# Mechatronics

# NS500 (PROFIBUS-DP)
# NS300 (DEVICE-NET)

# QUICK REFERENCE

OMRON

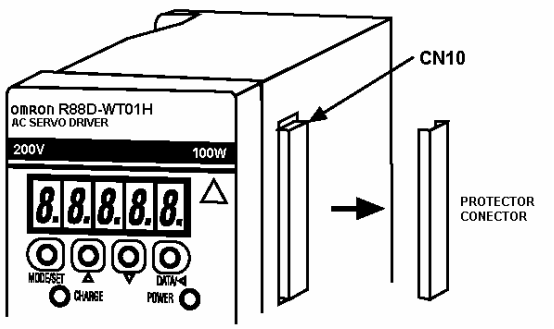
# QUICK REFERENCE:
# NS500 (PROFIBUS-DP)
# NS300 (DEVICE-NET)

## THIS MANUAL CONTAINS:

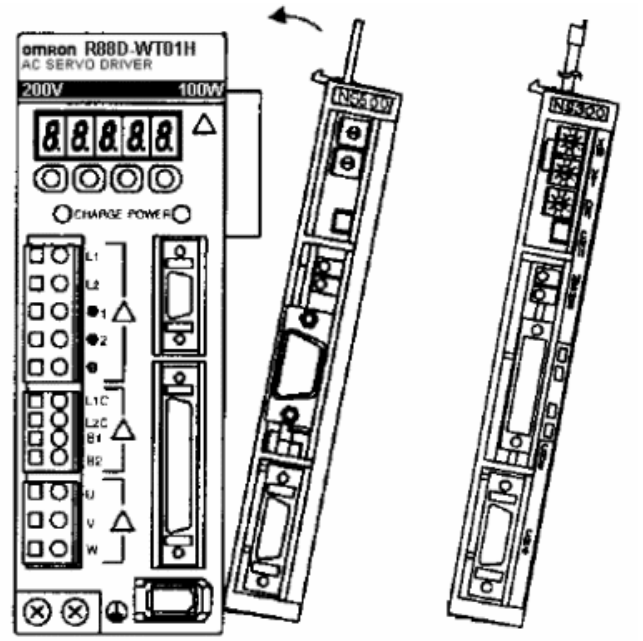
# 1- NS500/NS300 INSTALLATION.

Following steps should be used to install the NS500/NS300 into a W servodriver:

1.- Take off the protector in the CN10 connector situated in the right side of the servodriver.

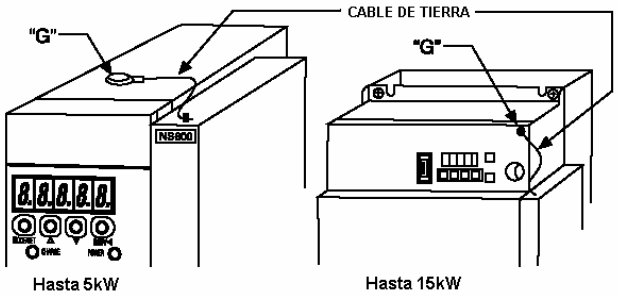


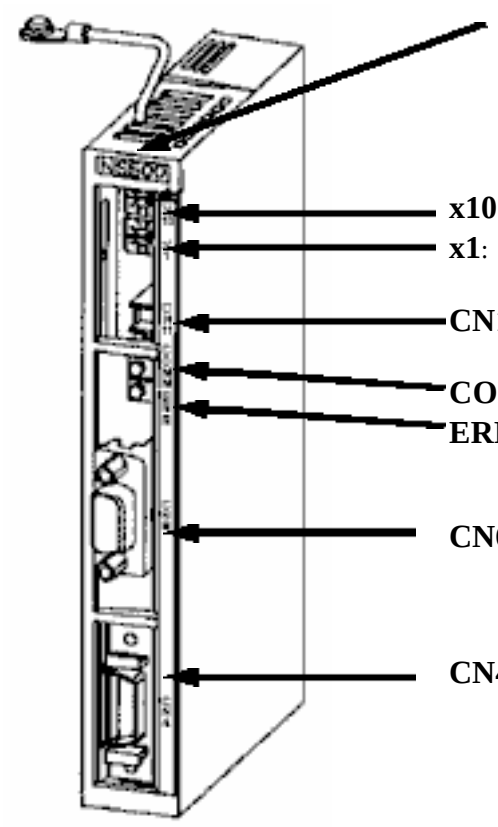

2.- Clamp the option board to the servodriver using the pieces in the bottom of the option board.




3.- Push the board following the arrow direction and clamp the option board using the upper piece.

4.- To connect the board ground cable to the servodriver use a Phillips screw M3x10 (up to 1kW) and M4x8 (up to 15 kW) with the point “G”.





# *Option board NS500 PROFIBUS-DP*

**x10**: Rotatory Switch for slave number

**x1**: Rotatory Switch for slave number

**CN11**: (RS232C) communication software connector (NsxxxSetuptool)

**COMM**: PROFIBUS communication state Led (green, red)

**ERR**: Option board error led

**CN6**: PROFIBUS-DP communication connector

**CN4**: I/O connector (close loop encoder input and emergency stop)

The reference code for the different connectors is this one:

| | | | |
|---|---|---|---|
| 20 PINS RIGHT ANGLE RECEPTACLE | | CN4 | Port |
| SERVOS CN4-V1 | CN11 | Port | |
| "SUB-D 9 pins male connector" | | CN6 | Port (PROFIBUS-DP) |

The configuration for the software cable to use the **NSxxx SETUP TOOL** software is this one:

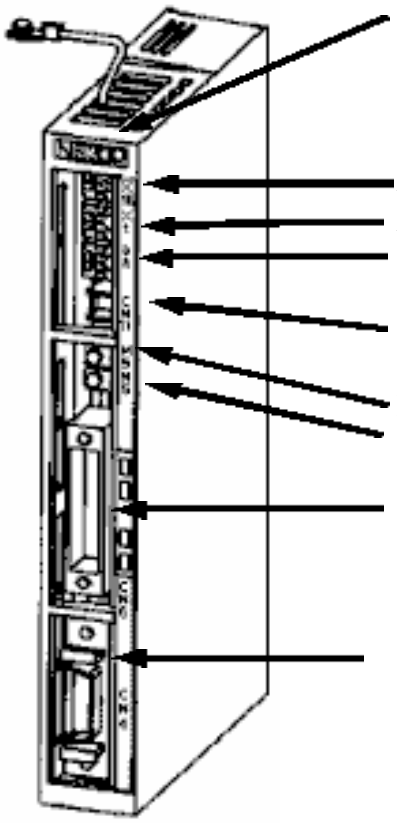
| **NS500 CN11** | **Details** | **PC (9 pines)** |
|---|---|---|
| 1 (TXD) | Serial data output | 2 (RXD) |
| 2 (RXD) | Serial data input | 3 (TXD) |
| 3 (GND) | Ground | 5 (GND) |
| 4 (GND) | Ground | 5 (GND) |

The CN6 connector is used for the PROFIBUS DP bus and the configuration is:

| | | |
|---|---|---|
| Pin 3: | RXD/TXD-P | Send / receive positive data |
| Pin 8: | RXD/TXD-N | Send / receive negative data |
| Pin 5: | DGND | Ground |
| Pin 6 | VP | +5V |

The communication cable between the Profibus-DP master and slave is flat style (direct), it means 3-3 and 8-8.

# *Option board NS300 DEVICE-NET*



**x10**: Rotatory Switch for slave number

**x1**: Rotatory Switch for slave number

**DR**: Rotatory Switch for the Device-Net communication speed

**CN11**: (RS232C) Software communication connector (NsxxxSetuptool)

**MS**: NS300 state led (green, red)

**NS**: Device-Net communication state led (green, red)

**CN6**: DEVICE-NET communication connector

**CN4**: I/O connector (close loop encoder input and emergency stop)

The code references for the different connectors are these ones:

20 PINS RIGHT ANGLE RECEPTACLE CN4 port
SERVOS CN4-V1 CN11 port

The communication cable for the programming software **NSxxx SETUP TOOL** is this one:

| **NS300 CN11** | **Details** | **PC (9 pines)** |
|---|---|---|
| 1 (TXD) | Serial data output | 2 (RXD) |
| 2 (RXD) | Serial data input | 3 (TXD) |
| 3 (GND) | Ground | 5 (GND) |
| 4 (GND) | Ground | 5 (GND) |

The communication cable between the Device-Net master and the slave is a flat style (direct) following the connector colours.

# 2- CONFIGURATION

When an option board NS500/NS300 is connected with a servodriver a new system is formed with different characteristics than the servodriver by itself.

- PARAMETERS: Some parameters of the servodriver are no more time useful (for example, speed and torque control parameters, because with the option board the control method only could be position control). Some other parameters are added (for example, Pn823: Origin search method, etc…)

- INPUT / OUTPUT: Like the parameters some inputs disappear and other specific one appears. The inputs and outputs are no more time configurable and some of themes are fixed on CN1 and CN4 of NS500 and NS300.

- Programming software **NSxxx SETUP TOOL**: It should be used to program and monitor the status of NS300/NS500 because new parameters are not accessible using the servodriver display.

## 2.1.- INPUTS / OUTPUTS

**CN1 CONNECTOR (SERVODRIVER)**

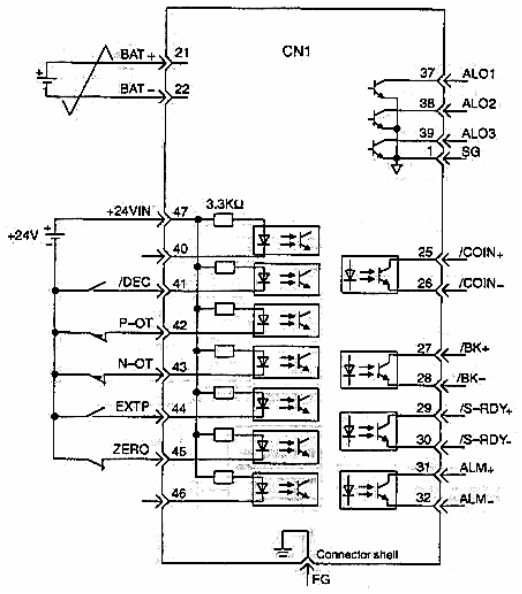
INPUTS

| | | |
|---|---|---|
| CN1-47 | | +24VIN Input common. 24 VDC are needed to supply the inputs. |
| **CN1-41** | **/DEC** | **Origin proximity signal (HOMING function).** |
| CN1-42 | POT | Positive overtravel limit. |
| CN1-43 | NOT | Negative overtravel limit. |
| **CN1-44** | **EXTP** | **External positioning signal.** |
| **CN1-45** | **ZERO** | **Zero point signal (origin).** |
| | | This one is used to fix an origin without the encoder Z signal. |

OUTPUTS

| | | |
|---|---|---|
| CN1-31, 32 | /ALM | Alarm: Turns OFF when an error is detected. |
| CN1-25, 26 | COIN | Positioning complete outputs (INPOSITION). |
| CN1-27, 28 | BKIR | Brake interlock output. |
| CN1-29, 30 | READY | Servodriver without alarm and ready. |
| CN1-37/38/39,1 | AL01/AL02/AL03 | Alarm output code. |

There are no encoder signals.



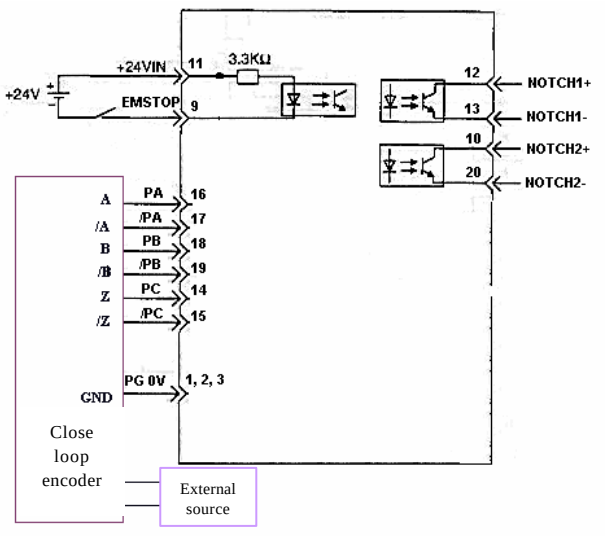

## CN4 CONNECTOR (OPTION BOARD NS500 / NS300)

INPUTS

| | | |
|---|---|---|
| CN4-11 | +24V/COM | Input common. 24VDC for the inputs power supply. |
| CN4-9 | EMSTOP | Emergency stop input |
| CN4-14 | PC | External encoder input phase Z. |
| CN4-15 | /PC | External encoder input phase /Z |
| CN4-16 | PA | External encoder input phase A |
| CN4-17 | /PA | External encoder input phase /A |
| CN4-18 | PB | External encoder input phase B |
| CN4-19 | /PB | External encoder input phase /B |
| CN4-1/2/3 | PG 0V | Ground |

OUTPUTS

| | | |
|---|---|---|
| CN4-12, 13 | NOTCH1 | Notch output 1 (Turns On when actual position is inside range 1). |
| CN4-10, 20 | NOTCH2 | Notch output 2 (Turns On when actual position is inside range 2). |




**CLOSE LOOP WITH EXTERNAL ENCODER**

In the NS500/NS300 is possible to add an external encoder to close the position control loop. To make it work is necessary 5V line-driver encoder output and connected directly to the controlled axis.

Only two parameters are needed to configure the external encoder when this one has been connected.

Pn206: External encoder pulses/revolution

Pn002.3: to enable the close loop
- → 0 = disable, not used.
- → 1 = Close loop encoder without Z phase
- → 2 = Close loop encoder with Z phase
- → 3 = Close loop encoder without Z phase used in reverse mode
- → 4 = Close loop encoder with Z phase used in reverse mode.

# 2.2.- PARAMETERS

The parameters that remain in the servodriver are these ones:

| Category | Parameter Number | Name | Unit | Setting Range | Default Setting |
|---|---|---|---|---|---|
| Function Selection Parameters | **Pn000** | **Pn000.0** Rotation direction<br>**Pn000.1=1** Control method (fixed to position)<br>**Pn000.2** Serial slave address<br>(See note 3) | — | — | 0010 |
| | **Pn001** | **Pn001.0** Stopping method for Alarm or Servo-off<br>**Pn001.1** Overtravel stopping method<br>**Pn001.2** AC/DC power supply<br>**Pn001.3** Warning output selection<br>(See notes 1 and 3) | — | — | 0000 |
| | **Pn002** | **Pn002.0** Not used<br>**Pn002.1** Not used<br>**Pn002.2** Absolute encoder<br>**Pn002.3** Close loop encoder<br>(See note 3) | — | — | 0000 |
| | Pn003 | Function Selection Application Switches 3 | — | — | 0002 |
| Gain Parameters | Pn100 | Speed Loop Gain | Hz | 1 to 2000 | 40 |
| | Pn101 | Speed Loop Integral Time Constant | 0.01ms | 15 to 51200 | 2000 |
| | Pn102 | Position Loop Gain | $s^{-1}$ | 1 to 2000 | 40 |
| | Pn103 | Inertia Ratio | % | 0 to 10000 | 0 |
| | Pn104 | Not Used | — | — | — |
| | Pn105 | Not Used | — | — | — |
| | Pn106 | Not Used | — | — | — |
| | Pn107 | Bias | Rpm | 0 to 450 | 0 |
| | Pn108 | Bias Width Addition | Ref. units | 0 to 250 | 7 |
| | Pn109 | Feed-forward | % | 0 to 100 | 0 |
| | Pn10A | Feed-forward Filter Time Constant | 0.01ms | 0 to 6400 | 0 |
| | **Pn10B** | **Gain-related Application Switches** (See note 3) | — | — | 0000 |
| | Pn10C | Mode Switch Torque Reference | % | 0 to 800 | 200 |
| | Pn10D | Mode Switch Speed Reference | Rpm | 0 to 10000 | 0 |
| | Pn10E | Mode Switch Acceleration | 10rpm/s | 0 to 3000 | 0 |
| | Pn10F | Mode Switch Error Pulse | Ref. units | 0 to 10000 | 0 |
| | **Pn110** | **Online Autotuning Switches** (See note 3) | — | — | 0010 |
| | **Pn111** | **Speed Feedback Compensation** (See note 2) | % | 1 to 100 | 100 |
| | Pn112 | Reserved parameters (Do not change) | % | 0 to 1000 | 100 |
| | Pn113 | | — | 0 to 10000 | 1000 |
| | Pn114 | | — | 0 to 400 | 200 |
| | Pn115 | | — | 0 to 1000 | 32 |
| | Pn116 | | — | 0 to 1000 | 16 |
| | Pn117 | | % | 20 to 100 | 100 |
| | Pn118 | | % | 20 to 100 | 100 |
| | Pn119 | | s–1 | 1 to 2000 | 50 |
| | Pn11A | | 0.1% | 1 to 2000 | 1000 |
| | Pn11B | | Hz | 1 to 150 | 50 |
| | Pn11C | | Hz | 1 to 150 | 70 |
| Gain Parameters | Pn11D | Reserved parameters (Do not change) | % | 1 to 150 | 100 |
| | Pn11E | | % | 1 to 150 | 100 |
| | Pn11F | | ms | 1 to 2000 | 0 |
| | Pn120 | | 0.01 ms | 1 to 51200 | 0 |
| | Pn121 | | Hz | 10 to 250 | 50 |
| | Pn122 | | Hz | 0 to 250 | 0 |
| | Pn123 | | % | 0 to100 | 0 |
| Position Parameters | **Pn200** | **Position Control Reference Selection Switches** (See note 3) | — | — | 0000 |
| | **Pn201** | **Not used** | p/r | 16 to 16384 | 16384 |
| | **Pn202** | **Not used** (Electronic gear Pn810) | — | 1 to 65535 | 4 |
| | **Pn203** | **Not used** (Electronic gear Pn811) | — | 1 to 65535 | 1 |
| | Pn204 | Not Used | — | — | — |
| | **Pn205** | **Multi-turn Limit Setting** (See notes 1 and 3) | rev | 0 to 65535 | 65535 |
| | Pn206 | **Close loop encoder revolutions** | — | — | — |
| | **Pn207** | **Not used** (See note 3) | — | — | 0001 |
| | **Pn208** | **Not used (**See note 3) | 0.01ms | 0 to 6400 | 0 |
| | Pn300 | Not Used | — | — | — |
| | Pn301 | Not Used | — | — | — |
| | Pn302 | Not Used | — | — | — |
| | Pn303 | Not Used | — | — | — |

| | | | | | |
|---|---|---|---|---|---|
| Speed Parameters | Pn304 | Jog speed by digital operator | — | — | — |
| | Pn305 | Not Used | — | — | — |
| | Pn306 | Not Used | — | — | — |
| | Pn307 | Not Used | — | — | — |
| | Pn308 | Speed Feed-Forward Filter Time Constant | 0.01ms | 0 to 65535 | 0 |
| Torque Parameters | Pn400 | Not Used | — | — | — |
| | Pn401 | Torque Reference Filter Time Constant | 0.01ms | 0 to 65535 | 100 |
| | Pn402 | Forward Torque Limit | % | 0 to 800 | 800 |
| | Pn403 | Reverse Torque Limit | % | 0 to 800 | 800 |
| | Pn404 | Forward rotation external current limit | — | — | — |
| | Pn405 | Reverse rotation external current limit | — | — | |
| | Pn406 | Emergency stop torque | — | — | — |
| | Pn407 | Not Used | — | — | — |
| | Pn408 | Torque Function Switches | — | — | 0000 |
| | Pn409 | Notch Filter Frequency | Hz | 50 to 2000 | 2000 |
| Sequence Parameters | Pn500 | Not Used | — | — | — |
| | Pn501 | Not Used | — | — | — |
| | Pn502 | Rotation Detection Level | rpm | 1 to 10000 | 20 |
| | Pn503 | Not Used | — | — | — |
| | Pn504 | Not Used | — | — | — |
| | Pn505 | Overflow Level | 256 ref.units | 1 to 32767 | 1024 |
| | Pn506 | Brake Reference Servo OFF Delay Time | 10ms | 0 to 50 | 0 |
| | Pn507 | Brake Reference Output Speed Level | rpm | 0 to 10000 | 100 |
| | Pn508 | Timing for Brake Reference Output during Motor Operation | 10ms | 10 to 100 | 50 |
| | Pn509 | Momentary Hold Time | ms | 20 to 1000 | 20 |
| | Pn50A | Input Signals Selection 1 | — | — | 8881 (fixed) |
| | Pn50B | Input Signals Selection 2 | — | — | 8888 (fixed) |
| | Pn50C | Not used | — | — | 8888 (fixed) |
| | Pn50D | Not used | — | — | 8888 (fixed) |
| | Pn50E | Output Signals Selection 1 | — | — | 3000 (fixed) |
| | Pn50F | Output Signals Selection 2 | — | — | 1200 (fixed) |
| | Pn510 | Output Signals Selection 3 | — | — | 0000 (fixed) |
| | Pn511 | Reserved (do not change) | — | — | 8ED8 (fixed) |
| | Pn512 | Output Signals Reversal Settings | — | — | 0000 |
| Other Parameters | Pn600 | Regenerative Resistor Capacity (See note 4) | 10 W | 0 to capacity (See note 5) | 0 |
| | Pn601 | Reserved parameter (Do not change) | — | 0 to capacity (See note5) | 0 |

The new parameters added in the option board are these ones:

**Origin search parameters**

| | |
|---|---|
| Pn800 | Homing mode |
| Pn801 | Bit 0: Origin search direction |
| | Bit 1: Origin proximity input signal setting /DEC |
| | Bit 2: Origin input signal setting /ZERO |
| | Bit 3-15: - |
| Pn802 | Homing feed speed |
| Pn803 | Homing approach speed |
| Pn804 | Homing Creep speed |
| Pn805 | Final running distance for homing |
| Pn806 | Home position signal output width |
| Pn809 | Home position offset |
| Pn80A | Homing Accel/Decel time |

**Mechanical system parameters**

| | |
|---|---|
| Pn810 | Electronic gear (numerator) |
| Pn811 | Electronic gear (denominator) |
| Pn812 | Coordinate system type |
| Pn813 | Ref unit value per load shaft rotation (resolution) |
| Pn814 | Backlash compensation value |
| Pn815 | Backlash compensation direction |
| Pn816 | Stored stroke limit (+) |
| Pn817 | Stored stroke limit (-) |
| Pn818 | Mechanical function selection |
| Pn819 | Overtravel signal function select |
| Pn81A | Overtravel stopping method |
| Pn81B | Emergency stop signal function selection |

**Speed acceleration parameters**

| | |
|---|---|
| Pn821 | Positioning feed speed |
| Pn822 | Positioning acceleration time |
| Pn823 | Positioning deceleration time |
| Pn824 | Positioning 2-stage Acceleration/Deceleration switching speed |
| Pn825 | Positioning 2-stage Acceleration/Deceleration time |
| Pn826 | Positioning acceleration/deceleration time |
| Pn827 | External signal positioning speed |
| Pn829 | Acceleration/deceleration filter selection |
| Pn830 | Feed operation reference unit of feed speed (Byte 4-7) |
| Pn831 | Feed operation speed |
| Pn832 | Feed operation acceleration time |
| Pn833 | Feed operation deceleration time |
| Pn834 | Feed operation 2-stage Accel/decel switching |
| Pn835 | Feed operation 2-stage Accel/decel time |
| Pn836 | Feed operation 2-stage Accel/decel profile |
| Pn840 | Positioning exponential Accel/Decel time constant |
| Pn841 | Positioning Accel/decal bias speed |
| Pn842 | Positioning moving average time constant |
| Pn843 | Maximum feed speed |
| Pn844 | Step feed distance 1 |
| Pn845 | Step feed distance 2 |
| Pn846 | Step feed distance 3 |
| Pn847 | Step feed distance 4 |

**Positioning**

| | |
|---|---|
| Pn850 | Positioning completion range |
| Pn851 | Positioning completion Time-Out time |
| Pn852 | Positioning NEAR detection range |
| Pn853 | Quick positioning (Station Nº positioning) |
| Pn854 | External signal positioning approach speed |
| Pn855 | External signal positioning moving distance |
| Pn856 | Function setting of external signal positioning |
| Pn85A | Station number |

**Positioning with multi stage feeding function**

| | |
|---|---|
| Pn861 | Speed switching point number |
| Pn862 | Feed speed at start |
| Pn863 | Speed switching position 1 |
| Pn864 | Speed switching position 2 |
| Pn865 | Speed switching position 3 |
| Pn866 | Speed switching position 4 |
| Pn867 | Speed switching position 5 |
| Pn868 | Speed switching position 6 |
| Pn869 | Speed switching position 7 |
| Pn86A | Speed switching position 8 |
| Pn86B | Speed switching position 9 |
| Pn86C | Speed switching position 10 |
| Pn86D | Speed switching position 11 |
| Pn86E | Speed switching position 12 |
| Pn86F | Speed switching position 13 |
| Pn870 | Speed switching position 14 |
| Pn871 | Speed switching position 15 |
| Pn872 | Speed switching position 16 |
| Pn873 | Switching speed at position 1 |
| Pn874 | Switching speed at position 2 |
| Pn875 | Switching speed at position 3 |
| Pn876 | Switching speed at position 4 |
| Pn877 | Switching speed at position 5 |
| Pn878 | Switching speed at position 6 |
| Pn879 | Switching speed at position 7 |
| Pn87A | Switching speed at position 8 |
| Pn87B | Switching speed at position 9 |
| Pn87C | Switching speed at position 10 |
| Pn87D | Switching speed at position 11 |
| Pn87E | Switching speed at position 12 |
| Pn87F | Switching speed at position 13 |
| Pn880 | Switching speed at position 14 |
| Pn881 | Switching speed at position 15 |
| Pn882 | Switching speed at position 16 |

**Passing point output positioning**

| | |
|---|---|
| Pn890 | Notch signal setting |
| Pn891 | Notch signal polarity select |
| Pn892 | Notch output position 1 |
| Pn893 | Notch output position 2 |
| Pn894 | Notch output position 3 |
| Pn895 | Notch output position 4 |

**System**

| | |
|---|---|
| Pn8FD | Error mask |
| Pn8FE | Parameter check sum rebuild |
| Pn8ff | Factory setting reset |

**Point table**

| | |
|---|---|
| Pn900 | Target position 1 |
| ⋮ | ⋮ |
| Pn931 | Target position 2 |
| Pn940 | Feed speed for target 1 |
| ⋮ | ⋮ |
| Pn971 | Feed speed for target 50 |

## ORIGIN SEARCH PARAMETERS

These parameters are related only with this function, speed, distance, operation type, etc... So they'll be explained in the origin search section of this guide.

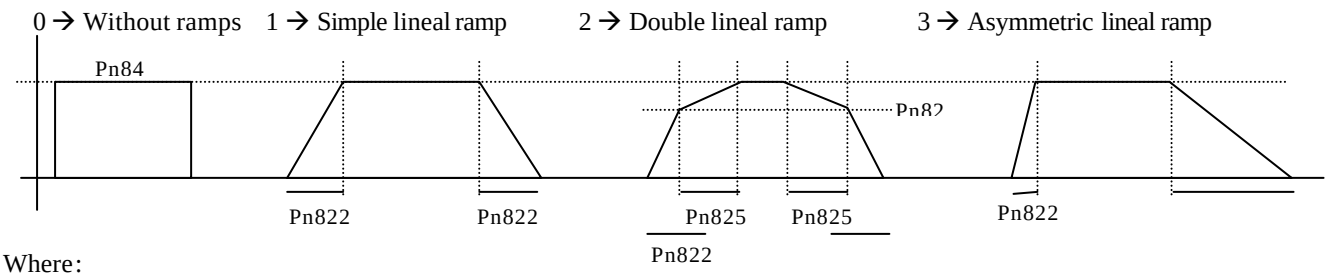
## MECHANICAL SYSTEM PARAMETERS

Gear ratio selection
These two parameters are used to define the ELECTRONIC GEAR or PULSE RATE, in other words the relationship between pulses (internally used by the driver) and the reference units (that are managed by the user).
Depending of the mechanical system coupled to the motor and the reference units select by the user different values should be applied to Pn810/Pn811

For example: With a 13 bits (2048ppr) servomotor with a gear reduction 1/5 and connected to a landscrew with 5mm/rev of pitch. If the user wants to use mm and use two decimals instead of pulses the calculation will be this one.
1 motor revolution = 2048pulsos x4(feedback) = 8192 pulses. With a gear reduction of 1/5 we need 5 rotation of the motor for each revolution of the gear: 1 rev = 8192 x 5 = 40960 pulses
Pn810/Pn811 = 40960 pulses / 500 (0.01mm); it means that 40960 feedback pulses correspond to 500 reference units, in this case 0.01mm.
The other mechanical system parameters are used to define lineal or circular coordinates, the backlash compensation, the software and hardware limits and the stop method for limits or alarms.

## SPEED PARAMETERS, ACCELERATION AND DECELERATION

Several speeds and distance for different positioning commands and also the acceleration and deceleration time could be configured on this section.
In Pn826 and Pn829 is possible to set the acceleration/deceleration profile.

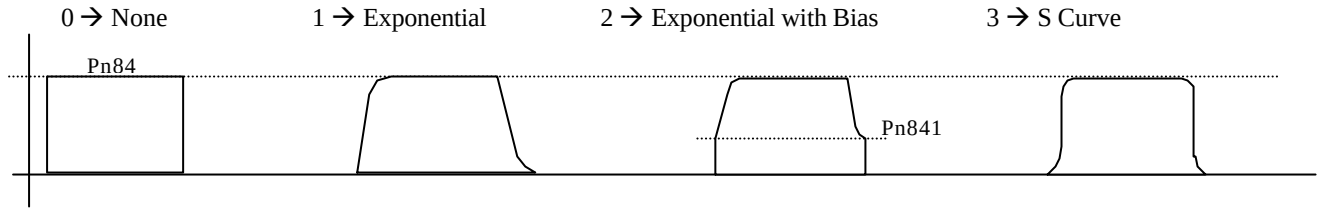



Where:
Pn822 is the acceleration time
Pn843 is the maximum speed
Pn824 is the switch speed to use the 2nd accel/decel time parameters
Pn825 is the 2nd accel/decel time
Pn823 is the deceleration time.

Is possible to apply a filter to these profiles in parameter Pn829:

0 → **None**
1 → **Exponential** with Pn840 as a time constant
2 → **Exponential with bias**, with Pn840 as a time constant and Pn841 as a bias speed. Filter only applied up the bias speed
3 → **S Curve** with Pn843 as a time constant

Is possible to apply filters 0 and 3 to any profile but filters 1 and 2 are only applicable to the profile without ramps.




**POSITIONING PARAMETERS**

These ones are used to define the external positioning and some status bits.

**MULTI SPEED PARAMETERS**

Are useful to define the 16 multi speeds and the positions where the multi-speeds will be applied.

**NOTCH OUTPUT PARAMETERS**

The parameters allow the definition and activation of NOTCH1 and NOTCH2 external outputs. This outputs turns on when the actual position is inside the defined range and when the position command include the Notch output.

**POINT TABLE PARAMETERS**

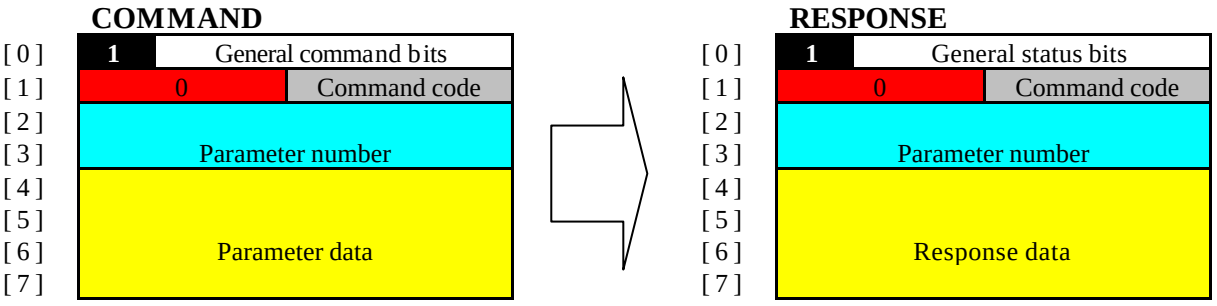
These parameters define the target position and the target speed for a positioning; up to 50 table points are available. To use this table positioning is necessary to use the point table command.

# 3- PROGRAMMING

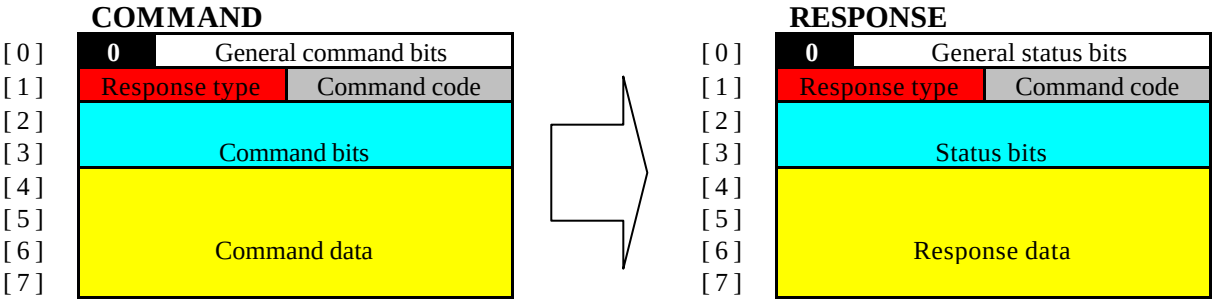
The option boards NS500/NS300 for the W servos always work as 8 bytes mixed slaves, PROFIBUS-DP / DEVICE-NET. It means that each board uses 8 input bytes and 8 output bytes, these bytes are used to program or command the module (outputs) and monitor the status (inputs).

Depending of each function the 8 bytes format change. The functions could be spared into parameter command (write/read) and movement commands.

**PARAMETER COMMAND FORMAT (WRITE /READ)**




**MOVEMENT COMMAND FORMAT**




In both cases we have

**General command bits**

| Byte | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
|---|---|---|---|---|---|---|---|---|
| [0] | MOD | 0 | ALRST | ESTP | 0 | 0 | SVON | C_STRT |

Where:

**MOD** is the **command type**, movement (**0**) or parameter read/write (**1**)

**ALRST** is the **alarm reset** command

**ESTP** is the **software emergency stop** bit negative logic

**SVON** is the **RUN (servo on) bit**

**C_STRT** is the bit that executes the selected command in the command code. Gives the **START** of a movement command

**General status bits**

| Byte | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
|---|---|---|---|---|---|---|---|---|
| [0] | MOD_R | READY | PWRON | ESTP_R | ALRM | WARN | SVON_R | C_STRT_R |

Where:

**MOD_R** shows the command type, movement (**0**) or parameters (**1**).

**READY** shows that NS500/NS300 unit is **ready** to receive commands
**PWRON** shows that unit is powered (**Power ON**)
**ESTP_R** shows if the **emergency input** is active (0) or not (1).
**ALM_R** shows if option board is in **alarm**
**WARN** shows if the unit has detected some **warning (AL9x)** in the servo or in the NSxxx
**SVON_R** shows if the motor is in **RUN** or Baseblock
**C_STRT_R** shows if one movement command has been received (**C_STRT)**

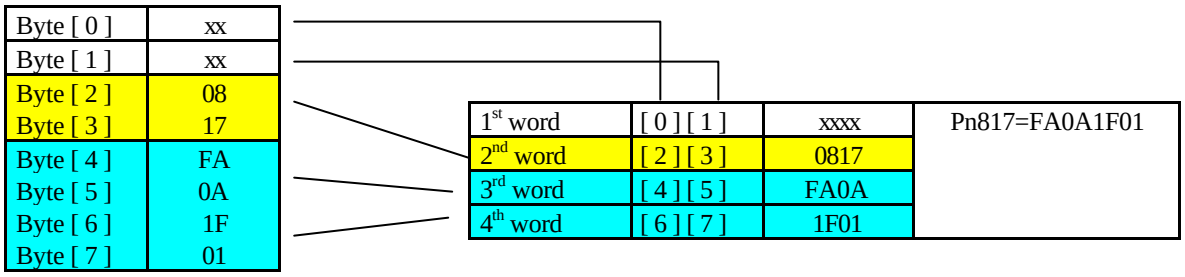
## 3.1. - DATA MANAGEMENT

PROFIBUS-DP and DEVICE-NET works in byte level but Omron's PLCs works in word level so we'll find some differences depending of the bus master used in the network.

**PROFIBUS-DP (C200HW-PRM21 master)**

With Profibus-DP master the bytes are ordered in this way: First the most significative weight and follow by the less significative one.

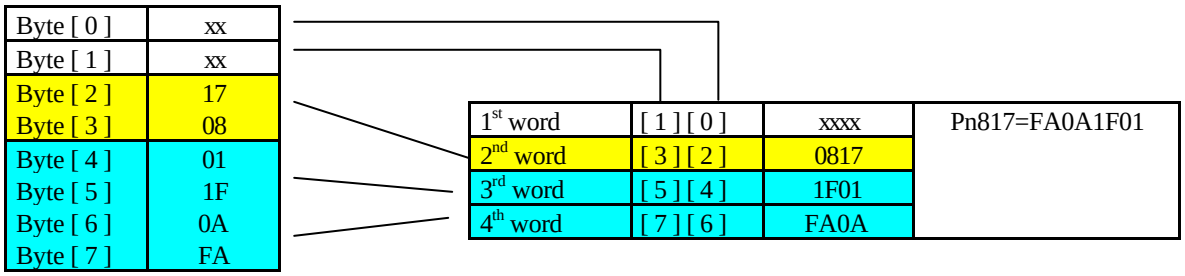
Ex:




**DEVICE-NET (C200HW-DRM21-V1 or CS1W-DRM21 master)**

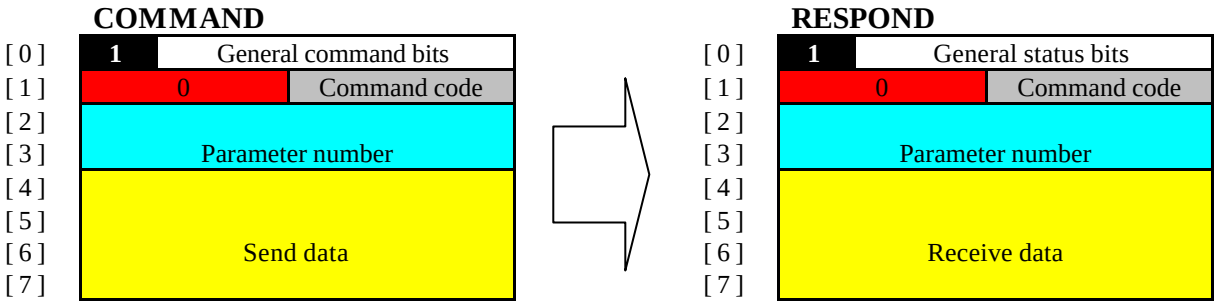
This devicenet masters works in reverse order compared with the Profibus: In this case the less significative weight bytes are allocated first and more significative one are placed after it.

Ex:



# 3.2. - PARAMETER COMMANDS

These commands are used to read and write parameters into the servodriver or into the option board also is possible to establish the zero point or define a position. The format is this one:




The command codes are:

| | |
|---|---|
| **0000** | **No operation** |
| **1000** | **Read parameter** |
| **1001** | **Write parameter** |
| **1010** | **Actual position definition** |
| **1100** | **Read alarm** |
| **1110** | **NSxxx module reset (Store parameters in FLASH RAM)** |

Here we can see some examples about these commands, for this examples we have used IR050 for outputs and IR350 for inputs (NS500) and channels 3200 and 3300 for the NS300.

**3.2.1. - PARAMETER READ** (Pn817: Negative software limit)

PROFIBUS-DP

1. Change the command code (1000b = 8h) → [1] = 08
2. Parameter number (817hex) → [2][3] = 08 17
3. Active the execution command bit C_STRT → change from [0]= 80 to [0]=81

It means:

| | | |
|---|---|---|
| c050 → [0][1] → 8008 | When bit 050.08 turns On we have | c0350 → E008 |
| c051 → [2][3] → 0817 | | c0351 → 0817 |
| c052 → [4][5] → 0000 | | c0352 → FA0A |
| c053 → [6][7] → 0000 | | c0353 → 1F01 |

So the response is FA0A1F01h, that in this case is negative so it's necessary to convert using the two's complement (NOT+1). After applying the conversion the number we found is 5F5E0FFh that converted in decimal result 9999999.

**Pn817 = -99999999**

DEVICE-NET

1. Change the command code (1000b = 8h) → [1] = 08
2. Parameter number (817hex) → [2][3] = 17 08
3. Activate the execution command bit C_STRT → Change from [0]= 80 to [0]=81

It means:

| | | |
|---|---|---|
| c3200 → [1][0] → 0880 | When bit 3200.00 turns On we have | c3300 → 08F1 |
| c3201 → [3][2] → 0817 | | c3301 → 0817 |
| c3202 → [5][4] → 0000 | | c3302 → 1F01 |
| c3203 → [7][6] → 0000 | | c3303 → FA0A |

So the response is FA0A1F01h, that in this case is negative so it's necessary to convert using the two's complement (NOT+1). After applying the conversion the number we found is 5F5E0FFh that converted in decimal result 9999999.

**Pn817 = -99999999**

**3.2.2. - PARAMETER WRITE** (Pn100: Speed loop gain = 100Hz)

PROFIBUS-DP

1. Change the command code (1001b = 9h) → [1] = 09
2. Parameter numb er (100hex) → [2][3] = 01 00
3. Parameter data (100 = 64hex) → [4][5][6][7] = 00 00 00 64
4. Activate the execution command bit C_STRT → Change from [0]= 80 to [0]=81

It means:

| | | |
|---|---|---|
| c050 → [0][1] → 8009 | When bit 050.08 turns On we have | c0350 → E009 |
| c051 → [2][3] → 0100 | | c0351 → 0100 |
| c052 → [4][5] → 0000 | | c0352 → 0000 |
| c053 → [6][7] → 0064 | | c0353 → 0064 |

So we write **Pn100 = 100**

In the same way that in read command, if the data is negative it should be write in two's complement (hexadecimal value inverter (NOT)+1)

DEVICE-NET

1. Change the command code (1001b = 9h) → [1] = 09
2. Parameter number (100hex) → [3][2] = 01 00
3. Parameter data (100 = 64hex) → [7][6][5][4] = 00 00 00 64
4. Activate the execution command bit C_STRT → Change from [0]= 80 to [0]=81

It means:

| | | |
|---|---|---|
| c3200 → [1][0] → 0980 | When bit 3200.00 turns On we have | c3300 → 09F1 |
| c3201 → [3][2] → 0100 | | c3301 → 0100 |
| c3202 → [5][4] → 0064 | | c3302 → 0064 |
| c3203 → [7][6] → 0000 | | c3303 → 0000 |

**3.2.3. - SET ACTUAL POSITION** (Define the actual position as -23456)

PROFIBUS-DP

1. Change the command code (1010b = Ah) → [1] = 0A
2. Parameter number (none) → [2][3] = 00 00
3. Parameter data (position) –23456 = FFFFF6D7h → [4][5][6][7] = FF FF F6 D7
4. Activate the execution command bit C_STRT → Change from [0]= 80 to [0]=81

It means:

| | | |
|---|---|---|
| c050 → [0][1] → 800A | When bit 050.08 turns On we have | c0350 → E00A |
| c051 → [2][3] → 0000 | | c0351 → 0000 |
| c052 → [4][5] → FFFF | | c0352 → FFFF |
| c053 → [6][7] → F6D7 | | c0353 → F6D7 |

DEVICE-NET

1. Change the command code (1010b = Ah) → [1] = 0A
2. Parameter number (none) → [3][2] = 00 00
3. Parameter data (position) –23456 = FFFFF6D7h → [7][6][5][4] = D7 F6 FF FF
4. Activate the execution command bit C_STRT → Change from [0]= 80 to [0]=81

It means:

| | | |
|---|---|---|
| c3200 → [1][0] → 0A80 | When bit 3200.00 turns On we have | c3300 → 0AF1 |
| c3201 → [3][2] → 0000 | | c3301 → 0000 |
| c3202 → [5][4] → F6D7 | | c3302 → F6D7 |
| c3203 → [7][6] → FFFF | | c3303 → FFFF |

**3.2.4. - ALARM READ** (Up to 4 alarms could be read)

PROFIBUS-DP

1. Change the command code (1100b = Ch) → [1] = 0C
2. Parameter number (none) → [2][3] = 00 00
3. Parameter data (none) → [4][5][6][7] = 00 00 00 00
5. Activate the execution command bit C_STRT → Change from [0]= 80 to [0]=81

It means:

| | | |
|---|---|---|
| c050 → [0][1] → 800C | When bit 050.08 turns On we have | c0350 → E10C |
| c051 → [2][3] → 0000 | | c0351 → 0000 |
| c052 → [4][5] → 0000 | | c0352 → 9999 |
| c053 → [6][7] → 0000 | | c0353 → 9972 |

When 99 is displayed it means that there is no alarm in the driver. 72 correspond to A.72→ Overload. In this example only the alarm 72 has been reproduced. If no alarm is detected the values will be 9999 and 9999.

DEVICE-NET

1 Change the command code (1100b = Ch) → [1] = 0C
2 Parameter number (none) → [3][2] = 00 00
3 Parameter data (none) → [7][6][5][4] = 00 00 00 00
4 Activate the execution command bit C_STRT → Change from [0] = 80 to [0] =81

It means:

| | | |
|---|---|---|
| c3200 → [1] [0] → 0C80 | When bit 3200.00 turns On we have | c3300 → 0CF1 |
| c3201 → [3] [2] → 0000 | | c3301 → 0000 |
| c3202 → [5] [4] → 0000 | | c3302 → 9999 |
| c3203 → [7] [6] → 0000 | | c3303 → 9999 |

**3.2.5. - NSxx UNIT RESEST**

This command reset the servodriver and the NSxx (it disconnects the power supply for a while). Before the reset command all the RAM parameters are write to the FLASH RAM.

PROFIBUS-DP

1. Change the command code (1110b = Eh) → [1] = 0E
2. Parameter number (none) → [2][3] = 00 00
3. Parameter data (none) → [4][5][6][7] = 00 00 00 00
4. Activate the execution command bit C_STRT → Change from [0]= 80 to [0]=81

It means:

| | |
|---|---|
| c050 → [0] [1] → 800E | When bit 050.08 turns ON the unit make the reset |
| c051 → [2] [3] → 0000 | |
| c052 → [4] [5] → 0000 | |
| c053 → [6] [7] → 0000 | |

DEVICE-NET

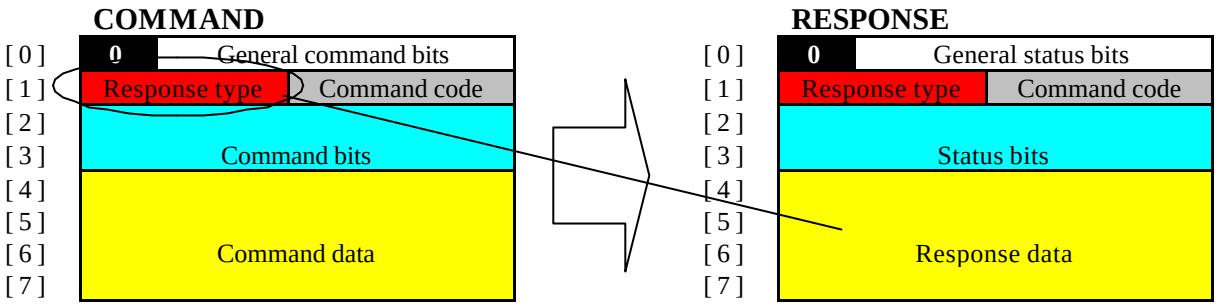
1. Change the command code (1110b = Eh) → [1] = 0E
2. Parameter number (none) → [3][2] = 00 00
3. Parameter data (none) → [7][6][5][4] = 00 00 00 00
4. Activate the execution command bit C_STRT → Change from [0]= 80 to [0]=81

It means:

| | |
|---|---|
| c3200 → [1] [0] → 0E80 | When bit 3200.00 turns ON the unit make the reset |
| c3201 → [3] [2] → 0000 | |
| c3202 → [5] [4] → 0000 | |
| c3203 → [7] [6] → 0000 | |

## 3.3. - MOVEMENT COMMANDS

This kind of commands are used to make servo movements, the format of this messages is as follows.




- The command codes are this ones:

| | |
|---|---|
| **0000** | **No operation** |
| **0001** | **Simple positioning** |
| **0010** | **External positioning** |
| **0011** | **Positioning with notch signal output** |
| **0100** | **Multi-speed positioning** |

- The **response type** define the kind of data that will be receive in the response data message:

| | |
|---|---|
| 0000 | Command position |
| 0001 | Present position |
| 0010 | Position error |
| 0011 | Command speed (1000 command unit) |
| 0100 | Present speed (1000 command unit) |
| 0101 | Torque (%) |
| 1010 | Station number |
| 1011 | Point table number |

- **Command bits**

| Byte | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
|---|---|---|---|---|---|---|---|---|
| [2] | HOME | PTBL | STN | STEP | FEED | 0 | HOLD | CANCEL |
| [3] | 0 | 0 | 0 | 0 | 0 | 0 | DIR | INC |

Where:

**CANCEL** Stops the positioning execution (deceleration to zero) and also reset the stored command position.

**HOLD** Stops the positioning execution (deceleration to zero) and wait for operation restart.

**FEED** While this bit is ON the motor run with the speed selected in command data or the feed speed depending of the selection in parameter Pn830, the rotation direction is fixed by status of DIR bit.

**STEP** This command executes a positioning with a STEP distance selected in the command data that correspond with parameters Pn844/../Pn847 and is make in the DIR direction

**STN** Station command that executes one positioning to the desired station in the DIR direction is possible to use incremental or absolute movements with the INC bit.

**PTBL** Execute a point table positioning.

**HOME** Execute an origin search command

**INC** This bit indicates if the coordinate system will be absolute or incremental when the STN and PTBL commands are used.

**DIR** This bit indicates the movement direction for FEED, STEP and STN commands.

**Status bits**

| Byte | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
|---|---|---|---|---|---|---|---|---|
| [2] | HOME_R | PTBL_R | STN_R | STEP_R | FEED_R | 0 | HOLD_R | PRGS |
| [3] | POT | NOT | INPOS | NEAR | HOME_P | | DIR_R | INC_R |

Where:

**PRGS** Flag that indicates operation in progress (active during a positioning)
**HOLD_R** Flag that turns on when the HOLD command is received in the NSxx board.
**FEED_R** Flag that turns on when the FEED command has been received, but it doesn't indicate that a movement is in process.
**STEP_R** Flag that turns on when STEP command is active and turns OFF when the command has finished.
**STN_R** Flag that turns on when STN command is active and turns OFF when the command has finished.
**PTBL_R** Flag that turns on when PTBL command is active and turns OFF when the command has finished.
**HOME_R** Flag that turns on when HOME command is active and turns OFF when the command has finished..
**INC_R** Flag that indicates that positinings are defined to work in absolute (0) or Incremental (1) coordinates.
**DIR_R** Flag that shows the rotation direction: (0) Forward / (1) Reverse.
**HOME_P** Flag that turns on when the servomotor is in the origin (range Pn806).
**NEAR** Flag that turns on when servomotor is close to the target position (range Pn852).
**INPOS** Flag that turns on when the servomotor has arrived to the target position /(INPOSITION) (range Pn850).
**NOT** Shows the negative overtravel limit input status NOT (CN1)
**POT** Shows the positive overtravel limit input status POT (CN1)

### 3.3.1 HOME COMMAND (ORIGIN SEARCH)

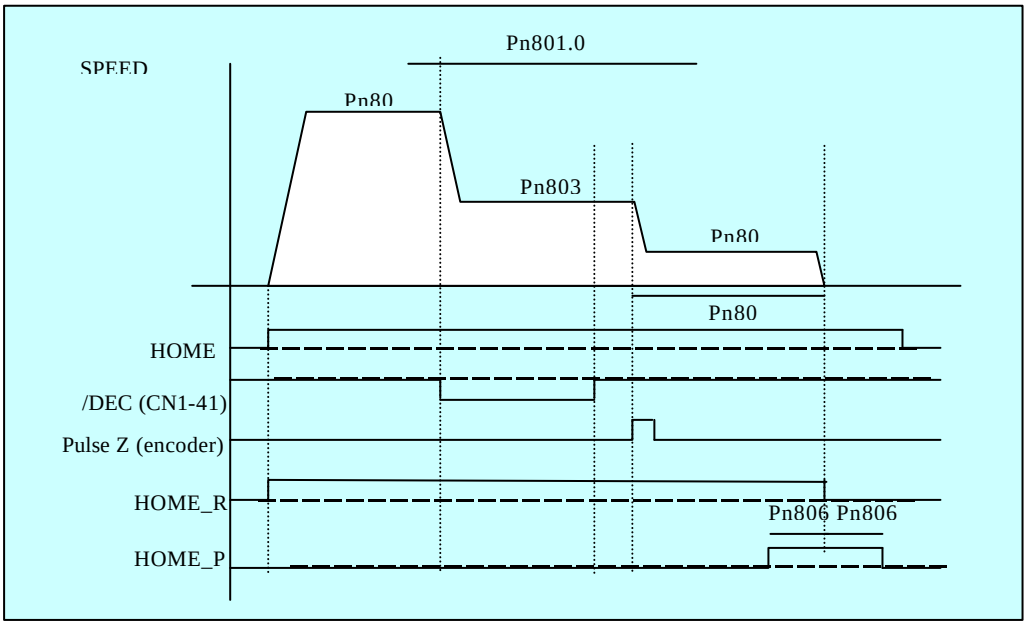
This function uses external signal to define a fix point (hardware) as origin, which will be used as a reference for any positioning.

There are 4 available methods in Pn800:

0 → 3-step deceleration using DEC signal (CN1-41) and Z phase
1 → 2-step deceleration using ZERO signals (CN1-45)
2 → 3-step deceleration using DEC (CN1-41) and ZERO (CN1-45)
3 → 2-step deceleration using Z encoder phase

In parameter Pn801.0 is possible to define the origin search direction, in the Pn801.1 the activation level for DEC signal and in the Pn801.2 if the Zero signal will work in rising or falling edge.

Pn800 = 0 (Using the DEC signal and the Z encoder phase)



When the HOME command turns on, the origin search starts in the selected direction (Pn801.0) at the origin search speed Pn802. When the rising edge of /DEC signal (CN1-41) comes, the speed decrease to the origin approach speed Pn803 and in the first pulse of Z phase after the falling edge of the /DEC signal the servomotor moves according the selected position in Pn805 (Zero point return final travel distance) at the zero point return creep speed Pn804.

The acceleration and deceleration ramp are defined in parameter Pn80A.

Also two flags are available for monitor the operation status:

HOME_R that remains active from the command start until the origin has been established.

HOME_P that turns on when servomotor arrives to the origin, with a range defined in Pn806.

## ORIGIN SEARCH PARAMETERS

| | |
|---|---|
| Pn800 | Zero point return mode |
| Pn801* | Bit 0: Origin search direction: Fwd(0) Rev(1) |
| | Bit 1: Origin proximity level /DEC |
| | Bit 2: Zero point signal level /ZERO |
| | Bit 3–15: - |
| Pn802 | Zero point return speed |

| | |
|---|---|
| Pn803 | Zero point return approach speed |
| Pn804 | Zero point return creep speed |
| Pn805 | Zero point return final travel distance |
| Pn806 | Zero point output width |
| Pn809 | Zero point offset |
| Pn80A | Zero point Accel/decel time |

This is the more complex function and corresponds to the 0 value of Pn800. With the other values 1, 2 and 3 we'll have this behaviour:

-(1) Only the ZERO signal (CN1-45), so the operation will start at the approach origin speed and in the rising edge of Zero signal the final positioning distance will be done.

- (2) /DEC and ZERO signal. Is the same case that in mode (0) but the signal that is used for the final positioning is the ZERO (CN1-45) instead the Z phase pulse.

-(3) Only Z phase, the movement will start at the origin approach speed and when the first rising edge of Z phase arrives the final distance positioning will be executed.

Command execution example:

1. - Turn On the software emergency stop bit (ESTP)

PROFIBUS-DP
c050 → [0] [1] → 1000
c051 → [2] [3] → 0000
c052 → [4] [5] → 0000
c053 → [6] [7] → 0000

DEVICE-NET
c3200 → [1][0] → 0010
c3201 → [3][2] → 0000
c3202 → [5][4] → 0000
c3203 → [7][6] → 0000

2. - Turn on the servo RUN signal (SVON)

PROFIBUS-DP
c050 → [0] [1] → 1200
c051 → [2] [3] → 0000
c052 → [4] [5] → 0000
c053 → [6] [7] → 0000

DEVICE-NET
c3200 → [1][0] → 0012
c3201 → [3][2] → 0000
c3202 → [5][4] → 0000
c3203 → [7][6] → 0000

3. - Turn on the origin search bit (HOME) with the selected response type (ex.: present speed)

PROFIBUS-DP
c050 → [0] [1] → 1240
c051 → [2] [3] → 8000
c052 → [4] [5] → 0000
c053 → [6] [7] → 0000

c0350 → [0][1] → 7240
c0351 → [2][3] → 8100
c0352 → [4][5] → xxxx
c0353 → [6][7] → xxxx

Where xxxxxxx is the response value.

DEVICE-NET
c3200 → [1] [0] → 4012
c3201 → [3] [2] → 0080
c3202 → [5] [4] → 0000
c3203 → [7] [6] → 0000

c3300 → [1][0] → 4062
c3301 → [3][2] → 0081
c3302 → [5][4] → xxxx
c3303 → [7][6] → xxxx

Where xxxxxxx is the response value.

### 3.3.2 FEED COMMAND

While the FEED command is active the motor will run indefinitely in the direction selected in DIR bit and at the speed selected in parameter Pn831 (Constant feed speed)

The acceleration and deceleration ramps are defined by parameters Pn829, Pn832 and Pn843

The speed could be changed by the override value if the FEED bit remains activated.

Also is possible to define the speed depending on Pn830 configuration:

- Pn830 = 1 → Parameter data is take as speed for feeding in 1000 reference units/min.
- Pn830 = 0 → Parameter data is take as override (0% to 200% of the FEED speed selected in Pn831), but never up to the maximum speed Pn843

The parameters used are these ones:

| | |
|---|---|
| Pn829 | Filter selection acc/dec(-, exp., exp. bias., average) |
| Pn830 | Bit 0: Override (0), direct speed in command message (1) |
| Pn831 | Constant FEED speed |
| Pn832 | Acceleration time for constant FEED |
| Pn833 | Deceleration time for constant FEED |
| Pn834 | Switch speed for constant feed second Acc/dec |

| | |
|---|---|
| Pn835 | Acc/dec time for constant feed second Acc/dec |
| Pn836 | Acc/dec type for constant feed(-,single,double,asy) |
| Pn840 | Time constant for exponential Acc/dec |
| Pn841 | Bias speed for exponential Acc/dec |
| Pn842 | Time constant of travelling average |
| Pn843 | Maximum feed speed |

Command execution example

1. - Turn on the software emergency stop bit (ESTP)

| | | | |
|---|---|---|---|
| | c050 → [0] [1] → 1000 | c3200 → [1][0] → 0010 | |
| PROFIBUS-DP | c051 → [2] [3] → 0000 | c3201 → [3][2] → 0000 | DEVICE-NET |
| | c052 → [4] [5] → 0000 | c3202 → [5][4] → 0000 | |
| | c053 → [6] [7] → 0000 | c3203 → [7][6] → 0000 | |

2. - Turn on the servomotor RUN bit (SVON)

| | | | |
|---|---|---|---|
| | c050 → [0] [1] → 1200 | c3200 → [1][0] → 0012 | |
| PROFIBUS-DP | c051 → [2] [3] → 0000 | c3201 → [3][2] → 0000 | DEVICE-NET |
| | c052 → [4] [5] → 0000 | c3202 → [5][4] → 0000 | |
| | c053 → [6] [7] → 0000 | c3203 → [7][6] → 0000 | |

3. - Select the rotation direction with DIR
   Byte [3] = 0000 0000 forward
   Byte [3] = 0000 0010 reverse.
   Select the speed
   [4][5][6][7] (PROFIBUS-DP)
   [7][6][5][4] (DEVICE-NET)
   Always in hexadecimal and depending of Pn830 value.
   Select the response type (ex.: present speed [1] = 0100 000)

If Pn830 =1 (x1000 units/min), Pn810 = 1 y Pn811 = 1, we have a 2048ppr encoder and the desired speed is 2000 rpm, we should select:

1 rev = 2048p x 4 = 8192p → 2000 rev = 8192 x 2000 = 16384000. Speed = 16384 (x1000/min) = 4000h

| | | | |
|---|---|---|---|
| | c050 → [0] [1] → 1200 | c3200 → [1][0] → 0012 | |
| PROFIBUS-DP | c051 → [2] [3] → 0000 | c3201 → [3][2] → 0000 | DEVICE-NET |
| | c052 → [4] [5] → 0000 | c3202 → [5][4] → 4000 | |
| | c053 → [6] [7] → 4000 | c3203 → [7][6] → 0000 | |

If Pn830 = 0 (override over Pn831), Pn831 = 24000 (x1000/min) and the desired speed is 2000rpm:

2000rpm = 16384000 / 24000000 = 0.682 → 68% = 44h → [4][5][6][7] = 0000 0044 en PROFIBUS

[7][6][5][4] = 0044 0000 en DEVICE-NET

| PROFIBUS-DP | | DEVICE-NET | |
|---|---|---|---|
| c050 → [0] [1] → 1200 | | c3200 → [1][0] → 0012 | |
| c051 → [2] [3] → 0000 | | c3201 → [3][2] → 0000 | |
| c052 → [4] [5] → 0000 | | c3202 → [5][4] → 0044 | |
| c053 → [6] [7] → 0044 | | c3203 → [7][6] → 0000 | |

4. - Turn on the FEED bit

Byte [2] = 0000 1000 to execute the command.

| PROFIBUS-DP | |
|---|---|
| c050 → [0] [1] → 1240 | c0350 → [0][1] → 6240 |
| c051 → [2] [3] → 0800 | c0351 → [2][3] → 0900 |
| c052 → [4] [5] → 0000 | c0352 → [4][5] → xxxx |
| c053 → [6] [7] → 4000 | c0353 → [6][7] → xxxx |

Where xxxxxxxx is the response value.

| DEVICE-NET | |
|---|---|
| c3200 → [1] [0] → 4012 | c3300 → [1][0] → 4062 |
| c3201 → [3] [2] → 0008 | c3301 → [3][2] → 0009 |
| c3202 → [5] [4] → 0000 | c3302 → [5][4] → xxxx |
| c3203 → [7] [6] → 0000 | c3303 → [7][6] → xxxx |

Where xxxxxxxx is the response value.

### 3.3.3 STEP COMMAND

While the STEP command bit is ON, the motor will run in the desired direction DIR at the specific speed Pn821 during the step distance specify by Pn844 (STEP1), Pn845 (STEP2), Pn846 (STEP3) or Pn847 (STEP4).

The acceleration and deceleration ramps are defined in parameters Pn822 to Pn829 and Pn840 to Pn843

It's not possible to change the speed that is fixed in parameter Pn821

The related parameters are these ones:

| | |
|---|---|
| Pn821 | Feed speed for positioning |
| Pn822 | Acceleration time for positioning |
| Pn823 | Deceleration time for positioning |
| Pn824 | Switch speed for second Acc/dec for positioning |
| Pn825 | Acc/dec time for second Acc/dec for positioning |
| Pn826 | Acc/dec type for positioning |
| Pn829 | Filter selection |
| Pn840 | Time constant for exponential Acc/dec |

| | |
|---|---|
| Pn841 | Bias speed for exponential Acc/dec |
| Pn842 | Time constant of travelling average |
| Pn843 | Maximum feed speed |
| Pn844 | Step distance 1 |
| Pn845 | Step distance 2 |
| Pn846 | Step distance 3 |
| Pn847 | Step distance 4 |

Command execution example:

1. - Turn on the software emergency stop (ESTP)

| | | | |
|---|---|---|---|
| | c050 → [0] [1] → 1000 | c3200 → [1][0] → 0010 | |
| PROFIBUS-DP | c051 → [2] [3] → 0000 | c3201 → [3][2] → 0000 | |
| | c052 → [4] [5] → 0000 | c3202 → [5][4] → 0000 | DEVICE-NET |
| | c053 → [6] [7] → 0000 | c3203 → [7][6] → 0000 | |

2. - Turn on the servomotor RUN bit (SVON)

| | | | |
|---|---|---|---|
| | c050 → [0] [1] → 1200 | c3200 → [1][0] → 0012 | |
| | c051 → [2] [3] → 0000 | c3201 → [3][2] → 0000 | |
| PROFIBUS-DP | c052 → [4] [5] → 0000 | c3202 → [5][4] → 0000 | DEVICE-NET |
| | c053 → [6] [7] → 0000 | c3203 → [7][6] → 0000 | |

3. - Select the motor rotation direction DIR

Byte [3] = 0000 0000 forward, [3] = 0000 0010 reverse.

Select the step distance

[4][5][6][7] 0000 0000 (STEP1) to 0000 0003 (STEP4) PROFIBUS-DP

[7][6][5][4] 0000 0000 (STEP1) to 0000 0003 (STEP4) DEVICE-NET

Select the response type (ex.: present position [1] = 0001 000)

| | | | |
|---|---|---|---|
| | c050 → [0] [1] → 1210 | c3200 → [1][0] → 1012 | |
| | c051 → [2] [3] → 0000 | c3201 → [3][2] → 0000 | |
| PROFIBUS-DP | c052 → [4] [5] → 0000 | c3202 → [5][4] → 0002 | DEVICE-NET |
| | c053 → [6] [7] → 0002 | c3203 → [7][6] → 0000 | |

4. - Turn on the STEP bit byte [2] = 0001 0000 to execute the command

| | | |
|---|---|---|
| PROFIBUS-DP | c050 → [0] [1] → 1210 | c0350 → [0][1] → 6210 |
| | c051 → [2] [3] → 1000 | c0351 → [2][3] → 1100 |
| | c052 → [4] [5] → 0000 | c0352 → [4][5] → xxxx |
| | c053 → [6] [7] → 0002 | c0353 → [6][7] → xxxx |

Where xxxxxxx is the presesnt position.

| | | |
|---|---|---|
| DEVICE-NET | c3200 → [1] [0] → 1012 | c3300 → [1][0] → 1062 |
| | c3201 → [3] [2] → 0010 | c3301 → [3][2] → 3010 |
| | c3202 → [5] [4] → 0002 | c3302 → [5][4] → xxxx |
| | c3203 → [7] [6] → 0000 | c3303 → [7][6] → xxxx |

Where xxxxxxx is the presesnt position

### 3.3.4 POINT TABLE COMMAND

This command executes a positioning stored in the point table (Pn900 to Pn971) when the PTBL bit is selected. In these tables is possible to store the position and the speed for each position. Also is possible to select the direction DIR and the coordinate type (INC or ABS).

The acceleration and deceleration ramps are defined in parameters Pn822 to Pn826, Pn829, and Pn840 to Pn843.

The related parameters are these ones:

| | |
|---|---|
| Pn822 | Acceleration time for positioning |
| Pn823 | Deceleration time for positioning |
| Pn824 | Switch speed for second Acc/dec for positioning |
| Pn825 | Acc/dec time for 2nd Acc/dec positioning |
| Pn826 | Accel/dec type for positioning |

| | |
|---|---|
| Pn829 | Filter selection |
| Pn840 | Time constant for exponential Acc/dec |
| Pn841 | Bias speed for exponential Acc/dec |
| Pn842 | Time constant of travelling average |
| Pn843 | Maximum feed speed |

And the positioning table:

| | |
|---|---|
| Pn900 | Target position 1 |
| Pn901 | Target position 2 |
| … | … |
| Pn92F | Target position 48 |
| Pn930 | Target position 49 |
| Pn931 | Target position 50 |
| Pn940 | Target speed 1 |
| Pn941 | Target speed 2 |
| … | … |
| Pn971 | Target speed 50 |

**All the speeds are define in 1000 steps/min**

Command execution example

1. - Turn on the software emergency stop bit (ESTP)

PROFIBUS-DP c050 → [0] [1] → 1000
DEVICE-NET c3200 → [1] [0] → 0010

2. - Turn on the servomotor RUN bit (SVON)

PROFIBUS-DP c050 → [0] [1] → 1200
DEVICE-NET c3200 → [1][0] → 0012

3.-Select the rotation direction DIR bit

Byte [3] = 0000 0000 forward, [3] = 0000 0010 reverse.

Select the coordinate system INC bit

Byte [3] = 0000 0000 ABS [3] = 0000 0001 INC

Select the response type (ex.: present position [1] = 0001 000)

Select the point table to be executed: Position 27 → [4] [5] [6] [7] = 0000 001B (PROFIBUS-DP)
[7][6][5][4] = 0000 001B (DEVICE-NET);

| PROFIBUS-DP | | DEVICE-NET |
|---|---|---|
| c050 → [0] [1] → 1210 | | c3200 → [1][0] → 1012 |
| c051 → [2] [3] → 0003 | INCremental and reverse | c3201 → [3][2] → 0300 |
| c052 → [4] [5] → 0000 | | c3202 → [5][4] → 001B |
| c053 → [6] [7] → 001B | | c3203 → [7][6] → 0000 |

Position 27 (Pn91A) and speed 27 (Pn95B)

4. - Turn on the PTBL bit byte [2] = 0100 0000 to execute the command

| | | |
|---|---|---|
| PROFIBUS-DP | c050 → [0] [1] → 1210 | c0350 → [0][1] → 6210 |
| | c051 → [2] [3] → 4003 | c0351 → [2][3] → 4103 |
| | c052 → [4] [5] → 0000 | c0352 → [4][5] → xxxx |
| | c053 → [6] [7] → 001B | c0353 → [6][7] → xxxx |

Where xxxxxxxx is the present position.

| | | |
|---|---|---|
| DEVICE-NET | c3200 → [1] [0] → 1012 | c3300 → [1][0] → 1062 |
| | c3201 → [3] [2] → 0340 | c3301 → [3][2] → 0341 |
| | c3202 → [5] [4] → 001B | c3302 → [5][4] → xxxx |
| | c3203 → [7] [6] → 0000 | c3303 → [7][6] → xxxx |

Where xxxxxxxx is the present position

### 3.3.5 SIMPLE POSITIONING

This command is used to make movements to a target position send by communications. To use it is necessary to write the command 0001 in byte [1] so it should be executed by the C_STRT bit. The coordinate type could be select by the INC bit.

The speed for this positioning is fixed to parameter Pn821 (Feed speed for positioning).

Acceleration and deceleration ramps are selected by parameter Pn822 to Pn826, Pn829, and Pn840 to Pn843.

The related parameters are these ones:

| | |
|---|---|
| Pn821 | Feed speed for positioning |
| Pn822 | Acceleration time for positioning |
| Pn823 | Deceleration time for positioning |
| Pn824 | Switch speed for 2nd Acc/dec |
| Pn825 | Acc/dec time for 2nd Acc/dec for positioning |
| Pn826 | Acc/dec type for positioning |

| | |
|---|---|
| Pn829 | Filter selection |
| Pn840 | Time constant for exponential Acc/dec |
| Pn841 | Bias speed for exponential Acc/dec |
| Pn842 | Time constant of travelling average |
| Pn843 | Maximum feed speed |

Execution command example:

1. - Turn on the software emergency stop bit (ESTP)

PROFIBUS-DP c050 → [0] [1] → 1000
DEVICE-NET c3200 → [1] [0] → 0010

2. - Turn on the servomotor RUN bit (SVON)

PROFIBUS-DP c050 → [0] [1] → 1200
DEVICE-NET c3200 → [1][0] → 0012

3.-Select the SIMPLE POSITIONING command ([1] = xxxx 0001)

Select the rotation direction DIR bit
Byte [3] = 0000 0000 forward, [3] = 0000 0010 reverse.

Select the coordinate system INC bit
Byte [3] = 0000 0000 ABS [3] = 0000 0001 INC

Select the response type (ex.: present position [1] = 0001 000)
Select the final position in hexadecimal
[4][5][6][7] PROFIBUS-DP
[7][6][5][4] DEVICE-NET
Example -10000 = FFFF D8F0;

| PROFIBUS-DP | | | DEVICE-NET |
|---|---|---|---|
| c050 → [0] [1] → 1211 | | c3200 → [1][0] → 1112 | |
| c051 → [2] [3] → 0001 | INC | c3201 → [3][2] → 0100 | |
| c052 → [4] [5] → FFFF | | c3202 → [5][4] → D8F0 | |
| c053 → [6] [7] → D8F0 | -10000 units | c3203 → [7][6] → FFFF | |

4. - Turn on the C_STRT start bit on byte [0] (change from [0] = 12 to [0] = 13) to execute the command

| PROFIBUS-DP | | When bit c0050.8 turns on | |
|---|---|---|---|
| | c050 → [0] [1] → 1211 | | c0350 → [0][1] → 7311 |
| | c051 → [2] [3] → 0001 | | c0351 → [2][3] → 0003 |
| | c052 → [4] [5] → FFFF | | c0352 → [4][5] → xxxx |
| | c053 → [6] [7] → D8F0 | | c0353 → [6][7] → xxxx |

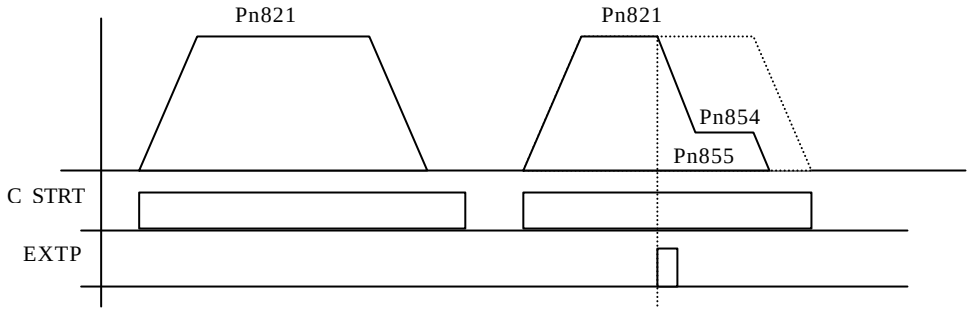
Where xxxxxxx is the present posititon.

| DEVICE-NET | | When bit c3200.00 turns on | |
|---|---|---|---|
| | c3200 → [1] [0] → 1112 | | c3300 → [1][0] → 1162 |
| | c3201 → [3] [2] → 0100 | | c3301 → [3][2] → 3100 |
| | c3202 → [5] [4] → D8F0 | | c3302 → [5][4] → xxxx |
| | c3203 → [7] [6] → FFFF | | c3303 → [7][6] → xxxx |

Where xxxxxxx is the present position

### 3.3.6 EXTERNAL POSITIONING

This command executes a movement to the desired position in the command parameters at the selected speed (Pn821). If the EXTP (CN1-44) turns on during the positioning the motor will decelerate to the approach speed (Pn854) and it will move the travel positioning distance (Pn855) from the External positioning signal.
The 0010 command in byte [1] should be used and the C_STRT bit it's needed to start the movement. Also the coordinate type (ABS or INC) could be defined in INC byte [3].




The acceleration and deceleration ramps are defined in parameters Pn822 to Pn826, Pn829, and Pn840 to Pn843.

The related parameters are these ones:

| | |
|---|---|
| Pn821 | Feed speed for positioning |
| Pn822 | Acceleration time for positioning |
| Pn823 | Deceleration time for positioning |
| Pn824 | Switch speed for 2nd Acc/dec for positioning |
| Pn825 | Acc/dec time for 2nd Acc/dec for positioning |
| Pn826 | Acc/dec type for positioning |
| Pn829 | Filter selection |

| | |
|---|---|
| Pn840 | Time constant for exponential Acc/dec |
| Pn841 | Bias speed for exponential Acc/dec |
| Pn842 | Time constant of travelling average |
| Pn843 | Maximum feed speed |
| Pn854 | Approach speed for external positioning |
| Pn855 | Travel distance for external positioning |
| | |

Command execution example

1. - Turn on the software emergency stop bit (ESTP)

PROFIBUS-DP c050 → [0] [1] → 1000
DEVICE-NET c3200 → [1] [0] → 0010

2. - Turn on the servomotor RUN bit (SVON)

PROFIBUS-DP c050 → [0] [1] → 1200
DEVICE-NET c3200 → [1][0] → 0012

3. - Select the external positioning code ([1] = xxxx 0010)
Select the coordinate type using the INC bit ([3] = 0000 0000 ABS, [3] = 0000 0001 INC)
Select the response type (ex.: present position [1] = 0001 000)
Select the target position in hex [4] [5] [6] [7] (PROFIBUS-DP) or [7] [6] [5] [4] (DEVICE-NET).
Ex 245670 = 0003 BFA6;

| PROFIBUS-DP | | DEVICE-NET |
|---|---|---|
| c050 → [0] [1] → 1212 | | c3200 → [1][0] → 1212 |
| c051 → [2] [3] → 0001 | INC | c3201 → [3][2] → 0100 |
| c052 → [4] [5] → 0003 | | c3202 → [5][4] → BFA6 |
| c053 → [6] [7] → BFA6 | 245670 unidades | c3203 → [7][6] → 0003 |

4. - Turn on the C_STRT bit byte [0] (go from [0] = 12 to [0] = 13) to execute the command

PROFIBUS-DP

| | | |
|---|---|---|
| c050 → [0] [1] → 1312 | When c0050.8 turns on | c0350 → [0][1] → 7312 |
| c051 → [2] [3] → 0001 | | c0351 → [2][3] → 0001 |
| c052 → [4] [5] → 0003 | | c0352 → [4][5] → xxxx |
| c053 → [6] [7] → BFA6 | | c0353 → [6][7] → xxxx |

where xxxxxxx is the present position.

DEVICE-NET

| | | |
|---|---|---|
| c3200 → [1] [0] → 1213 | When c3200.00 turn on | c3300 → [1][0] → 1263 |
| c3201 → [3] [2] → 0100 | | c3301 → [3][2] → 3100 |
| c3202 → [5] [4] → BFA6 | | c3302 → [5][4] → xxxx |
| c3203 → [7] [6] → 0003 | | c3303 → [7][6] → xxxx |

Where xxxxxxx is the present position.

### 3.3.7 POSITIONING WITH NOTCH SIGNALS

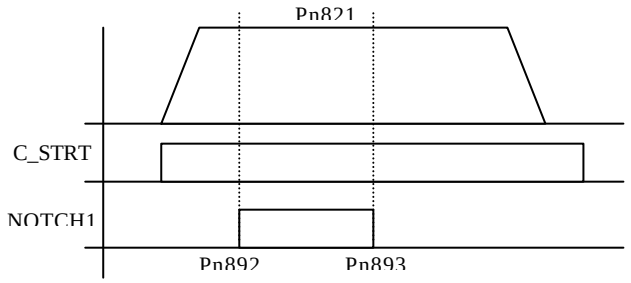
This command executes one movement to the specified point and the preset speed (Pn821) like in a simple positioning. The only difference is referred with the present position, if the actual position is between some levels defined in parameters Pn892 and Pn893 for Notch1 (CN4-12/13) or Pn894 and Pn895 for Notch2 (CN-10/20) this output will turn on. The command code is 0011 in byte [1] and movement starts with the C_STRT bit. The coordinate type (ABS or INC) is defined in INC byte [3].

The acceleration and deceleration ramps are defined in parameters Pn822 to Pn826, Pn829, and Pn840 to Pn843.

The related parameters are these ones:

| | |
|---|---|
| Pn821 | Feed speed for positioning |
| Pn822 | Acceleration time for positioning |
| Pn823 | Deceleration time for positioning |
| Pn824 | Switch speed for 2nd Acc/dec for positioning |
| Pn825 | Acc/dec time for 2nd Acc/dec for positioning |
| Pn826 | Acc/dec type for positioning |
| Pn829 | Filter selection |
| Pn840 | Time constant for exponential Acc/dec |
| Pn841 | Bias speed for exponential Acc/dec |
| Pn842 | Time constant of travelling average |
| Pn843 | Maximum feed speed |

| PARAMETERS RELATED WITH NOTCH | |
|---|---|
| Pn890 | Notch signal output position setting |
| Pn891 | Bit 0: Enable the NOTCH1 output |
| | Bit 1: Disable the NOTCH2 output |
| Pn892 | Notch 1 lower limit |
| Pn893 | Notch 1 upper limit |
| Pn894 | Notch 2 lower limit |
| Pn895 | Notch 2 upper limit |




Command execution example

1. - Turn on the software emergency stop (ESTP)

| | |
|---|---|
| PROFIBUS-DP | c050 → [0] [1] → 1000 |
| DEVICE-NET | c3200 → [1] [0] → 0010 |

2. - Turn on the servomotor RUN bit (SVON)

| | |
|---|---|
| PROFIBUS-DP | c050 → [0] [1] → 1200 |
| DEVICE-NET | c3200 → [1][0] → 0012 |

3. - Select the positioning with notch signal outputs command code ([1] = xxxx 0011)
Select the coordinate type using the INC bit ([3] = 0000 0000 ABS, [3] = 0000 0001 INC)
Select the response type (ex.: present position [1] = 0001 000)
Select the target position in hex [4] [5] [6] [7] (PROFIBUS-DP) or [7] [6] [5] [4] (DEVICE-NET).
Example 200000 = 0003 0D40;

| PROFIBUS-DP | | | DEVICE-NET |
|---|---|---|---|
| c050 → [0] [1] → 1213 | | c3200 → [1][0] → 1212 | |
| c051 → [2] [3] → 0001 | INC | c3201 → [3][2] → 0100 | |
| c052 → [4] [5] → 0003 | | c3202 → [5][4] → BFA6 | |
| c053 → [6] [7] → 0D40 | 200000 units | c3203 → [7][6] → 0003 | |

4. - Turn on the C_STRT bit byte [0] (change from [0] = 12 to [0] = 13) to execute the command

| | | | |
|---|---|---|---|
| PROFIBUS-DP | c050 → [0] [1] → 1313 | When c0050.8 turn on | c0350 → [0][1] → 7313 |
| | c051 → [2] [3] → 0001 | | c0351 → [2][3] → 0001 |

c052 → [4] [5] → 0003 c0352 → [4][5] → xxxx
c053 → [6] [7] → 0D40 c0353 → [6][7] → xxxx
where xxxxxxx is the present position.

DEVICE-NET c3200 → [1] [0] → 1313 when c3200.00 turns on c3300 → [1][0] → 1363
c3201 → [3] [2] → 0100 c3301 → [3][2] → 3100
c3202 → [5] [4] → 0D40 c3302 → [5][4] → xxxx
c3203 → [7] [6] → 0003 c3303 → [7][6] → xxxx
Where xxxxxxx is the present position

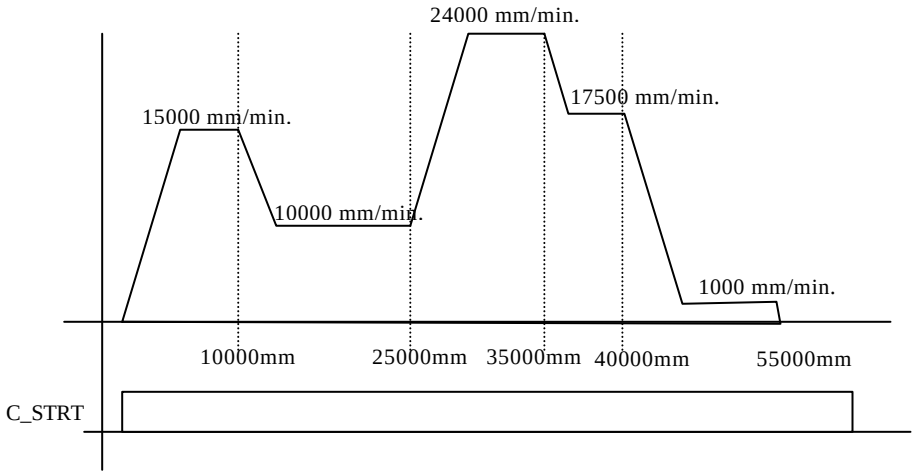
### 3.3.8 MULTI-SPEED POSITIONING

This command executes a positioning to the target point with an initial speed defined in Pn861 just like in simple positioning. But if during the positioning the present position goes trough a speed change point (Pn863 to Pn872) the speed will change to the selected speed (Pn873 to Pn882) if the behaviour has been enable (Pn861).
The command code is the 0100 in byte [1] and is executed by C_STRT bit. The coordinate type (ABS or INC) is defined in INC byte [3].
The acceleration and deceleration ramps are defined in parameters Pn822 to Pn826, Pn829, and Pn840 to Pn843.
The related parameters are these ones:

| | |
|---|---|
| Pn822 | Acceleration time for positioning |
| Pn823 | Deceleration time for positioning |
| Pn824 | Switch speed for 2nd Acc/dec for positioning |
| Pn825 | Acc/dec time for 2nd Acc/dec for positioning |
| Pn826 | Acc/dec type for positioning |
| Pn829 | Filter selection |
| Pn840 | Time constant for exponential Acc/dec |
| Pn841 | Bias speed for exponential Acc/dec |
| Pn842 | Time constant of travelling average |
| Pn843 | Maximum feed speed |

| MULTI-SPEED PARAMETERS | |
|---|---|
| Pn861 | Number of points for speed switching |
| Pn862 | Initial feed speed for positioning |
| Pn863 | Speed switching position 1 |
| Pn864 | Speed switching position 2 |
| Pn865 | Speed switching position 3 |
| … | … … … … … … … |
| Pn872 | Speed switching position 16 |
| Pn873 | Switching speed 1 |
| Pn874 | Switching speed 2 |
| Pn875 | Switching speed 3 |
| … | … … … … |
| Pn882 | Switching speed 16 |

Example:




1. - Parameters that should be programmed are these ones:
   Pn861 = Number of points for speed switching = 4
   Pn862 = Initial feed speed for multi-speed positioning = 15000
   Pn863 = Speed switching position 1 = 10000
   Pn864 = Speed switching position 2 = 25000
   Pn865 = Speed switching position 3 = 35000
   Pn866 = Speed switching position 4 = 40000
   Pn873 = Switching speed 1 = 10000
   Pn874 = Switching speed 2 = 24000
   Pn875 = Switching speed 3 = 17500
   Pn876 = Switching speed 4 = 1000

2. - Turn on the software emergency stop (ESTP)

| | |
|---|---|
| PROFIBUS-DP | c050 → [0] [1] → 1000 |
| DEVICE-NET | c3200 → [1] [0] → 0010 |

3. - Turn on the servomotor RUN bit (SVON)

| | |
|---|---|
| PROFIBUS-DP | c050 → [0] [1] → 1200 |
| DEVICE-NET | c3200 → [1][0] → 0012 |

4. - Select the command code for Multi-speed positioning ([1] = xxxx 0100)
Select the coordinate type using the INC bit ([3] = 0000 0000 ABS, [3] = 0000 0001 INC)
Select the response type (ex.: present position [1] = 0001 000)
Select the target position in hex [4] [5] [6] [7] (PROFIBUS-DP) or [7] [6] [5] [4] (DEVICE-NET).
Example 55000 = 0000 D6D8;

| PROFIBUS-DP | | | DEVICE-NET |
|---|---|---|---|
| c050 → [0] [1] → 1214 | | c3200 → [1][0] → 1412 | |
| c051 → [2] [3] → 0001 | INC | c3201 → [3][2] → 0100 | |
| c052 → [4] [5] → 0000 | | c3202 → [5][4] → D6D8 | |
| c053 → [6] [7] → D6D8 | 55000 units | c3203 → [7][6] → 0000 | |

5. - Turn on the C_STRT bit byte [0] (change from [0] = 12 to [0] = 13) to execute the command

PROFIBUS-DP

| | | |
|---|---|---|
| c050 → [0] [1] → 1314 | When c0050.8 turns on | c0350 → [0][1] → 7314 |
| c051 → [2] [3] → 0001 | | c0351 → [2][3] → 0001 |
| c052 → [4] [5] → 0000 | | c0352 → [4][5] → xxxx |
| c053 → [6] [7] → D6D8 | | c0353 → [6][7] → xxxx |

Where xxxxxxx is the present position.

DEVICE-NET

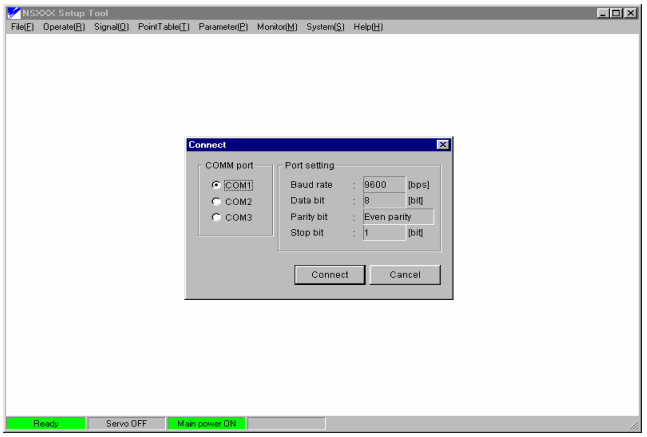
| | | |
|---|---|---|
| c3200 → [1] [0] → 1413 | When c3200.00 turns on | c3300 → [1][0] → 1363 |
| c3201 → [3] [2] → 0100 | | c3301 → [3][2] → 3100 |
| c3202 → [5] [4] → D6D8 | | c3302 → [5][4] → xxxx |
| c3203 → [7] [6] → 0000 | | c3303 → [7][6] → xxxx |

Where xxxxxxx is the present position

# 4- SOFTWARE (NSxxxSETUPTOOLS)

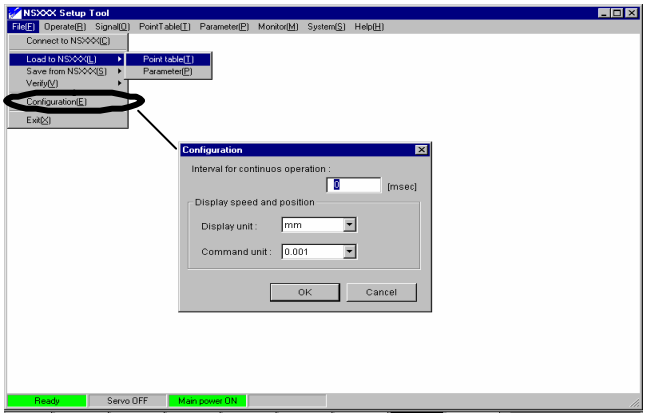
## 4.1. - INTRODUCTION

When the software starts the connection dialogue will appear. Here is possible to select the communication port where the NS500/300 is connected. The communication parameters are fixed. Also is possible to access this one from the FILE/CONNECT TO NSxxx menu.



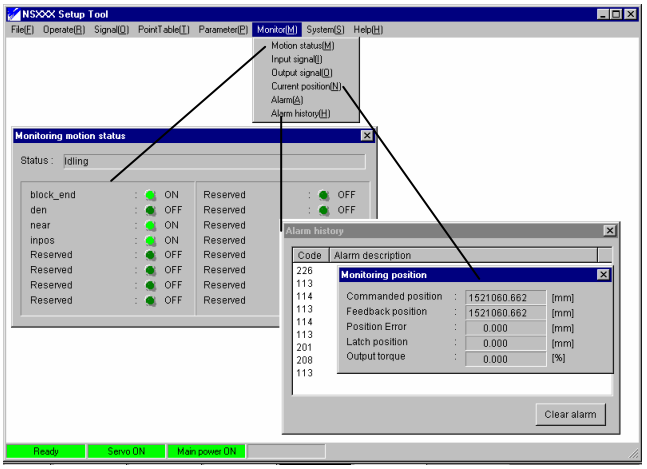

When the communication has started is possible to see some servodriver status flags like READY, SERVO ON/OFF and MAIN POWER status in the bottom of the screen.

**FILE MENU:**



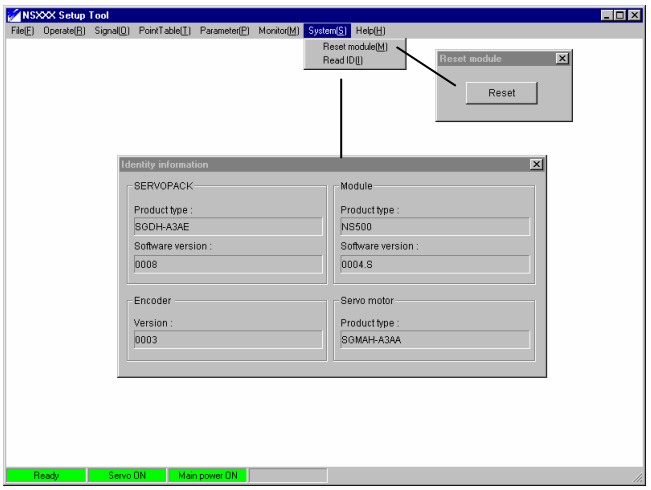

On this menu we have several options like “CONNECT TO NSxxx”. Also is possible to save parameters or point tables from a file to the NS “LOAD TO NSxxx. Also is possible to store the NS parameters or tables into a file “SAVE FROM NSxxx”, and also is possible to verify the content. In the last option is possible to configure the units that should be used in the software but this is only for monitor issues, so really is not changing any parameter in the NS

**MONITOR:**




From this option is possible to monitor the internal status flags from the NS board “MONITOR STATUS”, the external inputs and outputs status of the servodriver and option board “INPUT SIGNAL / OUTPUT SIGNAL”. Also is possible to monitor the actual position and the error counter “CURRENT POSITION”, the active alarms “ALARM” and the alarm history “ALARM HISTORY” or clear this history “CLEAR”

## SYSTEM:

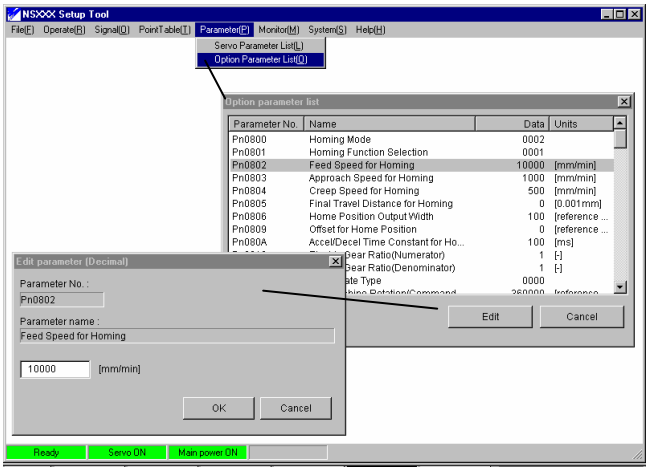



These option present two different functions:

READ ID: Reads the firmware version and model of the servodriver, NSxxx and servomotor.

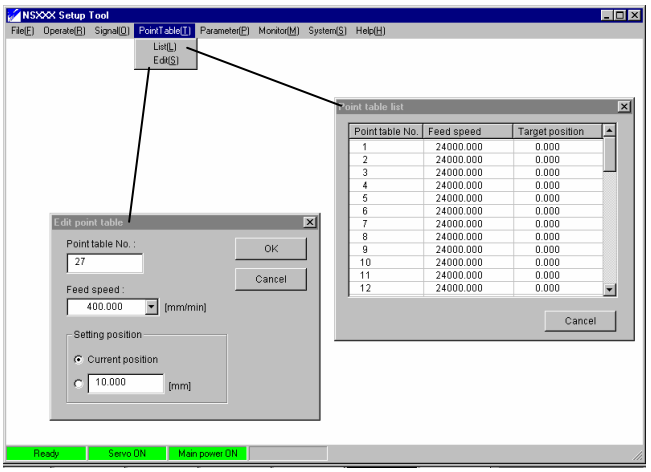
RESET MODULE: Store all the work memory RAM into the EEPROM, so this option is needed to make some of the changes active and to save the entire configuration. After the reset is made the NSxxx is initialised.

## PARAMETER



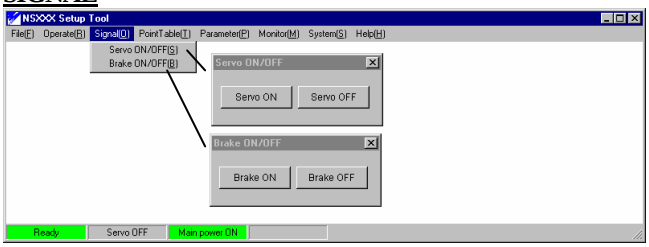

Here is possible to see or edit the module and servodriver parameter list "OPTION PARAMETER LIST" or "SERVO PARAMETER LIST" for each case.

## POINT TABLE



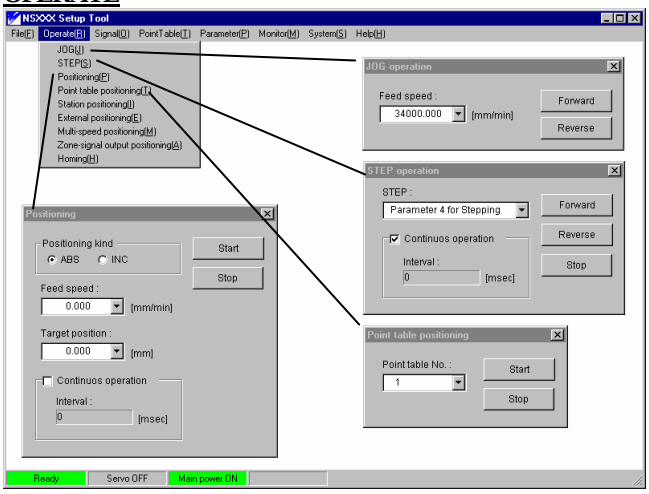

With this option is possible to see or edit all the point table parameters
Pn900 to Pn931 for positions
Pn980 to Pn9B1 for speeds.

## SIGNAL




Here is possible to start the servodriver activating the RUN (SERVO ON/OFF) bit. Also is possible to realise the brake (BRAKE ON/OFF)

## OPERATE



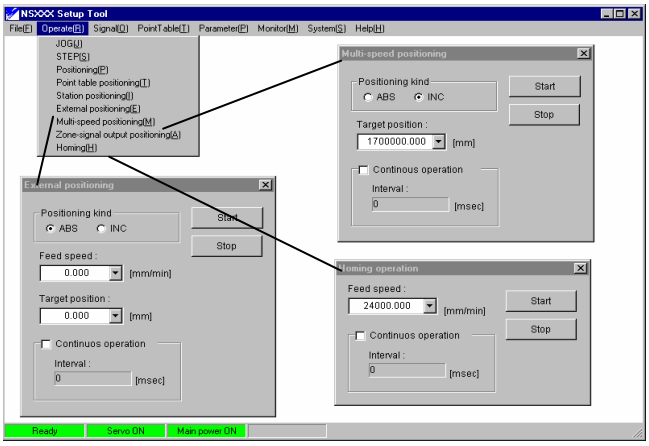

Here is possible to execute the different movements that this board allows. Several options are available:

**JOG**: Command that allows a continuous movement at the desired speed where also is possible to select the rotation direction FORWARD ó REVERSE. While the key is pressed the movement continues.

**STEP**: Command that execute a positioning with the speed and target distance fixed by parameters. Also is possible to select the rotation direction by FORWARD or REVERSE.

**POSITIONING**: Command that moves to the desired point at selected speed where also is possible to select between ABS or INC coordinates.

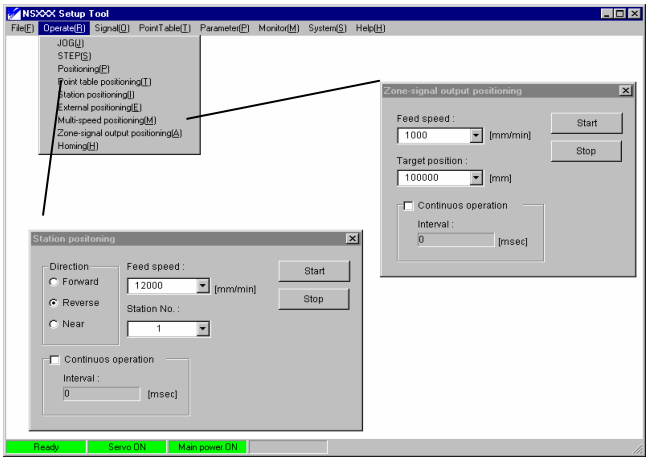
**POINT TABLE POSITIONING**: Command that executes a movement stored in one of the 50 positions table.




**MULTI-SPEED POSITIONING**: Command that executes a positioning movement to a fixed point where speed change depending of the present position. Both position and speeds are fixed by parameters.

**EXTERNAL POSITIONING**: Command to execute a positioning to a selected distance and speed in INC or ABS coordinates. But if during positioning the external signal turns on the actual movement will stop following another movement store in parameters will be performed.

**HOMING OPERATION**: Command that executes an origin search operation.




**ZONE-SIGNAL OUTPUT POSITIONING**: Command that executes a positioning with a specified speed and target position. The NOTCH signal will activate if the actual position is inside some position zones.

**STATION POSITIONING**: Command that executes a movement to different stations in which the total distance has been divided. It's possible to use this one in INC or ABS coordinates.

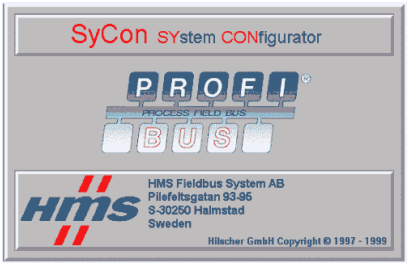
# 5- FIELD BUS CONFIGURATION

## 5.1- PROFIBUS DP CONFIGURATION

Next steps are needed to configure the Profibus DP bus:

**1- Master C200HW-PRM21 and slaves (in this case NS500) installation and wiring.**

**2- Master configuration with the SYCONDP software**



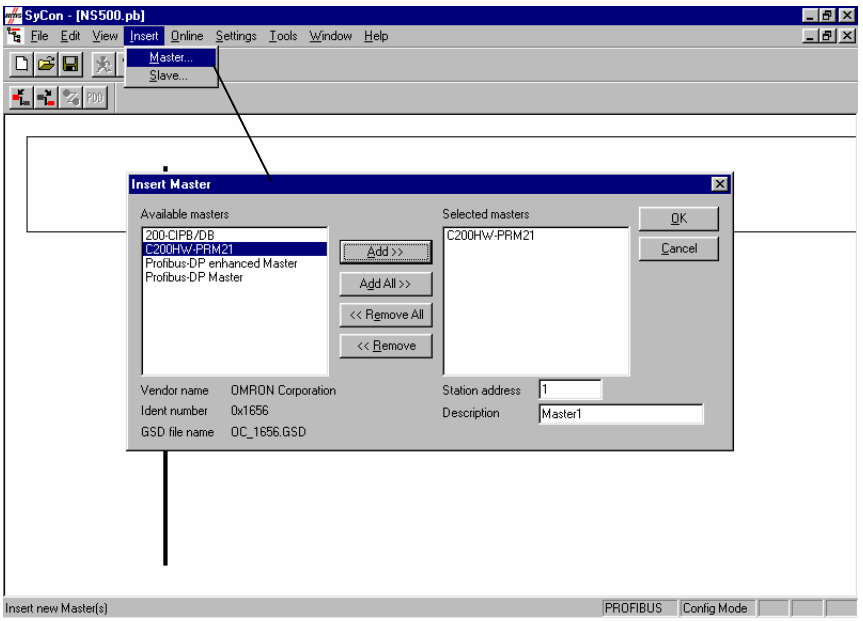

With this software is possible to supply information about the slaves connected to the bus and also configure some basic parameters. It's not possible to activate the C200HW-PRM21 master without this software.

The C200HW-PRM21 master has a special port dedicated to the configuration, and this port is the one that should be used to program the unit. The cable configuration is this one:

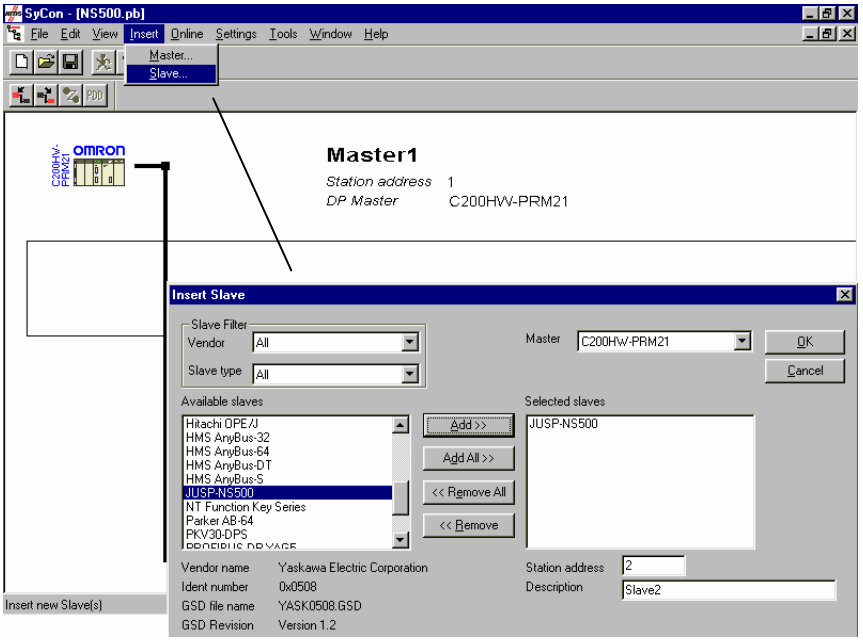
| PC | | C200HW-PRM21 |
|---|---|---|
| 2 | —— | 2 |
| 3 | —— | 3 |
| 5 | —— | 9 |

These are the step to follow: - Add the hardware components of the system (master / slaves) and define the bus configuration.
- Define the slave data allocation into the unit internal buffers.
- Define the communication parameters.
- Transfer the configuration to the Profibus master C200HW-PRM21.

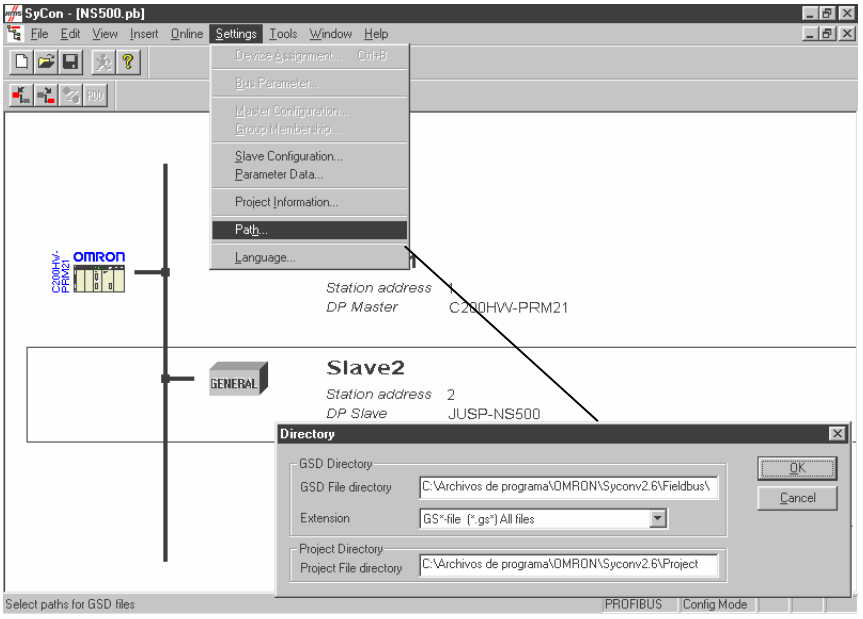
First point is add the master devices



Select the bus slaves



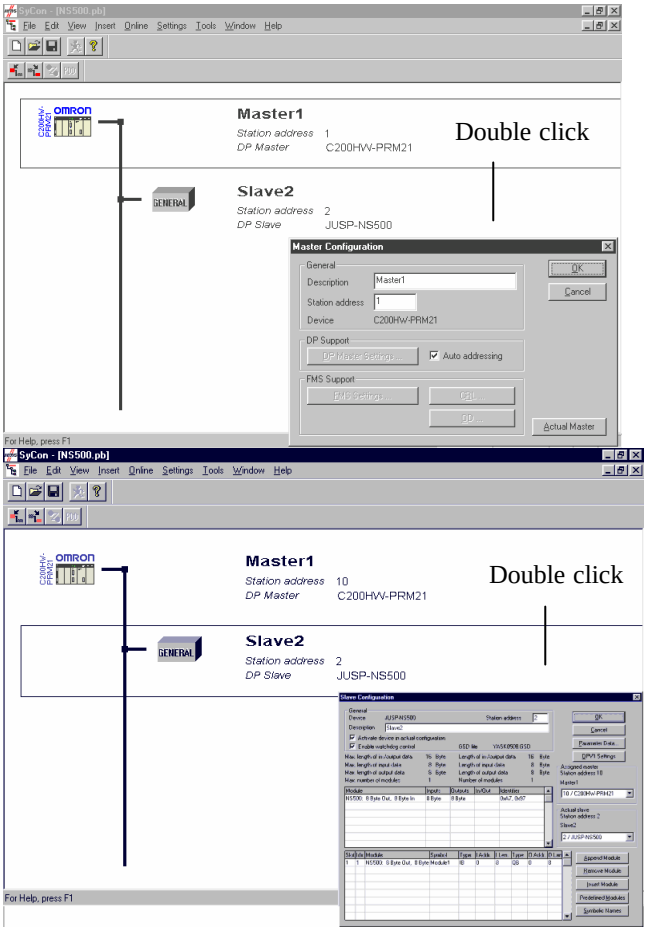

Each Profibus device has its own .GSD file that should be supplied with the device. It's necessary to copy this file into the "PATH..." directory selected in the "Settings" menu, because in any other case the device will not be listed by the software.



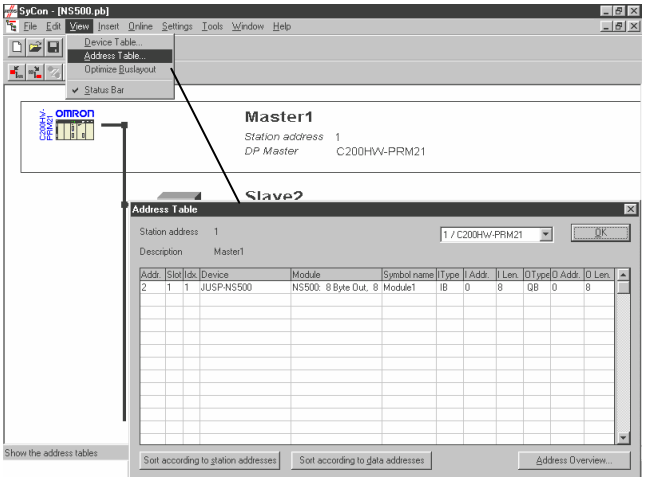

The GSD file for the NS500 is the: "YASK0508.gsd"

When the bus has been created it's necessary to define the master and slaves (in case that available) parameters like the Profibus number.

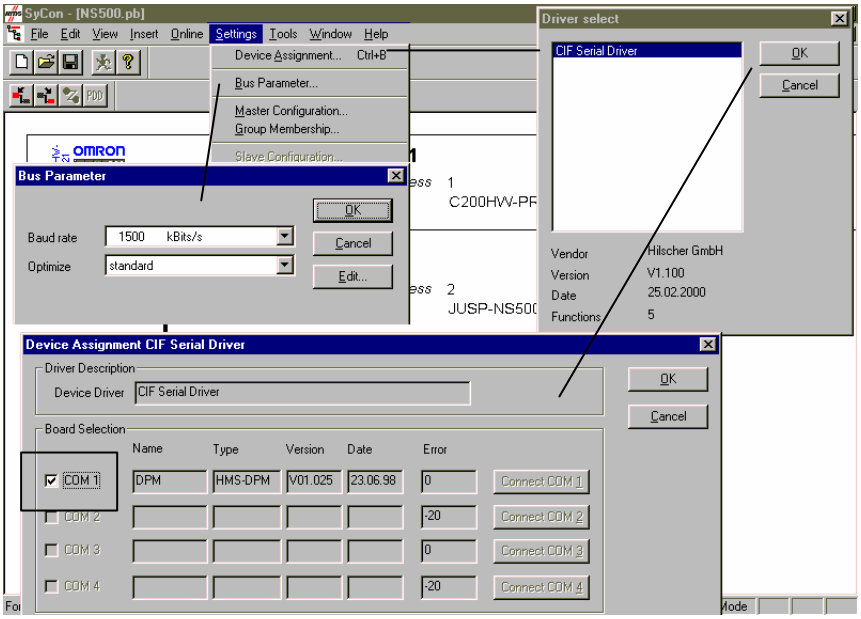
In the master only is necessary to select the unit number and the autoaddressing function. In this case no modification is needed and only it's important check the unit number in the master and slave.



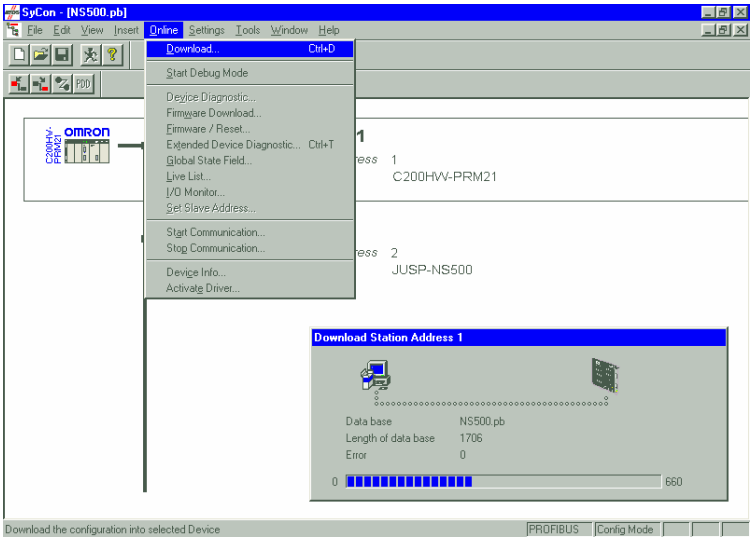

To check the address assignation the "*Address Table...*" function should be used in the "*View*" menu.



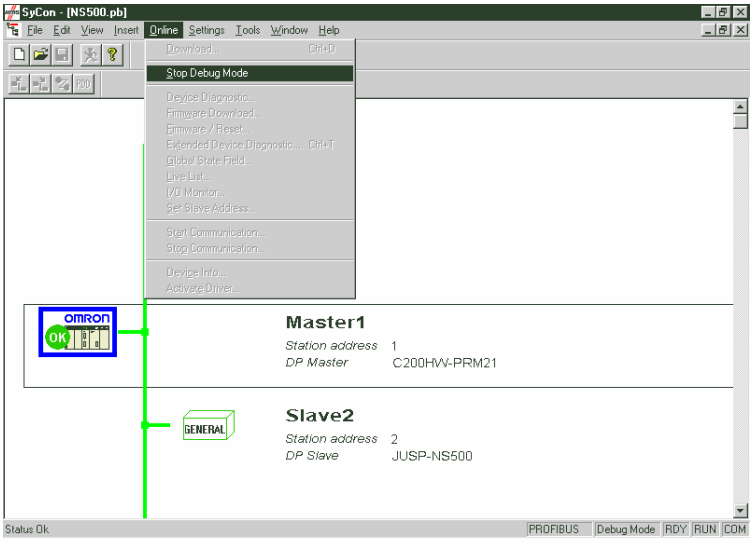

Before the download of the configuration into the master the bus communication speed should be configured in "Bus parameter" and selecting the communication port in "Device assignment".




Last point is transfer the configuration to the master




Once the configuration has been transmitted is possible to use the debug option to check that all is working and there is no problem in any slave.




3- **Start the bus communication.** To do it only is necessary to turn on the bit 0 in the 1st channel of the special unit area in the PLC:

| | | |
|---|---|---|
| - C200Halpha | (100 + (10xNºunidad)).00 | Ex.: Unit Nº = 0 → 100.00 |
| - CS1 | (2000 + (10xNºunidad)).00 | Ex.: Unit Nº = 0 → 2000.00 |

4- **Channel assignment.** Each slave has a different number of I/O bytes depending of the slave type. For the NS500 we have 8 bytes for the input and 8 bytes for the outputs, it means that 4 channels (word) for inputs and 4 for outputs are needed in the master.

**The addresses are assigned from channel c0050 (PLC→NSxxx) and from c0350 (NSxxx→PLC)**

For this example only one slave is used and the assigned channels are c0050/c0051/c0052/c0053 for outputs and c0350/c0351/c0352/c0353 as inputs.

# 5.2- DEVICE-NET CONFIGURATION

The steps that should be following to configure the device-net bus are these ones:

1. **Master C200HW-DRM21-V1 or CS1W-DRM21 and slaves (in this case NS300) installation and wiring.**

2. **Master configuration with the Device-Net software configurator.**


With this software is possible to supply information to the master about the slaves and some basic parameters of the bus.

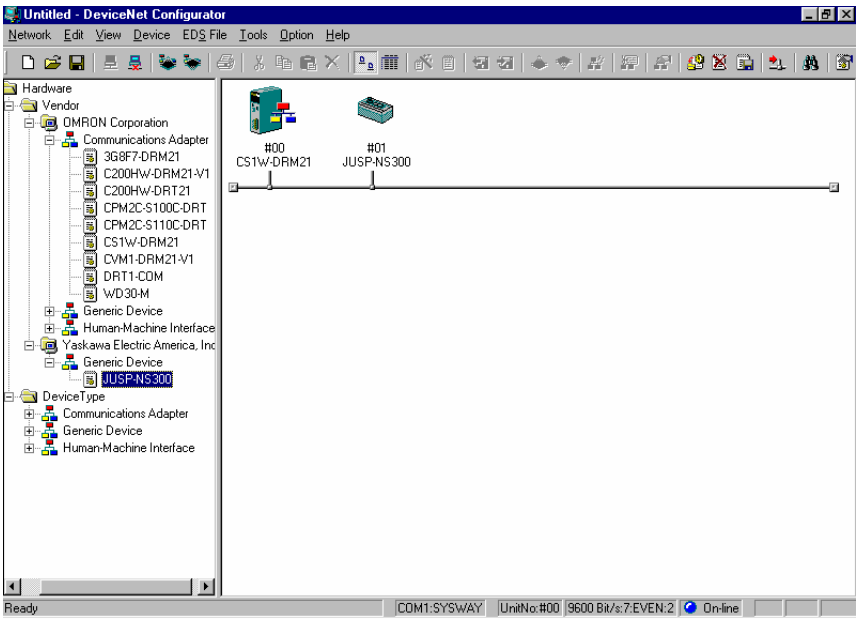
The CS1W-DRM21 master should be programmed trough the PLC communication port but for the C200HW-DRM21 the 3G8E2-DRM21 (PCMCIA to Device-Net) adaptor is needed. The cable for the CS1W-DRM21 has the same configuration than the programming cable for the PLC.

| PC | CS1W-DRM21 |
|---|---|
| 2 | 2 |
| 3 | 3 |
| 5 | 9 |

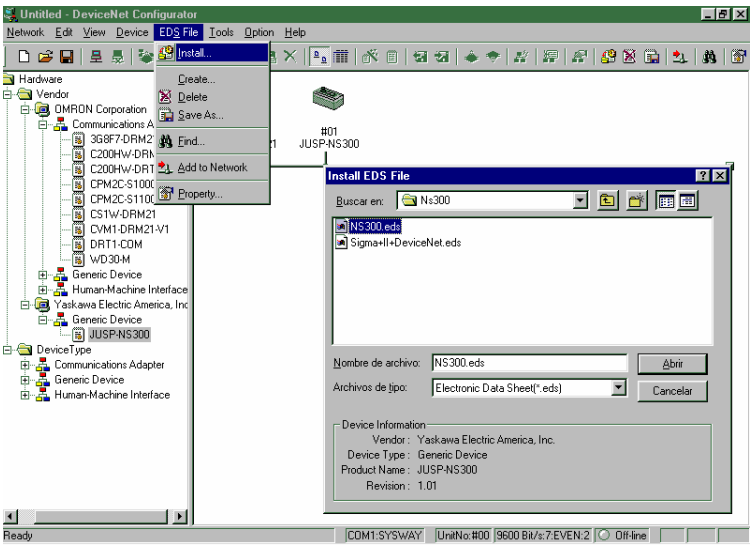
Follow these steps:

- Add all the devices that bus should contain (master / slave) and define the bus configuration.
- Define the slave data allocation into the internal buffers of the unit.
- Define the communication parameters.
- Download the configuration into the Device-net master.

First point is add the master and slaves devices



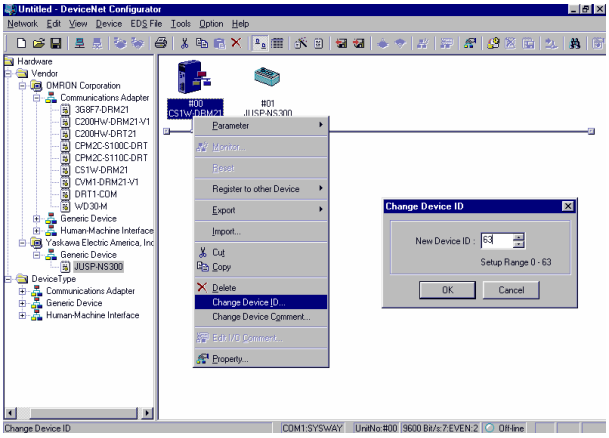
Each device has its own .EDS file that should be supply with the device. This file should be added to the software by the "INSTALL" option in the "EDS FILE" menu and selecting the desired EDS file in the corresponding folder.

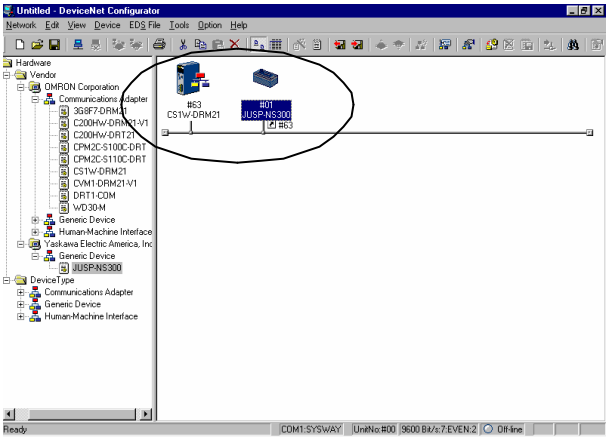



The EDS file for the NS300 is the: "NS300.EDS"

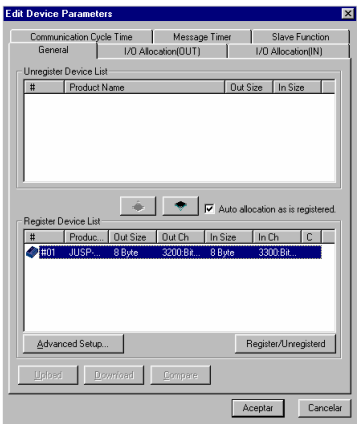
When the bus has been create is time to define the unit numbers for each bus device according to the hardware selection (Rotatory switches in the front side of slave and master).

Next step is reference each slave with a master, to do it is possible to drag the slave to the master.



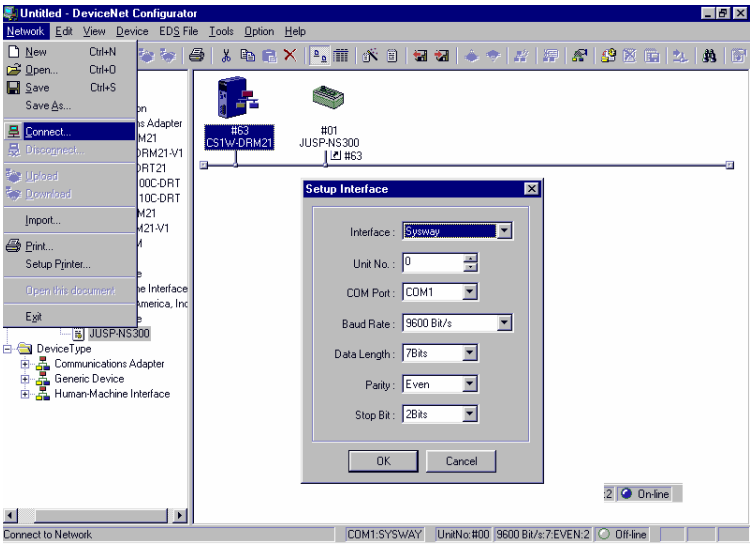

To check the assignation in the master only is necessary to double-click over the master and this window should appear:



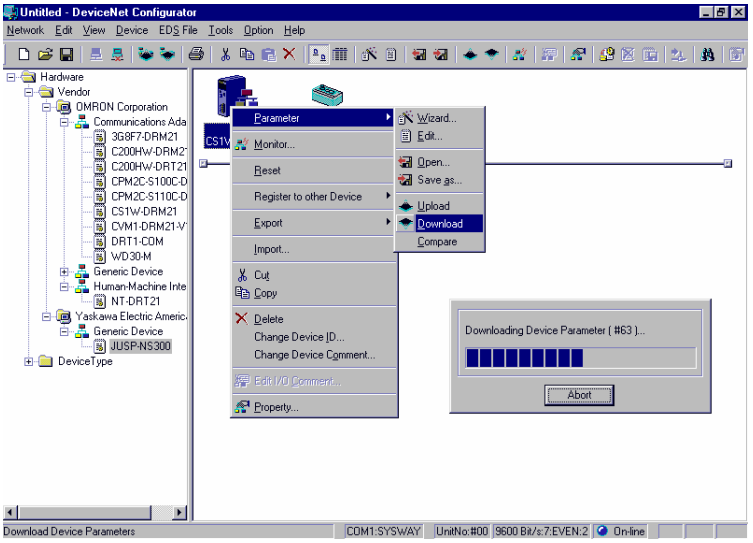

Where is possible to check the input and output interface channels between the master and the slaves.

In this case the first slave #1 (NS300) occupies:
-4 output channels (c3200, c3201, c3202 and c3203)
-4 input channels (c3300, c3301, c3302 and c3303)

When the bus has been set-up it's time to communicate with the master. To do it use the "connect" option in the "Network" menu, here is possible to select the PLC port configuration.




Final step is download the configuration, use the left mouse button over the master and a new window will appear, use the "Download" option in the "parameter" menu to start the transmission.




When all the information has been transferred the bus will start the communication, if there is no error. The master checks that the entire configuration is Ok and that all slaves are working properly.