

# Connecting UniOP to SAIA S-BUS

Connect to SAIA controllers using the  
S-BUS protocol

---

**Copyright © 2007 Sitek S.p.A. – Verona, Italy**

Subject to change without notice

The information contained in this document is provided for informational purposes only. While efforts were made to verify the accuracy of the information contained in this documentation, it is provided “as is” without warranty of any kind.

Third-party brands and names are the property of their respective owners.

[www.uniop.com](http://www.uniop.com)

## Contents

1	Introduction .....	4
2	Setting-up UniOP for Communication.....	4
2.1	Controller Setup .....	4
3	PLC Communication settings.....	5
4	Data Field Properties .....	6
4.1	Node Override.....	7
	Appendix A. Communication Error Codes .....	8

## 1 Introduction

To create a UniOP application for connection to a SAIA controller via S-BUS serial link, select the driver “SAIA S-BUS” from the list of available communication drivers in the Change Controller Driver... dialog box.

The SAIA S-BUS communication driver is delivered with the Designer file D32uplc083.dll.

---

**Note:** *In case the SAIA S-BUS driver is not shown in the list, make sure that the file D32uplc083.dll is present in the Designer installation folder and then press the Refresh button in the Change Controller Driver... dialog box.*

---

## 2 Setting-up UniOP for Communication

Setting-up UniOP for serial communication with SAIA controllers is straightforward. Information is available in this chapter.

### 2.1 Controller Setup

Use the Controller Setup dialog box to choose the SAIA controller type you want to connect and the communication parameters.

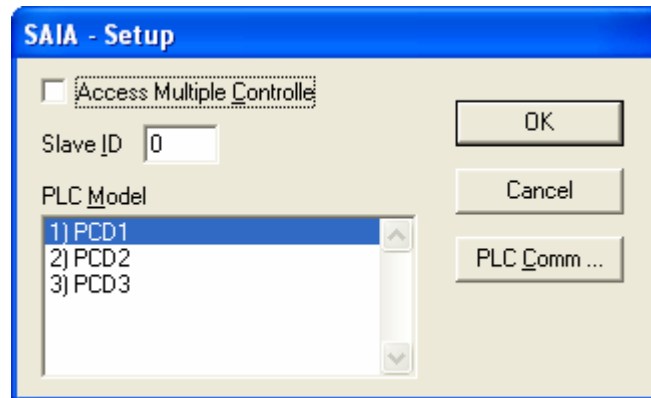


Figure 1

This dialog box allows additionally to configure a PLC network. Example below shows the network of two PLCs with Slave IDs 1 and 3.

The “PLC Comm...” button allows to set the communication parameters, except for the Parity settings.

Regardless of the Parity setting a driver will always use SAIA standard; first character is sent with Mark parity, while the rest of telegram characters are sent with Space parity.

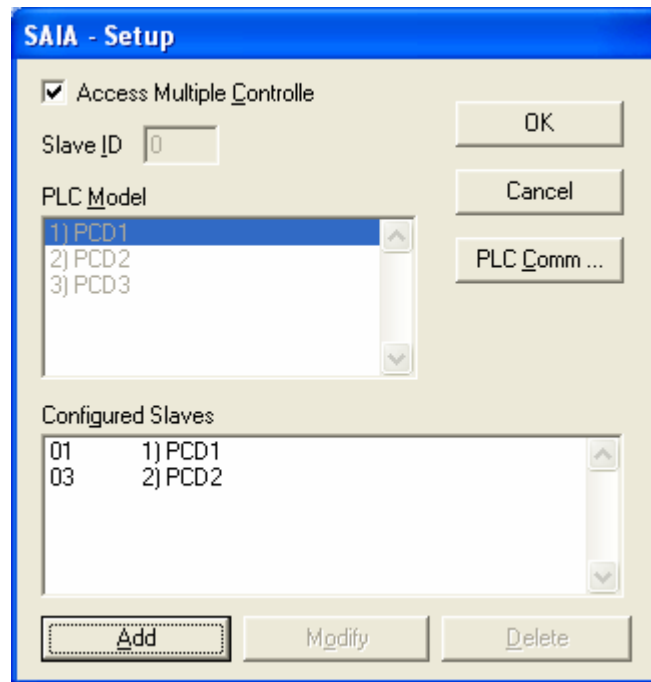


Figure 2

### 3 PLC Communication settings

SAIA PCD plc has more communication port to exchange data information with other devices. Usually plc programming port (PORT 0) can be use with programming protocol (P8) or with S-BUS protocol. This depend by the hardware configuration settings.

Plase note that when you enable S-BUS over PORT 0, you can not have connection with PLC programming SW on PGU port (RS-232).

For this reason we suggest to use PORT 1 for UniOP communication with following parameters:

cpu 0

station 1

38400

S-Bus mode: Data (S2) → "No parity"

And then leave PORT 0 avaiable for PLC programming software connection

The table below is a summary of all possible connections between UniOP and SAIA controllers using built-in ports and optional communication equipments ("F" modules).

CPU Type	Connection	Port Name	Protocol	UniOP cable	Communication
PCD1/PCD2	PGU port – 9 pin	PORT 0	P8	CA43	RS-232
PCD2	Screw Terminal Bus connection	PORT 0	SBUS	CA176	RS-485, 2 wire
PCD1/PCD2	PCD7.F110 module	PORT 1	SBUS	<b>Not Tested</b>	RS-485, 4 wire
PCD1/PCD2	PCD7.F110 module	PORT 1	SBUS	CA180	RS-485, 2 wire
PCD1/PCD2	PCD7.F120 module	PORT 1	SBUS	CA178	RS-232
PCD2	PCD2.F520/F530 Module	PORT 2	SBUS	CA179	RS-232

PCD2	PCD2.F520/F530 Module	PORT 3	SBUS	CA181	RS-485, 2 wire
PCD2	PCD2.F520/F530 Module	PORT 3	SBUS	CA177	RS-485, 4 wire

## 4 Data Field Properties

This dialog box allows a user to configure a reference to the PLC's memory. Drop down box "Data Type" lists the memory areas of the controller and supported by the UniOP communication driver.

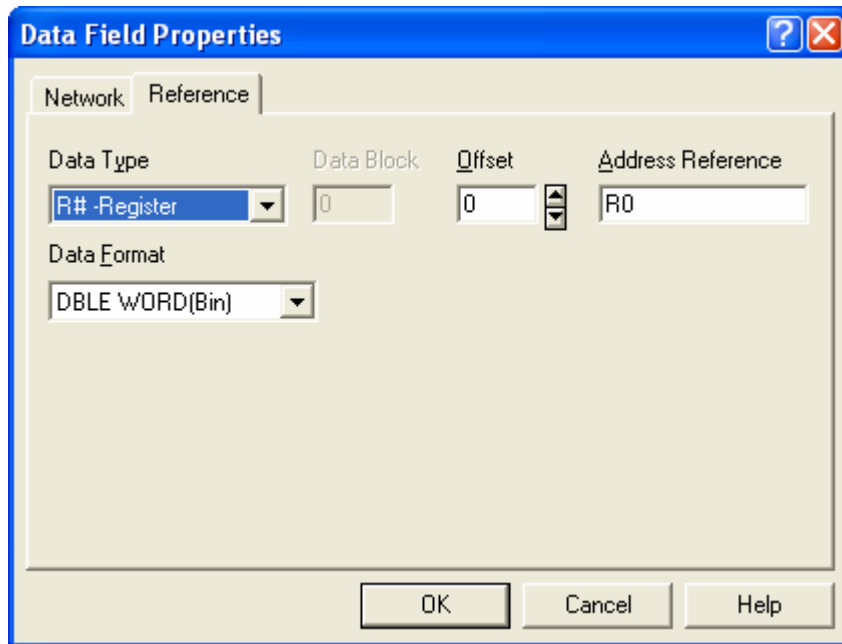


Figure 3

## 4.1 Node Override

The SAIA S-BUS driver offers one special Data Type called “Node Override” as shown in the figure below.

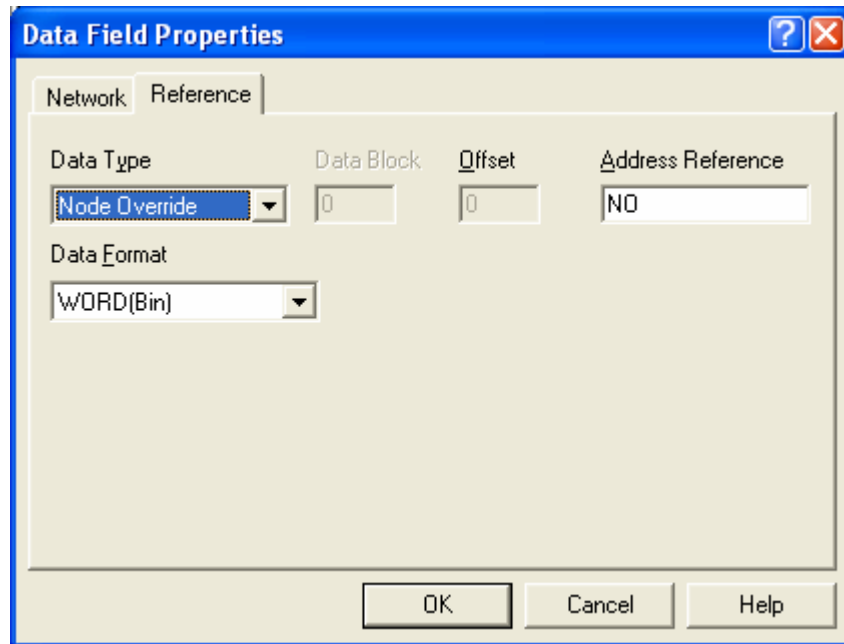


Figure 4

There is one dedicated Node Override variable per each controller configured in the PLC network (see Figure 2).

At run-time the operator can change it, and access the data from physically different controller, identified with a different Slave ID.

At programming time the “Node Override” field can be added to any page of the project for any slave of interest. At run-time it is enough to change the value of the variable “Node Override” and the panel will immediately start to communicate to the controller with the entered Slave ID. All requests addressed to the old node number will be immediately re-directed to the new node number. Initial value of “Node Override” variable is -1. That is the value user has to enter when he wants to switch the communication back to the default Slave ID specified in the Controller Setup dialog.

Node Override settings are retained after power cycle in battery backup memory, and therefore supported only by panels with battery. The data is protected by both, a checksum and a copy of the data.

---

**Note:** From driver version 5.25 - 4.10, the limit of a maximum number of six (6) node overrides, present since the previous versions has been removed and the feature is available *ONLY* on units with battery back-up.

---

## Appendix A. Communication Error Codes

Current communication status is displayed in the System Menu of the UniOP.

A message and a numeric error code describe the error status.

The message reports the current communication status. The number shows the code of the current communication error or, if the communication is correct, the code of the last error encountered. When the error code 0 is shown, it means there have been no communication errors since this system start-up.

Code	Description	Notes
00	No error	There are no communication errors and there have been no errors since start-up.
05	Timeout error	A timeout occurred during communication to the slave.
06	Response error	PLC returned a NAK response or a buffer overrun occurred.
07	General error	Internal software error, should never occur
11	Line error	Wrong communication parameters detected; parity, baud rate. Number of data, start or stop bits.
12	Checksum error	Checksum error in response frame of the PLC.