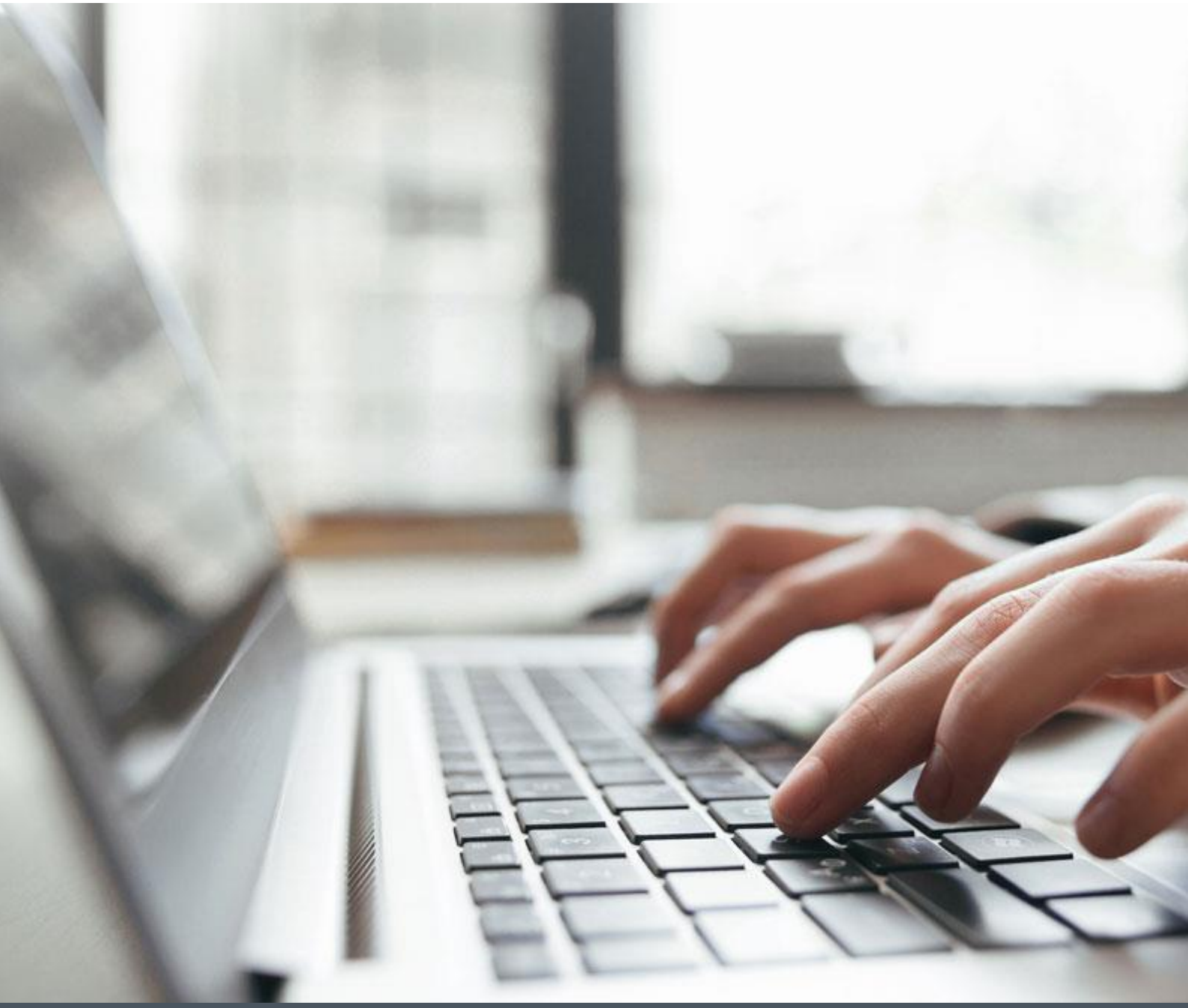


Application Note

Starting up a Nanotec Controller/Drive with IDEC SmartAXIS FT1A-H40RSA

Version 1.0.0

www.nanotec.com

Contents

1 Prerequisites

The slave drive (Nanotec Controller) must be configured beforehand. Proper operation of the motor and slave drive must be ensured before the example can be used. Make sure that the controller/drive operation is not hindered, e.g. by a stand-alone program running on the slave.

2 Hardware

IDEC SmartAXIS FT1A-H40RSA, System Software Version 2.30
N5-2-4, Firmware version FIR-v1825-B577172

3 Software

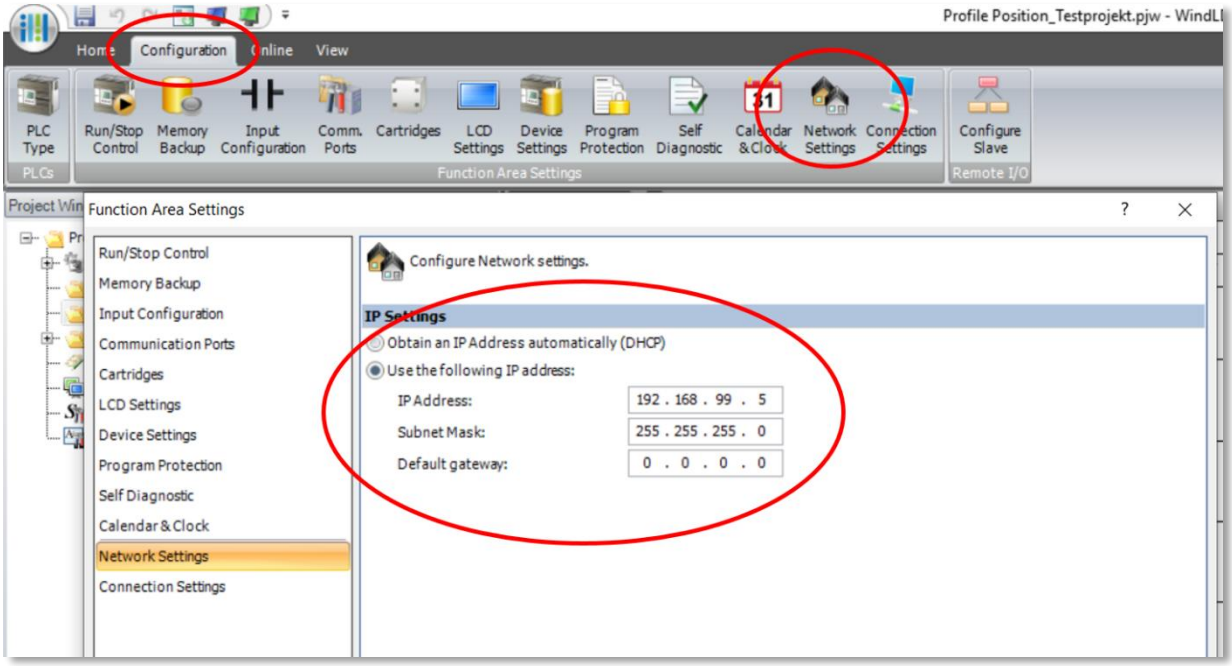
WindLDR, Version 8.12.0

4 Creating a new project and configuring the communication

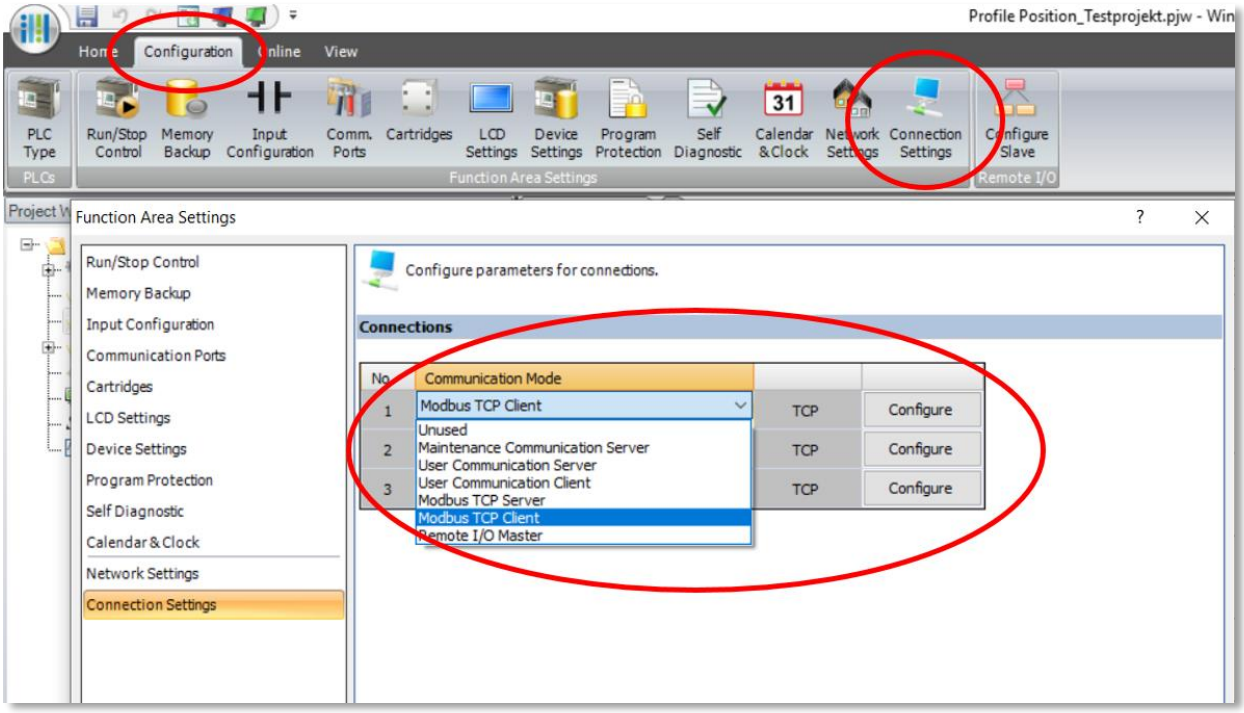
1 Create a new project in WindLDR and select your PLC. The given examples are for Ladder programming.

2 Go to tab **Configuration** -> **Network Settings**.

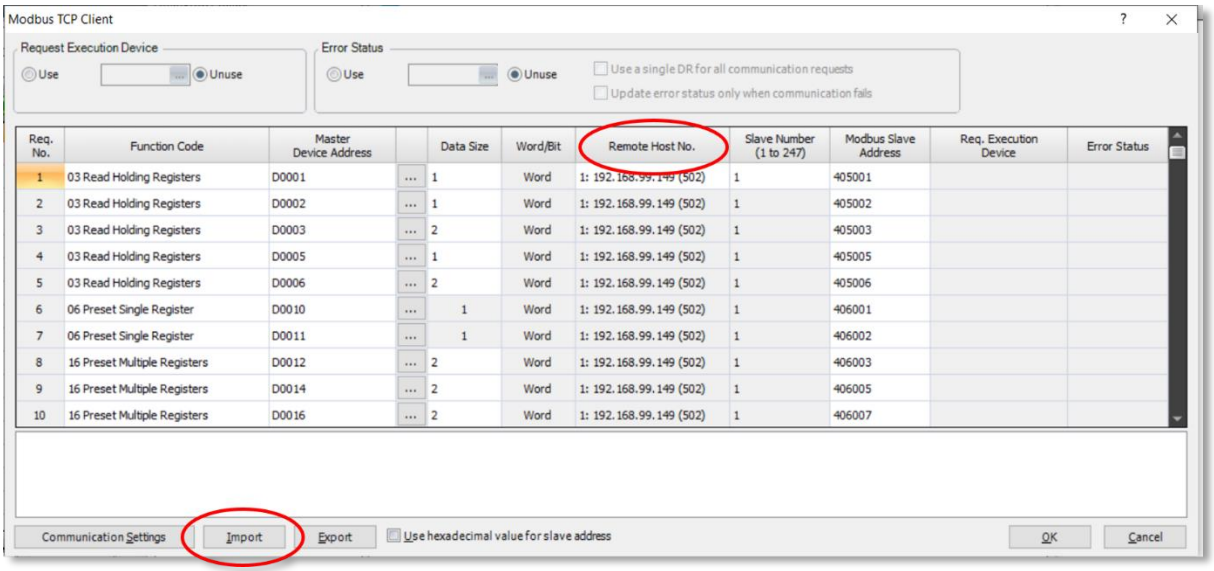
3 Set the appropriate IP address for your PLC. It must match the drive IP address, e.g. 192.168.99.5 for the PLC and 192.168.99.149 for the drive.



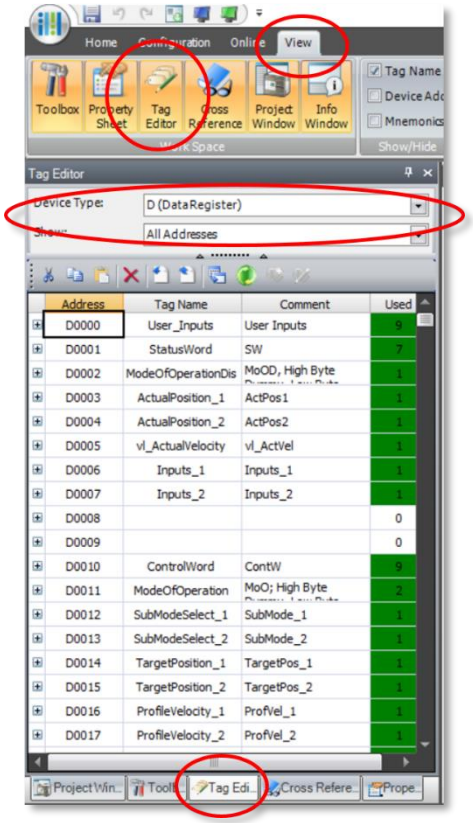
4 Go to tab **Configuration** -> **Connection Settings** and choose **Modbus TCP Client** in the drop-down menu for **Communication Mode**.



5 Click on **Configure** and set up the communication mapping.
For both examples below there are configurations available for import. These are suitable for the standard mapping of the drive.
When importing you still need to set up the **Remote Host No.** with the right IP address of your drive.



6 For a clearer view of the registers you can set tag names and comments for the used registers.
Go to the **Tag Editor** and name the used registers accordingly.
For both examples below there are configurations available that can be imported. Please note that the used Device Type is D (Data Register).



Standard default mapping Tx PDO (defined in Nanotec CANopen object 0x3602:xx):

Nanotec CANopen Object		IDEC Modbus Slave Address
0x6041	Statusword	405001
0x6061	Modes of Operation Display	405002
0x6064	Position Actual Value	405003 and 405004
0x6042	Vl Velocity Actual Value	405005
0x60FD	Digital Inputs	405006 and 405007
0x2292:01	PDI Status	404997
0x603F	Error Code	404998
0x2292:02	PDI Return Value	404999 and 405000

Standard default mapping Rx PDO (defined in Nanotec CANopen object 0x3502:xx):

Nanotec CANopen Object		IDEC Modbus Slave Address
0x6040	Controlword	406001
0x6060	Modes of Operation	406002
0x3202	Motor Drive Submode Select	406003 and 406004
0x607A	Target Position	406005 and 406006
0x6081	Profile Velocity	406007 and 406008
0x6042	Vl Target Velocity	406009
0x60FE:01	Digital Outputs	406010 and 406011
0x2291:01	PDI Set Value 1	405997 and 405998
0x2291:02	PDI Set Value 2	405999
0x2291:03	PDI Set Value 3	406000 Byte 0
0x2291:04	PDI Command	406000 Byte 1

5 Example file for Profile Position Mode

This example shows how the Profile Position Mode could be used in a WindLDR ladder program.

5.1 Set up and description of the positioning example project

Set up the example:
You simply need to import the example project and adjust your communication settings.

1 Click on the round button in the top left corner and choose **Open** -> **WindLDR Project**. Then choose the file **Application Note – Nanotec – IDEC SmartAxis FT1A-H40RSA – Example Project – Profile Position**.

2 Go to tab **Configuration** -> **Network Settings**.

3 Set the appropriate IP address for your PLC. It must match the drive IP address, in the example it is set to 192.168.99.5, the drive address was set beforehand to 192.168.99.149.

4 Go to tab **Configuration** -> **Connection Settings**

5 Click on **Configure** and adjust the **Remote Host No.** according to your drive settings.
When using the standard PDO mapping now the communication should be set up correctly.

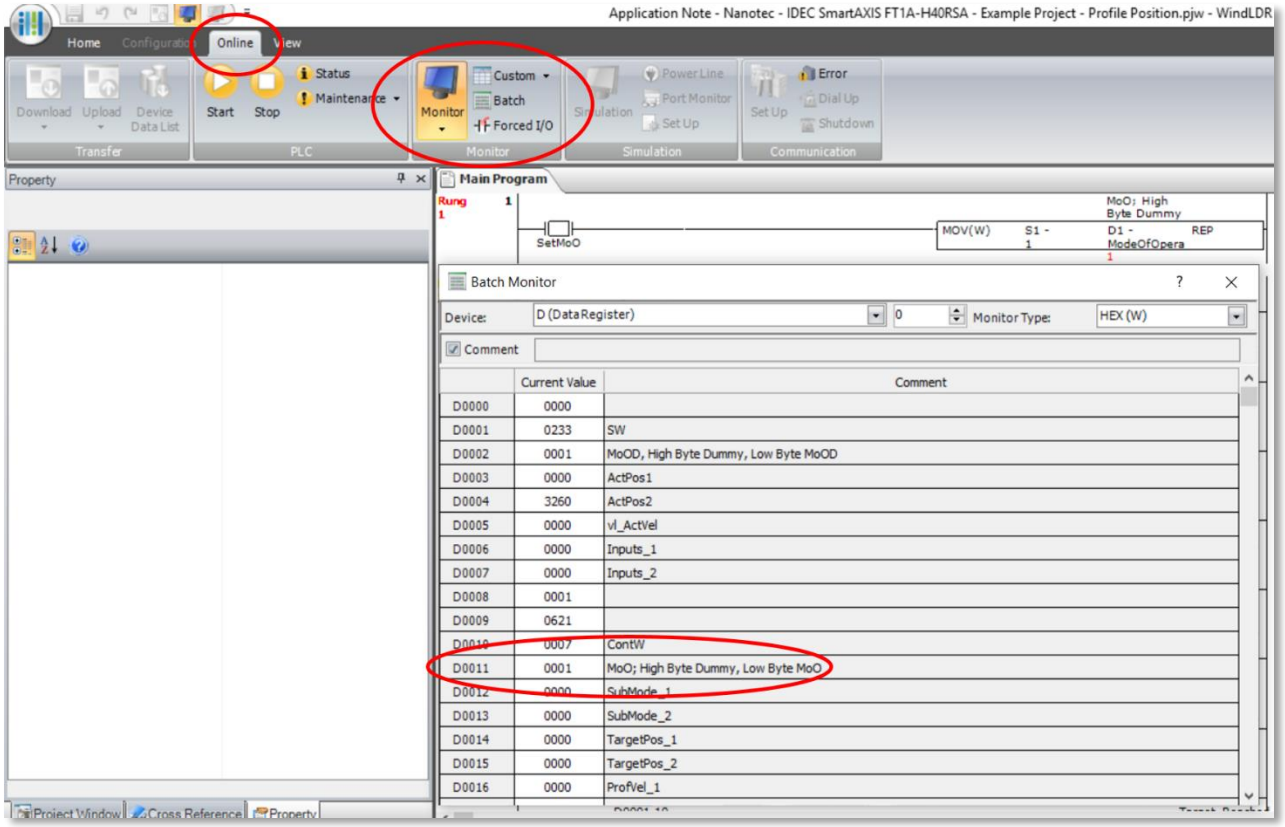
6 To run the example, right-click anywhere in the ladder structure, go to **Online** -> **Convert, Download, and Monitor**. Afterwards the program should be running on the PLC.

Description of used Device Addresses:

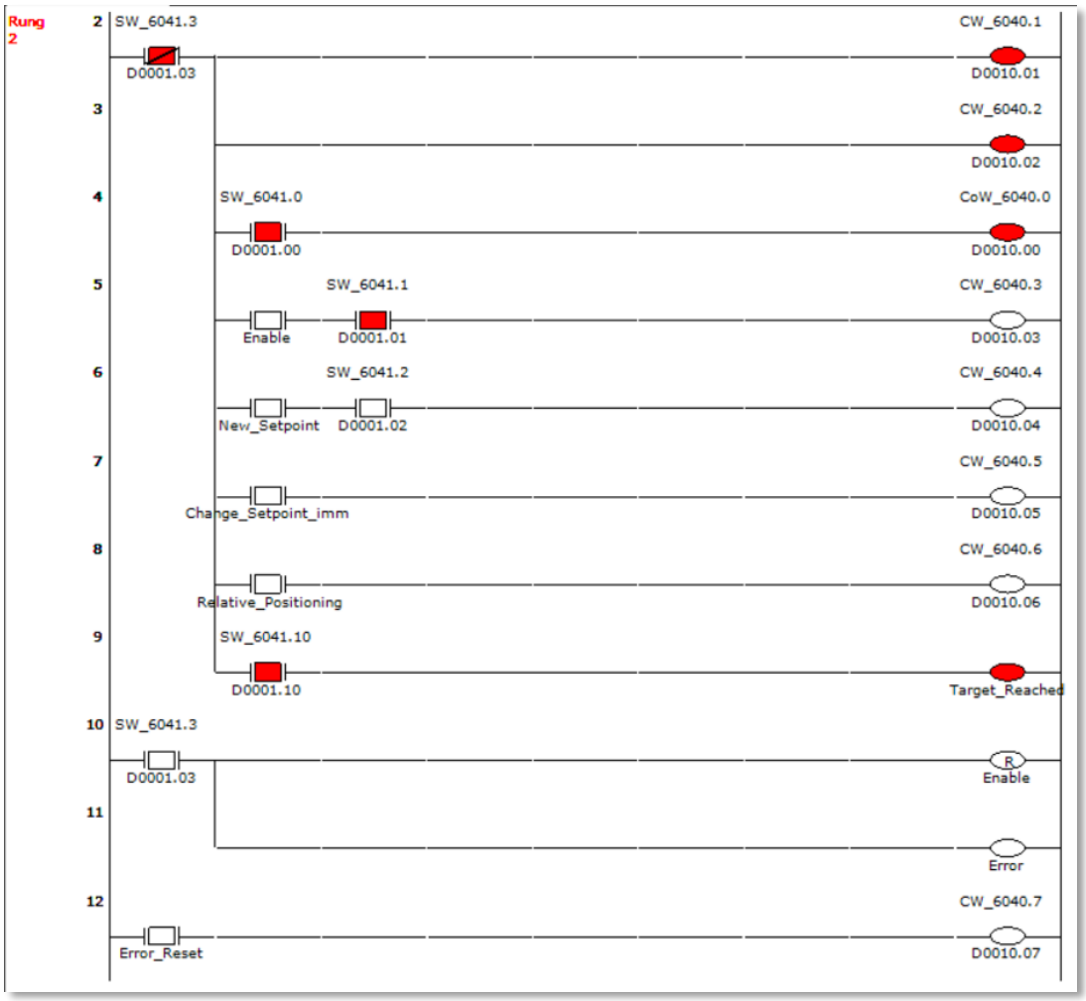
Address	Tag Name	Description
D0001	*StatusWord*	gives the current state of the Statemachine
D0002	*ModeOfOperationDis*	Low Byte shows active Operation Mode, High Byte is a Dummy
D0003	*ActualPosition_1*	Shows Actual Position value, high part of DWORD
D0004	*ActualPosition_2*	low part of DWORD
D0005	*vl_ActualVelocity*	Actual Velocity value
D0006	*Inputs_1*	Input mask value, high part of DWORD
D0007	*Inputs_2*	low part of DWORD
D0010	*ControlWord*	controls the Statemachine
D0011	*ModeOfOperation*	Low Byte controls Operation Mode, High Byte Dummy
D0012	*SubModeSelect_1*	Selection of control loop options, high part of DWORD
D0013	*SubModeSelect_2*	low part of DWORD
D0014	*TargetPosition_1*	Sets Target Position, high part of DWORD
D0015	*TargetPosition_2*	low part of DWORD
D0016	*ProfileVelocity_1*	Sets Profile Velocity, high part of DWORD
D0017	*ProfileVelocity_2*	low part of DWORD
D0018	*vl_TargetVelocity*	Target Velocity for velocity mode
D0019	*Outputs_1*	sets Output mask value, high part of DWORD
D0020	*Outputs_2*	low part of DWORD

Description of Ladder Program:

Rung 1:
This Rung simply shows how a Device Address could be set in a ladder rung. But when online and monitoring the Device Addresses can also be written to directly, for example in the **Batch Monitor**. In the example the *ModeOfOperation* is set to “1”, which is Profile Position Mode.



Rung 2:
This Rung represents the Statemachine of the Nanotec drive. The *ControlWord* is set depending on the *StatusWord* and user inputs (*Enable, New_Setpoint, Change_Setpoint_imm, Relative_Positioning*).
If a fault is recognized by the drive the drive will be disabled and the *Error* Output is set.
An error can be reset with the *Error_Reset* input.



5.2 How to start positioning movements

1 Set the *Enable* input to power up the motor.

2 Set the desired *Target Position* in **Online** -> **Monitor** -> **Batch**.
Please note: The *Target Position* is composed of two Device Addresses. This is due to the fact that the device addresses are set up to be 16 bit values, as is common for Modbus Registers. The target position however is a 32 bit value in our object dictionary. So, we need two 16 bit values to combine them to a 32 bit value.
Target Position = D0014 (TargetPosition_1), *D0015 (TargetPosition_2)*
E.g. “-2000” (dec) = “FFFF F830” (hex) = [*D0014* == “FFFF” (hex), *D0015* == “F830” (hex)]

3 Set the optional inputs (*Relative_Positioning*, *Change_Setpoint_imm*) according to your needs.
The user inputs are set as bits in Data Register *D0000*.

4 Set the input *New_Setpoint* to start a movement.

5 By toggling the input *New_Setpoint* you can start new positioning movements.

6 Example for operation with Plug & Drive Interface (PDI)

This example shows the principle of using the PDI in a WindLDR ladder program.
For further information on the application of PDI please refer to the Functional Description Plug&Drive-Interface.

6.1 Set up and Description of the PDI example project

Set up the example:
You simply need to import the example project and adjust your communication settings.

1 Click on the round button in the top left corner and choose **Open** -> **WindLDR Project**. Then choose the file **Application Note – Nanotec – IDEC SmartAxis FT1A-H40RSA – Example Project – Profile Position**.

2 Go to tab **Configuration** -> **Network Settings**.

3 Set the appropriate IP address for your PLC. It must match the drive IP address, in the example it is set to 192.168.99.5, the drive address was set beforehand to 192.168.99.149.

4 Go to tab **Configuration** -> **Connection Settings**

5 Click on **Configure** and adjust the **Remote Host No.** according to your drive settings.
When using the standard PDO mapping now the communication should be set up correctly.

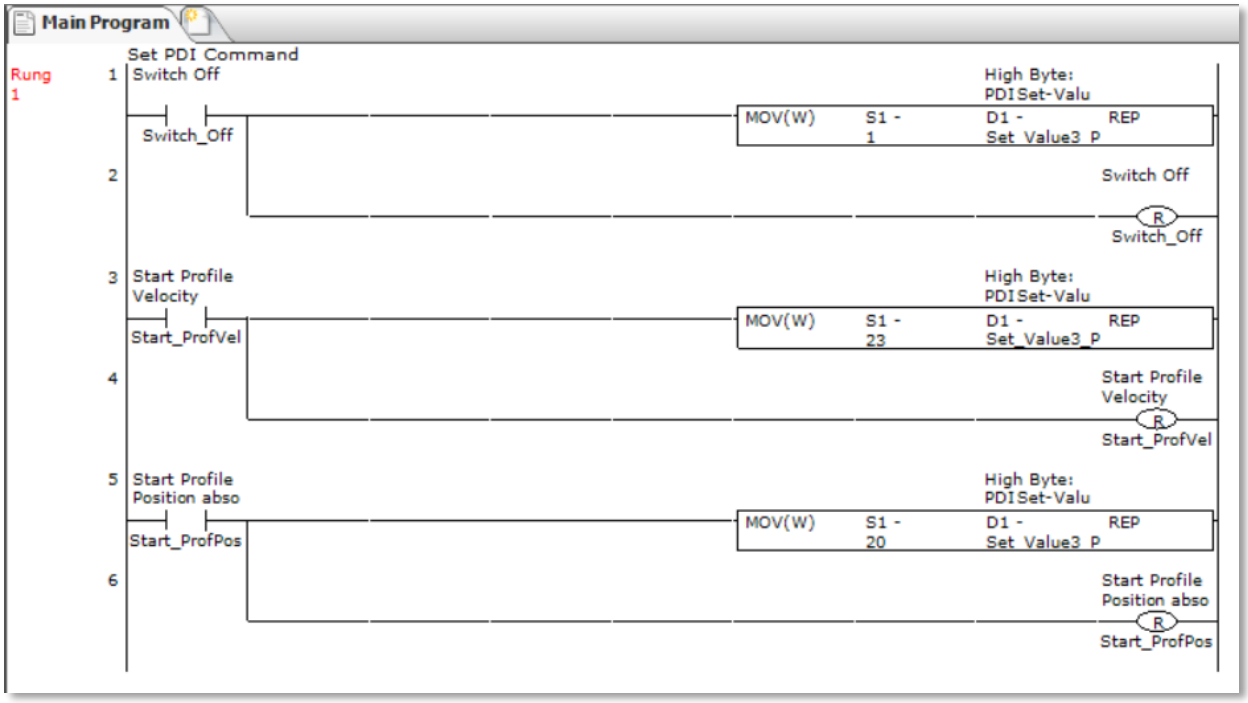
6 To run the example, right-click anywhere in the ladder structure, go to **Online** -> **Convert, Download, and Monitor**. Afterwards the program should be running on the PLC.

Description of used Device Addresses:

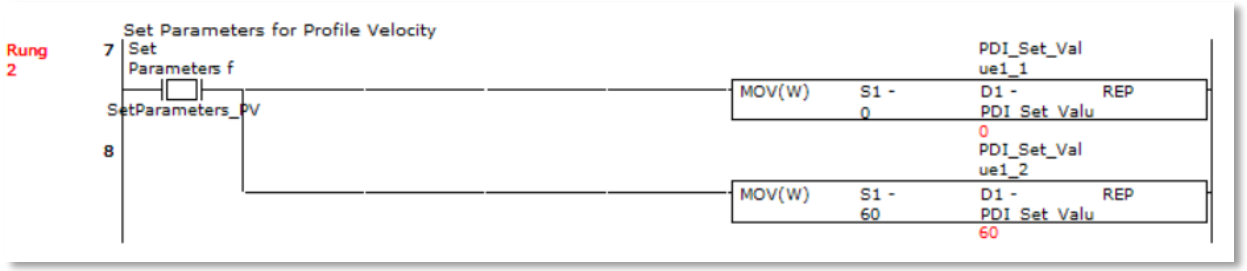
Address	Tag Name	Description
D0001	*PDI_Status*	gives the current state of the drive and PDI interface
D0002	*Error_Code*	shows the error code for the latest error
D0003	*PDI_Return_Value_1*	shows the return value, depending on the used mode, high part of DWORD
D0004	*PDI_Return_Value _2*	low part of DWORD
D0005	*PDI_set_Value1_1*	used differently depending on Mode, mostly for target Values, high part of DWORD
D0006	*PDI_set_Value1_2*	low part of DWORD
D0007	*PDI_set_Value2*	use depends on mode
D0008	*Set_Value3_PDI_Cmd*	High Byte: PDI Set Value 3 is used for reading/writing of objects and settings of Homing Method Low Byte: PDI-Cmd is used as control command

Description of Ladder Program:

Rung 1:
This Rung is used to set the PDI Command for the Operation Mode you want to start. In the example there are three different Commands, which are values “1” (Switch Off), “23” (Profile Velocity), and “20” (Profile Position absolute).
Switch Off will disable the drive, the motor will no longer be powered.
Profile Velocity will start the Profile Velocity mode with whatever the profile settings are. The settings must be checked and adjusted before starting the mode. This is possible with **Rung 2**.
Profile Position absolute will start the Profile Position mode with an absolute position movement. The settings for the movement must be checked and adjusted before starting the mode. This is possible with **Rung 3**.
All commands are automatically reset after they are sent.

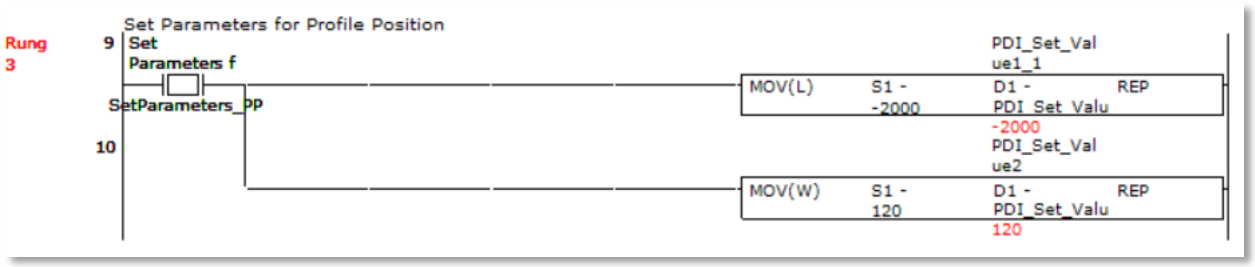


Rung 2:
This Rung sets the parameter settings for the Profile Velocity Mode. Here the PDI Set Value 1 is set. Note that the PDI Value 1 is a 32-bit value. Therefore, two registers need to be set, in this case *PDI_set_Value1_1* to “0” and *PDI_set_Value1_2* to “60”.
PDI Set Value 1 is then “60” rpm (default unit).



Rung 3:
This Rung sets the parameter settings for the Profile Position absolute Mode. Here the PDI Set Value 1 and PDI Set Value 2 are set.
Instead of writing the two registers for *PDI_Set_Value1* with WORD data type (as shown in **Rung 2**) it is also possible to write two registers at once by choosing data type LONG. This way it is also easily

possible to write negative values. In the example the value is set to “-2000”. The position unit by default is 3600 steps/revolution.
The *PDI_Set_Value2* is set to “120”. This is the speed for the position movement.



6.2 How to use the PDI

In general, the PDI can be used to start simple movements and can be used as an alternative to the Device Profile given in CiA 402 standard.
PDI commands can be used to start movements directly without using the Statemachine. By setting the PDI command to a certain value the corresponding drive profile will be started directly. It is important to check and adjust the boundary conditions for the profile before starting it.

Using Profile Velocity Mode:

1 Set the input *SetParameters_PV* in **Rung 2** to write the set values to PDI Set Value 1. In the example the speed will be set to 60 rpm (default unit).

2 Reset the input *SetParameters_PV* in **Rung 2** after the value is set.

3 Set the input *Start_ProfVel* in **Rung 1** to write the value “23” to *Set_Value3_PDI_Cmd*. By setting the PDI Command to “23” the Profile Velocity Mode is started, the motor will move with the set speed.

4 To stop the movement set the input *Switch_Off* in **Rung 1**.

Using Profile Position absolute Mode:

1 Set the input *SetParameters_PP* in **Rung 3** to write the set values to PDI Set Value 1 and PDI Set Value 2. In the example the position will be set to “-2000” (1/10th of a degree) and the speed will be set to 120 rpm (default unit).

2 Reset the input *SetParameters_PP* in **Rung 3** after the values are set.

3 Set the input *Start_ProfPos* in **Rung 1** to write the value “20” to *Set_Value3_PDI_Cmd*. By setting the PDI Command to “20” the Profile Position absolute Mode is started, the motor will move with the set speed to the set target position.

4 To stop the movement or simply shut down the motor set the input *Switch_Off* in **Rung 1**.

7 Liability

This Application Note is based on our experience with typical user requirements in a wide range of industrial applications. The information in this Application Note is provided without guarantee regarding correctness and completeness and is subject to change by Nanotec without notice.

It serves as general guidance and should not be construed as a commitment of Nanotec to guarantee its applicability to all customer applications without additional tests under the specific conditions and – if and when necessary – modifications by the customer.

The provided information does not replace datasheets and other product documents. For the latest version of our datasheets and documentations please visit our website at www.nanotec.com.

The responsibility for the applicability and use of the Application Note in a particular customer application lies solely within the authority of the customer. It is the customer's responsibility to evaluate, investigate and decide, whether the Application Note is valid and suitable for the respective customer application, or not.

Defects resulting from the improper handling of devices and modules are excluded from the warranty. Under no circumstances will Nanotec be liable for any direct, indirect, incidental or consequential damages arising in connection with the information provided.

In addition, the regulations regarding the liability from our Terms and Conditions of Sale and Delivery shall apply.

8 Copyrights and contact

Nanotec Electronic GmbH & Co. KG
Kapellenstraße 6
85622 Feldkirchen
Germany

Tel. +49 (0)89 900 686-0
Fax +49 (0)89 900 686-50
info@nanotec.de

www.nanotec.com