Fast Track Engineering

SIEMENS

White Paper

When designing automation projects, a nearly seamless planning process and intelligent mass data processing are the most important factors for reducing time and costs.

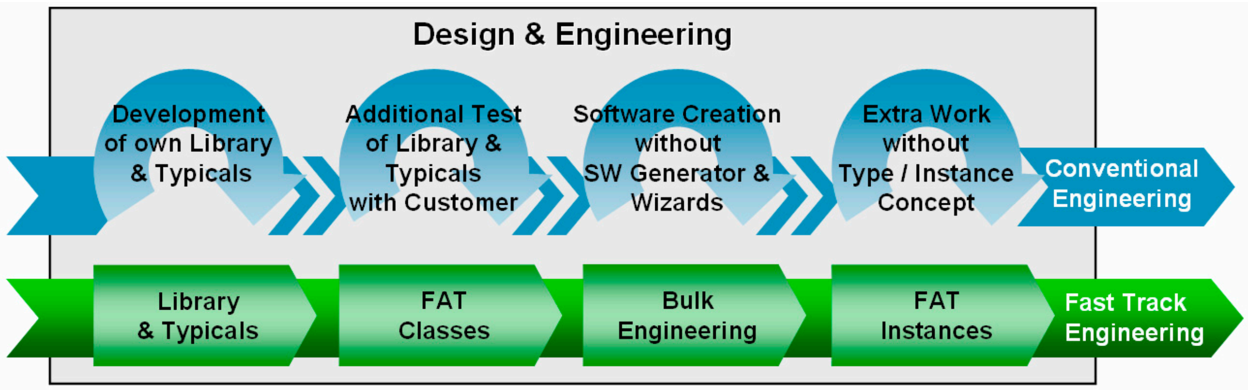
September 2011

Superior engineering quality, compressed project duration, tackling of system interfaces – these are today's challenges that must be confronted by process control technology engineering (or PCT planning). In order to master these, two factors play a decisive role: the bi-directional exchange of data between technical process and technical processes control planning, and the re-use of neutrally labelled system components or sub-systems. Structured, transparent work and comprehensive data management hold the promise of high efficiency – not only for the planning process, but also for changes or expansions to the operation. An explanation of which methods can be used to shorten engineering processes is provided using the distributed control system Simatic PCS 7 as an example.

Table of Contents

Motivation

Engineering of process plants is a complex technical process for which the work is frequently performed among various specialist disciplines. Since the growing complexity of modern systems has resulted in significant increases in information volume and density, the work of the planning engineers is supported by trade-specific tools. The use of various tools has resulted in a large number of interfaces within the engineering workflow. The transfer points frequently consume large amounts of time and expenses: Data must be transferred to the subsequent technical crew in a suitable form. This results in repeated data entry, consistency checks and conflicts.
Optimisation potential exists at the boundaries between individual trades, and also when re-using standardised solutions, such as software modules for PCT engineering of system components.[1] Even if each system is unique, a multitude of solutions from already planned systems can be applied to the new project. Furthermore, many forms of re-use apply in practice: Systems components that occur in large numbers in similar form can be efficiently applied to the project using sample solutions. Data consistency, central data management and repeated use of standardised solutions are important approaches for efficient system engineering.
For the second aspect in particular, a coordinated set of tools for mass data processing is a significant benefit, in addition to the early inclusion of the control technological engineering into the overall planning process.



[1] *in this regard also see R. Alznauer, K. Auer, A. Fay: Re-use of automation information and solutions. Automatisierungstechnische Praxis, Issue 3/2003, p. 31-35. Or: A. Fay, M. Schleipen, Mathias Mühlhause: How can the engineering process be systematically improved? In: Automatisierungstechnische Praxis, Issue 1-2/2009, p. 80-85.*

General Methodology

General Process Flow

The system planning process begins with the process description. The engineering tools of the individual trades are supplied with basic information by means of a written specification. An important tool for this is the P&ID – Piping and Instrumentation Diagram. This flowchart combines the technical process fundamentals with the pipeline technology and the process management technology requirements. Each planning step increases the level of detail and adds ever more information. Amounts and volume flows are determined for the ultimately intended process, equipment and their functions are assigned, etc. Using technical process flow descriptions, the automation engineer can generate function blocks for individual control functions, or prepare function block diagrams for process control sequences. All information and specifications are employed by PCT engineering. The software engineer refers to these in the P&IDs, the instrument and signal lists, functional block diagrams, current flow plans, etc. The PCT specifications derived from the information are implemented in the process control configuration. This process is iterative and the cycle is repeated several times. Only when the functional prototypes meet customer specifications are these released for mass data processing and equipped with customised logic and corresponding parameters.

Object Orientation

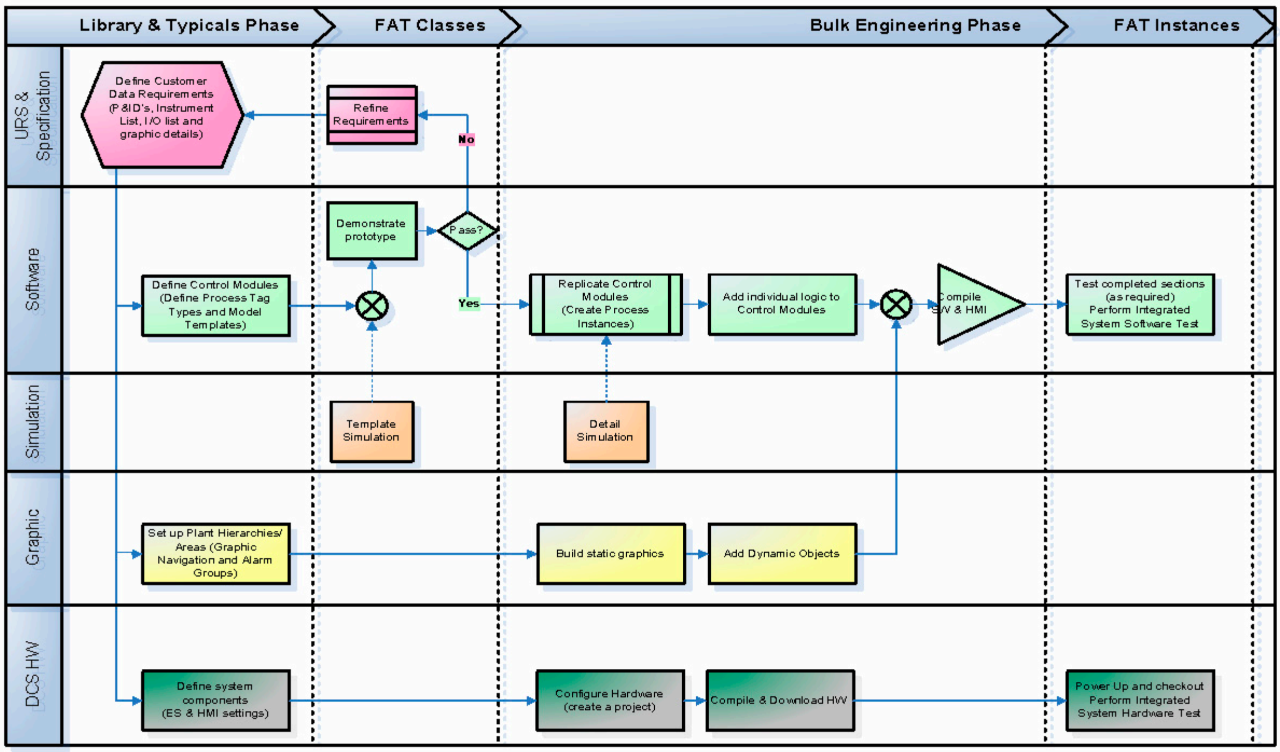
Object oriented modular technology has established itself as the fundamental technology for modern process control systems. Functions are combined in a function block; a complete control system is comprised of a multitude of modules. The goal of object-oriented software structuring is to create a simple yet unambiguous image of the actual system structure by means of a corresponding modularisation of the application software. For example, the motor control function has at least one function block assigned to it. This module realises the entire control technology for the motor, provides the required protection and monitoring functions, as well as operating and monitoring options. The entire functionality required for the use of the motor is combined into a function block. This enables a simple programme structure consisting of these functional modules, meaning that required functionalities are not repeatedly programmed anew. Once modules have been tested, they can be re-used in other programmes. Each module is assigned a unique type name, contains methods, statistical and dynamic parameters, as well as input and output interfaces for data exchange with other function blocks.
Function block libraries are frequently created where a type object is stored for each important process control function. Any number of object instances with unique features can be created from these for the design of the control programmes. Concrete values are added to the abstract descriptions of the type objects to make the object instances. Each instance of the type object performs the same function, but the actual values can differ. When modifying a function block, this rigorous separation of type and instance permits changes to be made only to the type object, and to then expand all object instances with a new functionality, without having to modify each individual object.

The Planning Phase

The general planning process described in the previous chapter plays an important role in system life cycles, since this is where the technical, organisational and economic foundation is laid. In regard to the quality of the automation solution, an important role is played during this phase by the interaction between the procedural and automation technology planning processes. The cornerstone for the control technology concept is laid during the concept planning phase: The foundation for measurement & control technology is recorded in the P&ID; field devices, apparatus and equipment are then assigned to these during the detailed plan, and recorded in the 2D or 3D layout plan. Functional specifications follow: formulas, operating modes, interlocks, as well as safety and shutdown functions. In conjunction with instrumentation and signal lists, these form the foundation for the individual control functions, which are matched to the task or selected from libraries. The structuring of the information, consistency checks and data analysis are crucial for the efficiency of all subsequent PCT planning steps. The same applies to the definition of the interfaces between the computer-based tools for the process engineers and the automation planners, since both make frequent revisions.

As soon as the functional prototypes correspond to the required specifications, the instances of the type objects can be created and the mass data processing can begin. The hardware and software project need to be generated in addition to the OS (operator system) images, notifications and alarms for subsequent process monitoring and operation. All planning steps, specifications and modifications are documented for improved comprehension.

Various disciplines refer to each other during the entire planning phase.

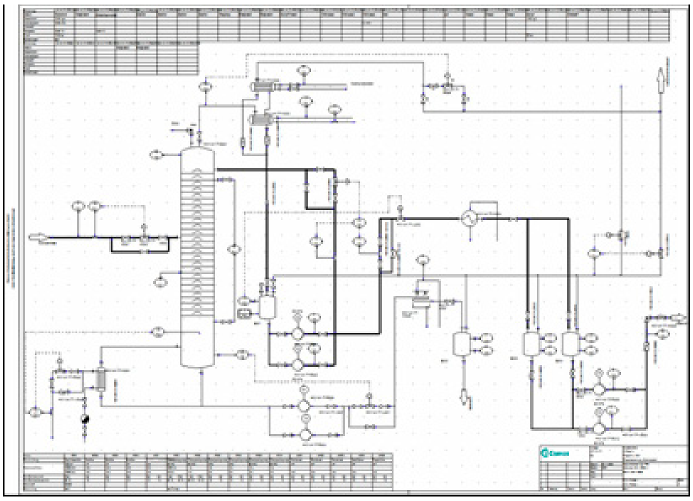


Data Foundation

The underlying basis for the PCT project is typically created with information from the so-called user requirement specifications (URS), i.e. the specification document and also data from planning tools such as EPlan, ELCAD, SmartPlant or COMOS PT. This data cannot be imported into the engineering system of a control system without complications, since neither the data structure nor the syntax match. When automating a process control system, it is important to be able to describe the individual automation functions in sufficient detail. Today this is primarily provided for by the P&ID with the corresponding labelling system, the measurement location list derived from this, as well as signal and interlock lists.

P&I Diagram

The pipeline and instrument diagram equally documents the structure and function of the process system for the process and automation technology. Since the tools for processing P&ID differ from PCT engineering tools, standardisation of communication formats and interfaces is mandatory. The IEC-Norm 62424[2]provides crucial information about this: With the neutral data-model CAEX, it introduces a data interface between tools for P&I processing and PCT planning. The XML-based and object-oriented data model enables data exchange, independent of manufacturer and trade. The principle is based on the storage of hierarchical object information, by which system structures can be represented. However, CAEX is not a rigid data model, since objects and relationships are not predefined. This also applies to the numerical identification of measurement locations. The user can continue using his established nomenclature model, provided that uniqueness is ensured. A unique labelling system is mandatory, especially when communication reaches beyond trade boundaries



Object orientation and a consistent data model are not only important for preparing P&ID flowcharts – efficient planning is based on these.

[2] *IEC PAS 62424 "Specification for representation of process control engineering requests in P&I Diagrams and for data exchange between P&ID tools and PCE-CAE tools", VDE-Verlag GmbH, Berlin. More regarding this in: Drath, Rainer / Mayr, Gerald: IEC PAS 62424 – Grafische Darstellung [graphical representation]: PCT tasks and data exchange for engineering systems. A norm for the interdisciplinary exchange of planning data across disciplinary boundaries of the process and control technology planning. In: atp – Automatisierungstechnische Praxis 49 (2007), issue 5, p. 22-29.*

Instrument List

The instrument list can be derived from the P&ID. It contains all measurement points used for the project, using a unique name (process tag) and detailed information. These names already contain basic information for the PCT engineer. Depending on the regulation, the process tag "TIC 240" can represent the following information:

- T The recorded physical metric is a temperature.
- I The temperature is recorded as an analogue value.
- C The temperature is used for control purposes.
- 240 unique number of the measurement location

The automation engineer will select a suitable type object that is consistent with this information and covers the functionality of the measurement location. Additional features, such as measurement location comments, type and manufacturer of the field device, value ranges, unit of measure, etc. are assigned to the measurement location. Paper lists that were used in the past have been replaced by spreadsheets. While these no longer require the complete re-entry of data, they are frequently characterised by inconsistencies and different information levels when information is exchanged between planning tools and the control engineering technology tools. This can frequently result in labour-intensive processing steps.

Instrument List

TH1	TH2	TAG No	Designation	PCS 7 Tag Type	Stationsname	Lin	Un	Hy	Lin	U
Distillation	Azeotrop Column	E_3402	1 Speed Motor wit	MOT_1sp_1fb_1cm__Std	AS01					
Distillation	Azeotrop Column	E_3403	1 Speed Motor wit	MOT_1sp_1fb_1cm__Std	AS01					
Distillation	Azeotrop Column	E_3404	Variable Speed Mc	MOT_Vsp	AS01					
Distillation	Azeotrop Column	FI_340101	Analog Monitoring	AMON__Std	AS01					
Distillation	Azeotrop Column	FI_340103	Analog Monitoring	AMON__Std	AS01					
Distillation	Azeotrop Column	FI_340104	Analog Monitoring	AMON__Std	AS01					
Distillation	Azeotrop Column	FI_340202	Analog Monitoring	AMON__Std	AS01					
Distillation	Azeotrop Column	FIC_340003	Controller for cont	CTRL_ConVal	AS01					
Distillation	Azeotrop Column	FIC_340302	Controller for cont	CTRL_ConVal	AS01					
Distillation	Azeotrop Column	FV_340001	Analog Valve witl	Val_An_ConVal	AS01					
Distillation	Azeotrop Column	FV_340004	Analog Valve witl	Val_An_ConVal	AS01					
Distillation	Azeotrop Column	LIC_340301	Controller for cont	CTRL_ConVal	AS01					
Distillation	Azeotrop Column	LIC_340401	Controller for cont	CTRL_ConVal	AS01					
Distillation	Azeotrop Column	LV_340301	Analog Valve witl	Val_An_ConVal	AS01					
Distillation	Azeotrop Column	PI_340001	Analog Monitoring	AMON__Std	AS01					
Distillation	Azeotrop Column	PI_340002	Analog Monitoring	AMON__Std	AS01					
Distillation	Azeotrop Column	QI_340301	Analog Monitoring	AMON__Std	AS01					
Distillation	Azeotrop Column	TI_340001	Analog Monitoring	AMON__Std	AS01					
Distillation	Azeotrop Column	TI_340002	Analog Monitoring	AMON__Std	AS01					
Distillation	Azeotrop Column	TI_340003	Analog Monitoring	AMON__Std	AS01					
Distillation	Azeotrop Column	TI_340004	Analog Monitoring	AMON__Std	AS01					
Distillation	Azeotrop Column	TI_340005	Analog Monitoring	AMON__Std	AS01					
Distillation	Azeotrop Column	TI_340006	Analog Monitoring	AMON__Std	AS01					
Distillation	Azeotrop Column	TI_340007	Analog Monitoring	AMON__Std	AS01					
Distillation	Azeotrop Column	TI_340102	Analog Monitoring	AMON__Std	AS01					
Distillation	Azeotrop Column	TI_340202	Analog Monitoring	AMON__Std	AS01					
Distillation	Azeotrop Column	TI_340301	Analog Monitoring	AMON__Std	AS01					
Distillation	Azeotrop Column	TI_340302	Analog Monitoring	AMON__Std	AS01					
Distillation	Azeotrop Column	V_340002	Valve with 2 Feed	VAL_2fb_1cm	AS01					
Distillation	Azeotrop Column	V_340003	Valve with 2 Feed	VAL_2fb_1cm	AS01					

Signal List

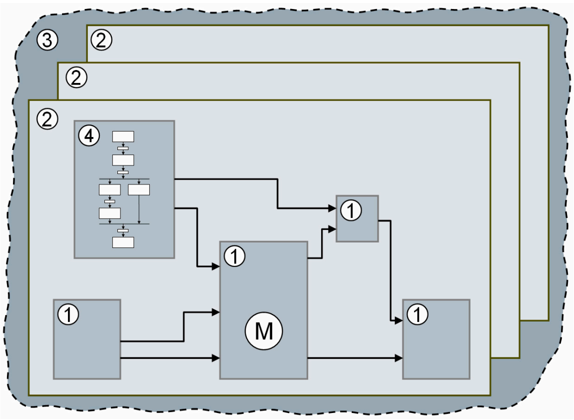
The instrument list forms the basis for the signal list. The signal list contains information about signals, signal parameters, as well as measurement points and range. On the one hand, the signal list can be employed to design the control system, and on the other to supply the automation engineer with data for developing the parameters of type objects. Similar to the instrument list, data is frequently exchanged between the planning tool and the automation engineering function by means of spreadsheets. Without standardisation, extensive manual manipulation and verification which must be repeated each time the planning data changes are necessary here as well.

Function Block Diagrams

The function block diagrams contain information about those measurement points that must be locked under certain circumstances: To maintain safe system operation, it is necessary to monitor process manipulations, to potentially suppress user inputs, to control actuators, and to disable functions. If the interlock condition is activated while the system is in operational mode, the pre-programmed safety function must be executed.

Data Structuring and Modularisation

The larger the system, and the more complex the processes, the more information must be provided by the planning tools to the automation engineers and their systems. A clean data structure and a sensible process model are essential to achieving the smoothest possible processes, to realising short project durations, and to reducing costs. Hierarchical structuring and modularisation are interdependent. Whereas the hierarchical structure is largely determined by the process control system, modularisation is driven by the process control implementation. Simatic PCS 7 supports the hierarchical structuring principle by means of the technology hierarchy. The project structure and the technology hierarchy (TH) are implemented by creating hierarchy folders. CFC and SFC plans for the automation systems, images and reports for operator stations and supplemental documentation are stored in the folders. A well-planned technology hierarchy promotes finding objects and is a condition for re-use of generic solutions, as well as automatic generating mechanisms.
The modularisation and re-use principle is realised in Simatic PCS 7 projects by mass engineering features. The re-use of standardised and parametric solutions provides a high potential to increase efficiency, however, the selection must be made with great care. The following introduces various basic elements for efficient PCT engineering.



Representation of the data structure:
(1) = Function block type
(2) = Control module type
(3) = Template
(4) = SFC-Type

Function Block Type

Function block types or module types are pre-coded programme elements used to process repeating functions. These can be inserted into CFC plans (continuous function chart), and can then be encoded there as an instance with the proper parameters, and adapted to the project requirements. For object-oriented engineering, the module type defines the characteristics for all instances (conductions) of this type. To ensure that a module type is used throughout the project in only one version, it should be stored in a central function block library. Control technology equipment, such as valves or motors, can be modelled in the PCT planning with the aid of function blocks. Control inputs are generally performed indirectly via an instance of the corresponding module type. Function blocks permit control, monitoring and operation of such equipment by providing corresponding interfaces for positioning and adjustment signals, as well as for parametric and monitoring functions. Such modules can also contain interlock functions, which automatically transition the device into a defined safety setting.
Function blocks form the basis for developing performance solutions. If these are comprehensively tested, as is the case with the Siemens DCS Simatic PCS 7, their use ensures consistently high quality and reliability of the deployed algorithms. They modularise and type-specify repeating functionalities, which can be changed centrally and therefore markedly reduce the effort for developing performance automation solutions.

Control Module Type

A control module type is employed for basic automation of certain control technology functions, such as a level control mechanism. Control module types contain individual function blocks. These are inserted into the CFC plan. In this case all input and output parameters are uniquely defined as parameters or signals. A control module type is established from this CFC plan with all generally applicable parameters. Based on an import file, Simatic PCS 7 can create any number of measurement points by copying a control module type with the aid of the import/export assistant (IEA). The measurement points can then be manually adapted and correspondingly connected to other specific automation tasks. Here as well, the type-instance concept preserves the benefit of central modification. The use of control module types contributes significantly to reducing engineering costs.

Templates

Templates permit the configuration of even larger units, such as sub-systems (so-called units) as generic solutions. This is an advantage when projects feature similar structures. A process control unit consists of system components, such as apparatus and/or machines, inclusive of sensors, actuators and matching automation software. Examples for these are reactors, distillation columns or tanks. A template encompasses all functions required for automating this unit: CFC plans, SFC plans, OS images and reports. These elements are compiled into a hierarchy folder. The template is centrally stored in a master data library. After the engineering and the assignment of an import file, any number of instances can be generated automatically. These instances can be adapted to the specific automation task.

The common objective of the basic elements presented here is the re-use of complete, fully parameterable solutions – from individual control modules to entire units. This primarily creates benefits for the frequent changes during the planning process, since only one type needs to be processed centrally, and the change can then be transferred to the instances. This results in increased efficiency during tests and validations as well. Only the type object needs to be tested and/or modified, and not all the individual solutions. The operation also benefits from the similar structure of the solutions during the debugging process.

The basic elements are developed based on the international norms IEC 61508 and 61511, i.e. by applying Functional Safety Management processes, systematic defects are already avoided during the design process, and the software can be used as a part of the system safety life cycle later on.
When designing an automation system using Simatic PCS 7, general design principles for complex systems can be employed, which have been proven on multiple occasions. The three most important principles are[3]:

- the principle of hierarchical structuring, which is ensured by generating the technological hierarchy.
- the principle of modularisation, which is adhered to by the strict object orientation, the type/instance concept, and the library elements in connection with this.
- the principle of re-use, which is preserved by the presented templates.

SFC Type

SFC types (sequential function charts) permit the definition of process controls as re-usable sample solutions. An SFC type is a sequential function that is prepared with an editor and can be inserted into a CFC plan. In this case, a process capable SFC instance is prepared with concrete process connection. This permits the creation of SFC types for all processes that occur on multiple sequences in a project, such as heating, agitating, starting, etc. However, the use of an SFC plan makes more sense if a process sequence is used only once. In contrast to the SFC plan, the SFC type does not have process sequence characteristics, but instead has the benefit that instances can be created. The user also benefits from the central ability to make changes: When adding process values or set points in SFC types, these are automatically added to all SFC instances.

---

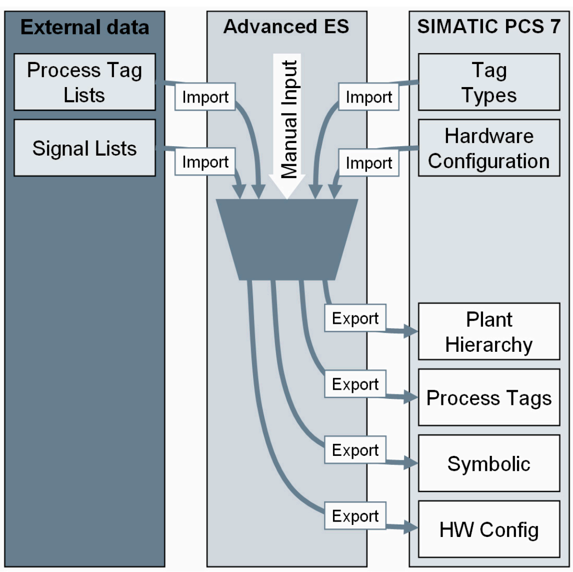
[3] *see also Lauber, R., Göhner, P., Prozessautomatisierung 2, Springer Verlag Berlin, 1999.*

Mass Data Processing

By preparing and/or selecting the standardised basic elements that are appropriate for the automation project, an important step for the detail engineering has been completed. Now, a large number of objects that match the specifications have to be prepared in an expeditious and error-free manner. The instancing process of the respective types described in Simatic PCS 7 makes the creation of the derivatives possible with ease. Next to the creation of hardware concepts and measurement points, preparing the system charts, messages and alarms presents a challenge during this phase.

Process Tag Generation

The objective of efficiently generating measurement points is to largely generate process tags automatically by using already prepared control module types and the specified instrument lists. The seamless data exchange between the procedural planning and the control technological engineering is ensured by software tools, such as Simatic PCS 7 Advanced Engineering System (AdvES) by Siemens. It prepares data for use in the PCS 7 Engineering System (ES), and provides an application for data import and for mass processing. After an initial assignment, AdvES recognises the control module types, and automatically generates the control module instances. This principle enables the compilation of various designations with the same meaning across process interfaces.



This results in Simatic PCS 7 creating the technological hierarchy, control module instances with signal and parameter settings, as well as the hardware configuration.

Graphics Generation

The preparation of process and system diagrams for operating and monitoring systems is greatly simplified if the basic elements, such as individual control modules, are equipped with corresponding operating and monitoring symbols. Ideally, these static or dynamic symbols are organised in libraries and can be inserted automatically into the corresponding image during development of the process displays. Pre-configured image modules (faceplates) can be recalled via module symbols. These dynamic image elements, which are connected to the control module by means of parameters, are the foundation for the user interface in Simatic PCS 7.

The process pictures created using this method are inserted into the previously automatically created technological hierarchy to provide a well-structured operating and monitoring platform to the system operator.

Messages and Alarms

These must be prepared to inform the operators about events. Various classes of messages are defined, depending on their origin. Control technology messages are triggered by component malfunctions that are recognised and notified by the control system. These have to be separately created in Simatic PCS 7, since they are prepared automatically by function blocks, and already exist. Process notifications, such threshold limit violations of measured values, or operating notifications, provide information about process events. These are also already pre-defined for modules from the PCS 7 standard library, therefore eliminating the need to separately prepare these. However, based on need, message texts can be changed, and a notification priority can be defined in the object characteristics of control module types. Operating messages are created when process parameters are manipulated. These are also already prepared in the available display modules and are generated automatically.

Dataflow when using the Simatic PCS 7 Advanced Engineering System

Safety Applications

Safety systems (safety instrumented systems, SIS) generally play an important role in process control systems. The definition and description of the SIS requirements, in the form of specifications, are the basis for planning safety applications. The safety requirements specification contains the functional description of the safety functions, and all conditions by which these are triggered. With the aid of the Simatic Safety Matrix, safety functions can already be recorded, described and formulated during the basic planning phase. Process engineers can populate the Cause & Effects Matrix with information without having programming knowledge. The definitions of causes, their effects and the corresponding responses are table-based. The described safety functions can be deployed in the Simatic PCS 7 Engineering-System without conversion, manual adjustments, or similar means. The conversion to programme processes is performed automatically: The Simatic Safety Matrix Engineering Tool creates logic on the basis of control module types by using function blocks from the library for safety-oriented applications and generates the corresponding channel drivers for all safe input/output channels.

Route Control

To plan systems with expansive route or pipeline networks and flexible material transportation paths represents a particular challenge. This challenge can be tackled optimally if all technological functions required for the material transport are available in the library as standardised elements. Such modules can be interface modules that are specific to material transport elements, such as control, sensor or expected value elements, route modules or sequencing functions. The route control library is available to Simatic PCS 7 for this purpose. The route control assistant automatically accepts route control-specific project data to import into the PCS 7 project. It performs a plausibility review, defines the communication connections and creates the corresponding notifications.

Conclusion

Specific conditions need to be fulfilled to structure the automation technology engineering with Simatic PCS 7 in the most efficient manner. Initially, the automation project should be tied into the overall planning process as early as possible. The modularisation of the systems from an automation technology point of view simplifies the project structure. Object-oriented planning and the re-use of control module types and/or templates using the type/instance concept are elementary in this case. Centrally modifiable standardised function blocks, even for advanced process controls, already exist in the Simatic PCS 7 standard library. The engineering effort can be greatly reduced with the aid of these tested modules. Of course, Simatic PCS 7 can also be used to conveniently create application-specific modules, allowing one to flexibly realise entirely unique requirements. However, in doing so, it is worth noting that the in-house preparation, care and maintenance of modules consume resources. Therefore, a careful review is needed to determine if this effort is worthwhile.
The deployment of the AdvES Software-Tool not only permits the import of system data from CAD/CAE tools but also provides for the option to automatically generate the AS configuration by copying control module types and templates, and also to process the parameters. This ensures prompt engineering with a minimum of errors. Generating process pictures, messages and alarms for instance-specific modules such as valves, regulators, pumps etc., is performed automatically within the PCS 7 engineering.
Furthermore, it is also important to optimise the exchange process between the various planning tools to reduce engineering time and effort. Standardised interfaces permit the exchange of data across trade or tool boundaries. In this case, the Simatic PCS 7 Advanced Engineering System can provide valuable services with unambiguously defined exchange mechanisms, standardised interfaces, automated data consistency and plausibility checks.
Centralised engineering of all aspects specific to automation, including batch automation, route control, safety applications with the help of coordinated tools and pre-coded solutions, not only compresses project durations, but also has a positive influence on the overall life cycle costs.

www.siemens.com

Siemens AG
Industry Automation
IA AS S MP
Roland Wieser
76181 Karlsruhe
Germany

Bibliography

- Alznauer, Richard / Auer, Karl / Fay, Alexander: Wiederverwendung von Automatisierungs-Informationen und -Lösungen.
  In: atp Automatisierungstechnische Praxis 45 (2003), no. 3, pp. 31-35.
- Au, Helmut von: Effizienzsteigerung im Anlagen-Engineering.
  (White Paper, Siemens AG, Industry Sector, March 2010, www.siemens.com)
- Fay, Alexander / Mühlhause, Mathias / Schleipen, Miriam u. a.: Wie kann man den Engineering-Prozess systematisch verbessern?
  In: atp – Automatisierungstechnische Praxis 1-2.2009, pp. 80-85.
- Humphrey, David W. / Miller, Paul (Hrsg.): Modern Engineering Tools Drive Productivity in Manufacturing.
  (ARC White Paper, November 2010, ARCweb.com)
- IEC PAS 62424 „Specification for representation of process control engineering requests in P&I Diagrams and for data exchange between P&ID tools and PCE-CAE tools“, VDE-Verlag GmbH, Berlin.
- Lauber, Rudolf / Göhner, Peter (Hrsg.): Prozessautomatisierung, vol. 2. Modellierungskonzepte und Automatisierungsverfahren, Softwarewerkzeuge für den Automatisierungsingenieur, Vorgehensweise in den Projektphasen bei der Realisierung von Echtzeitsystemen, Berlin u. a. 1999.
- Mayr, Gerald / Drath, Rainer: IEC PAS 62424 – Grafische Darstellung PLT-Aufgaben und Datenaustausch zu Engineering-Systemen. Eine Norm für den interdisziplinären Austausch von Planungsdaten über die Gewerkegrenzen der Verfahrens- und Leittechnikplanung.
  In: atp Automatisierungstechnische Praxis 49 (2007), no. 5, pp. 22-29.
- Pfeiffer, Bernd-Markus / Lorenz, Otmar: Unit-orientierte Musterlösungen für Advanced Control. Beispiel Destillationskolonne.
  VDI-Berichte no. 2032, 2008,pp. 11-14.
- SIMATIC Process Control System PCS 7 Engineering System (V7.1) Configuration Manual, published by Siemens AG, Nuremberg 2009.
- SIMATIC PCS 7 V7.1 GMP Engineering Manual Edition 06/2010. Guidelines for Implementing Automation Projects in a GMP Environment, published by Siemens AG, Industry Sector, Industry Automation, Karlsruhe 2010.
- Vogel-Heuser, Birgit (Hrsg.): Automation & Embedded Systems. Effizienzsteigerung im Engineering, Kassel 2009.
  (Embedded systems I, Tagungen und Berichte 1)