

# Connecting UniOP to CoDeSys Controllers

This Technical Note contains the information required to connect via serial communication UniOP to controllers based on the 3S CoDeSys system.

The CoDeSys communication driver is associated to the Designer file D32UPLC175.DLL.

## Contents

1. Introduction .....	2
2. CoDeSys parameters and Designer's tags .....	2
3. Controller Setup .....	3
4. The Data Field Properties dialog box.....	4
5. Data Formats .....	5
6. RDA, Alarms, Mailbox and Data Transfer .....	5
7. Communication Parameter Setting .....	5
Appendix A. Communication Error Codes .....	7

## 1. Introduction

To create a Designer project for communication with controller based on CoDeSys, select the driver “CoDeSys” from the list of available communication drivers in the Configure Controller dialog box.

---

*Note: In case the CoDeSys is not shown in the list, make sure that the file D32UPLC175.DLL is present in the Designer folder and then press the Refresh button.*

---

## 2. CoDeSys parameters and Designer’s tags

This chapter describes how to import CoDeSys PLC parameters into Designer’s UI using Designer’s utility application TagEditor.

When configuring PLC using manufacturer’s PC software make sure to enable Symbol file creation (file with .SYM extension). It can be done under the CoDeSys programming software, by selecting “Project\Option\Symbol configuration” and mark the check box “Dump symbol entries” like shown in the picture below.

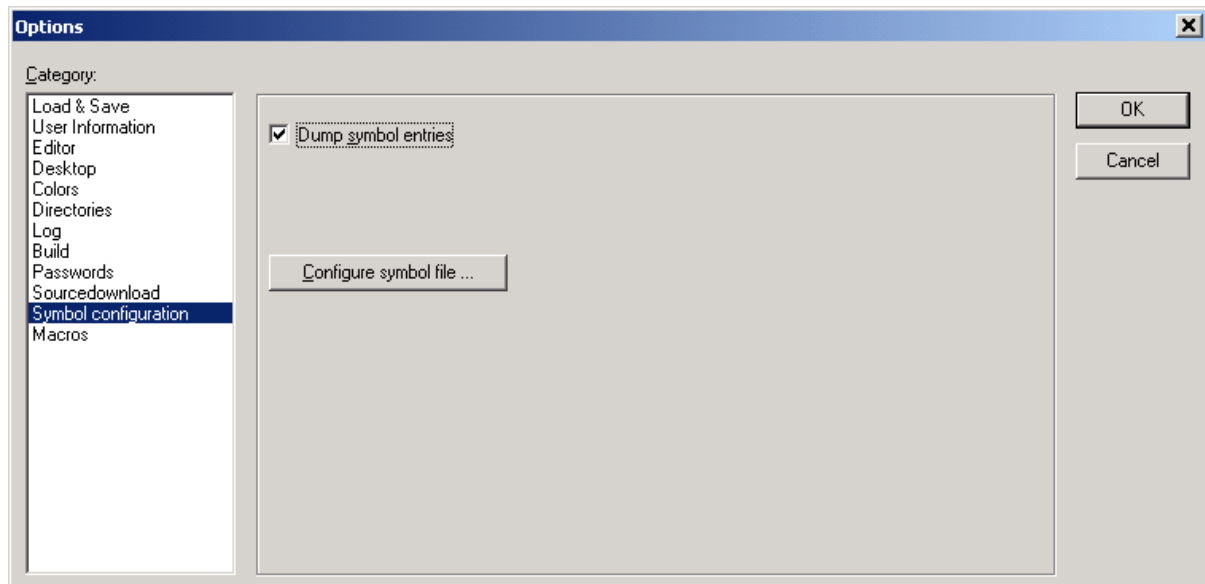


Figure 1 – Symbol configuration

Now, TagEditor can be used to import parameters contained in this file. TagEditor will prepare Tag Database (.MDB file) which can be used in standard way from Designer’s UI to import Tags. For more details on how to import MDB file check Designer manual.

### 3. Controller Setup

Figure 2 below shows the “Controller Setup...” dialog box for the CoDeSys driver.

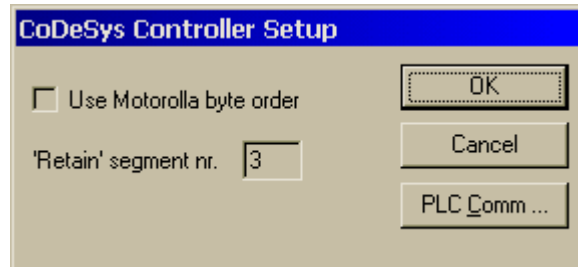


Figure 2 – Controller Setup Dialog Box

The option “Use Motorola byte order” defines the byte order that will be used by UniOP’s communication driver when sending communication frames to the PLC.

NOTE: this is not applied to the byte order of the Data *Received From/Sent To* controller! To obtain right Data byte order, make sure to use appropriate “Data Format”, for instance: WORD, or WORD INV. (Data Format can be selected in Data Field Properties dialog box described in chapter 4. Also, check chapter 5 for more details about Data Formats)

The option “Retain segment nr” enables user to define the Retain segment number configured in the PLC in order to match the programming software settings.

## 4. The Data Field Properties dialog box

The CoDeSys Data Field Properties dialog box is shown in the figure below.

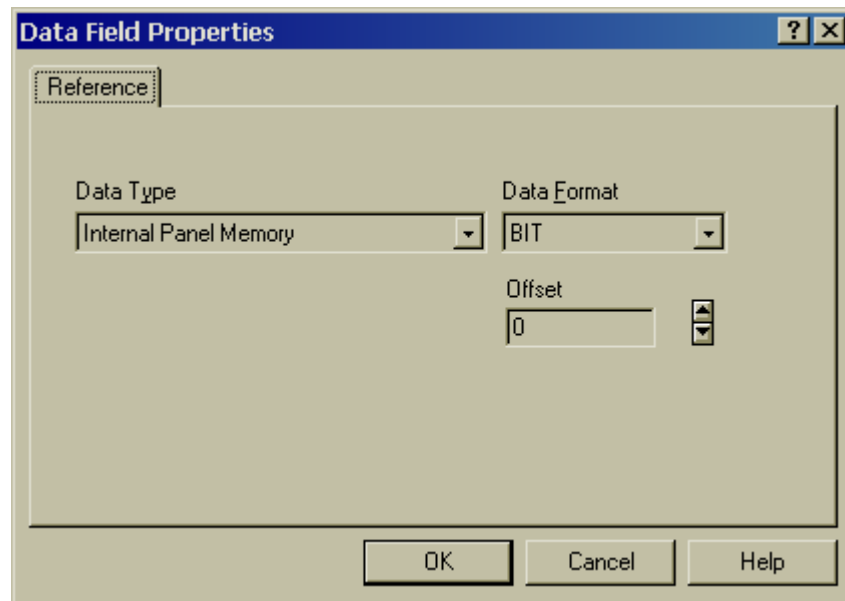


Figure 3 – The Data Field Properties dialog box

The “Data Type” combo box is used to select the memory area to be referenced. The following memory areas can be selected:

Name	Offset range
Internal Panel Memory	0-511
PLC Memory	0-65535
PLC Input	0-65535
PLC Output	0-65535
PLC Retain	0-65535
PLC Parameters	0-65535

### Internal Panel Memory

This is a memory area internal to the UniOP panel and does not have any relation with the controller’s memory. Any access to this area does not generate any message on the serial line.

### PLC Memory, PLC Input, PLC Output, PLC Retain

This is the controller’s general-purpose memory. Code number (POURef) for PLC Retain is adjustable in Controller Setup.

### PLC Parameters

In order to access PLC Parameters, the user needs to import parameters from the symbol file, as described in previous chapter.

## 5. Data Formats

The CoDeSys communication driver handles standard as well controller-specific data formats, such as those needed for the TIME and POINTER data types.

All standard data types are available both in 'normal' and 'inverted' low-level data storage. This will make possible to connect to controllers that are based both on high-endian and little-endian data formats.

Some comments on special data types are given below.

1) Designer provides the special Data Format "TIME" and "TIME INV". Values read from the PLC will be displayed in the following format: 1h2m3s4ms.

Data entry for TIME variables can be done only using macro command "STEP" available in Designer Macro Editor, with appropriate step sizes:

	Use step size:
hours	3600000
minutes	60000
seconds	1000
milliseconds	1

2) Designer will treat POINTER type data as DWORD when presenting its Data Format in the user interface.

Please remind that special data formats do not support scaling and cannot be used in UniNet clients.

## 6. RDA, Alarms, Mailbox and Data Transfer

All available data types, can be used as reference for RDA, Alarms, Mailbox and Data Transfer.

NOTE: Data Type PLC Parameters can be used for such purpose only if PLC's memory is configured in appropriate way. To check this, see Designer manual and appropriate memory mappings for each of these purposes.

## 7. Communication Parameter Setting

Communication parameters for serial communication can be adjusted by clicking the "PLC Comm..." button in the "Controller Setup" dialog box.

The default communication parameters are defined as follows:

	Default value
Speed	19200 baud
Start bits	1
Data	8
Parity	None

Stop bits	1
-----------	---

Connection to the PLC is always point-to-point (one operator panel connected to one PLC).

## Appendix A. Communication Error Codes

Current communication status is displayed on the system page of the UniOP. Beside the string, describing current state of the communication (OFF, ON, ERR), there is an additional error code representing the last (which may be not the current one) error encountered. The codes are:

<b>Code</b>	<b>Description</b>	<b>Notes</b>
<b>00</b>	No error	There are no communication errors and there have been no errors since start-up.
<b>04</b>	Negative ACK	NAK returned by the controller while UniOP is waiting for ACK, or invalid data in frame.
<b>05</b>	Time out (receiving)	No response received from the controller within the timeout period.
<b>06</b>	Response error	Wrong response telegram structure.
<b>07</b>	General communication error	General un-known communication error.
<b>11</b>	Line error	Wrong bad baud rate, parity, etc.
<b>12</b>	Checksum error	Wrong checksum received from the controller.