User Guide

CleverAx

Single-Axis Motion Controller

Safety Information

Persons supervising and performing the electrical installation or maintenance of a Drive and/or its external Option Unit must be suitably qualified and competent in these duties. They should be given the opportunity to study and if necessary to discuss this User Guide before work is started.

The voltages present in the Drive and external Option Units are capable of inflicting a severe electric shock and may be lethal. The Stop function of the Drive does not remove dangerous voltages from the terminals of the Drive and external Option Unit. AC supplies should be removed before any servicing work is performed.

The installation instructions should be adhered to. Any questions or doubt should be referred to the supplier of the equipment. It is the responsibility of the owner or user to ensure that the installation of the Drive and external Option Unit, and the way in which they are operated and maintained complies with the requirements of the Health and Safety at Work Act in the United Kingdom and applicable legislation and regulations and codes of practice in the country in which the equipment is used.

The Drive software may incorporate an optional Auto-start facility. In order to prevent the risk of injury to personnel working on or near the motor or its driven equipment and to prevent potential damage to equipment, users and operators, all necessary precautions must be taken if operating the Drive in this mode.

The Stop and Start inputs of the Drive should not be relied upon to ensure safety of personnel. If a safety hazard could exist from unexpected starting of the Drive, an interlock that electrically isolates the Drive from the AC supply should be installed to prevent the motor being inadvertently started.

General Information

The manufacturer accepts no liability for any consequences resulting from inappropriate, negligent or incorrect installation or adjustment of the optional operating parameters of the equipment or from mismatching the Drive with the motor.

The contents of this User Guide are believed to be correct at the time of printing. In the interests of a commitment to a policy of continuous development and improvement, the manufacturer reserves the right to change the specification of the product or its performance, or the contents of the User Guide, without notice.

Author: U79
Part Number: 0400-0003
Issue Code: CXNU1
Issue Date: November 1995

Contents

1 Description

1.1 Main features

CleverAx is a programmable, closed-loop position controller having the following control features:

- Position control with trapezoidal profiling, or acceleration and deceleration S-ramps
- Digital lock with variable ratio and trigger programming
- Electronic cam
- Use of ladder diagram (PLC type) programming language
- Eurocard format
- 24VDC supply

The control system consists of the following three modules:

- PLC module
- Motion module
- Serial Link module

Software parameters are used to configure the control system.

The internal interface meets the requirements of standard applications.

1.2 Inputs and outputs

- Differential 14-bit analog output for the Drive speed reference.
- Differential analog general-purpose 14-bit outputs.
- Two differential analog general-purpose 10-bit inputs.
- 15 general purpose digital inputs.
- Three high-speed digital inputs.
- Seven digital general purpose outputs.
- Two encoder interfaces (Master and Slave).
- Two standard serial ports (RS232C and RS485). The RS232 serial port allows the board to be connected to a PC for creating programs and setting parameters.

1.3 Software tools

- Terminal emulator software
- Full-screen editor
- Compiler
- Debugger
- Download function to download the program to the CleverAx card
- Save command to store the program on a PC disk
- Load command to load the desired program from a PC disk

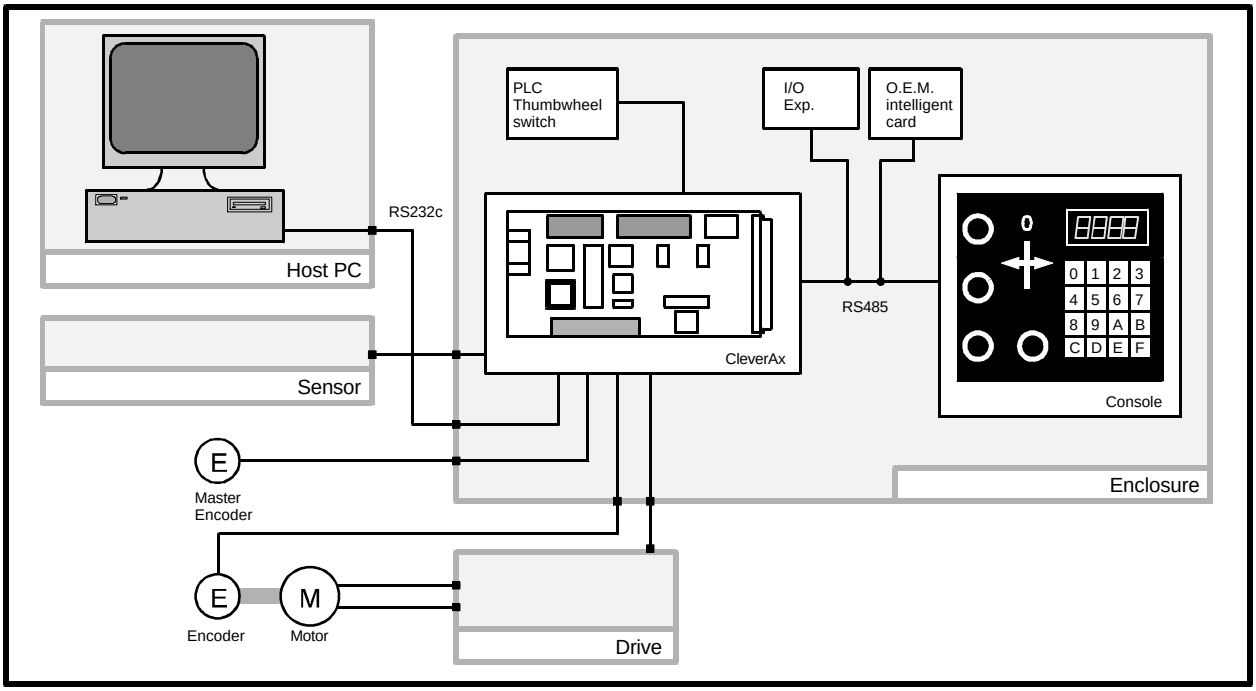


Figure 1–1 Control System using the CleverAx card

1.4 Control functions

Position loop

Position loop control can be carried out in either of the following ways:

- Proportional and integral with feed-forward control
- Proportional, integral and differential control

The card controls the Drive using the torque or speed input of the Drive.

Position control

The axis position is controlled using trapezoidal or S-ramp speed profiles. The acceleration and deceleration ramps are programmable.

The following functions are avalaible:

- Homing
- Position counter freeze
- Compare axis with 4 stored positions
- End profiling flag
- Servo error
- Electric hand-wheel

Digital Lock

Digital Lock allows the Drive to be synchronized in speed and phase with the master axis, using a master encoder as reference. A reduction ratio is programmable between −2 and +2, with a resolution of 9 decimal places.

The controlled axis can be locked by the following:

- A digital input signal
- The master encoder index
- The master encoder index masked from a digital input

Other configurations can be created using a PLC program.

When the slave axis is locked, it is possible to add a position or speed offset.

Electronic cam

Using a look-up table, it is possible to define the position of the controlled axis in relation to the position of the master axis. The electronic cam is enabled by a START command and stops when the last position is reached in the table or when a STOP command is given.

Different cam profiles can be programmed up to a limit defined by the available memory.

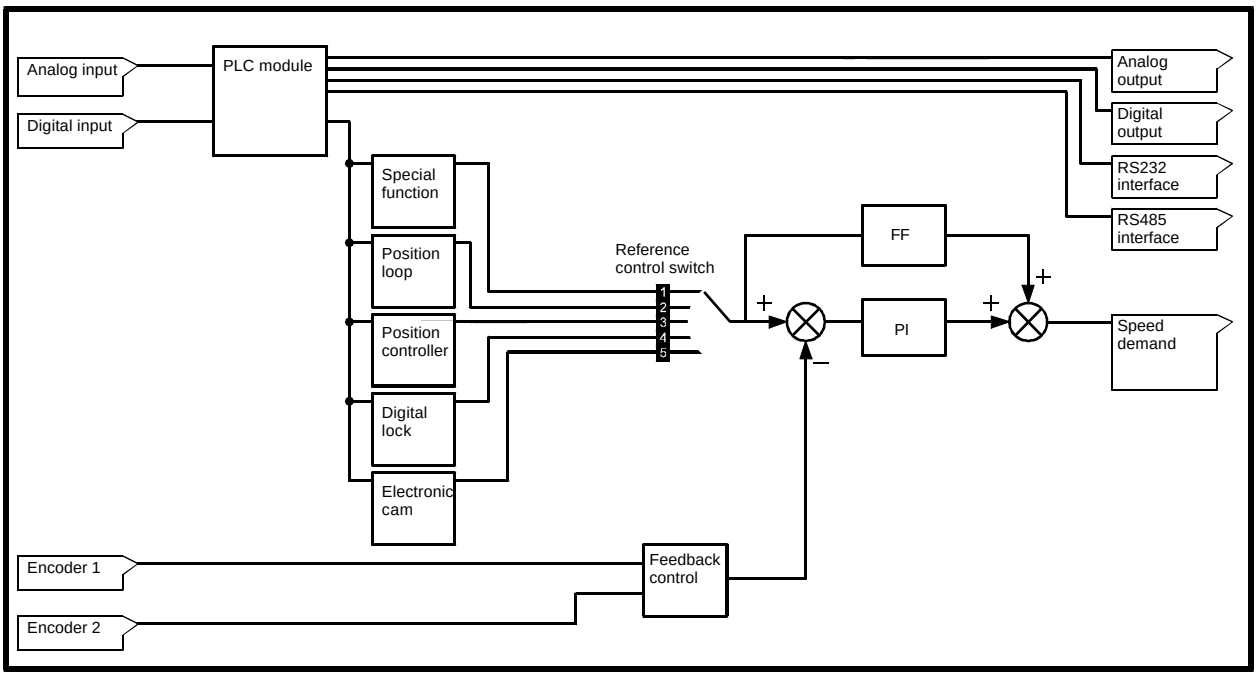


Figure 1–2 CleverAx Control system

2 Data

Supply voltage

24VDC ±25%

Maximum current demand

CleverAx card:	250 mA
Main encoder:	250 mA
Auxiliary encoder:	250 mA
Logic output:	100 mA (for each output)
Analog output:	2 mA

Operating temperature

0°C to +45°C (32°F to 113°F)

Speed reference output

±10V differential (14-bit equivalent)

Analog output

±10V differential (14-bit equivalent)
Impedance: < 100Ω

Analog input

±10V differential (10-bit equivalent)

Logic output

+24V, 100mA

Logic input impedance

15kΩ

Range of analog inputs

±10V

Drive enable output at low level

0V

Drive enable output at high level

24V

Serial interface

Standard RS232C and RS485

Max encoder frequency

120kHz (before internal multiplication by 4)

Encoder interface

Standard RS422

High speed counter (P71 and P72)

Max input frequency: 5kHz

Protection

Short-circuit protection of logic outputs
Short-circuit of auxiliary voltage outputs

Dimensions

Single Eurocard

CPU

Processor:	80C166
Clock frequency:	40MHz
RAM:	8kB
EPROM:	32kB
EEPROM:	8kB

Watchdog

A Fault output signal produced when processor error occurs

Memory map

Address	Description
H0000 to H7FFF	Firmware EPROM
H8000 to H9FFF	Monitor (H9000 to H9FFF – expansion)
HA000 to HBFFF	RAM
HC000 to HF9FF	EEPROM
HFA00 to HFFFF	Microprocessor registers (HFA00 to HFDFF – system stack)

Control modules

Motion module
PLC module
Serial link

Each module uses a set of parameters. The parameters are organized in logical areas as shown in the following table.

Parameter map

Parameters	Description
P1 to P49	Motion parameters
P50 to P73	PLC special register and PLC flags
P74 to P81	Serial link parameters
P84 to P128	User parameters

Parameter loading and saving

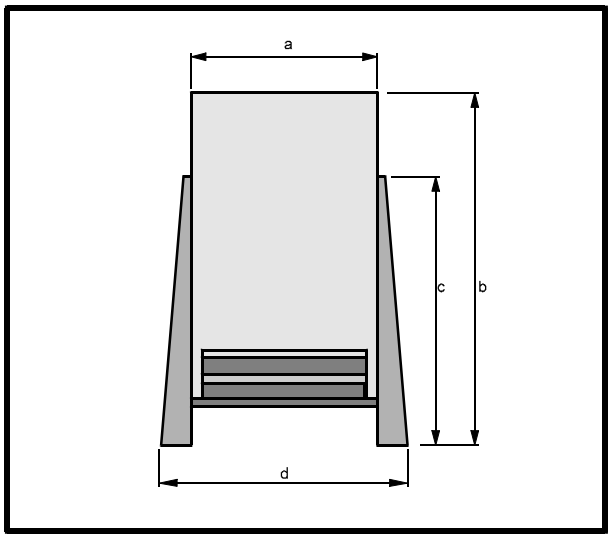
Parameters	Description	Load bit	Save bit
P11 to P30	Motion module parameters	Auto	P73.12
P32 to P49			
P73.10	PLC module flag parameters	Auto	P73.12
P73.11			
P74	Link module parameters	Auto	P73.12
P100 to P128	User parameters	P73.19	P73.18

PLC module

Function	Description
Constant scanning	The time between scanning cycles can be adjusted
Programming method	Relay logic
Type of control	Cyclical operation
Program memory	Internal EEPROM
Maximum number of instructions	999
Instruction type	High and low level instructions
Number of inputs	19
Number of outputs	8
Number of internal relays	224
Number of special internal relays	96 (**P31**, **P32** and **P73**)
Number of timers	6
Number of counters	3
Number of data registers	46 (from **P82** to **P128**)
Number of special registers	25 (all PLC and LINK parameters)
Number of Motion parameters	49
Number of index registers	1 (**P80**)
Number of labels	100

3 Mechanical Installation

Insert the CleverAx card in a slot of a standard 19" rack, or panel-mount the card using a standard card holder having screw terminals.



Dimensions		mm	in
Internal width	a	100	$3^{15}/_{16}$
Height	b	191	$7^{1}/_{2}$
Height of pillars	c	144	$5^{11}/_{16}$
External width	d	132	$5^{3}/_{16}$

Figure 3–1 Dimensions of the standard card holder

4 Electrical Installation

4.1 Connections

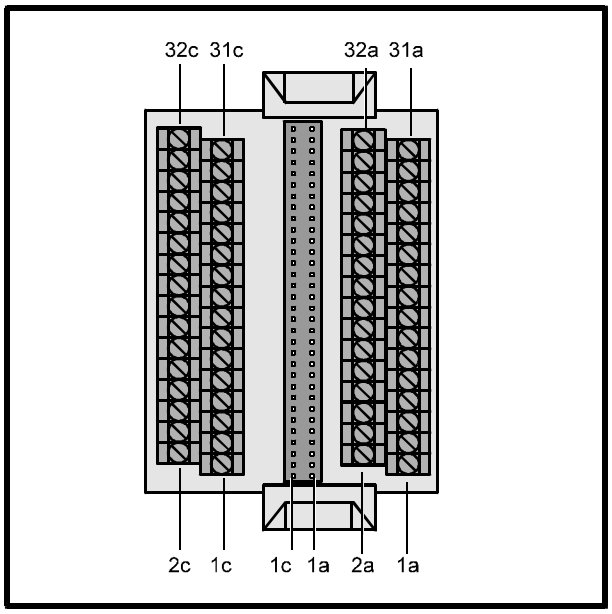


Figure 4–1 Locations of the connectors

Connector side A

Pin No	Function	Type	Description
1a	Speed reference (non inverting)	0	Speed reference output signal applied to the Drive
2a	Speed reference (inverting)	0	
3a	AUX analog output (non inverting)	0	Auxiliary analog output
4a	AUX analog output (inverting)	0	
5a	GND		Ground
6a	+A0	I	Analog input 0 (non inverting)
7a	–A0	I	Analog input 0 (inverting)
8a	+A1	I	Analog input 1 (non inverting)
9a	–A1	I	Analog input 1 (inverting)
10a	RLB		Line receiver load resistor
11a	RLA		
12a	RX	I	Serial communications line receiver
13a	$\overline{RX}$	I	Serial communications line receiver (inverting input)
14a	TX	0	Serial communications line transmitter
15a	$\overline{TX}$	0	Serial communications line transmitter (inverting output)
16a	+5V	0	Master encoder power supply
17a	GND		Ground
18a	+PhA2	I	Master encoder phase A
19a	–PhA2	I	
20a	+PhB2	I	Master encoder phase B
21a	–PhB2	I	
22a	+PhC2	I	Master encoder phase C
23a	–PhC2	I	
24a	+5V	0	Slave encoder power supply
25a	GND		Ground
26a	+PhA1	I	Slave encoder phase A
27a	–PhA1	I	
28a	+PhB1	I	Slave encoder phase B
29a	–PhB1	I	
30a	+PhC1	I	Slave encoder phase C
31a	–PhC1	I	
32a	+24V	I	CleverAx power supply

Connector side C

Pin No	Function	Type	Description
1c	OUT 1	O	General purpose output
2c	OUT 2	O	General purpose output
3c	OUT 3	O	General purpose output
4c	OUT 4	O	General purpose output
5c	OUT 5	O	General purpose output
6c	Servo enable	O	Enable the controlled servodrive
7c	OUT 7	O	General purpose output
8c	OUT 8	O	General purpose output
9c	GND		Ground
10c	+24V	O	Internally connected to the power supply
11c	IN 1	I	General purpose input
12c	IN 2	I	General purpose input
13c	IN 3	I	General purpose input
14c	IN 4	I	General purpose input
15c	IN 5	I	General purpose input
16c	IN 6	I	General purpose input
17c	IN 7	I	General purpose input
18c	IN 8	I	General purpose input
19c	External trigger	I	Digital lock external trigger
20c	Store Slave position	I	Freeze Slave position and high speed counter P72
21c	Store Master position	I	Freeze Master position and high speed counter P71
22c	IN 12	I	General purpose input
23c	IN 13	I	General purpose input
24c	IN 14	I	General purpose input
25c	IN 15	I	General purpose input
26c	IN 16	I	General purpose input
27c	IN 17	I	General purpose input
28c	IN 18	I	General purpose input
29c	CleverAx ENABLE	I	CleverAx enable signal
30c	GND		Ground
31c	FAULT	O	Fault monitor terminal
32c	GND		Ground

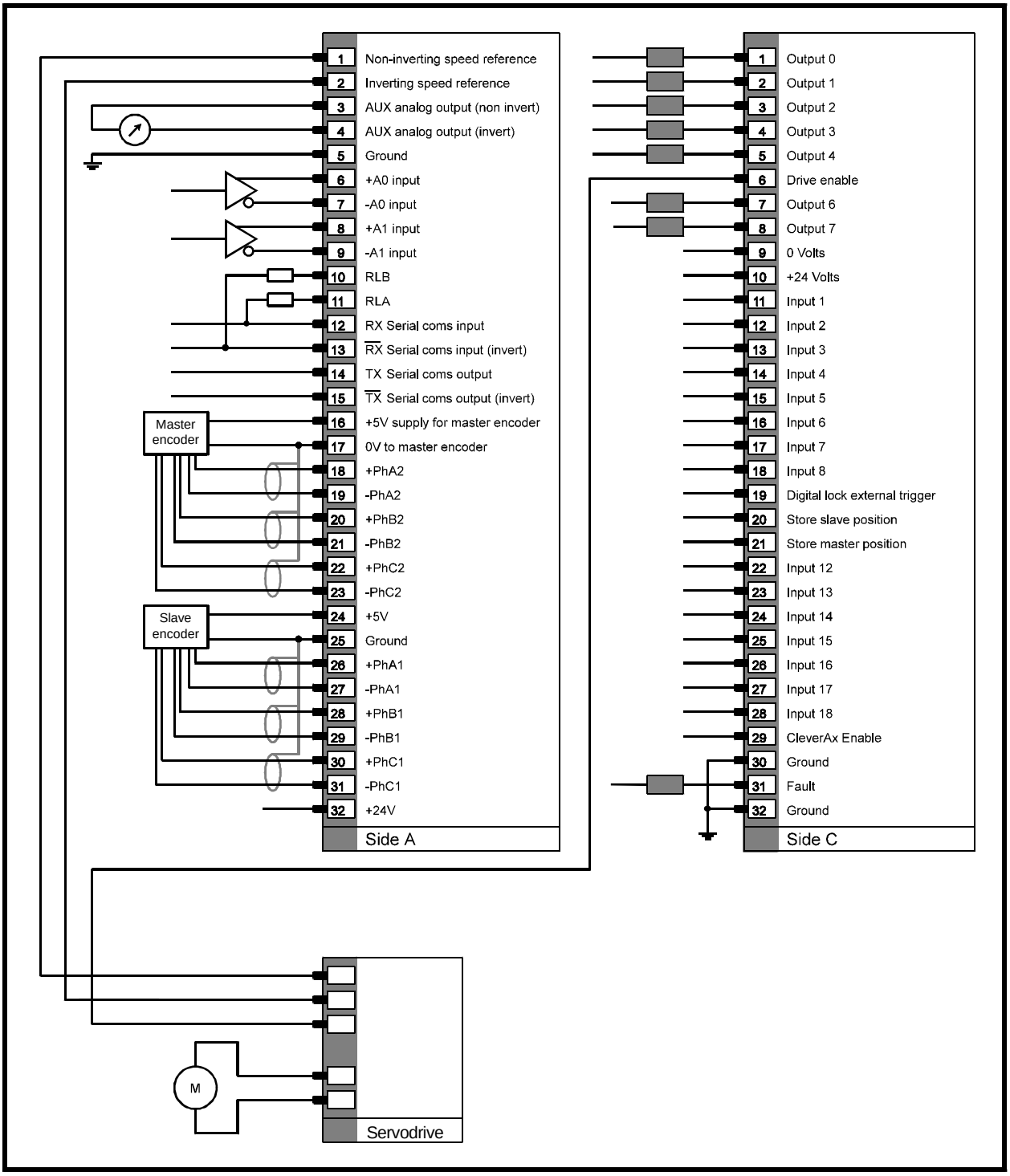


Figure 4–2 Signal connections

5 Parameters

5.1 PLC module

The CleverAx card contains a complete PLC module. This module is programmable using relay logic and communicates with the Motion module using a parameter-based interface.

CDT (CleverAx Development Tool) software runs on a PC. The PC is connected to the CleverAx card using a RS232 serial link, allowing the user to perform the following:

Read and write parameter values

Download and debug programs

Examples of program listings are given in Chapter 11 *Applications*.

PLC module specification

Function	Description
Constant scanning	The time between scanning cycles can be adjusted
Programming method	Relay logic
Type of control	Cyclical operation
Program memory	Internal EEPROM
Maximum number of instructions	999
Instruction type	High and low level instructions
Number of inputs	19
Number of outputs	8
Number of internal relays	224
Number of special internal relays	96 (**P31**, **P32** and **P73**)
Number of timers	6
Number of counters	3
Number of data registers	46 (from **P82** to **P128**)
Number of special registers	25 (all PLC and LINK parameters)
Number of Motion parameters	49
Number of index registers	1 (**P80**)
Number of labels	100

PLC instructions (ladder diagram program)

Mnemonic	Syntax	Description
LD (LOAD)	LD **P32.1**	Puts on the top of the stack the specified bit (bit 1 of **P32**)
OUT (OUTPUT)	OUT **P32.5**	The content of the top of the stack is loaded in the specified bit (bit 5 of **P32**)
AND (LOGIC AND)	AND **P31.1**	Performs a logic AND between the stack and the specified bit (stack AND bit 1 of **P31**)
OR (LOGIC OR)	OR **P31.1**	Performs a logic OR between the stack and the specified bit (stack OR bit 1 of **P31**)
LDN (LOAD NOT)	LDN **P31.1**	Puts on the top of the stack and inverts the specified bit (bit 1 of **P32** inverted)
OUTN (OUTPUT NOT)	OUTN **P31.1**	The content of the top of the stack is negated and loaded in the specified bit (NOT stack in bit 5 of **P32**)
ANDN (AND NOT)	ANDN **P31.1**	Performs a logic AND between the stack and the specified bit negated (stack and NOT bit 1 of **P31**)
ORN (OR NOT)	ORN **P31.1**	Performs a logic OR between the stack and the specified bit negated. (stack OR NOT bit 1 of **P31**)
ADDW (ADDING WORD)	ADDW **P1 P2 P3**	**P1** = **P2** + **P3**
SUBW (SUBTRACT WORD)	SUBW **P1 P2 P3**	**P1** = **P2** − **P3**
MULW (MULTIPLY WORD)	MULW **P1 P2 P3**	**P1** = **P2** × **P3**
DIVW (DIVIDE WORD)	DIVW **P1 P2 P3**	**P1** = **P2** ÷ **P3**
ORW (LOGIC OR WORD)	ORW **P1 P2 P3**	**P1** = **P2** OR **P3**
ANDW (LOGIC AND WORD)	ANDW **P1 P2 P3**	**P1** = **P2** AND **P3**
XORW (LOGIC XOR WORD}	XORW **P1 P2 P3**	**P1** = **P2** XOR **P3**
COMW (COMPLEMENT WORD)	COMW **P2 P3**	**P2** = complement bit to bit of **P3**
LSHW (LOGIC LEFT SHIFT WORD)	LSHW **P2 P3**	**P2** = Left Shift of **P3**

Mnemonic	Syntax	Description
RSHW (LOGIC RIGHT SHIFT WORD)	RSHW **P2 P3**	**P2** = Right Shift of **P3**
MOVW (MOVE WORD)	MOVW **P8** 200000	Load immediate (**P8** = 200000)
JUMP	JUMP TEST	Unconditioned jump to label (test)
START	START	Divides the area for cam co-ordinates from the ladder diagram program
XY=	XY=10,10	Cam co-ordinates
BREAK	BREAK	Divides different cam table
END (END PLC PROGRAM)	END	Defines the end of the program
ANDST (LOGIC AND)	ANDST	Logic AND between stack operands
ORST (LOGIC OR)	ORST	Logic OR between stack operands
SET (SET BIT)	SET **P32.0**	Set bit 1 in **P32** at 1 if stack is true
RES (RESET BIT)	RES **P32.1**	Set bit 1 in **P32** at 0 if stack is true
WRPAR (Write in parameter area)	WPAR **P8**,(13) Direct addressing	Writes **P8** value into parameter area address 13 eg. if **P8** = 128, then **P13** = 128
[Address range 1 to 128]	WPAR P8,P100 Indirect addressing	Writes **P8** value into parameter area at the address specified by the **P100** content. eg. if **P8** = 128 and **P100** = 13, then **P13** = 128
WRPLC (Write PLC area)	WPLC **P8**,(200) Direct addressing	Writes **P8** value into PLC area address (200) eg. **P8** = 128, then PLC instruction 200 = 128
[Address range 1 to 999]	WPLC **P8**,P100 Indirect addressing	Writes **P8** value into PLC area at the address specified by the **P100** content eg. if **P8** = 128 and **P100** = 200, then PLC instruction number 200 = 128

Mnemonic	Syntax	Description
WRMEM (Write Expansion)	WMEM **P8**,(512) Direct addressing	Writes **P8** value into the Expansion port address (512)
[Address range 1 to 1024]	WMEM **P8**,P100 Indirect addressing	Writes **P8** value at Expansion port addressed by **P100** content
RDPAR (Read parameter area)	RDPAR **P8**,(13) Direct addressing	Reads the address (13) content and store it in **P8** eg. if **P13** = 255, then **P8** = 255
[Address range 1 to 128]	RDPAR **P8**,P100 Indirect addressing	Reads the parameter value at the address specified by **P100** and stores it in **P8**. eg. if **P100** = 13 and **P13** = 255, then **P8** = 255
RDPLC (Read PLC area)	RDPLC **P8**,(200) Direct addressing	Reads the PLC area address (200) content and stores it in **P8** eg. if PLC instruction number 200 = 255, then **P8** = 255
[Address range 1 to 999]	RDPLC **P8**,P100 Indirect addressing	Reads the PLC area address specified by the **P100** content and stores it in **P8**. eg. if **P100** = 200 and PLC instruction number 200 = 255, then **P8** = 255
RDMEM (Read Expansion)	RDMEM **P8**,(512) Direct addressing	Reads the Expansion port address (512) content and stores it in **P8**
[Address range 1 to 1024]	RDMEM **P8**,P100 Indirect addressing	Reads the Expansion port address specified by **P100** content and stores it in **P8**

Ladder diagram instructions internal code

Mnemonic	Hexadecimal code	Effect on Stack
LD	H81 00 PP bb	+
OUT	H82 00 PP bb	–
AND	H83 00 PP bb	=
OR	H84 00 PP bb	=
LDN	HC1 00 PP bb	+
OUTN	HC2 00 PP bb	–
ANDN	HC3 00 PP bb	=
ORN	HC4 00 PP bb	=
ADDW	H11 SS AA BB	–
SUBW	H12 SS AA BB	–
MULW	H13 SS AA BB	–
DIVW	H14 SS AA BB	–
ORW	H21 SS AA BB	–
ANDW	H22 SS AA BB	–
XORW	H23 SS AA BB	–
COMW	H24 00 AA BB	–
LSHW	H31 00 AA BB	–
RSHW	H32 00 AA BB	–
JUMP	H33 00 00 XX	–
MOVW	H80 00 00 PP H xx xx xx xx	–
START	HE1 00 00 00	=
XY=	HE1 y yy yy yy	=
END	HFF 00 00 00 00	=
ORST	HA1 00 00 00	–
ANDST	HA3 00 00 00	–
SET	H 85 00 00 00	–
RES	H 86 00 00 00	–
BREAK	HEE 00 00 00	=

SS	Parameter result
bb	Bit
PP	Parameter
AA	Parameter operand
XX	Program line number (max 999)
BB	Parameter operand
xx xx xx xx	32-bit number
y yy yy yy	28-bit number
–	Decrease stack
+	Increase stack
=	No stack modification

Mnemonic	Hexadecimal code	Effect on Stack
Direct addresses		
WRPAR	H41 PP 8 xxx	–
WRPLC	H41 PP 4 xxx	–
WRMEM	H41 PP 0 xxx	–
Indirect addresses		
WRPAR	H41 PP a0 PP	–
WRPLC	H41 PP 60 PP	–
WRMEM	H41 PP 20 PP	–
Direct addresses		
RDPAR	H42 PP 8 xxx	–
RDPLC	H42 PP 4 xxx	–
RDMEM	H42 PP 0s xxx	–
Indirect addresses		
RDPAR	H42 PP a0 PP	–
RDPLC	H42 PP 60 PP	–
RDMEM	H42 PP 20 PP	–

PP	Parameter
XXX	Address value
–	Decrease stack
+	Increase stack
=	No stack modification

PLC module — dedicated parameters

Parameter (and bit)	Description
P50	Digital inputs
P54	Digital outputs
P58, P59	Auxiliary analog output is the difference between **P58** and **P59**
P60	Analog input 0 (non-inverting)
P61	Analog input 0 (inverting)
P62	Analog input 1 (non-inverting)
P63	Analog input 1 (inverting)
P64, P73.1	**P64** = Timer 0. **P73.1** = 1 when Timer 0 = 0
P65, P73.2	**P65** = Timer 1. **P73.2** = 1 when Timer 1 = 0
P66, P73.3	**P66** = Timer 2. **P73.3** = 1 when Timer 2 = 0
P67, P73.4	**P67** = Timer 3. **P73.4** = 1 when Timer 3 = 0
P68, P73.5	**P68** = Timer 4. **P73.5** = 1 when Timer 4 = 0
P69, P73.6	**P69** = Timer 5. **P73.6** = 1 when Timer 5 = 0
Timers are decremented every sampling time (500µs or 1ms)	

Parameter (and bit)	Description
P70, P73.7	P70 = Counter 0. P73.7 = 1 during the rising edge of the clock
P71, P73.8	P71 = Counter on rising edge of input 11. P73.8 = 1 resets P71
P72, P73.9	P72 = Counter on rising edge of input 10. P73.9 = 1 resets P72
P73.10	P73.10 = 1 sets PLC program for autoload and autorun
P73.11	Sampling times: 0 = 1ms, 1 = 0.5 ms Use the following procedure to change the sampling time: Program P73.11 with the desired value Save in non-volatile memory Remove and re-connect the supply to the CleverAx card.
P73.12	P73.12 = 1 saves Motion* parameters and PLC flags
P73.13	P73.13 = 1 loads default parameters
P73.15	P73.15 = 1 indicates a short-circuit on a digital output
P73.16	Zero Flag for arithmetic and logic operations
P73.17	P73.17 = 1 Start constant scan for PLC program using parameter P49 P49 is used only to sample the axis speeds
P73.18	P73.18 = 1 saves Custom parameters*
P73.19	P73.19 = 1 loads Custom parameters*
P73.27	P73.27 = 1 loads program stored in EEPROM
P73.28	P73.28 = 1 stores program in EEPROM
P73.29	P73.29 = 1 when Stack Error occurs
P73.31	P73.31 = 1 erases stored program
P73.32	P73.32 = 1 executes the stored program P73.32 = 0 stops the running program

* See Parameter map

5.2 Serial Link module

Reserved parameters

Parameter	Description
P74	Serial communication address*
P75	Specifies the parameter number
P76	Specifies the parameter value
P77	Specifies the position of the decimal point in the parameter value
P78	Operation code P78 = 1 starts transmission (set by the program) P78 = 2 ends the transmission (set by the system) P78 = 3 indicates acknowledge received (set by the system) P78 = 4 Starts receive (set by the program) P78 = 5 ends receive (set by the system) P78 = 6 indicates acknowledge (set by the system)

Address format example

Set address = 01

H4A = 30 30 31 31	or	P74 = 808464689
Hexadecimal format (if used)		Decimal format (if used)

When CleverAx is the Master of the system, P74 contains the address of the Slave module to be connected.

When CleverAx is the Slave, P74 contains the CleverAx address.

Parameter	Description
P80	Ladder diagram program instruction pointer (max 999)
P81	Instruction pointed by P80
P82	Character buffer. Contains the last character received on RS232 serial interface
P83, P73.20	Contains the numerical value that will be sent to the Serial Link when P73.20 is in transition from 0 to 1
P84, P73.21	Contains the address of the first character of the alphanumerical string that will be sent to the Serial Link when P73.21 is in transition from 0 to 1

Parameters P82, P83 and P84 are fully available to the user as general purpose registers when the RS232 serial link is not used.

5.3 Motion module

The main functions of the Motion module are as follows:

Motion control
Speed control
Position control

Parameters are used to set the configuration of the motion and loop gains for each axis.

Reserved parameters

Parameter	Description
P1	Master encoder position
P2	Internal absolute position reference
P3	Internal speed reference
P4	Slave reference value
P5	Not used
P6	Slave position offset
P7	Slave speed offset
P8	Slave encoder position
P9	Loop position reference
P10	Position error
P11	Digital lock ratio
P12	Acceleration ramp
P13	Deceleration ramp
P14	Max speed (for **P2** and **P6** during digital lock phasing)
P15	Main analog output offset
P16	Loop correction limit
P17	Maximum following error (**P32.1** flag)
P18	Position compare 1 value
P19	Position compare 2 value
P20	Position compare 3 value
P21	Position compare 4 value
P22	Freeze and store Slave position
P23	Freeze and store Master position
P24	Master speed
P25	Slave speed
P26	Manual speed
P27	Feed forward, derivative if **P32.6** = 1
P28	Proportional gain
P29	Integral gain
P30	Emergency position reference
P31	Flag bit parameter
P32	Switch bit parameter
P33	Position reference 1
P34	Position reference 2
P35	Position reference 3
P36	Position reference 4
P37	Position reference 5
P38	Position reference 6
P39	Set point speed after digital lock disable
P40	Position offset during digital lock operations
P44	Position target after digital lock disable
P45	Store location for Slave position when enabled the interrupt on the slave index occurrence
P47	Number of the Cam table to be loaded
P48	Recovery position (monitor the absolute position)
P49	PLC program constant scan or speed evaluation

Description of the motion parameters

The motion parameters have the following format and limits:

Position parameters

Length: 32 bits
Maximum value: ±2 147 483 648 counts
Counts = encoder steps × 4

Speed parameters

Maximum value: 480000 counts per sec

Acceleration parameters

Expressed in counts per sec^2

P1
Master encoder position

This parameter is the Master encoder step 32-bit counter.

The encoder signals from the Master encoder are multiplied by four and stored in this register.

This parameter is used in digital lock and cam applications.

P2
Internal absolute position reference

This parameter is a 32-bit register that contains the target position. When selected, the internal position references are set as follows:

P31.13 = 0
P32.2 = 0
P32.3 = 1

P3
Internal speed reference

When the CleverAx is programmed to operate in speed mode as follows:

P32.2 = 0
P32.3 = 0

This parameter becomes the internal speed reference and is programmable to a maximum value of 480000 counts per sec.

If the encoder is directly fixed to the motor shaft, the motor speed is calculated as follows:

$$\text{Motor speed} = \frac{P3}{4} \times \frac{60}{\text{Encoder steps per revolution}}$$

P4
Slave reference value

This parameter is the Slave axis position reference that is directly loaded in **P9** (loop position reference) when the cam function is disabled (**P32.4** = 1).

P5

Not used

P6
Slave position offset

When this parameter is programmed with a value other than zero, a position offset is added to the target position in **P4**.

This **secondary** profile is executed during the **primary** motion (with or without S-ramp) after **P32.8** is adjusted.

When the execution of this **secondary** profile is completed, **P31.11** is set at 1.

P7
Slave speed offset

This parameter is programmed with a speed value. The speed value is added to the computed speed necessary to reach the **P4** target position. The added profile is executed (with or without S-ramps — **P32.20**).

P8
Slave encoder position

When **P32.16** = 1 (CleverAx software enabled), **P8** becomes a copy of **P48** (Slave encoder counter).

When **P32.16** = 0 (CleverAx software disabled), **P8** is loaded with the position reference value held in **P9**.

P9
Position Loop reference

This parameter gives the position target value used as the position reference in the input to the control loop.

P10
Position error

This parameter indicates a value which is the difference between the position loop reference **P9** and the actual Slave position **P8**.

P11
Digital lock ratio

When **P32.2** = 1, Digital lock is selected. The Slave position reference **P4** is calculated as the Master encoder position **P1** multiplied by the content of **P11**.

The ratio can be a value from 0 to 2 as a decimal number with 9 decimal places.

P11 has to be programmed with the first ten digits of the decimal part of the desired ratio value divided by 10.

Example:

To program a ratio of 1:1.5,
P11 = 1 500 000 000

To program a ratio of 1:0.876 543 21,
P11 = 0 876 543 210

To program a ratio of 1:0.001 23,
P11 = 001 230 000 (or 1 230 000)

The direction of the Slave motion is reversed when **P32.1** = 1.

P12
Acceleration ramp

This parameter is the acceleration value used to reach either of the following:

The speed reference when this is greater than the actual speed

Maximum programmed speed during a positioning profile

P13
Deceleration ramp

This parameter is the deceleration value used to reach either of the following:

The speed reference when this is less than the actual speed

Zero speed during a positioning profile

P14
Maximum speed during digital lock phasing

The slave axis must accelerate to follow the correct profile when any of the following conditions occurs:

P32.2 = 1 (Digital lock enabled)
P6 ≠ 0 (Slave position offset enabled)
P2 ≠ 0 (Internal position reference enabled)

The maximum speed during phasing is limited by **P14**.

P15
Main analog output offset

This parameter allows the offset of the analog reference output to be corrected

P16
Loop correction limit

This parameter sets the maximum permissible value for the reference output.

When **P16** = 65535 (the maximum value), no limiting function is performed on the output data.

P17
Maximum following error

This parameter sets the maximum following error allowed in the specific application.
The following error is computed as the difference between the position reference and the actual position of the controlled axis and is contained in **P10**.

When the value of **P10** becomes greater than the value of **P17**, **P32.1** = 1.

P18
Position compare 1 value

When **P32.9** = 0, this parameter stores a position value that is compared with the actual position of the Master encoder

When **P32.9** = 1 this parameter stores a position value that is compared with the actual position of the Slave encoder

When the actual position becomes greater than the value programmed in **P18**, **P31.2** = 1.

P19
Position compare 2 value

When **P32.10** = 0, this parameter stores a position value that is compared with the actual position of the Master encoder

When **P32.10** = 1 this parameter stores a position value that is compared with the actual position of the Slave encoder

When the actual position becomes greater than the value programmed in **P19**, **P31.3** = 1.

P20
Position compare 3 value

When **P32.11** = 0, this parameter stores a position value that is compared with the actual position of the Master encoder

When **P32.11** = 1 this parameter stores a position value that is compared with the actual position of the Slave encoder

When the actual position becomes greater than the value programmed in **P20**, **P31.4** = 1.

P21
Position compare 4 value

When **P32.12** = 0, this parameter stores a position value that is compared with the actual position of the Master encoder

When **P32.12** = 1 this parameter stores a position value that is compared with the actual position of the Slave encoder

When the actual position becomes greater than the value programmed in **P21**, **P31.5** = 1.

P22
Freeze and store Slave position

Following a low-to-high transition at input 10 (terminal 20c) the actual Slave position is stored in **P22**.

P23
Freeze and store Master position

Following a low-to-high transition on input 11 (terminal 21c) the actual Master position is stored in **P23**.

P24
Master speed

This parameter indicates the Master encoder speed in counts per sampling time

P25
Slave speed

This parameter indicates the Slave encoder speed in counts per sampling time

P26
Manual speed

This parameter sets the required axis speed in counts per sec when operating in manual mode with:

P32.2 = 0
P32.3 = 1
P31.13 = 1

P27
Feedforward or derivative gain

Depending on the setting of **P32.6**, the value programmed in this parameter is as follows:

When **P32.6** = 0, **P27** is the Feedforward gain
When **P32.6** = 1, **P27** is the loop Derivative gain

P28
Proportional gain

P29
Integral gain

P30
Emergency position reference

This parameter becomes the position reference when **P32.5** = 0. It is normally used during emergency management.

P31
Flag bit parameter

The bits in this parameter are as follows:

Parameter and bit	Description
P31.1	Servo error
P31.2	Position compare **P18**
P31.3	Position compare **P19**
P31.4	Position compare **P20**
P31.5	Position compare **P21**
P31.6	Slave index detect
P31.7	Master index detect
P31.8	Digital input 9 detect
P31.9	
P31.10	**P31.10** = 0 indicates end of **P2** profile **P31.10** = 1 indicates positioning in progress
P31.11	**P31.11** = 0 indicates end of **P6** Profile **P31.11** = 1 indicates positioning in progress
P31.12	**P31.12** = 1 indicates Digital Lock is in phase (active only when **P32.27** = 0)
P31.13	**P31.13** = 1 indicates Manual operation **P31.13** = 0 indicates Automatic operation
P31.14	Joystick clockwise
P31.15	Joystick anti-clockwise
P31.16	Limit switch (automatically managed only when **P2** is the position reference)
P31.17	Limit switch (automatically managed only when **P2** is the position reference)
P31.18	Emergency
P31.19	
P31.20	
P31.21	
P31.22	
P31.23	
P31.24	
P31.26	
P31.27	Select reference from **P33**
P31.28	Select reference from **P34**
P31.39	Select reference from **P35**
P31.30	Select reference from **P36**
P31.31	Select reference from **P37**
P31.32	Select reference from **P38**

P32
Switch bit parameter

The bits in this parameter are as follows:

Parameter and bit	Description
P32.1	**P32.1** = 1 inverts the Slave reference
P32.2	**P32.2** = 1 enables Digital Lock
P32.3	**P32.3** = 0 selects source **P3** **P32.3** = 1 selects source **P2**
P32.4	**P32.4** = 0 inserts a cam
P32.5	**P32.5** = 0 selects emergency reference
P32.6	**P32.6** = 0 selects Feedforward gain **P32.6** = 1 selects Derivative gain
P32.7	Not used. **P32.7** must always be = 1
P32.8	**P32.8** = 0 enables S-ramps using **P2** as position reference
P32.9	**P32.9** = 0 selects Master encoder comparator **P32.9** = 1 selects Slave encoder comparator
P32.10	**P32.10** = 0 selects Master encoder comparator **P32.10** = 1 selects Slave encoder comparator
P32.11	**P32.11** = 0 selects Master encoder comparator **P32.11** = 1 selects Slave encoder comparator
P32.12	**P32.12** = 0 selects Master encoder comparator **P32.12** = 1 selects Slave encoder comparator
P32.13	**P32.13** = 1 enables Trigger on the Slave index
P32.14	**P32.14** = 1 enables Trigger on the Master index
P32.15	**P32.15** = 1 enables Trigger on digital input 9 (terminal 19c)
P32.16	**P32.16** = 1 enables CleverAx software
P32.17	Reset
P32.18	Servo OK (input 18 on terminal 28c)
P32.19	**P32.19** = 0 enables S-ramps using **P3** as position reference
P32.20	**P32.20** = 0 enables S-ramps using **P7** as position reference
P32.21	
P32.22	**P32.22** = 0 starts the cam
P32.23	**P32.23** = 1 selects cam auto stop
P32.24	
P32.25	
P32.26	Enables trigger on Master index masked by digital input 9
P32.27	**P32.27** = 0 enables digital lock phasing
P32.28	**P32.28** = 0 enables digital lock ramps
P32.29	
P32.30	
P32.31	
P32.32	**P32.32** = 0 inserts filter (τ = 8ms) for Derivative or Feedforward control

P33
Internal position reference 1

A position reference value can be programmed into this parameter and the following five **Internal position reference** parameters (**P33** to **P38**). Simple position profiles can be created by selecting all or some of these parameters.

To select **P33** set **P31.27** at 1.

P34
Internal position reference 2

To select **P34** set **P31.28** at 1.

P35
Internal position reference 3

To select **P35** set **P31.29** at 1.

P36
Internal position reference 4

To select **P36** set **P31.30** at 1.

P37
Internal position reference 5

To select **P37** set **P31.31** at 1.

P38
Internal position reference 6

To select **P38** set **P31.32** at 1.

P39
Set point speed after digital lock disable

At the instant Digital Lock is enabled, **P39** is set equal to the Master encoder speed multiplied by the ratio programmed in **P11**. The value of **P39** remains unchanged if the Master encoder speed subsequently changes or if the ratio is changed.

After Digital Lock is disabled, the slave axis uses **P39** as the speed reference.

If speed or ratio is likely to change when Digital Lock is enabled, before disabling Digital Lock it is good practice to load the actual slave encoder speed value (held in **P25**) into **P39**.

P40
Position offset during digital lock operation

This parameter allows a position offset to be added to the controlled axis when Digital Lock is enabled.

This parameter could be used to insert a pre- or post-trigger value.

P44
Position target after digital lock disable

When Digital Lock is used in position mode (phased) and **P32.3** = 1, and Digital Lock is then disabled, **P44** becomes the target position reference for the Slave axis.

P45
Store location for Slave position

When Slave Index Detect Interrupt is enabled (**P32.13** = 1), at the first occurrence of a Slave encoder index signal the actual Slave position is stored in **P45**.

P47
Cam table number

Contains the number of the cam table to be loaded and executed

P48
Slave encoder position

This parameter is the Slave encoder step counter and contains the absolute position of the Slave.

P49
PLC program constant scan or speed evaluation

The sampling time programmed in **P73.11** multiplied by the value stored in **P49** is the PLC scan time.

Example:

When **P73.11** = 1 (1ms sampling time) and **P49** = 10, the PLC scan time is 10ms.

When **P73.11** = 0 (0.5 ms sampling time) and **P49** = 5, the PLC scan time is 2.5 ms

6 Commissioning

6.1 Offset adjustment

Use the following procedure to set the offset:

1. Connect terminal 29c to +24V to enable the CleverAx.
2. Set **P54.6** and **P32.16** at 1 to enable the control software.
3. Set **P28** at a suitable value proportional gain.
4. Monitor **P10** (position error), and adjust **P15** (internal offset) to obtain the lowest value in **P10**.

6.2 Reset

During commissioning, it may be necessary to reset the CleverAx card.

Reset the card by interrupting the +24V supply for several seconds.

6.3 Recall default status

Parameters can be reset to their default values at any time by setting **P73.13** at 1.

Set **P32.17** at 1 to reset (to zero) the following parameters:

P1, P2, P3, P4, P5, P6, P7, P8, P9, P10, P30, P48

Default values

P0 to **P10**	0.
P11	1000000000
P12	**P13** = **P14** = 100000
P15	−512
P16	32000
P26	400
P27	146
P28	1280
P49	40
P32.4	1
P32.5	1
P32.7	1
P32.22	1
P74	H4A = 30303131
P73.11	1

6.4 Serial ports

The RS232 port uses a 9-pin D-type connector for connection to a PC for the following purposes:

Change parameter values
Edit programs
Change configuration

The RS485 port is used for multi-drop systems.

The baud rate for both interfaces is set at 9600.

RS232 link

Use a three-wire screened cable to connect the CleverAx card to a PC.

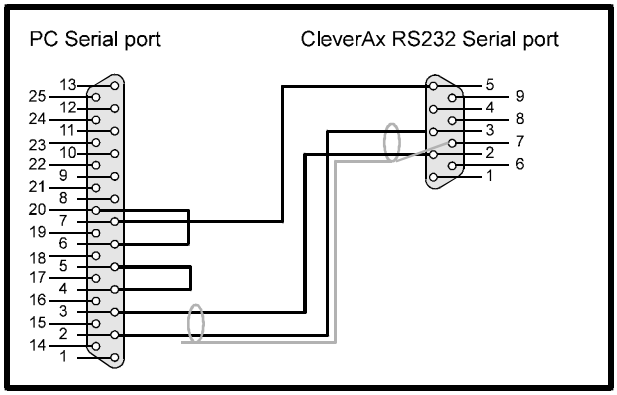


Figure 6–1 Serial communications connections

CleverAx serial port 9-way	PC serial port 25-way
Pin 2 (Tx)	Pin 3
Pin 3 (Rx)	Pin 2
Pin 5 (GND)	Pin 7
	Connect pin 4 to pin 5
	Connect pin 6 to pin 20
Connect pin 7 to screen	

RS485 interface

Enter the required address (1 to 32) in parameter **P74**. Default address = 1.

The protocol is standard ANSI x 3.28-2.5 A4 which allows operation with all digitally controlled Drives manufactured by Control Techniques.

The BCC (Block Checksum) function is always enabled.

6.5 Start-up procedure

1. Connect the CleverAx card to the PC.
2. Insert the diskette containing the CDT file into the floppy diskette drive of the computer.
3. Type **CDT <CR>** to start the CDT program.
4. Press the F4 key on the keyboard to select Terminal Emulation.
5. Apply +24V to the CleverAx card. A welcome message and the prompt **ClAx>** are displayed.

Parameters can now be read and modified.

6.6 Editing a PLC program

1. Connect the CleverAx card to the PC.
2. Insert the diskette containing the CDT file into the floppy diskette drive of the computer.
3. Type **CDT <CR>** to start the CDT program.
4. Press the F5 key on the keyboard to select the editor
5. Write or edit the program
6. Press the F6 key on the keyboard to compile the program. The download function loads the program into the CleverAx memory.

7 Diagnostics

CleverAx enabled

Red LED 1	
Lit	Hardware enable signal and software enable command are present
Unlit	Hardware enable signal not present or software enable command not received

PLC in RUN mode

Red LED 2	
Lit	PLC program running (P73.32=1)
Unlit	PLC program stopped (P73.32=0)

24V supply

Green LED	
Lit	24V supply applied
Unlit	24V supply removed

8 Configuring CleverAx

8.1 Position loop

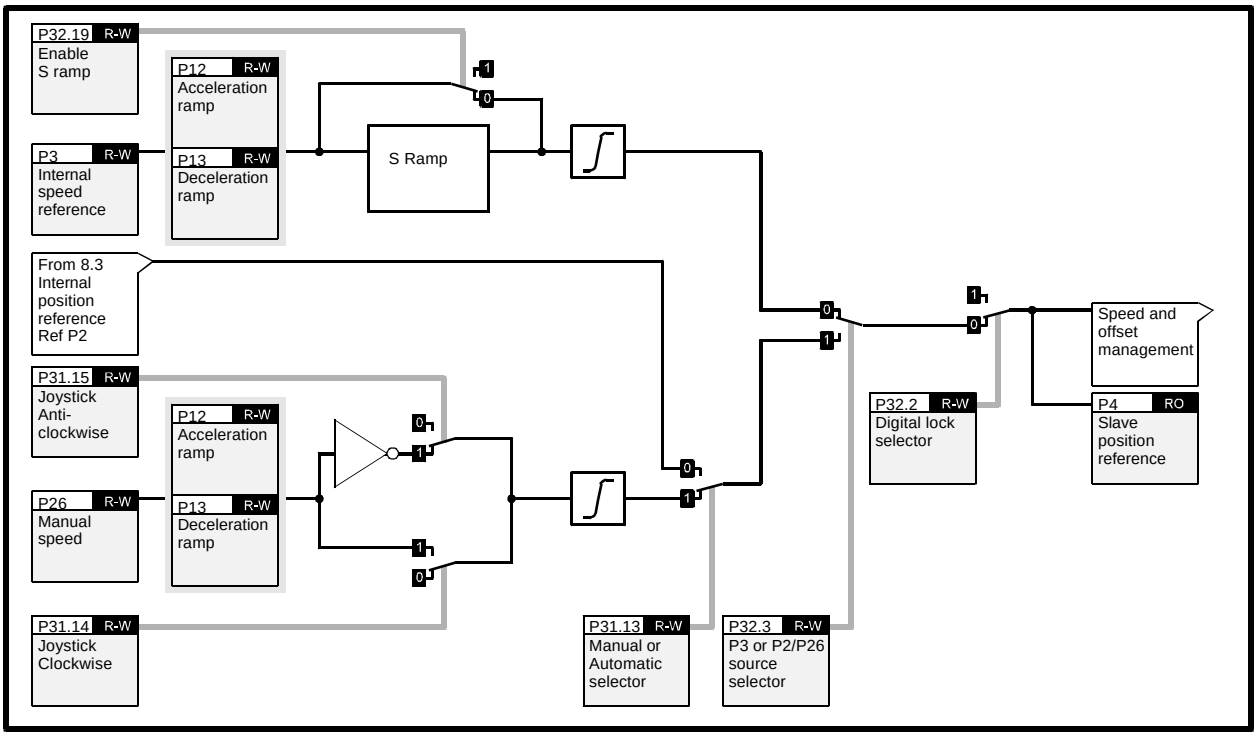
To evaluate error in the actual position, the position loop requires a position reference held in parameter **P4** to be compared with the motor position.

The position reference can be one of the following:

The programmed position
The output of the Digital Lock controller
The internal multi-reference generator

8.2 Manual or internal speed reference selection (Speed mode)

CleverAx can operate in Speed mode to follow an internal or external speed reference value.

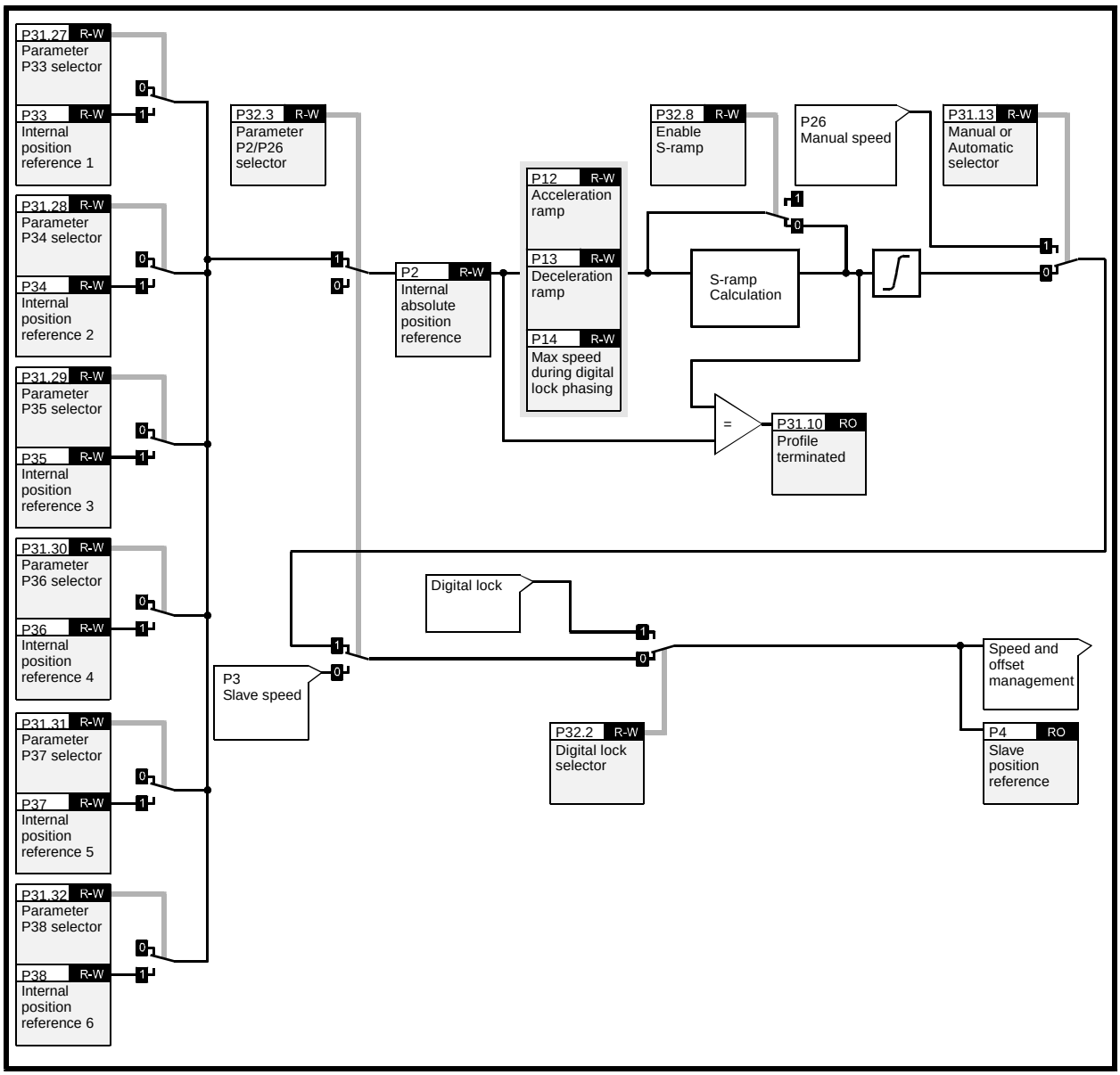


8.3 Internal position reference

When Digital Lock is disabled, Manual mode is disabled and **P32.3** = 1. **P2** then becomes the target position reference.

P2 may be set at one of six pre-programmed position references stored in parameters **P33** to **P38**. Each position reference parameter has a latch bit (**P31.27** to **P31.32**).

A transition from zero to one of a latch bit loads the content of the relevant position reference into **P2**.



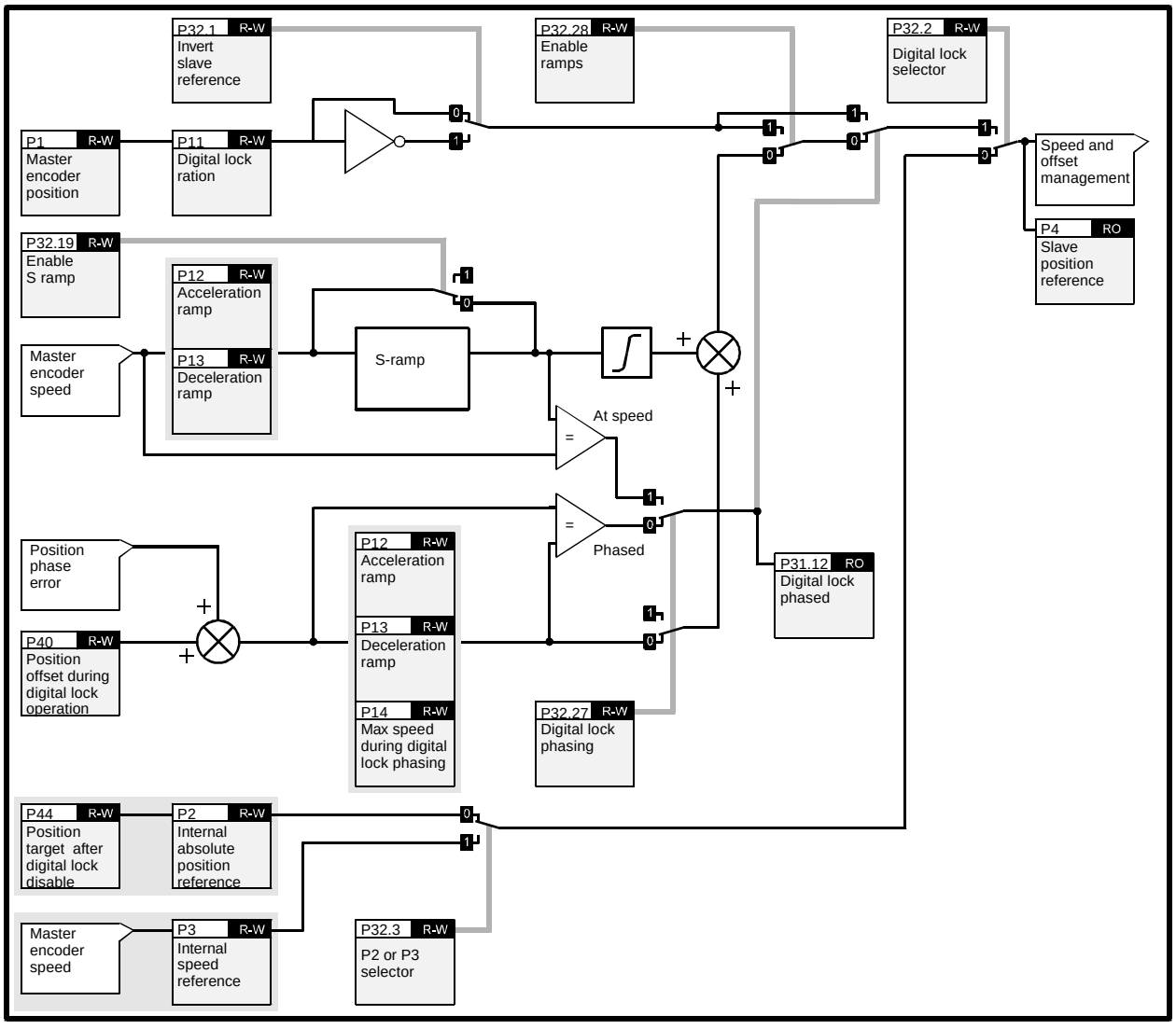
8.4 Digital Lock

Digital Lock can synchronize a Slave axis to a Master axis in the following ways:

Speed ratio
Position

If the Master is already running, synchronisation is possible with or without ramps.

When Digital Lock is enabled with phase recovery, Digital Lock can not be disabled before phase synchronization occurs (**P31.12** = 1). If an attempt is made to disable Digital Lock when the phases are not synchronized, position lock will be lost.

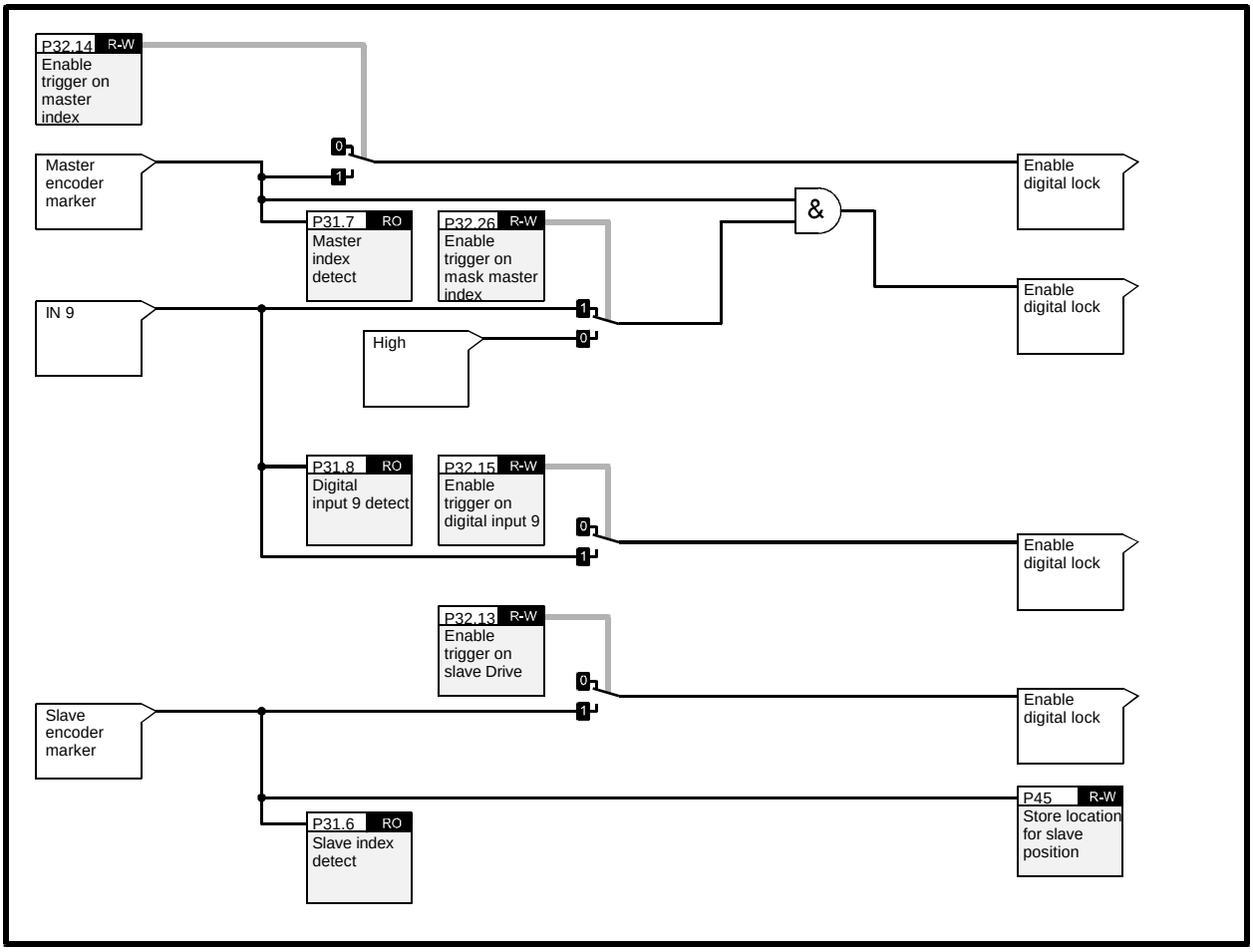


8.5 Digital Lock trigger and Slave Index detection

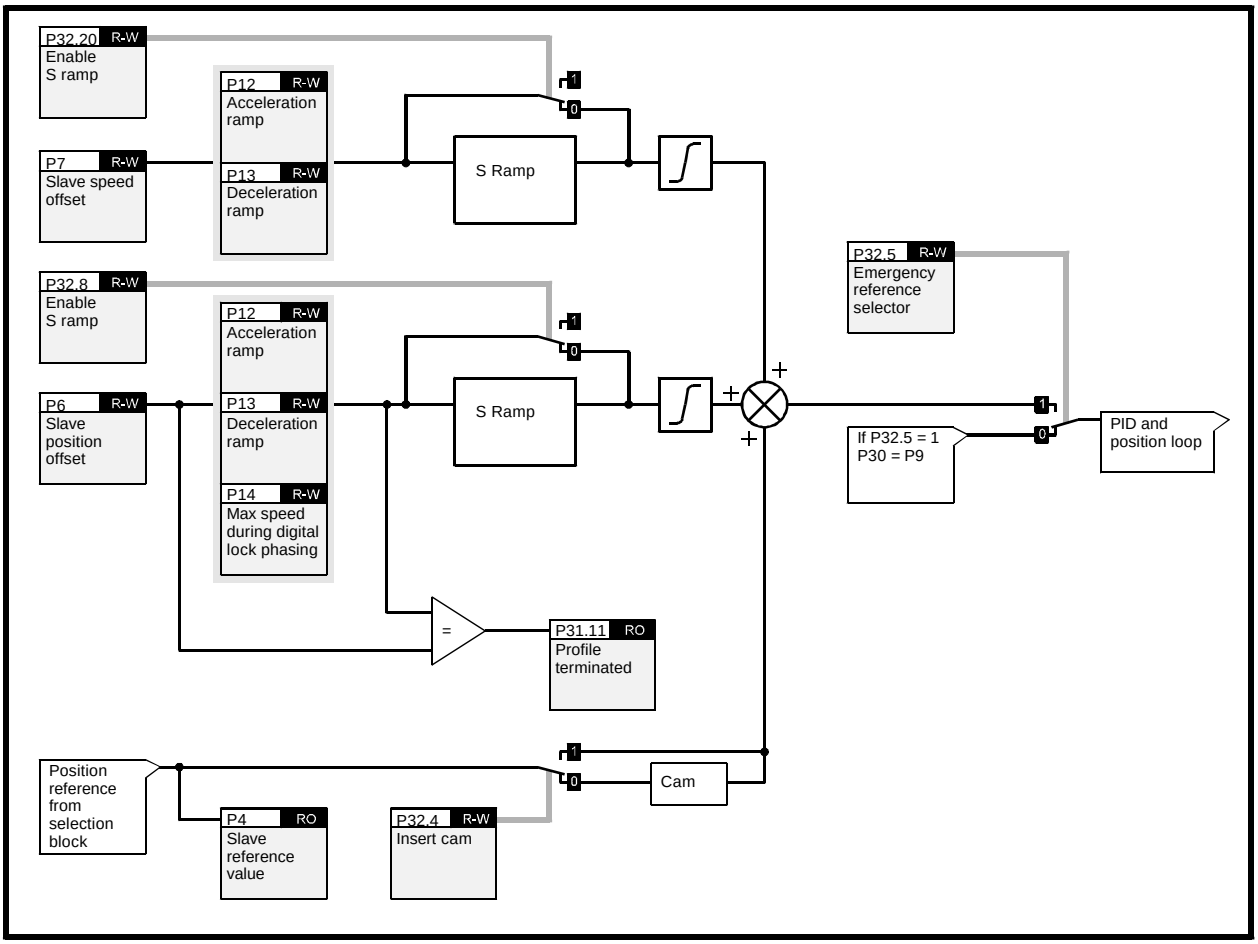
Digital Lock can be enabled in any of the following ways:

- Setting **P32.2** at 1
- Digital input signal (enabled by setting **P32.15** at 1) with a maximum delay of 5µs
- At the first occurrence of the Master encoder index (enabled by setting **P32.14** = 1)
- At the first occurrence of the Master encoder index when digital input IN9 is logic high (enabled by setting **P32.26** at 0 and **P32.14** at 0)

A pre-/post-trigger function is available using **P40**.



8.6 Speed and position offset management



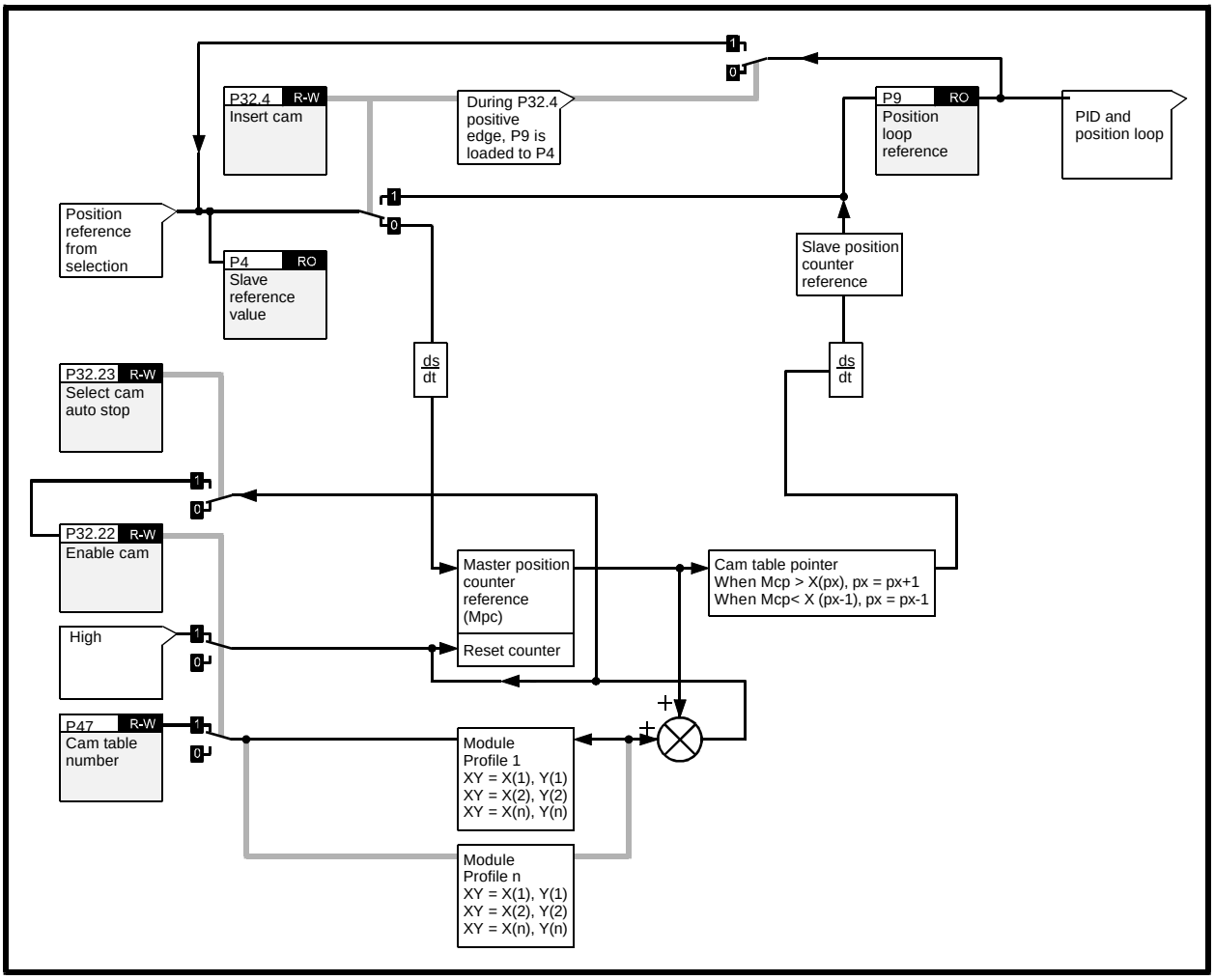
8.7 Electronic cam

Electronic cam describes the motion profile of the Slave, in relation to the position of the Master.

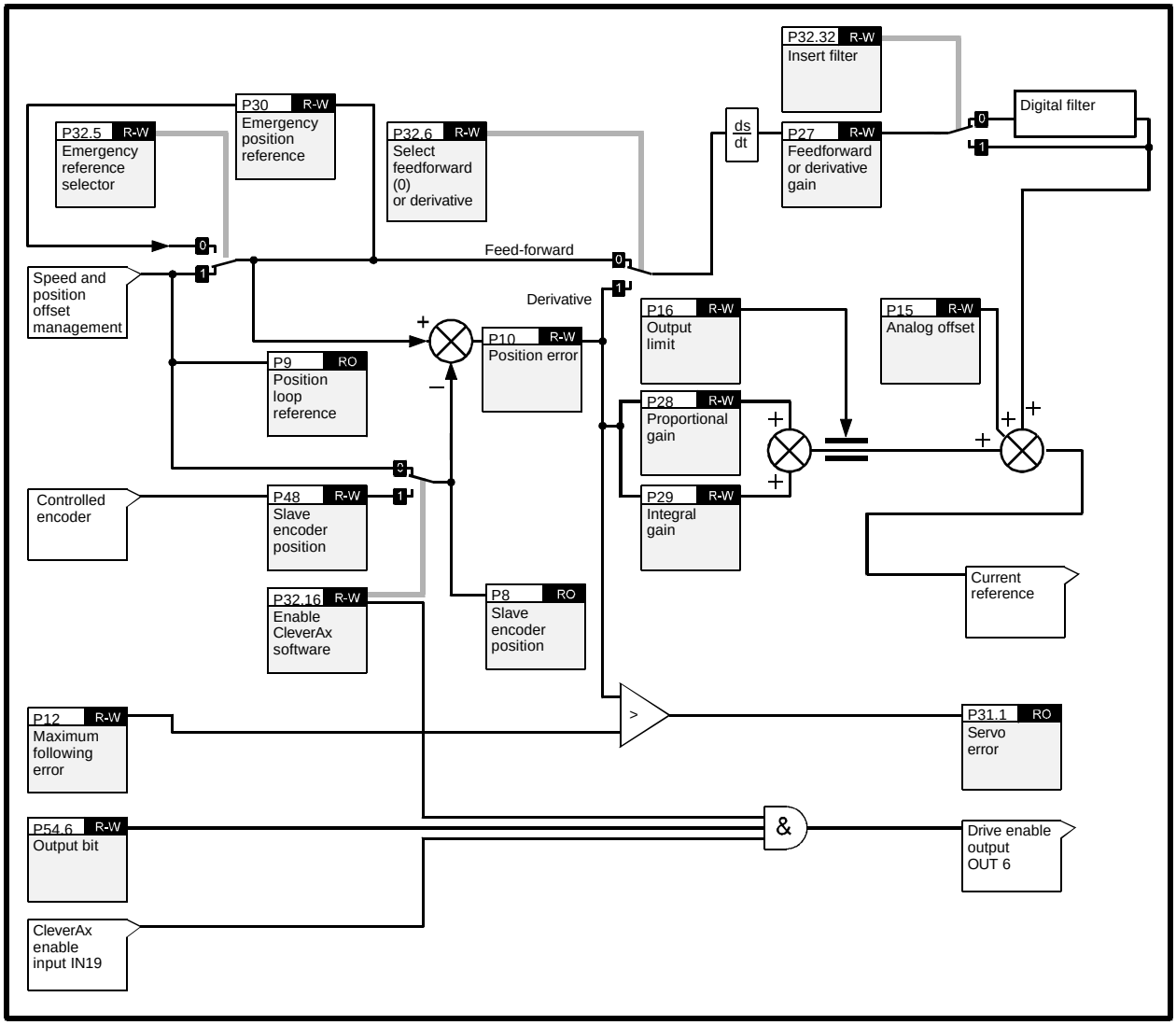
A look-up table (named MODULE) allows the Slave axis position to be specified for each position of the Master axis. The path between two consecutive points is computed using a linear interpolation algorithm.

In the case of a closed cam, the last position of the Master specified in the table is used to reset the counter and re-start the Slave at the first line.

A number of different tables can be held in memory and selected using **P47**. The length and number of tables is limited only by the amount of available memory.



8.8 PID and position loop



9 Software Tools

9.1 Terminal Emulation mode — command syntax

Write a parameter

Pxxx=value<**CR**>

P must be a capital letter
xxx is the parameter number
= is the equal sign
value is the desired setting value for the specified parameter
<CR> is the ENTER key
No spaces are allowed between characters.

Read a parameter value

Pxxx=<**CR**>

P must be a capital letter
xxx is the parameter number
= is the equal sign
<CR> is the ENTER key
No spaces are allowed between characters.

Repeat the last command

R<**CR**>

Interrupt repetitive display

Enter key (<**CR**>**)**

9.2 PLC program instruction syntax

The PLC program is a compiled program which optimizes the instruction execution time. During the compile phase, a complete syntax error detection is performed. The syntax is shown in the PLC instruction tables in Chapter 5 *Parameters*.

Refer to the following:

COMMENT LINE

When the first character of the line is a blank (space) character or a star (*) character, all the line is assumed to be a comment line and not compiled.

All the characters following an instruction are assumed to be comments.

LABEL

The label format is as follows:

|LABEL

(The pipe character | must be the first character on the line.)

INSTRUCTION

The instruction format is as follows:

Mnemonic

5 characters maximum
10 blank characters maximum
Capital letter P
Parameter identification

The first letter of the mnemonic field must be the first character of the line.

No blank (space) characters are allowed between P and the parameter identification number.

If the instruction requires multiple paragraphs to be defined, the parameter list must use a comma "," or a blank space for separation.

The immediate value (as in the MOVW instruction) must be written at the beginning of the line that follows the instruction line.

Examples of valid instruction syntax are as follows:

```
LD  P32.1
ADDW      P1,P2,P3
ADDW   P1 P2 P3
MOVW     P8
200000
JUMP LABEL
```

Instruction lines with the following syntax are evaluated as direct commands from the serial link and are given priority for execution:

#<parameter number>=<parameter value>#

9.3 How to use the CDT program

Text editor

F1 key <new>

Press this key to erase the loaded file and the displayed text. The following message is displayed:

THE COMMAND IS <NEW FILE> ARE YOU SURE Y/N ?

F2 key <load>

Press this key to open a windows with all the files having extension **.ldg** listed.

F3 key <save>

Press this key to save the current file. If the current file has no name, the following message is displayed:

FILE HAS NO NAME, SAVE TO ?

If the file is already named, the file is saved with the message:

SAVE "filename.ldg"

F4 key <term>

Press this key to select Terminal Emulation so that the CleverAx card can be controlled using the RS232 serial link.

F5 key <editor>

Press this key to return to the editor from the Terminal Emulation and Compiler functions.

F6 <comp>

Press this key to start the ladder diagram program compiler

The file in the editor is compiled. Two files are created having the following extensions:

.obj
.hex

The filename is the same as that of the edited file. If the edited file has no name, the compiled files are named: NO_NAME.

The file having extension **.ldg** contains the source code of the ladder diagram program

The file having extension **.obj** contains the source code, the compiled code and the stack value.

The file having extension **.hex** contains the executable compiled code.

When compilation is completed, the following message is displayed:

DOWNLOADING file.ldg Y/N?

If **Y** is typed, the code is downloaded into the CleverAx card via the RS232 serial link.

If **N** is typed, no action occurs.

F7 key <ink>

Press this key allows to change the background colour of the screen.

F8 key <debug>

Press this key to run the debugger utility. Use the arrows keys to modify parameter values.

F9 key <help>

Press this key to list descriptions of the function keys.

F10 key <quit>

Press this key to return to DOS.

9.4 How to use the CMT program

Use the CMT program to copy automatically the contents of the memory of one CleverAx card to the memory of another CleverAx card.

Command screen

F1 key <read>

Press this key to load parameter values (1 to 128) and PLC instructions (1 to 999) using the RS232 serial link.

The following message is displayed:

> **Connect the CleverAx source and press any key when ready**

Press a key. The copy process starts and the following messages are displayed in sequence:

> **>> Set PLC in STOP mode <<**
> **>> Disable PID loop control <<**
> **Uploading Parameter area %100**
> **Uploading PLC program into buffer**
> **Processing data for Checksum test %100**

F2 key <write>

Press this key to write the data into the destination CleverAx card using the RS232 serial link.

The following message is displayed:

> **Connect CleverAx destination and press any key when ready**

Press a key. The process starts and the following messages are displayed in sequence:

> **>> Set PLC in STOP mode <<**
> **>> Disable PID loop control <<**
> **Downloading Parameter area %100**
> **Downloading PLC program %100**
> **Saving parameter area and PLC instructions to EPROM**
> **Ready for new operation**

F9 key <help>

Press this key to list descriptions of the function keys.

F10 key <quit>

Press this key to quit the CMT program and return to DOS.

10 PID Algorithms

Proportional gain

$$\mathbf{Vout} = \mathbf{Step\ error} \times K_P \times 1.192 \times 10^{-6}$$

Integral gain

$$\frac{\mathbf{Vout}}{\mathbf{sec}} = \mathbf{Step\ error} \times K_I \times K_1$$

Where:

$K_1 = 2.328 \times 10^{-3}$ for 500µs sampling time

$K_1 = 1.164 \times 10^{-3}$ for 1ms sampling time

Feed-forward gain

$$K_{ff} = \frac{K_2}{V_{MAX}}$$

Where:

$K_2 = 58.59 \times 10^6$ for 500µs sampling time

$K_2 = 29.29 \times 10^6$ for 1ms sampling time

V_{MAX} = Maximum axis speed for 9.155V analog output

V_{MAX} unit is Step ÷ seconds

11 Applications

Notes

Digital inputs are identified by the following notation:

CDI 0 to CDI 18

Digital outputs are identified by the following notation:

CDO 0 to CDO 7

In the main body of this User Guide, the digital inputs are identified as IN 1 to IN 19 and the digital outputs as OUT 1 to OUT 8.

The relationships between the notations are as follows:

CDI 0 = IN1 = P50.1 = pin 11c
CDI 1 = IN2 = P50.2 = pin 12c
CDI 18 = IN19 = P50.19 = pin 29c
CDO 0 = OUT1 = P54.1 = pin 1c
CDO 1 = OUT2 = P54.2 = pin 2c
CDO 7 = OUT8 = P54.8 = pin 8c

11.1 Linear Positioner with Auto Learning

This application is a linear positioner with the following features:

- Up to six positions can be memorized
- Control a limit switch function
- Homing procedure
- Emergency stop
- S-ramp or linear ramp

The application is programmed by the user using a thumbwheel switch. Position settings can be saved in EEPROM.

Inputs

CDI 0
CDI 1
CDI 2

Thumbwheel switch inputs, as follows:

- CDI 0 — Least significant bit
- CDI 2.— Most significant bit

CDI 3

Not used.

CDI 4

Joystick input for clockwise motor rotation

If the joystick input is on, the speed is set at 500 step/sec. If the joystick is still on the axis after 2 seconds, the speed will increase to 10,000 step/sec.

When the joystick is released, the axis decelerates to zero speed. This gives an inch function.

CDI 5

Joystick input for counterclockwise motor rotation

If the joystick input is on, the speed is set at 500 step/sec. If the joystick is still on the axis after 2 seconds, the speed will increase to 10,000 step/sec.

When the joystick is released, the axis decelerates to zero speed. This gives an inch function.

CDI 6

Automatic/Manual mode selector.

Apply logic 0 for Manual mode of operation.
Apply logic 1 for Automatic mode of operation.

CDI 7

Axis positioning.

When Automatic mode is selected, logic 1 applied to this input positions the axis at the position set on the thumbwheel switch.

When Manual mode is selected, this input has no effect.

CDI 8

Position learning.

When Manual mode is selected, logic 1 applied to this input causes the axis position set by the thumbwheel switch to be saved in RAM (volatile memory).

When Automatic mode is selected, this input has no effect.

CDI 9

Homing request.

A homing procedure must be performed when power is applied to the CleverAx card.

Apply logic 1 to CDI 18 **Enable**

When logic 1 is applied to CDI 9, the joystick is programmed for clockwise rotation. The motor rotates clockwise until the axis reaches the point where the homing sensor causes input CDI 10 to becomes open-circuit.

At this moment, the slave index trigger is armed, and the motor reverses its rotation (counterclockwise joystick **P31.15** = 1).

The motor rotates until the marker phase of the encoder input of the CleverAx card is activated. This is set as position zero.

CDI 10

Homing sensor input.

This input must be normally closed.

CDI 11

Clockwise limit switch input.

CDI 12

Counterclockwise limit switch input.

CDI 13

Emergency stop input.

This input should be connected to a normally closed contact. When the contacts are opened, the emergency procedure is started.

This input is latching. To clear the emergency condition power must be removed from the CleverAx card.

CDI 14

Ramps selector.

Apply logic 0 to enable linear ramps.
Apply logic 1 to enable S-ramps.

CDI 15

Save parameters.

Apply logic 1 to save the positions stored in RAM to the EEPROM (non-volatile memory).

CDI 16

Not used.

CDI 17

Drive normal input.

Connect to the **Drive normal** output of the Drive.

CDI 18

CleverAx card Enable

Outputs

CDO 0

Homing output signal.

The output signal alternates between logic 0 and logic 1 during the homing procedure. When the homing procedure is finished, the output signal is at logic 1.

CDO 1

Automatic/Manual mode.

Logic 1 — Automatic mode
Logic 0 — Manual mode

CDO 2

End positioning.

Logic 1 when positioning has finished.

CDO 3

Not used.

CDO 4

Not used.

CDO 5

Drive enable.

Connect to the Drive Enable input of the Drive.

CDO 6

Saving position.

Logic 1 when position or parameters are being saved in EEPROM.

CDO 7

Emergency output.

An alternating signal appears during an emergency.

Application description

Homing

Initiate axis homing by applying logic 1 to input CDI 9.

The effects of this operation are as follows:

The motor rotates clockwise at a speed set in parameter **P26.** The Homing output signal alternates between logic 0 and logic 1.

When the axis intercepts the Homing sensor, the input signal applied to CDI 10 is logic 0 (contacts closed). The motor reverses its rotation and stops when its position corresponds with the encoder **c** phase (marker pulse).

The axis is in now in position 0. The Homing output signal is at logic 1.

Memorizing position

1. Ensure the CleverAx card is in Manual mode
2. Use the joystick to move the axis to the rquired position.
3. Set the thumbwheel switch at the number that is to be related to the position.
4. Apply logic 1 to CDI 8.
5. The output signal on CDO 6 will become logic 1 during the save process.

Axis positioning

Note

Axis positioning can be performed when linear or S-ramp is used (logic 0 applied to CDI 14 or logic 1 applied to CDI 14, respectvely).

When the required number of positions have been memorized, apply logic 1 to input CDI 6 to select Automatic mode.

Apply logic 1 to CDI 7. The axis moves to the position set on the thumbwheel switch.

Save positions and PLC program

To save the memorized positions, apply logic 1 to the **Save parameters** input CDI 15.

Load and execute the program

An MS DOS compatible development program (Cdt CleverAx development tools) is supplied with the CleverAx card. Refer to the relevant documentation.

1. At the DOS prompt, type: **CDT**.
2. Press F2.
3. Select the file **Lin_Pos.Ldg**.
4. Press the Enter key.
5. The file **Lin_Pos.Ldg** is displayed in Windows Notepad.
6. Press F6 to start the compiler.
7. At the prompt **Downloading file Y/N?** press **Y**.
8. When loading has finished press F5 followed by F4.
9. Check that parameter **P73.32** = 0 and **P32.16** = 0.
10. Set **P73.10** at 1(Autostart flag).
11. Set **P73.12** at 1 (Save Parameter).
12. Wait until both red LEDs become unlit.
13. Set **P73.28** at 1.
14. Wait until both red LEDs become unlit.
15. You may now remove power from the card. When power is next applied, the application will be automatically loaded from memory and run.

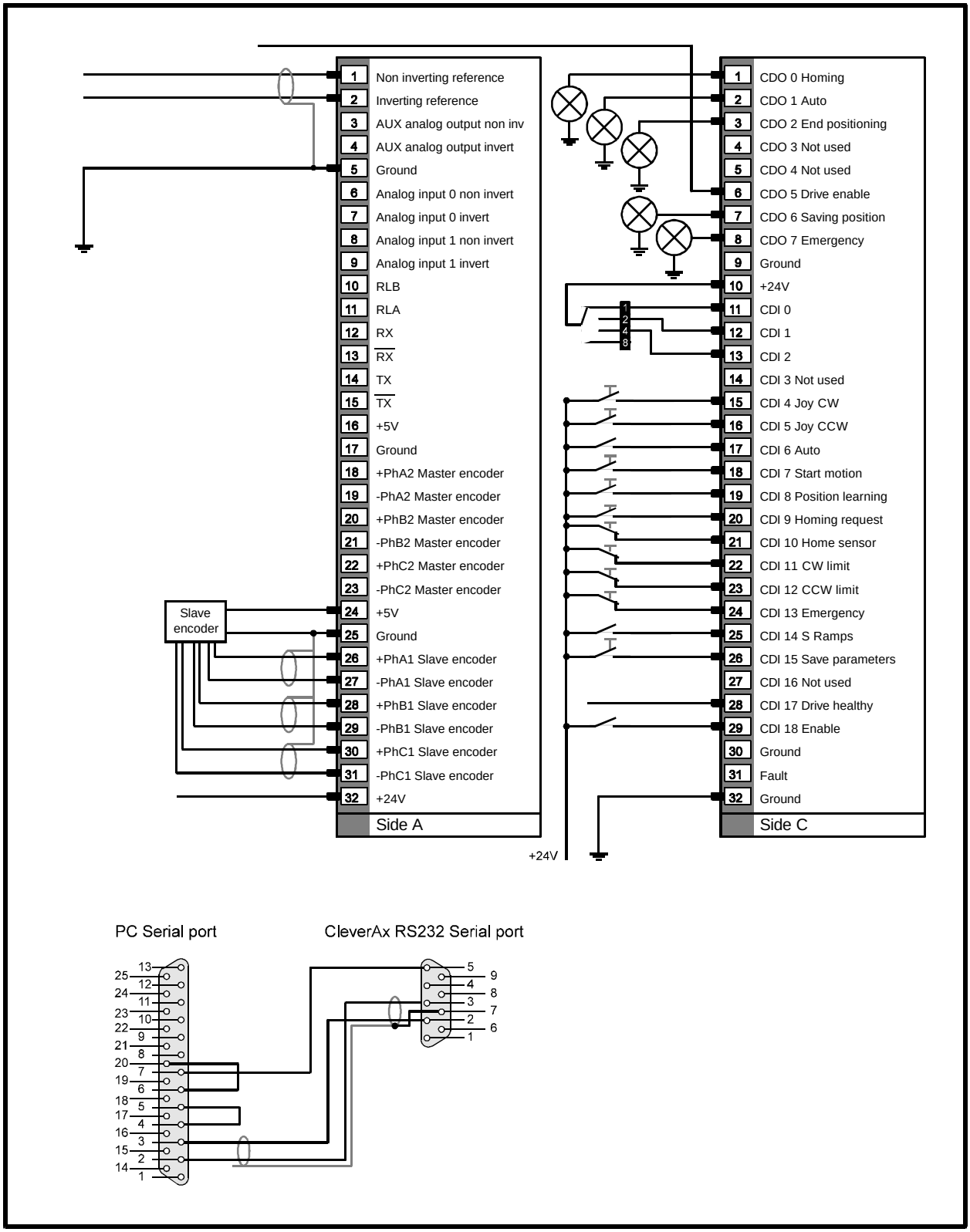
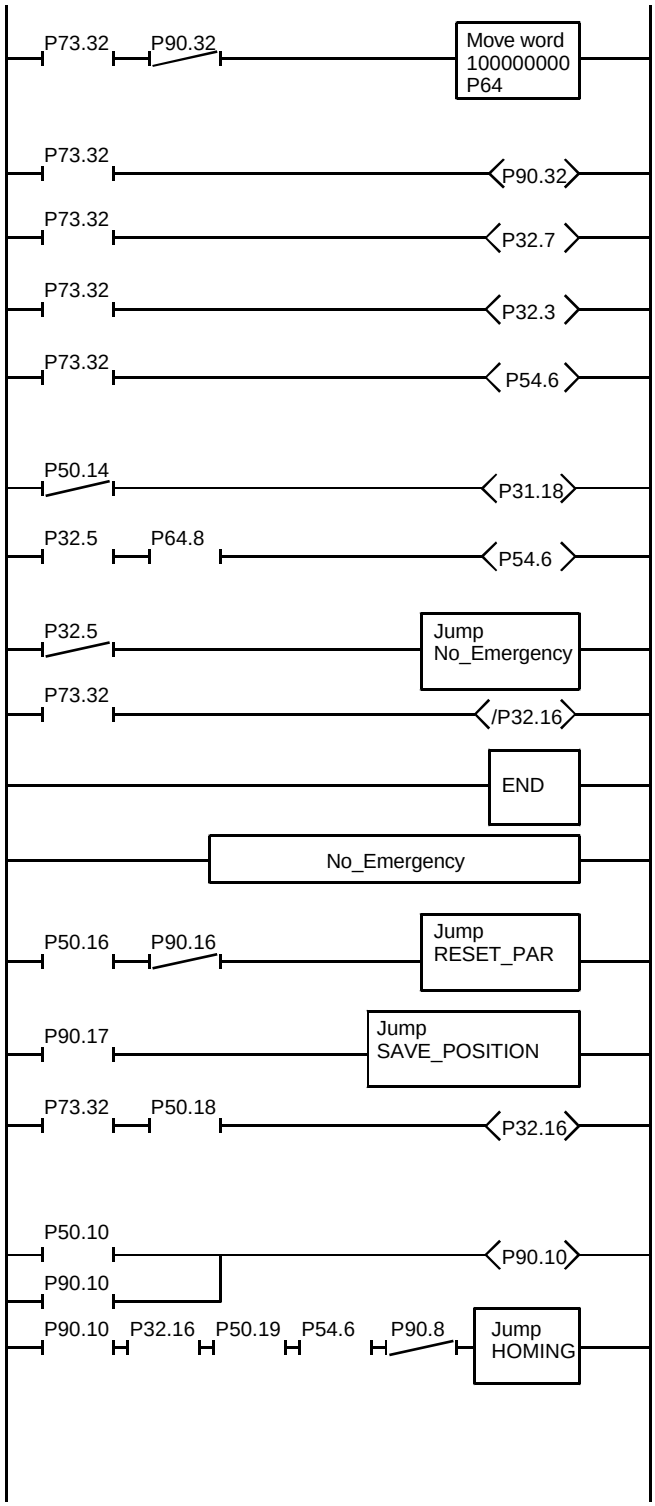
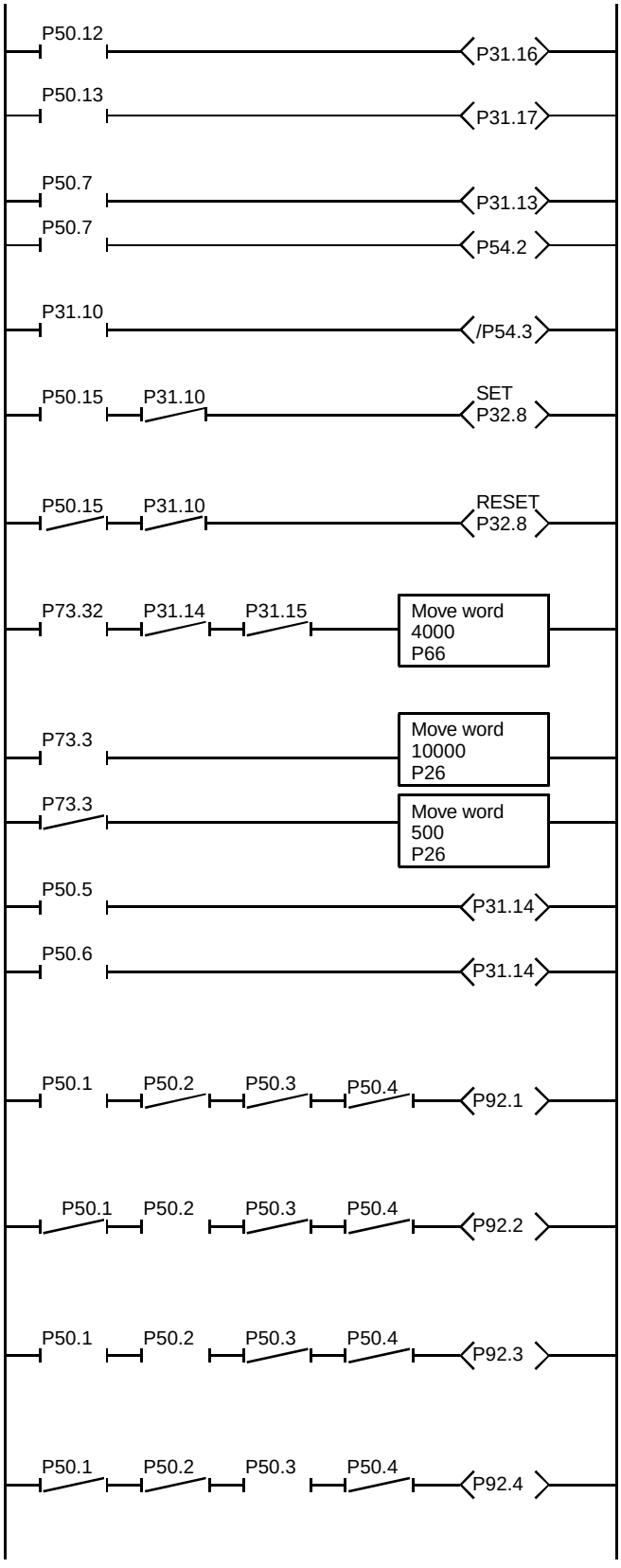


Figure 11–1 Connections to the CleverAx card for Linear Positioner with Auto Learning

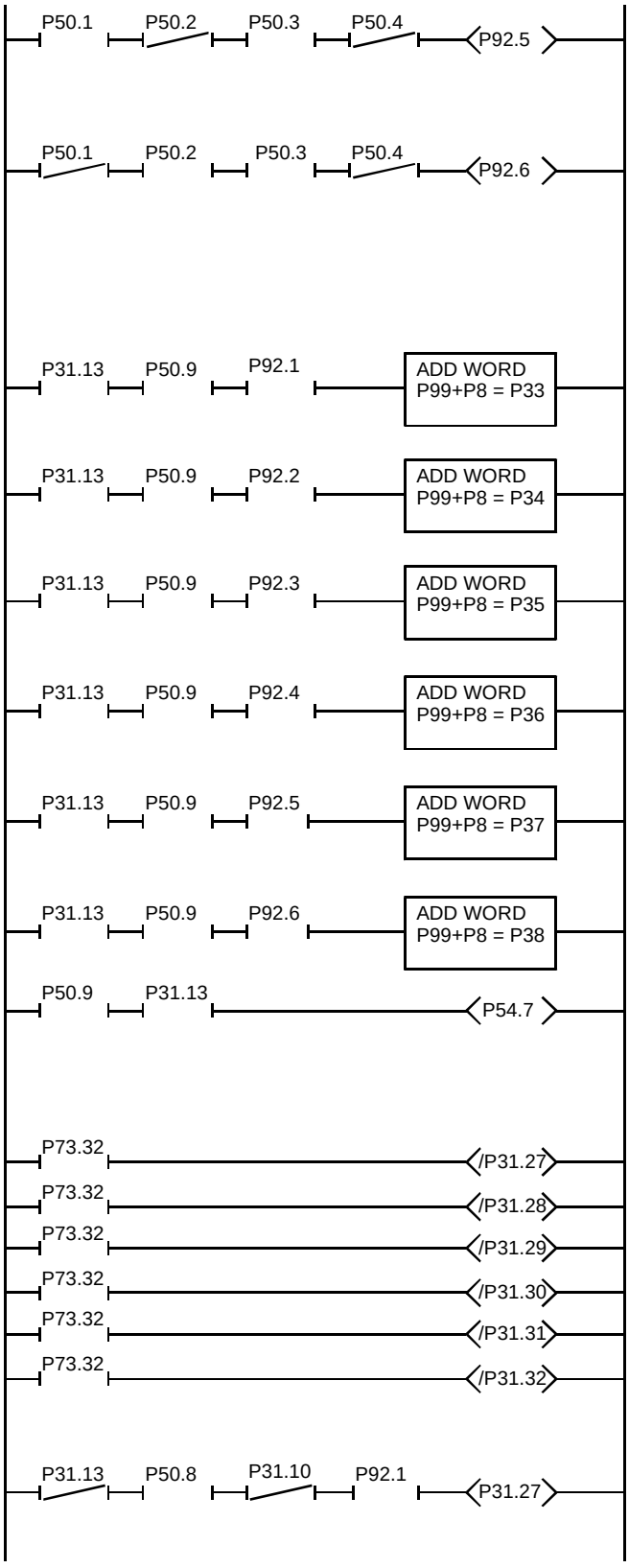
```
*Linear Positioner
*Author: Control Techniques Soprel
*Date: 20 October 1993

************************************

LD      73.32
ANDN    P90.32
MOVW P64                Set timer
100000000

LD      P73.32
OUT     P90.32

LD      P73.32
OUT     P32.7

LD      P73.32
OUT     P32.3

LD      P73.32
OUT     P54.6

************************************

LDN     P50.14          Emergency
OUT     P31.18

LDN     P32.5
AND     P64.8
OUT     P54.8

LD      P32.5
JUMP    NO EMERGENCY

LD      P73.32
OUTN    P32.16

END

| NO EMERGENCY

************************************

*SAVE POSITION PARAMETER

LD      P50.16
ANDN    P90.16
JUMP    RESET_PAR

LD      P90.17
JUMP    SAVE_POSITION

LD      P73.32          CleverAx software
AND     P50.18          enable
OUT     P32.16

************************************

LD      P50.10          Enable homing
OR      P90.10          request
OUT     P90.10

LD      P90.10          Test for homing
AND     P32.16          function and jump
AND     P50.19          homing routine
AND     P54.19
ANDN    P90.8
JUMP    HOMING

************************************
```



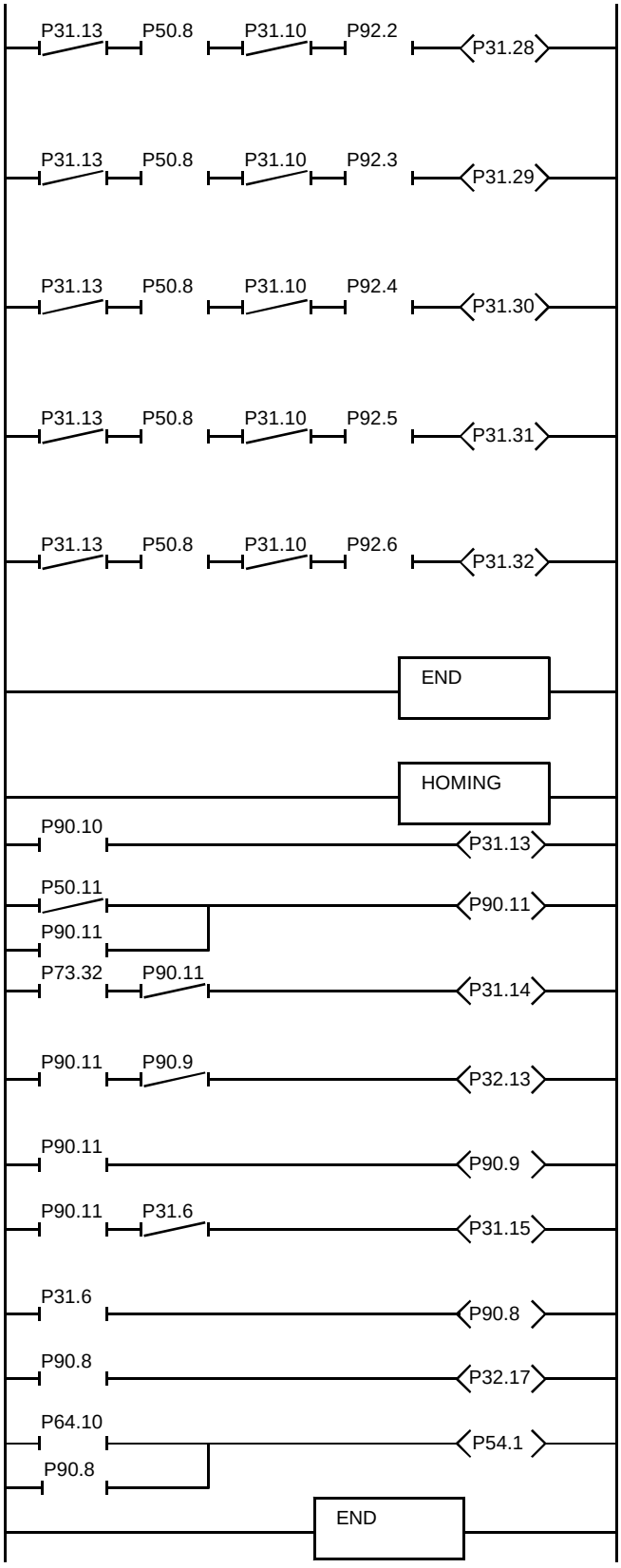
```
LDN     P50.12          Limit switch CW
OUT     P31.16

LDN     P50.13          Limit switch CCW
OUT     P31.17

************************************
LD      P50.7           Automatic/manual
OUTN    P31.13          selector
LD      P50.7
OUT     P54.2           Automatic/
                        Manual LED
************************************
LD      P31.10          End profiling
OUTN    P54.3

************************************
LD      P50.15
ANDN    P31.10          Linear or S-ramp
SET     P32.8           selector

LDN     P50.15
ANDN    P31.10
RES     P32.8

************************************
LD      P73.32
ANDN    P31.14          Auto selection
ANDN    P31.15          delay
MOVW    P66             Manual speed
4000                    in fast

LD      P73.3
MOVW    P26             Manual fast
10000                   speed

LDN     P73.3           Manual slow
MOVW    P26             speed
500

LD      P50.5           CW joy input
OUT     P31.14          control

LD      P50.6           CCW joy input
OUT     P31.15          control

************************************
*THUMBWHEEL SWITCH DECODER

LD      P50.1           Code 1
ANDN    P50.2
ANDN    P50.3
ANDN    P50.4
OUT     P92.1

LDN     P50.1           Code 2
AND     P50.2
ANDN    P50.3
ANDN    P50.4
OUT     P92.2

LD      P50.1           Code 3
AND     P50.2
ANDN    P50.3
ANDN    P50.4
OUT     P92.3

LDN     P50.1           Code 4
ANDN    P50.2
AND     P50.3
ANDN    P50.4
OUT     P92.4
```



```
LD      P50.1           Code 5
ANDN    P50.2
AND     P50.3
ANDN    P50.4
OUT     P92.5

LDN     P50.1           Code 6
AND     P50.2
AND     P50.3
ANDN    P50.4
OUT     P92.5

**********
*AUTO LEARNING CONTROL
*ONLY IN MANUAL MODE

LD      P31.13          POSITION 1
AND     P50.9           Auto learning
AND     P92.1
ADDW    P33, P99, P8

LD      P31.13          POSITION 2
AND     P50.9           Auto learning
AND     P92.2
ADDW    P34, P99, P8

LD      P31.13          POSITION 3
AND     P50.9           Auto learning
AND     P92.3
ADDW    P35, P99, P8

LD      P31.13          POSITION 4
AND     P50.9           Auto learning
AND     P92.4
ADDW    P36, P99, P8

LD      P31.13          POSITION 5
AND     P50.9           Auto learning
AND     P92.5
ADDW    P37, P99, P8

LD      P31.13          POSITION 6
AND     P50.9           Auto learning
AND     P92.6
ADDW    P38, P99, P8

LD      P50.9           Position saving
AND     P31.13          display
OUT     P54.7

**********
*RESET MULTI VALUE SELECTOR  INDEX

LD      P73.32
OUTN    P31.27          <P31.27>
LD      P73.32
OUTN    P31.28          <P31.28>
LD      P73.32
OUTN    P31.29          <P31.29>
LD      P73.32
OUTN    P31.30          <P31.30>
LD      P73.32
OUTN    P31.31          <P31.31>
LD      P73.32
OUTN    P31.32          <P31.32>

LDN     P31.13          Positioner at
AND     P50.8           POSITION 1 <P33>
ANDN    P31.10
AND     P92.1
OUT     P31.27
```

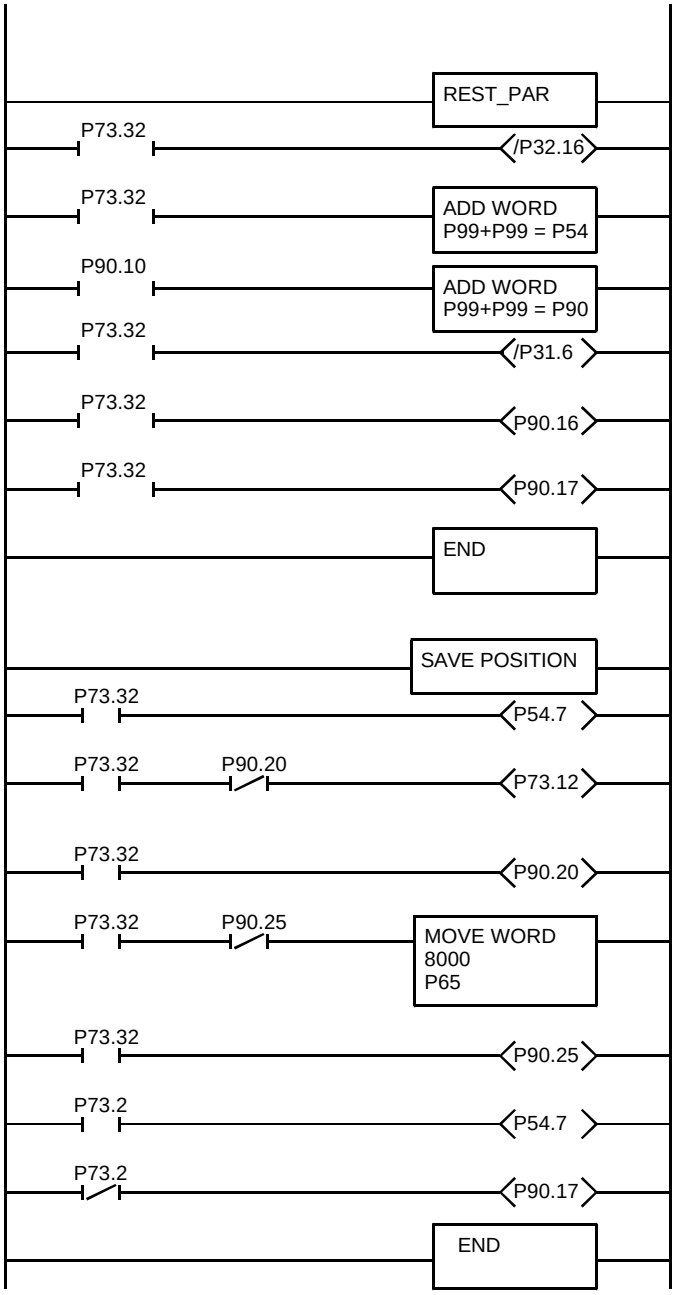


```
LDN     P31.13          Positioner at
AND     P50.8           POSITION 2  <P34>
ANDN    P31.10
AND     P92.2
OUTP    P31.28

LND     P31.13          Positioner at
AND     P50.8           POSITION 3  <P35>
ANDN    P31.10
AND     P92.3
OUT     P31.29

LDN     P31.13          Positioner at
AND     P50.8           POSITION 4  <P36>
ANDN    P31.10
AND     P92.4
OUT     P31.30

LDN     P31.13          Positioner at
AND     P50.8           POSITION 5  <P37>
ANDN    P31.10
AND     P92.5
OUT     P31.31

LDN     P31.13          Positioner at
AND     P50.8           POSITION 6  <P38>
ANDN    P31.10
AND     P92.6
OUT     P31.32

END

**********************************
*HOMING PROCEDURE

| HOMING

LD      P90.10          FORCED MANUAL
OUT     P31.13          MODE

LDN     P50.11          HOME SENSOR
OR      P90.11          LATCH
OUT     P90.11

LD      P73.32          CE JOY. CONTROL
ANDN    P90.11
OUT     P31.14

LD      P90.11          SLAVE INDEX
ANDN    P90.9           TRIGGER ARMED
OUT     P32.13

LD      P90.11
OUT     P90.9

LD      P90.11          CCW JOY. CONTROL
ANDN    P31.6
OUT     P31.15

LD      P31.6           END HOMING
OUT     P90.8

LD      P90.8           ORIGIN P8 = 0
OUT     P32.17

LD      P64.10          HOMING TRIP LED
OR      P90.8
OUT     P54.1

END

**********************************
```



```
*PARAMETER RESET

| RESET_PAR

LD      P73.32
OUTN    P32.16

LD      P73.32
ADDW    P54, P99, P99

LD      P73.32
ADDW    P90, P99, P99

LD      P73.32
OUTN    P31.6

LD      P73.32
OUT     P90.16

LD      P73.32
OUT     P90.17

END

************************************
*POSITIONS SAVING PROCEDURE

| SAVE_POSITION

LD      P73.32
OUT     P54.7

LD      P73.32
ANDN    P90.20
OUT     P73.12

LD      P73.32
OUT     P90.20

LD      P73.32
ANDN    P90.25
MOVW    P65
8000

LD      P73.32
OUT     P90.25

LD      P73.2
OUT     P54.7

LDN     P73.2
OUTN    P90.17

END

************************************
```

Programming



11.2 Digital lock

This is a digital lock application with programmable ratio between master and slave. Features are as follows:

Ratio is programmable by a thumbwheel switch or external PLC.

Lock

Unlock

Two different lock modes (with or without phasing recover)

Linear or S-ramp during the phasing lock

Inputs

CDI 0
CDI 1
CDI 2
CDI 3

Inputs connected to thumbwheel switches 4 and 5 (5 is the least significant digit).

CDI 4
CDI 5
CDI 6
CDI 7

Inputs connected to thumbwheel switches 2 and 3.

CDI 8
CDI 9

Inputs connected to thumbwheel switch 1 (most significant digit).

CDI 10

Digital lock enable.

CDI 11

Line speed selection.

Selects the axis speed when digital lock is disabled. When logic 0 is applied, the speed is zero. When logic 1 is applied, the speed is that set in parameter **P83**.

CDI 12

Increase position offset.

When logic 1 is applied, an offset will be added to the actual slave speed. This increments position offset between the master and slave axes.

CDI 13

Decrease position offset.

When logic 1 is applied, an offset will be subtracted to the actual slave speed. This decrements position offset between the master and slave axes.

CDI 14

Ramps selector.

Apply logic 1 to select linear ramps.
Apply logic 0 to select S-ramps.

CDI 15

Connect this input to output CDO 0.

CDI 16

Connect this input to output CDO 1.

CDI 17

Enables space auto-phase when P32.27 = 0. Lost space is recovered during phase lock.

CDI 18

CleverAx card enable.

Outputs

CDO 0

Thumbwheel switch supply (digit 3 and digit 5).

Applies the supply to digit 3 and digit 5 of the thumbwheel switch when the data is ready to be received.

CDO 1

Thumbwheel switch supply (digit 1 and digit 4).

Applies the supply to digit 2 and digit 4 of the thumbwheel switch when the data is ready to be received.

CDO 2

Supplies digit 1 of the thumbwheel switch during digit scan.

CDO 3

Not used.

CDO 4

Not used.

CDO 5

Drive enable output

CDO 6

Logic 1 when the slave axis is in phase with the master axis.

CDO 7

Following error output (set in parameter **P17**).

Application description

When logic 1 is applied to input CDI 10 (Digital lock enable), digital lock occurs at the ratio set on the thumbwheel switches. (The first digit is the integer and digits 2, 3, 4 and 5 are the decimal places.)

The slave axis can start from zero speed (logic 0 applied to input CDI 11), or from a speed set in parameter **P83** (logic 1 applied to input CDI 11).

The slave axis reaches the master axis by following the selected ramp (linear ramp when logic 0 is applied to CDI 14, or S-ramps when logic 1 is applied to CDI 14).

When logic 0 is applied to input CD17, the slave will become phase-locked with the master axis, and will subsequently recover the lost space.

Position offset can be applied by applying logic 1 to input CDI 12 (increment) or to input CDI 13 (decrement).

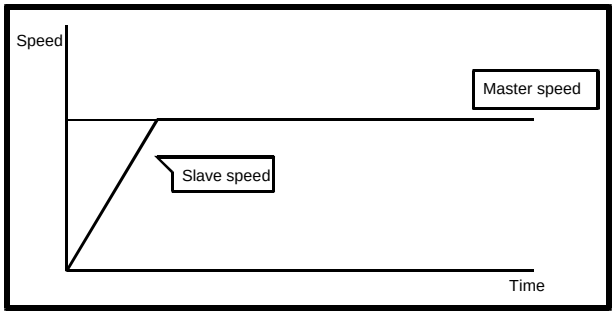


Figure 11–2 Digital lock with linear ramps — phase not locked

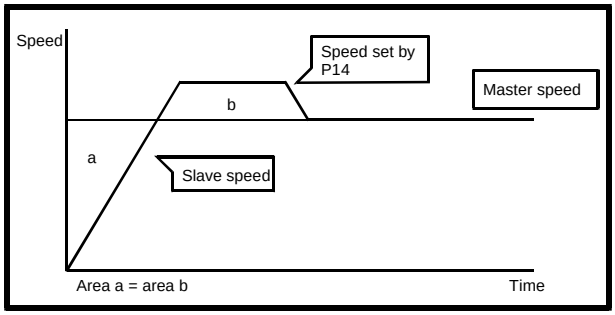


Figure 11–3 Digital lock with linear ramps — phase locked

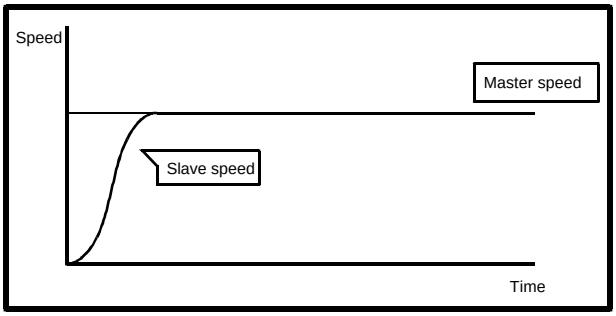


Figure 11–4 Digital lock with S-ramp — phase not locked

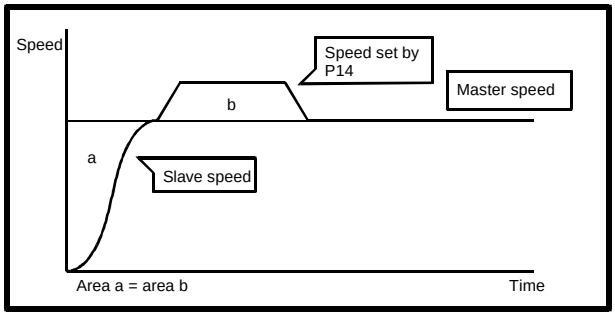


Figure 11–5 Digital lock with S-ramp — phase locked

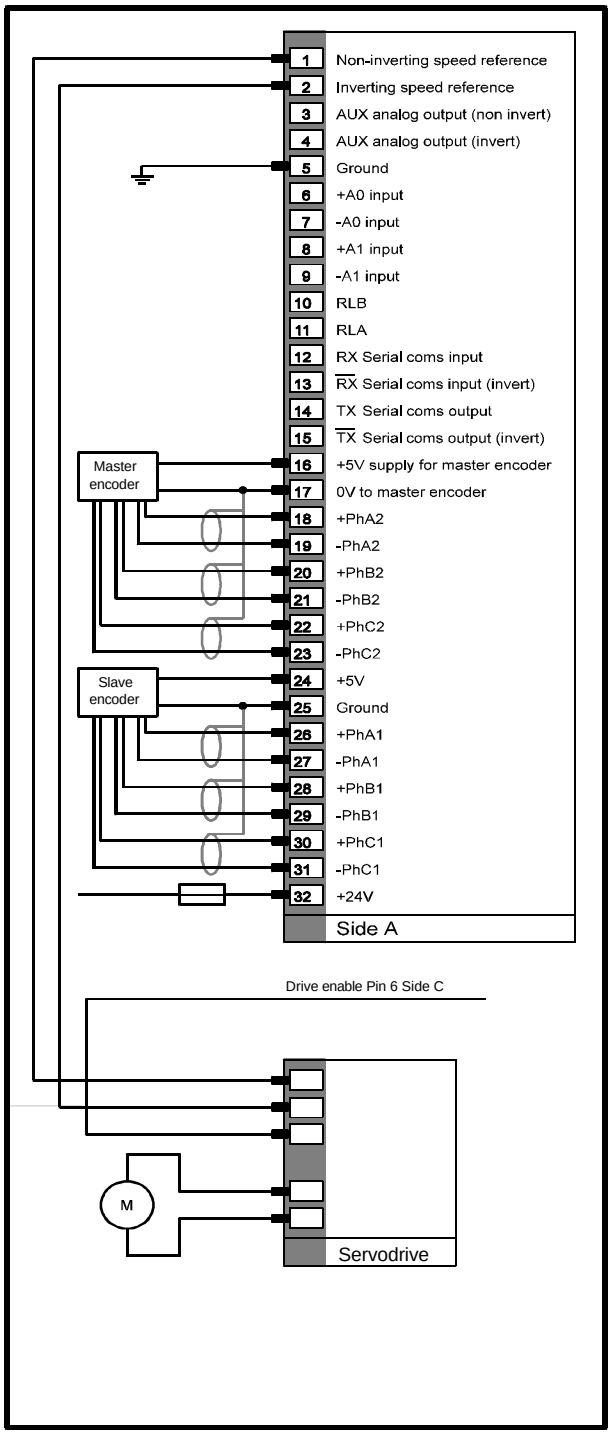


Figure 11–6 Side A connections to the CleverAx card for Digital Lock

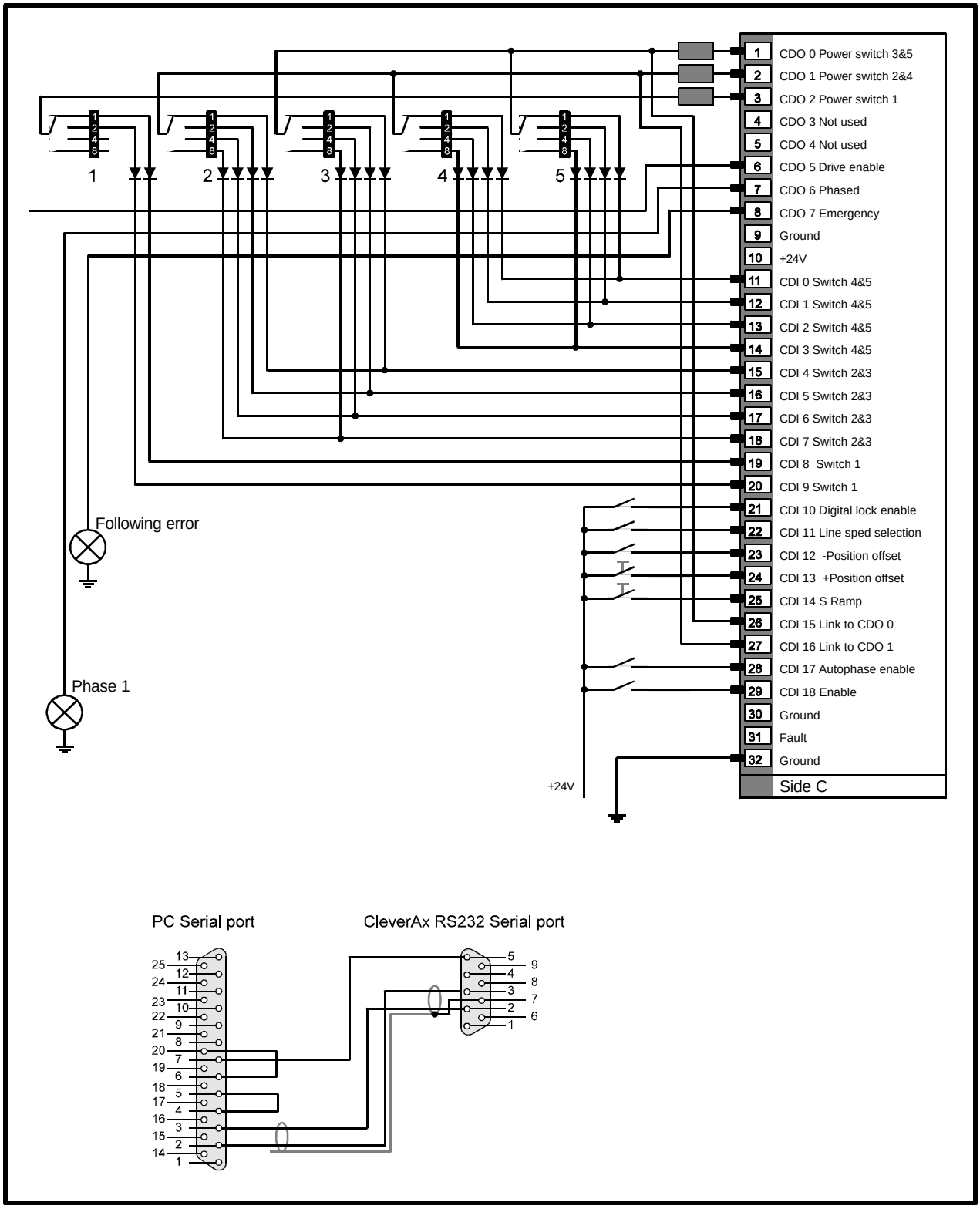
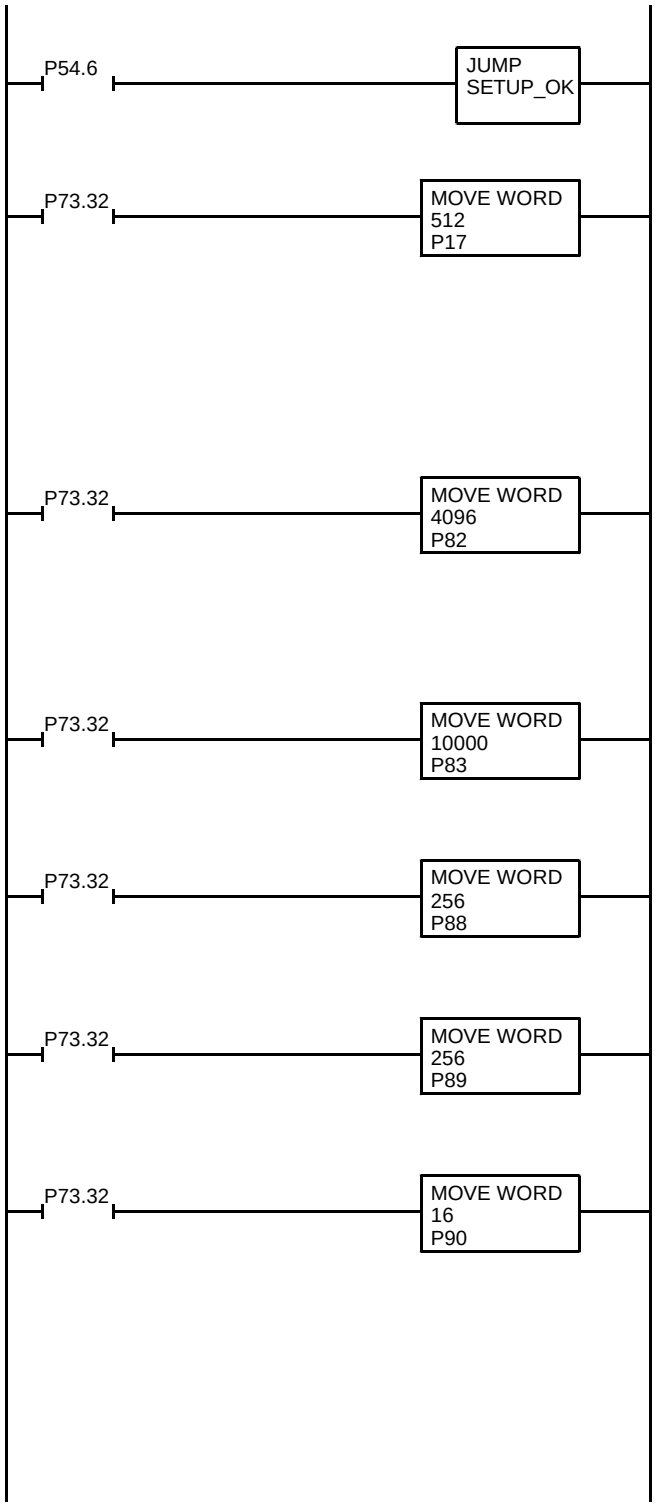


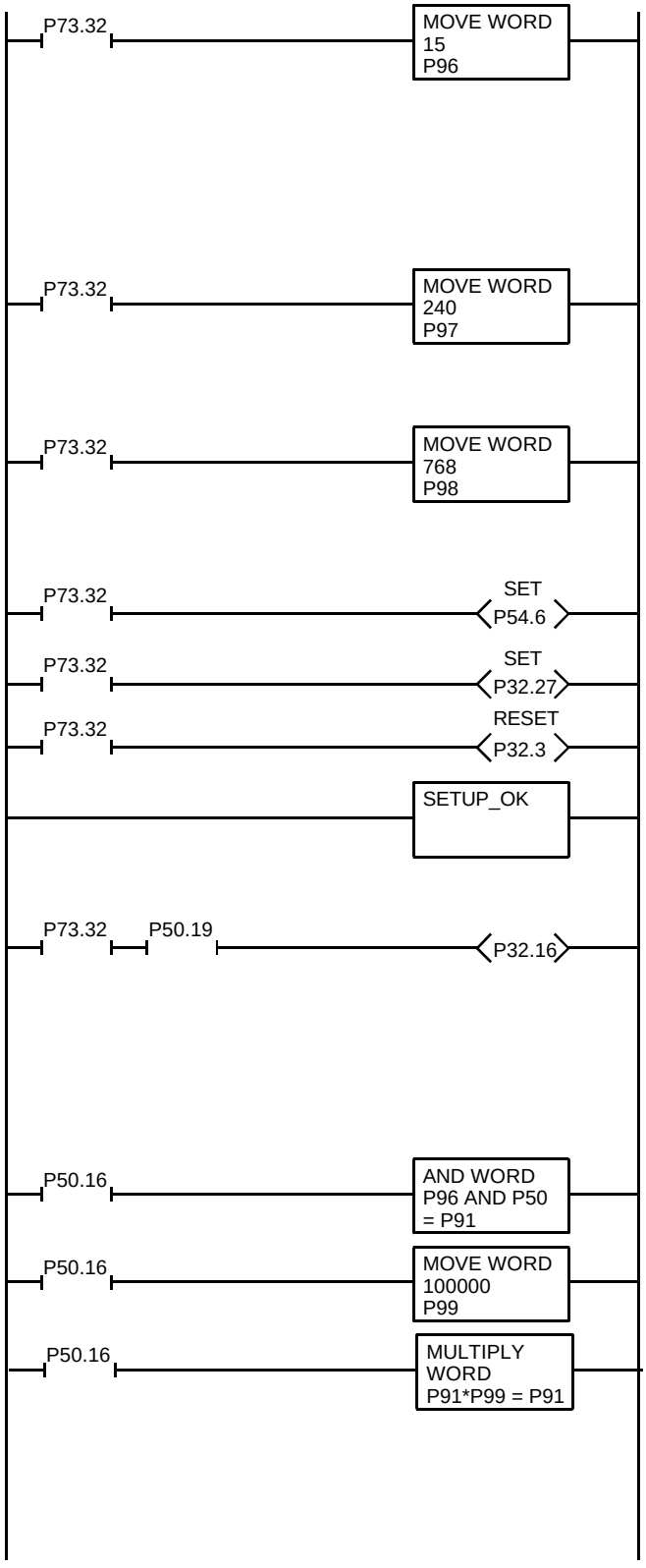
Figure 11–7 Side C connections to the CleverAx card for Digital Lock

Digital lock
Author: Control Techniques Soprel
Date: 5 November 1993

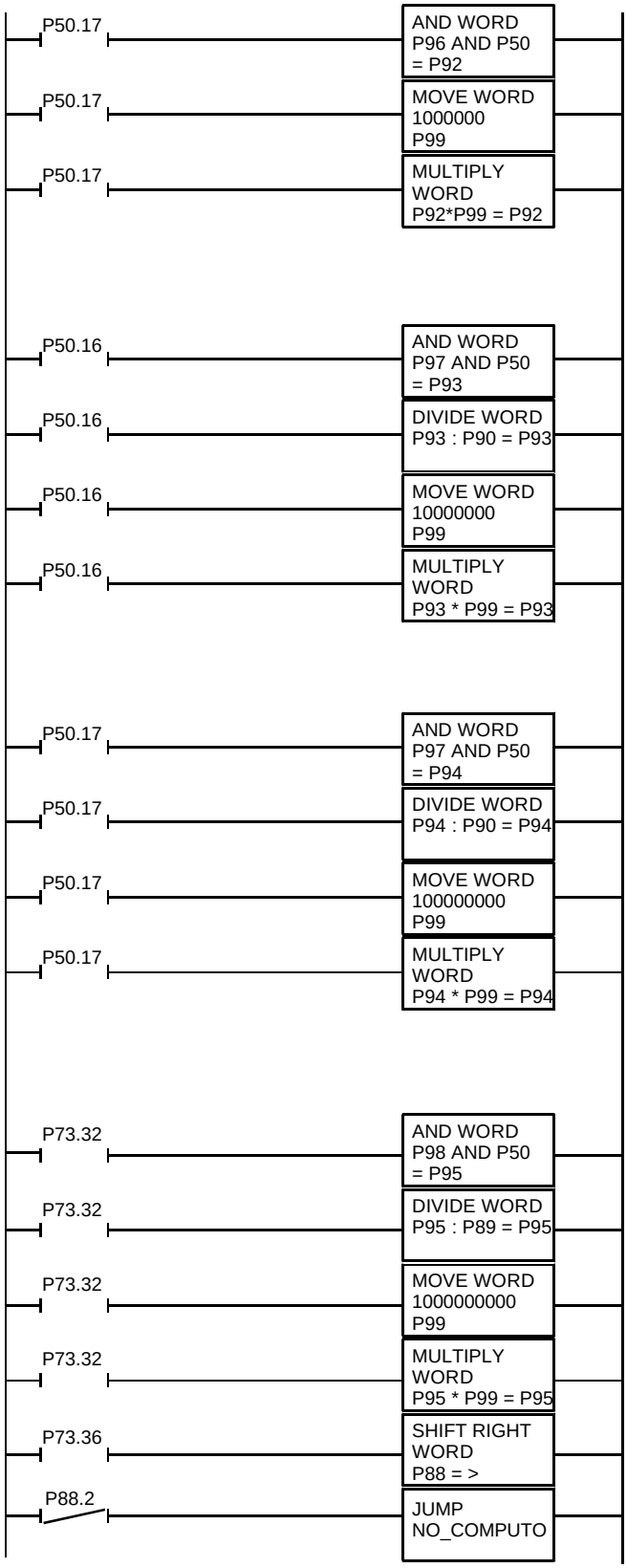
```
************************************
*SETUP
************************************
LD      P54.6
JUMP    SETUP_OK

************************************
*SET MAXIMUM DELTA ERROR
************************************
LD      P73.32
MOVW P17
512

************************************
*
*LOAD 4096 IN P82
*
*SPEED FOR SPACE OFFSET
*
*CONTROLLER STEP / SEC.
*
************************************
LD      P73.32
MOVW P82
4096
************************************
*
*LOAD 10000 IN P83
*
*BASE SPEED STEPS / SEC.
*
************************************
LD      P73.32
MOVW P83
10000

************************************
*LOAD 256 IN P88
************************************
LD      P73.32
MOVW P88
256

************************************
*LOAD 256 IN P89
************************************
LD      P73.32
MOVW P89
256

************************************
*LOAD 16 IN P90
************************************
LD      P73.32
MOVW P90
16

************************************
*
*LOAD 15 IN P96
*
*MASK INPUT
*
*D10, D11, D12, D13
*
************************************
```



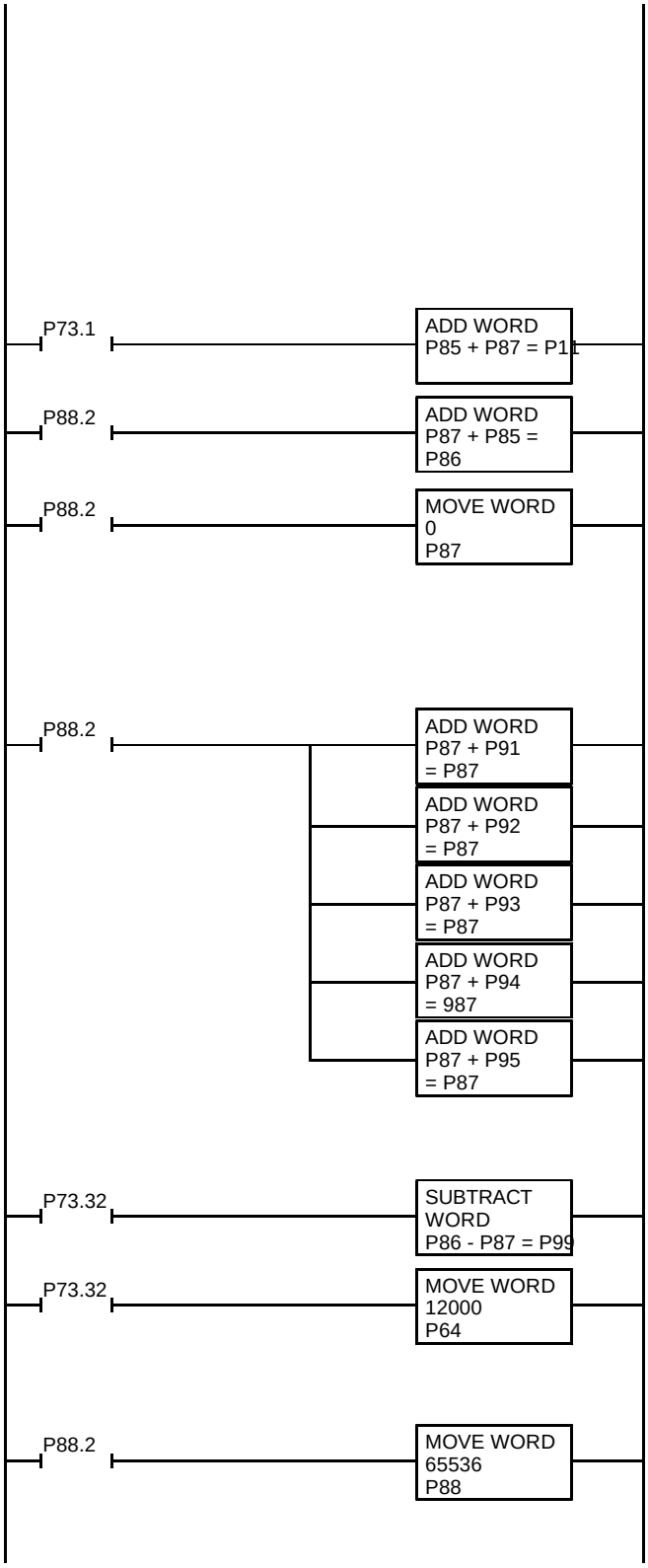
```
LD      73.32
MOVW P96
15
**********************************
*
*LOAD 240 IN P97
*
*MASK INPUT
*
*D14, D15, D16, D17
*
**********************************
LD      73.32
MOVW P97
240

**********************************
*LOAD 768 IN P98
**********************************
LD      73.32
MOVW P98
768

**********************************
*SET SWITCH BIT
**********************************
LD      P73.32
SET     P54.6

LD      P73.32
SET     P32.27

LD      P73.32
RES     P32.3

|SETUP_OK
**********************************
*
*CleverAx OK
*
**********************************
LD      P73.32
AND     P50.19
OUT     P32.16
**********************************
*THUMBWHEEL SWITCH DECODER
*
*READING FIRST NUMBER
*P91 = (P50 AND P96)
*P99 = 100000
*P91 = P91*P99
**********************************
LD      P50.16
ANDW  P91, P96, P50

LD      P50.16
MOVW P99
100000

LD      P50.16
MULW  P91, P91, P99

**********************************
*READING SECOND NUMBER
*
*P92 = (P50 AND P96)
*
*P99 = 1000000
*P92 = P92*P99
**********************************
```



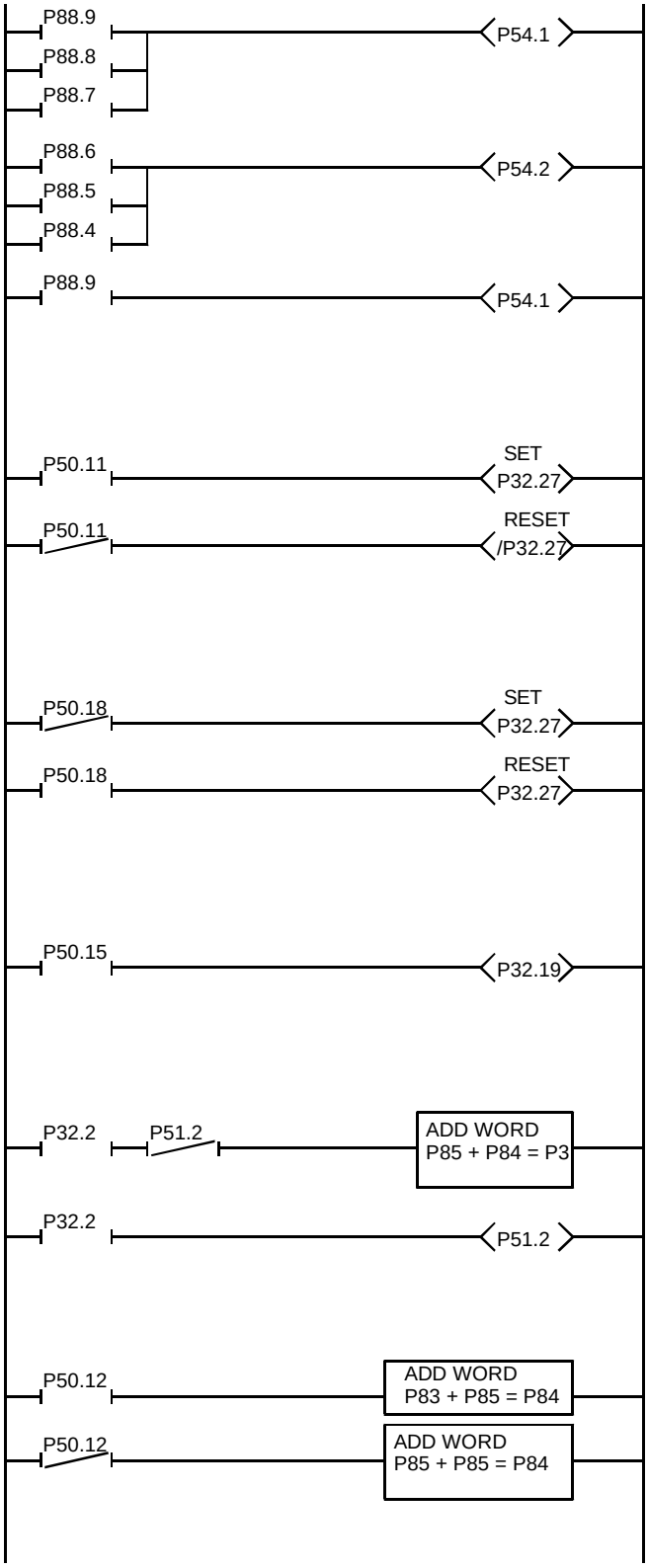
```
LD       P50.17
ANDW  P92, P96, P50

LD       P50.17
MOVW P99
1000000

LD       P50.17
MULW  P92, P92, P99
***********************************
*READING THIRD NUMBER
*P93 = (P50 AND 097) >>4
*P99 = 10000000
*P93 = P93*P99
***********************************
LD       P50.16
ANDW  P93, P97, P50

LD       P50.16
DIVW   P93, P93, P90

LD       P50.16
MOVW P99
10000000

LD       P50.16
MULW  P93, P93, P99
***********************************
*READING FOURTH NUMBER
*P94 = (P50 AND P97) >>4
*P99 = 100000000
*P94 = P94*P99
***********************************
LD       P50.17
ANDW  P94, P97, P50

LD       P50.17
DIVW   P94, P94, P90

LD       P50.17
MOVW P99
100000000

LD       P50.17
MULW  P94, P94, P99
***********************************
*READING FIFTH NUMBER
*P95 = (P50 AND P98) >>8
*P99 = 1000000000
*P95 = P95*P99
***********************************
LD       P73.32
ANDW  P95, P98, P50

LD       P73.32
DIVW   P95, P95, P89

LD       P73.32
MOVW P99
1000000000

LD       P73.32
MULW  P95, P95, P99

LD       P73.32
RSHW  P88, P88

LDN      P88.2
JUMP   NO_COMPUTO
```



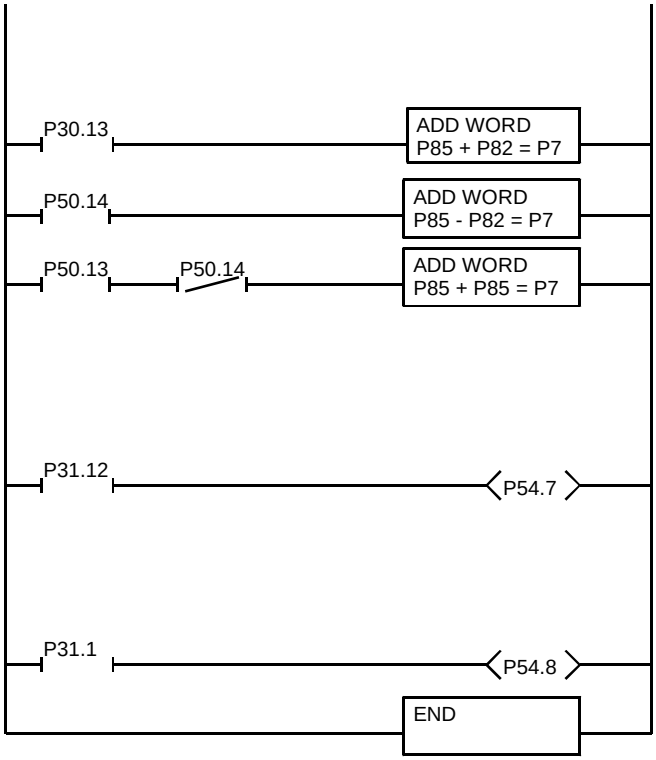
```
**********************************
*COMPUTING RATIO
*IF (TIMER P64 = 0)
*{
*P11 = P87
*}
*P86 = P87 P86 IS P87 (T-1)
*P87 = 0
*P87 = P87+P91
*P87 = P87+P92
*P87 = P87+P93
*P87 = P87+P94
*P87 = P87+P95
**********************************
LD      P73.1
ANDW  P11, P85, P87


LD      P88.2
ADDW  P86, P87, P85


LD      P88.2
MOVW P87
0

LD      P88.2
LD      P88.2
LD      P88.2
LD      P88.2
LD      P88.2

ADDW  P87, P87, P91


ADDW  P87, P87, P92


ADDW  P87, P87, P93


ADDW  P87, P87, P94


ADDW  P87, P87, P95
**********************************
*IF (P86<>P87)
*{
*P64 = 12000
*}
**********************************
LD      P73.32
SUBW  P99, P86, P87


LDN     P73.16
MOVW P64
12000

**********************************
NO_COMPUTO
**********************************
LD      P88.2
MOVW P88
65536
```



```
LD      P88.9           Multiplexing
OR      P88.8           output to enable
OR      P88.7           thumbwheel switch
OUT     P54.1


LD      P88.6
OR      P88.5
OR      P88.4
OUT     P54.2


LD      P73.32
OUT     P54.3

************************************
*
*Enable/disable D. lock
*
************************************
LD      P50.11
SET     P32.2

LDN     P50.11
RES     P32.2

************************************
*
*Enable space auto phase
*
************************************
LDN     P50.18
SET     P32.27

LD      P50.18
RES     P32.27

************************************
*
*Enable S ramp
*
************************************
LD      P50.15
OUT     P32.19

************************************
*
*Slave speed after D.lock disable
*
************************************
LDN     P32.2
ANDN    P51.2
ADDW    P3, P85, P84

LD      P32.2
OUT     P51.2
************************************
*
*Slave speed selector
*
************************************
LD      P50.12
ADDW    P84, P83, P85

LDN     P50.12
ADDW    P84, P85, P85
```



```
***********************************
*
*Slave SPACE OFFSET
*
***********************************
LD        P50.13
ADDW    P7, P85, P82

LD        P50.14
SUBW    P7, P85, P82

LDN       P50.13
ANDN     P50.14
ADDW    P7, P85, P85

***********************************
*
*Digital lock phased
*
***********************************
LD        P31.12
OUT      P54.7

***********************************
*
*Maximum following error
*
***********************************
LD        P31.1
OUT      P54.8

END
```



11.3 Digital lock using the serial link

This application produces digital lock between two DigitAx Drives.

The management of the information is obtained using a serial link between the CleverAx card and the two DigitAx Drives.

This program is an example of programming and management of the serial link. The program can be used in certain applications.

Inputs

CDI 0

Apply logic 0 to enable digital lock.

The CleverAx card sends the speed read from the master Drive to the slave Drive. The programmed ratio is applied.

Outputs

CDO 0

Sending data to the slave Drive.

This output is at logic 1 when the CleverAx card is sending data to the slave Drive.

CDO 1

Receiving data from the master Drive.

This output is at logic 1 when the CleverAx card is receiving data from the master Drive.

CDO 2

This output goes to logic 1 when a time-out error occurs when sending data to the slave Drive.

Time-out error occurs when data transmission is longer then 300msec.

CDO 3

This output goes to logic 1 when a time-out error occurs when receiving data to the master Drive.

Time-out error occurs when data transmission is longer than 300msec.

Application description

Serial link

CleverAx has two serial communications channels as follows:

RS232 interface

Used for programming, writing application programs and communicating with a keypad or microterminal.

RS485

Standard serial link used in industrial applications.
A high-level translator in the CleverAx software ensures that the protocol management is transparent to the user.

Application

Note

Two DigitAx Drives are used in this application. With a small amount of modification, Vector Drives may be used.

In the DigitAx Drive, the motor speed is set in parameter Pr59.

In the Vector Drive, the motor speed is set in parameter Pr70.

Use the following procedure:

1. Load **Pr64** (Down-timer counter) of the Drive with a value of 500 cycles. (This gives a duration of 250msec.)
2. Set the master Drive address (2) at **H4A = 30303232**
(where **H4A** is **Pr74** in hexadecimal)
3. Set parameter **Pr75** at 59 in order to read the motor speed.
4. Set parameter **Pr78** at 4 to start reading the motor speed.
5. The program now performs a scan of the values of **Pr78** and **Pr64**. If the data reception is complete (**Pr78** = 6) before the time limit set in **Pr64**, output CD0 3 will remain at logic 0. If the data is not received in this time, output CD0 3 will go to logic 1.
6. When input CDI 0 is at logic 1, no other operation will occur. When input CDI 0 is at logic 0, the program computes the speed (**Pr76** × ratio) to send to the slave Drive.
7. Load **Pr64** (Down-timer counter) of the Drive with a value of 500 cycles. (This gives a duration of 250msec.)
8. Set the Slave Drive address (2) at **H4A = 30303131**
(where **H4A** is **Pr74** in hexadecimal)
9. Set parameter **Pr75** at 0 in order to send the motor speed to the slave Drive.
10. Set parameter **Pr78** at 1 (Start reading).
11. The program now performs a scan of the values of **Pr78** and **Pr64**. If the data transmission is complete (**Pr78** = 4) before the time limit set in **Pr64**, output CD0 2 will remain at logic 0. If the data is not transmitted in this time, output CD0 2 will go to logic 1.
12. Repeat the program at step 1.

Load and execute the program

An MS DOS compatible development program (Cdt CleverAx development tools) is supplied with the CleverAx card. Refer to the relevant documentation.

1. At the DOS prompt, type: **CDT**.
2. Press F2.
3. Select the file**LINK.LDG**.
4. Press the Enter key.
5. The file **LINK.LDG** is displayed in Windows Notepad.
6. Press F6 to start the compiler.
7. At the prompt **Downloading file Y/N?** press **Y**.
8. When loading has finished press F5 followed by F4.
9. Check that **P73.32** = 0 and **P32.16** = 0.
10. Set **P73.10** at 1 (Autostart flag).
11. Set **P73.12** at 1 (Save parameters).
12. Wait until both red LEDs become unlit.
13. Set **P73.28** at 1 (Save program).
14. Wait until both red LEDs become unlit.
15. You may now remove power from the card. When power is next applied, the application will be automatically loaded from memory and run.

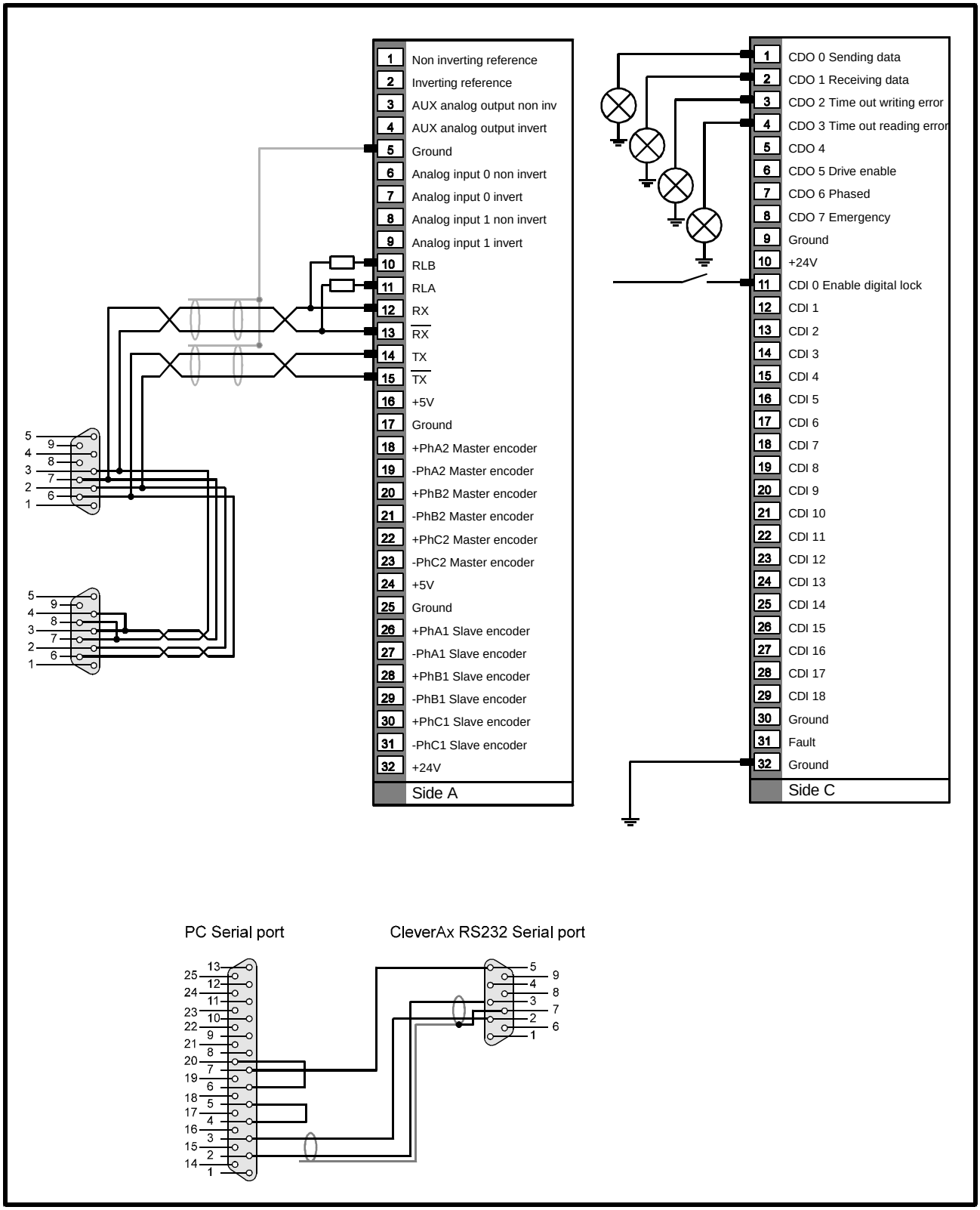
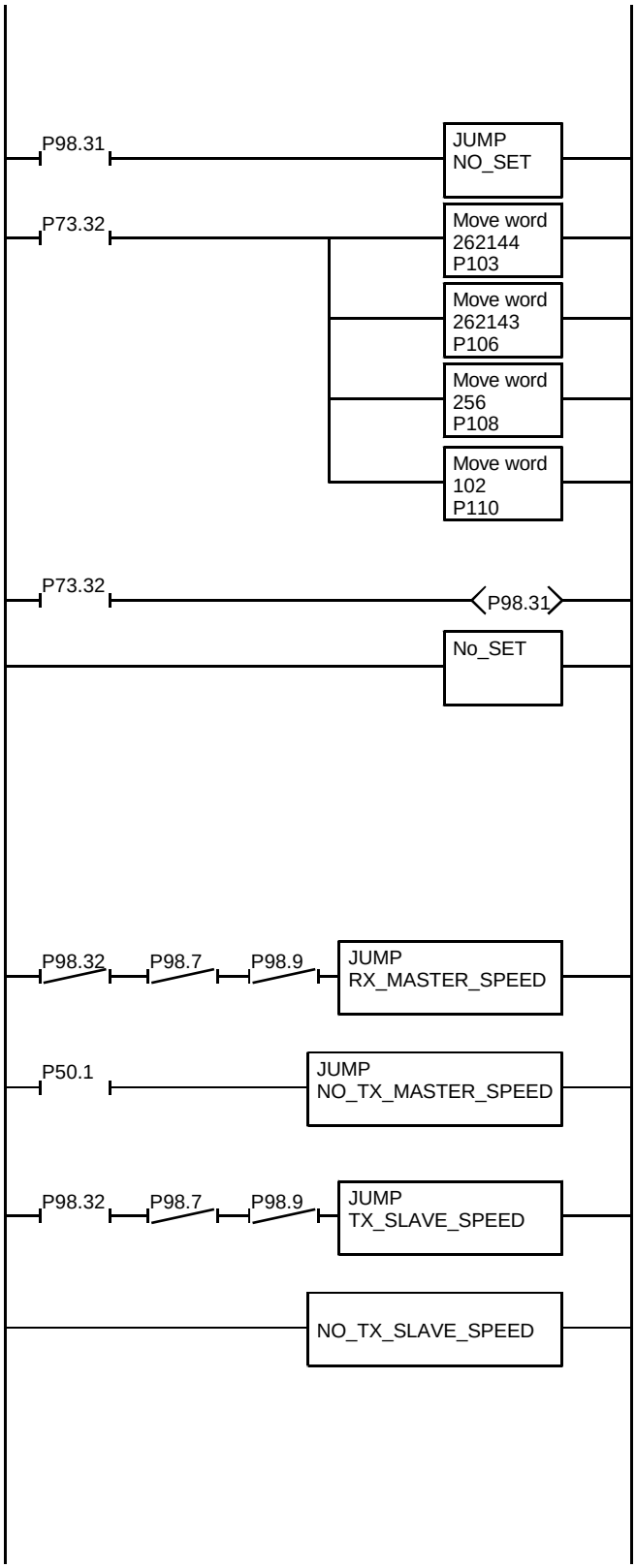
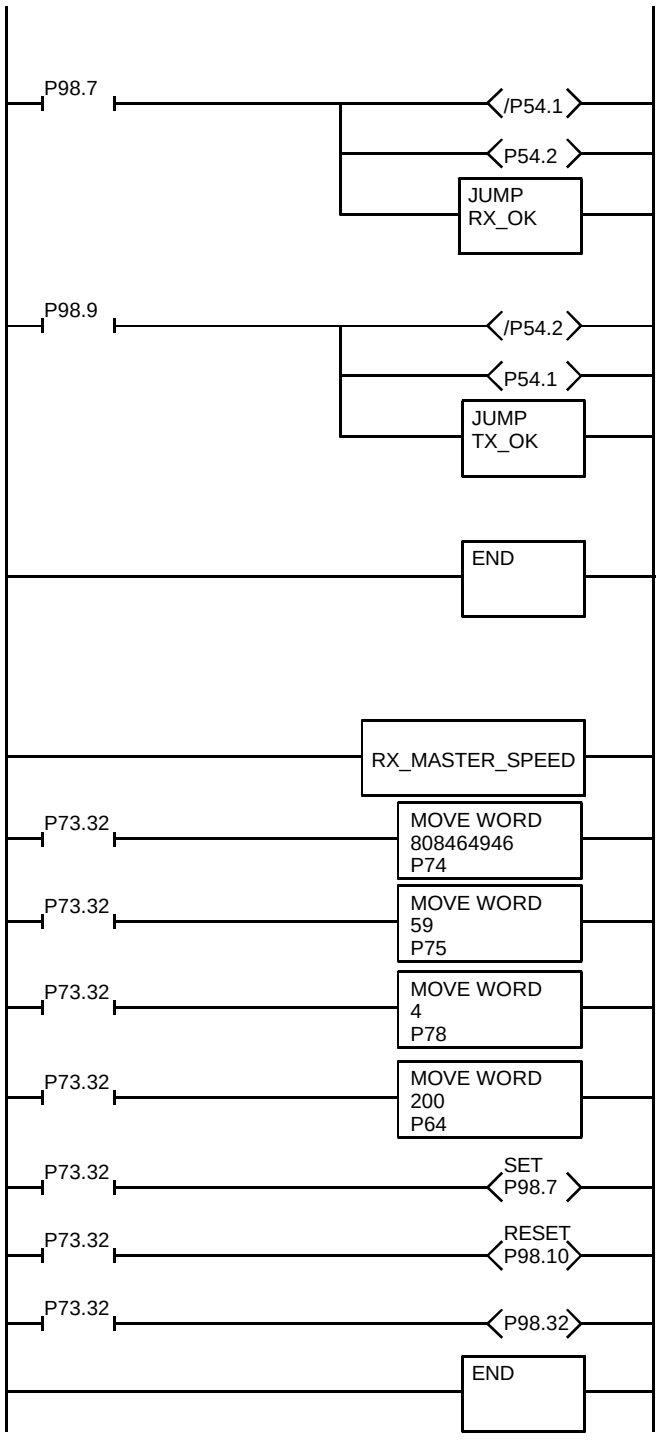


Figure 11–8 Connections to the CleverAx card for digital lock using the serial link

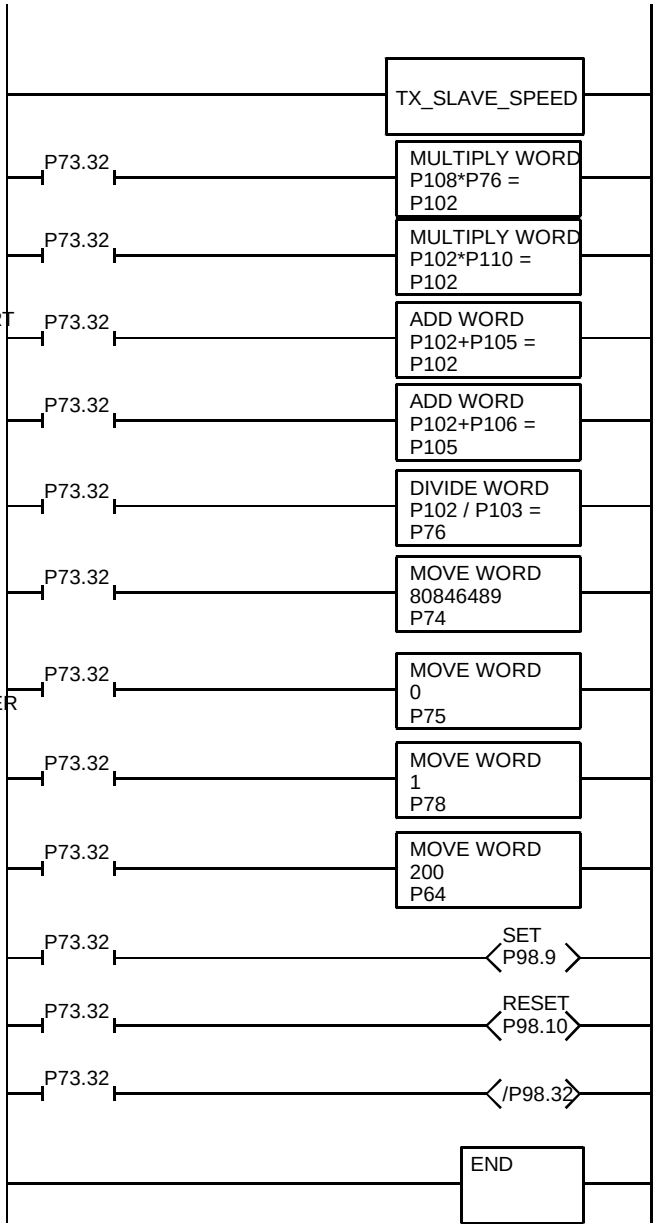
```
**********************************
*Digital speed lock using the serial link
*Author: Control Techniques Soprel
*Date: 15 December 1993
**********************************

LD      P98.31
JUMP NO_SET

LD      P73.32
LD      P73.32
LD      P73.32
LD      P73.32

MOVW P103           MASK RIGHT SHIFT 18
262144

MOVW P106           MASK HEX 3FFFF
262143

MOVW P108           MASK LEFT SHIFT 8
256

MOVW P110           RATIO INT(RATIO*1024)
102                 0.1*1024 = 102

LD      P73.32
OUT     P98.31

|NO_SET

**********************************
*
*MAIN LOOP PROGRAM
*
*
**********************************
**********************************
*
*READ MASTER SPEED
*
*
**********************************
LDN     P98.32
ANDN  P98.7
ANDN  P98.9
JUMP RX_MASTER_SPEED

LD      P50.1
JUMP NO_TX_SLAVE_SPEED

**********************************
*WRITE SLAVE SPEED
**********************************
LD      P98.32
ANDN  P98.7
ANDN  P98.9
JUMP TX_SLAVE_SPEED

|NO_TX_SLAVE_SPEED

**********************************
*RX AKNOWLDGE
**********************************
```



```
**********************************
*RX AKNOWLDGE
**********************************
LD      P98.7
LD      P98.7
LD      P98.7
OUTN  P54.1
OUT    P54.2
JUMP RX_OK

**********************************
*TX AKNOWLDGE
**********************************
LD      P98.9
LD      P98.9
LD      P98.9
OUNT  P54.2
OUT    P54.1
JUMP TX_OK



END

**********************************
*END MAIN  LOOP PROGRAM
**********************************

**********************************
*READ MASTER SPEED
**********************************
|RX_MASTER_SPEED


LD      P73.32
MOVW P74            SET DRIVE ADDRESS 2
808464946           (HEX 30303232)

LD      P73.32      P75 = 59 59 IS
MOVW P75            DIGITAX SPEED
59      _           PARAMETER

LD      P73.32      P78 = 4 START READ
MOVW P78            PARAMETER
4

LD      P73.32      PRESET TIMER FOR
MOVW P64            TIME OUT 2
200

LD      P73.32
SET     P98.7

LD      P73.32
RES     P98.10

LD      P73.32
OUT     P98.32

END
```



```
**********************************
*WRITE SLAVE SPEED
**********************************
|TX_SLAVE_SPEED

LD      P73.32              P102 = (P76*256)
MULW    P102 P108 P76

LD      P73.32              P102 = P102*RATIO
MULW    P102 P102 P110

LD      P73.32              P102 = P102+DECIMAL PART
ADDW    P102 P102 P105

LD      P73.32
ANDW    P105 P102 P106      P105 = P102 AND H3FFFF
        _

LD      P73.32              P76 = P102/262144
DIVW    P76 P102 P103

LD      P73.32              SET DRIVE ADDRESS 1
MOVW    P74                 (HEX 30303131)
808464689

LD      P73.32              P75 = 0 IS DIGITAX
MOVW    P75                 DIGITAL SPEED PARAMETER
0

LD      P73.3               P78 = 1 START WRITE
MOVW    P78                 PARAMETER
1

LD      P73.32              PRESET TIMER FOR
MOVW    P64                 TIMEOUT
200

LD      P73.32
SET     P98.9

LD      P73.32
RES     P98.10

LD      P73.32
OUTN    P98.32

END
```



```
**********************************
*TEST FOR DRIVE AKNOWDGE DURING TX
**********************************
|TX_OK

LD      P78.1           IF P78 = 3
AND     P78.2
ANDN    P78.3
ANDN    P78.4
OUT     P98.10


LD      P73.1           IF TIMER P64 = 0
OR      P98.10          OR P78 = 3
RES     P98.9


LDN     P98.9           TEST TIME OUT
ANDN    P98.10
SET     P54.3


END

**********************************
*TEST FOR DRIVE AKNOWDGE DURING RX
**********************************
|RX_OK


LDN     P78.1           IF P78 = 6
AND     P78.2
AND     P78.3
ANDN    P78.4
OUT     P98.10

LD      P73.1           IF TIMER P64 = 0
OR      P98.10          OR P78 = 6
RES     P98.7


LDN     P98.7           TEST TIMEOUT
ANDN    P98.10
SET     P54.4


END
```