

# Simplifying the Engineering of Modular PROFINET IO Devices

Dominik von Rohr, Max Felser  
Bern University of Applied Science  
3400 Burgdorf - CH  
Dominik.VonRohr@bfh.ch / Max.Felser@bfh.ch

Markus Rentschler  
Hirschmann A&C GmbH  
72654 Neckartenzlingen  
Markus.Rentschler@belden.com

## Abstract

*PROFINET IO is designed with a modular structure of the IO devices. Certified modular PROFINET IO devices are usually shipped with a General Station Description (GSD) file that does not contain the actual module configuration of the IO device. With common appliances the modules must subsequently be configured manually in the engineering tool. This can become an elaborate task in large PROFINET IO installations and it would therefore be useful to have some tool support to automatically detect and import the actually installed modules in an IO device. As a solution we propose a GSD file generator that creates the GSD file with the actual module configuration by retrieving the real data from the device via SNMP. This paper presents the implemented mechanism and gives some attention to certification issues.*

## 1. Introduction

Industrial Ethernet networks [1] can be composed of various devices, which can be built modularly. For configuration of a PROFINET IO [2] network, pc-based engineering tools are used. The GSD files of the PROFINET IO devices must therefore be imported into these engineering tools. For a modular device the user has to identify and select the applied modules out of all possible modules tediously by hand. In large networks and due to the variety of available modules this can become a very elaborate task.

We propose a solution to this problem by including a special GSD generator in the delivery of every modular PROFINET IO device. This GSD generator creates the GSD file according to the actual module configuration of the device. Additionally, by applying a configuration specific name to a GSD, the user is easily able to build a configuration library identifying the different configurations by name. In chapter 2, a brief introduction to PROFINET IO is given and the engineering problem for modular devices is highlighted. In chapter 3 and 4, our solution is presented, whereas in chapter 5 the impacts on processes and tools are discussed.

## 2. Introduction to PROFINET IO

PROFINET IO is a widely deployed Ethernet-based communication standard. It is possible to integrate existing field bus systems such as PROFIBUS [2] or INTERBUS [3] into a PROFINET IO network [4][5]. PROFINET IO networks are composed of the following main types of devices:

- **IO controller:** This device is responsible for one or more IO devices. A configuration is created in an engineering tool, stored in an IO controller and submitted to the associated IO devices during system startup.

- **IO device:** An IO device is a field device which is connected to sensors and actors. Each device in a PROFINET IO network has to be certified and an associated GSD file has to be provided.

- **IO supervisor:** An engineering tool or supervision station used to commission and manage the IO devices in the PROFINET IO system.

### 2.1. GSD File

GSD previously stood for “Generic Station Description”, but was later renamed as “General Station Description”. A GSD file is a device type specific description file. This description is realized with the General Station Description Markup Language (GSDML) based on the general eXtensible Markup Language (XML) [7]. The GSD file complies with ISO 15745 and contains information about all characteristics of the device a PROFINET IO engineering tool requires. This includes

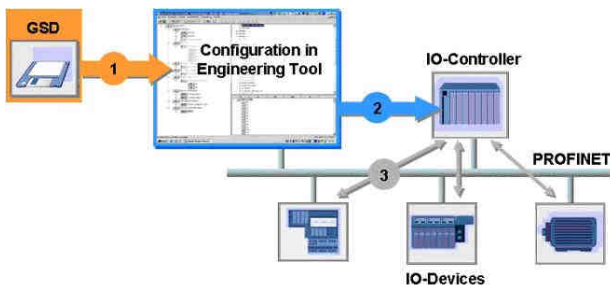
- IO device specifics such as parameters and alarms
- Device Access Points (DAP) as network interfaces
- the type of plug-in modules
- configuration parameters for each module
- error messages and other texts

Manufacturers of IO devices are obligated by PROFIBUS and PROFINET International (PI) [2] to provide a GSD file for the certification of their IO devices.

### 2.2. Engineering of PROFINET IO Networks

Installed PROFINET IO networks can be very extensive, containing many different devices. These

networks are configured and managed with engineering applications such as Simatic Step7, PCWorx, Molex. To configure a PROFINET IO network in such an engineering tool, the GSD file of each IO device has to be imported into the tool (Step 1 in Figure 1). Subsequently, all installed modules have to be identified, selected and the parameters of the IO devices and their modules have to be set. Subsequently, these configurations are loaded into the IO controller (Step 2 in Figure 1). Finally - at system startup - the configurations of every IO device are sent from the IO controller to all IO devices (Step 3 in Figure 1) [6].



**Figure 1. PROFINET IO network configuration**

Modular devices enable customers to build customized IO devices that fit their network requirements best. The GSD file of a modular IO device provides a list of all possible modules which can be connected to the slots of the device. To finally represent the actual IO device correctly in the engineering tool, the actually equipped modules must be identified and selected manually. This established approach becomes an elaborate and very time consuming task in large networks with a lot of modular IO devices. To increase efficiency in network engineering, it would therefore be helpful to have a tool that supports the detection of the actual module configuration of an IO device and automatically creates the associated GSD file before it is imported into the engineering tool.

The standard protocol in PROFINET for the configuration and commissioning of IO devices with an IO supervisor is the Device Configuration Protocol (DCP). To get access to the actual configuration of an IO device, the IO supervisor has to establish an Application Relation (AR) to the IO device and use a Data Record Communication Relation (CR), which is rather complicated and does not allow the complete identification of all modules and its parameters.

In PROFINET Conformance-Class B (CC-B) the support of Simple Network Management Protocol (SNMP) is mandatory for all IO devices. Network infrastructure devices like Switches are in CC-B regarded as IO devices. This makes it feasible to integrate the network diagnostic capabilities of SNMP into the diagnostics of the PROFINET IO application.

Hirschmann Automation & Control offers different industrial Switches that are supporting PROFINET CC-B and are highly modular (up to 26 different media modules are offered [8]). This creates a need for supporting the end user in the configuration process of these modular devices.

### 3. Proposed Solution

We propose a solution that uses the SNMP protocol to identify the installed modules in the IO device. With this method the GSD file generator collects information about the installed modules out of the Management Information Base (MIB) of the IO device with the help of the SNMP protocol. Using this information, a specific GSD file for this particular configuration can be built automatically. The following must be considered:

#### 3.1. Definition of the configuration

A GSD file can be created with a fixed configuration which is based on the real configuration. With this approach, all installed modules are assigned to their slot with the attribute "FixedInSlots". But in automation plants, the configurations of IO devices are not static. Modular switches are likely to be modified or extended with additional switching modules, e.g. if the network is extended during the lifetime of the installation.

A better approach is to assign the modules of the real configuration to their slots with the attribute "UsedInSlots". In this case all other possible modules of this IO device are kept in the GSD file as well and an adaptation of the configuration is still possible at a later moment.

#### 3.2. Identification of the configuration

Even though a modular IO device can support many different configurations, it is common practice to use multiple IO devices with the same configuration in an installation. Therefore it is desirable that an end user is able to define his own sets of typical configurations. To manage this efficiently an individual name for each configuration is required, assignable without restrictions by the planner of the network.

A typical engineering tool overloads GSD files with identical names in the device library. They will appear as two different IO devices in the library only if the GSD files are given different names. But if the DAP names within the files are the same, two IO devices will appear as two entries with the same name which are complicated to differentiate.

To ease the differentiation of various configurations, the configuration name must be added to all DAP names inside the GSD file. This allows a distinct identification of the configurations in the library of the engineering tool.

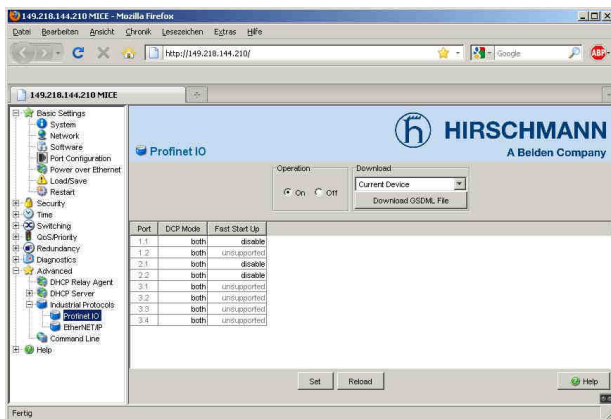
## 4. Realization

Hirschmann Automation & Control provides a web interface in all of their PROFINET IO capable Ethernet switches and routers. With this mechanism, a device can be managed and configured directly via a standard web browser. The web interface is an embedded JAVA applet provided to the browser application by a web server built into the device firmware. The information transfer related to configuration data items between the device firmware and web interface is performed via the SNMP protocol. The firmware provides access to all the system parameters of the device that are specified in the supported MIB databases.

### 4.1. GSD generator

For the first realization of the GSD generator an implementation in JAVA and integration into the web interface was used. Upon user request the collection of the information about the selected IO device is performed and the adapted GSD file is generated (Figure 2).

For further improvement, the GSD generator tool could be realized as a stand-alone tool or be integrated directly into the engineering tool.



**Figure 2. Webinterface with GSD Generator for a Hirschmann PROFINET IO device**

In the web interface (see figure 2) the user can choose whether he wants to download a basic GSD with all available modules included or a GSD file with additional information about the actual module configuration.

If the user decides to generate and download a GSD file containing the actual configuration by pressing the corresponding button, the GSD file generator issues SNMP-requests towards the IO device. All attached modules are registered in the device MIB. This information is retrieved by the GSD file generator and added as the actual module configuration to the generated GSD file - according to the slot rules of GSDML - and marked as "UsedInSlots". Additionally the user may name the configuration which hereafter is

added to the GSD filename and to all DAP names inside the GSD file.

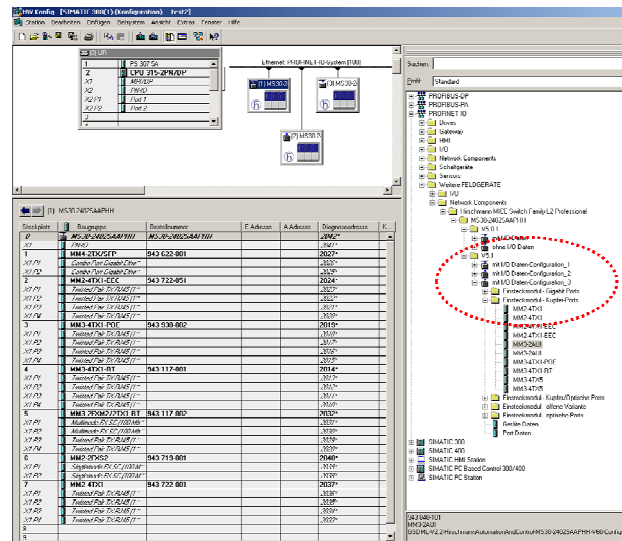
## 5. Results

The new GSD file generator allows the user to generate configuration specific GSD files.

### 5.1. Effect on the network configuration process

The new feature changes the network configuration process by increasing efficiency and by reducing possible failure causes due to the reduction of repetitive and error prone manual tasks.

In industry practice devices of the same type are often used in different applications. With the new functions the coordinating network engineer is able to build up a library with module configuration specific GSD files that can be deployed and reused in other projects which additionally increases efficiency.



**Figure 3. Simatic STEP7: Library with multiple configurations**

### 5.2. Effect on engineering tools

Handling of GSD files in the engineering tool is also significantly improved compared to the established approach. Since GSD files are managed by their names, there is now a better support to import different GSD files with different configurations (which are now identified by different configuration names), without losing a configuration by overwriting it.

In the device library of the engineering tool the requested device configuration can be selected by the user subsequently, based on the configuration name on the DAP.

### 5.3. Certification of the GSD File

PI [2] regulates the certification of PROFINET IO devices through an accredited PI Test Laboratory. Devices successfully passing the tests are awarded a certificate by PI and thereby permitted to use the PROFINET logo. To receive the certificate a sample of the IO device with associated GSD file must successfully pass several tests [9]:

- **State machine test:** The PROFINET IO communication related topics are tested with a standard PROFINET IO controller. The main topic of this test is to determine the correctness of the implemented communication state machines. But it is also verified that the GSD is importable by any standard engineering tool and that the GSD provides all necessary information about the PROFINET IO device.

- **Hardware test:** In this test suite the device hardware itself, such as its plug types and correct wiring of the plugs is tested. Additionally the functionality of the hardware is verified, for example auto-polarity, crossover function and auto-negotiation for 100Base-TX systems.

- **Interoperability test:** This are a series of tests designed to check the behavior of the device towards a supervisor. It includes the objective “checking the GSD-File” wherein a multitude of parameters and contents of the GSD file are tested, such as the slot rules.

Since PROFINET IO devices are certified together with a specific GSD file, there is an important topic that needs to be discussed regarding the new GSD file generating mechanism: How can a “dynamic” GSD file be certified when the certification process demands a “static” GSD file to be shipped with the device at the time of certification? Thanks to research and discussions with PNO members, the following facts can be outlined:

The dynamics of the modules in a GSD has already been used in the integration of fieldbus systems in PROFINET IO networks [4][5]. A field bus is represented by a single IO device in the PROFINET IO network and all the devices on the fieldbus are displayed as modules of the IO device. Therefore, at the moment of certification, it is not known for the PROFINET IO fieldbus interface, which “modules” will be part of the actual network. This situation is equivalent to devices with module configuration, since the only difference in the newly generated GSD files are these additional “module” elements.

Recapitulating we can state that only the part of the GSD file regarding the network is relevant for certification. The certificate guarantees that the certified PROFINET IO device with its associated GSD file can be integrated into an existing network without affecting it in an uncontrolled or negative manner. The additional module information in the new GSD files are not relevant at time of certification and will not affect the

network in any way, provided that the GSD generator adds the module information according to the slot rules.

## 6. Summary

Hirschmann Automation & Control provides to its customers a new solution for the simpler configuration of a PROFINET IO network. The new mechanism reduces the required time and effort and eliminates possible failure causes.

The new SNMP-based GSD file generator is able to create a GSD file containing the actual module configuration of a Hirschmann PROFINET IO device. A configuration name can be entered which is added to the filename and displayed in the library of the engineering tool.

This feature provided by Hirschmann devices is brand-new as it enables network engineers to build up libraries with different standard module configurations for used devices and therefore decreases the work load to set up a network.

Further improvements could be achieved by implementing the SNMP-based GSD generator as standalone application and/or by the generators subsequent integration in the engineering tool itself. In this case, the extension of the PNO Enterprise MIB with module specific information should be considered to allow a generic solution of the proposed mechanism.

## References

- [1] Felser, M.; , "Real Time Ethernet: Standardization and implementations," *2010 IEEE International Symposium on Industrial Electronics (ISIE)*, , vol., no., pp.3766-3771, 4-7 July 2010 doi: 10.1109/ISIE.2010.5637988
- [2] PNO, *Profibus User Organisation*, [www.profibus.com](http://www.profibus.com), 2011.
- [3] IB Club, *Interbus Club*, [www.interbusclub.com](http://www.interbusclub.com), 2011.
- [4] Jasperneite, J.; Feld, J.; , "PROFINET: an integration platform for heterogeneous industrial communication systems," *Emerging Technologies and Factory Automation, 2005. ETFA 2005. 10th IEEE Conference on*, vol.1, no., pp.8 pp.-822, 19-22 Sept. 2005
- [5] Jasperneite, J.; , "Fieldbus Integration to the Realtime Ethernet Standard PROFINET", *16th IFAC World Congress Prague*, Jul 2005.
- [6] Neumann, P.; Pöschmann, A.; , "Ethernet-based Real-Time Communications with PROFINET IO", *WSEAS 2005 Prague*,
- [7] W3C; , "Extensible Markup Language (XML) 1.0 (Fifth Edition)", *W3C Recommendation 26 November 2008*, <http://www.w3.org/TR/REC-xml/>, 2011
- [8] Hirschmann Product Catalogue 2011, available at <http://www.e-catalog.beldensolutions.com>
- [9] Profibus International; , "Test Specification for PROFINET IO Devices". Version 2.2.2, Jan 2008