

The module of subsystem “Data acquisition” <Siemens>

<i>Module:</i>	Siemens
<i>Name:</i>	Siemens DAQ
<i>Type:</i>	DAQ
<i>Source:</i>	daq_Siemens.so
<i>Version:</i>	1.2.0
<i>Author:</i>	Roman Savochenko
<i>Translated:</i>	Maxim Lysenko
<i>Description:</i>	Provides a data source PLC Siemens by means of Hilscher CIF cards by using the MPI protocol and Libnodave library for the rest.
<i>License:</i>	GPL

Contents table

The module of subsystem “Data acquisition” <Siemens>	1
Introduction	2
1. Communication controllers CIF	3
2. The controller of the data source	5
3. The parameters of the data source	7
4. Asynchronous recording mode	11

Introduction

The primary aim of creating the module is to provide support for industrial controllers of firms Siemens of series S7(S7–300, S7–400). Historically, access to the controllers of the firm in the Profibus network is made only through its own communication processor (CP5412, CP5613, etc.) and the protocol S7. These communications processors and API to the protocol S7 are rather expensive, in addition to the drivers for the communication processors and S7 API are closed and are only available for the platform Intel + Windows (I met the information on opportunities to buy for Linux).

As an alternative to these decisions of the company Siemens, which allows you to fully work with the controllers of Siemens, is the range of communication products of firm Hilscher(<http://hilscher.com>), through the communications processors CIF of series PB(Profibus) and the library Libnodave(<http://libnodave.sourceforge.net>).

Feature of Hilscher products is completely open specification of the protocol of exchange with the communication processor, the unified driver for all cards CIF, the availability of drivers for many common operating systems(OS) and openness of the driver for OS Linux(GPL).

The basis of the module is the driver of version 2.620 of Hilsher, kindly provided by Hilsher in the face of [Devid Tsaava](#) for the 2.6 series kernels of OS Linux. Everything needed files to building are included in the module and it is don't needed to satisfy any special dependencies. The driver version 2.620 for the CIF cards is available for download [cif2620.tbz](#).

The range of boards of CIF family of firm Hilsher and unified driver supports the widest range of equipment. To lay support all these features in this module without having all the equipment on hand, it is not possible. Therefore, the support of the equipment will be added on demand and availability of equipment. As of version 1.1.0 module provides support for data sources on the network via Profibus or MPI by means of MPI protocol at the network speed of 9600Bod to 12MBod. In particular, supported and it is carried out check on the controllers of the Siemens company of family S7 (S7–300, S7–400).

Library Libnodave is an implementation of the MPI, S7, ISO-TSAP and others protocols by means of revers-engineering, that are used in interaction with the controllers of Siemens. Library supports many MPI and USB adapters, as well as ProfiNet. Communication processors firm Siemens, on platforms other than Windows, the library doesn't support. At this stage, module support the protocol ISO-TSAP (ProfiNet) through the library Libnodave. Library Libnodave fully incorporated in this module and does not require a special permit of any dependencies during building and in the performance.

1. Communication controllers CIF

CIF family card driver supports the ability to install up to 4 CIF boards. In order to control the availability of cards in the system and their possible configurations, the module provides a form of control and configuration of the CIF-cards(Fig. 1).

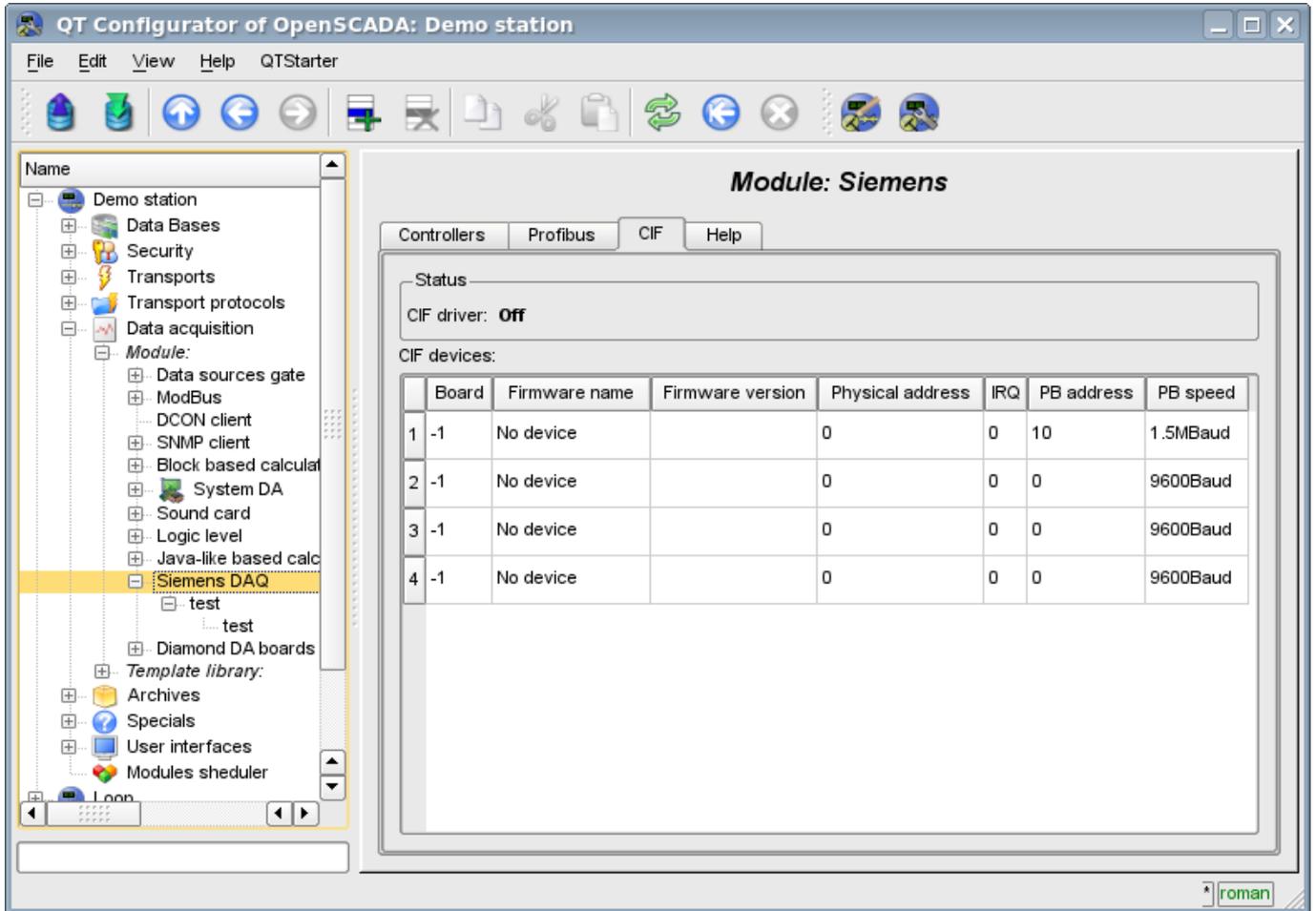


Fig.1. Configuration tab of CIF-boards.

Use this form you can verify the existence of communication processors and their configuration, and configure the network settings of PB Profibus in the form of addresses of communication processor and speeds of bus Profibus. In the other tab of the module(Fig.2) you can verify the presence of various stations in the network Profibus.

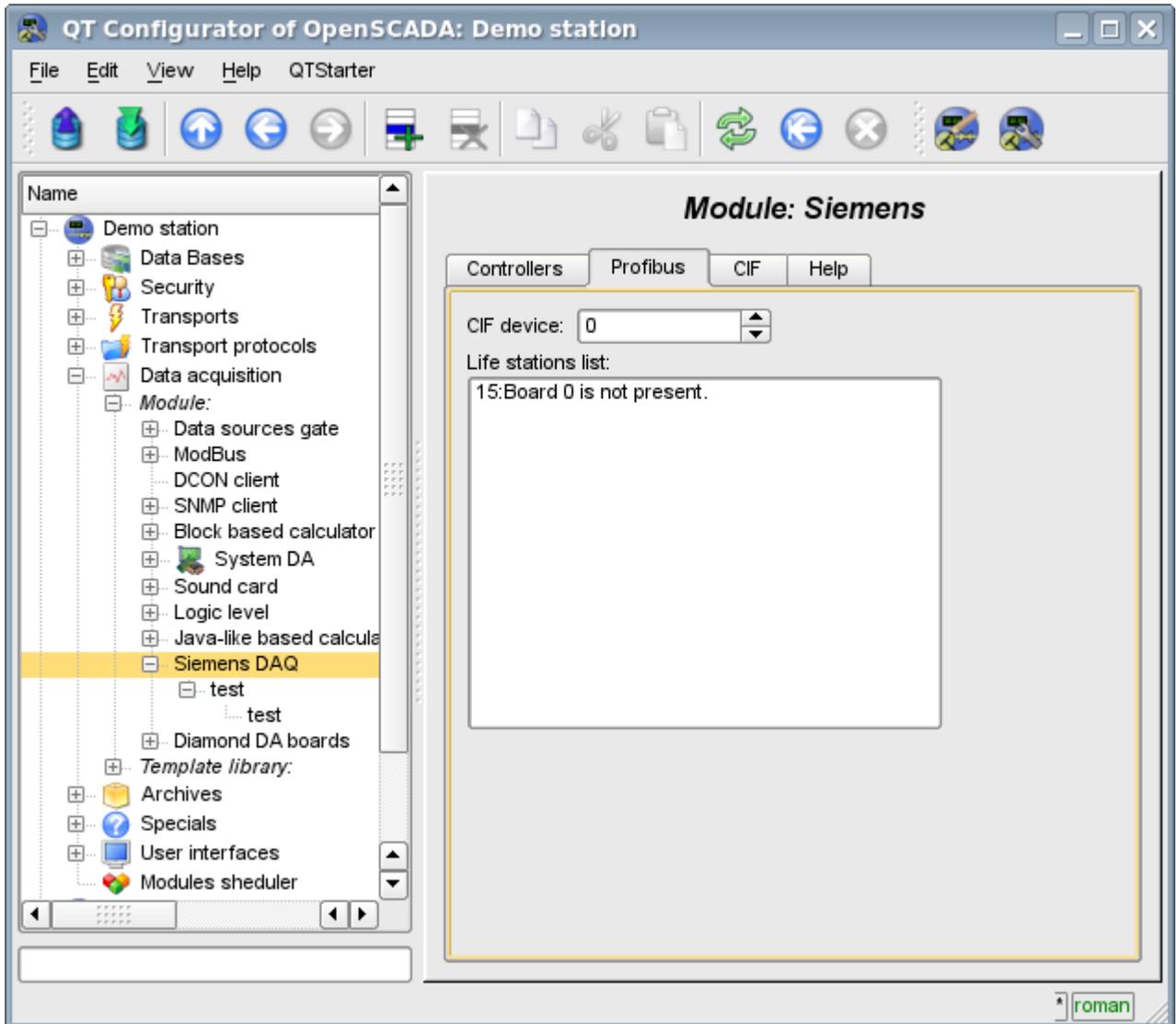


Fig.2. Monitoring tab of Profibus network.

2. The controller of the data source

To add a data source it is created and configured the controller in the system OpenSCADA. Example of the configuration tab of the controller of this type is depicted in Figure 3.

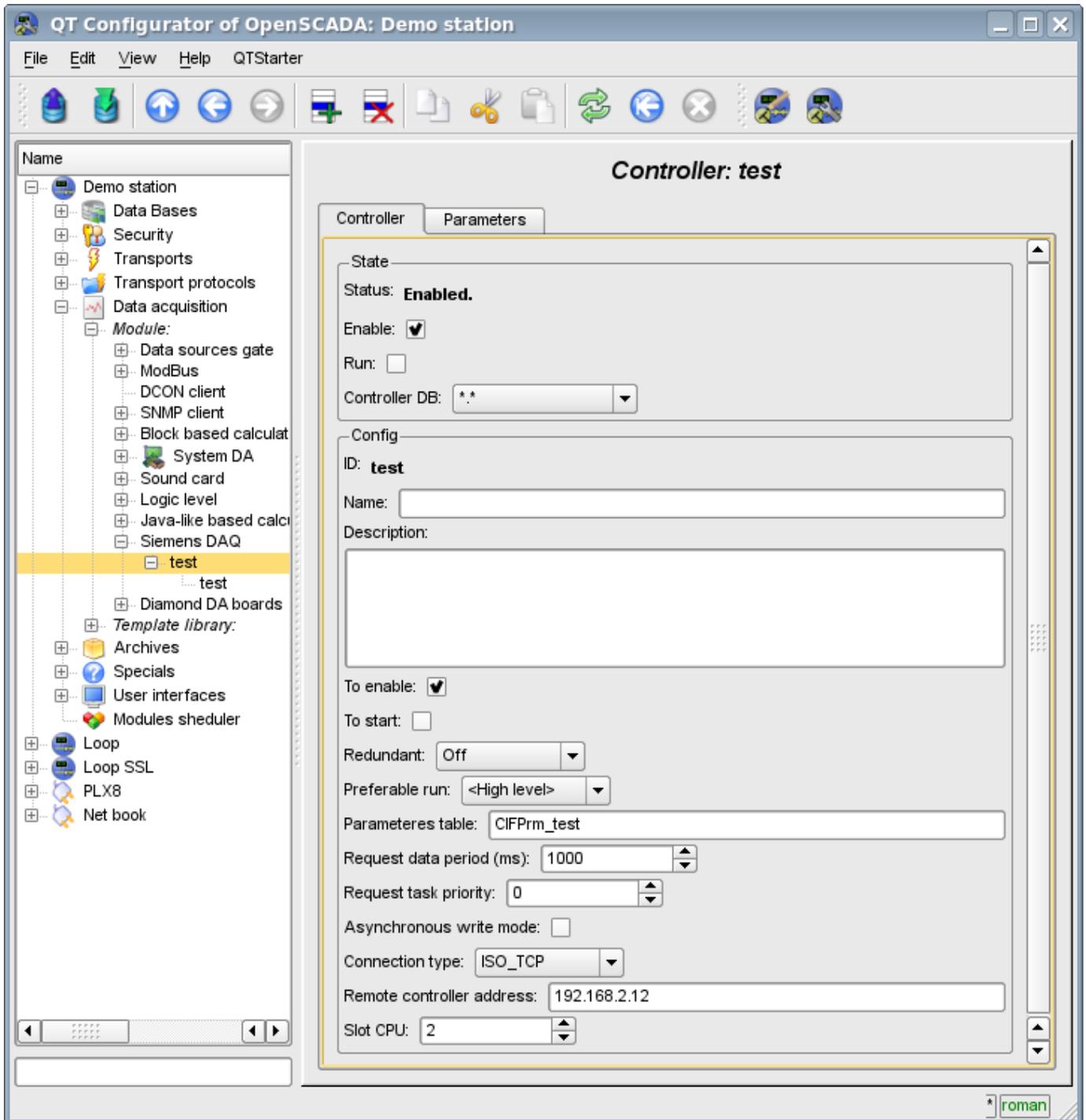


Fig.3. Configuration tab of the controller.

Using this tab you can set:

- The state of the controller, as follows: «Enabled», «Running» and the name of the database containing the configuration and interrogation time of the remote controller.
- Id, name and description of the controller.
- The state, in which the controller must be translated at boot: «Enabled» and «Running».
- Name of table to store the configuration of the parameters of the controller.

- The period and the priority of the task of data acquisition.
- The mode of the asynchronous recording in the remote controllers.
- Connection type. Supported CIF_PB and ISO_TCP
- Address of the industrial controller. If the connection type CIF it is the address of the network Profibus, and in the case of ISO_TCP it is the IP-address in the Ethernet.
- Slot CPU in which the central processor of the controller is placed.
- CIF card used for access to the industrial controller through CIF communication processors.

3. The parameters of the data source

Given the high intellectuality of data sources in the face of industrial controllers of Siemens S7–300 and S7–400, the options are executed on the basis of [templates](#). This approach allows us to go beyond a rigid list of types of parameters, which limits the possibilities of the controllers, and provide users with the ability to build the necessary types of parameters independently or use the library of already been developed types of parameters (templates).

Accordingly, the module provides only one type of parameters – “Logical”. Additional configuration fields of the parameters of the module(Figure 4) is the field of selection of template of the parameter.

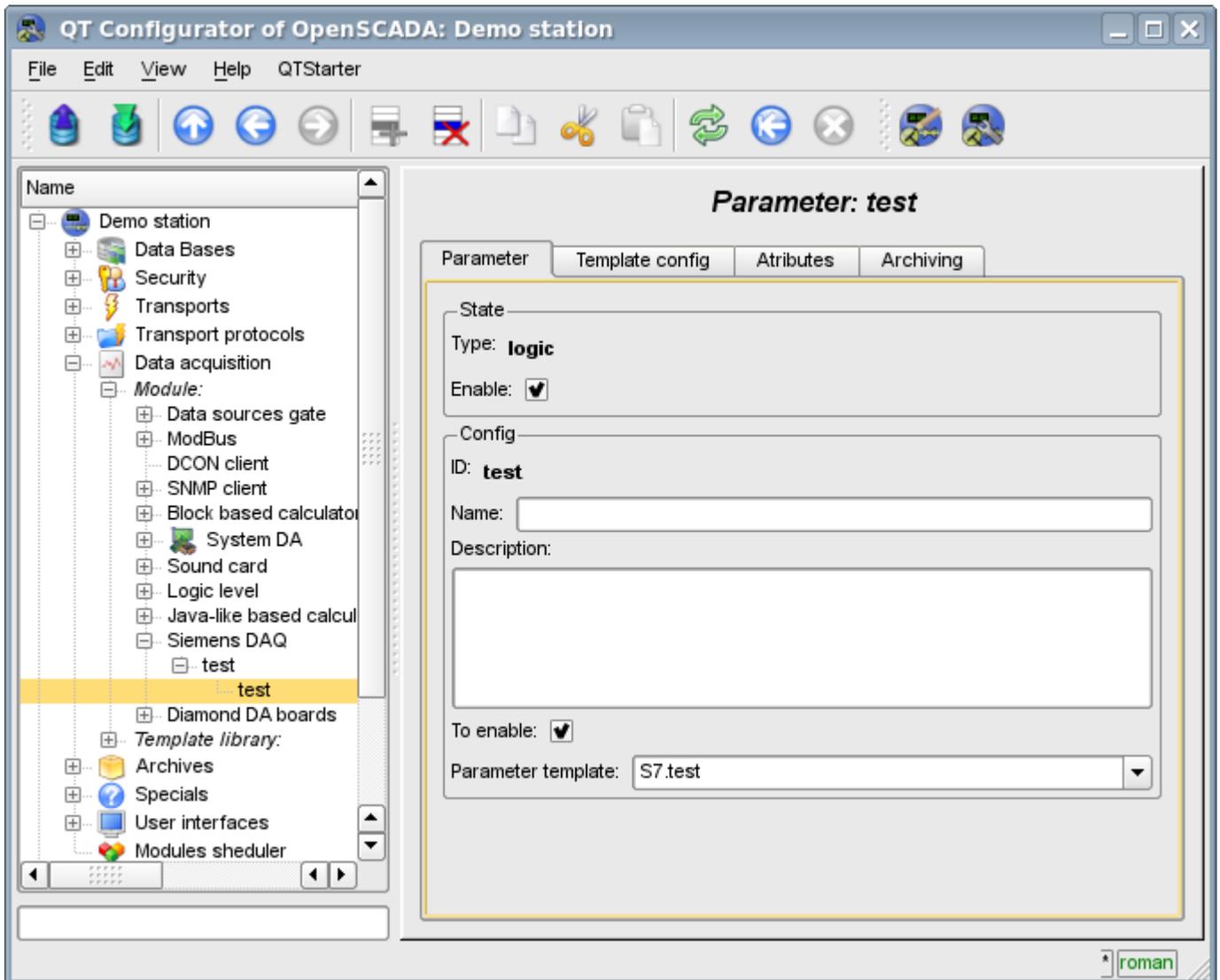


Fig.4. Configuration tab of the parameter.

To configure a template of parameter it is made the appropriate tab. The contents of this tab is defined by the configuration of the template that is the corresponding link fields and fields of setting the constants are formed.

Types of links depend on the type of parameter in the template (boolean, integer, real and string) and the definition of link value(for the group of links). Definition of the group link in the template is written in the format: "<Name of the link>|<The offset in the database>|<The size of the value>", where:

- <Name of the link> — Name of the group link. All links with the same name are grouped and shown as a link to the database or database with the specified offset.
- <The offset in the database> — Name of the offset in the data block (DB). If the only database in

the configuration of the template is specified the offset will be specified for the parameter, but if in the configuration of the template the offset will be specified too, the both offsets are summarized together. This approach allows to access a variety of structures in the single data block.

- *<the size of the value>* — Optional field that specifies a custom size of the value in the controller. The following sizes of types of values are provided:
 - *Integer*: — 1 byte(signed), 2 byte(signed by default) and 4 byte(signed).
 - *Real*: — 4 byte(float by default), 8 byte(double).
 - *Boolean*: — always one byte (with a bit through the point – DB1.10.1).
 - *String*: — 10 byte(by default) and 1–200 can be set.

An illustrative example of the overall process of the configuration of parameter form the template and to the values is shown in Figures from 5 to 8.

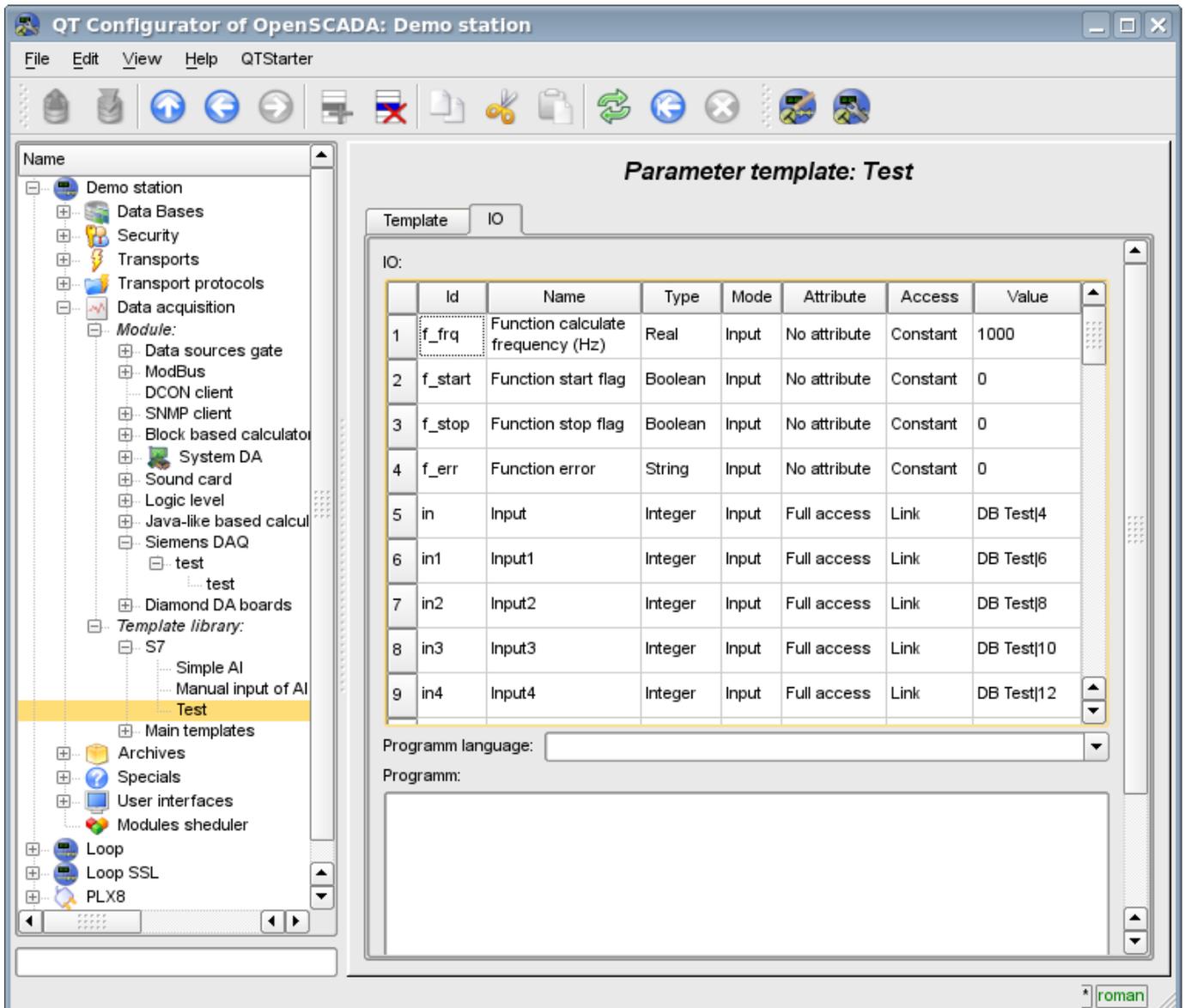


Fig.5. Example of the template with grouping.

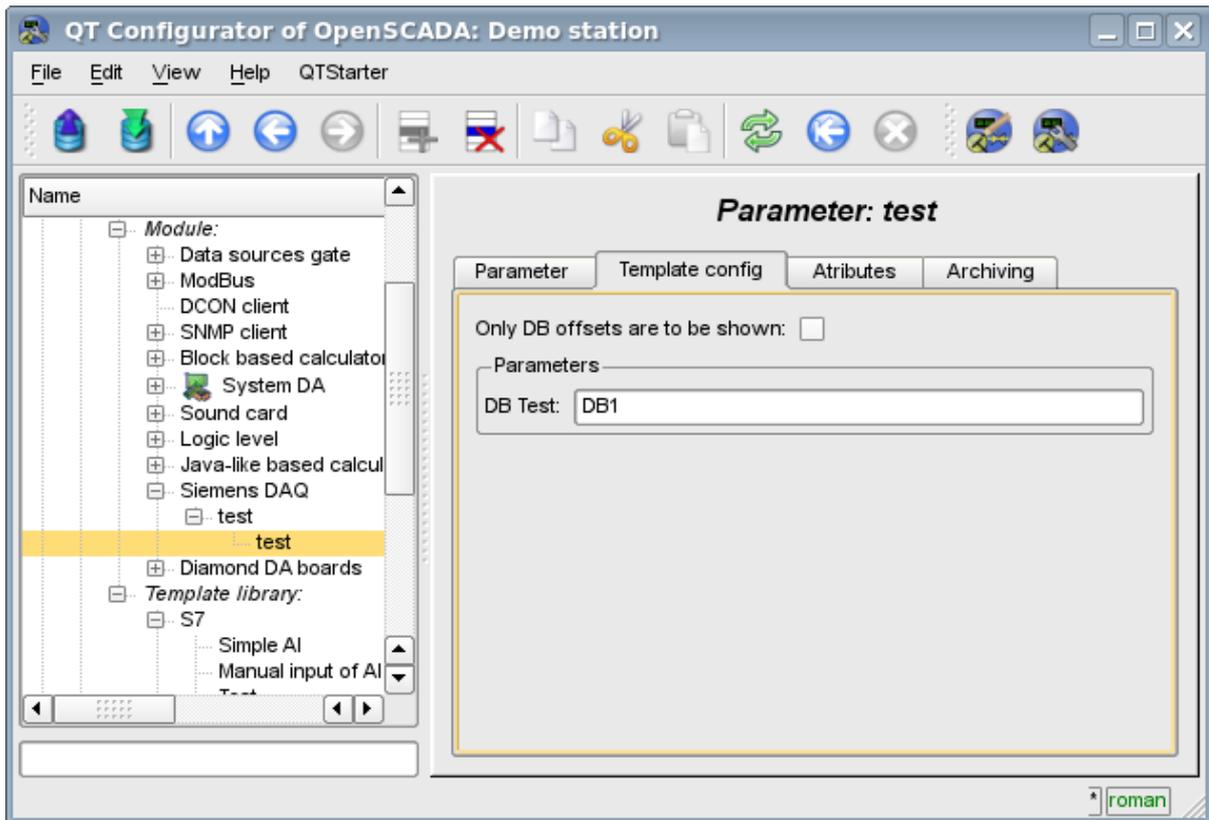


Fig.6. Configuration tab of the template of parameter

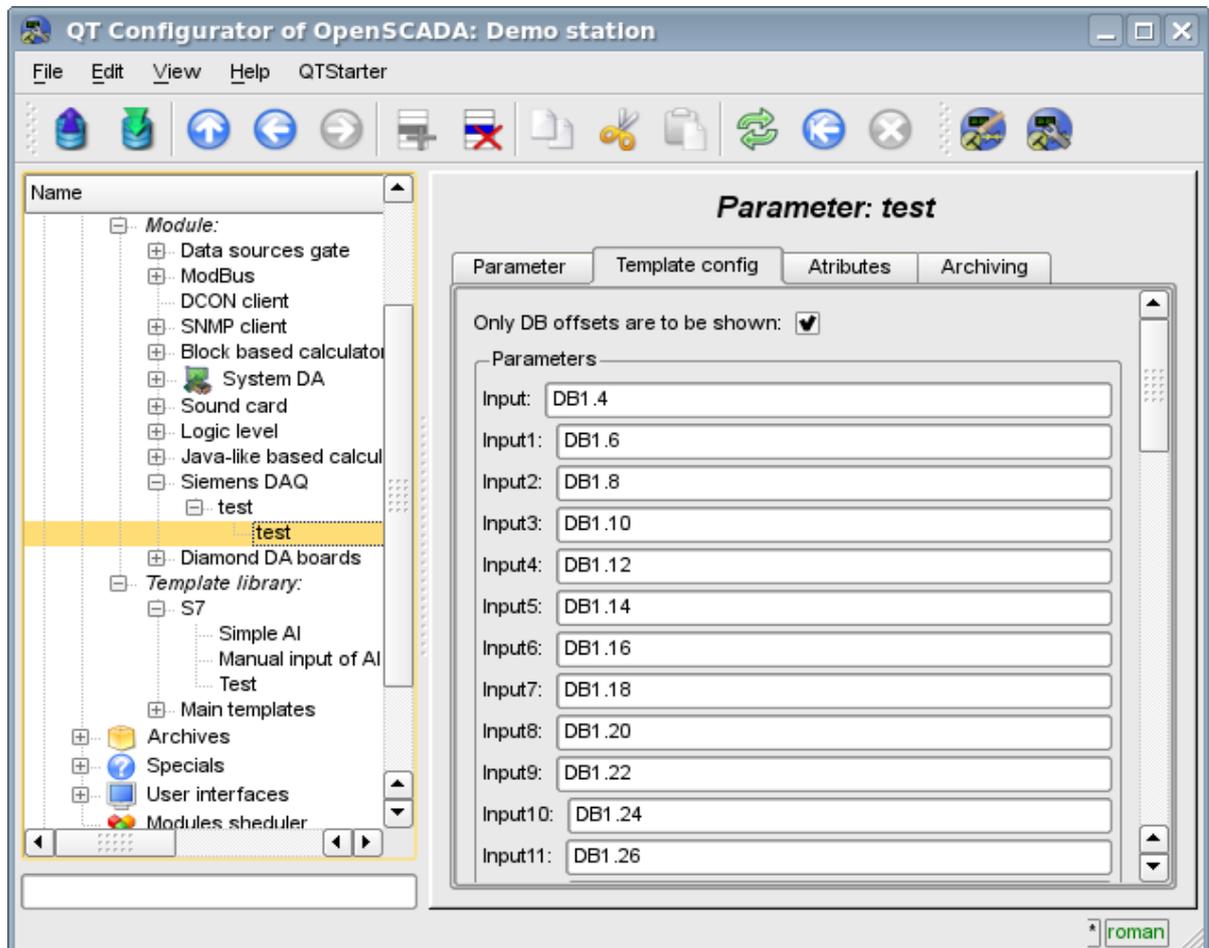


Fig.7. Configuration tab of template of the parameter with an indication of the parameters separately.

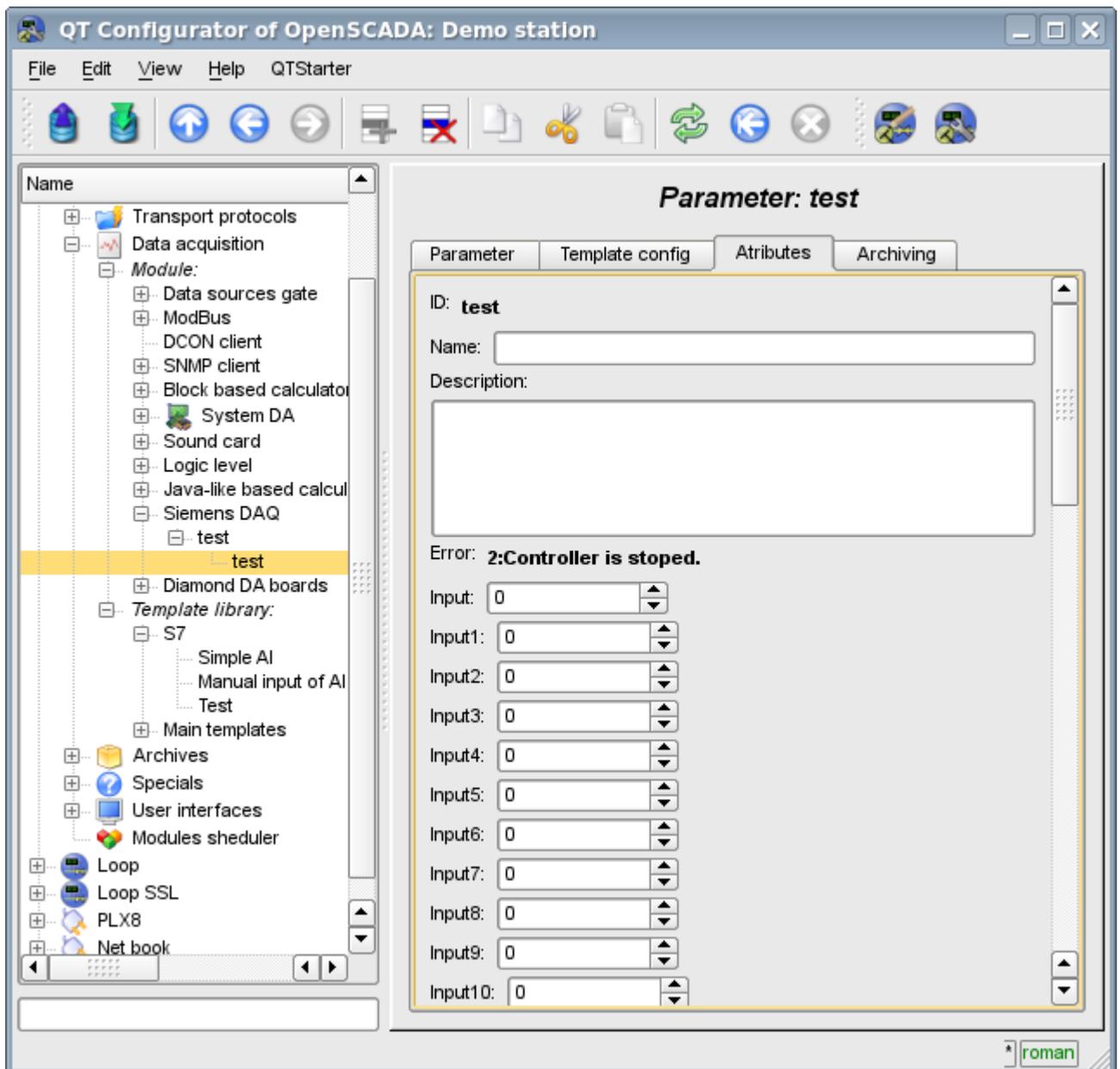


Fig.8. The values of the parameter.

Module supports only the data blocks(DB) of the controllers addressing!

4. Asynchronous recording mode

The standard recording mode for SCADA-systems interacting with the PLC, is the synchronous, because it allows to control the correctness of the conclusion of the record operation. However, in cases of recording multiple parameters at once and often, this approach is not justified in view of sending a multitude of small requests to the controller that overrides the PLC and has a large time interval. The solution is asynchronous recording of an adjacent values by means of the single block. This is supported by this module and allows you to record all parameters immediately by the adjacent blocks of 240 byte. Read and write in this mode is performed by the adjacent blocks with the periodicity of survey of the controller.