The LASAL project software offers all the advantages of modern automation technology. The use of innovative programming techniques enables fast and easy realization of machine concepts. Development time and time-to-market cycles are significantly reduced while having a higher quality of software. The reproduction of individual engineering objects with LASAL simplifies the communication between designer and software developer.

**Efficient and Simple Automation with the LASAL Software Package**

**Object-Oriented Programming**

LASAL is a highly modern and unique project tool: SIGMATEK was the first company to integrate object-oriented programming with client-server communication and graphic representation in control technology. With object orientation, LASAL defined a new standard for modularity and reusability. Because the properties of a class can be inherited, a classification of program parts in hierarchical levels is possible. Clear overview and structure are therefore guaranteed.

**„All In One“ Solution – One Software Package for All Aspects of Automation**

LASAL provides all functions needed for solving automation tasks in one package. PLC and high language programming, visualization, drive technology, service and diagnosis, as well as remote maintenance.

- **LASAL CLASS**: Object-oriented project development with highest modularity
- **LASAL SCREEN**: Visualization with all SIGMATEK display units
- **LASAL MOTION**: Drive technology
- **LASAL SERVICE**: Service and remote maintenance
Clearness through Graphic Representation

The complexity of the program is encapsulated in the graphic representation of the program components. This means, the program code is not visible at the first glance. The relationship of the program components to one another is shown, as well as the most important component data. This gives the developer a quick overview of the project structure with the interaction between the individual modules clearly visible. So, service technicians can diagnose the system quickly and easily.

Client-Server Technology

The communication between the program components is made possible through the clients and servers of a component. The client is the active part of the communication and the server reacts to requests of the clients. Using this technique, an event-triggered system can be implemented:

A program component will then work only when it is addressed (when a value is set in the visualization project for example). The CPU base load is significantly optimized in comparison to conventional systems.

LASAL Supports All Target Platforms

LASAL can be used with all platforms. The entire spectrum of SIGMATEK products, such as CPUs, terminals and industrial PCs are supported. Platforms can also be changed at any time without having to adapt the software.

Open Data Interface

For individual data exchange with external programs, an OPC server (OLE for Process Control) with a standard interface is available. The server supports write/read access on the respective data server. The user can then link, for example, process data to a visualization project, control system or an Excel file and create individual evaluations without worrying about proprietary data formats and protocols.

SIGMATEK also provides a DLL for online communication, enabling access to control data from Windows programs.
Real-Time with a Multilayer, Multi Tasking System

The real-time operating system supports tasks with various priorities and allows preemptive as well as cooperative task changes.

The visualization and the application can therefore run on one processor.

Worldwide Service Interface

Through the use of web technology, remote maintenance can be performed per modem or Internet regardless of location. The service technician can modify the program or perform a remote diagnosis quickly and easily.

Online Help System

LASAL has a comprehensive help system that describes all functions in detail. With help from a demonstration project, a fast introduction to the system is guaranteed.

Multilanguage Programming Tool

LASAL supports the most important programming languages. Applications can be created in Structured text (ST) Instruction list (IL) and Ladder diagramm (LD), all three comply with the IEC 61131-3 standard, Sequential Function Chart (Interpreter) and ANSI-C.
LASAL CLASS (Control Logic Application Software System) is “the” project tool to solve your automation tasks.

With a continuous operational concept and a clear user interface, LASAL CLASS offers a comfortable design environment for creating high performance control programs.

Object-Oriented Programming

LASAL expands the IEC 61131-3 standard with object-oriented programming and guarantees the simple and safe development of applications using the most modern technology. The total software costs are also reduced since the modularity, reusability of the classes and encapsulation of the user programs make applications more efficient and easier to test.

Behind each object, there is a class containing the program code and the corresponding data elements. Each class can therefore assume a specific task, such as measuring and evaluating temperature, filter functions, motor control, etc.

The various classes are managed in a clearly organized library (Class Library). The classes are linked to the project tree through importation. For information exchange, defined interfaces that can send and receive messages are available.
A class is integrated into a network from the project tree using the **drag & drop** method, creating an instanced object. The objects in the application need only be connected to one another and an application is created.

**Class Library**

The classes can be managed in a central library. When a class is imported, all required program components are copied, linked or referenced into the project tree.

In LASAL CLASS, a comprehensive library is available that contains a collection of various classes, such as memory, time and date, motion, logic, triggers, controllers, etc.

The user can create and use as many individual libraries as desired.
Inheritance, Aggregation and Creation of Complex Classes

LASAL CLASS supports modern programming techniques including inheritance, aggregation and creating complex classes.

**Inheritance** describes a relationship between general classes (base class) and a derived class. The derived class "inherits" the properties and is consistent with the base class, but contains additional information (attributes, operations, associations). Using this technique, new characteristics for machine parts can be implemented with minimum programming effort.

With aggregation, several individual classes can be combined into a complex class.

**Example:** The class »Counter« is defined as the base class from which the properties are inherited by the »PieceCounter« class. The »Ram« and »Trigger_plus« classes are also integrated into the »PieceCounter« class through aggregation.

Construction of a complex class through aggregation and inheritance
Presentation of the instanced, complex class in an object network

**Client-Server Technology**

As in a PC network, the client requests services that are provided by the respective server. Read and write access occurs over ONE connection.

Communication between server and client
- Servers automatically provide **standardized methods** as basic structures for designing a **class**. This standardization allows classes to be freely exchanged.

- If a **new method** is added to a class, **LASAL CLASS** supports the automatic creation of the method header with all transfer and return parameters. The code body is generated automatically.
- Servers and clients can be used as command channels, data channels or object channels.

Data Channel

A data channel provides standardized read/write methods for simple data exchange (strings, binary data, etc.). A heat regulator, for example, can be connected to any sensor to read its temperature value.

Command Channel

In addition to the methods of the data channel, standardized interfaces are available with the command channel. So it can send more complex commands with transfer and return values.

Object Channel

An object channel has the same properties as the command channel but refers to a specific class and therefore establishes the type security; only objects of the referenced class (and their derivations) can be connected. This avoids mistakes when assigning objects. With an object channel, all global methods of the referenced class can be retrieved. For example, a stepper or servomotor can be controlled over an object channel, provided that both classes have a common base class referenced to the object channel.
Graphic Representation

An essential property of LASAL CLASS is the object-oriented programming with graphic representation, graphic project construction and graphic debugging. A complete overview of the project, functions, data traffic and interfaces is available at a glance. Complex relationships are therefore easier to recognize, change and monitor.

Version Control System

As an additional plus, all source code files in LASAL CLASS are pure ASCII files; the implementation of a version control management is therefore possible.
Source Code Online Debugger

For fast and easy program troubleshooting, a Source Code Online Debugger with functions such as breakpoints, conditional breakpoints, scan counter, single-step processing and force is available.

Online debugger with real-time values

Scripting

With the standardized script language, Python, LASAL CLASS projects can be created and modified remotely-controlled. This allows various characteristics and types of one installation to be derived from the base project.

Real-Time Oscilloscope

The oscilloscope provides a real-time display of signal processes with an additional history function. The oscilloscope can be displayed in trend or classical view (with or without luminescence). In addition, a start trigger can be set, and a hold function is also integrated.
Task Viewer

With the "PlcTraceView" tool, the project time response can be recorded. This tool is used to analyze, diagnose or measure the response time of the task.
LASAL SCREEN is an HMI tool (Human Machine Interface) for the visualization of projects with all SIGMATEK display units.

**Simple Operation**

To create a visualization project, knowledge of a programming language is not required. LASAL SCREEN can be operated independently of the programming tool, resulting in short training time as well as a fast and simple start in project building.

**Import of Variables**

LASAL CLASS defines the variables available for the visualization project. Variables from one or more LASAL CLASS projects are imported to a LASAL SCREEN project. During runtime, the visualization project can access variables from several controls directly.

**Flexible Screen Construction**

The LASAL SCREEN tool supports various resolutions for all SIGMATEK graphic displays. The visualization projects can be scaled to various screen sizes both during and after the design phase.

In the target system, the operator interface (touch screen, keyboard or mouse) can be selected. The operation mode can also be individualized after completion of the visualization project. For project construction, integrated designs and a large graphic library are available. User-defined graphics can also be imported in standard formats (BMP, GIF, JPG).

With a global screen and individual screens derived therefrom, the project time can be reduced significantly. In addition, LASAL SCREEN provides such functions as alarm management, event management (logbook), trend representation, bar diagrams and recipe management.
Standard Graphic Functions

For creating screens, numerous standard functions for drawing, aligning, importing and translating are provided. Existing graphics in standard formats (BMP, GIF, JPG) can also be linked to the project, which results in significant time and cost savings.
Screen Configuration

Elements on the screen can be made visible or invisible through a predefined server or by setting a check bit. This offers remarkable advantages with serial applications since the screens adjust automatically depending on the machine type.

Online Language and Unit Conversion

LASAL SCREEN can combine any number of languages in a single project. Text information is input in ASCII or UNICODE format, thus all languages are supported. For the translation into another language, no knowledge of the tool is necessary. The translator contains a TXT file (ASCII or UNICODE text) that is later re-imported.

To ensure the clarity of the text, any number of text lists with individual names can be created. When compiling the visualization, a choice of languages needed can be chosen out of the whole language pool. This saves memory space in the target system. Individual languages from the project can be installed later in an existing machine.

Units can also be changed using the online conversion. Temperature can be converted from °C to °F, length from mm to inch or weight from kg to lbs automatically. The required conversion type is assigned directly to a variable. The programmer (LASAL CLASS) does not have to take any action for unit conversion, because all values in the system are always available in the LASAL basic unit.
Bubble Help

Each variable can be assigned an additional help text. If this variable is selected on the screen, additional information can be shown in the form of a bubble help text. This function provides the user with a better overview of the control functions.

Input Elements with Pull-down Menus

The operation of an input element can be effected through entering values directly, setting upper and lower limits or with a pull-down menu. In a pull-down menu, defined parameters are set in the form of a menu selection. This function provides the user with a selection of several setting options and raises the operational safety through a precisely defined status selection.

Exporting and Importing Project Parts

In LASAL SCREEN, project parts such as pictures, text lists and variables can be selectively exported or imported. Since a SCREEN project is generated through exportation, the user can create libraries with reusable elements here as well.
LASAL MOTION provides a large “construction kit” for the control of movements: all motion functions such as absolute, relative and endless positioning, CNC functions as well as coordinated movements and several referencing types are standard features.

The library contains a selection of coordinated movements such as linear interpolation with up to six axles, circular interpolation, cam discs, gear functions, flying saws as well as electrical waves and path movements. Axle movements can be executed using simple data inputs or instructions without any programming.

In LASAL CLASS, there are predefined parameters for SIGMATEK DIAS drives and motors. **System start-up, configuration and diagnosis are therefore simplified and the time needed is reduced.** In addition, **possible error sources are avoided.** User-defined parameters can also be stored, ensuring the optimal fit for specific user requirements.

Start-up and data diagnosis of the DIAS drives can be displayed at the same time on one screen.

The LASAL MOTION package simplifies all tasks involving drive technology. The modular construction allows optimal implementation of drive concepts. The project construction/start-up software for the SIGMATEK drives is integrated and a large drive library is available.
The configuration data is stored in the control system and so the drive always has the correct parameters. The servo amplifier can therefore be exchanged without additional labor and without software tool.

With LASAL MOTION, linking to virtual axles, creating and executing profiles, coupling several axles and applications from flying saws to unconventional applications are simple to realize.
Numerous service tools complete the LASAL automation software package from SIGMATEK. Remote maintenance, software updates and data exchange are performed comfortably using LASAL service tools.

### LASAL Remote Manager

The LASAL Remote Manager (LRM) is used for remote maintenance of machines. In addition to a tabular overview of user-defined machines (controls), this tool offers a remote view of the visualization. For data transfer between PC and control, a control explorer is available. Additional features: application start and stop, CPU reboot, software updates (application and operating system), setting and reading application data, etc. The connection between the PC (LASAL Remote Manager) and the control can be established over RS232, CAN bus, Ethernet and modem.

### LRMView

LRMView is an add-on software for the control system and provides the ability to display or control your on-site visualization project with a standard web browser (e.g. Internet Explorer) without having to install special software on the PC. Of course, the access to the system is protected with a user name and a password. LRMView is a modern Java applet with so-called PUSH technology that can be used with the control in conjunction on a web server.

### LRS API

With the LRS API (Application Programming Interface), any type of remote maintenance or visualization tool in the SIGMATEK control environment can be accessed. The API is part of a Windows DLL (Dynamic Link Library). All programming interfaces such as RS232, CAN bus, Ethernet and modem are supported.
**OPC Server**

For fast and simple data exchange with Windows applications, an OPC server with a standard interface is available, which supports read and write access to the respective data server. The user has the ability to integrate process data, for example, to a visualization, a guidance system or in Excel and evaluate them individually without worrying about proprietary data formats and protocols.

**Web Server**

The Web server in the control is a server that provides information over the Hypertext Transfer Protocol (HTTP). The websites created by the user must be written in HTML and stored in a user-defined directory in the control. Using a browser (e.g. Internet Explorer), the web site can be viewed. With this feature, the control application data can also be accessed completely independently of the active on-site visualization for remote maintenance purposes. Naturally, the access is password protected.
Object-Oriented Programming
LASAL Highlights: Compact

- **An Integrated Design Environment**
  
  with simple and comfortable operation for solving automation tasks: project design, programming, visualization, motion control, communication and (remote) maintenance in one system. Significant engineering savings are the result.

- **Innovative and Efficient**
  
  due to object-oriented programming with graphic display, client/server technology, inheritance and aggregation. These features allow the user to achieve simple testability, high modularity and reusability.

- **Clarity**
  
  thanks to object-oriented programming and graphic display: understandable and simple overview of the program interconnections, simple traceability and documentation.

- **Language Talent**
  
  applications can be created using structured text, instruction list or ladder diagram - all according to IEC 61131-3 standard -, interpreter and in ANSI-C.

- **Great Savings Potential**
  
  - only one design environment needed for all project phases
  - reduced programming time
  - lower maintenance due to the modularity provided by object orientation
  - one contact partner for the automation system (hard- and software)

- **Simple Reuse of Programs**
  
  with user-definable libraries and comprehensive SIGMATEK libraries, greater flexibility is achieved and time-to-market cycles are reduced.
■ **Flexible Implementation and Expansion**

supports all target platforms without porting, simple change from small and mid-size systems to large installations.

■ **Common Variable Pool**

CLASS defines which variables can be visualized in SCREEN, so double definition is avoided.

■ **Multi-User Project Structure Capability**

modules or program components can be developed separately, they can be put together in an integrated, complete project. Because only ASCII source files are used, a source code management program can be implemented.

■ **Automatic Project Generation and Modification**

Scripting: with the standardized script language, Python, scripts can be created, which can then generate complete programs for special characters and parameters of serial machines - virtually „at the push of a button“.

■ **Numerous Integrated Tools**

The efficient design and comprehensive analysis of programs are guaranteed by these tools:

■ Online debugger with all functions expected from an integrated design environment
■ Real time oscilloscope and trend display
■ Analysis of the time response for the real-time and multi-tasking operation system (PlcTraceView)
■ Project comparison
■ Boot disk manager for the program and visualization update using a USB stick

LASAL reflects your machine
LASAL Software