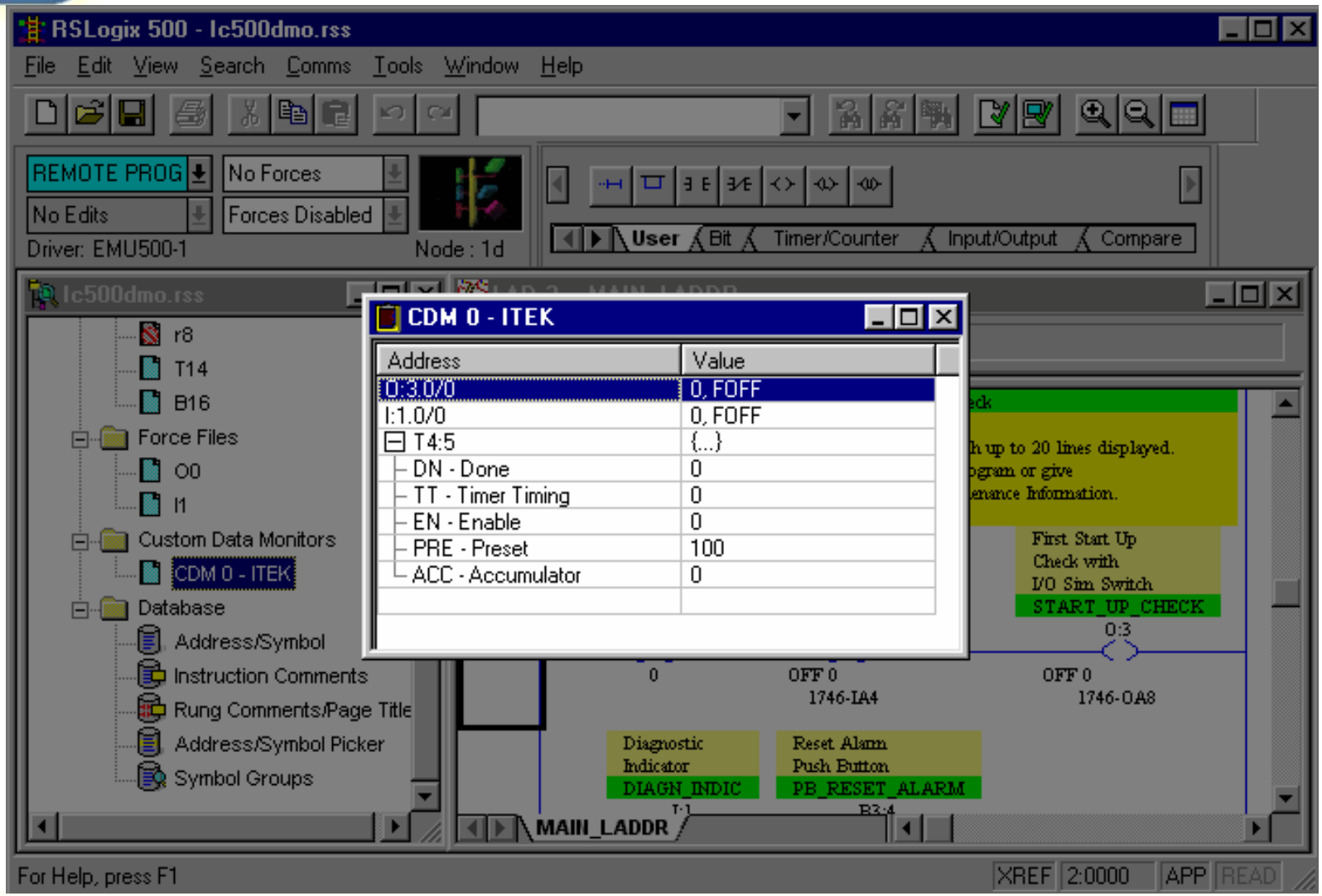
DATA MONITORING



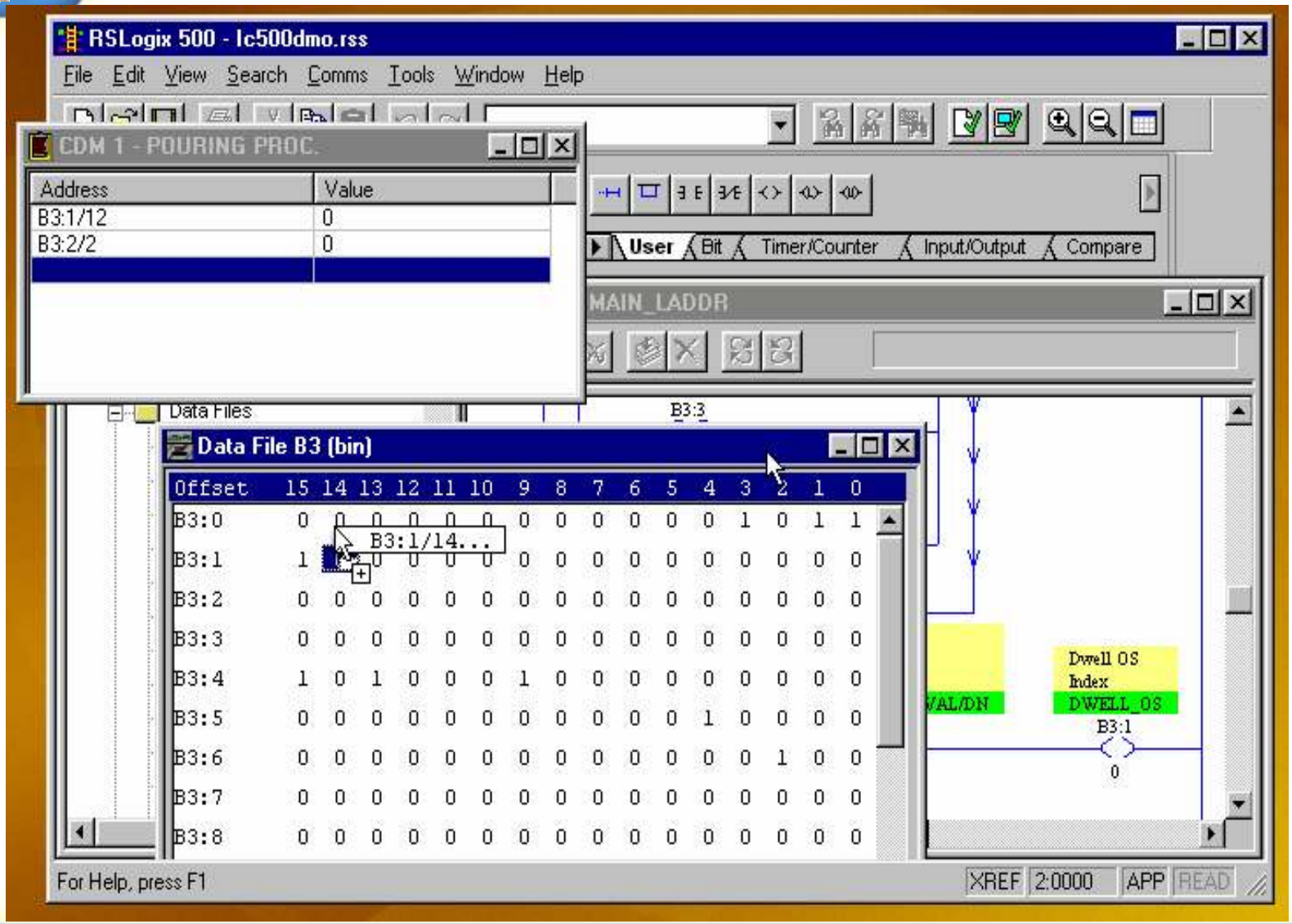
DATA FILE MONITORING



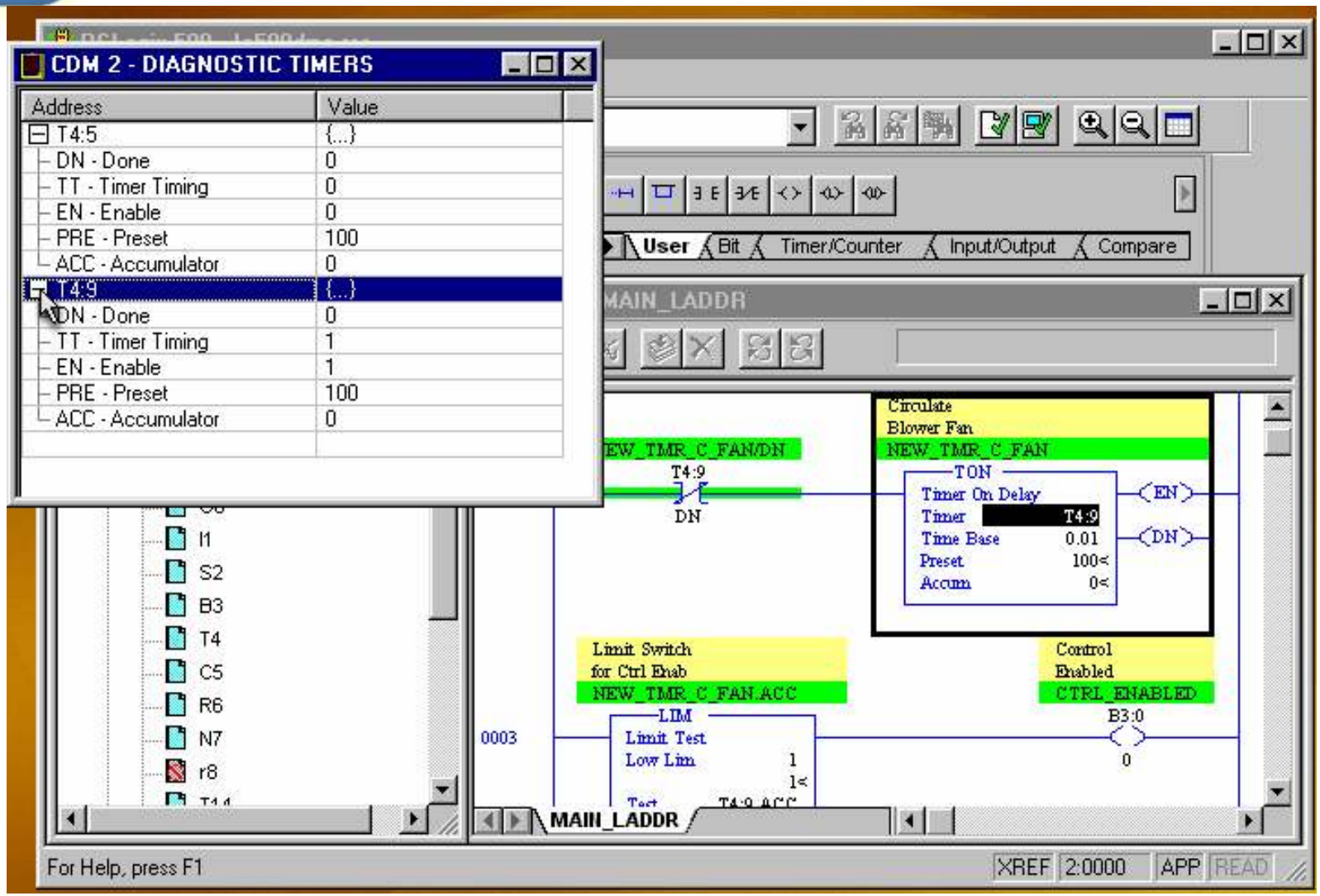
CUSTOM DATA MONITOR



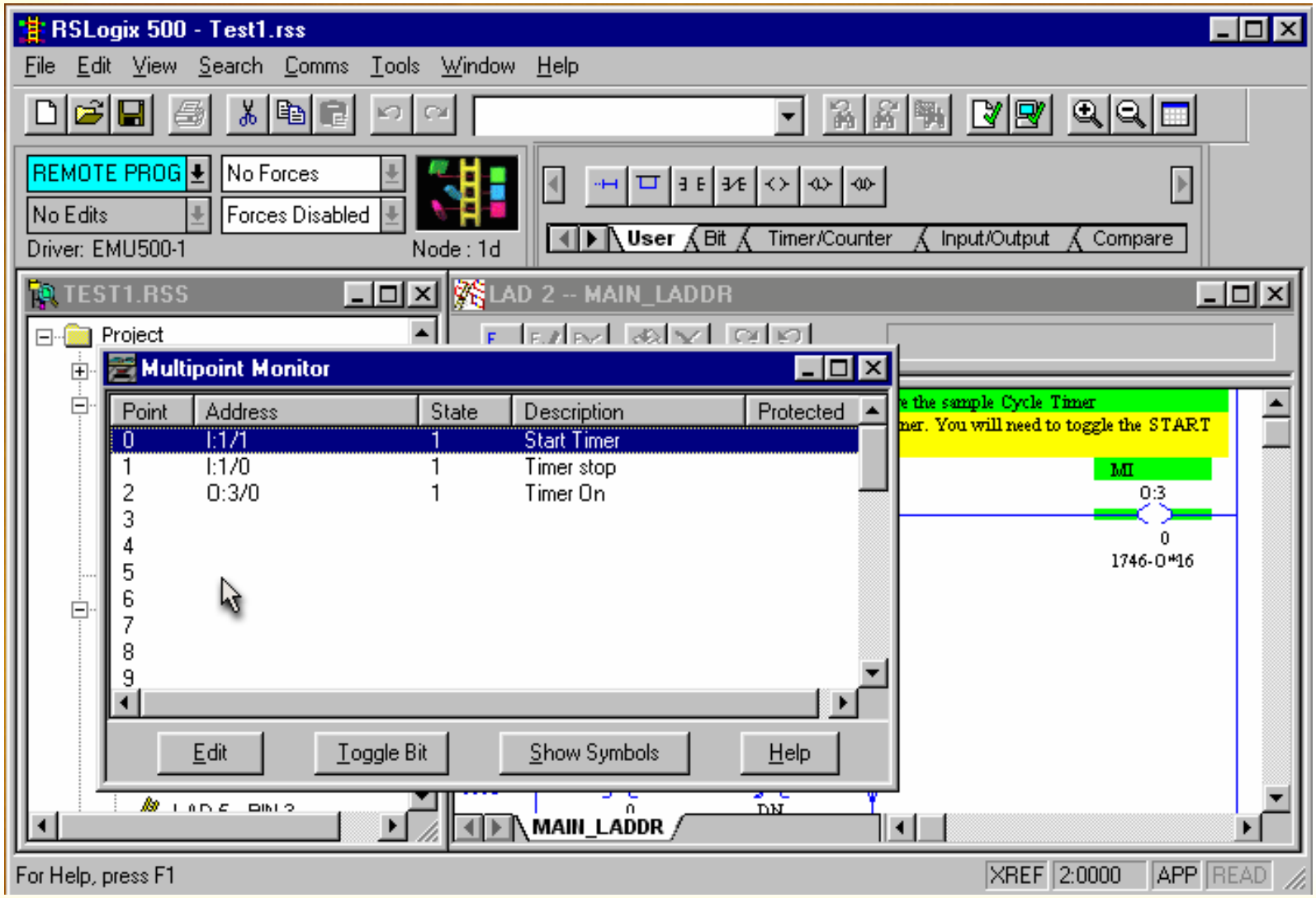
CUSTOM DATA MONITOR



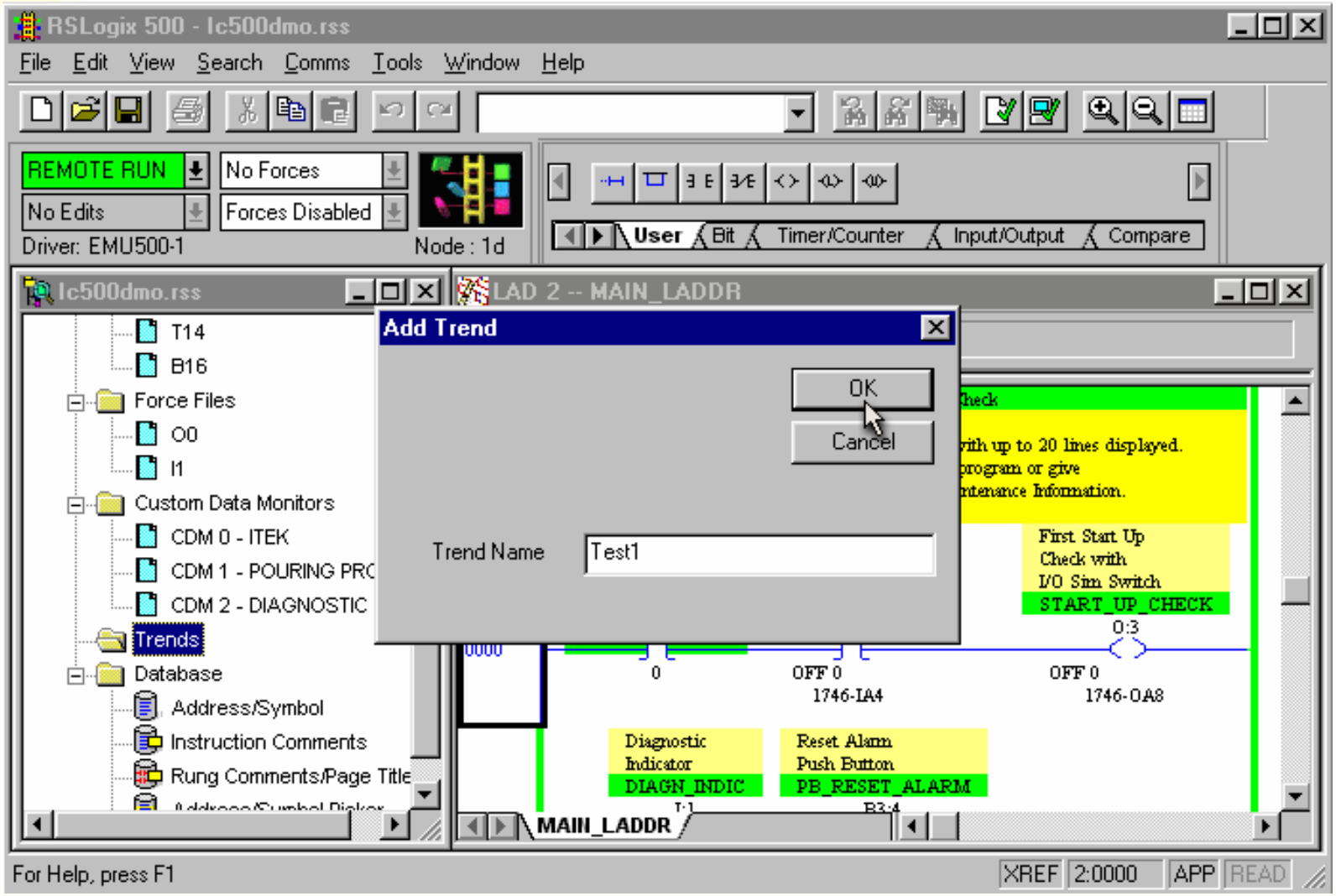
DATA DIAGNOSTICS



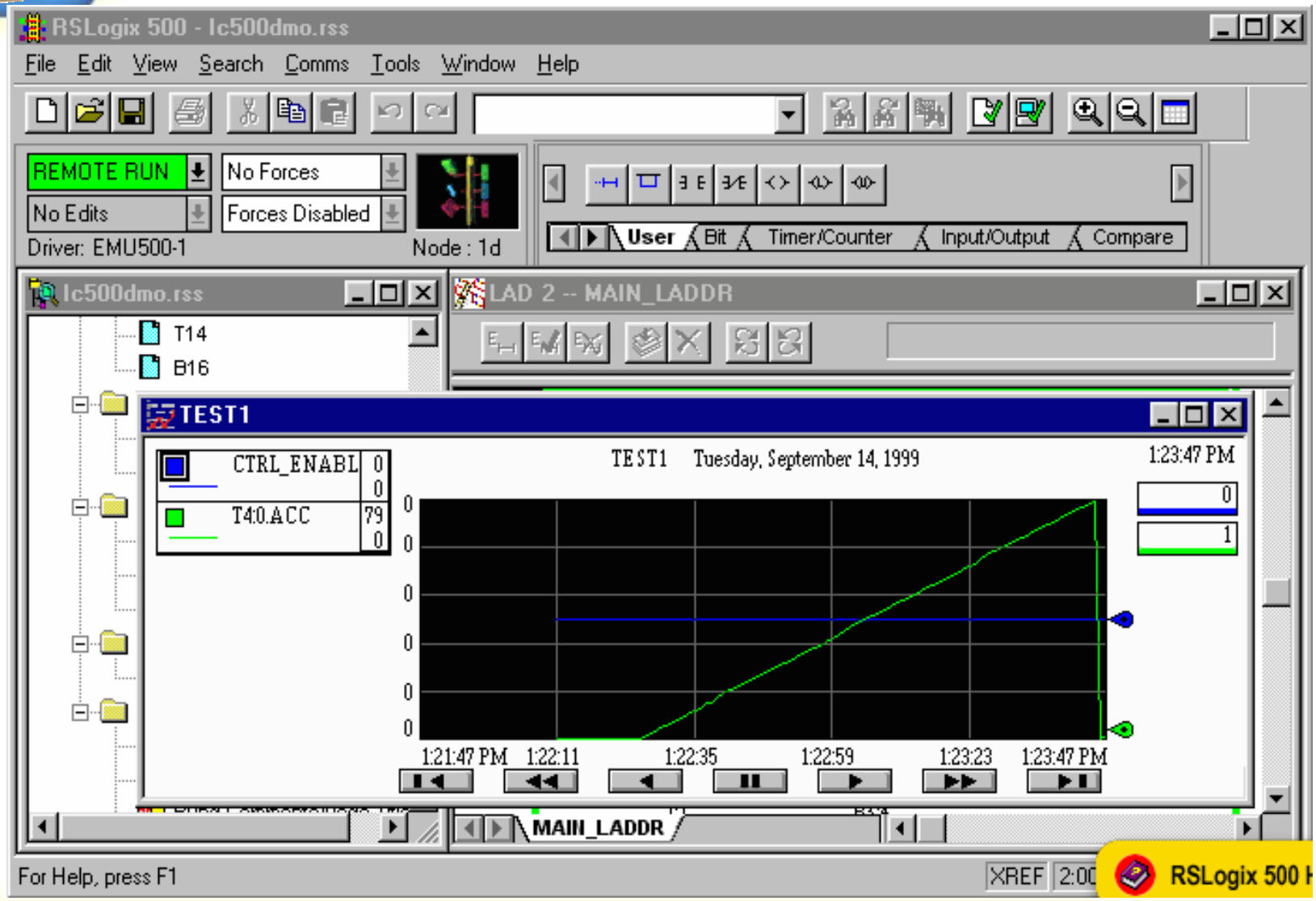
MULTIPOINT MONITORS



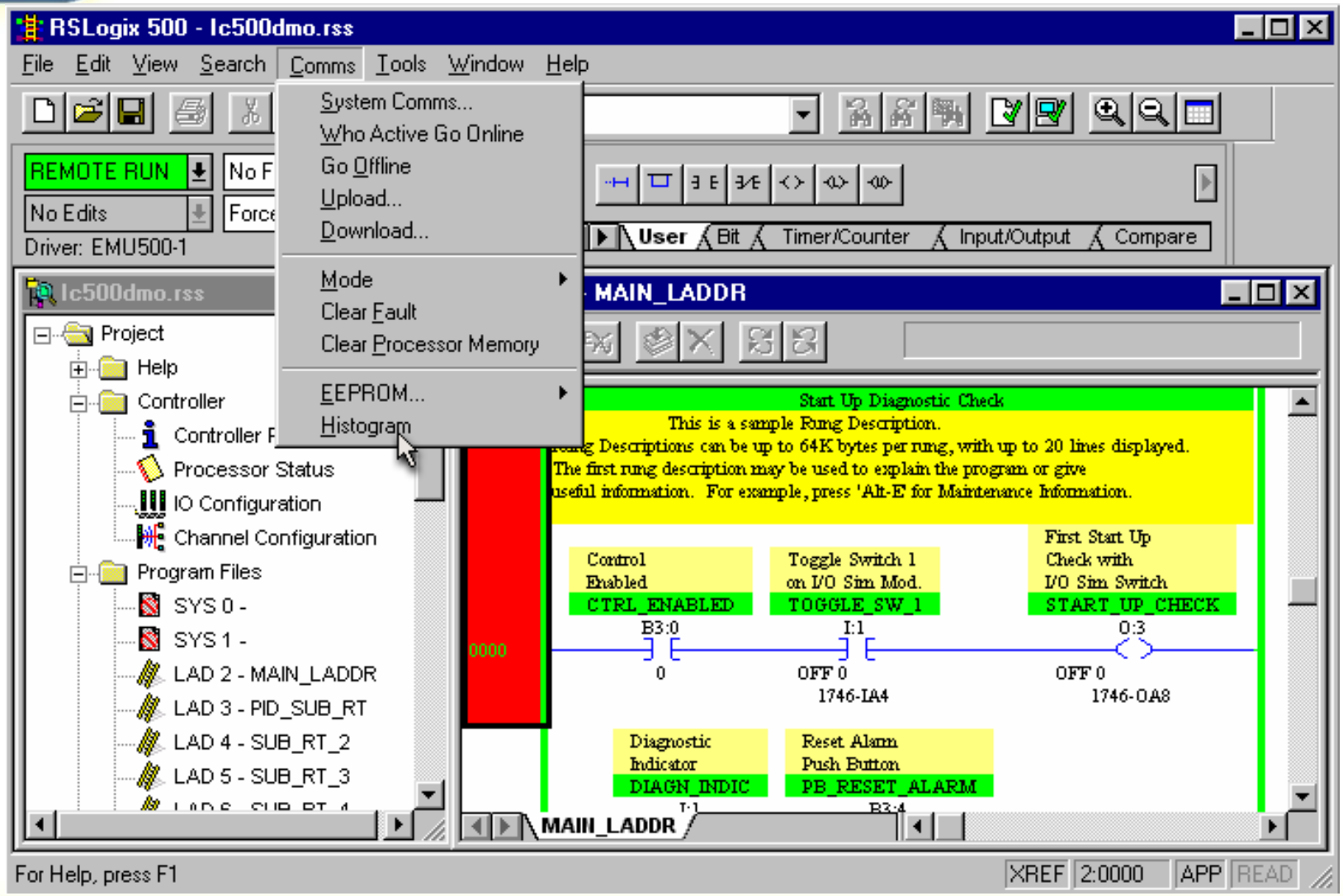
TREND MONITORING



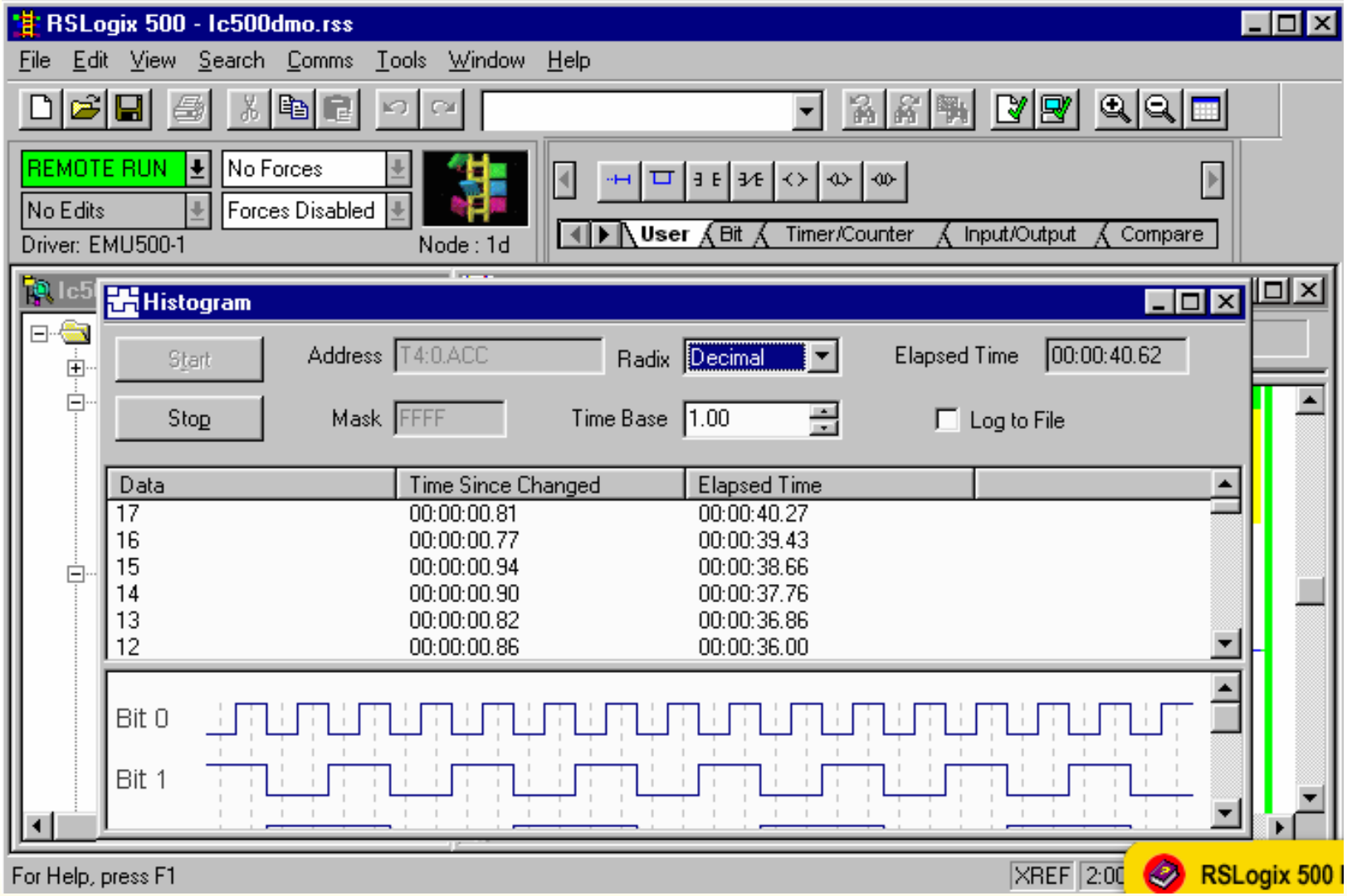
TREND MONITORING



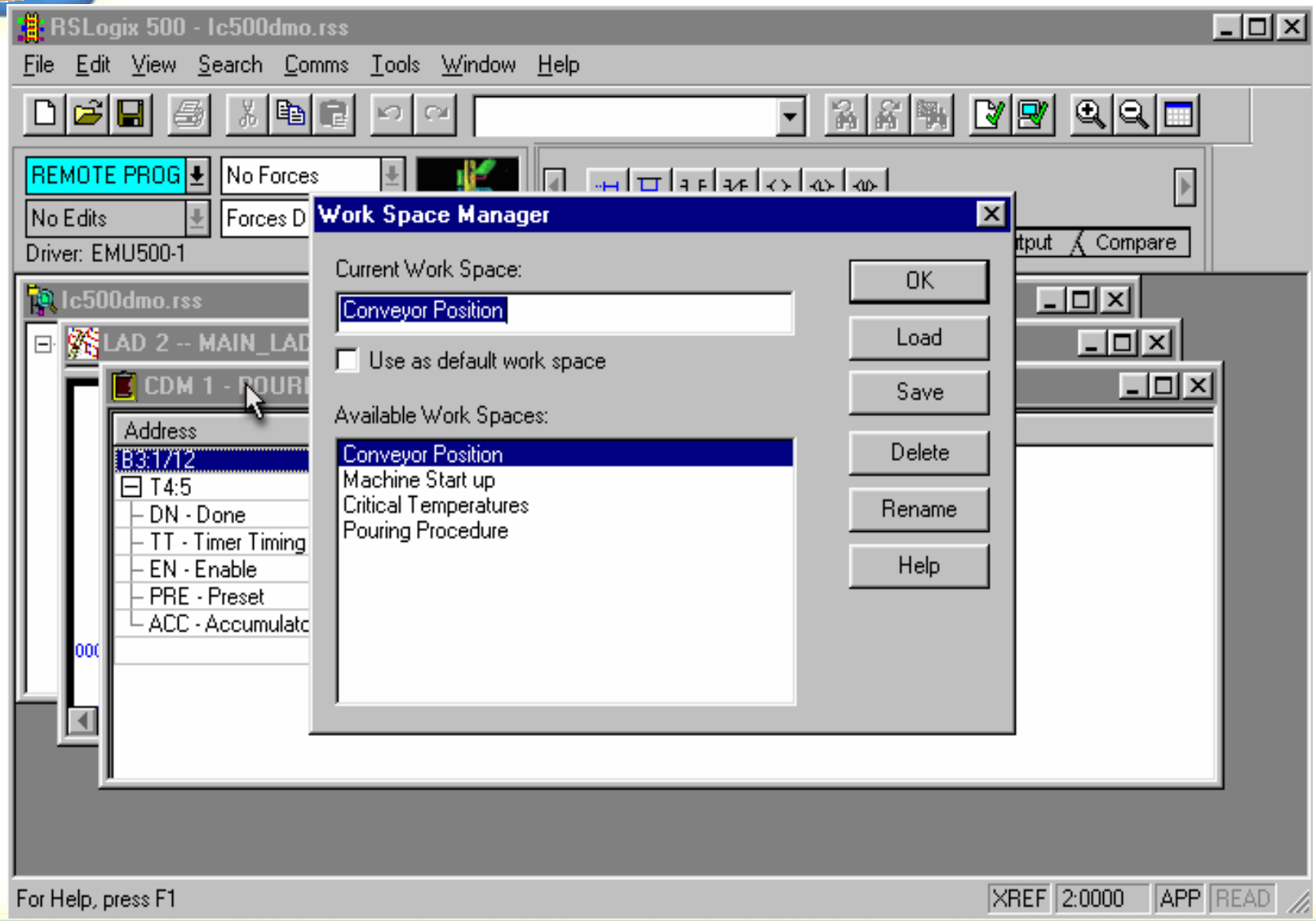
TREND DATA MONITORING



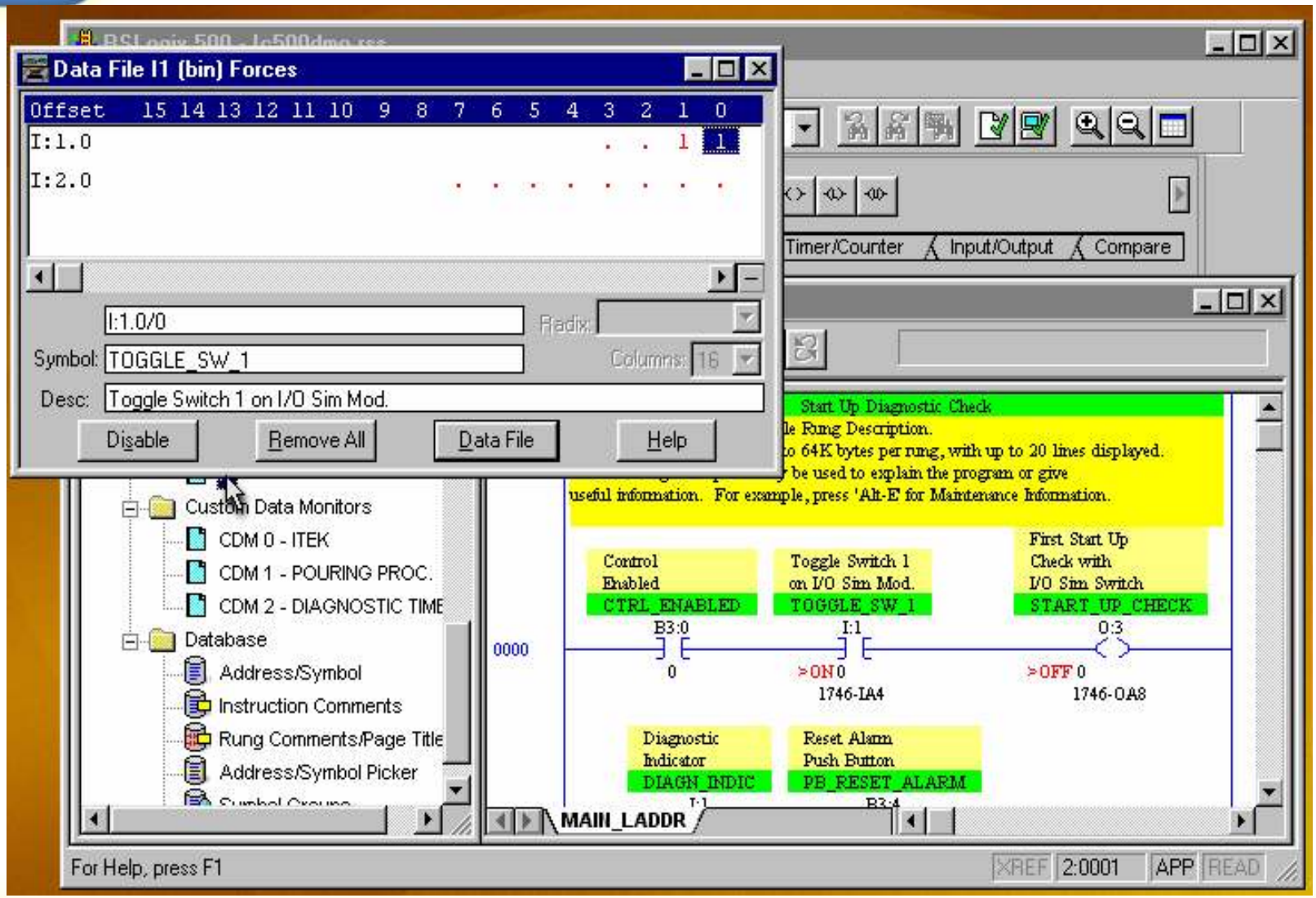
TREND DATA MONITORING



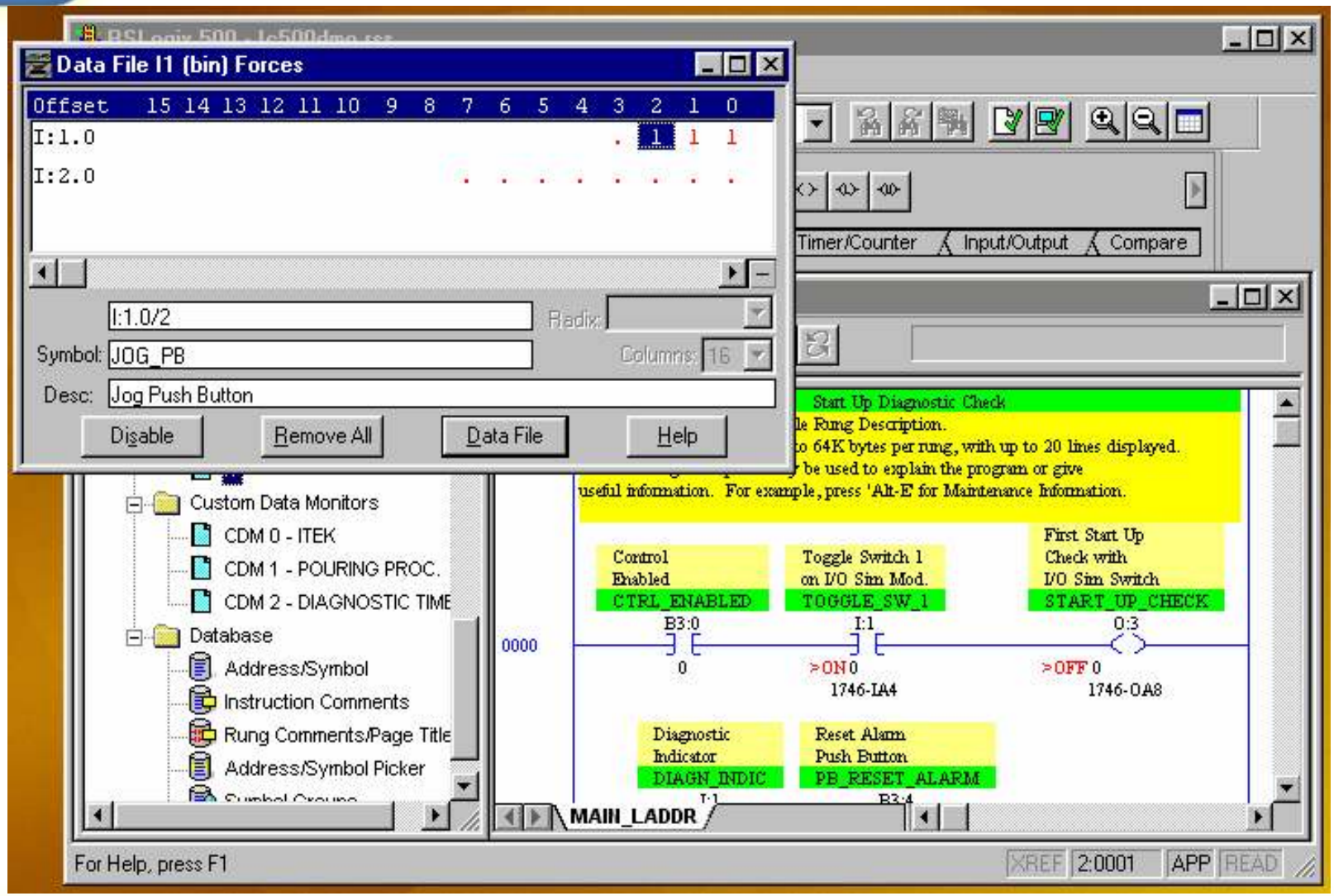
WORKSPACE MANAGER



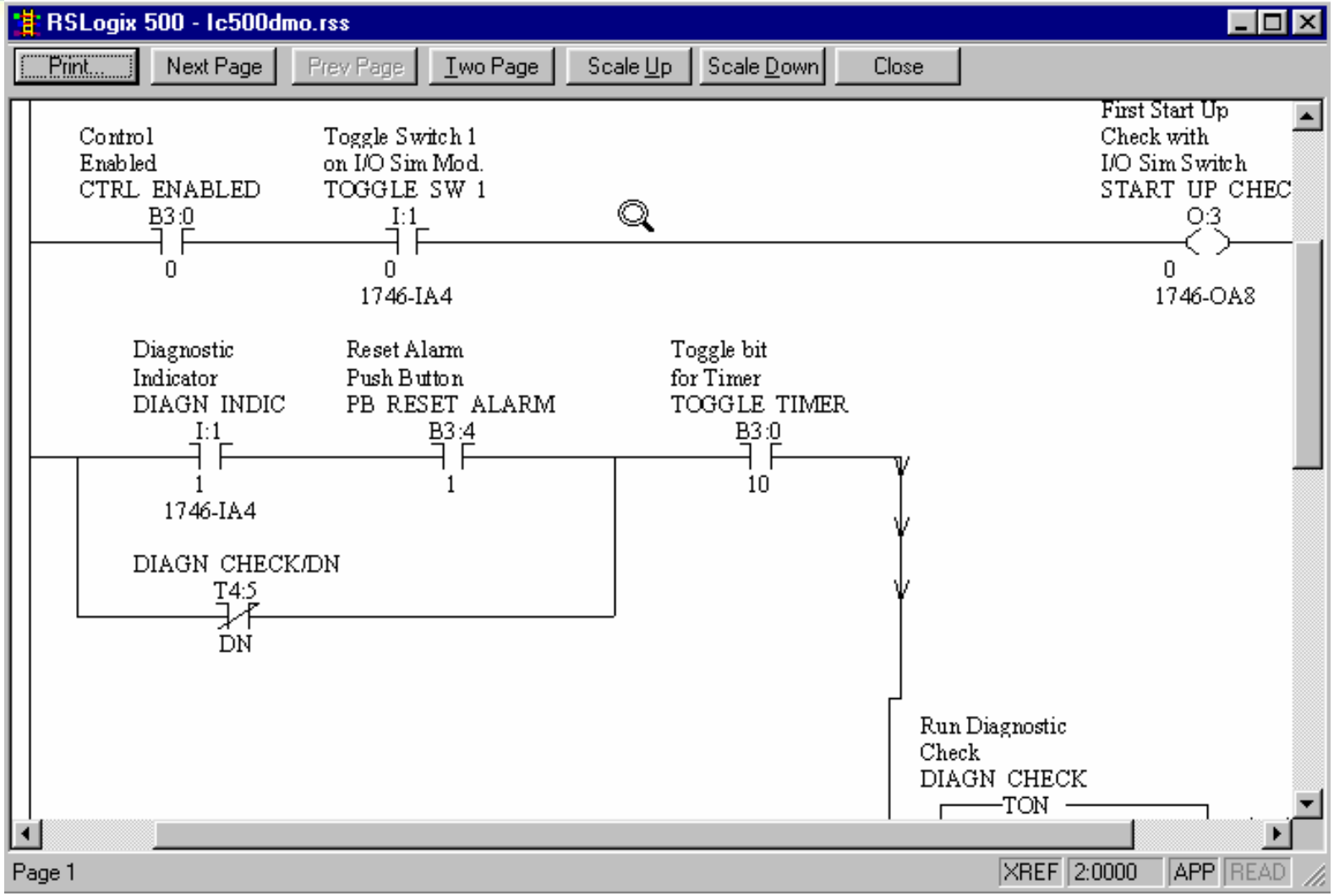
LOGIC FORCING



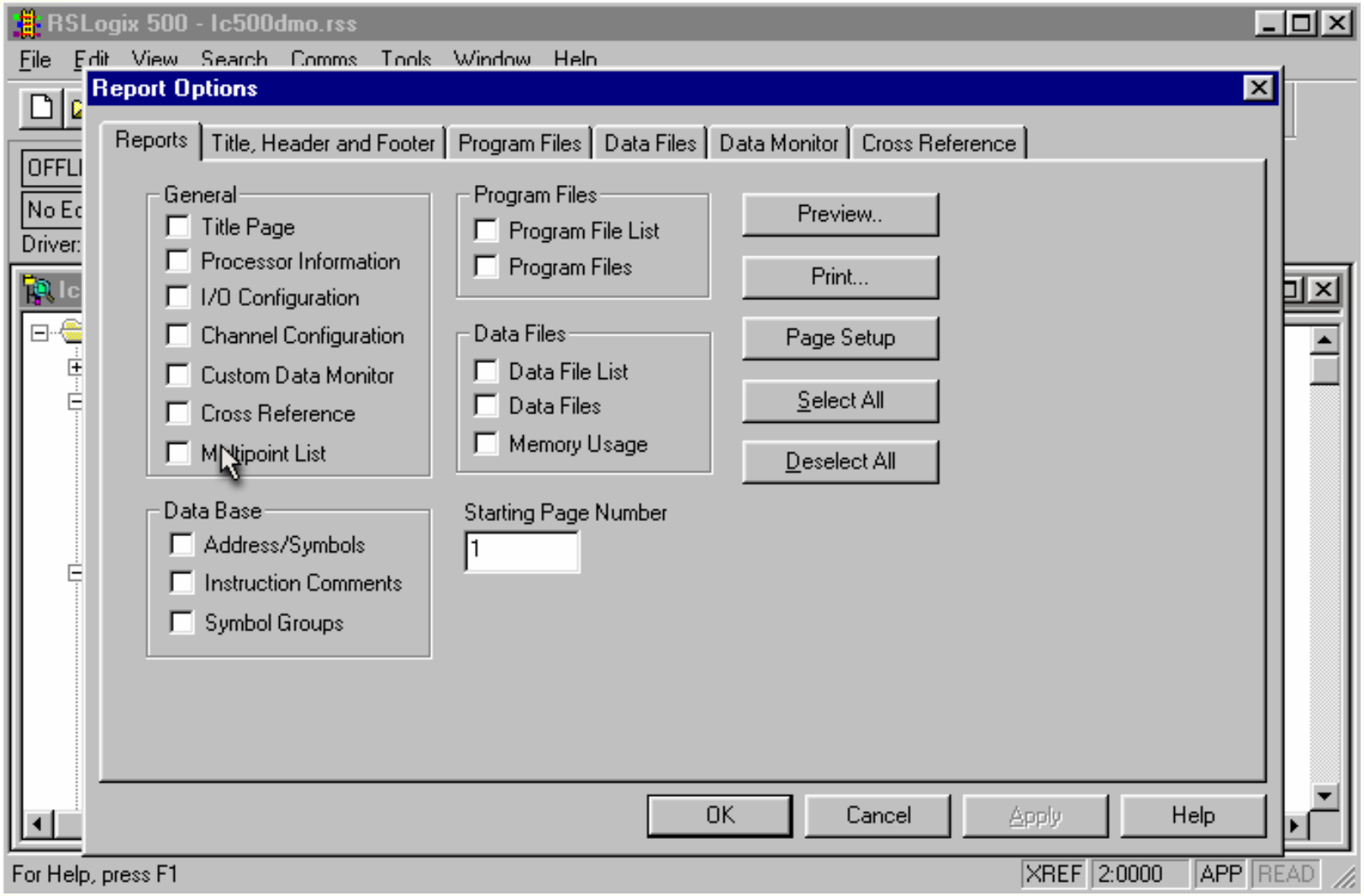
LOGIC FORCING



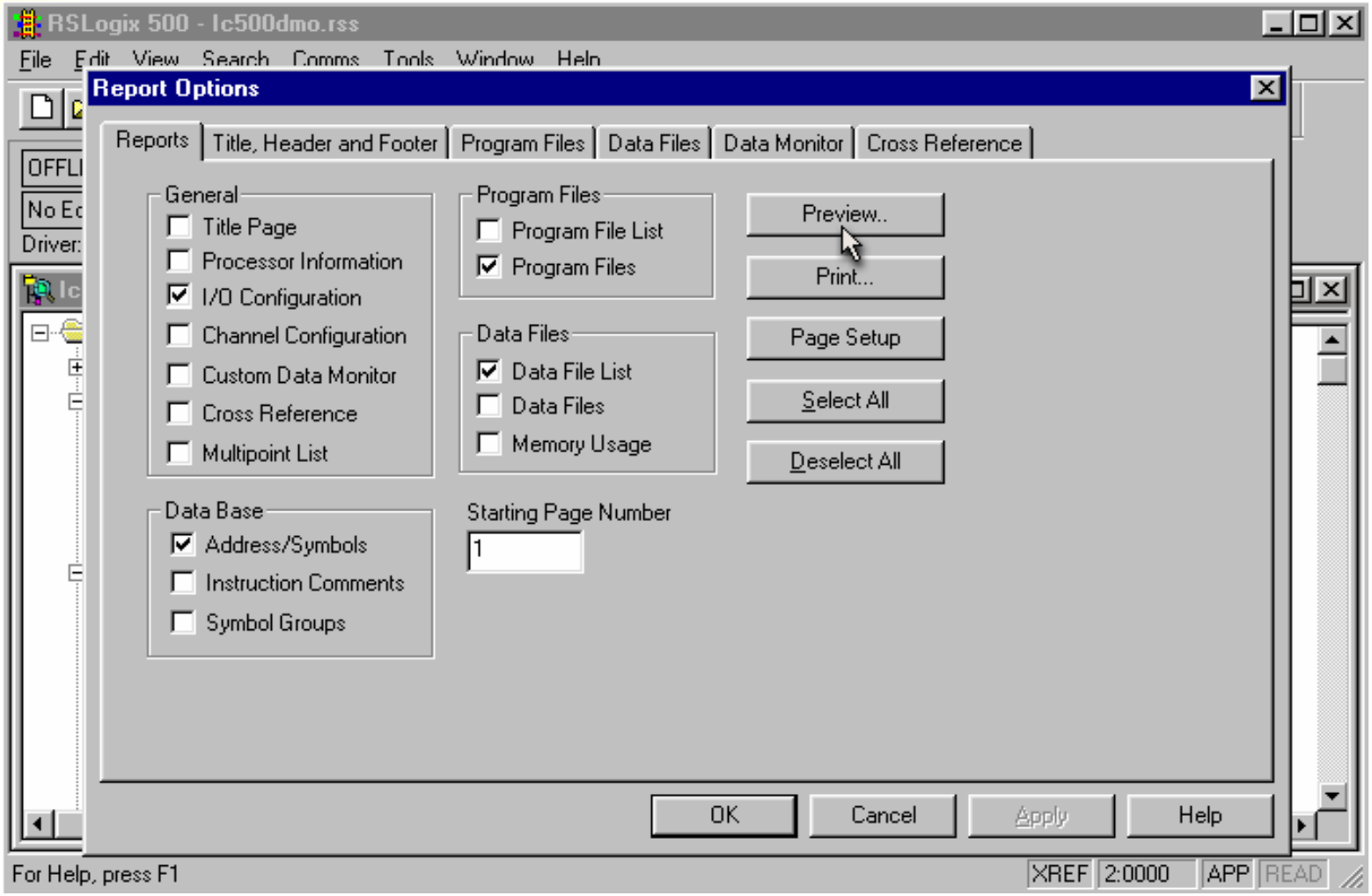
REPORTING PREVIEW



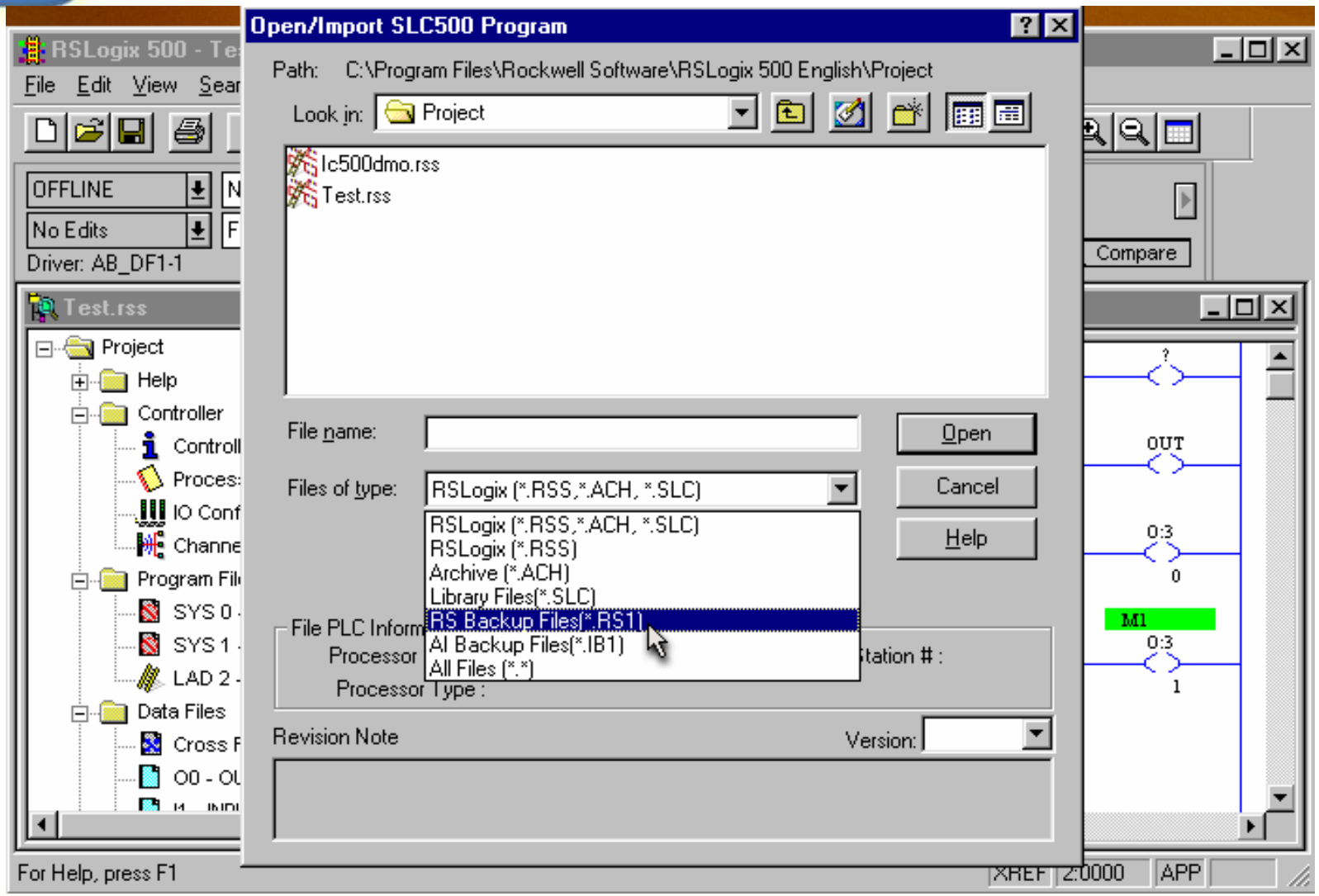
REPORT OPTIONS



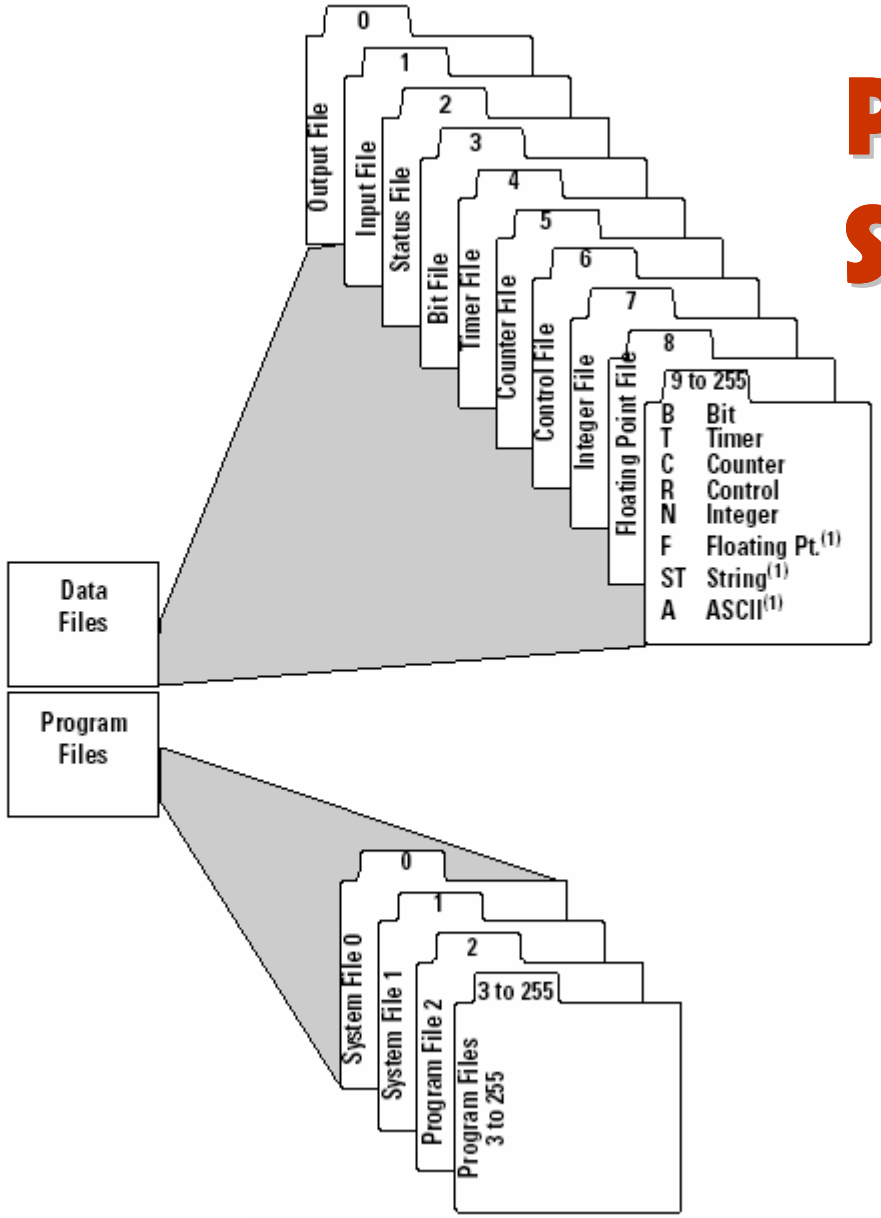
REPORT OPTIONS



BACKUP FILES



PLC File Structure



I/O ADDRESSING FORMATS

O:3/15	Output 15, slot3
O:5/0	Output 0, slot 5
O:10/11	Output 11, slot 10
I:7/8	Input 8, slot 7
I:2.1/3	Input 3, slot 2, word 1
O:5	Output word 0, slot 5
O:5.1	Output word 1, slot 5
I:8	Input word 0, slot 8

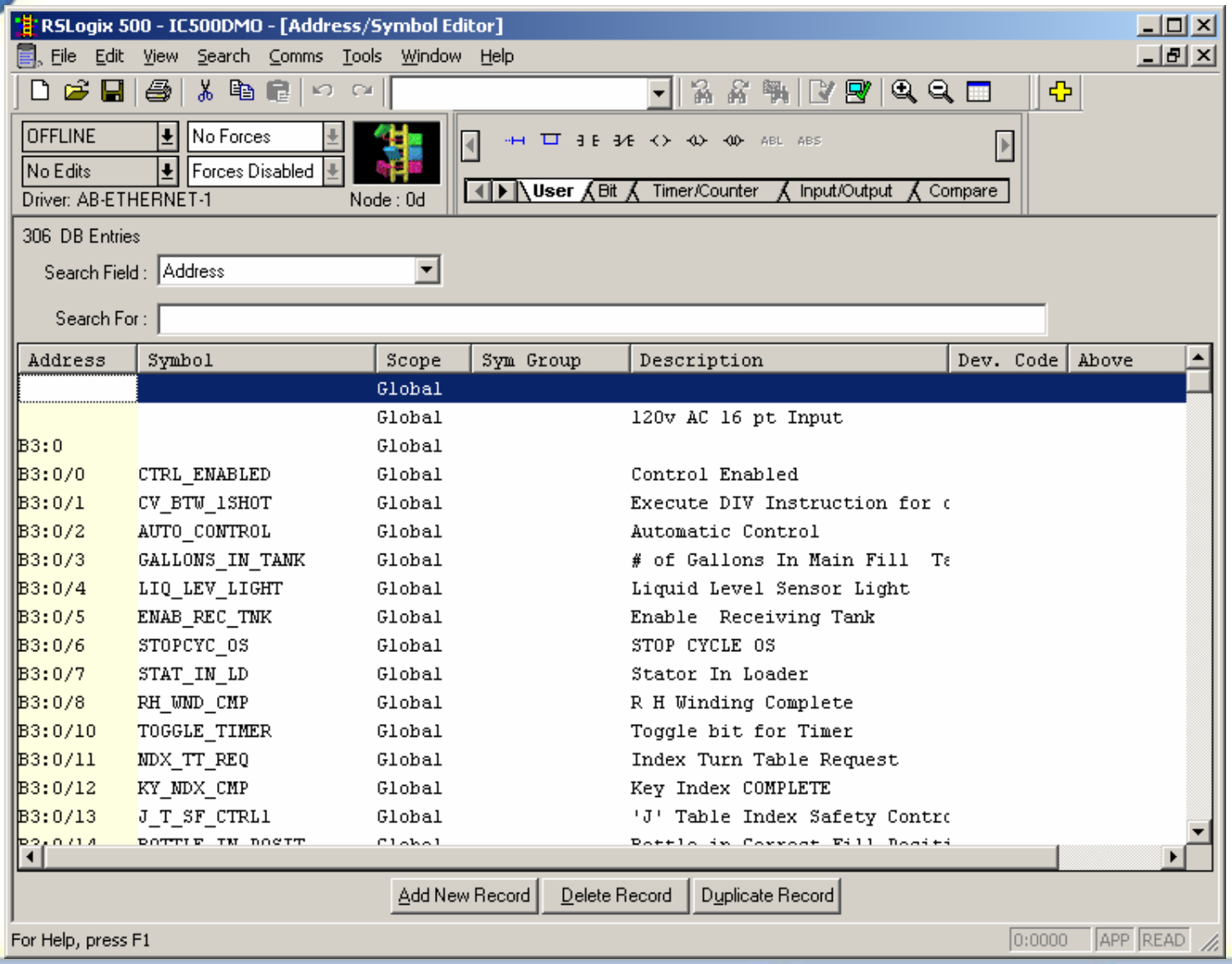
BIT DATA FORMATS

B3:3/14	Bit 14, element 3
B3:252/00	Bit 0, element 252
B3:9	Bit 62
B3/62	Bit 62
B3/4032	Bit 4032

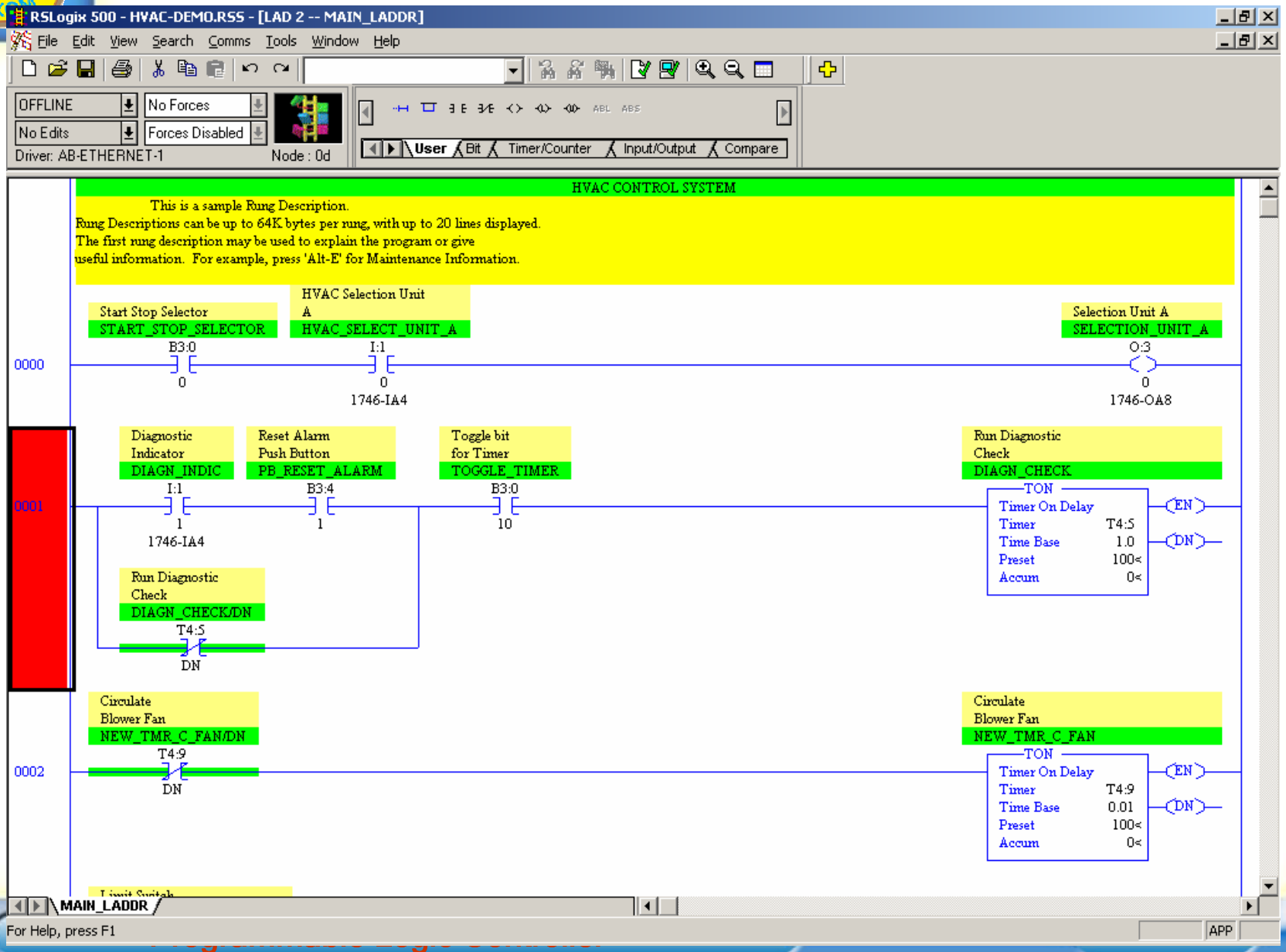
TIMER DATA FILE

T4:0/15 or T4:0/EN	Enable bit
T4:0/14 or T4:0/TT	Timer timing bit
T4:0/13 or T4:0/DN	Done bit
T4:0.1 or T4:0.PRE	Preset value of the timer
T4:0.2 or T4:0.ACC	Accumulated value of the timer
T4:0.1/0 or T4:0.PRE/0	Bit 0 of the preset value
T4:0.2/0 or T4:0.ACC/0	Bit 0 of the accumulated value

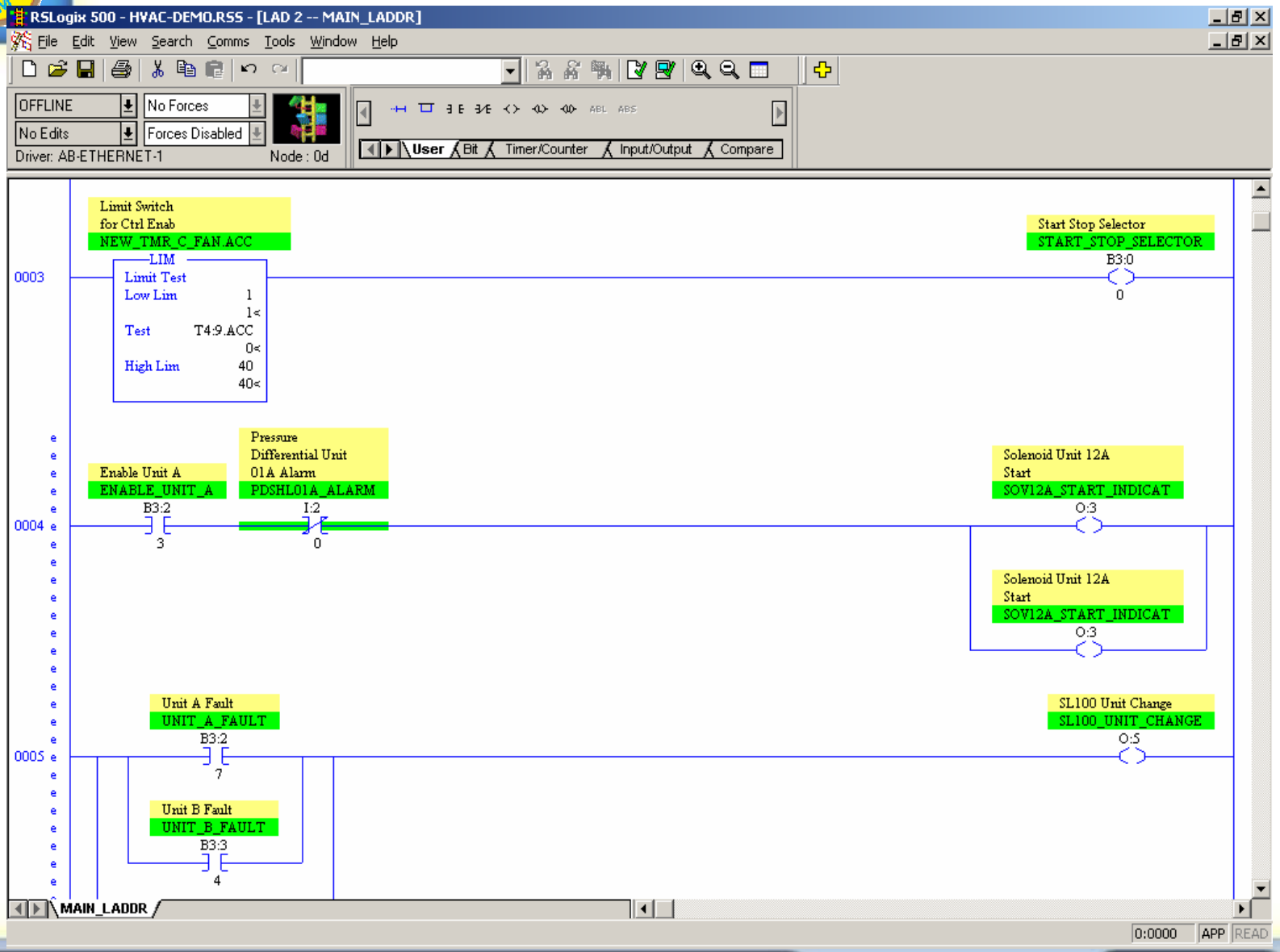
ADDRESS/SYMBOL EDITOR



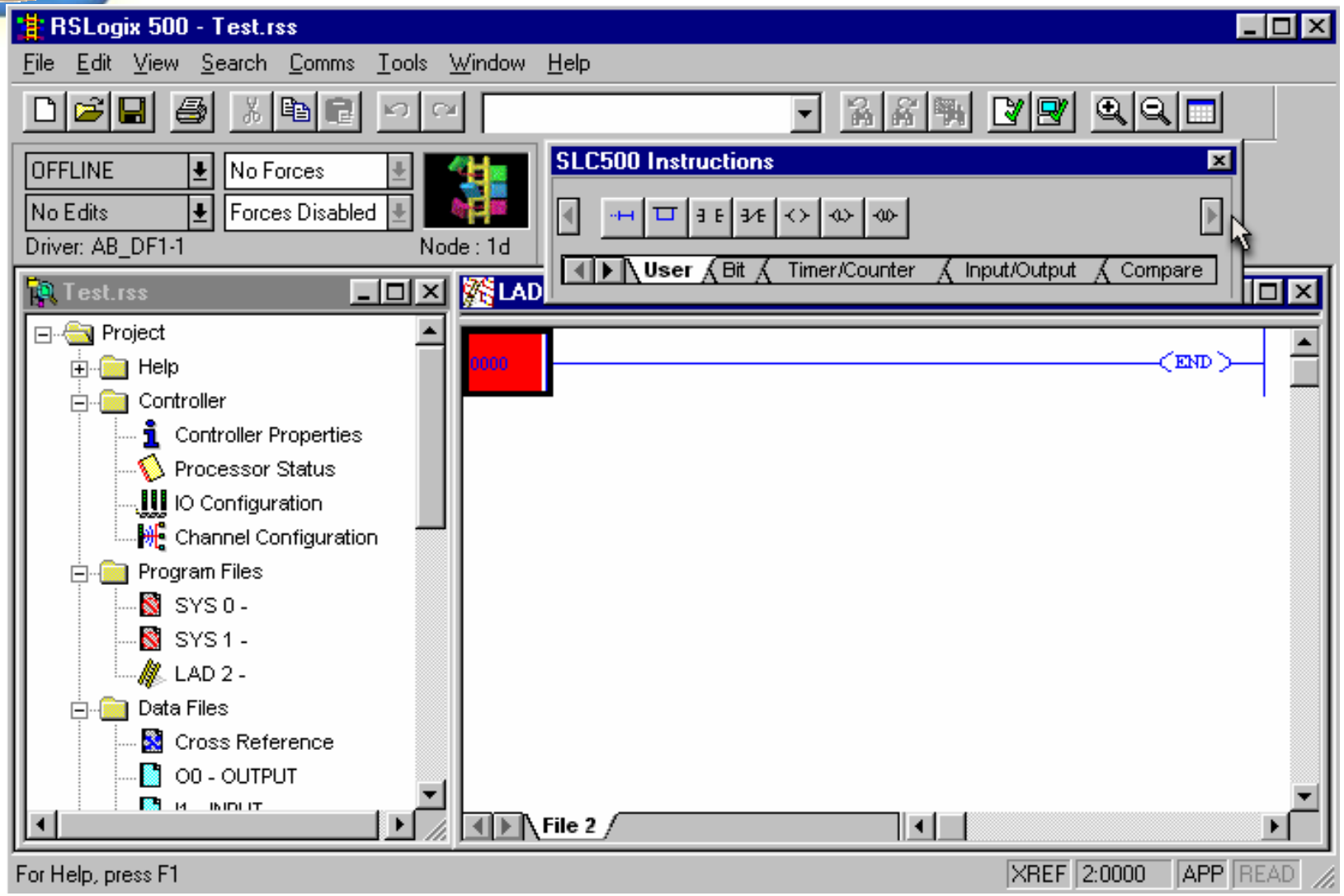
HVAC SAMPLE PROGRAM



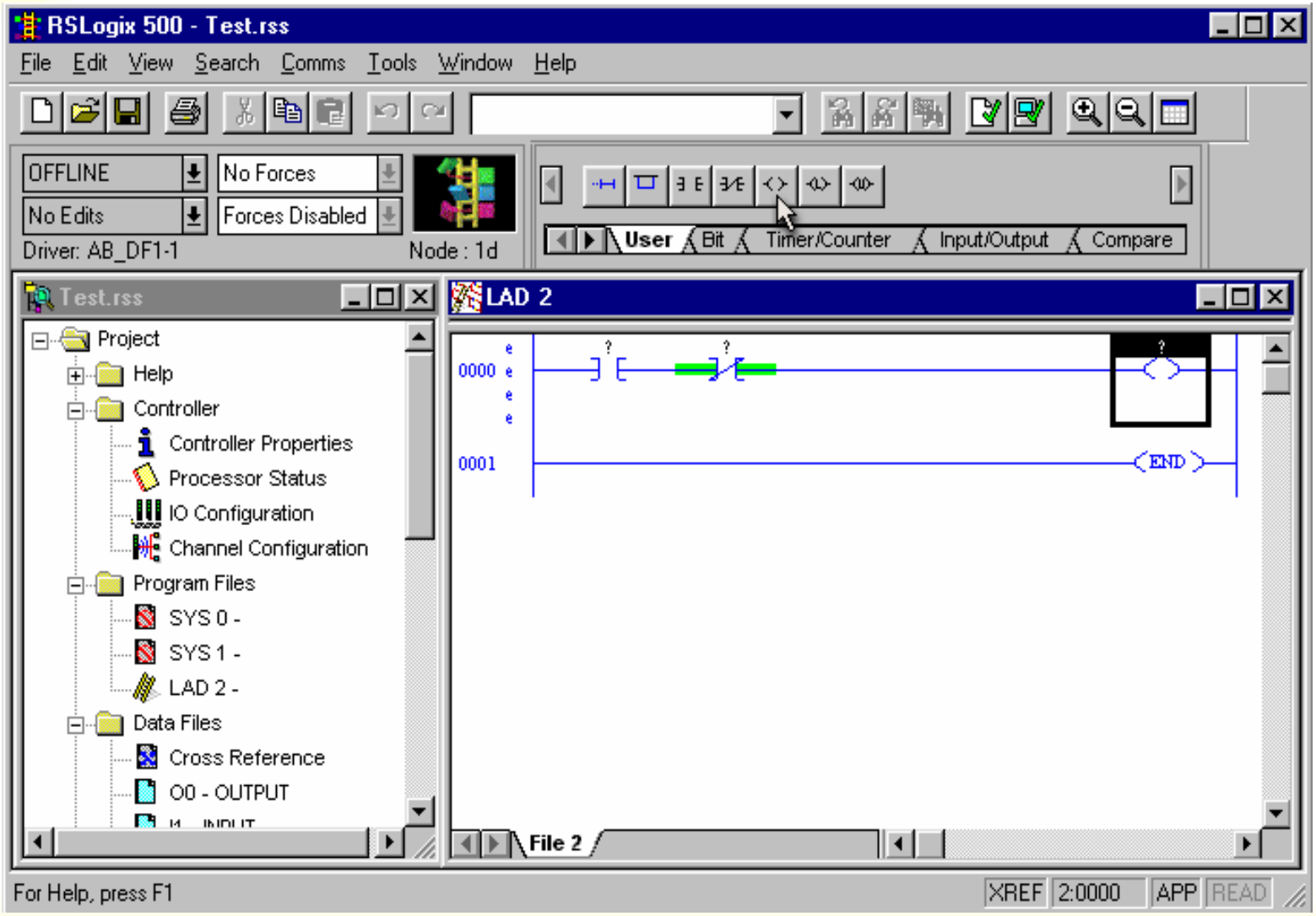
HVAC SAMPLE PROGRAM



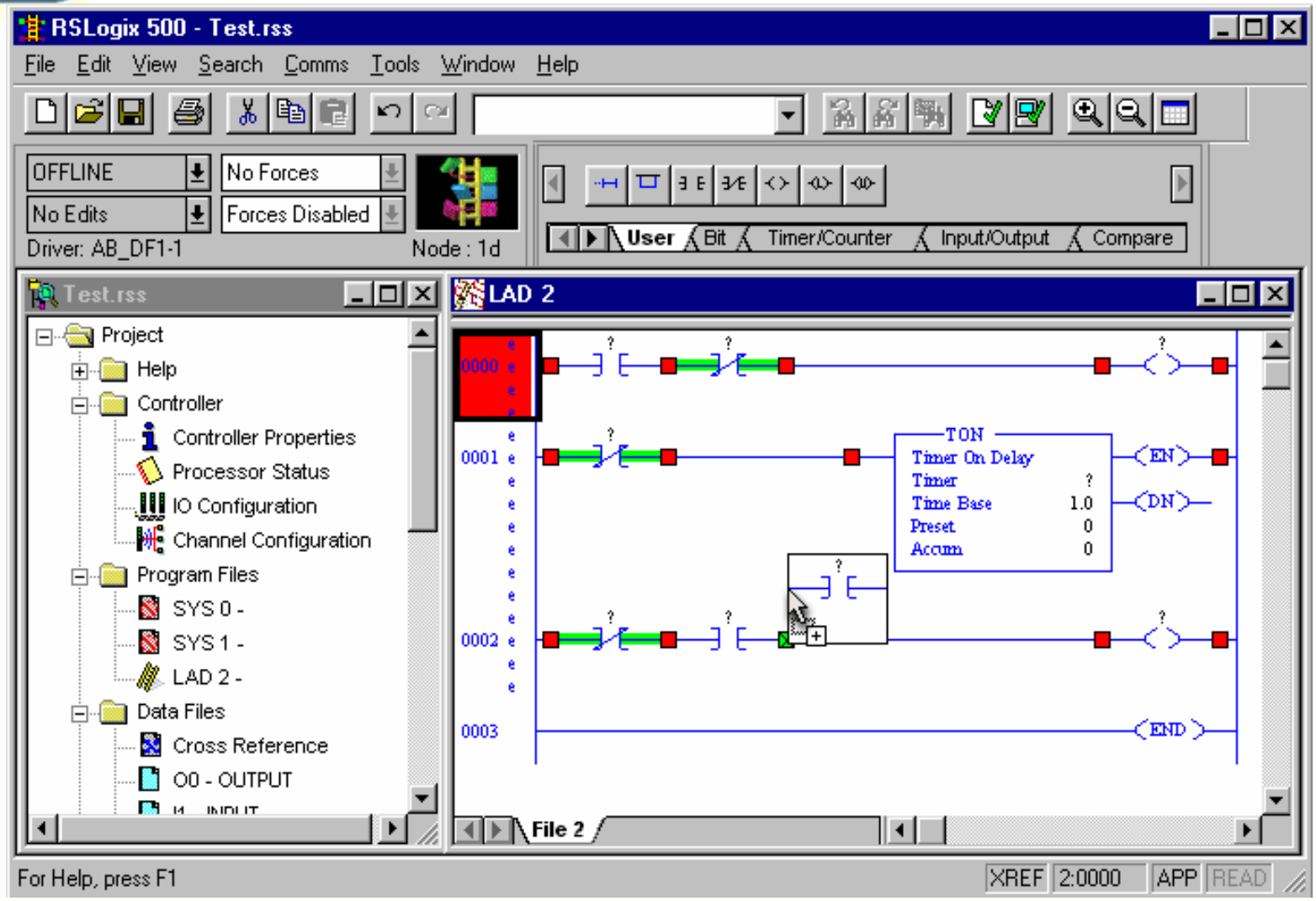
LADDER LOGIC EDITING



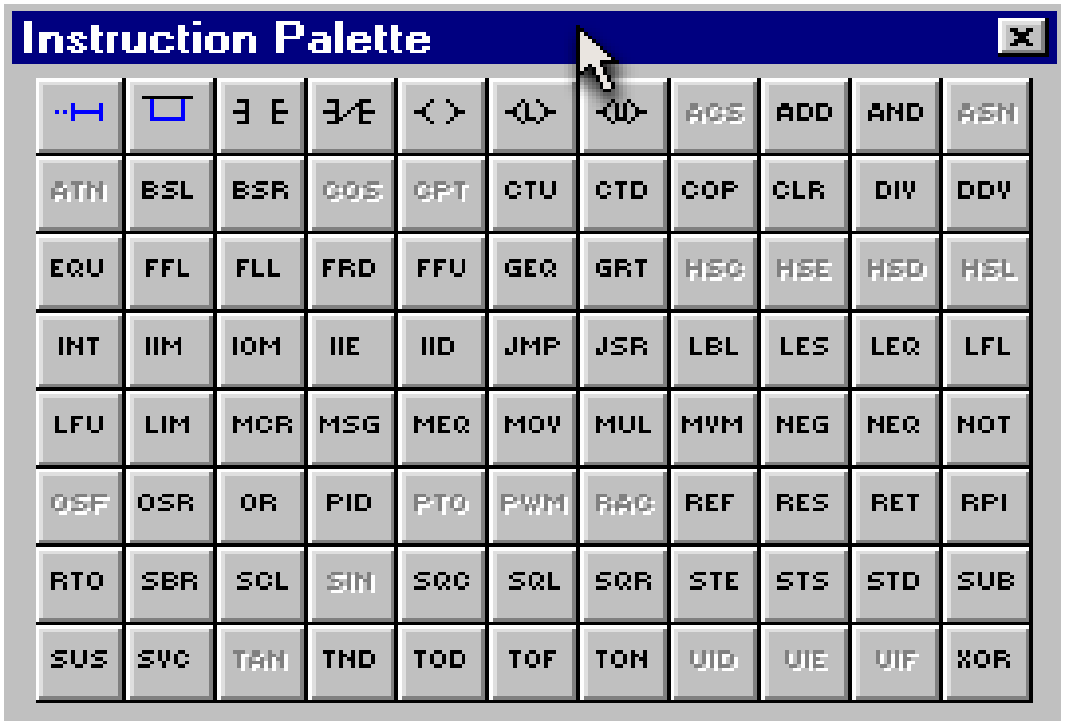
LADDER LOGIC EDITING



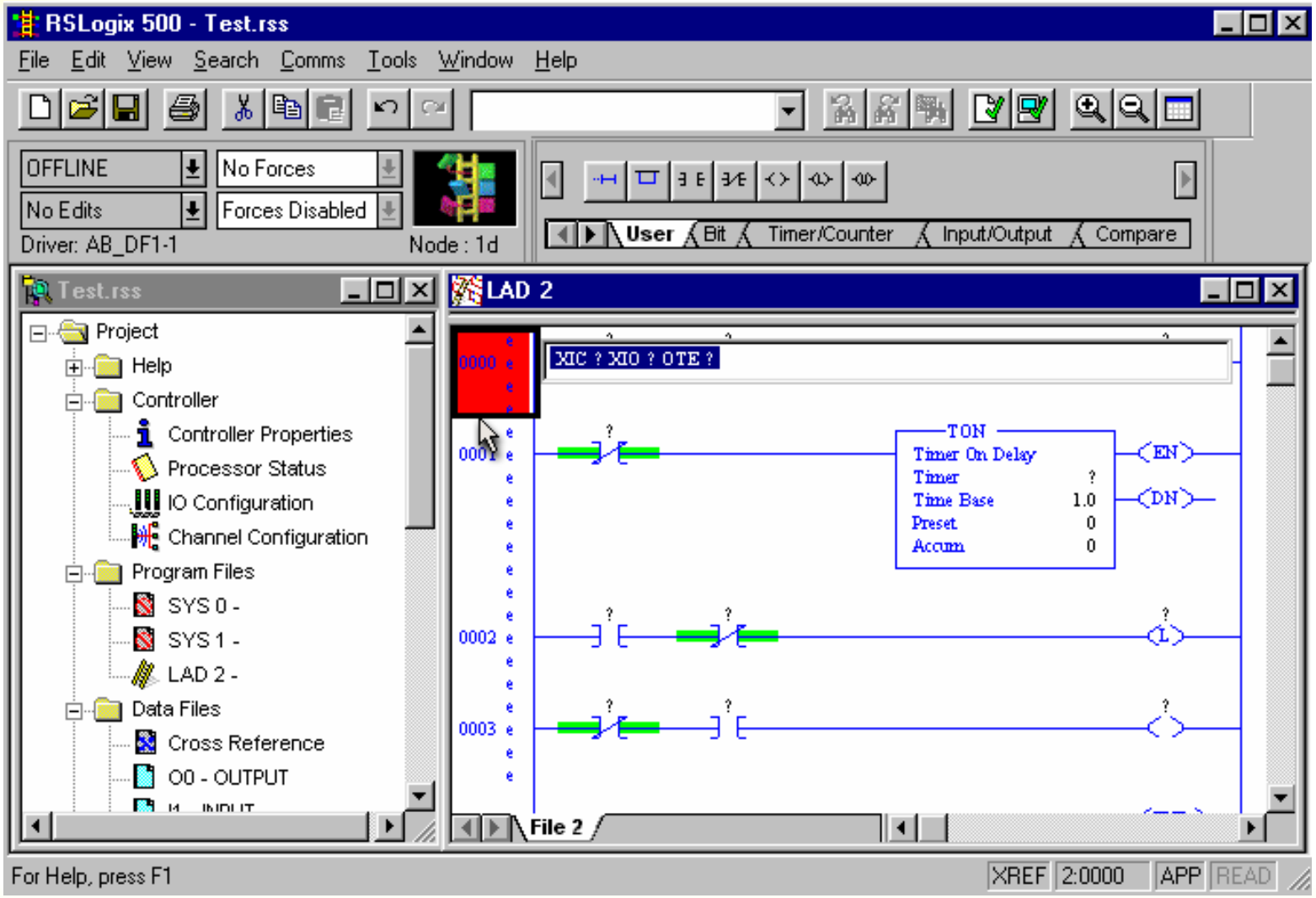
LADDER LOGIC EDITING



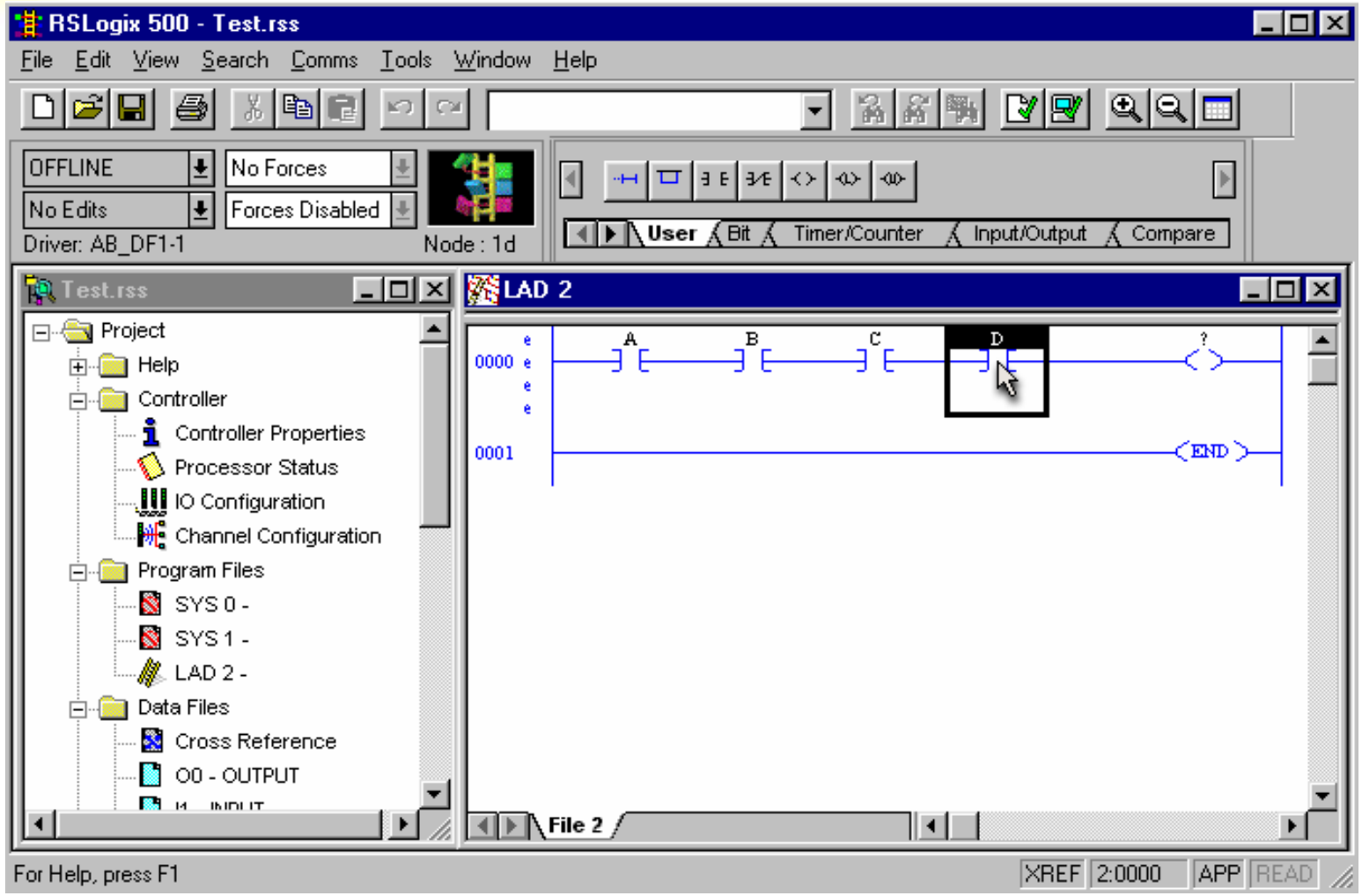
INSTRUCTION PALETTE



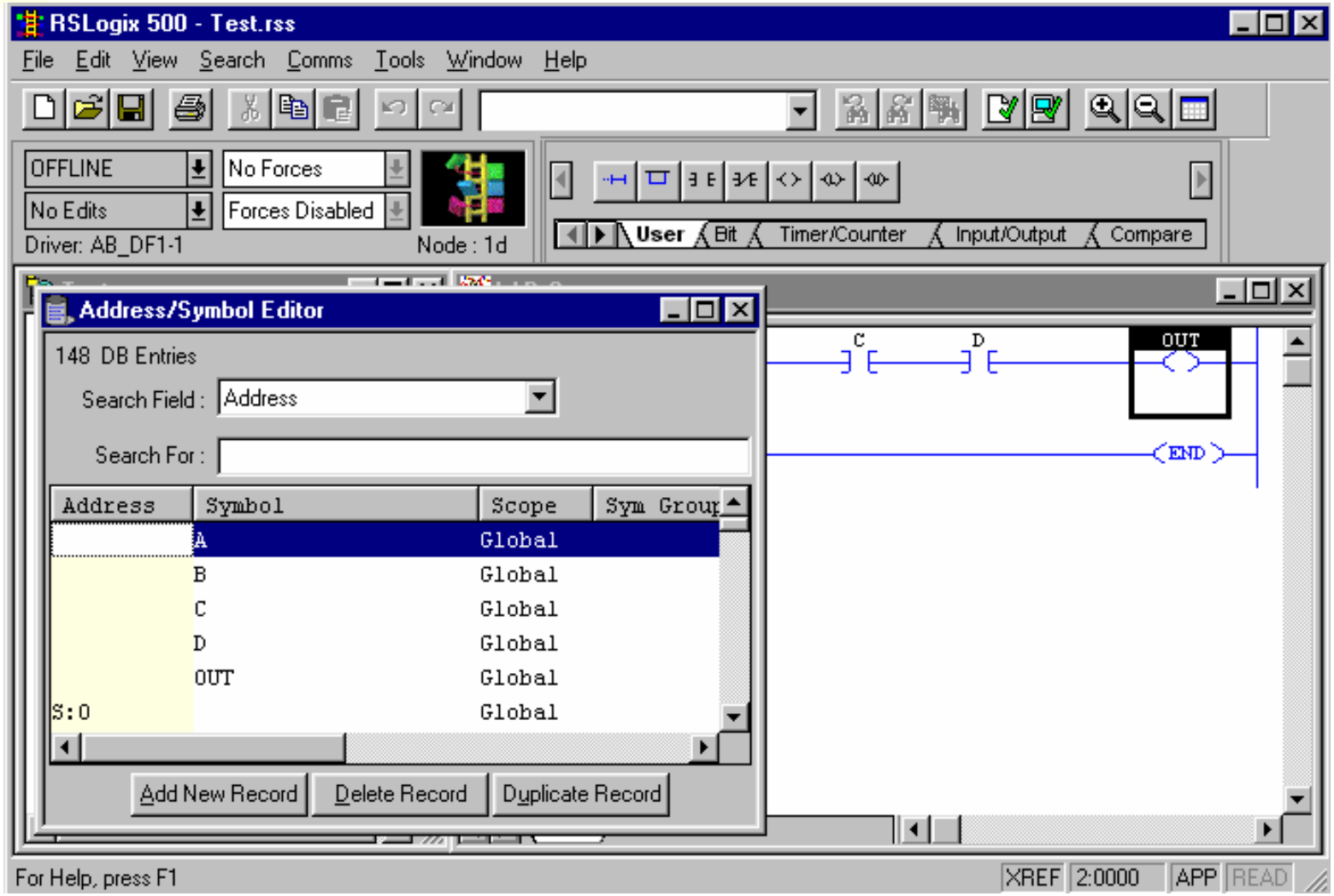



RSLogix 500 - Test.rss
File Edit View Search Comms Tools Window Help
OFFLINE
No Forces
No Edits
Forces Disabled
Driver: AB_DF1-1
Node : 1d
User Bit Timer/Counter Input/Output Compare
Test.rss
Project
Help
Controller
Controller Properties
Processor Status
IO Configuration
Channel Configuration
Program Files
SYS 0 -
SYS 1 -
LAD 2 -
Data Files
Cross Reference
O0 - OUTPUT
LAD 2
XIC ? XIO ? OTE ?
TON
Timer On Delay
Timer ?
Time Base 1.0
Preset 0
Accum 0
EN
DN
0000
0001
0002
0003
File 2
For Help, press F1
XREF 2:0000 APP READ

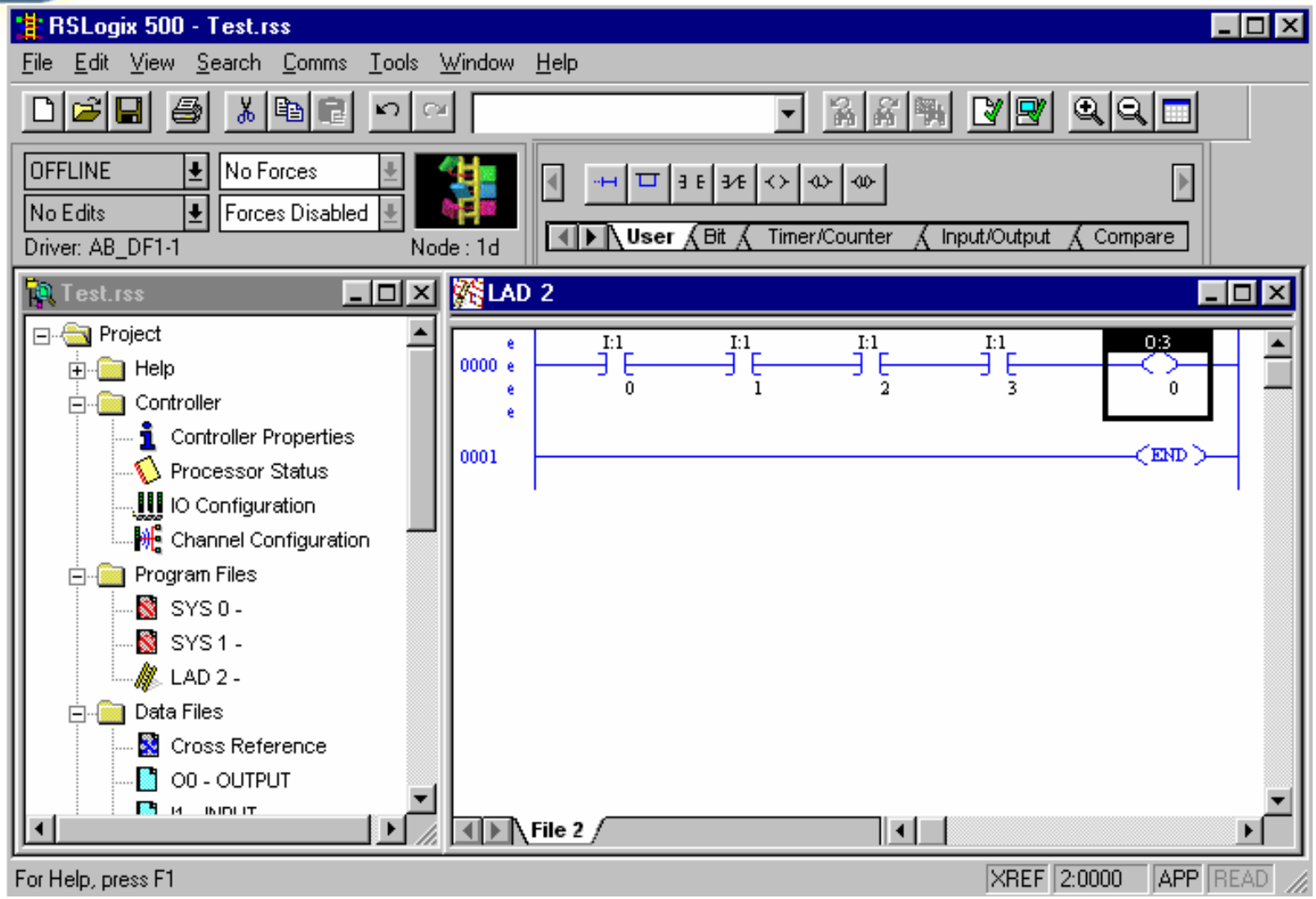
FREE FORM EDITOR



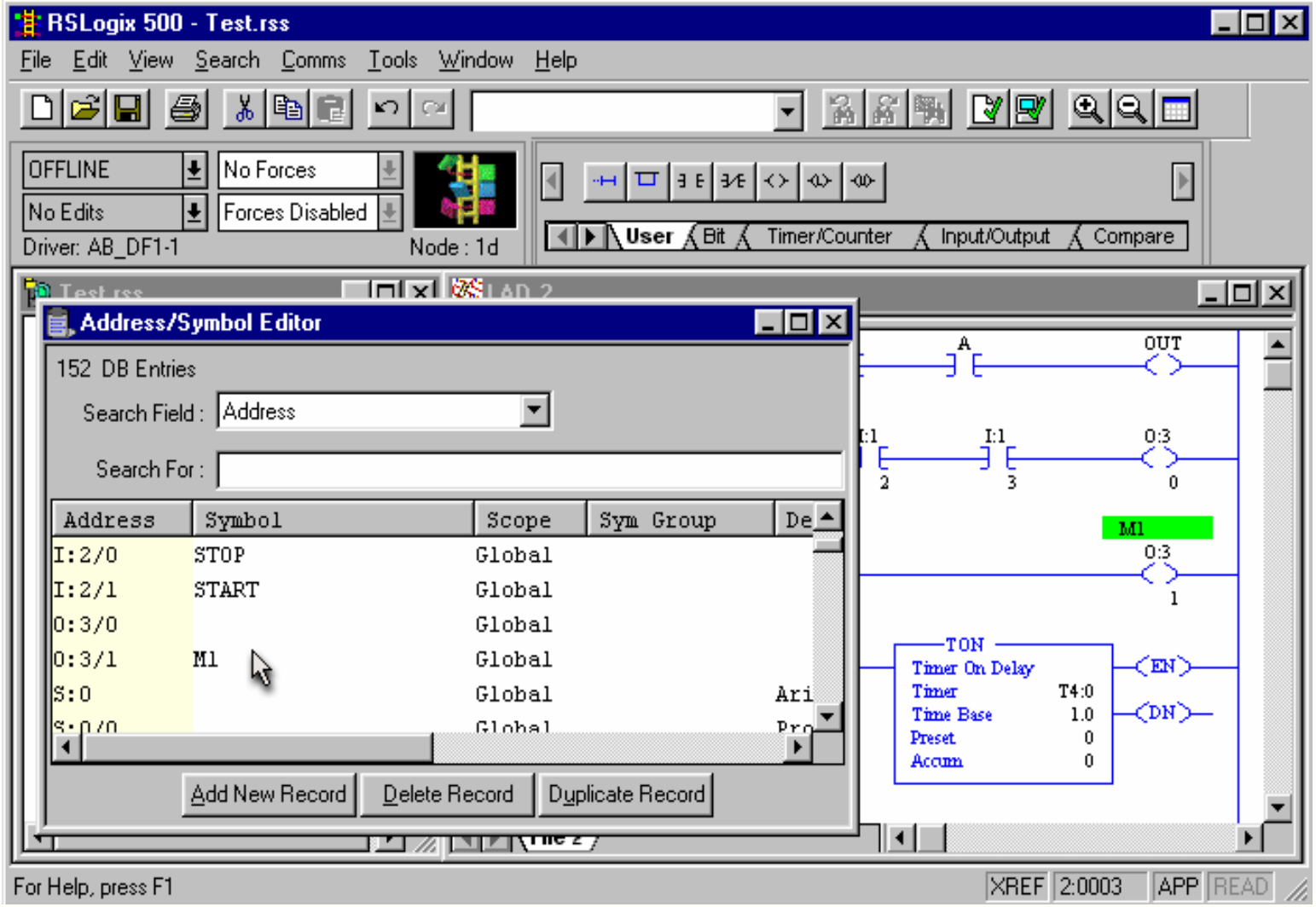
ADDRESS/SYMBOL EDITOR



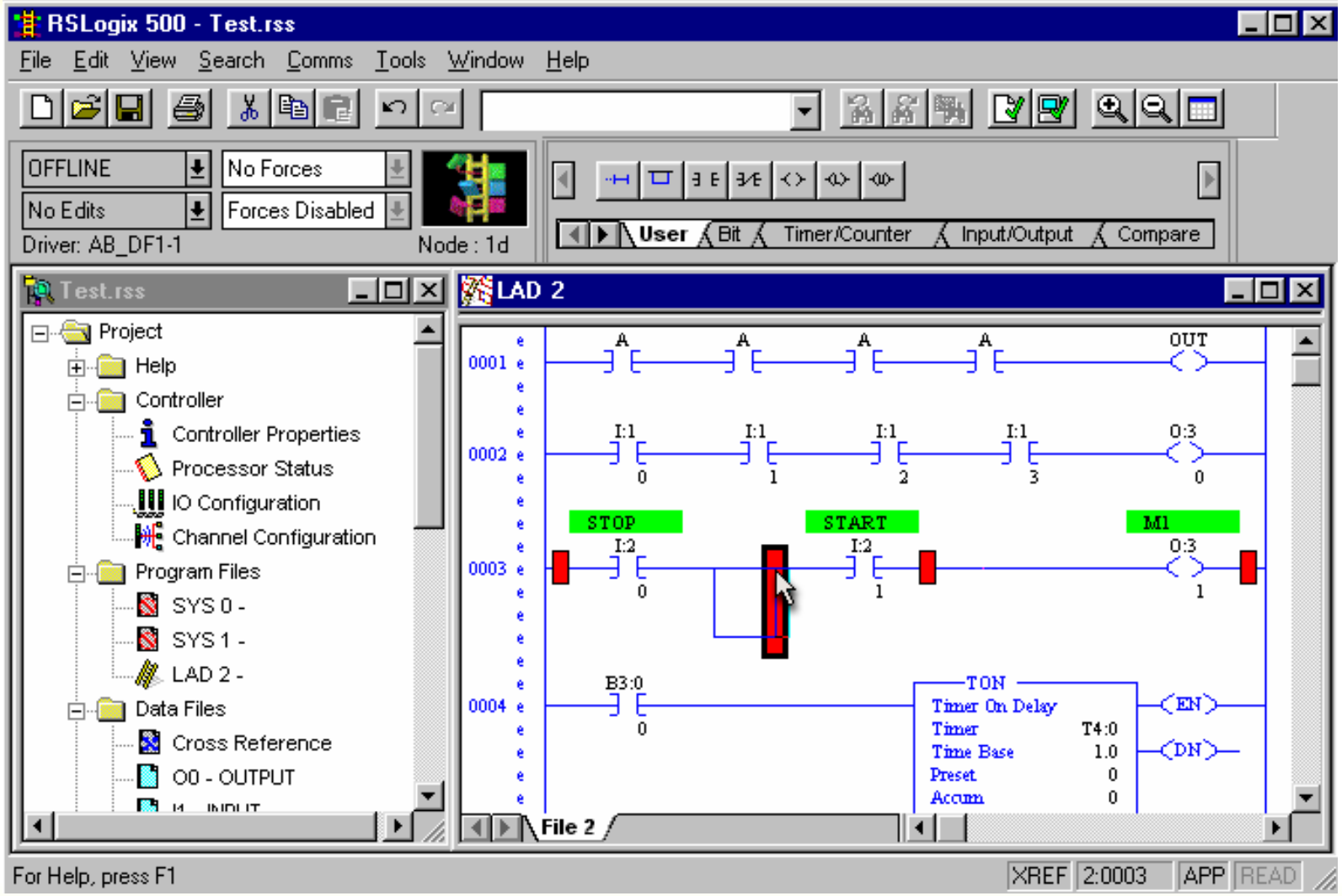
LADDER LOGIC EDITING



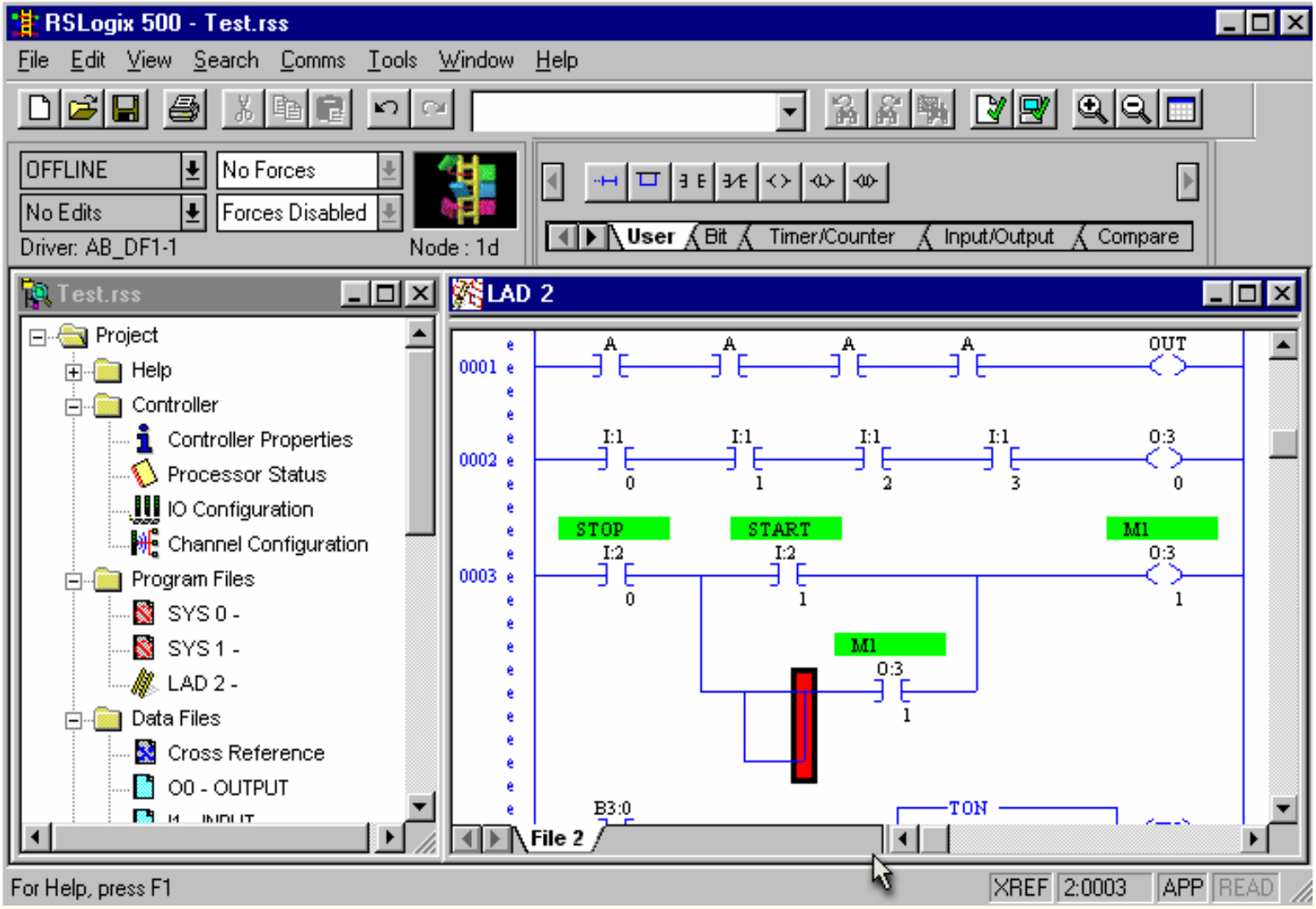
SYMBOLS


RSLogix 500 - Test.rss
File Edit View Search Comms Tools Window Help
OFFLINE
No Forces
No Edits
Forces Disabled
Driver: AB_DF1-1
Node : 1d
User Bit Timer/Counter Input/Output Compare
Address/Symbol Editor
152 DB Entries
Search Field : Address
Search For :
Address Symbol Scope Sym Group
I:2/0 STOP Global
I:2/1 START Global
0:3/0 Global
0:3/1 M1 Global
S:0 Global
Add New Record
Delete Record
Duplicate Record
A
OUT
I:1
2
I:1
3
0:3
0
M1
0:3
1
TON
Timer On Delay
Timer T4:0
Time Base 1.0
Preset 0
Accum 0
EN
DN
For Help, press F1
XREF 2:0003 APP READ

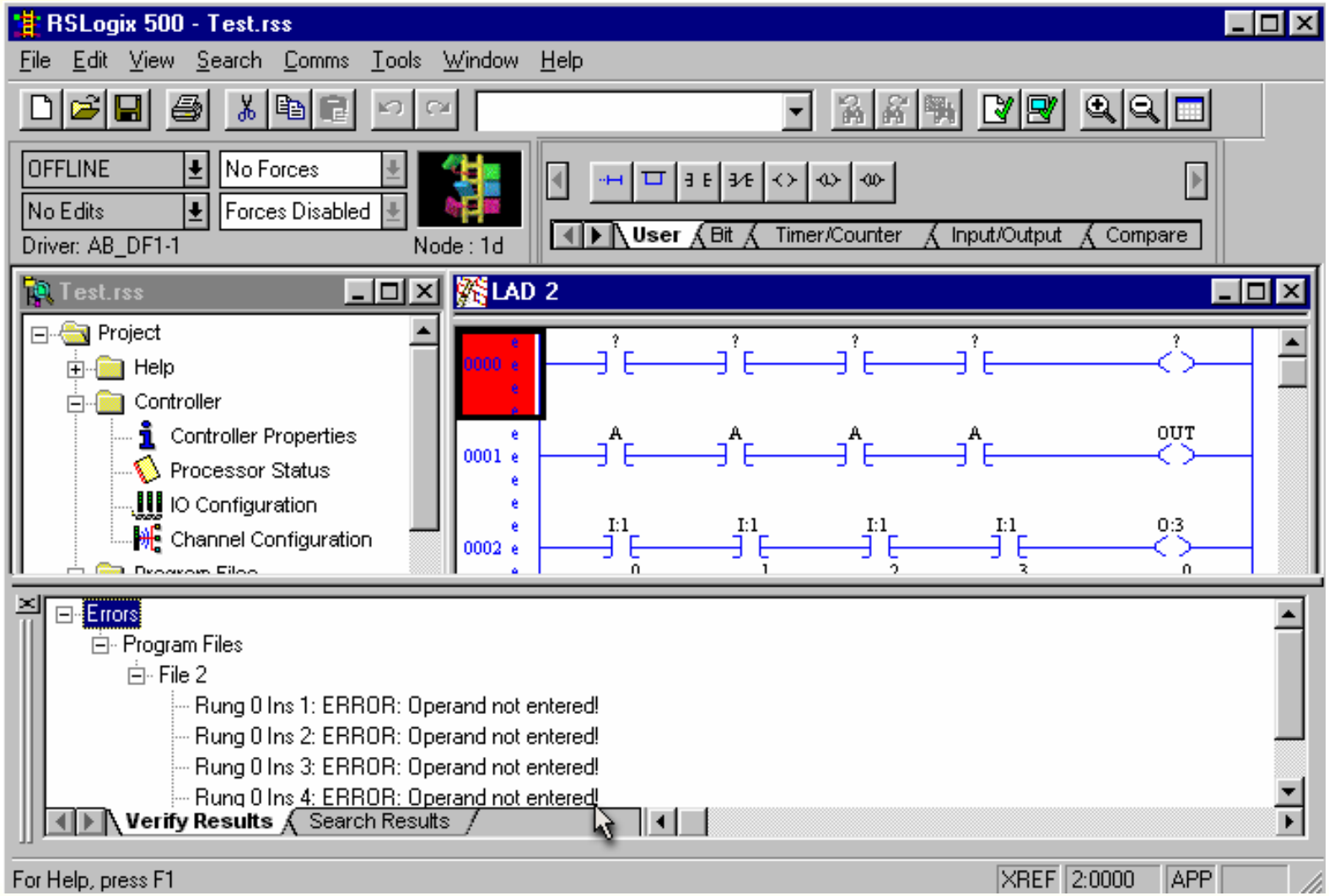
EXTENDED BRANCH



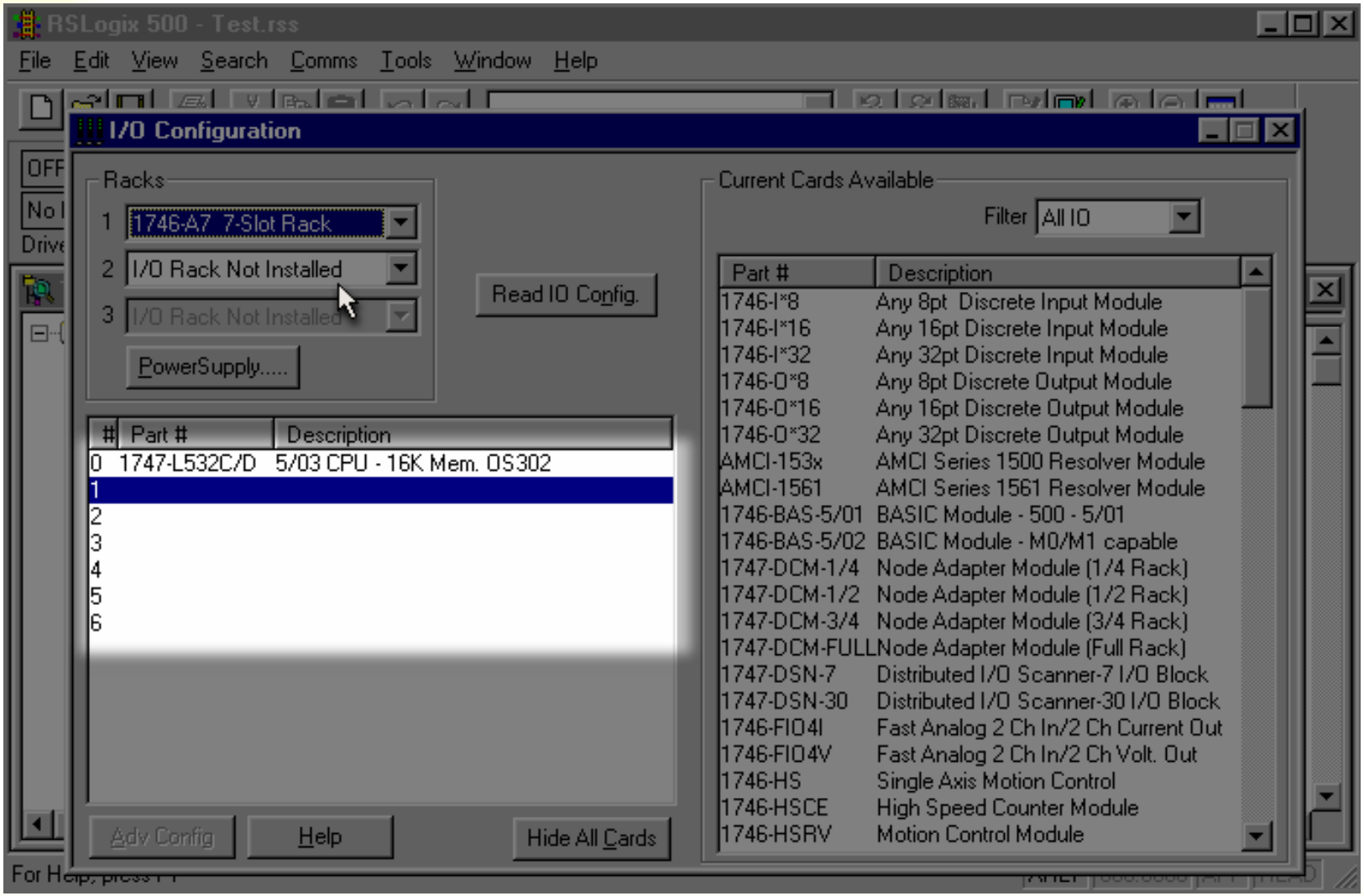
NESTED BRANCHES



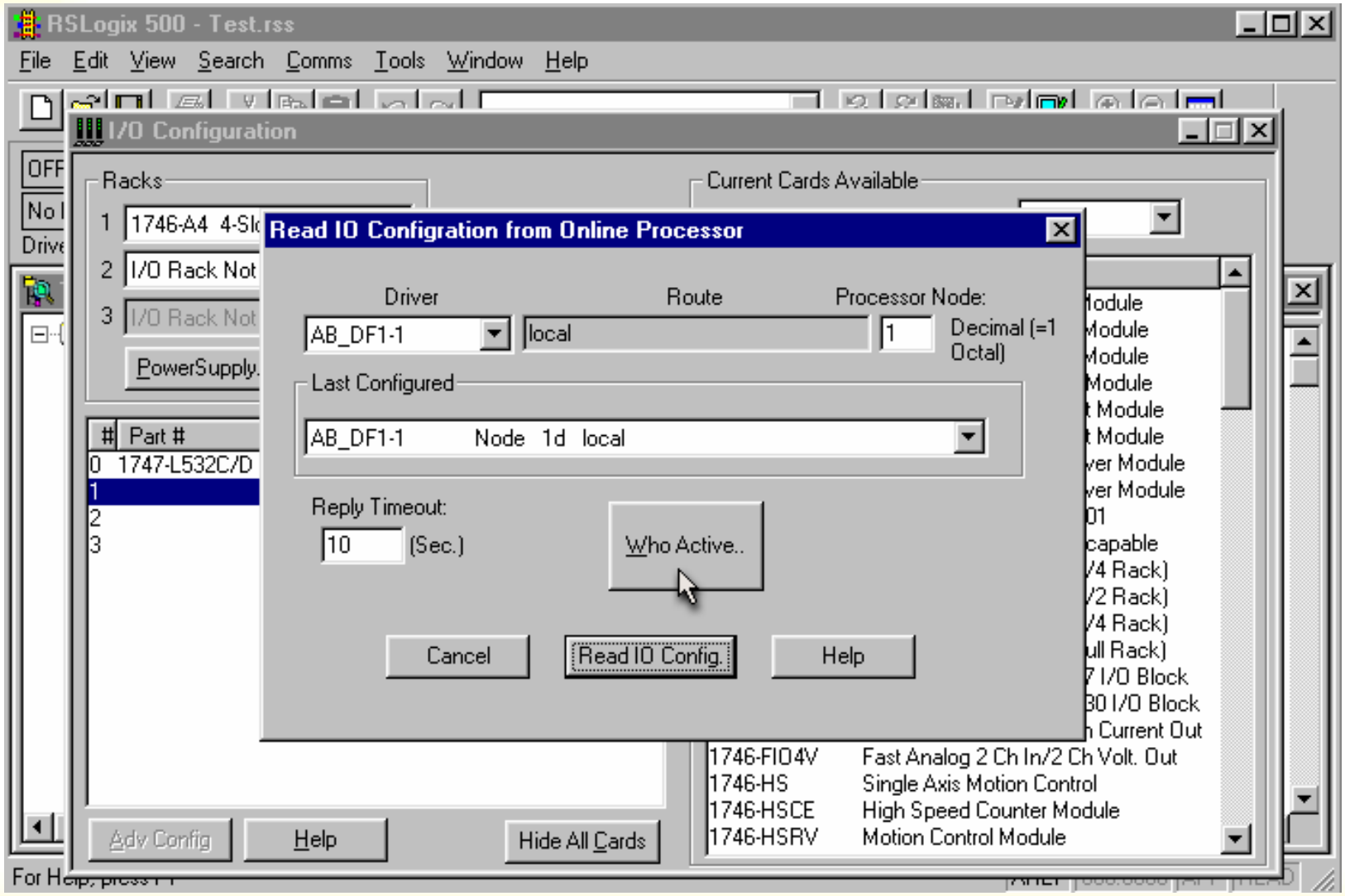
RUNG VERIFICATION



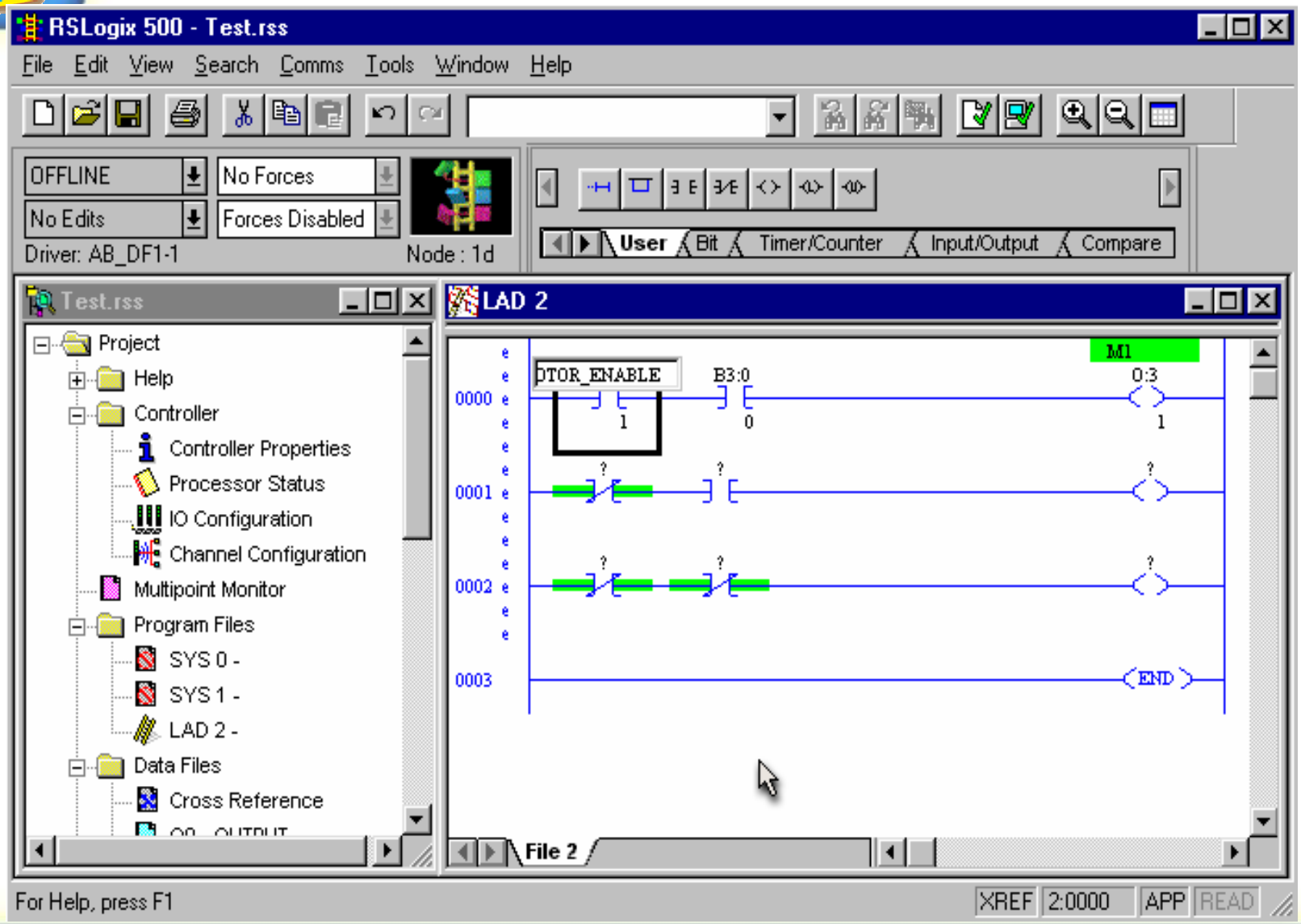
PROCESSOR AND RACK SETUP



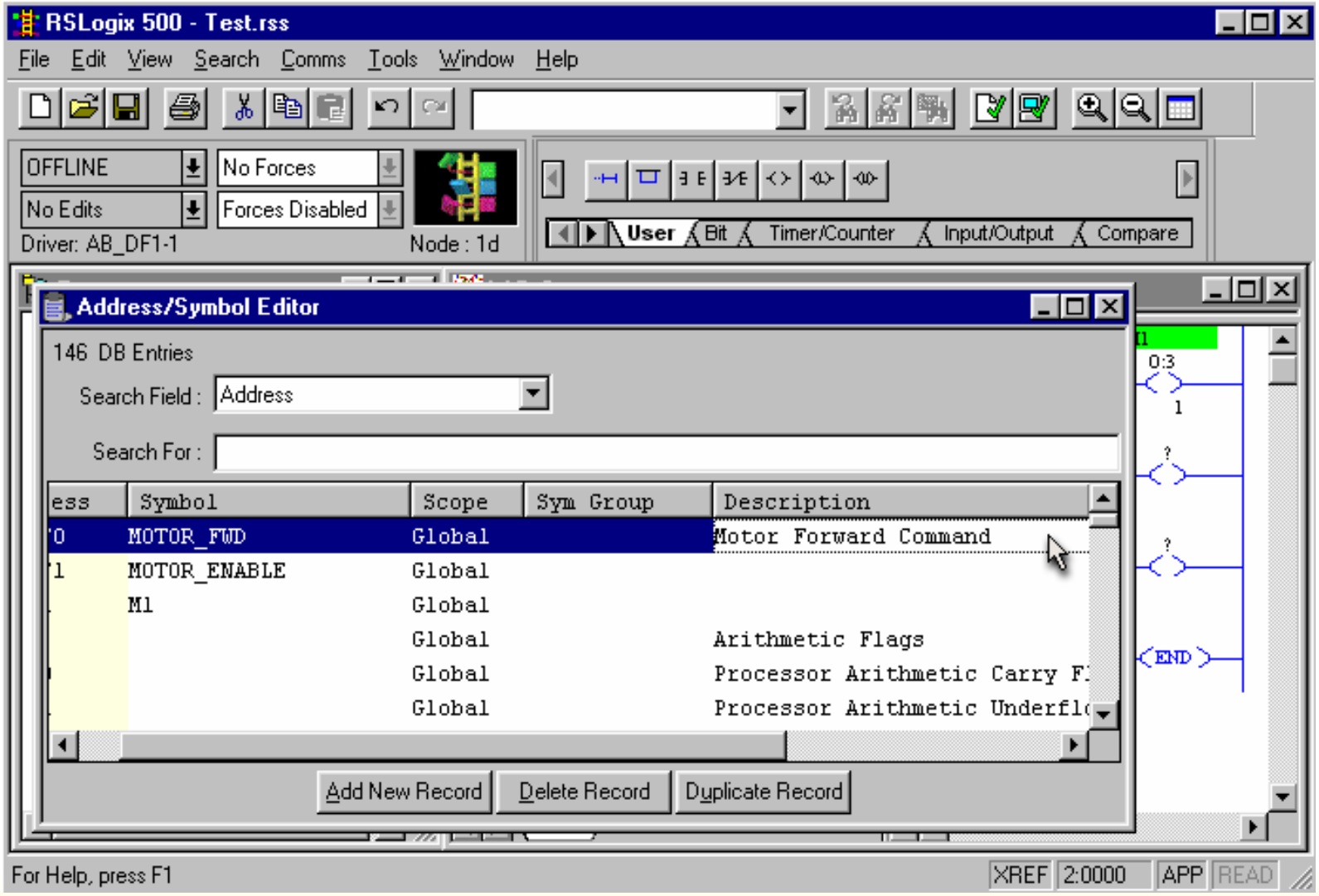
READ I/O CONFIGURATION



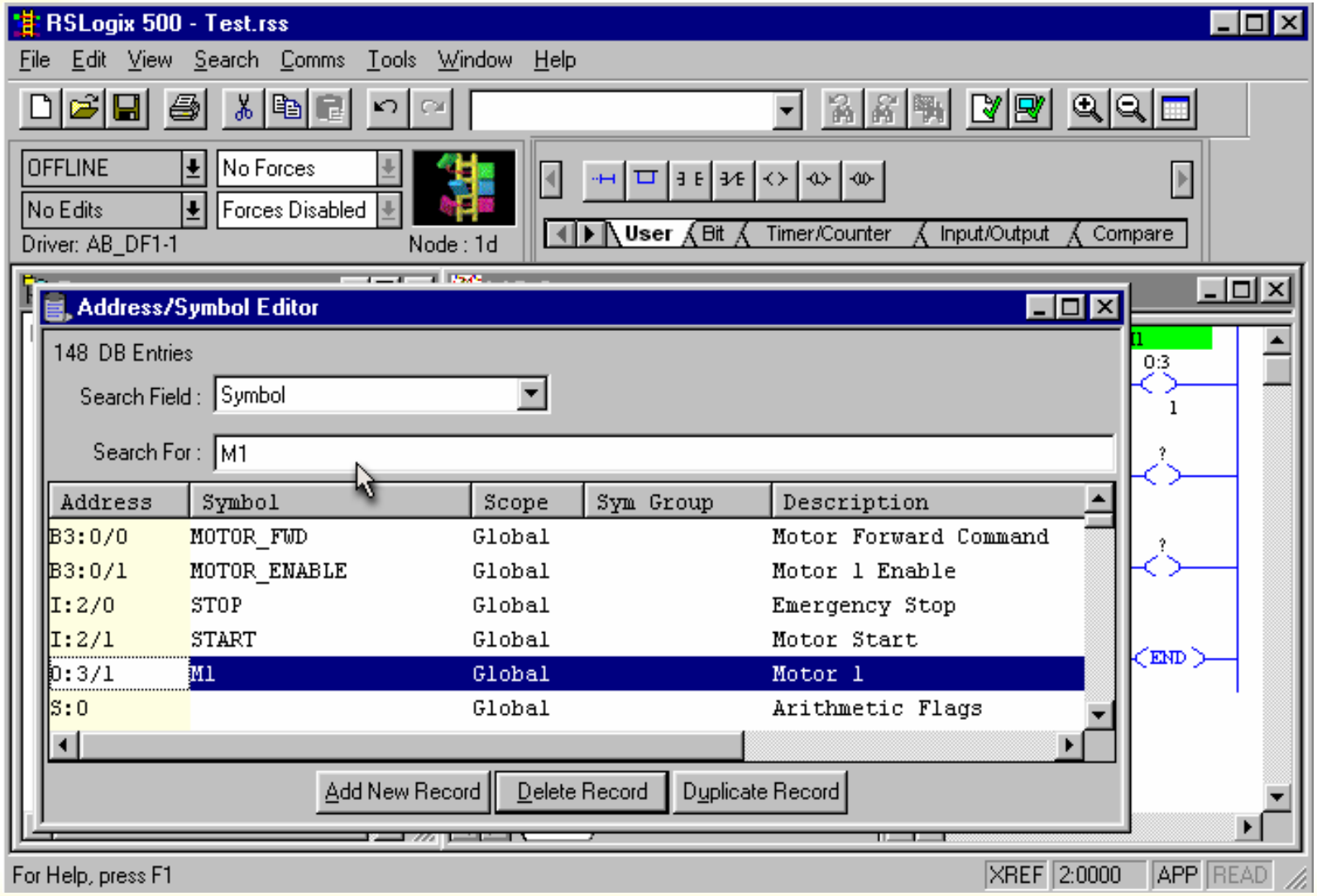
SYMBOL EDIT


RSLogix 500 - Test.rss
File Edit View Search Comms Tools Window Help
OFFLINE
No Forces
No Edits
Forces Disabled
Driver: AB_DF1-1
Node : 1d
User Bit Timer/Counter Input/Output Compare
Test.rss
Project
Help
Controller
Controller Properties
Processor Status
IO Configuration
Channel Configuration
Multipoint Monitor
Program Files
SYS 0 -
SYS 1 -
LAD 2 -
Data Files
Cross Reference
LAD 2
M1
DTOR_ENABLE
B3:0
0:3
0000
0001
0002
0003
END
File 2
For Help, press F1
XREF 2:0000 APP READ

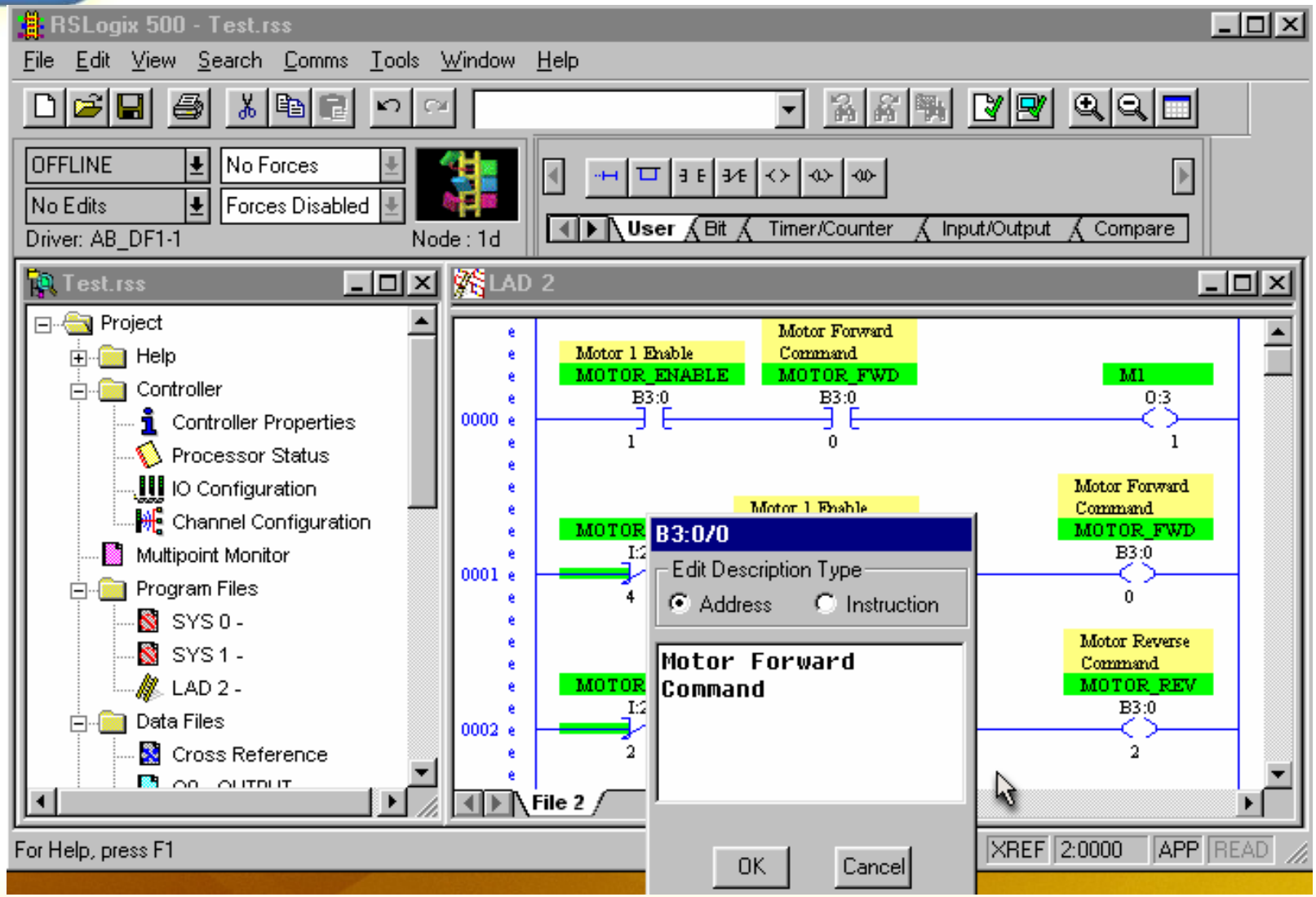
ADDRESS/SYMBOL EDITOR



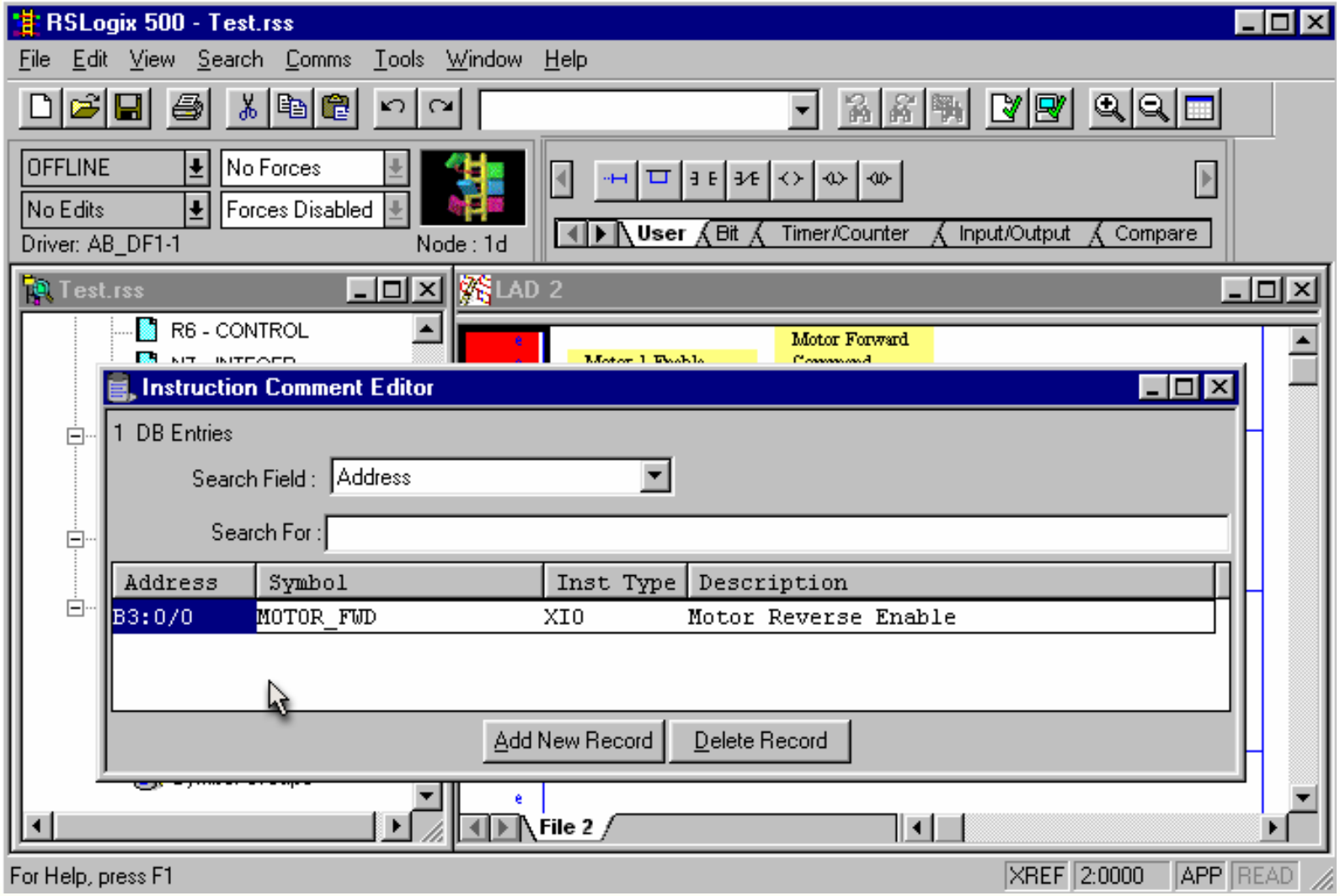
ADDRESS/SYMBOL SEARCHING



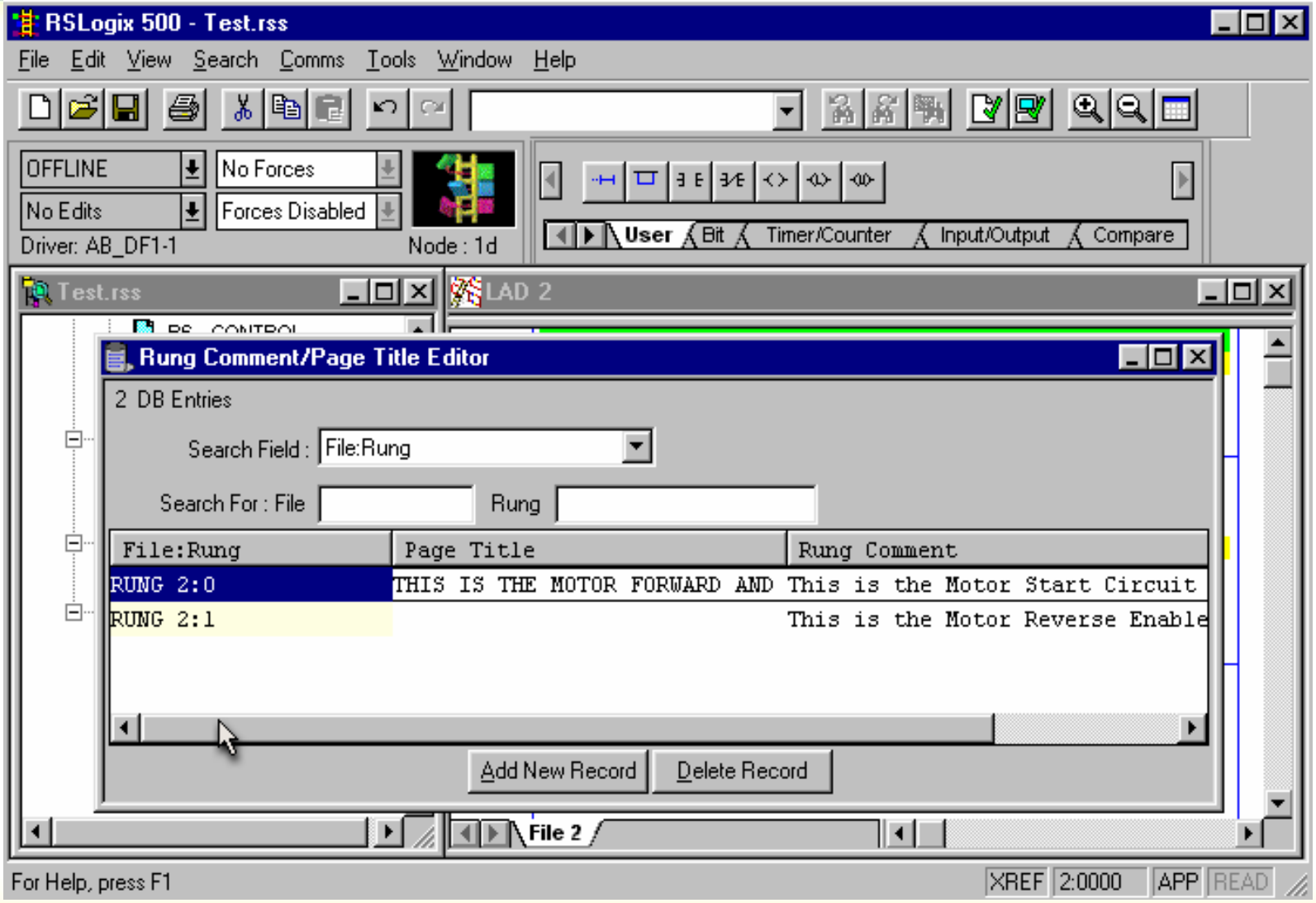
DESCRIPTION EDITING



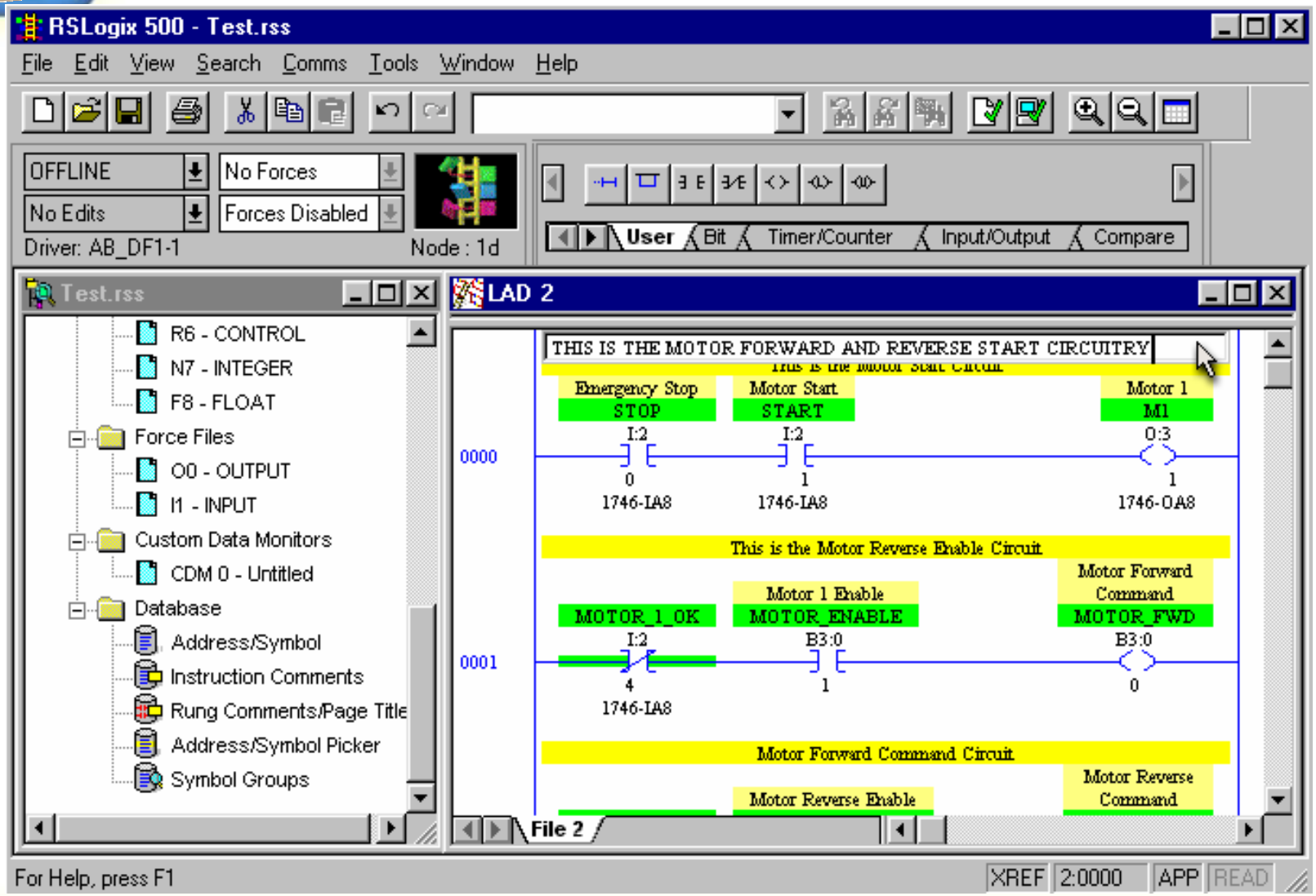
INSTRUCTION COMMENT



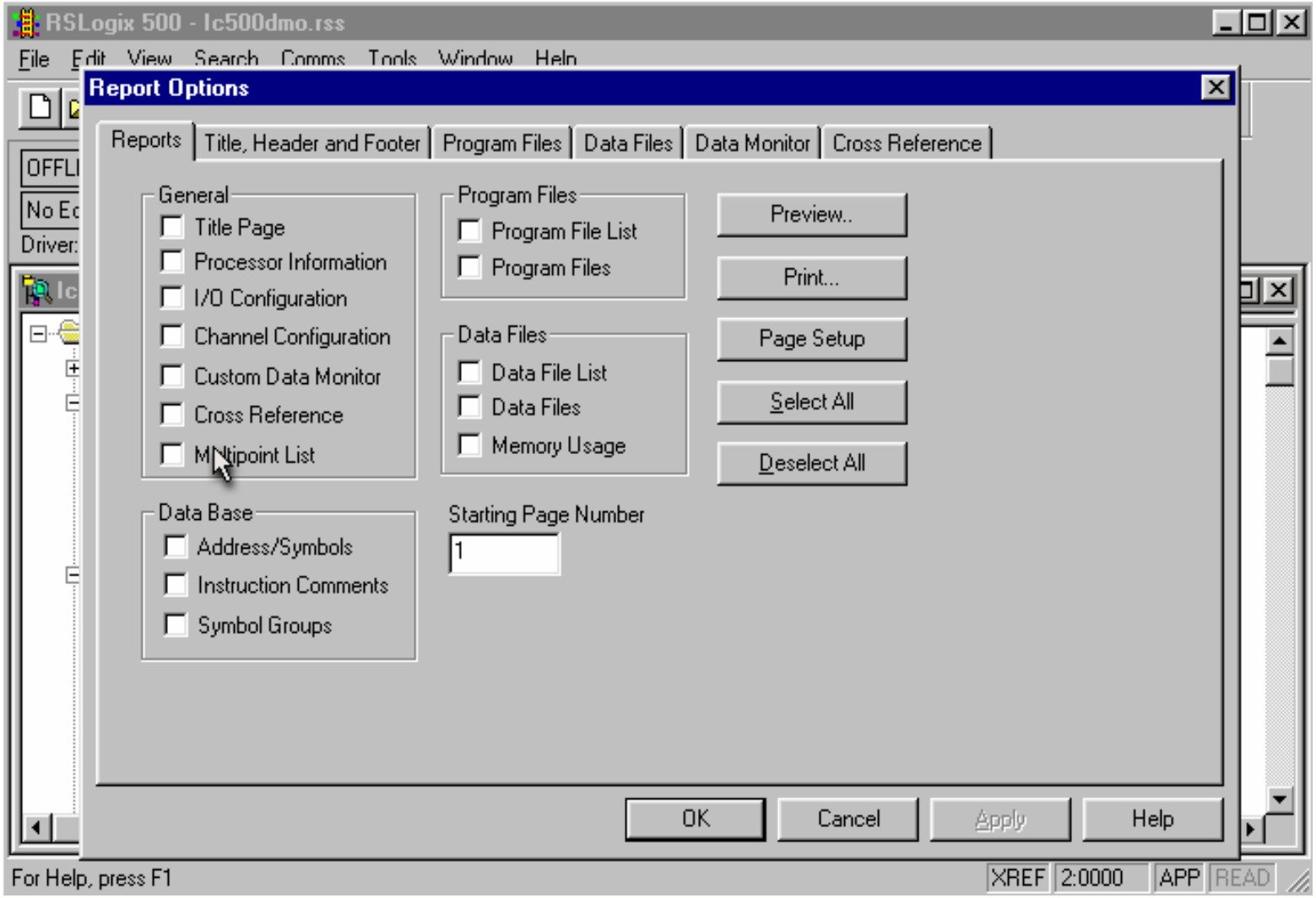
RUNG COMMENT



PAGE TITLE


RSLogix 500 - Test.rss
File Edit View Search Comms Tools Window Help
OFFLINE
No Forces
No Edits
Forces Disabled
Driver: AB_DF1-1
Node : 1d
User Bit Timer/Counter Input/Output Compare
Test.rss
R6 - CONTROL
N7 - INTEGER
F8 - FLOAT
Force Files
O0 - OUTPUT
I1 - INPUT
Custom Data Monitors
CDM 0 - Untitled
Database
Address/Symbol
Instruction Comments
Rung Comments/Page Title
Address/Symbol Picker
Symbol Groups
LAD 2
THIS IS THE MOTOR FORWARD AND REVERSE START CIRCUITRY
Emergency Stop
STOP
Motor Start
START
Motor 1
M1
I:2
I:2
O:3
0000
0
1
1
1746-IA8
1746-IA8
1746-OA8
This is the Motor Reverse Enable Circuit
Motor 1 Enable
MOTOR_ENABLE
Motor Forward Command
MOTOR_FWD
MOTOR_1_OK
I:2
B3:0
B3:0
0001
4
1
0
1746-IA8
Motor Forward Command Circuit
Motor Reverse Command
Motor Reverse Enable
File 2
For Help, press F1
XREF 2:0000 APP READ

REPORT OPTION



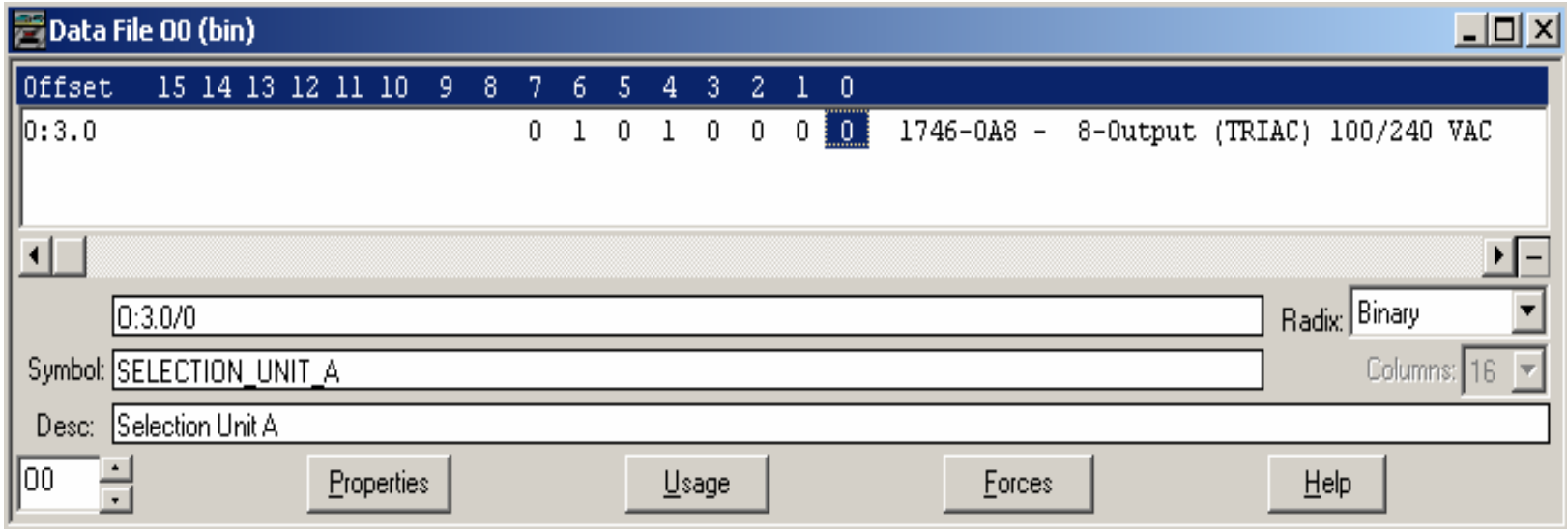
CROSS REFERENCE REPORT

Cross Reference Report - Sorted by Address

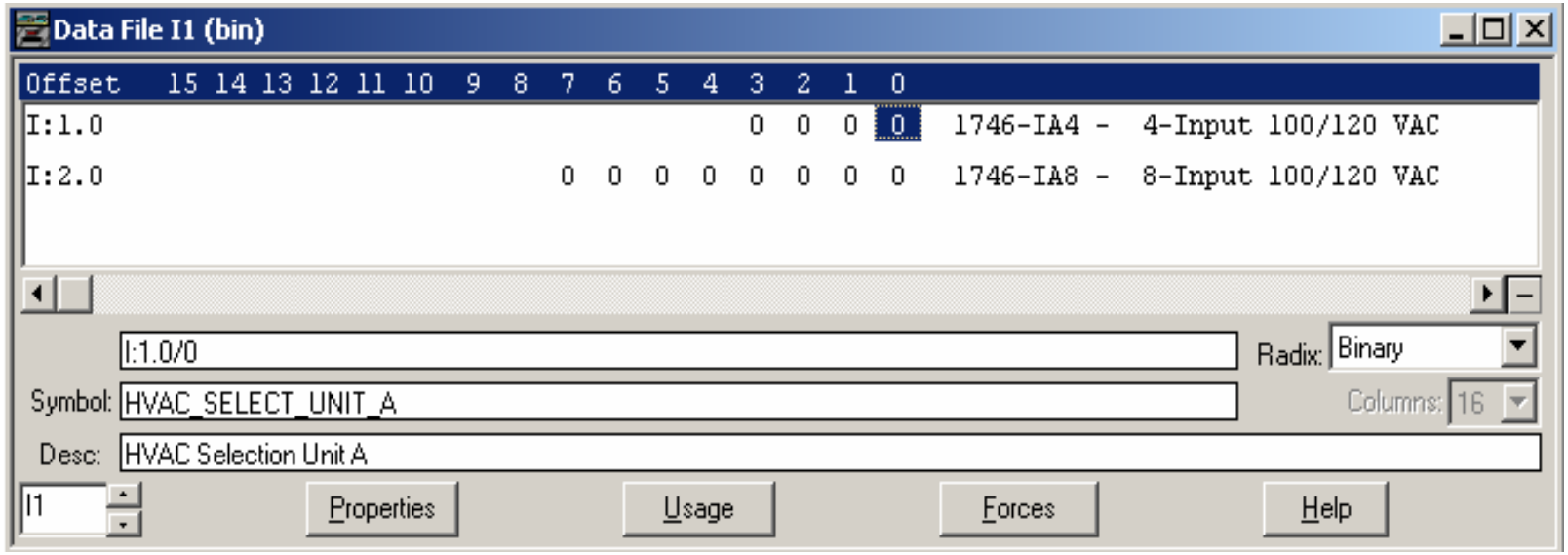
```
0:3/0        - {SELECTION UNIT A} Selection Unit A
                 OTE - File #2 MAIN_LADDR - 0
                 XIC - File #2 MAIN_LADDR - 105
0:3/1        - {CORE_TRANS_GUID} Core Transfer Guide Closed
                 XIO - File #2 MAIN_LADDR - 29, 35
0:3/2        - {TABLE_A_DOWN} Table for Step Conveyor in Down Position
                 OTE - File #2 MAIN_LADDR - 31
                 XIC - File #2 MAIN_LADDR - 33, 87
                 XIO - File #2 MAIN_LADDR - 22
0:3/3        - {FILL_STAT_READY} Fill Station A in Ready State
                 OTE - File #2 MAIN_LADDR - 36
                 XIC - File #2 MAIN_LADDR - 33
                 XIO - File #2 MAIN_LADDR - 29
0:3/4        - {HEAT_OFF} Heating Unit on Bottle Fill SHUT OFF
                 OTE - File #2 MAIN_LADDR - 14
                 XIC - File #2 MAIN_LADDR - 22, 50
                 XIO - File #2 MAIN_LADDR - 23
0:3/5        - {INJECT_RETR} Inject Retract High Speed
                 OTE - File #2 MAIN_LADDR - 21
                 XIO - File #2 MAIN_LADDR - 29
0:3/6        - {ALARM_DRY_MATL} Dry Material Low Alarm
                 OTE - File #2 MAIN_LADDR - 22
                 XIC - File #2 MAIN_LADDR - 15
0:3/7        - {INJECT_MOTOR} Enable Injection Motor for Pour
                 OTE - File #2 MAIN_LADDR - 50
                 XIC - File #2 MAIN_LADDR - 53
                 XIO - File #2 MAIN_LADDR - 29
I:1/0        - {HVAC_SELECT_UNIT_A} HVAC Selection Unit A
                 XIC - File #2 MAIN_LADDR - 0
I:1/1        - {DIAGN_INDIC} Diagnostic Indicator
                 XIC - File #2 MAIN_LADDR - 1, 24
I:1/2        - {JOG_PB} Jog Push Button
                 XIC - File #2 MAIN_LADDR - 20, 40
                       File #5 SUB_RT_3 - 0
```

Sort By Symbol | Refresh | Help

DISCRETE OUTPUT ADDRESSING



DISCRETE INPUT ADDRESSING



BINARY DATA ADDRESSING

Data File B3 (bin)

Offset	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	(Symbol) Description
B3:0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
B3:1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	(MOVE_B30_TO_B31)
B3:2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
B3:3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
B3:4	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	
B3:5	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	
B3:6	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	
B3:7	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
B3:8	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
B3:9	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
B3:10	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	

B3:0/0 Radix: Binary

Symbol: START_STOP_SELECTOR Columns: 16

Desc: Start Stop Selector

B3 Properties Usage Help

TIMER DATA STRUCTURE

Data File T4

Offset	EN	TT	DN	BASE	PRE	ACC	(Symbol) Description
T4:10	0	0	0	.01 sec	3000	0	
T4:11	0	0	0	1.0 sec	60	0	
T4:12	0	0	0	.01 sec	1000	0	
T4:13	0	0	0	1.0 sec	80	0	
T4:14	1	1	0	1.0 sec	500	7	(J_LINE_TMR)
T4:15	0	0	0	.01 sec	1	0	
T4:16	1	1	0	1.0 sec	500	0	(K_LINE_TMR)
T4:17	0	0	0	.01 sec	400	0	(TRANSFER_TMR)
T4:18	0	0	0	.01 sec	500	0	(INJECT_TMR)
T4:19	1	1	0	.01 sec	3000	746	(FREE_TMR_1)
T4:20	1	1	0	1.0 sec	50	0	(TMR_SOLEN_VAL) ControlsTanks

T4:0/EN Radix:

Symbol: ALN_FIL_ABORT/EN Columns: 6

Desc:

T4 Properties Usage Help

COUNTER DATA STRUCTURE

Data File C5

Offset	CU	CD	DN	OV	UN	UA	PRE	ACC	(Symbol) Description
C5:0	0	0	0	0	0	0	0	0	(INJECTOR_COUNT_) Blade AlignerInjector CountABORT
C5:1	1	0	0	0	0	0	20000	2	(CAP_COUNTER_1) Line ACap Counter #1
C5:2	0	0	0	0	0	0	500	0	(TABLE_INDEXED) New BottleTable Indexedinto Position
C5:3	0	0	0	0	0	0	400	0	(MAGAZ_CTR) MagazineCounter
C5:4	0	0	0	0	0	0	1000	348	(CTU_FREE)
C5:5	0	0	0	0	0	0	2000	0	
C5:6	0	0	0	0	0	0	1000	1	(COUNT_DOWN)
C5:7	0	0	0	0	0	0	500	0	
C5:8	0	0	0	0	0	0	0	0	
C5:9	0	0	0	0	0	0	0	0	
C5:10	0	0	0	0	0	0	0	0	

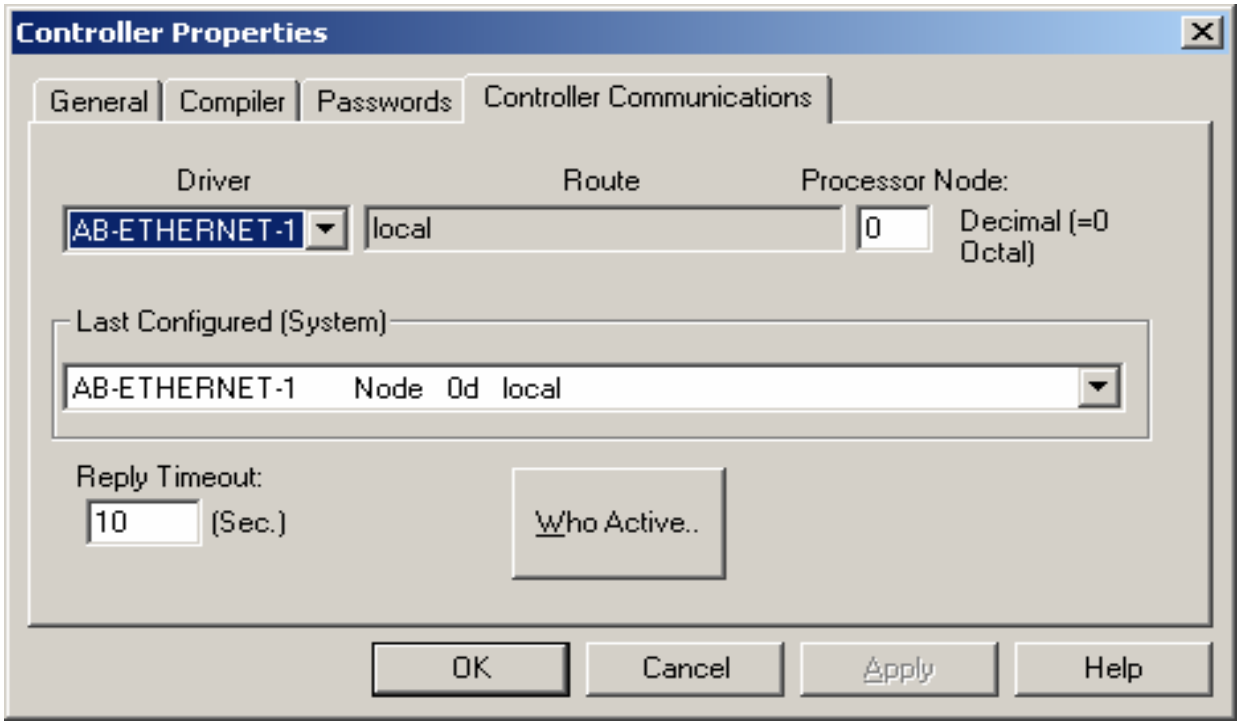
C5:0/CU Radix:

Symbol: INJECTOR_COUNT_/CU Columns: 8

Desc:

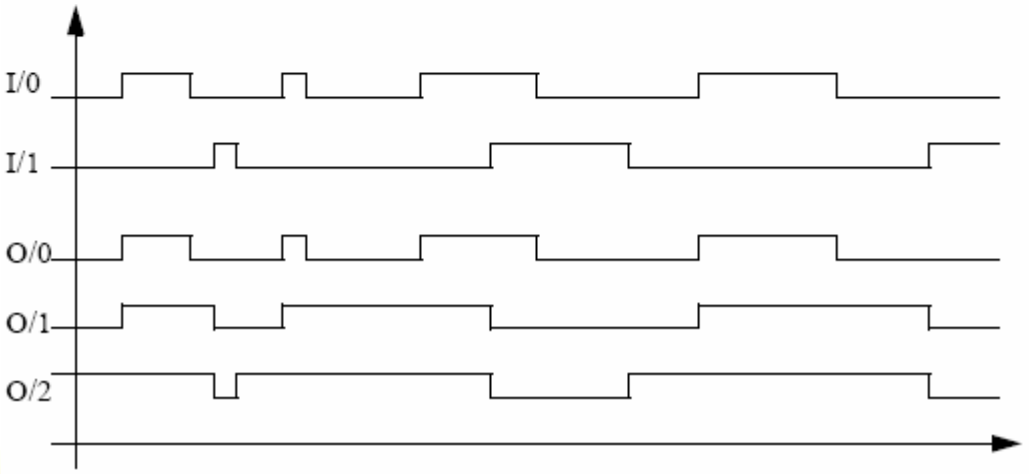
C5 Properties Usage Help

PLC COMMUNICATION SETUP

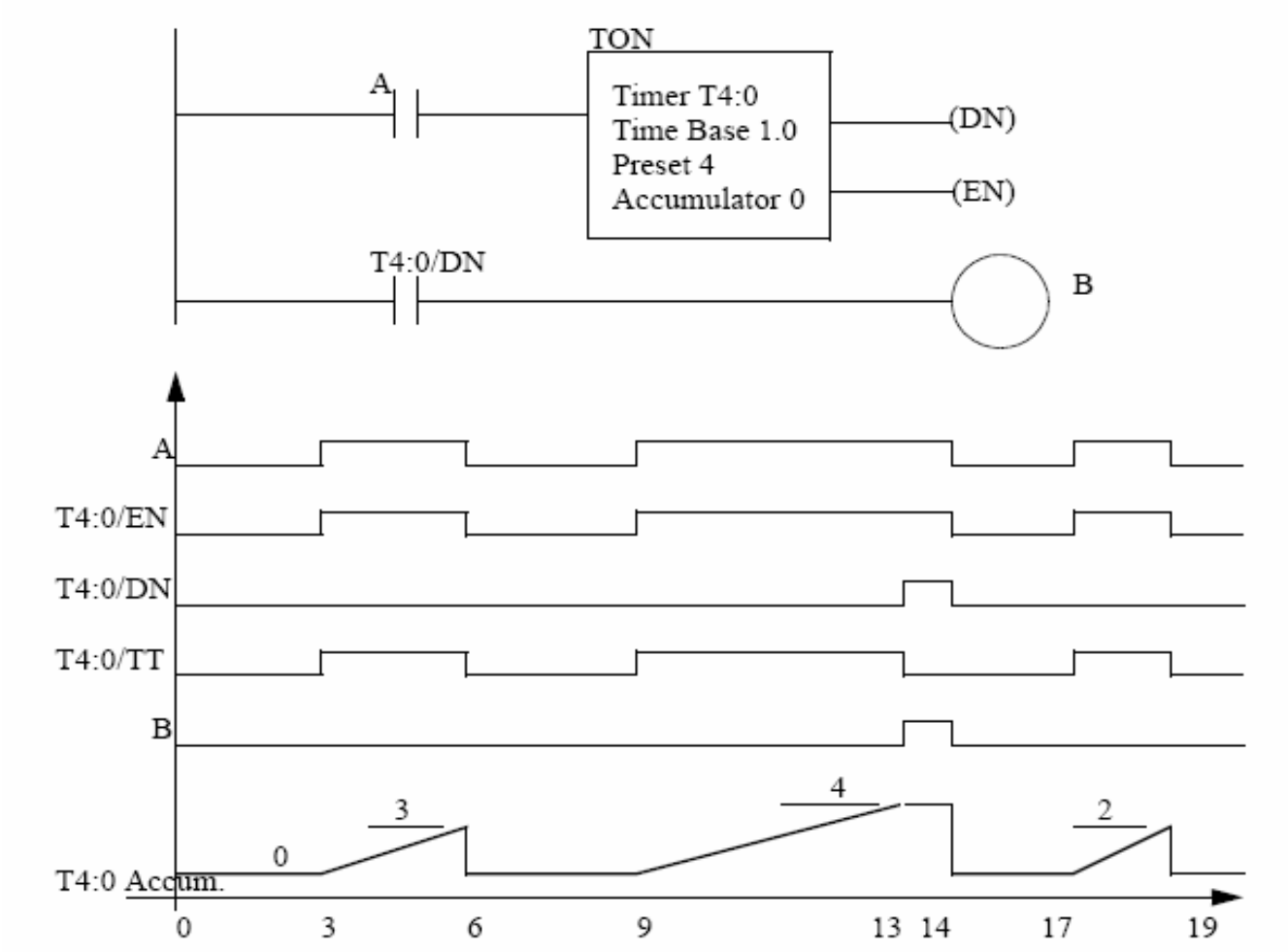


LATCH/UNLATCH EXAMPLE

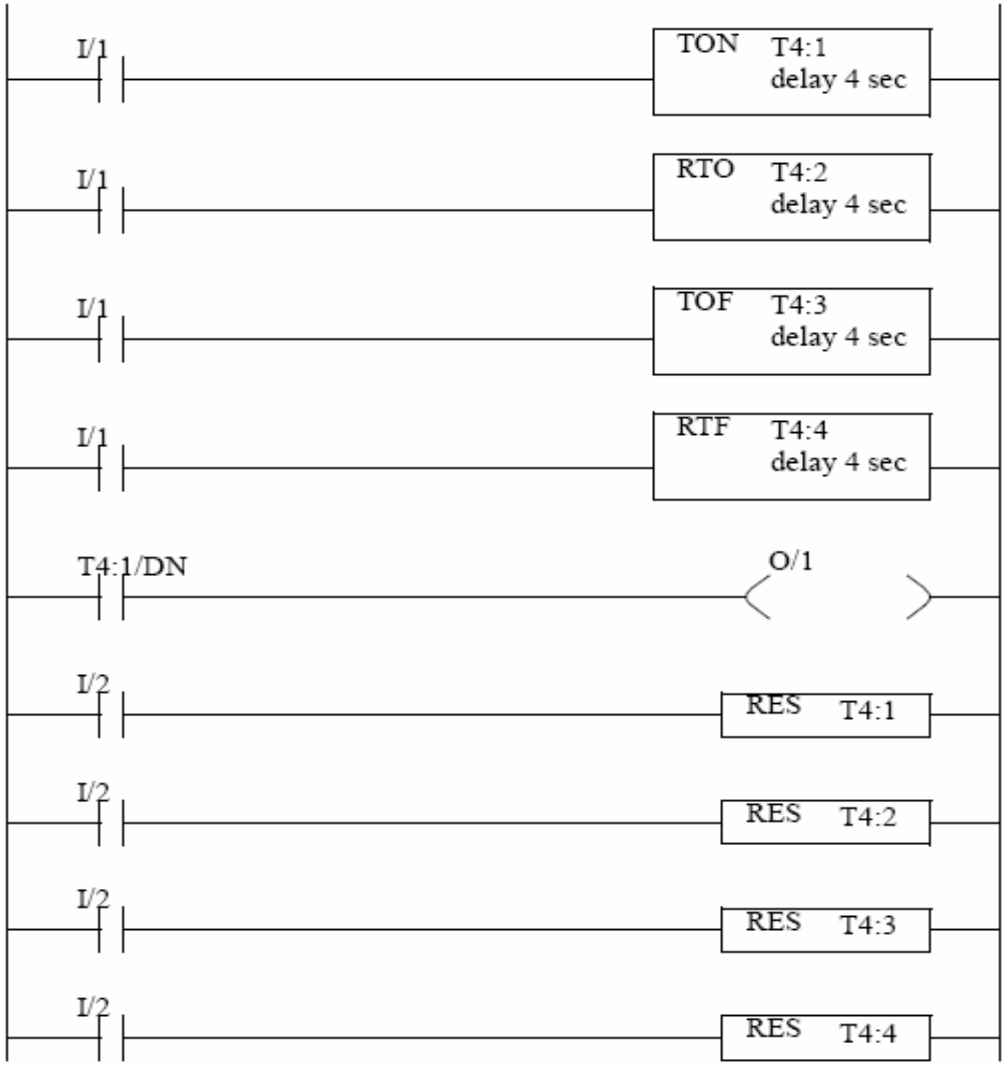




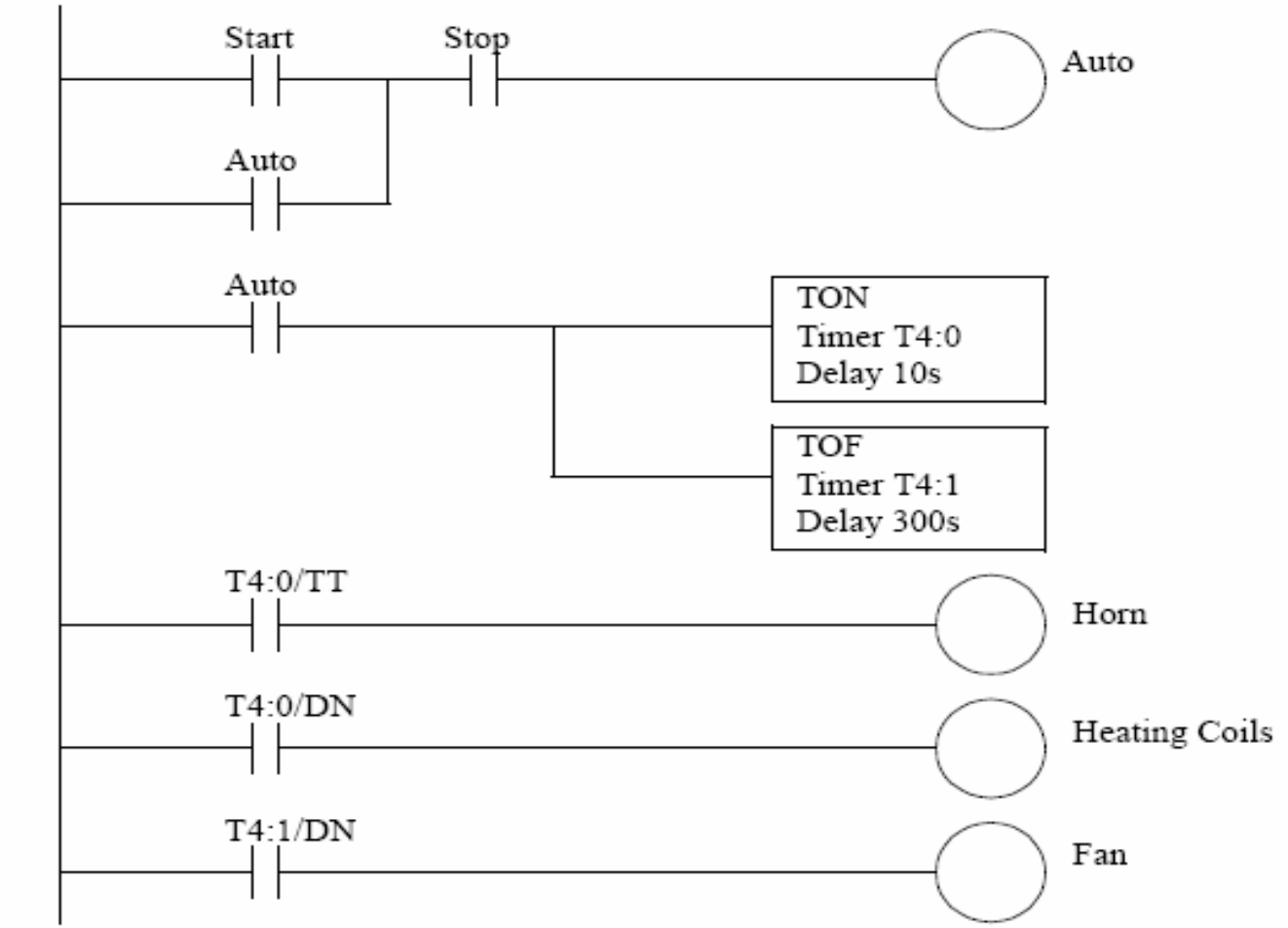
TON TIMER


TON
A
Timer T4:0
Time Base 1.0
Preset 4
Accumulator 0
(DN)
(EN)
T4:0/DN
B
A
T4:0/EN
T4:0/DN
T4:0/TT
B
T4:0 Accum.
0
3
4
2
0
3
6
9
13
14
17
19

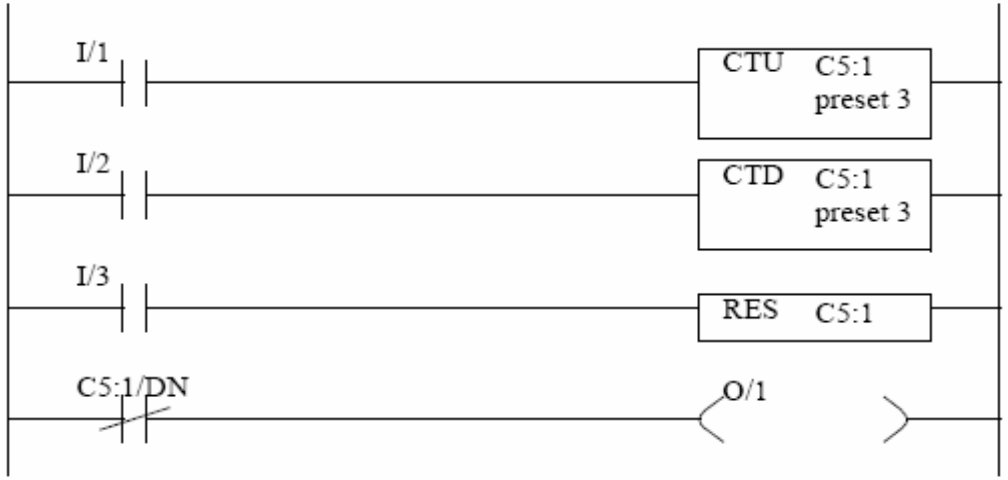
TIMER PROGRAM EXAMPLE

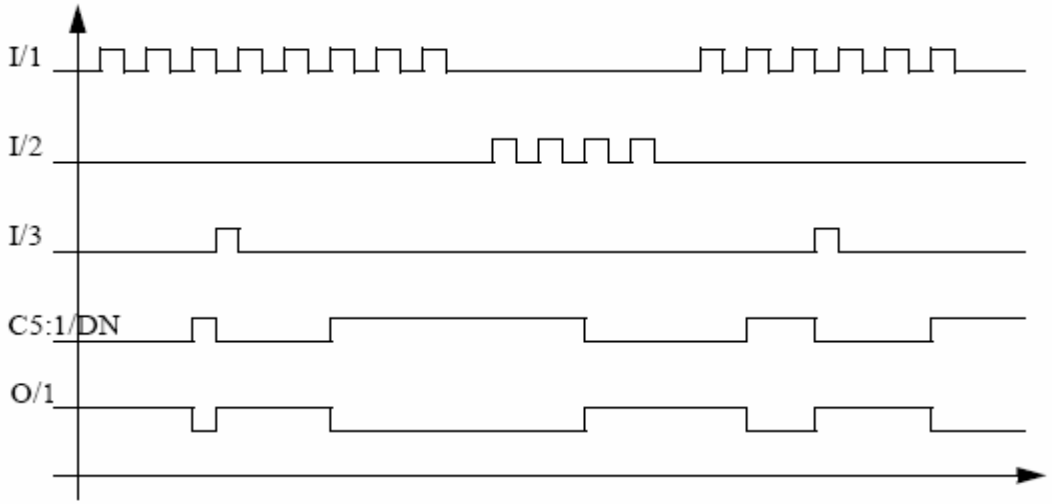


TIMER PROGRAM EXAMPLE

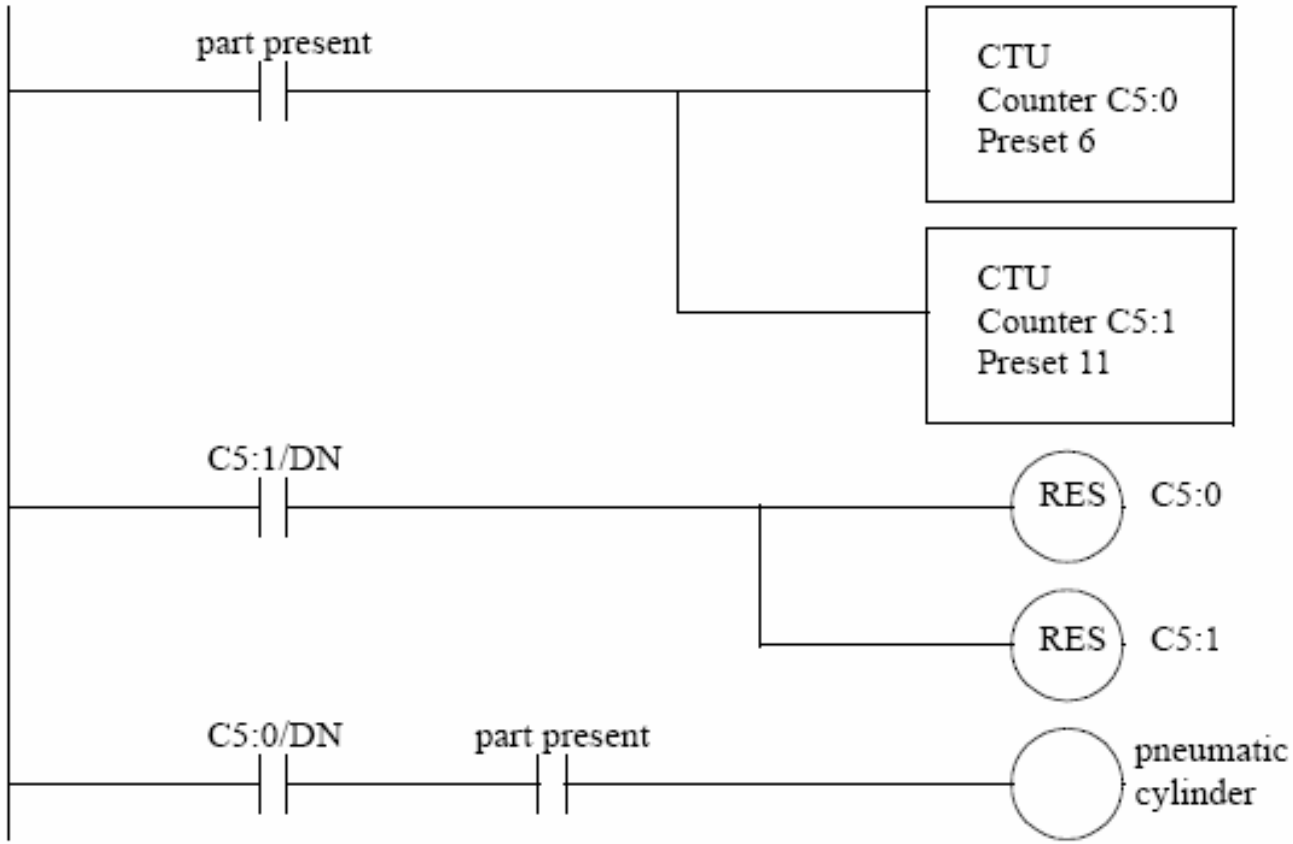


COUNTER PROGRAM EXAMPLE

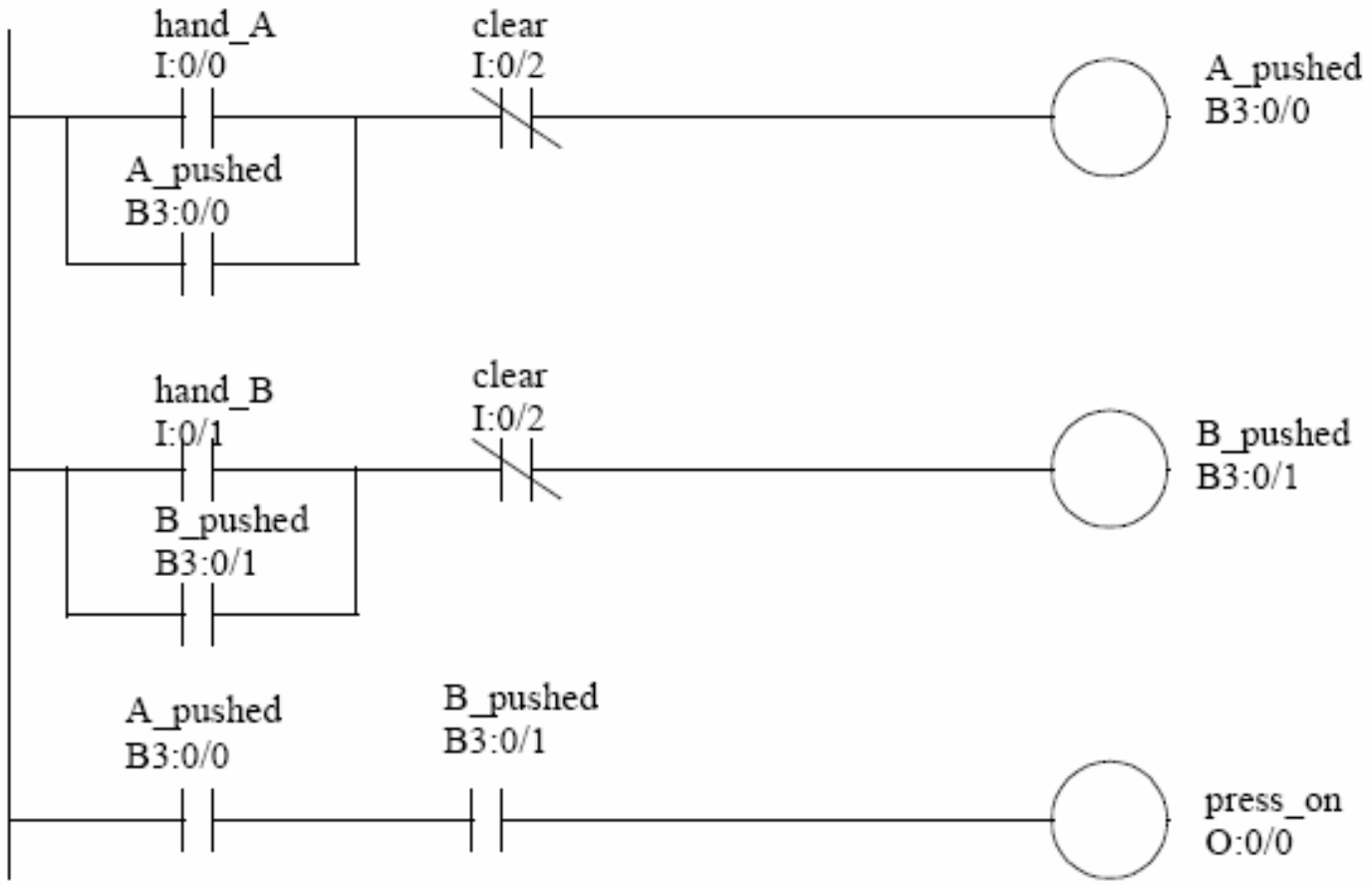




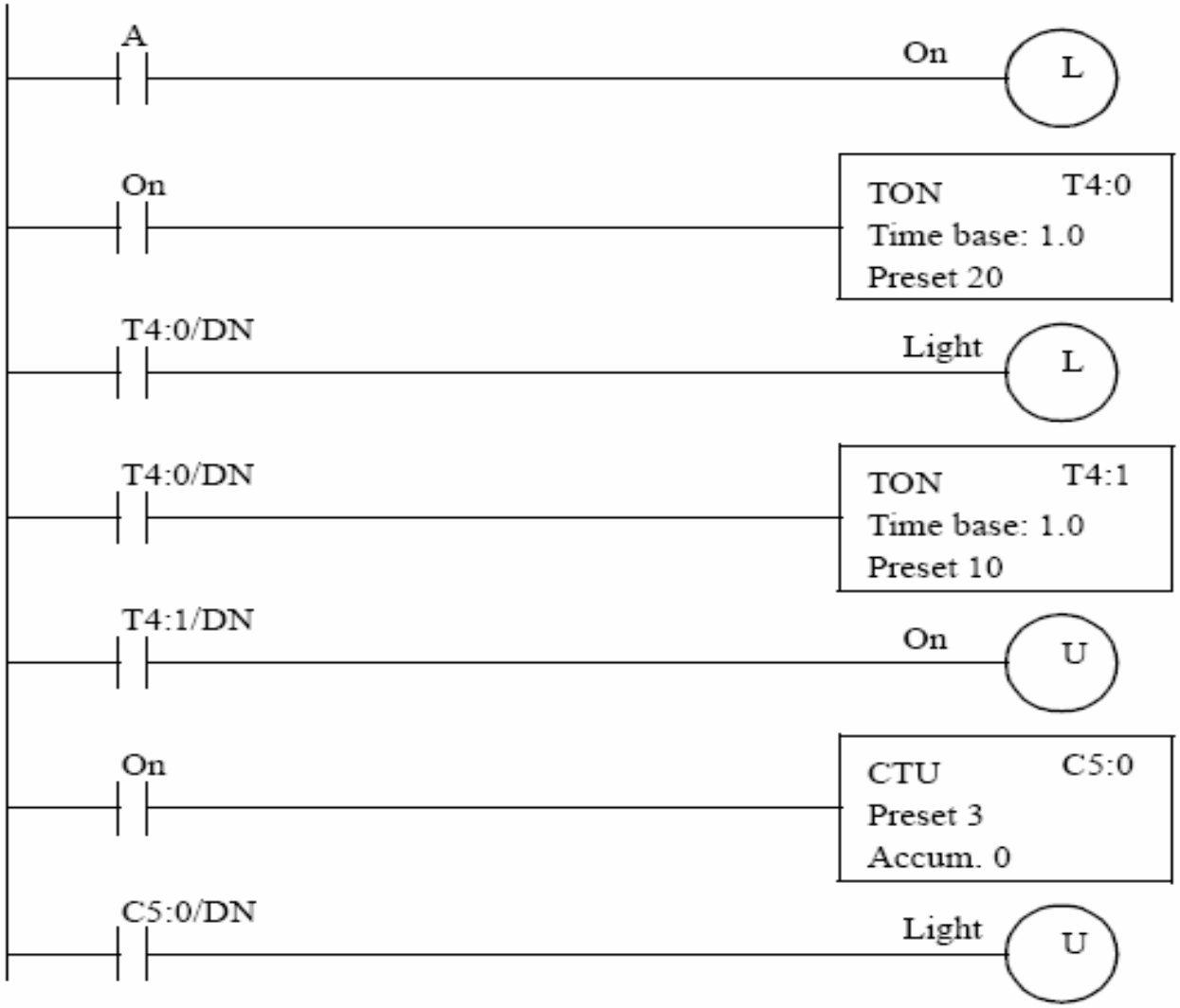
COUNTER PROGRAM EXAMPLE



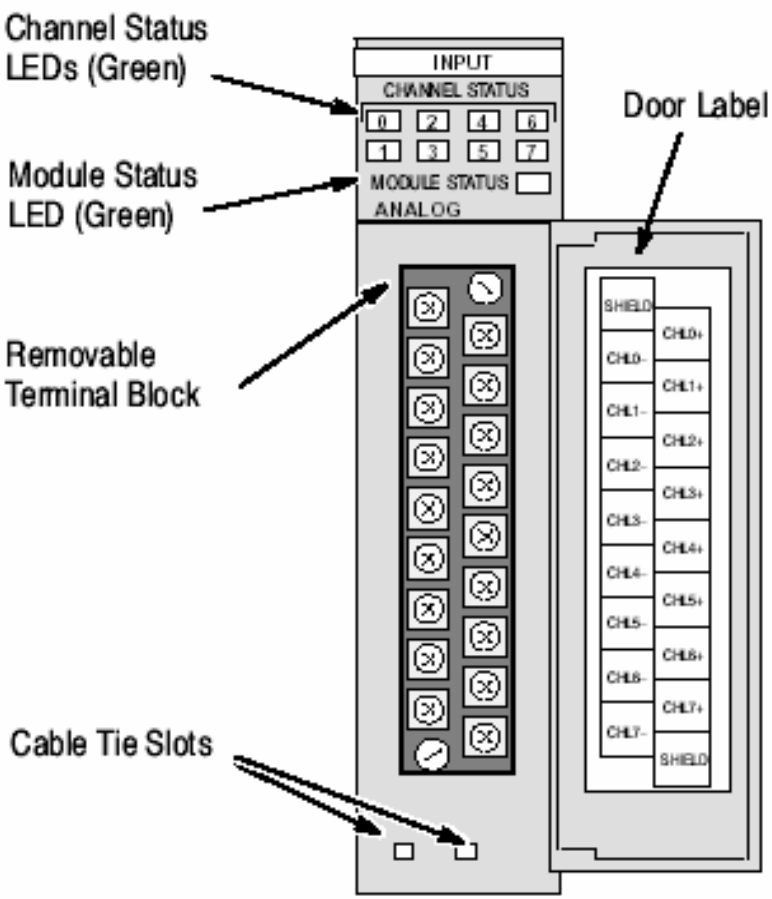
USE OF INTERNAL BIT



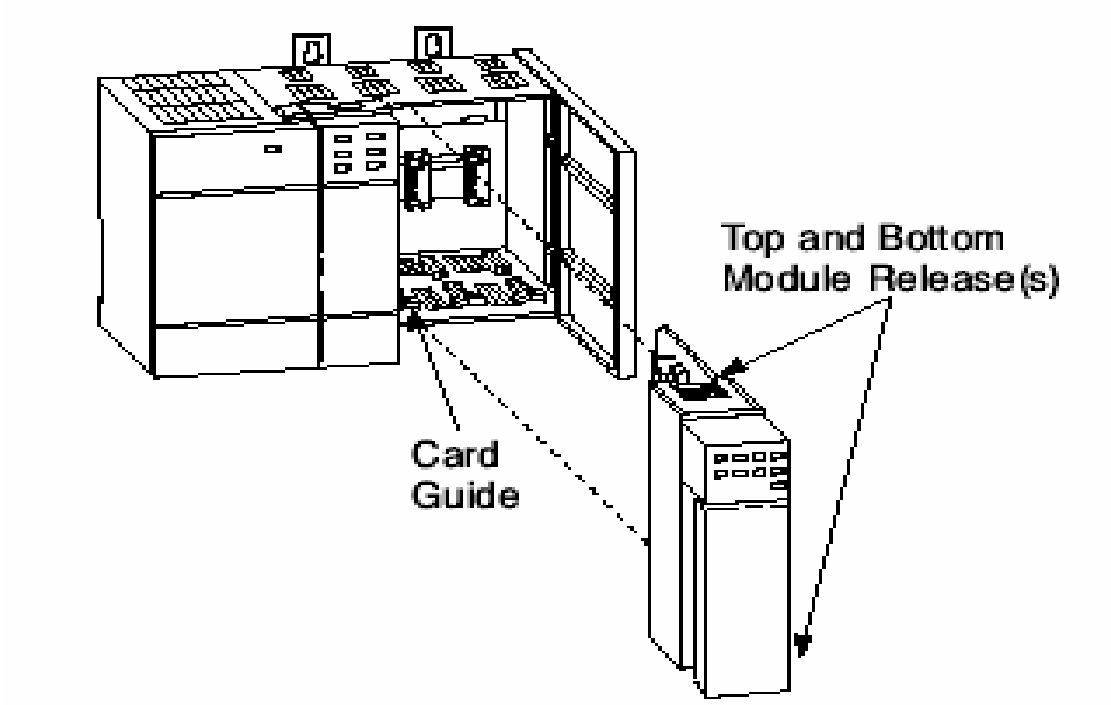
TIMER/COUNTER EXAMPLE



ANALOG INPUT MODULE



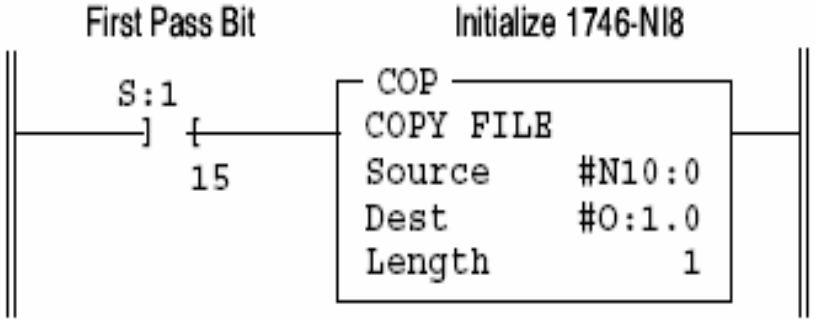
MODULE INSERT/RELEASE



AI PROGRAM CONFIGURATION

DATA FILE N10

Offset	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
N10:0	0	0	0	0	1	0	1	1	0	0	0	0	0	0	0	0



On power up, the first pass bit (S:1/15) is set for one scan, enabling the COPY instruction that transfers the channel configuration word 0. This configures and enables the channel.

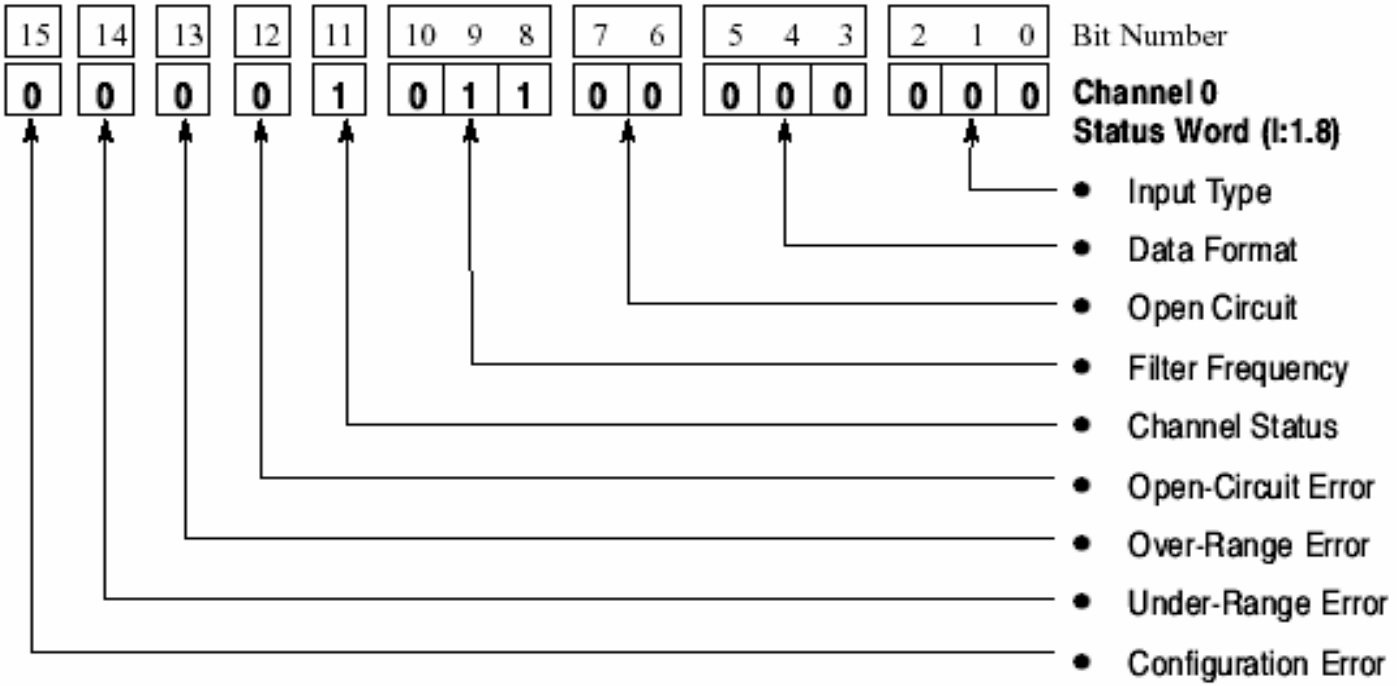
INPUT DATA & INPUT IMAGE

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	Bit Number
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	**Channel 0 Data Word (variable input data)**

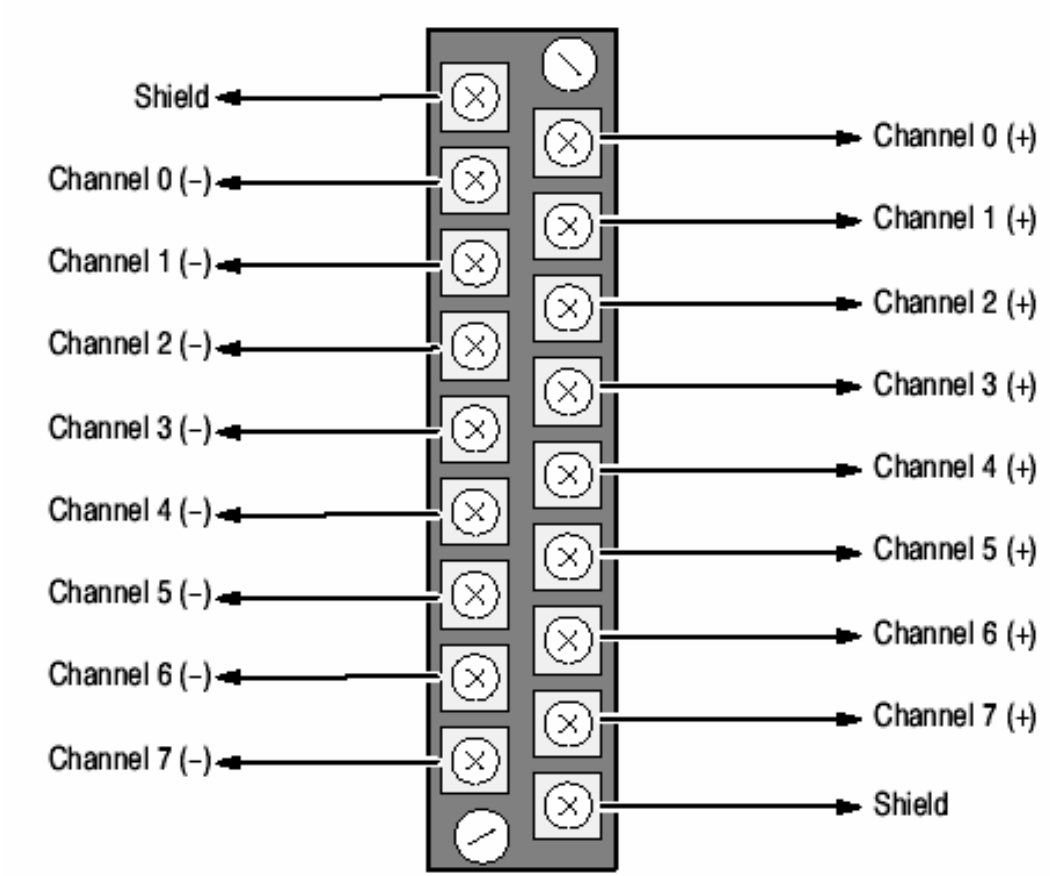
1746-NI8 Module Input Image – Data Word

			Class 1	Class 3
I:1.0	channel 0 data word	16 bit integer	•	•
I:1.1	channel 1 data word	16 bit integer	•	•
I:1.2	channel 2 data word	16 bit integer	•	•
I:1.3	channel 3 data word	16 bit integer	•	•
I:1.4	channel 4 data word	16 bit integer	•	•
I:1.5	channel 5 data word	16 bit integer	•	•
I:1.6	channel 6 data word	16 bit integer	•	•
I:1.7	channel 7 data word	16 bit integer	•	•

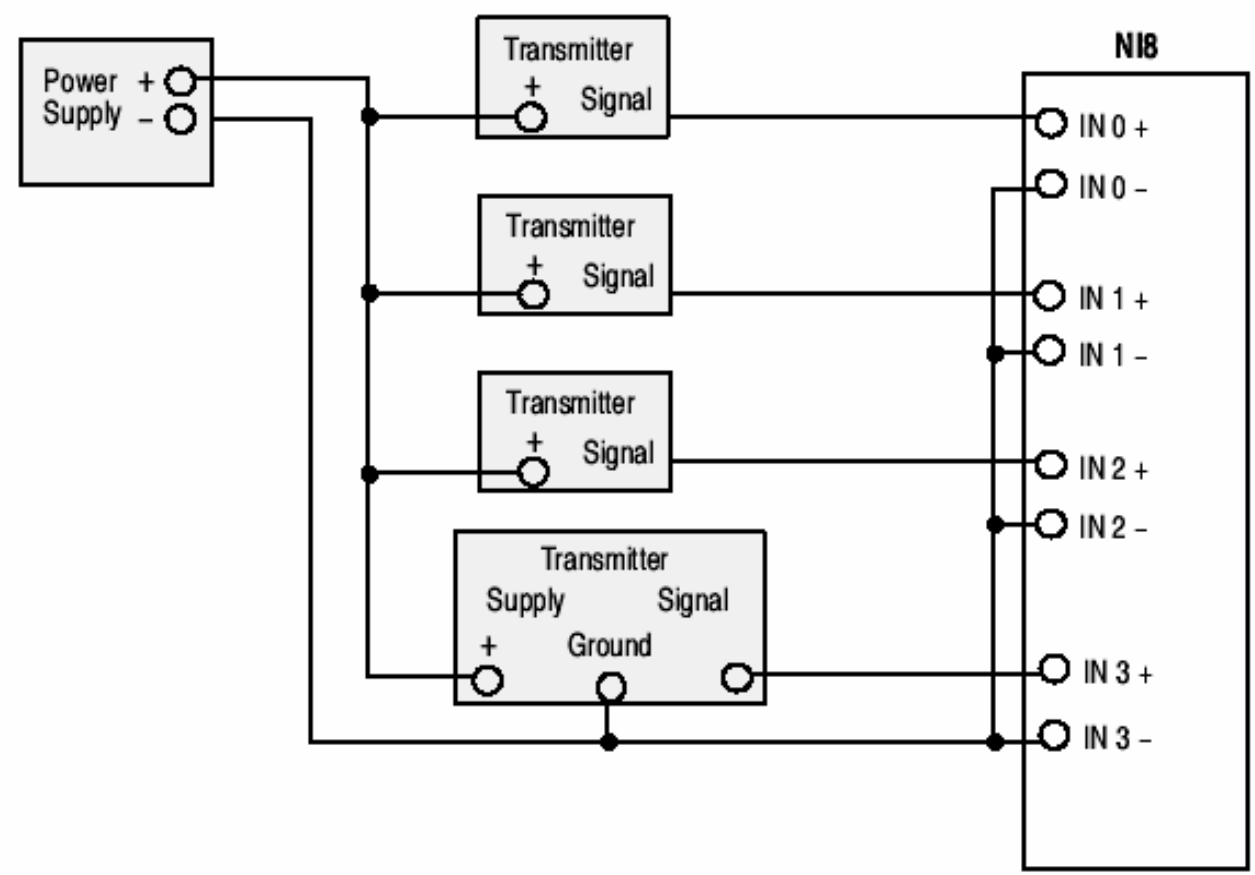
MODULE DIAGNOSTICS



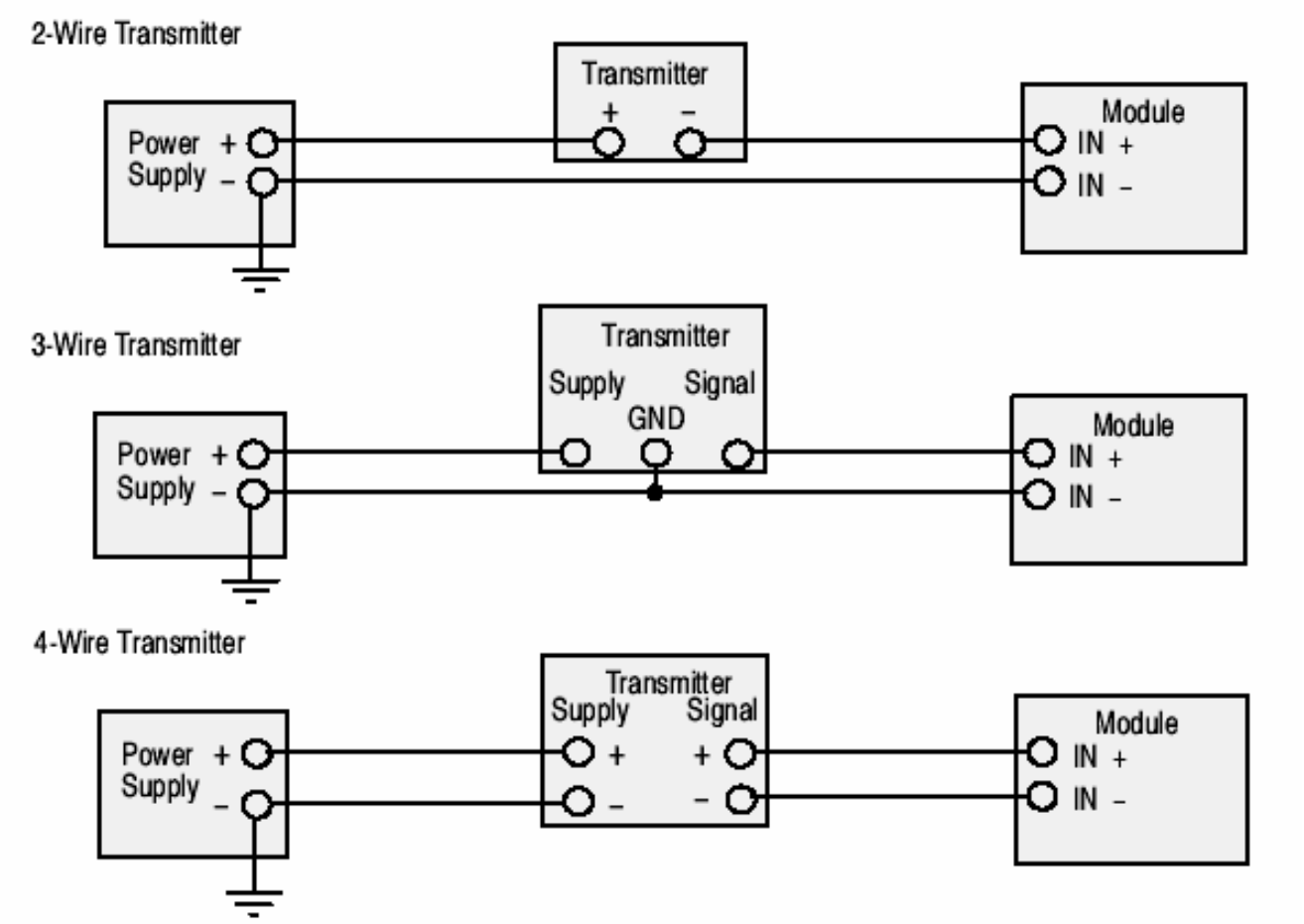
MODULE TERMINAL BLOCK



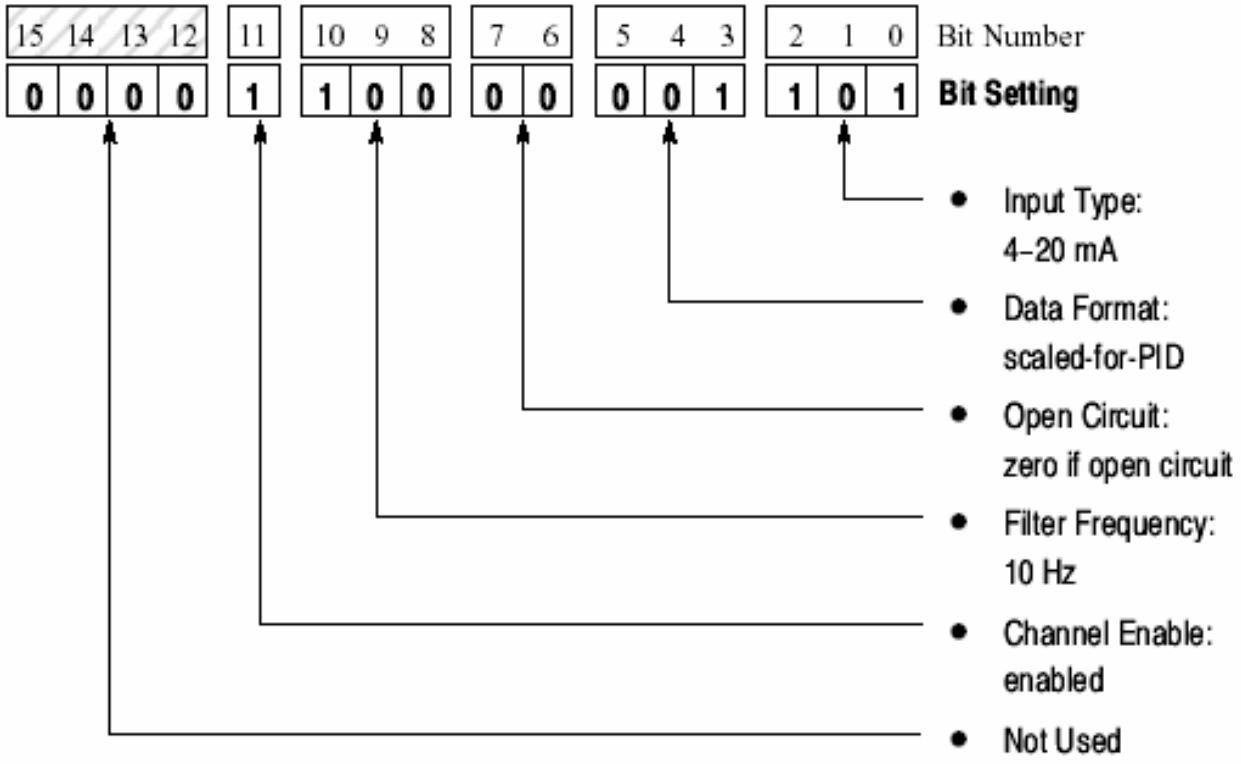
WIRING SCHEMATIC S/E



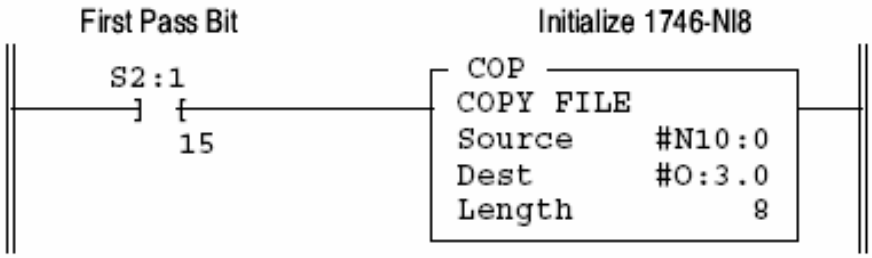
WIRING SCHEMATIC 2/3/4W



INITIAL PROGRAMMING



INITIALIZATION



On power up, bit S2:1/15 is set for the first program scan, and integer file N10 is sent to the eight 1746-NI8 channel configuration words.

DATA FILE N10

Offset	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
N10:0	0	0	0	0	1	1	0	0	0	0	0	0	1	1	0	1
N10:1	0	0	0	0	1	1	0	0	0	0	0	0	1	1	0	1
N10:2	0	0	0	0	1	1	0	0	0	0	0	0	1	1	0	1
N10:3	0	0	0	0	1	1	0	0	0	0	0	0	1	1	0	1
N10:4	0	0	0	0	1	1	0	0	0	0	0	0	1	1	0	1
N10:5	0	0	0	0	1	1	0	0	0	0	0	0	1	1	0	1
N10:6	0	0	0	0	1	1	0	0	0	0	0	0	1	1	0	1
N10:7	0	0	0	0	1	1	0	0	0	0	0	0	1	1	0	1

SIGNAL SCALING

Condition:

A transducer is being used to measure temperature. The 4–20 mA signal is proportional to 100–500°C (212–932°F). The channel data word is in engineering units format.The channel data = 5500.

Find: "Real" units equivalent in °F.

Equation:

Real unit = channel value x scaling factor x $\left(\frac{S_{HIGH} - S_{LOW}}{Input\ range} \right)$

= 5500 x (1 uA) x $\left(\frac{932^o F - 212^o F}{20\,mA - 4\,mA} \right) = 247{,}5^o F$

SIGNAL SCALING

Condition:

A transducer is being used to measure temperature. The 0-10 V dc signal is proportional to 0-10°C. The channel data word is in engineering units format. The channel data = 5000.

Find: "Real" units equivalent in °C.

Equation:

Real unit = channel value x scaling factor x $\left(\frac{S_{HIGH} - S_{LOW}}{Input\ range}\right)$

= 5000 x (1 mV) x $\left(\frac{10^{o}C - 0^{o}C}{10V - 0V}\right) = 5{,}00^{o}C$

SIGNAL SCALING

Condition:

A transducer is being used to measure temperature. The 4–20 mA signal is proportional to 100–500°C (212–932°F). The channel data word is in scaled-for-PID format. The channel data = 5500.

Find: "Real" units equivalent in °F.

Equation:

Real unit = $$S_{LOW} + \left[(S_{HIGH} - S_{LOW}) \times \left(\frac{channel\ \ value}{fullscale\ \ PID\ value} \right) \right]$$

$$= 212^{o}F + \left[(932^{o}F - 212^{o}F) \times \left(\frac{5500}{16383} \right) \right] = 453,71^{o}F$$

SIGNAL SCALING

Condition:

A transducer is being used to measure pressure. The –10V dc to +10V dc signal is proportional to 0 to 200 psi. The channel data word is in proportional counts data format. The channel data = 21567.

Find: "Real" units equivalent in psi.

Equation:

Real unit =

$$S_{LOW} + \left[(S_{HIGH} - S_{LOW}) \times \left(\frac{data\ value + |\ proportional\ count\ low\ default\ value\ |}{(proportional\ count\ \ fullscale\ default\ value)} \right) \right]$$

$$0\,psi + \left[(200\,psi - 0\,psi) \times \frac{(21567 + |-32678|)}{(65535)} \right] = 165.55\,psi$$

SIGNAL SCALING

Condition:

A transducer is being used to measure pressure. The –10V dc to +10V dc signal is proportional to 0 to 100 GPM. The channel data word is in 1746-NI4 data format. The channel data = 10000.

Find:

"Real" units equivalent in GPM.

Equation:

Real unit = $channel\ \ value \times \left(\frac{S_{HIGH} - S_{LOW}}{input\ \ range} \right)$

$$= 10000 \times \left(\frac{100\ GPM - 0\ GPM}{65535} \right)$$

$$= 15{,}26\ GPM$$

SIGNAL SCALING

Condition:

A transducer is being used to measure pressure. The –10V dc to +10V dc signal is proportional to 0 to 200 GPM. The channel data word is in user defined data format. The lower scale limit (configuration word 8) = 0. The upper scale limit (configuration word 9) = 20000. The channel data word = 16600.

Find:

"Real" units equivalent in GPM.

Equation:

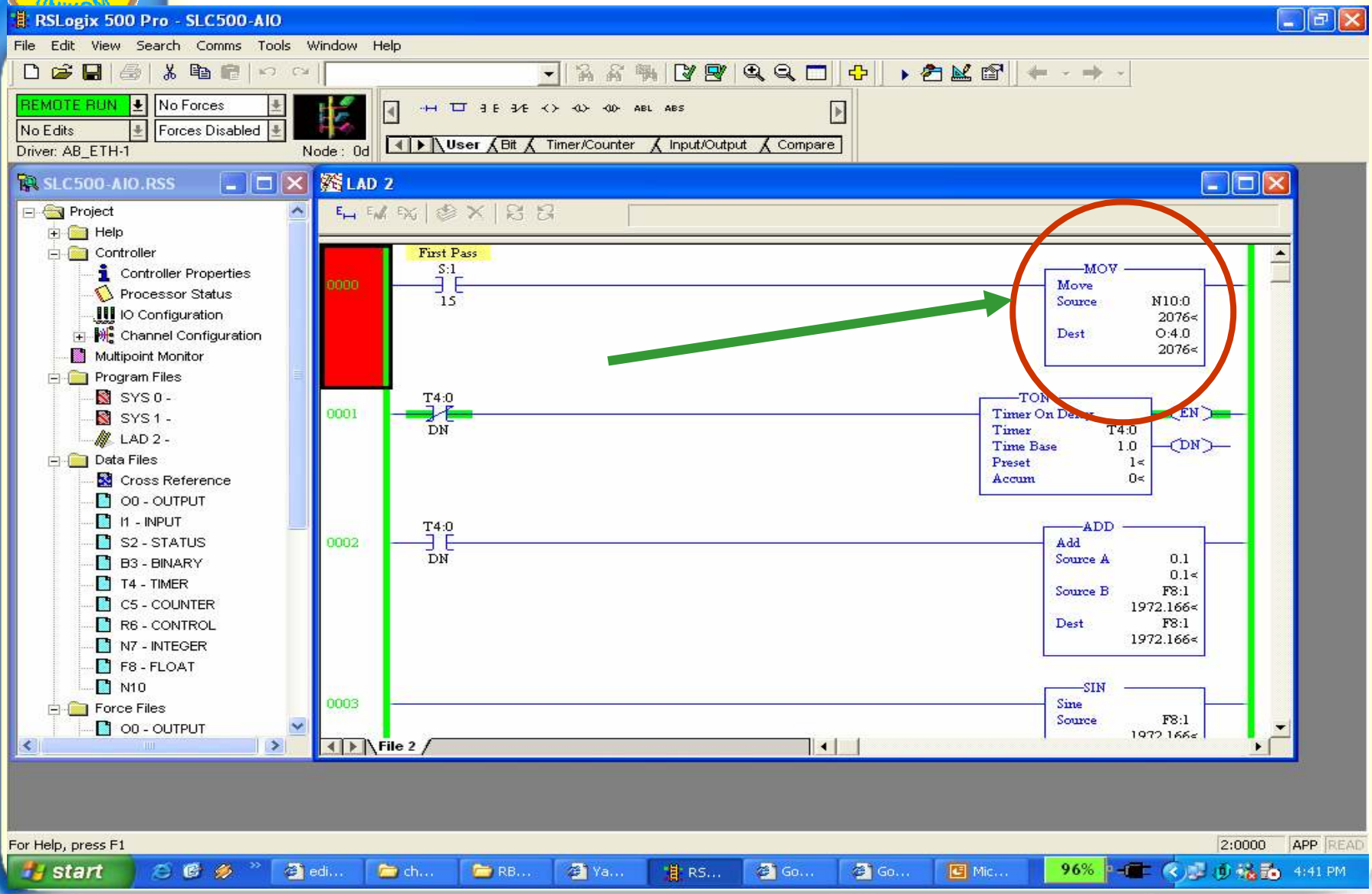
$$\text{Real unit} = \text{channel data word} \times \left[\frac{(S_{HIGH} - S_{LOW})}{(\text{upper scale limit} - \text{lower scale limit})} \right]$$

$$= 16600 \times \left[\frac{(200\ psi - 0\ psi)}{(20000 - 0)} \right] = 166.00\ psi$$

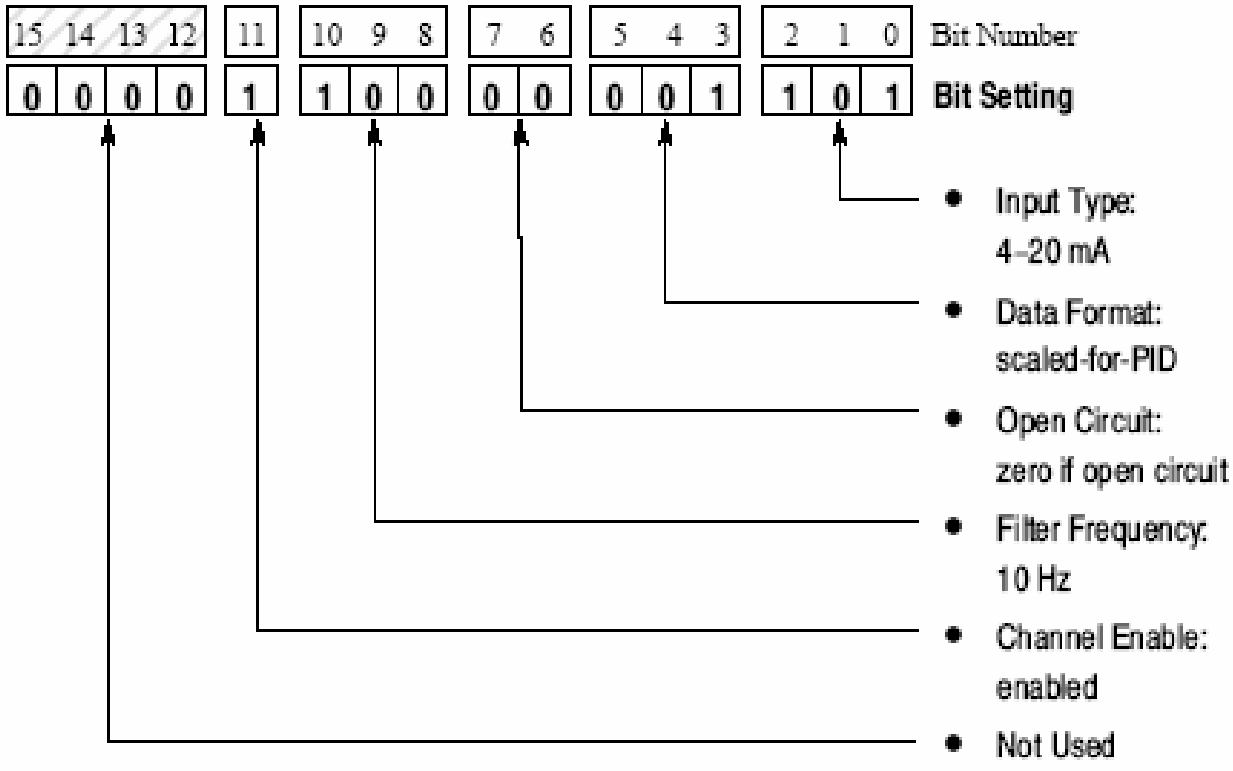
Analog Input BIT DEFINITIONS

Bit(s)	Define	These bit settings																Indicate this
		15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
0–2	Input Type														0	0	0	± 10 V dc
															0	0	1	1–5V dc
															0	1	0	0–5V dc
															0	1	1	0–10V dc
															1	0	0	0–20 mA
															1	0	1	4–20 mA
															1	1	0	± 20 mA
															1	1	1	0–1 mA
3–5	Data Format											0	0	0				Engineering Units
												0	0	1				Scaled-for-PID
												0	1	0				Proportional Counts
												0	1	1				1746-NI4 Data Format
												1	0	0				User Defined (Class 3)
												1	0	1				User Defined (Class 3)
																		Illegal (configuration error)
																		Illegal (configuration error)
6 and 7	Open Circuit									0	0							Zero
										0	1							Upscale
										1	0							Downscale
																		Illegal (configuration error)
8–10	Filter Frequency						0	0	0									No Filter
							0	0	1									75 Hz
							0	1	0									50 Hz
							0	1	1									20 Hz
							1	0	0									10 Hz
							1	0	1									5 Hz
							1	1	0									2 Hz
							1	1	1									1 Hz
11	Channel Status					0												Channel Disabled
						1												Channel Enabled
12	Open Circuit Error				0													No Error
					1													Open Circuit Detected
13	Over-Range Error			0														No error
				1														Over-Range Condition
14	Under-Range Error		0															No Error
			1															Under-Range Condition
15	Configuration Error	0																No Error
		1																Configuration Error

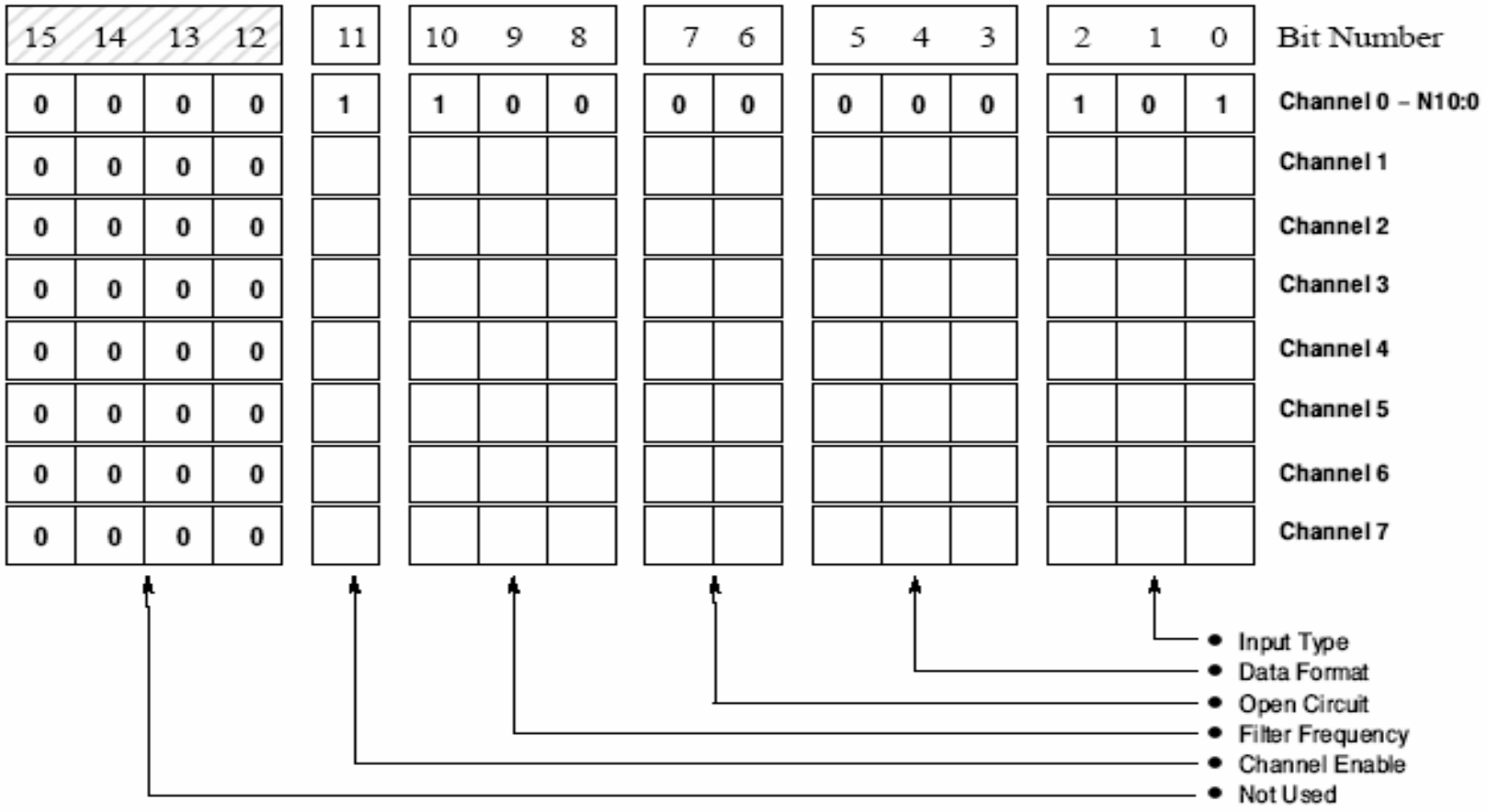
ANALOG INPUT CONFIGURATION



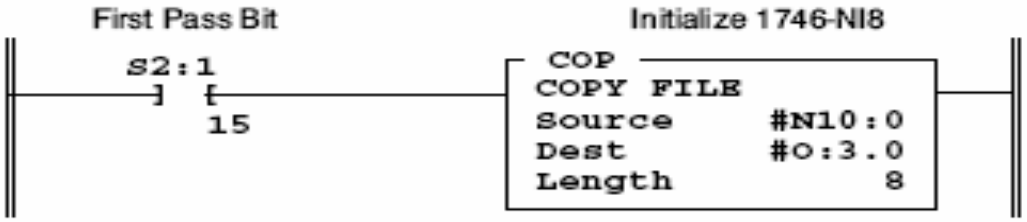
ANALOG INPUT INITIAL PROGRAMMING



CHANNEL CONFIGURATION WORKSHEET



ANALOG INPUT LADDER LOGIC CONFIGURATION

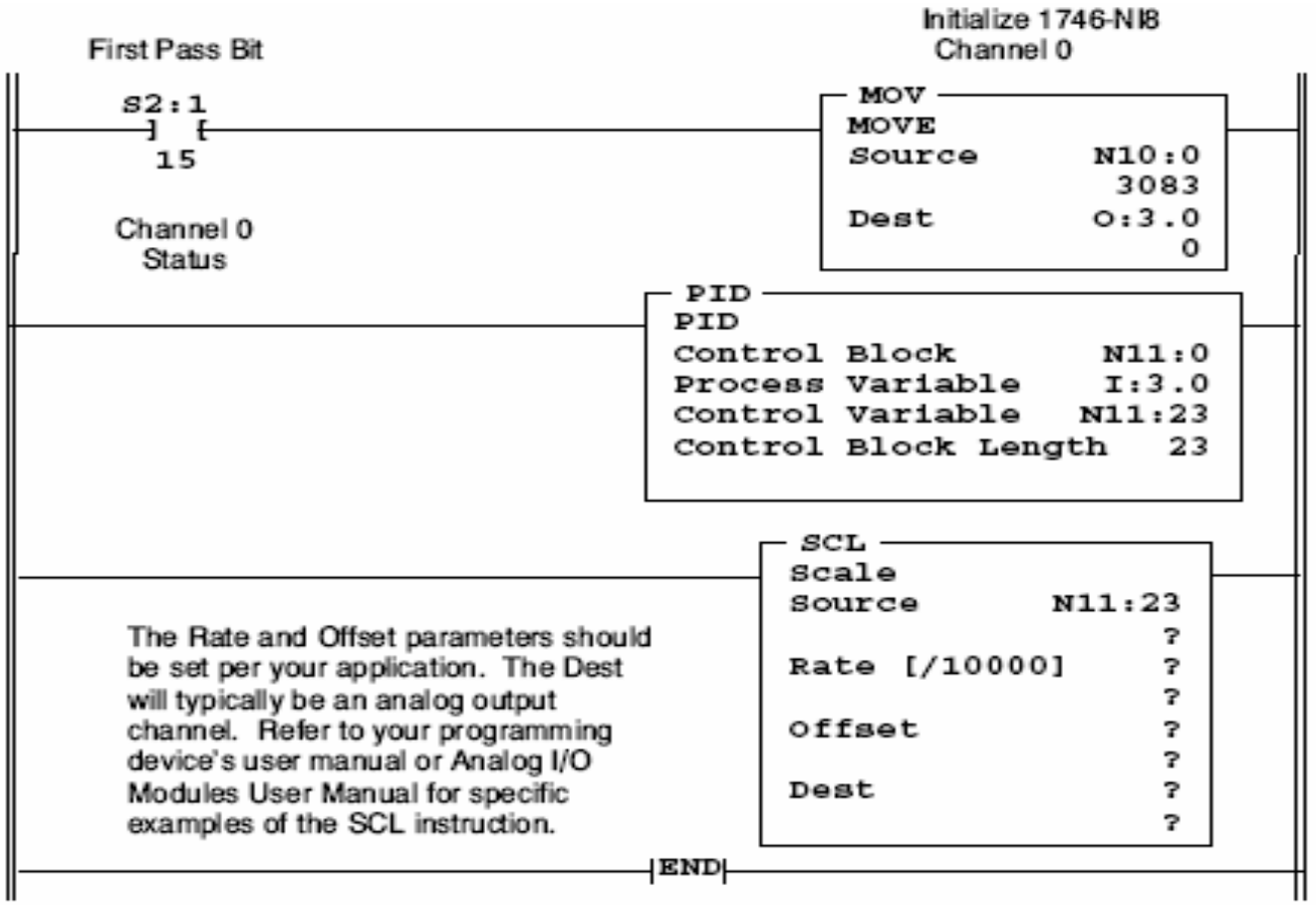


On power up, bit S2:1/15 is set for the first program scan, and integer file N10 is sent to the eight 1746-NI8 channel configuration words.

DATA FILE N10

Offset	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
N10:0	0	0	0	0	1	1	0	0	0	0	0	0	1	1	0	1
N10:1	0	0	0	0	1	1	0	0	0	0	0	0	1	1	0	1
N10:2	0	0	0	0	1	1	0	0	0	0	0	0	1	1	0	1
N10:3	0	0	0	0	1	1	0	0	0	0	0	0	1	1	0	1
N10:4	0	0	0	0	1	1	0	0	0	0	0	0	1	1	0	1
N10:5	0	0	0	0	1	1	0	0	0	0	0	0	1	1	0	1
N10:6	0	0	0	0	1	1	0	0	0	0	0	0	1	1	0	1
N10:7	0	0	0	0	1	1	0	0	0	0	0	0	1	1	0	1

INTERFACE TO PID INSTRUCTION



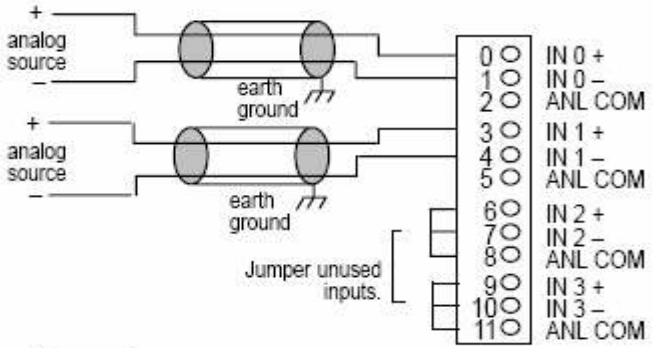
CHANNEL CONFIGURATION

Configure channel 0 of the 1746-NI8 module with the following setup:

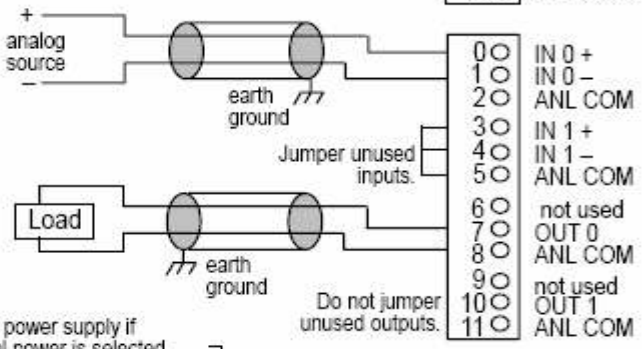
- 4–20 mA input type
- Engineering units data format
- Zero data word in the event of an open circuit
- 10 Hz input filter to reject high frequency noise and provide
- Rejection of 60 Hz line noise

ANALOG I/O CONFIGURATION

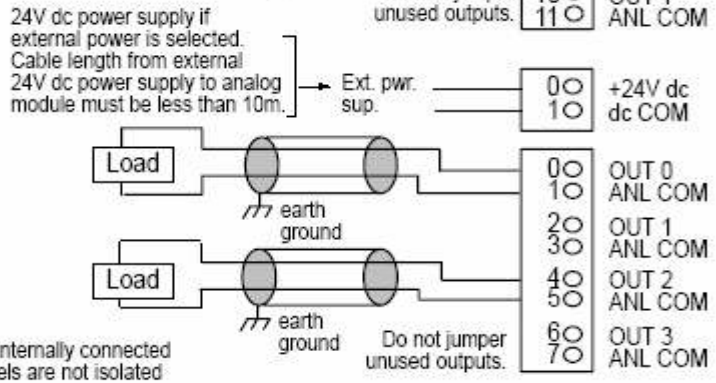
NI4



NIO4I & NIO4V



NO4I & NO4V



Analog commons are internally connected in the module. Channels are not isolated from each other.

ANALOG OUTPUT CONFIGURATION

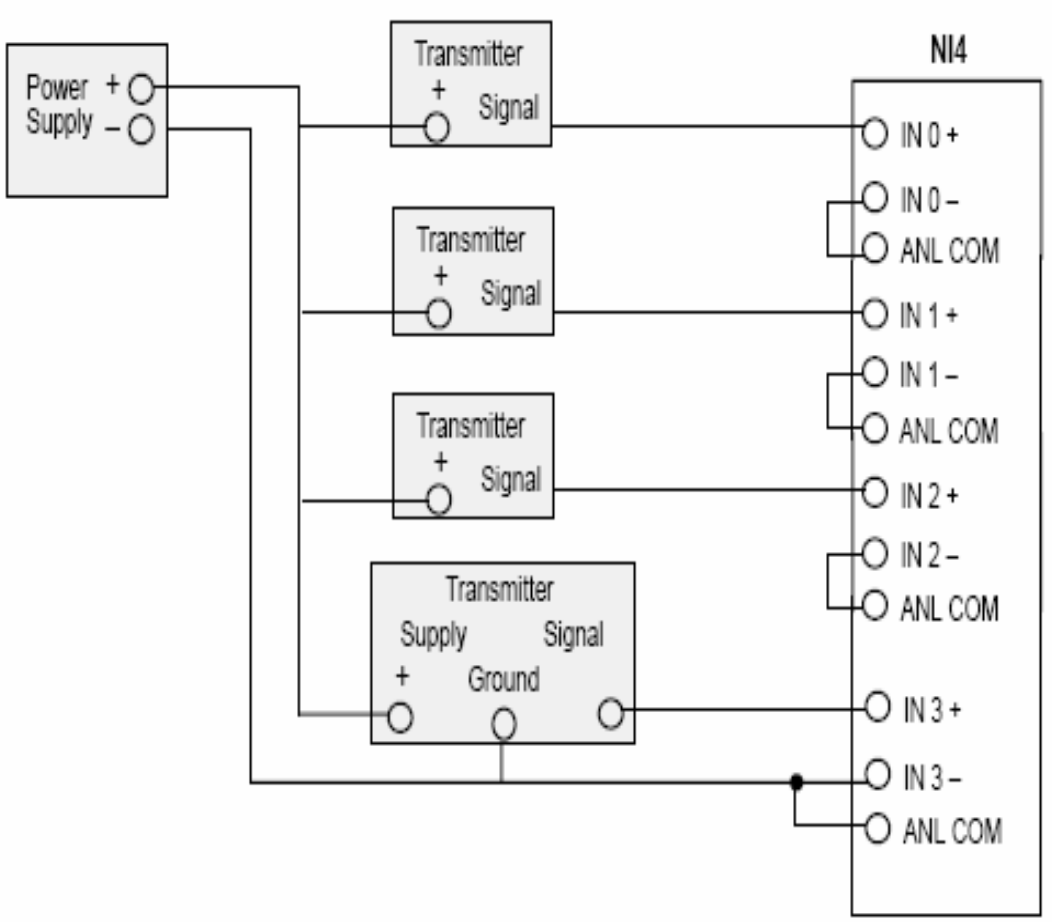
Address	NO4
O:e.0	Output Channel 0
O:e.1	Output Channel 1
O:e.2	Output Channel 2
O:e.3	Output Channel 3

Address	NIO4I, NIO4V
O:e.0	Output Channel 0
O:e.1	Output Channel 1

NO4I, NIO4I	
Current Range	Decimal Representation for Output Word
0 to 21 mA	0 to 32,764
0 to 20 mA	0 to 31, 208
4 to 20 mA	6,242 to 31,208

NO4V, NIO4V	
Voltage Range	Decimal Representation for Output Word
–10 to +10V dc	–32,768 to +32,764
0 to 10V dc	0 to 32,764
0 to 5V dc	0 to 16,384
1 to 5V dc	3,277 to 16,384

TRANSMITTER CONNECTION

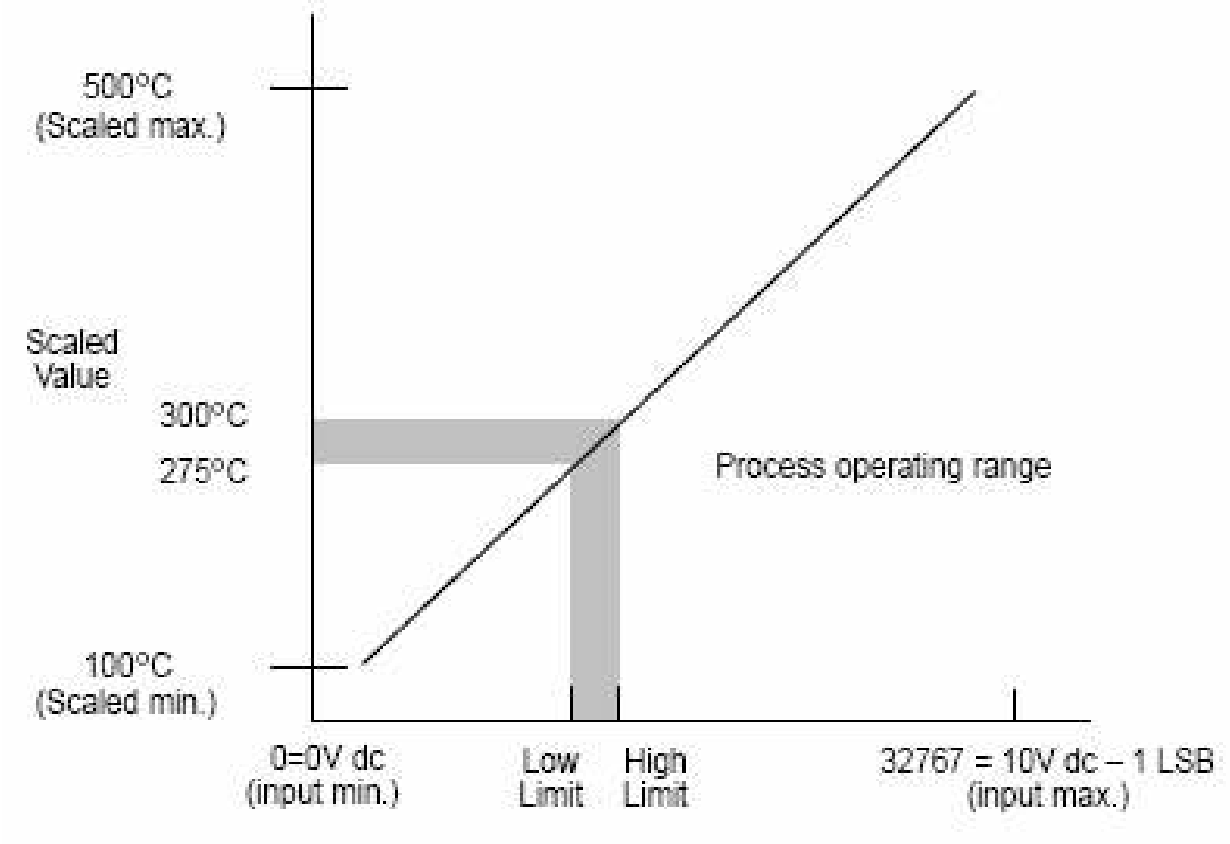


PROGRAMMING EXAMPLE

Assumption:

- The AI module is located in slot 3 of a modular system.
- A temperature transducer with a 0 to 10V dc output is wired to the second input channel on the analog module.
- The transducer voltage signal is proportional to a range of 100° C to 500° C (212° F to 932° F).
- The process temperature must stay between 275° and 300° C (527° F to 572° F). If the temperature deviates from this range, a flag is set and this out-of-range value is not processed. The data is presented in degrees C for monitoring and display purposes.

PROGRAMMING EXAMPLE



AI Scaling Graph

PROGRAMMING EXAMPLE

Scale Calculation:

Scaled value = (input value x slope) + offset

Slope = (scaled max. – scaled min.) / (input max. – input min.)

(500 – 100) / (32,767 – 0) = 400/32,767

Offset = scaled min. – (input min. x slope)

100 – (0 x (400/32,767)) = 100

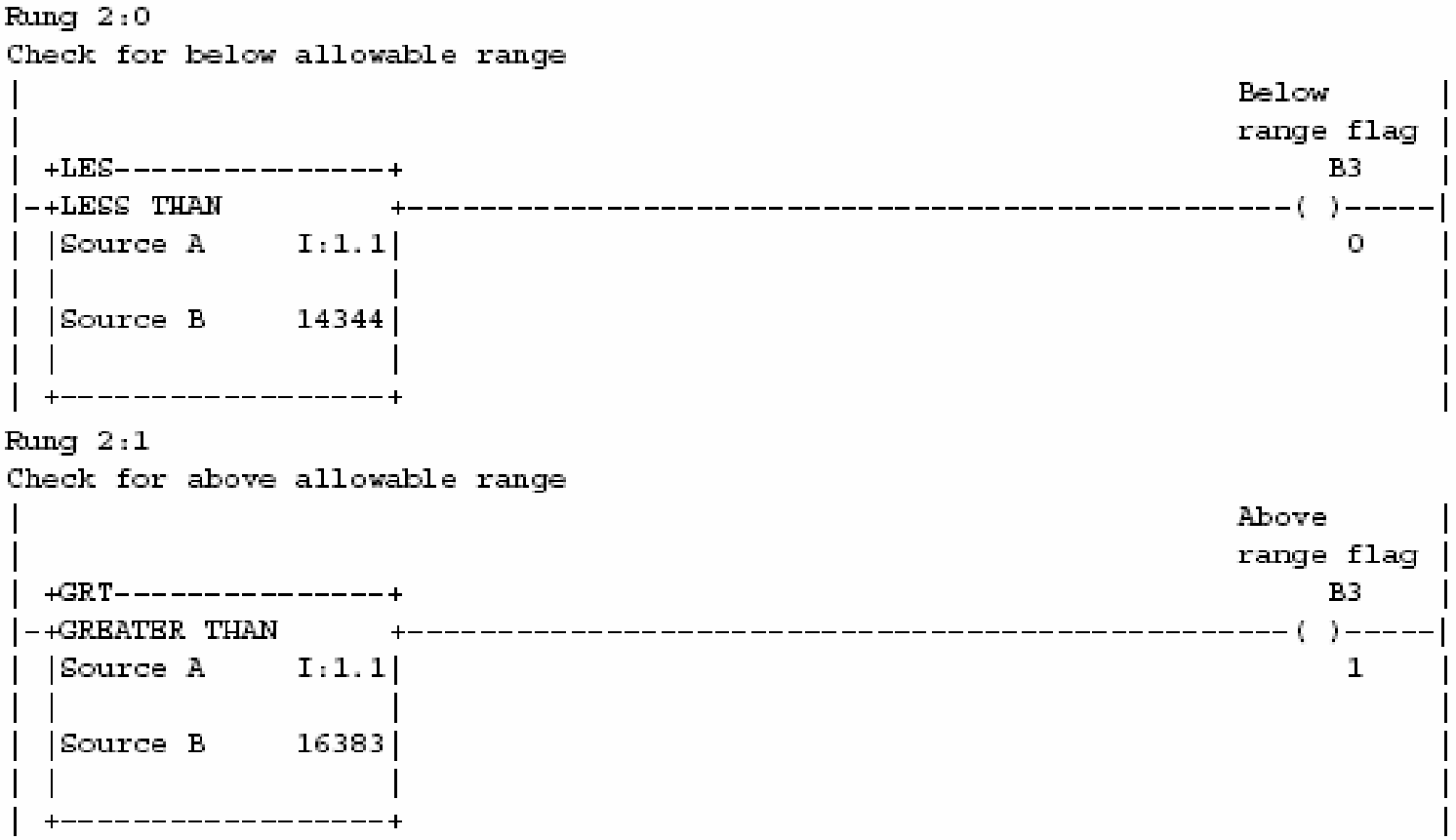
Scaled value = (input value x (400 / 32767)) + 100

Input value = (scaled value – offset) / slope

low limit (275 – 100) / (400/32,767) = 14,344

high limit (300 – 100) / (400/32,767) = 16,393

PROGRAMMING EXAMPLE



PROGRAMMING EXAMPLE

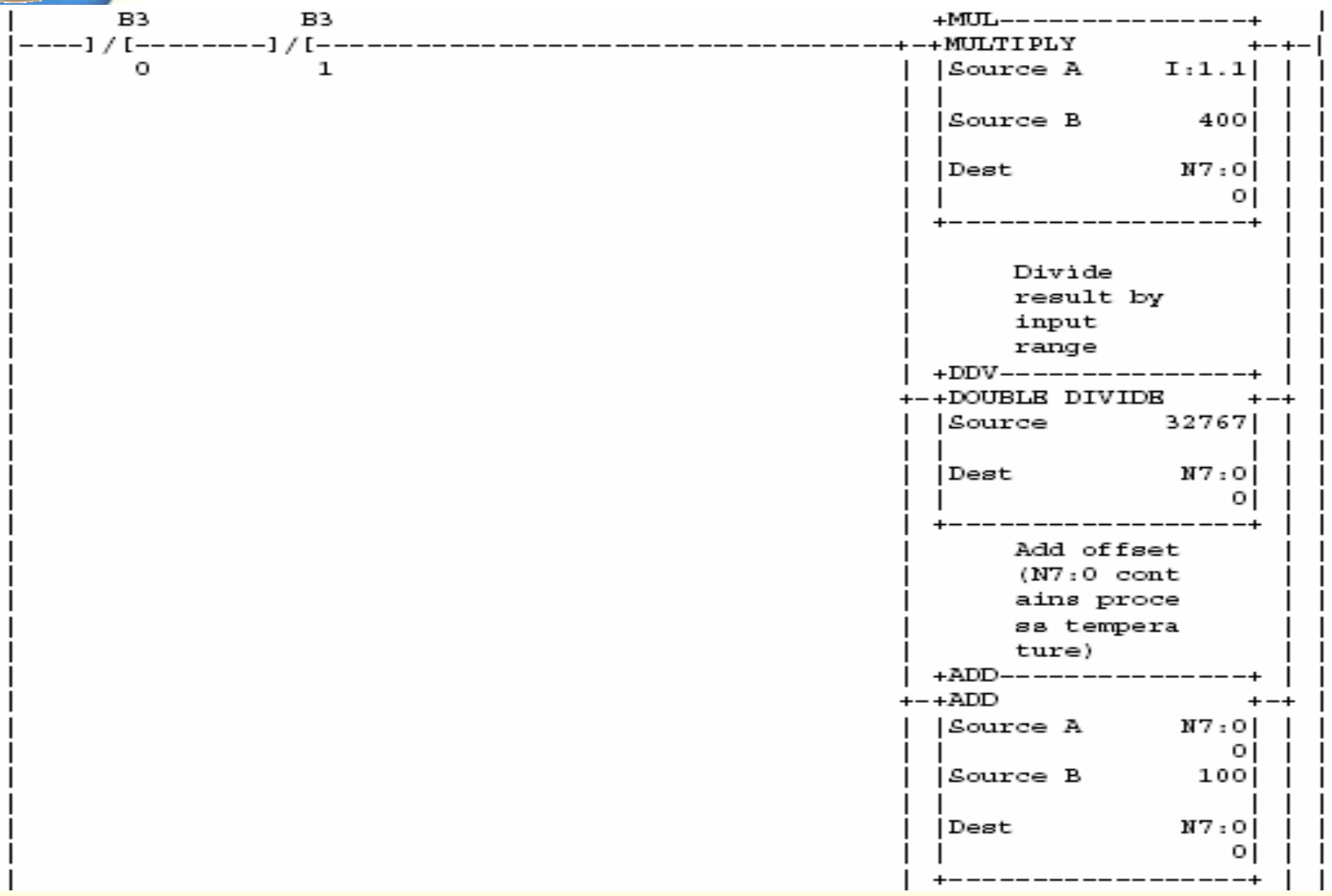
Rung 2:2
Turn on the below range alarm output when analog
input is below acceptible range



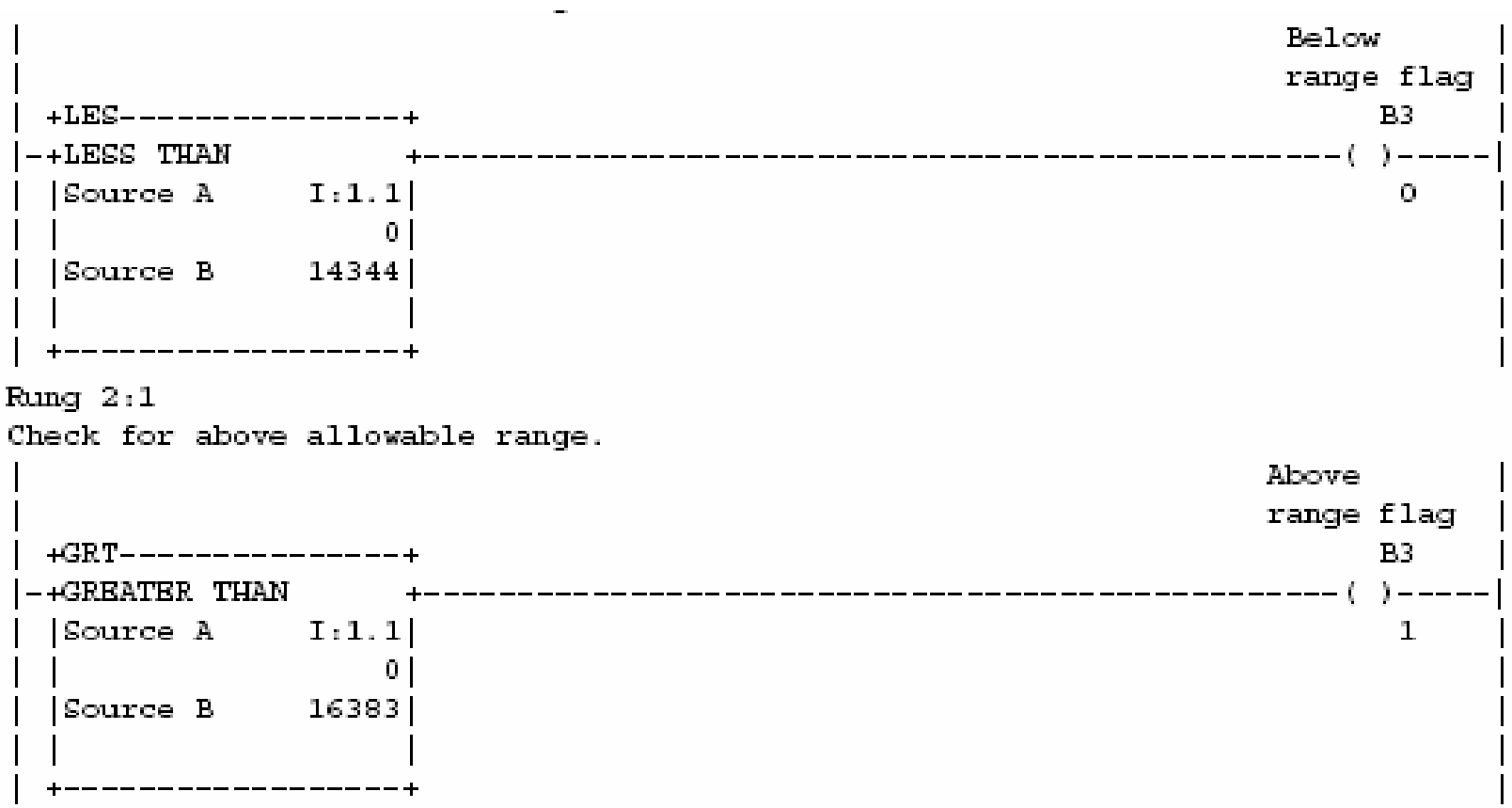
Rung 2:3
Turn on the above range alarm output when analog
input is above acceptible range



PROGRAMMING EXAMPLE



PROGRAMMING EXAMPLE



PROGRAMMING EXAMPLE

Rung 2:2
Turn on the below range alarm output when the analog input is below the acceptible range.



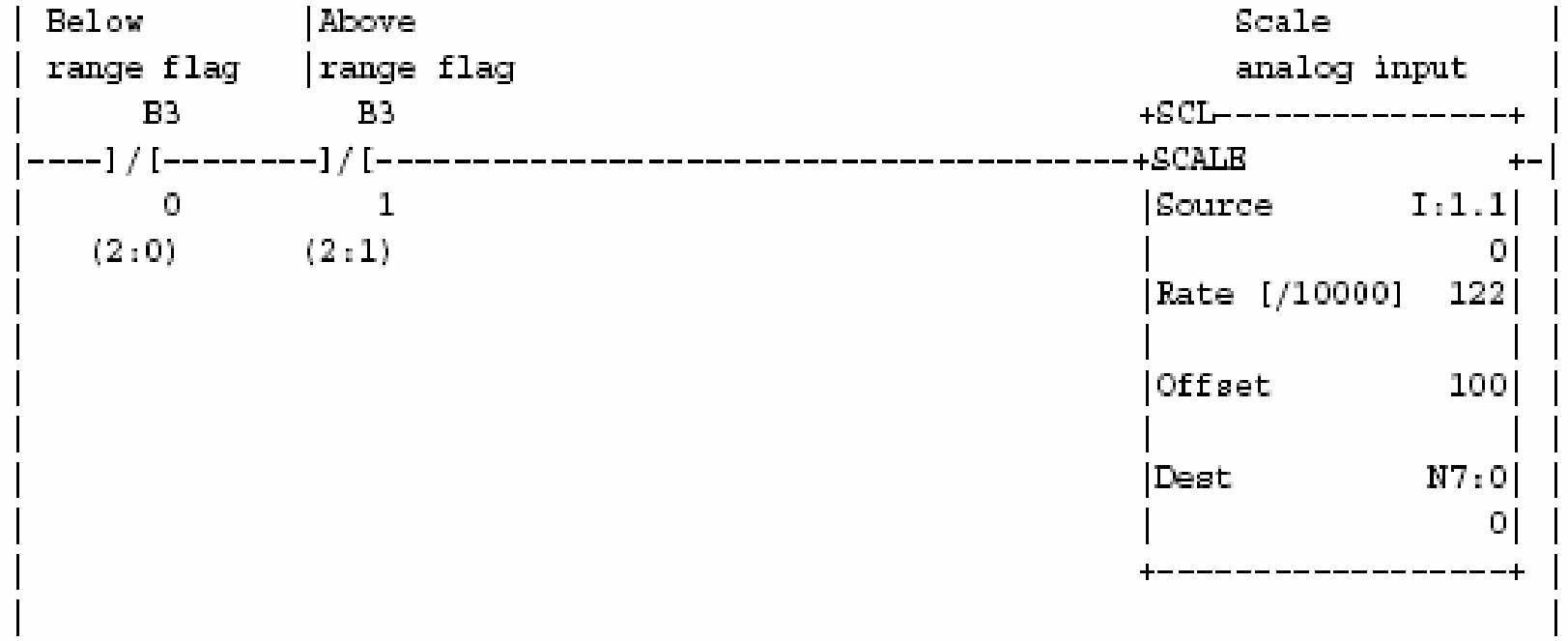
Rung 2:3
Turn on the above range alarm output when the analog input is above the acceptible range.



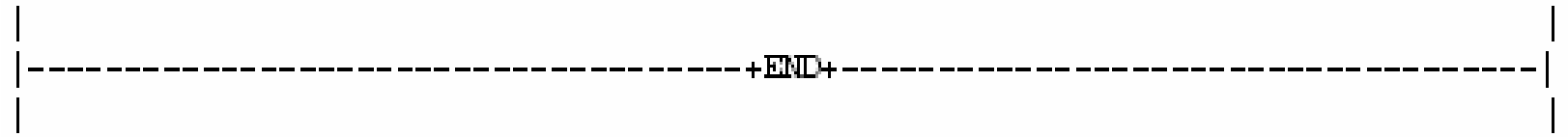
PROGRAMMING EXAMPLE

Rung 2:4

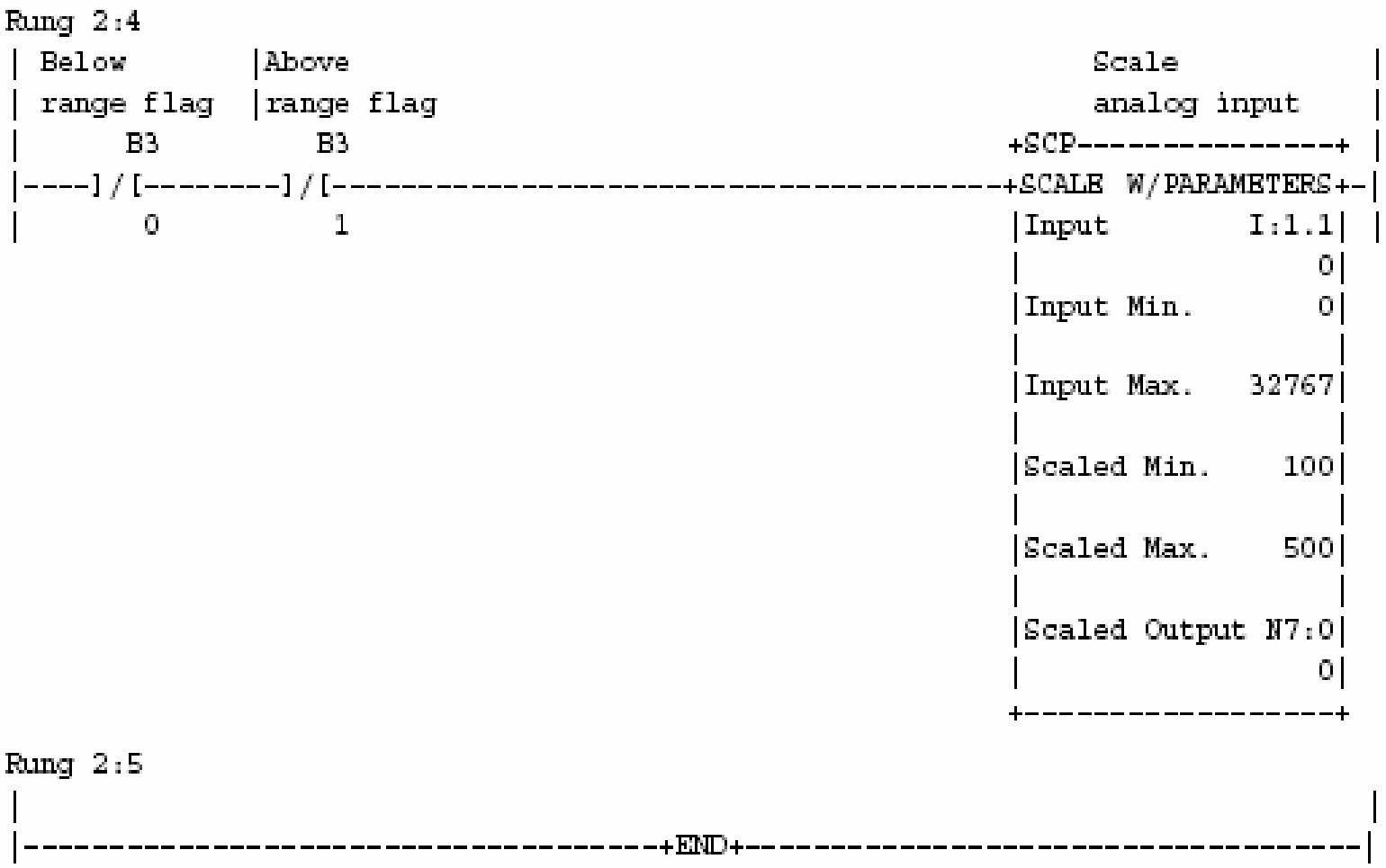
Scale analog input value and process the result only when it is within acceptible range.



Rung 2:5



PROGRAMMING EXAMPLE

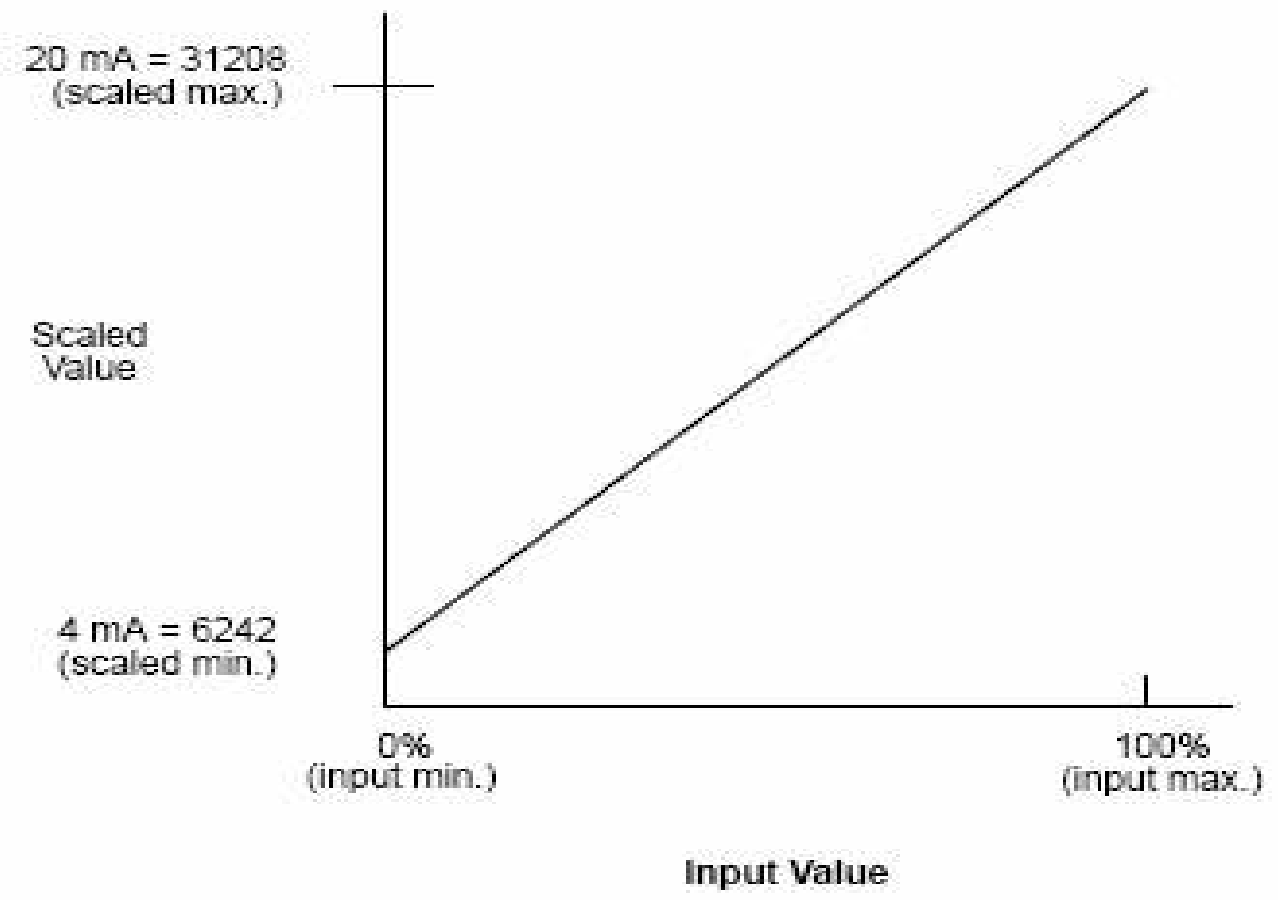


PROGRAMMING EXAMPLE (AO)

Assumption:

- The AO module is located in slot 2 of PLC system.
- An actuator from a flow control valve is wired to output channel 0.
- The actuator accepts a 4 mA to 20 mA signal proportional to 0 to
- 100% of the valve's opening. For this example the signal
- actuator can not receive a signal out of the 4 mA to 20 mA range.
- The percentage of valve opening is manually input to the PLC

PROGRAMMING EXAMPLE (AO)



Scaling Graph

PROGRAMMING EXAMPLE (AO)

Scaled value = (input value x slope) + offset

Slope = (scaled max. – scaled min.) / (input max. – input min.)

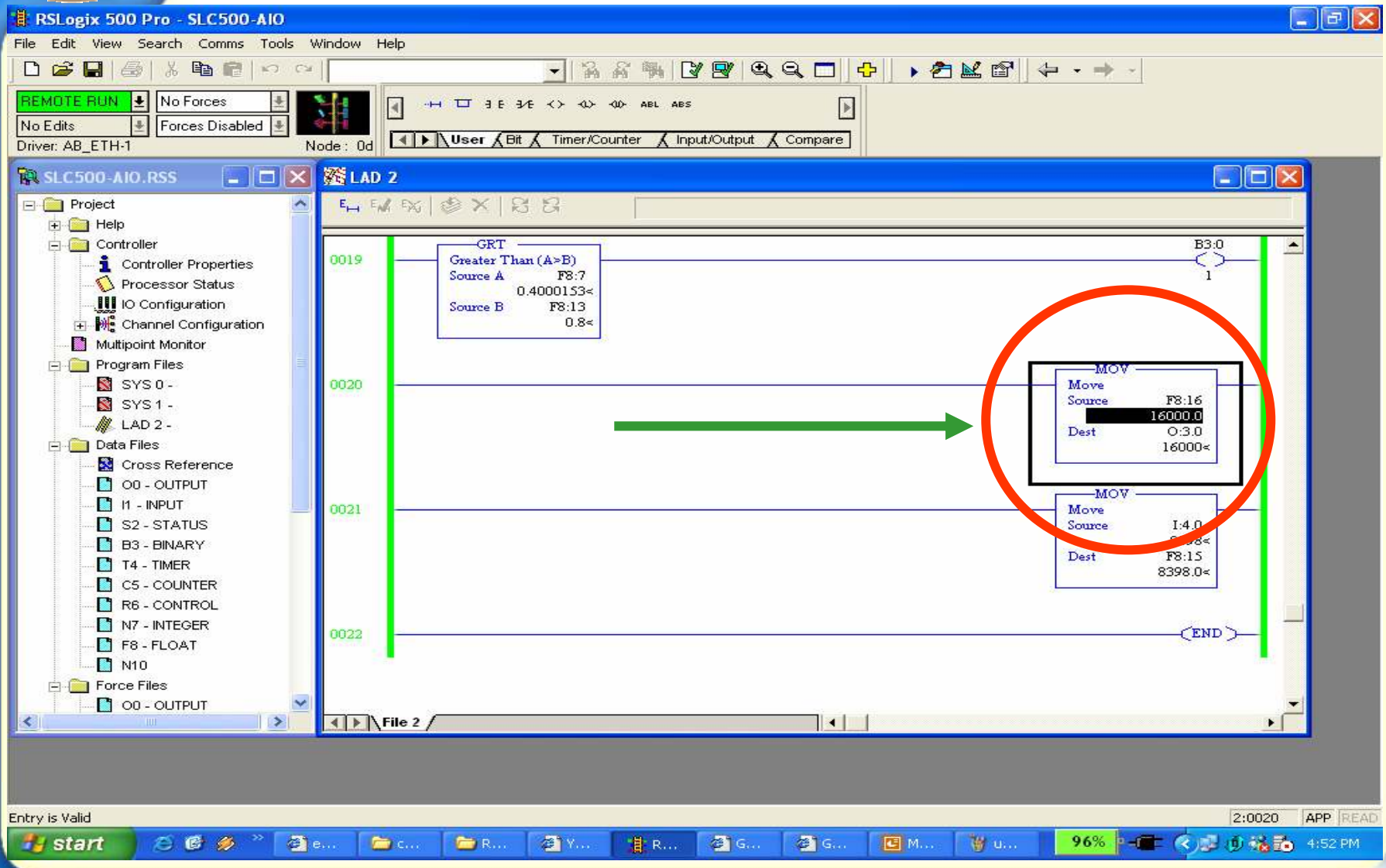
(31,208 – 6,242) / (100–0) = 24,966/100

Offset = scaled min. – (input min. x slope)

6,242 – (0 x (24,966/100)) = 6,242

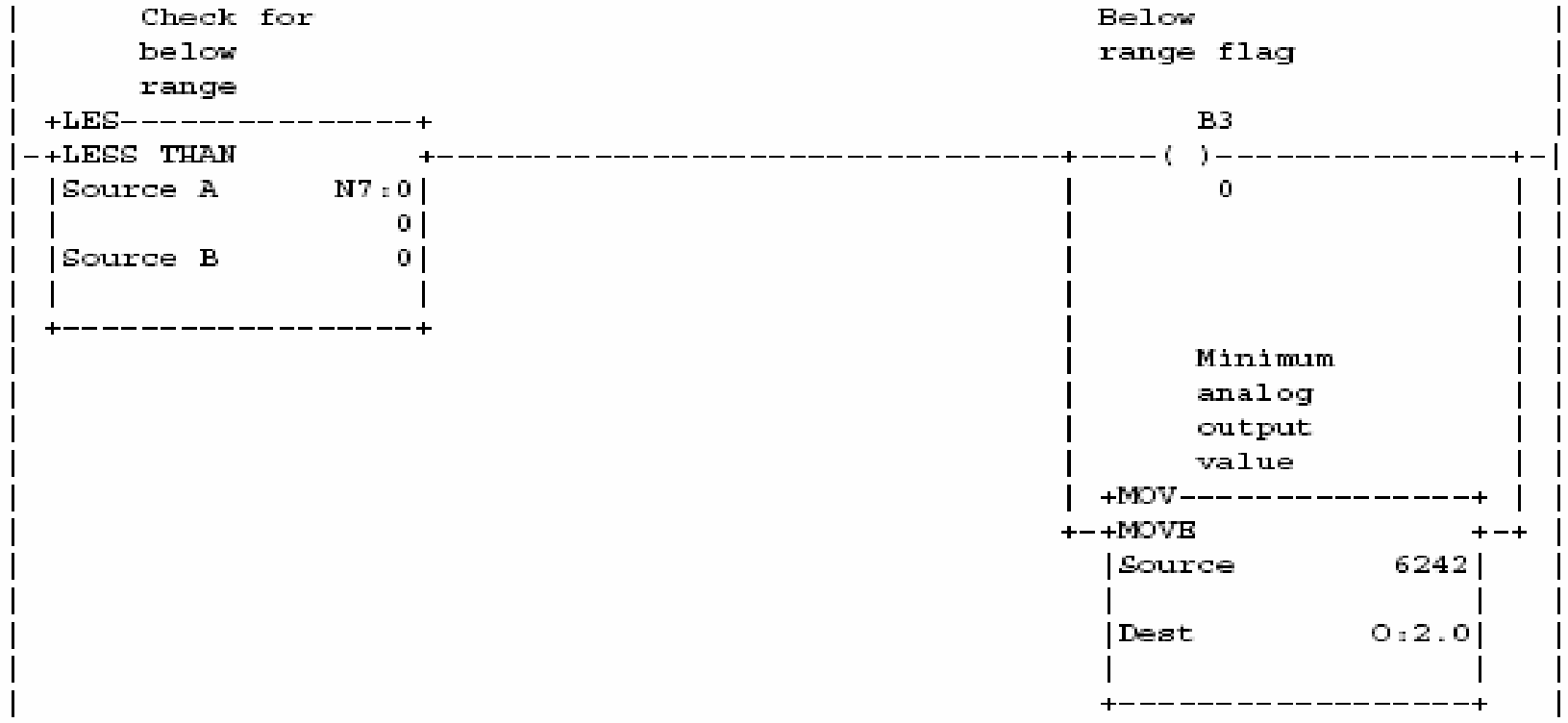
Scaled value = (input value x (24,966/100)) + 6,242

ANALOG OUTPUT COMMAND



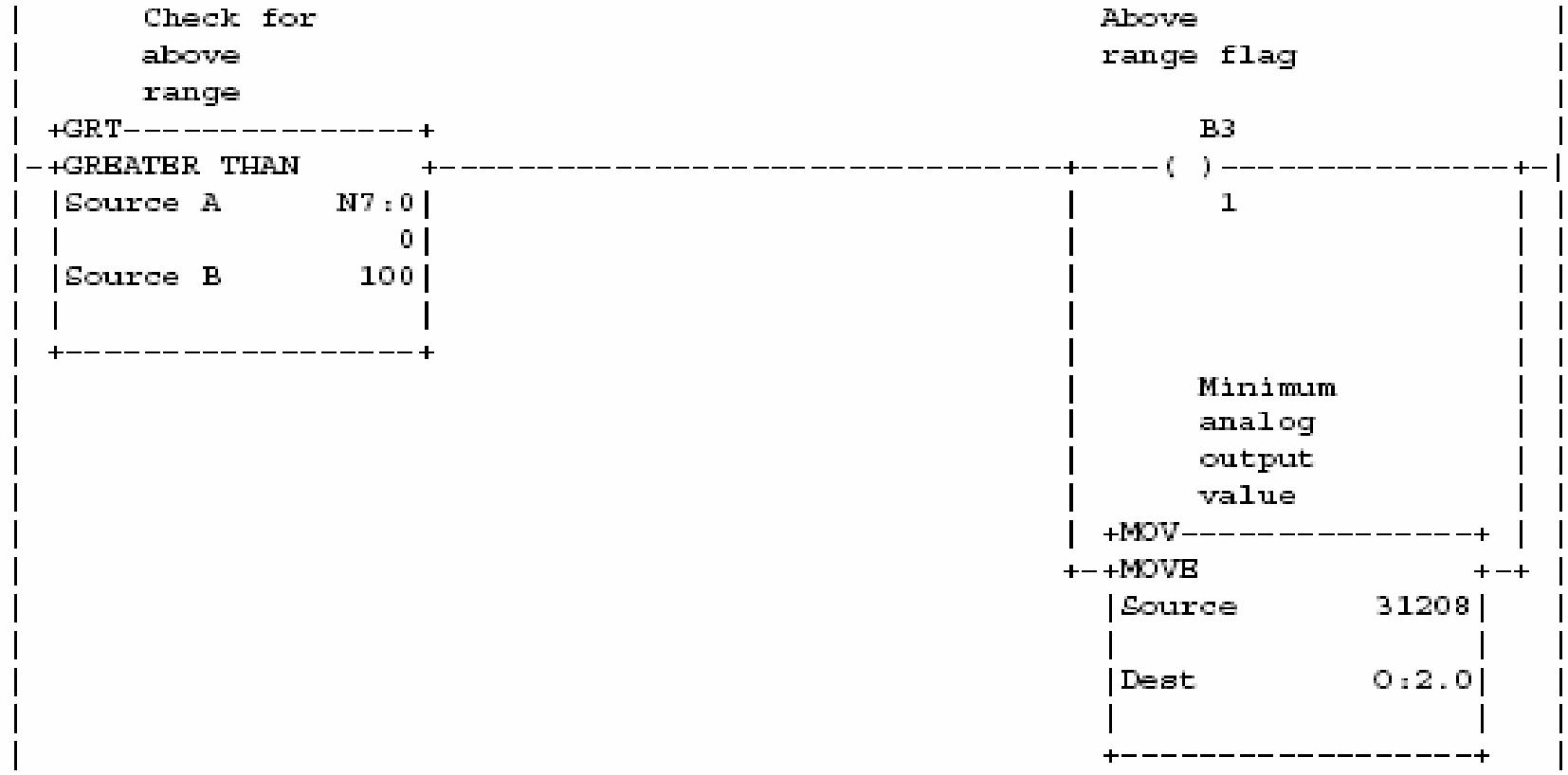
PROGRAMMING EXAMPLE (AO)

Rung 2:0
N7:0 contains percentage of valve open. If this value is less than 0, move the minimum value to the analog output (6242 decimal = 4 mA at the analog output).

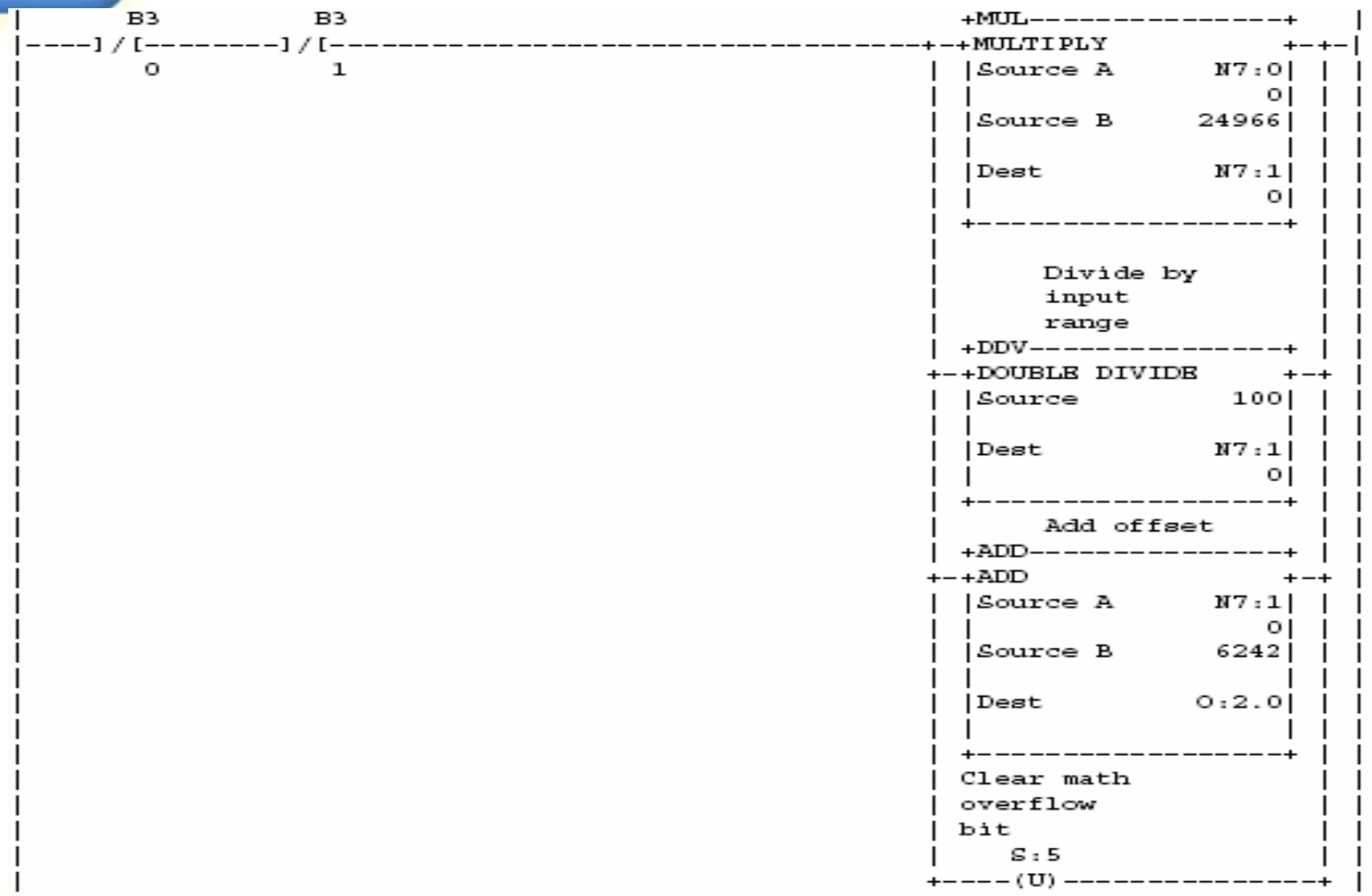


PROGRAMMING EXAMPLE (AO)

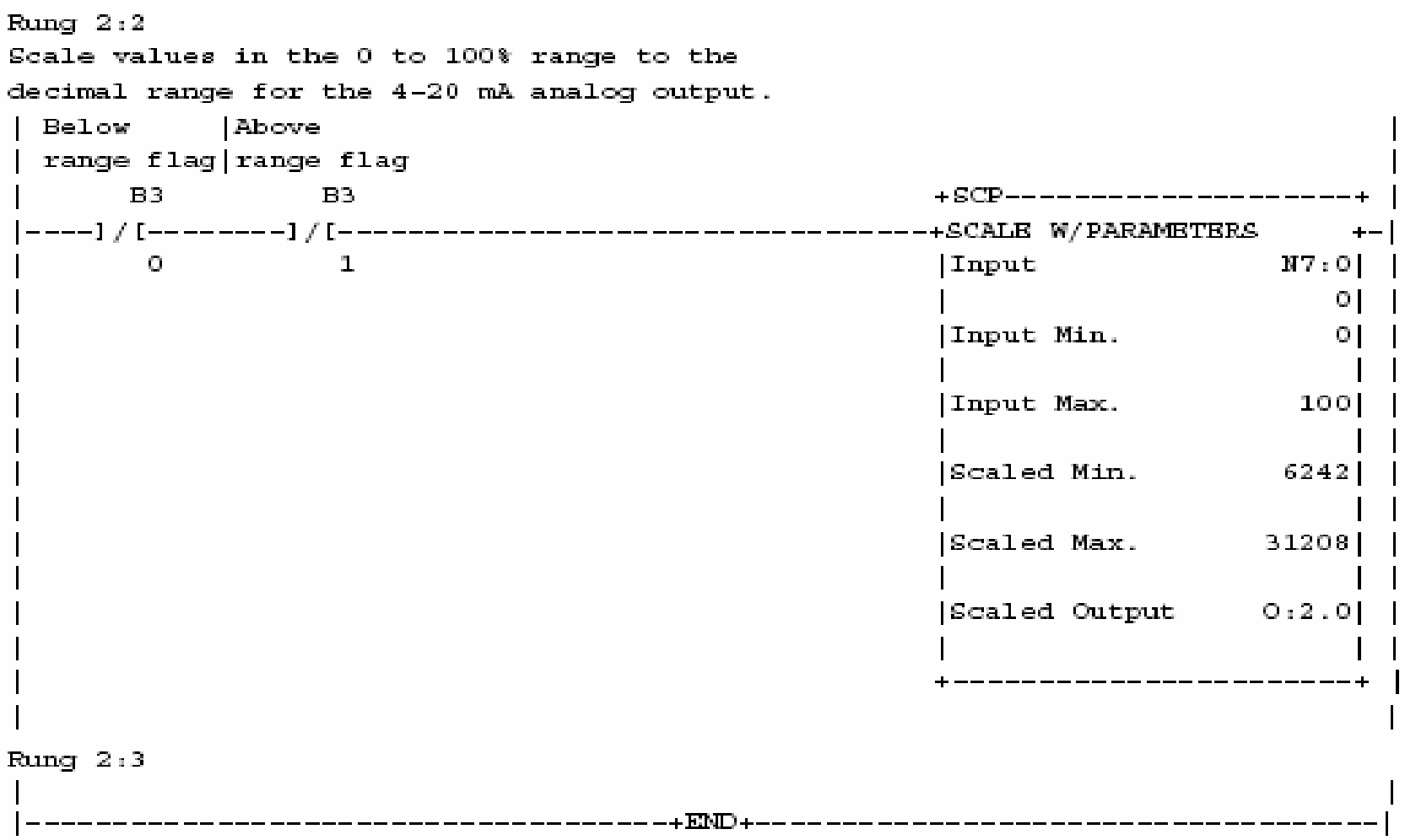
Rung 2:1
N7:0 contains percentage of valve open. If this value is greater than 100, move the maximum value to the analog output (31208 decimal = 20 mA at the analog output).



PROGRAMMING EXAMPLE (AO)



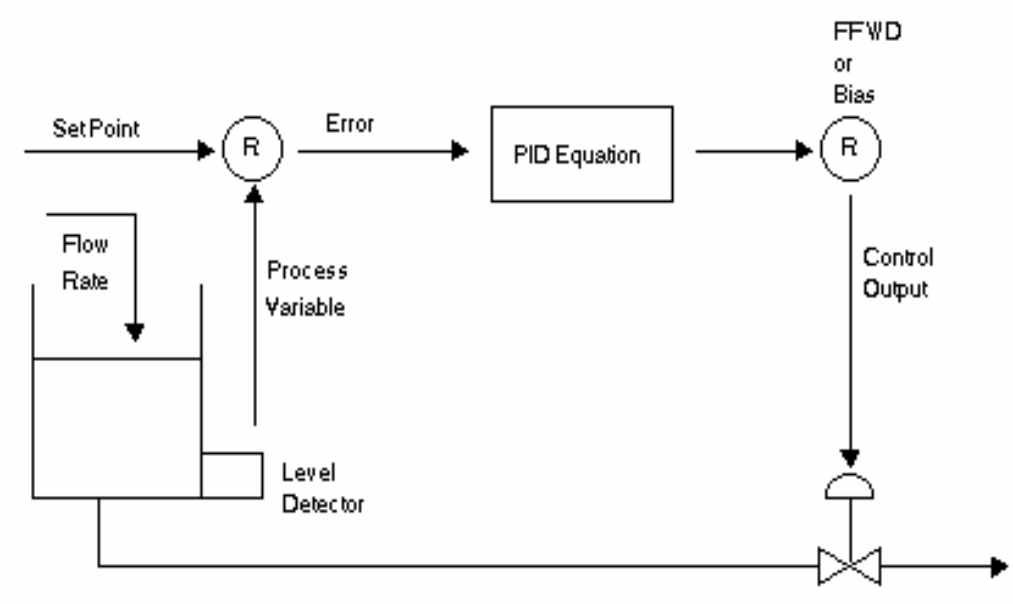
PROGRAMMING EXAMPLE (AO)

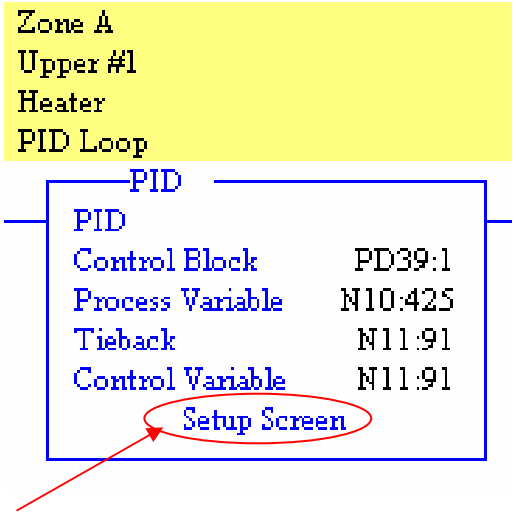


Process Control PID

- PID instruction controls a PID loop with inputs from an analog input module and an output to an analog output module.
- For temperature control, you can convert the analog output to a time proportioning on/off output for driving a heating or cooling unit.

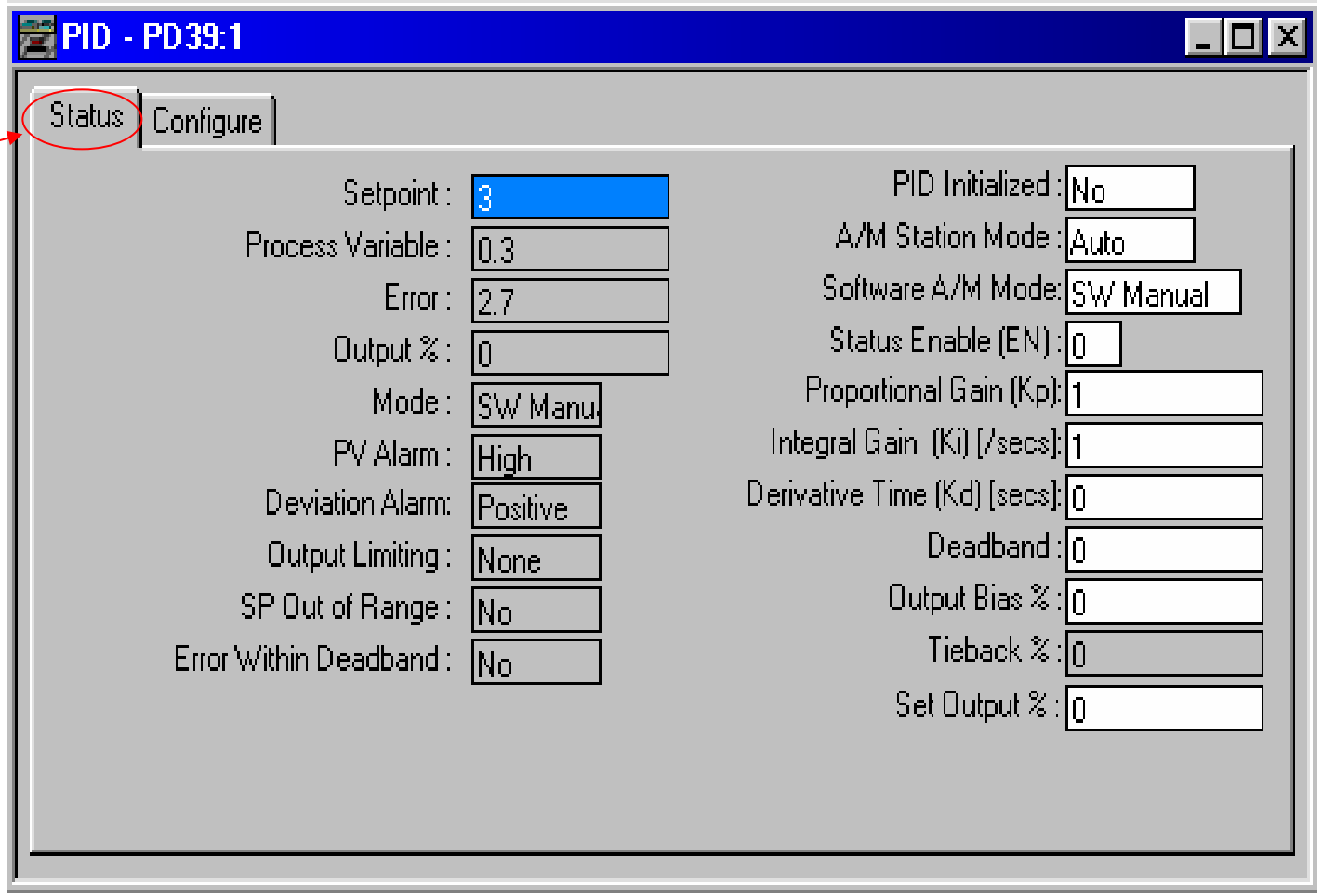
Process Control PID



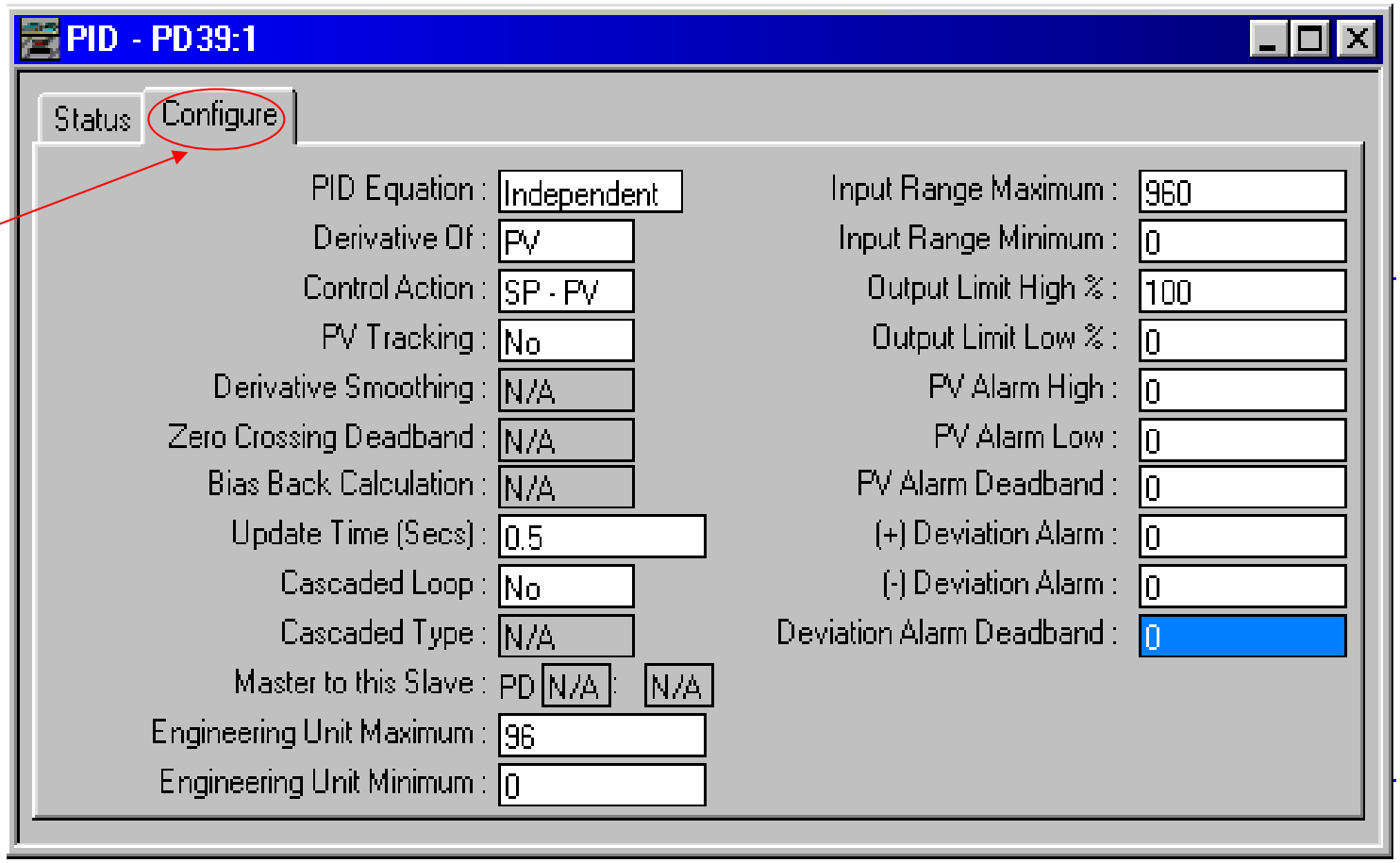


Example system use Control Process Automation PID.

Process Control PID



Process Control PID



Process Control PID

Parameter	Address Mnemonic	Description :
Setpoint	.SP	Enter a floating-point number in the same engineering units that are on the PID Configuration screen. Valid range is -3.4 E^{+38} to +3.4 E^{+38}.
Process Variable	.PV	Displays data from the analog input module that the instruction scales to the same engineering units that you selected for the setpoint.
Error	.ERR	Displays one of the following error values : Reverse acting : Error = PV-SP Direct acting : Error = SP-PV
Output %	.OUT	Displays the PID algorithm control output value (0-100%).
Mode	.MO .MO=0 .MO=1 .SWM=1	Displays operating mode : AUTO – automatic PID control MANUAL – control from a manual control station SW MANUAL – simulated manual control from the data monitor or ladder program

Process Control PID

Parameter	Address Mnemonic	Description :
PV Alarm	.PVHA=1 .PVLA=1	Displays whether the PV is within or exceeds the high or low alarm limits you selected on the PID Configuration screen. Displays one of the following : NONE – PV within alarm limits HIGH – PV exceeds high alarm limit (used with deadband) LOW – PV exceeds low alarm limit (used with deadband)
Deviation Alarm	.DVPA=1 .DVNA=1	Displays whether the error is within or exceeds the high or low deviation alarms you selected on the PID Configuration screen. Displays one of the following : NONE – error within deviation alarm limits POSITIVE – error exceeds high alarm (used with deadband) NEGATIVE – error exceeds low alarm (used with deadband)
Output Limiting	.OLH=1 .OLL=1	Displays wheter or not the instruction clamps the output at the high and low limiting values (.MAXO and .MINO) you selected on the PID Configuration screen. Displays one of the following : NONE – output not clamped HIGH – output clamped at the high end (.MAXO)

Process Control PID

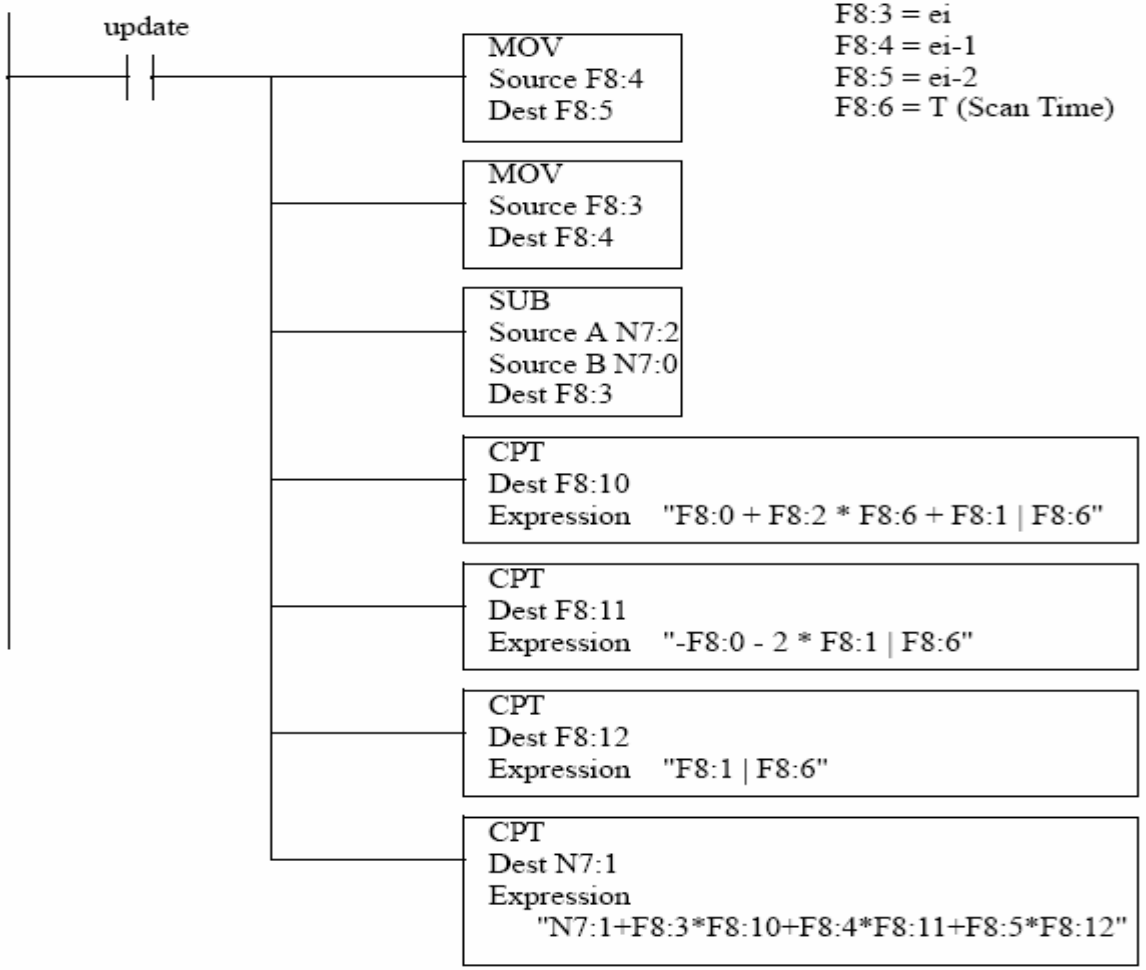
Parameter	Address Mnemonic	Description :
		LOW – output clamped at the low end (.MINO) The PID algorithm has a anti-reset-windup feature that prevents the integral term from becoming too large when the output reaches the high or low alarm limits. If the limits are reached, the algorithm stops calculating the integral term until the output comes back into range.
SP Out of Range	.SPOR=0 .SPOR=1	Displays whether or not the setpoint is out of the range of engineering units you selected on the PID Configuration screen. Displays one of the following : NO – SP within range YES – SP out of range Note : A major processor fault occurs if the SP is out of range when the instruction is the first enabled.

Process Control PID

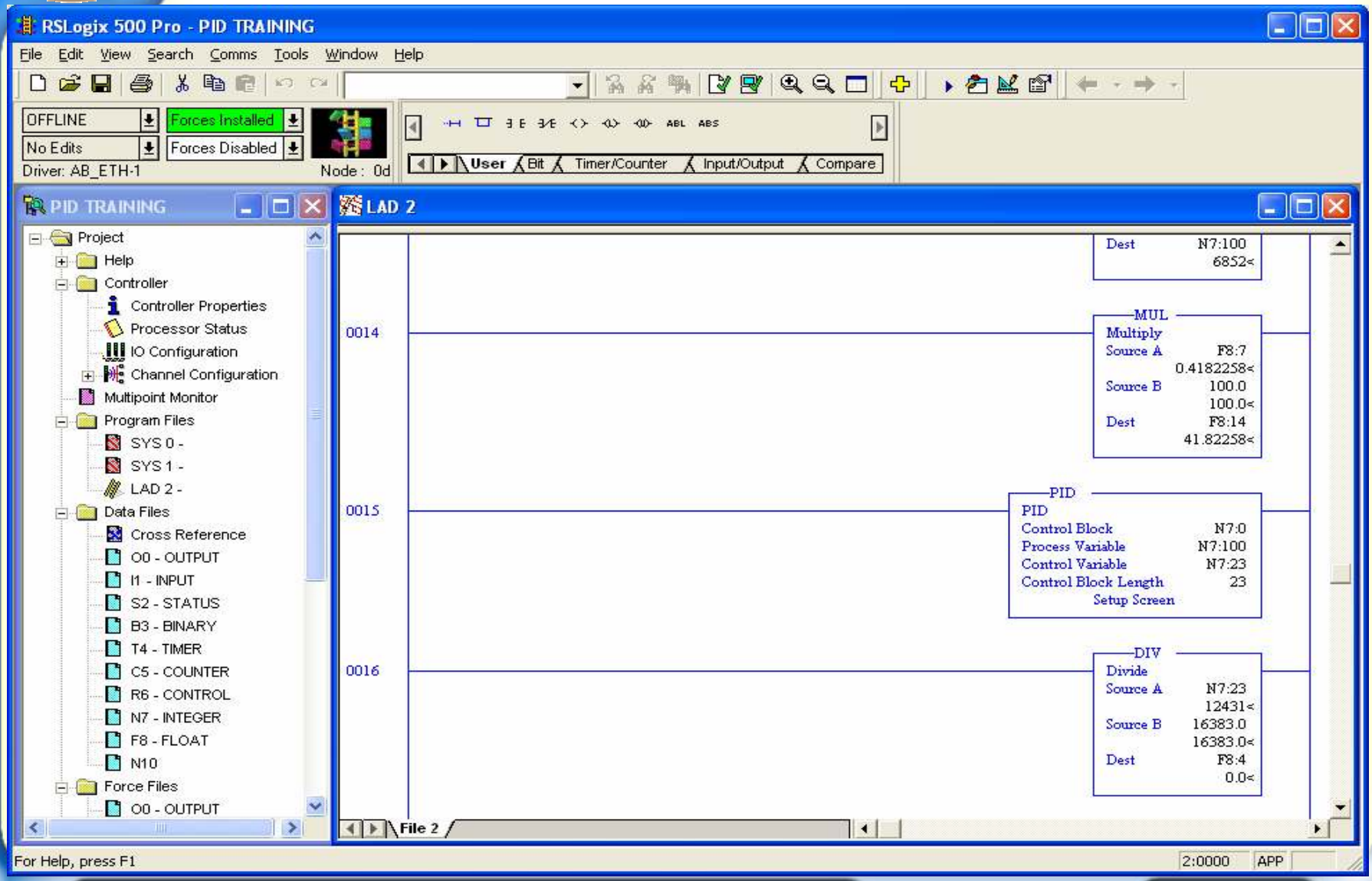
Assume the values:

N7:0 = Analog input value
N7:1 = Analog output value
N7:2 = Setpoint

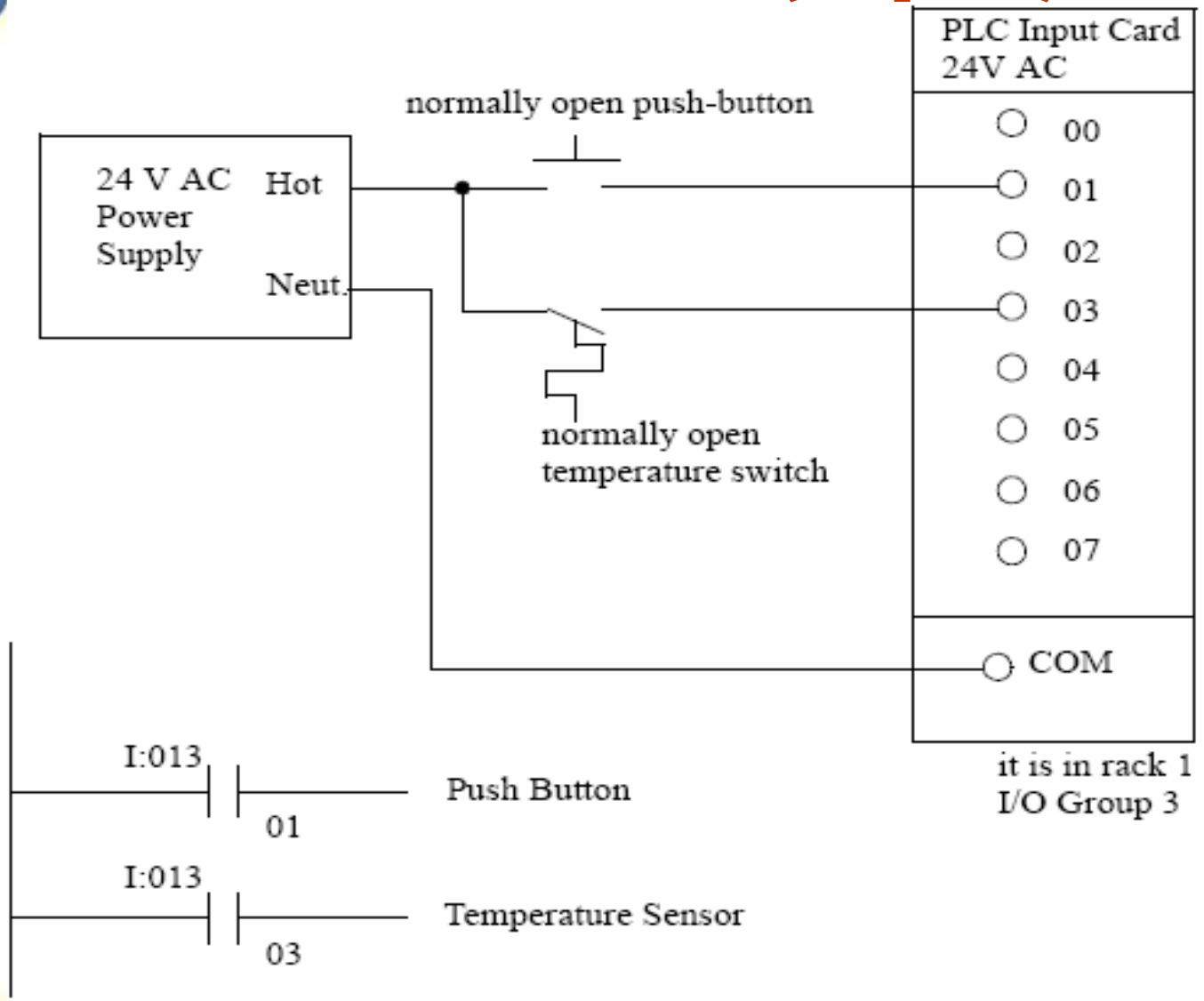
F8:0 = Kp
F8:1 = Kd
F8:2 = Ki
F8:3 = ei
F8:4 = ei-1
F8:5 = ei-2
F8:6 = T (Scan Time)



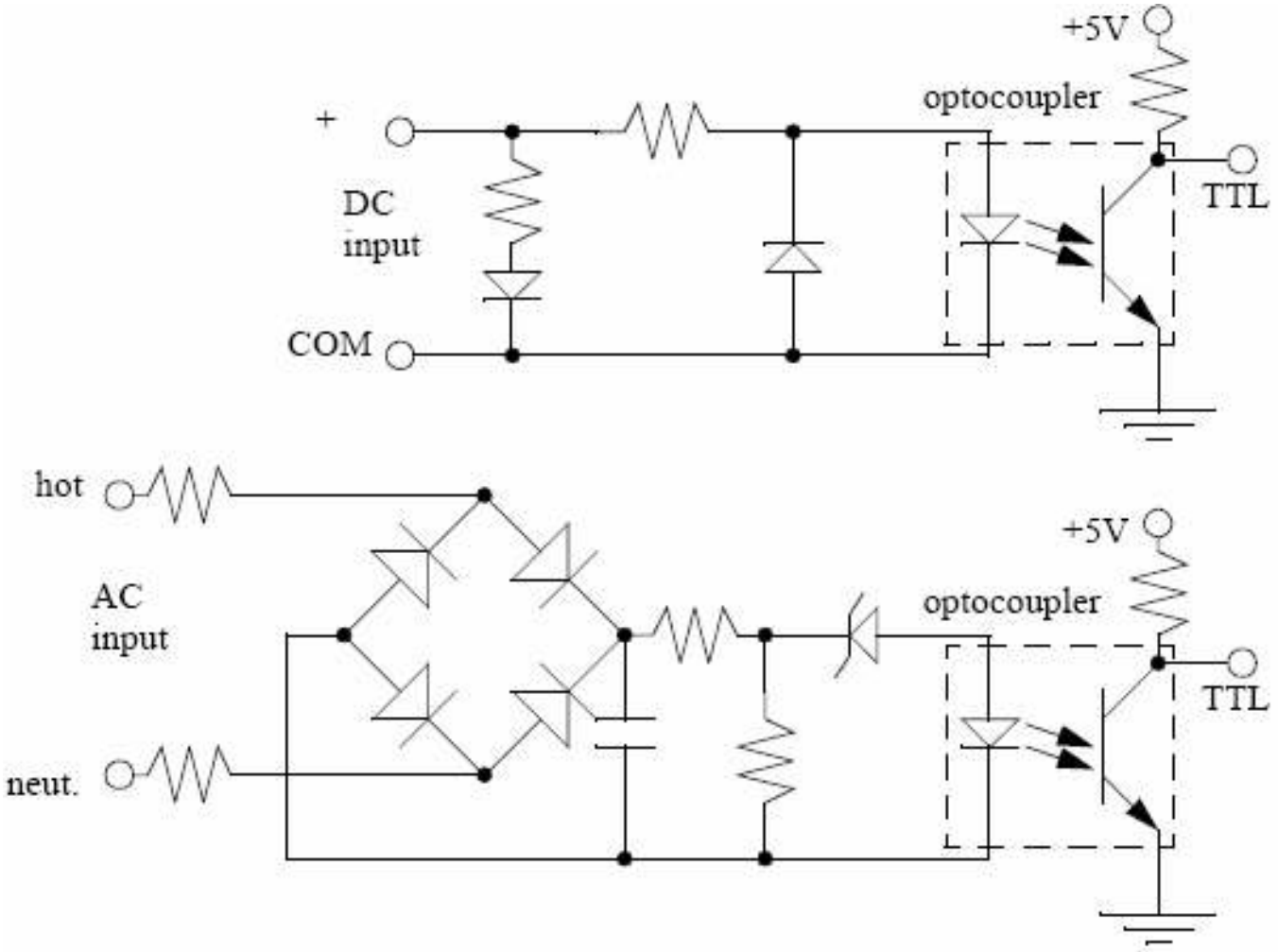
Process Control PID



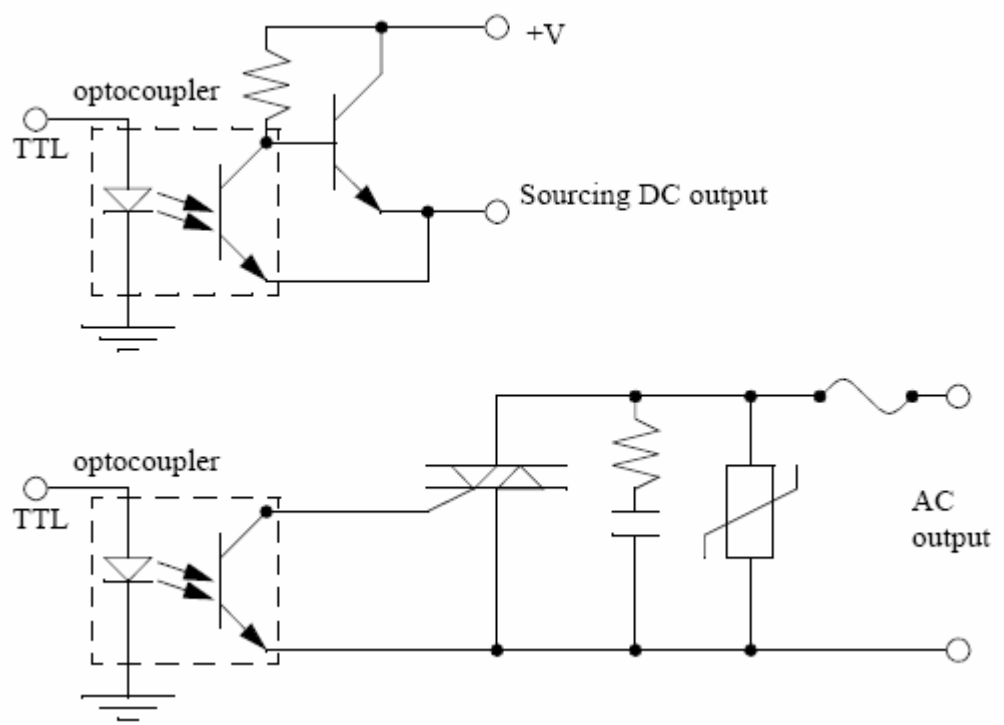
PLC WIRING (Input)

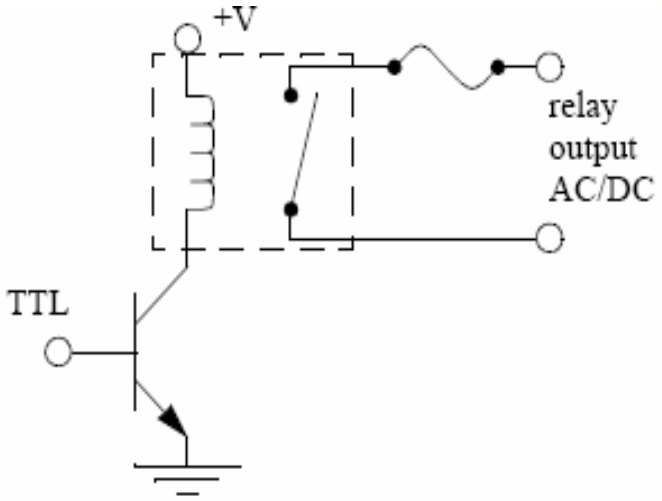


PLC WIRING (Input)

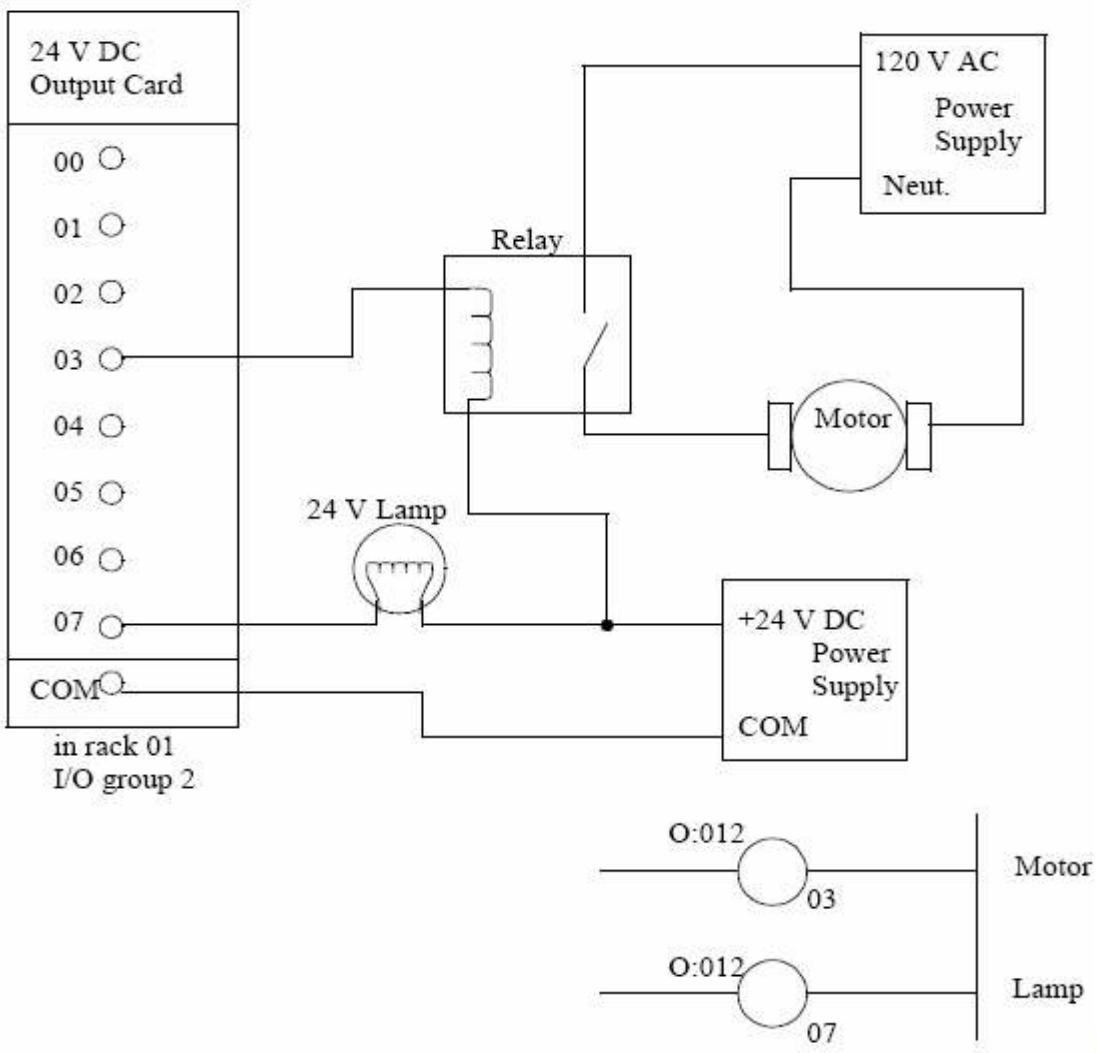


PLC WIRING (OUTPUT)

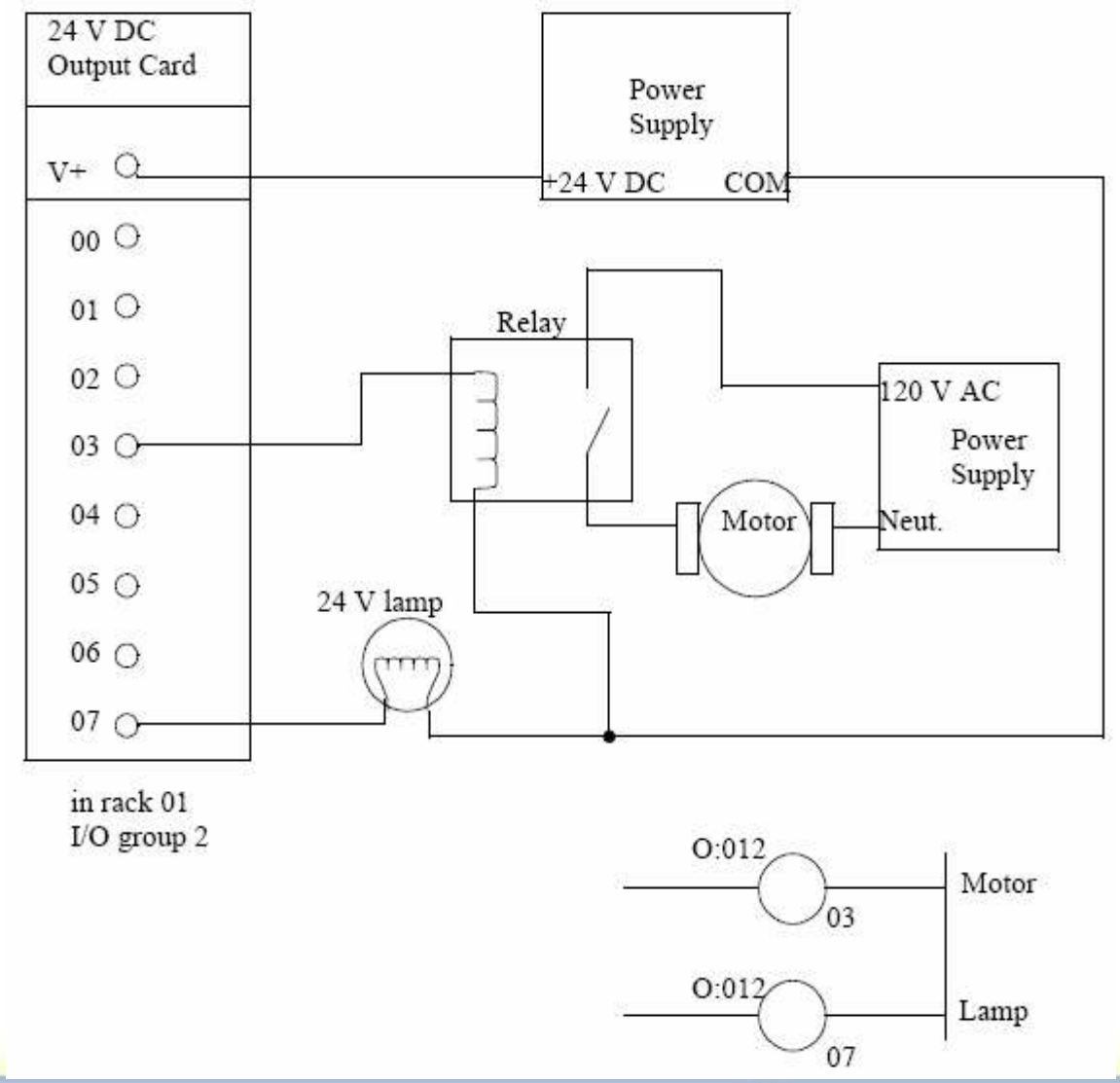




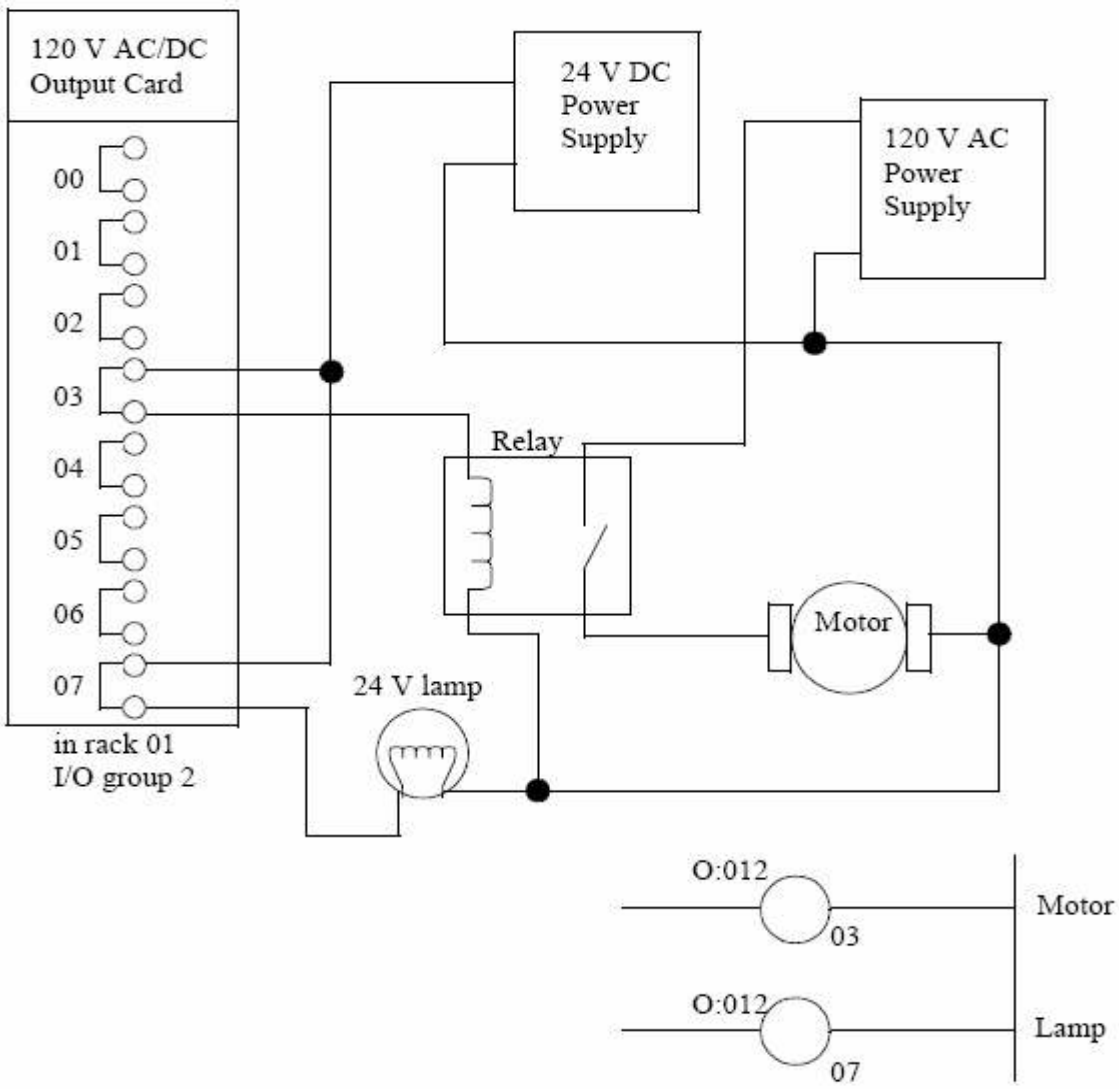
PLC WIRING (OUTPUT)



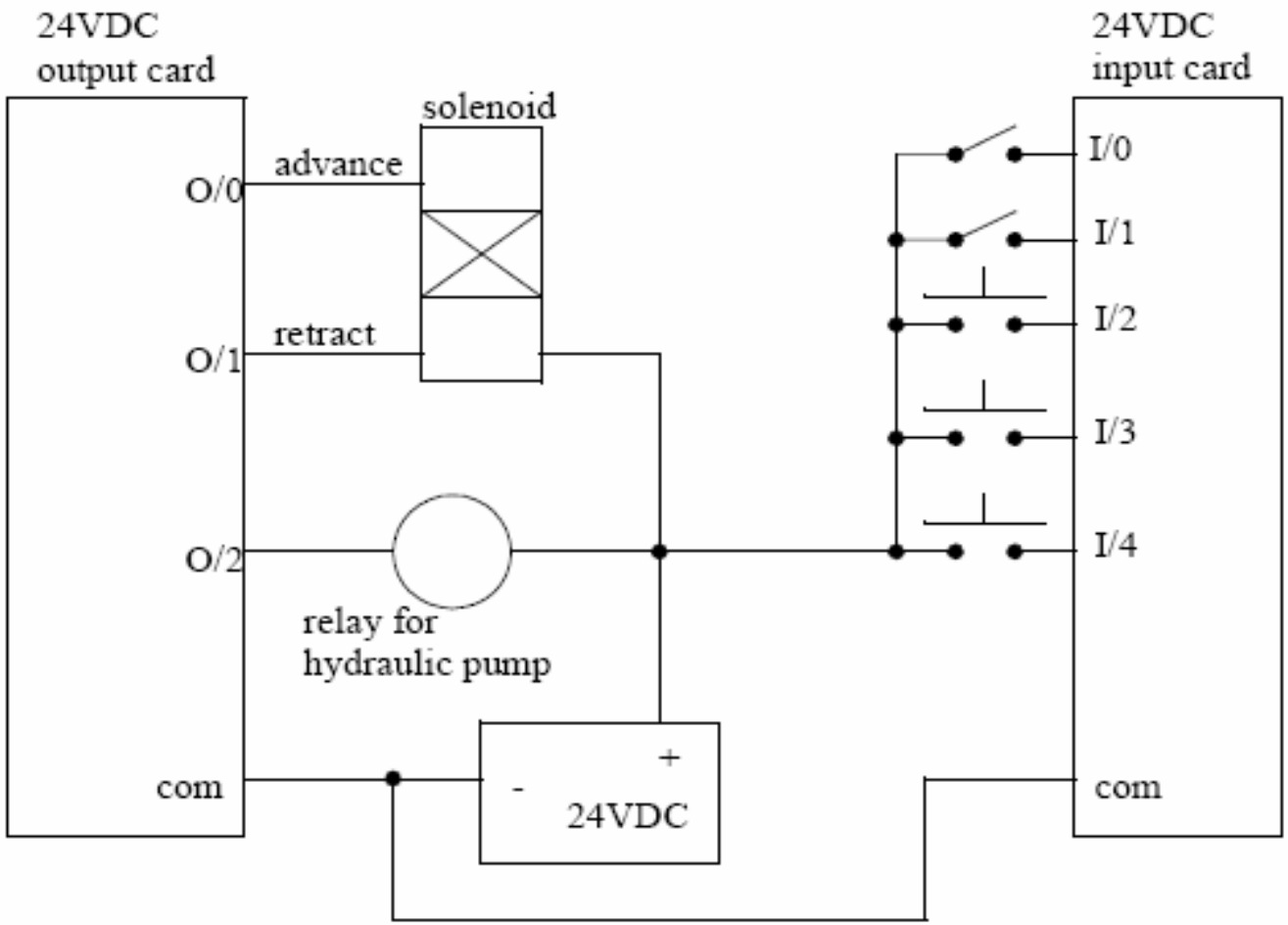
PLC WIRING (OUTPUT)



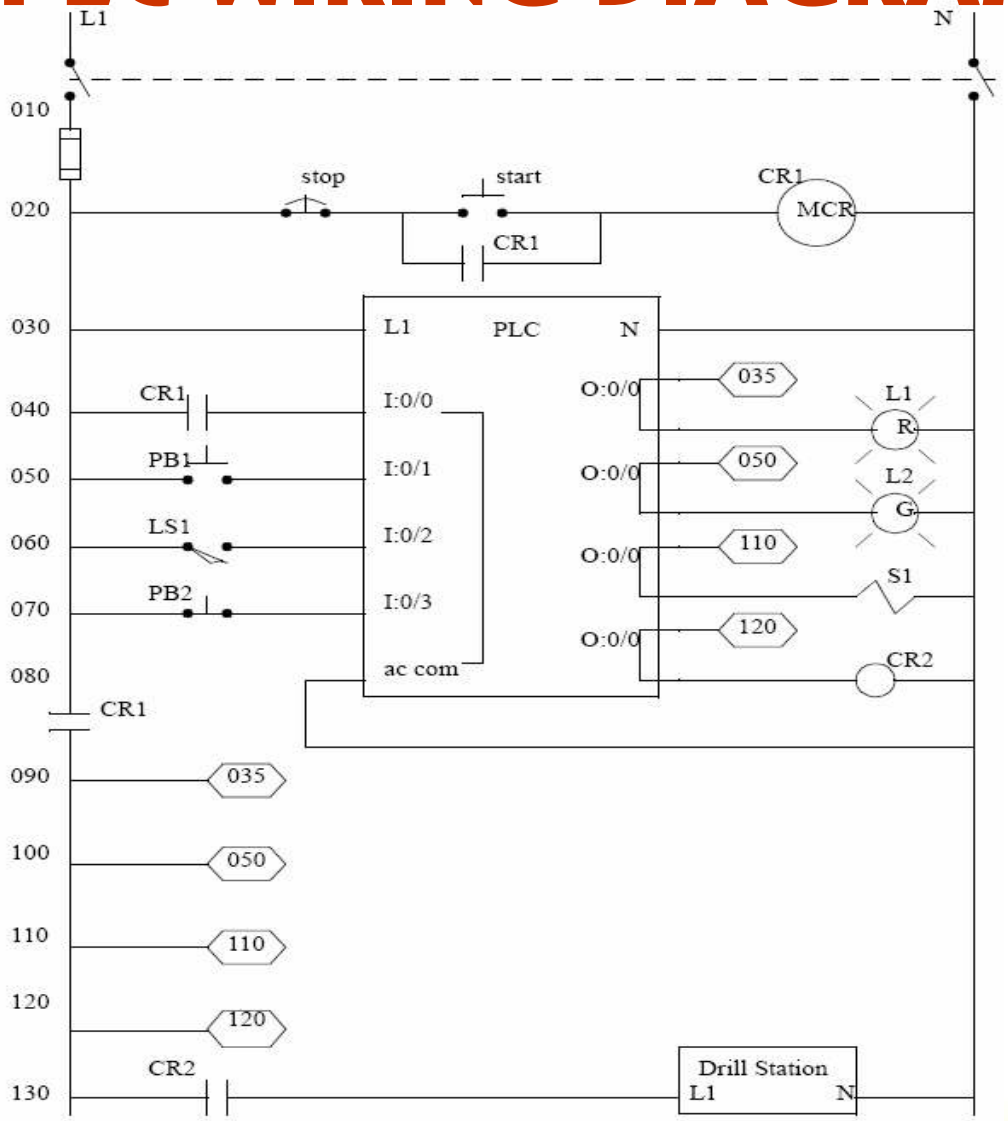
PLC WIRING (OUTPUT)



PLC WIRING (OUTPUT)



PLC WIRING DIAGRAM



GROUND LOOP ELIMINATION

