Reliance
Industrial SCADA/HMI system

Reliance 4

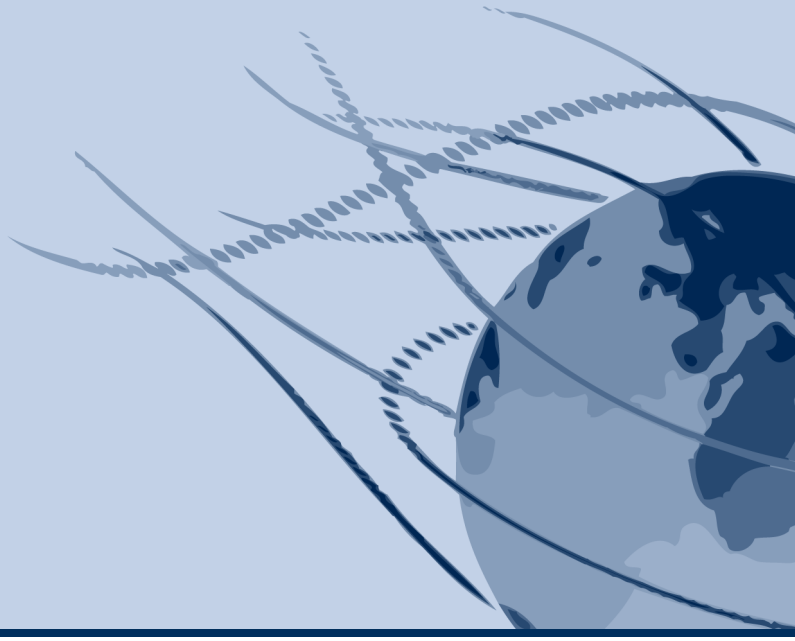
BUILDING AUTOMATION

GEOVAP

Reliance
Industrial SCADA/HMI system

Reliance 4

BUILDING AUTOMATION



GEOVAP

GEOVAP, spol. s r.o.
Cechovo nabrezi 1790
530 03 Pardubice
Czech Republic
+420 466 024 618
http://www.geovap.cz

Table of Contents

1 Introduction

For many years, the level of automation in private, family and purpose-built buildings has been on the increase. This is due in part to the fact that user demand for comfort is growing. Building automation also contributes to significant energy savings and energy consumption control. While apartment housing should be provided with security systems, purpose-built buildings, on the other hand, require flexibility so the purpose and use of the building can be changed in future.

1.1 Building Automation in Private Housing

Regarding trends in private apartment housing, it is impossible not to notice that a high number of automated functions are slowly but surely becoming a standard.

It is quite obvious that gas consumption controls are available - the control functions are integrated into heating systems. Individual room temperature control optimization is nearly always part of a new heating system installation. It is a common practice to integrate time programs or ripple controls (Ripple Control - used to control electrical energy consumption) into the temperature controls. It is taken for granted that these functions are part of the system. This brings the energy savings to the forefront.

Another example of automation in private housing is lighting control. Motion detectors are very often used in outdoor spaces to control lighting. A motion sensor, which is contained in a motion detector, looks for body heat and, together with a signal from a light sensor, the motion detector is only switched if the ambient light is dim enough. Although it is quite a simple automation function, it actually integrates event control and logical operation. Comfort is of the essence in this case.

It becomes more complicated if the entire lighting, for example, is supposed to be switched from one place. Using conventional wiring, this requirement can only be fulfilled with extensive cable distribution systems. However, new options appear when using bus systems and related communication among the components that control the lighting. For example, in the event of an intrusion in the night, the central button, located in the bedroom, can trigger an alarm. Here, the safety factor is at the forefront of importance.

To summarize, it is apparent that automation functions in private apartment housing are becoming more important mainly in the areas of:

- efficiency and energy savings

- comfort
- safety

1.2 Building Automation in Purpose Built Buildings

The term "purpose-built buildings" means buildings with a specific function. These are, for example, office buildings, retail centers, hospitals, station buildings, airport terminals, and underground garages.

Since they are properties for entrepreneurial purposes, they are often fitted with sophisticated control systems. These systems then provide for proper functioning of individual devices and, in many cases, are connected both mutually and to the main control room. Afterwards, communication is provided by data buses and networks. In addition to the optimization of energy consumption, the savings will also be reflected in a reduced number of operating personnel.

A friendly working environment for staff is also required, e.g., in the summer-time, staff should not have to endure high temperatures. This leads to the fact that new purpose-built building offices are fitted with cooling devices. Control of these systems in offices changes frequently. Today, shades and lighting are controlled by a personal computer. Both contribute to a convenient and friendly working environment.

Another requirement for the systems in purpose-built buildings results from the behavior of users. Space division requirements may change depending on the restructuring of the company. The demand for small offices may appear instead of large conference rooms. The design concept of the building and its operational-technical equipment must enable such changes. For example, deployment of light switches and their assignment to the corresponding lighting circuits do not require electrical cable connections, but the adaption is carried out by reprogramming the intelligent components. This brings the flexibility of the operating facilities to the forefront.

In summary, we can say that building automation in purpose-built buildings becomes more important mainly in the areas of:

- providing operation efficiency and energy savings
- communication via bus systems and networks
- comfort
- flexibility

1.3 Process Visualization in Building Automation

It's becoming increasingly apparent that user comfort and ease of operation stand out in the forefront of interest. Because of this, the following requirements can be mentioned:

- easy operation, monitoring and overview of all functions, multiple accessibility
- optional specific modifications by users themselves, e.g., lengthening the “on” time of staircase lighting
- compatibility with the building system engineering, Internet, etc.

Ergonomically oriented visualization projects offer the required functionality related with higher comfort and usability typically represented by built-in touch screens or computers with LCD touch monitors.

These touch devices offer many options to centrally control the automation in the building:

- all lights can be switched and dimmed - the light status is shown graphically
- with just a glance, you can get an overview of the light status on the ground floor
- in another window, you can monitor and control the lighting throughout the building
- monitor and control the HVAC systems, cooling system, etc.
- besides, the visualization project can provide functionality of the alarm/event records, archiving of the recorded data, and graphic presentation - visualization of the status of the operational-technical equipment in the building.

1.4 Summary

Above we tried to indicate the current trends and importance of building automation from the perspective of functions and areas of use including the use of visualization projects (systems) in building automation.

This documentation provides an introduction to working with the **Reliance** SCADA/HMI system and its use in visualizations of the **Tecomat Foxtrot®** (hereinafter CFox) building wiring applications. Working in the Reliance environment requires some orientation. We highly recommend that you attend our training courses organized by **GEOVAP.** Good skills in using the *FoxTool* development environment (application with the *CP-1000/CU2-01M* central unit) are also required. This environment is intended for deploying the **CFox** *system applications easily and quickly.* For using the *CP-10xx Foxtrot* central units, knowledge of the *MOSAIC* development environment is necessary. In either case, whether you are using CU2-01M or CP-10xx Foxtrot, it is recommended that you attend our training courses organized by **Teco** or **GEOVAP.**

2 Reliance - Industrial SCADA/HMI System

2.1 About Reliance

Reliance is a modern SCADA/HMI system designed to visualize and control industrial processes and for building automation. **Reliance** development is based on many years of experience in building large applications. Continuous input from customers also contributes to its improvement. As a result, a robust, reliable and easily scalable system, optimized even for large applications, is available and ready to be used.

The most appreciated qualities of **Reliance** are its clarity, intuitiveness, and user-friendly environment. We are also pleased that the systems integrators who switched to the **Reliance** system from other SCADA systems claim it is 2-3 times faster to develop an application in Reliance than in SCADA systems they had worked with before. Among the main priorities of the system are its stability, reliability, user-friendliness, and the speed of development.

Thanks to the applied industrial systems, visualizations are easily accessible both from the local network and the Internet or mobile devices (e.g., PDA). The whole system can be protected by access rights. Using the **Reliance** system, authorized users can then set operation parameters and directly control the selected equipment.

Thousands of **Reliance** installations have been successfully deployed to control and monitor system processes in a wide range of areas, such as:

- intelligent buildings (building automation)
- gas industry
- chemical production
- HVAC systems in building complexes
- operator's rooms for control of city heating
- coal-fired and hydro-electric power stations
- sewage treatment plants
- drinking water treatment plants
- glass factories
- mills
- malt houses and breweries
- meat processing plants and food production

- traffic control systems
- visualizations of various production lines
- and many others...

You will find more examples of Reliance installations in the Success Stories section.

2.2 Basic Program Functions

The basic functions of the **Reliance** system are:

- developing a visualization project
- export of the visualization project for remote users
- loading and operation of the visualization project
- receiving (reading) data from devices via communication drivers
- graphic display (visualization) of the operational data
- data exchange (including alarms/events) between runtime modules via the *TCP/IP* protocol
- archiving of data via various database machines *(Paradox, DBase, SQL)*
- generating and processing of alarms/events
- running scripts
- sending email messages
- sending and receiving SMS messages
- recipe processing
- displaying alarms/events (both current and historical)
- creating print outputs (tables, trends, custom reports, alarm/event print reports)
- user management, log-on and log-off
- user's system control (**Reliance Control** and **Reliance Control Server** only)
- system part control limitation according to the access rights of the user
- program language and project switching during runtime
- multimedia support
- access to the visualization via the *Internet/Intranet* **(Web Client)**

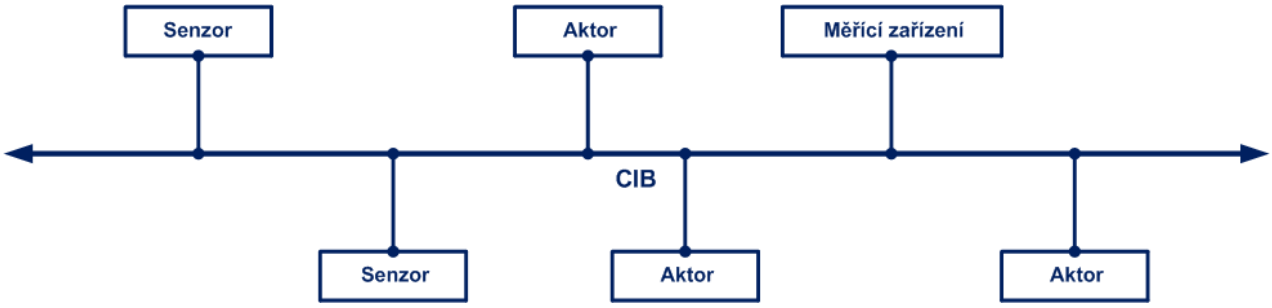
- Postmort - a function designed for replaying the operation of a monitored process and operators' commands
- support for *ActiveX* components
- support for multi-monitor systems
- user log-on and verification using a fingerprint scanner
- support for mobile devices **(Mobile Client)**

You can find more information about **Reliance** in the Products section.

3 Building Automation - CFox

3.1 About CFox

CFox building wiring uses a dual-conductor installation bus. In building system engineering, it's a system which is used for network connection of devices (sensors, actuators, control and monitor devices and measuring devices - picture No.1). The implementation of **CFox** is adapted to electrical installation. This makes the building's functions and automated processes possible.



Picture No. 1 - Information network connection of devices

The data designed for mutual communication is loaded into the data message and transferred via the installation bus according to the established rules. The whole concept is based on the principle of the peripheral unit with central control. These units are connected to the system very simply - by two conductors with an arbitrary topology. The concept is named **CIB (Common Installation Bus)** and is fully integrated with the *CP-1000 (CU2-01M)* and *CP-10xx Foxtrot®* central modules.

The installation bus protocol is optimized so that the response time from the input via the central unit to the output is guaranteed to be no more than *150 ms* when the bus is full. A time period of less than *300 ms* is not, by humans, perceived as a delay. Therefore, it is a solution suitable even for lighting control. The required speed of response is achieved by the *19.2 kbit/s transfer speed. CIB* can be used not only for buildings and wiring, but also for measuring and control systems with a sufficient response up to *150 ms.*

The **CFox** system is used for implementation of lighting operation and control, shades settings, heating controls, air-conditioning, etc. It is installed in new residential houses and purpose-built buildings as well as in older buildings under reconstruction or renovation.

3.2 Use of CFox

More comfort, more safety and higher efficiency - these are the main factors of the widening use of electronics in both apartment housing and purpose-built buildings.

To implement building automation and control, in particular for:

- input and output functions
- data processing functions
- control functions
- operating functions

a number of the following components are available in modern buildings:

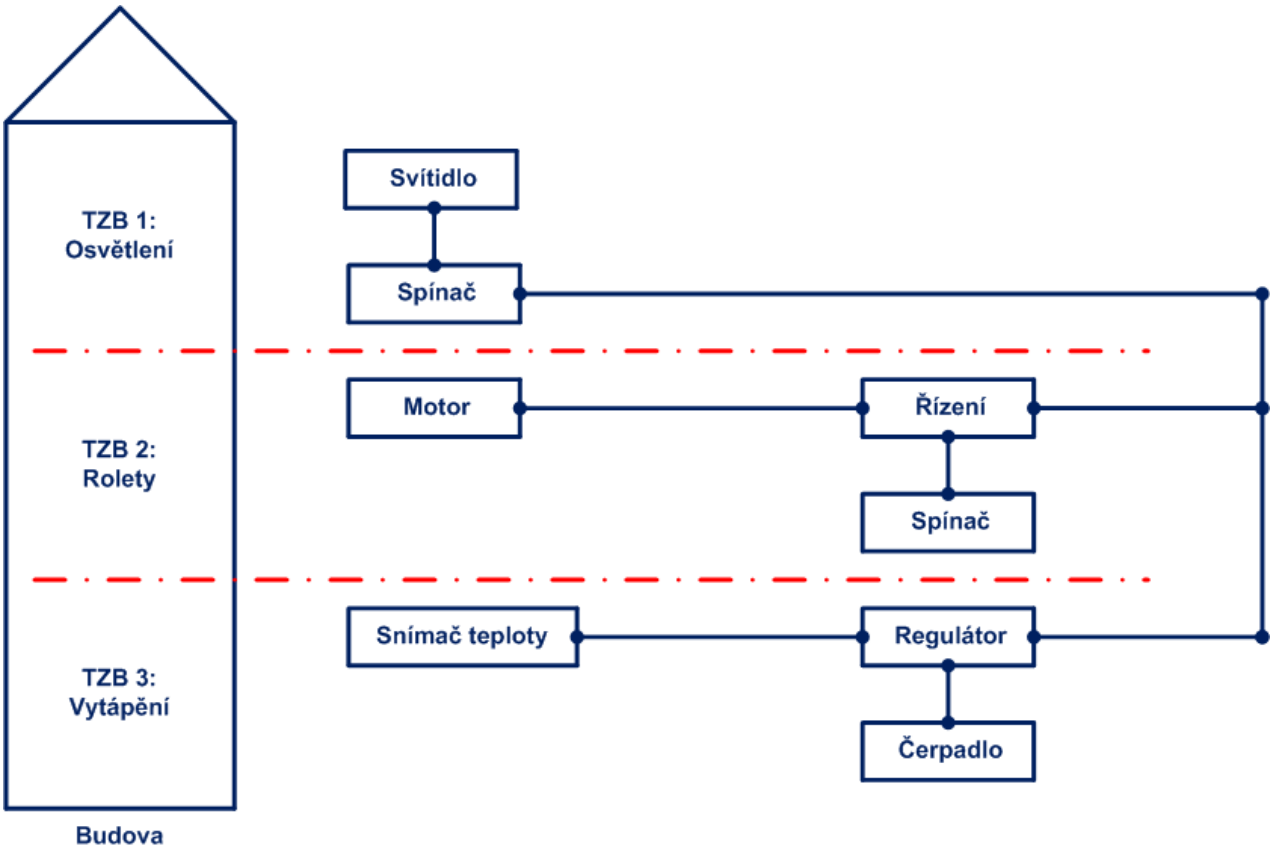
- sensors (e.g., light, motion, and push-button sensors)
- actuators (switch, dim, and valve control actuators)
- monitor and control devices
- operating, measuring and tracking devices (devices for visualization, e.g., control desks)

Mutual communication, i.e., information exchange, is required for the compatibility of these devices and the building functions.

3.2.1 Conventional Apartment and Building Wiring

Technical equipment (lighting, air-conditioning, ventilation, etc.) for conventional wiring in apartments and buildings is designed and supplied by a specialized professional company.

The sensors and actuators are locally connected to the monitor, control and visualization devices - see picture No. 2. However, this leads to significant demands on designing, cable installation and results in difficulties in detecting errors after it is brought on-line and during maintenance.

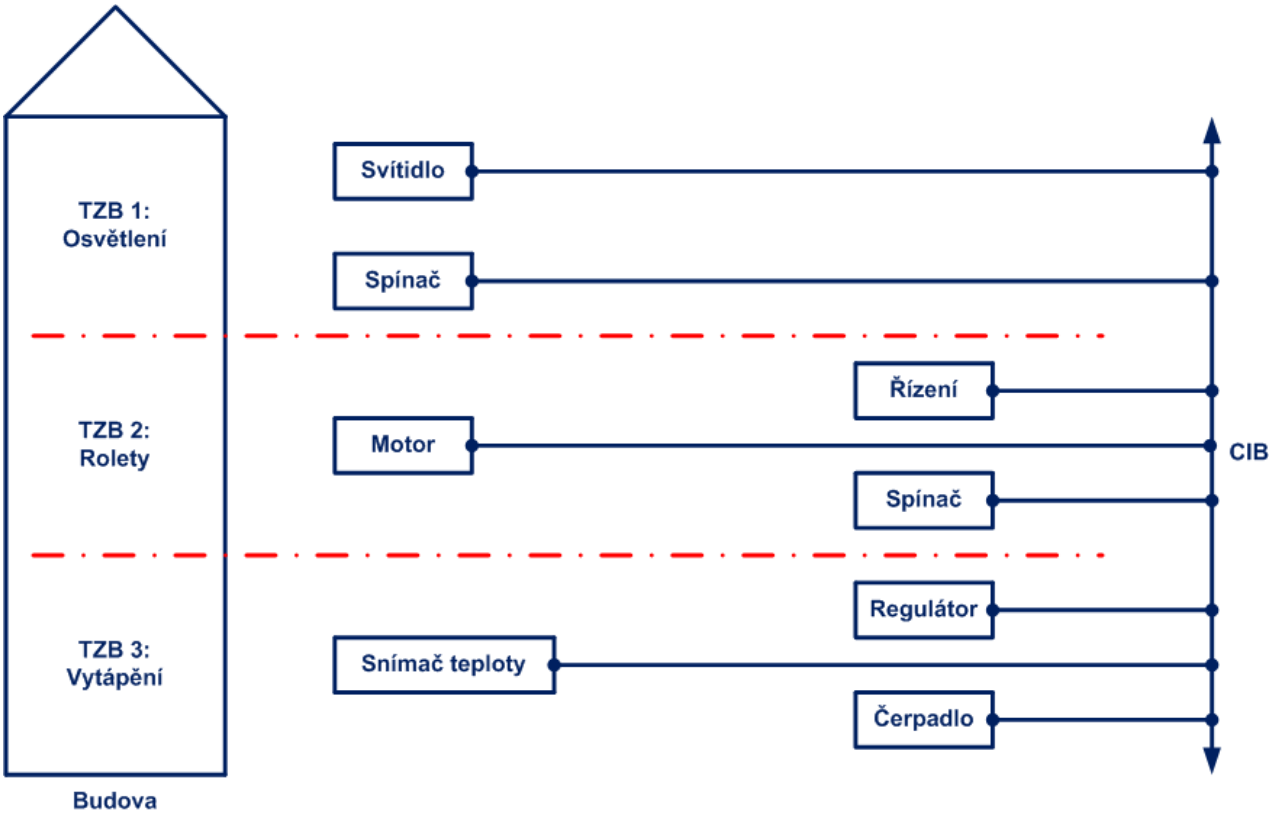


Picture No. 2 - Conventional wiring

Connecting the autonomous technical devices of the buildings, e.g., for common operation purposes, is either impossible or at least very demanding. With the growing range of functionalities and comfort requirements, the conventional technical devices are becoming more demanding, chaotic and expensive. The solution of this problem is building system engineering.

3.2.2 Building System Engineering with CFox

System installations with integrated TZB (building interior installations) make fewer demands on cable distribution systems - picture No. 3.



Picture No. 3 - System installation with integrated TZB

CFox was developed as a system designed to cover all important installations that can be used for TZB. TZB can be designed and installed as a system, which unifies the professions that implement them.

This simplifies the design and implementation of building functions and allows for greater functionality, flexibility and comfort in buildings without additional costs.

Programming (development) environments define what a particular device should do. Installations are then flexible and can adapt to new requirements. No matter where this type of installation is used (in family houses, apartment houses or purpose-built buildings), it enables automatic control of lighting, heating, ventilation, and alarm systems. The main benefits of these applications are their efficiency and improved property utilization.

The following overview shows several examples of the benefits.

In regard to comfort:

- all lights in a building can be switched on and off by pressing one button
- by pressing a button, a predefined light scene can be activated
- all shades on a story can be rolled up or down by pressing one button
- comprehensive and comfortable visualization

In regard to safety:

- after it gets dark, driveways or entrances are automatically lit when a car (or a person) enters the grounds
- system installations monitor windows - central reports of open/closed windows are available
- all lights in a building can be switched on by pressing an alarm button
- when a building (or parts of a building) is empty, a presence simulation can be performed by setting particular sequences

In regard to efficiency:

- if a window is open, a heating control turns off
- shades are set during sunlight hours so that it affects the room temperature

4 Cornerstones of CFox

To begin with, it is necessary to emphasize that it was not our intention to present this documentation as a detailed treatise on the **CFox** system and its tools. This is the subject of professional training of the **CFox** system which is organized either by **Teco** or **GEOVAP**. Below is the list of the system's basic premises.

4.1 Common Installation Bus (CIB)

CIB is characterized by "free topology". This means any branching can be applied, it is not recommended to only use a ring topology. The work of systems integrators during implementation involves installing a dual-conductor cable. The polarity must match when interconnecting actuators and sensors. This limits the number of supply conductors since the installation bus not only transmits data but also provides supply voltage for connected sensors and actuators.

Common Installation Bus (CIB):

- one branch of the bus has a considerable range of up to *550 m*
- modular and configurable system
- communication, i.e., data transmission is performed in accordance with specific rules *(master/slave model)*
- transferred information flows as a compact data message (the message size does not exceed *256 byte/message*)
- two branches of *CIB* can be connected to a *CP-1000 (CU2-01M)* central unit and only one *CIB* branch can be connected to a *CP-10xx Foxtrot®* central unit - in total, *32 units* can be connected to each branch

If more branches (or units) are required, the system can be simply expanded using external master modules:

- each of these modules allow for connecting an additional two branches of *CIB*
- the maximum distance between an installed external master module and a central unit is *300 m*

This enables increasing the number of connected actuators and sensors as well as covering even large facilities (applications).

The communication system operates with supply voltage in two levels:

- 24 V DC
- 27.2 V DC

Higher supply voltage allows standby storage batteries to be connected and continuously recharged via the *CIB* separation module. During an outage (*230 V AC* network), these storage batteries enable central units and the units connected to *CIB* branches to be in operation.

A hardware address is decisive for the determination of sensors' and actuators' addresses in *CIB*. Hardware address:

- is assigned by the manufacturer and marked on each unit
- is the unique physical address of each unit - a 4-digit code (16 bit number) in hexadecimal format
- can be "read" electronically in a central unit (programmers can read hardware addresses through the development environment and concurrently write them in a table of installation components, later, they may simply refer to unit names by project).

The CIB communication system:

- monitors all *CIB* branches, i.e., it is notified of disconnections or failures of any unit
- these states can be read via the development environment
- has a function that allows you to update firmware of units connected to *CIB* branches (in the programming environment, this can be done by simply pressing a relevant button - this function can also be performed remotely)

4.2 CP-1000 Basic Module

Currently, central control and visualization of automation projects as well as remote management and control are expected. Supervision and control via SMS, as well as using the Internet are also put into practice. The executive central unit concept meets these

requirements. This unit supervises the *CIB* branches and can transmit data for the visualization.

CP-1000 Basic Module:

- is configurable via the *FoxTool* development environment
- the maximum number of units in all *CIB* branches is *320*
- two branches of *CIB* can be connected to the *CP-1000 (CU2-01M)* unit
- *32 units* can be connected to each branch, *64 units* in total
- the system can be extended with external master modules and the *TCL2* bus - each of these modules allow for connecting an additional two branches of *CIB*
- has the Ethernet 100 Mbps interface available - this can be used for connection to a PC or LAN
- the interface is intended for configuring via the *FoxTool* development environment
- the system can be viewed over the Internet via this port using a standard Web browser, the basic module is provided with an integrated Web server
- serial channel CH1 for connecting a GSM communicator is available
- 4x DI (universal digital inputs), 2x DI (digital inputs for connecting the Ripple Control signal and monitor 230V AC), 2x RO (relay outputs)

Picture No. 4 - CP-1000 Basic Module

4.3 FoxTool Development Environment

FoxTool Development Environment*:*

- is intended for deploying the **CFox, RFox** system application easily and quickly
- in connection with the *CP-1000 (CU2-01M)* central unit, it is designed for programming beginners, i.e., for persons unfamiliar with the PLC (Programmable Logic Controller) terminology and standards
- offers accessible configurable functions which are most commonly used in building automation (lighting, shades, heating, air-conditioning, building surveillance, alarm reports and basic communication via PC and cell phone)

Basic Program Functions:

- it operates on a standard PC with OS Windows XP/Vista/7
- provided with options to install in a selected language and to switch the language during runtime
- Project Manager function (set of information related to each central unit),
- central unit simulator (SoftPLC) may also be installed
- operates in two modes (Manager and Designer)
- Device Manager (configuration of up to 10 *CIB* branches)
- Time Program Manager
- Time Event Manager
- SMS Manager
- Action Manager
- internal data for communication with master systems or for visualization (**Reliance** SCADA/ HMI system) is accessible via *CP-1000 (CU2-01M)* and *FoxTool.*

The configuration itself runs as a dialog, the operator:

- decides what initializes the input
- decides how and on which output the system responds
- can make necessary parameter settings

- additional actions or commands can be added, e.g., sending an alarm, SMS or setting one of the 32 internal flags (bits), 32 counters and 32 timers
- logical functions can be applied

4.4 CP-10xx Foxtrot Basic Modules

The *Tecomat Foxtrot®* basic modules (PLCs) are also available for the **CFox** intelligent wiring system applications. These central units represent high-performance PLCs which supervise operations in the *CIB* branches and can transmit data for a visualization.

The basic module - the *Tecomat Foxtrot®* central unit - is a separate control system fitted with a power unit, communication channels, inputs and outputs. The *MOSAIC* development environment is used for programming this system.

All the basic modules of the *Tecomat Foxtrot®* system consist of the following parts:

- a central unit with a CPU, two serial channels, an Ethernet 100 Mbps interface and a *TCL2* bus for communication with peripheral modules
- an M12-01 internal master module providing communication in *CIB*
- the third part is peripheral, it varies depending on type and consists of inputs and outputs: CP-1004, CP-1005, CP-1014, and CP-1015,
- additional display and buttons are contained in the basic modules CP-1014 and CP-1015
- the maximum number of units in all *CIB* branches is *288*
- one *CIB* branch can be connected to the basic module of the *Tecomat Foxtrot®* system, which is *32 units* in total
- if more branches are required, the system can be expanded using external master modules and the *TCL2* bus - each of these modules allow for connecting an additional two branches of *CIB*
- if the number of units is not sufficient, you can create a network using more PLCs within the Ethernet LAN or in the RS485 bus
- has the Ethernet 100 Mbps interface available - this can be used for connection to a PC or LAN
- the interface is intended for programming via the *MOSAIC* development environment

- using this interface, data for a convenient visualization on your PC, provided by the **Reliance** SCADA/HMI system, is available
- the system can be viewed over the Internet via this interface using a standard Web browser, because the central unit is provided with an integrated Web server

Picture No. 5 - CP-1004 central unit with peripheral I/O

A spectrum of peripheral modules with wireless communication - *RFox* with I/O modules - can be added to the system. This makes *Tecomat Foxtrot®* even more flexible:

- you can combine conventional PLC peripheral modules
- *CIB* components
- and also the *RFox* wireless installation

4.5 Mosaic Development Environment

Basic features:

- development environment for creating and debugging programs for the *Tecomat®* programmable systems

- the environment is developed in accordance with the internationally accepted IEC EN-611-31-3 standard (this standard defines the program structure and programming languages) for PLC
- all-in-one package
- Lite test version available
- full version protected by a HW key - license portability
- regular update
- language versions (CZ, EN, DE, and RU)
- designed for Windows XP/Vista/7, 32 bit, and 64 bit

Programming:

- in accordance with the IEC EN-611-31-3 standard
- graphical languages - Function Block Diagram (FBD), Ladder Diagram (LD)
- textual languages - Structured Text (ST), Instruction List (IL)
- option to combine various languages
- both standard and user-defined data types including structures and fields
- on-line program change (programming during runtime)
- both standard and user function/function block libraries are available

MOSAIC software tools:

- PLC simulator - SimPLC (integrated)
- control panel simulator - PanelSim
- control panel tool - PanelMaker
- graphical control panel tool - GPMaker
- tool for creating Web pages for a Web server - WebMaker
- tool for defining and monitoring control loops - PIDMaker
- tool for monitoring tag trends - GraphMaker
- creating library function blocks

5 Reliance and Tecomat Foxtrot Central Unit Data

5.1 Basic Premises

5.1.1 Data Exchange

Communication between the *CP-10xx Foxtrot®* central units (hereinafter PLC) and **Reliance** SCADA/HMI system runs via the *EPSNET* communication protocol. A communication driver for the Tecomat® PLC (native communication driver) is part of **Reliance** SCADA/HMI system.

For a detailed description of the EPSNET protocol, we recommend the following documentation: Programmable 32 bit Tecomat PLC Serial Communication, 16th edition, March 2009

EPSNET is communication based on sending UDP packets - the PLC responds to a query of the master system or PC. The PLC itself does not send any data without having such a function programmed.

The PLC memory is divided into the following three basic areas:

- inputs - X - input images from input units
- outputs - Y - output images from output units
- user registers - R - operational data

Each input, output or operational tag has its unique address. This direct linking is used in communication. The *FoxTool* and *MOSAIC* development environments generate the so-called **pub file (file xxx.pub)* which contains the linking description.

The only difference between the two tools is that the addresses can change when saving the configuration with the *FoxTool* development environment while the *MOSAIC* environment allows keeping the addresses unchanged.

Example of an input linking:

- room_7_1_TERM X F 76 REAL PUB_INOUT:
- room_7_1_TERM - input named in FoxTool or MOSAIC

- X - data area
- F - intended for backward compatibility with older programs
- 76 - address
- REAL - tag data format
- PUB_INOUT - intended for further programs
- this line defines that the tag room_7_1_TERM is located at X76 and is in REAL format

5.1.2 Generating a PUB File

In this documentation, it is not our intention to describe the *FoxTool* and *MOSAIC* development environments.

For the description of how to generate a PUB file, see the following recommended documentation: Starting with the MOSAIC Environment, 7th edition, February 2008.

It is also recommended that you visit our training courses organized by **Teco** or **GEOVAP**.

For generating a PUB file, the creation of a PLC control program (for the CFox applications) by the FoxTool or MOSAIC development environment (according to the type of PLC) is necessary.

5.1.3 Importing the PUB File into a Project in Reliance Design

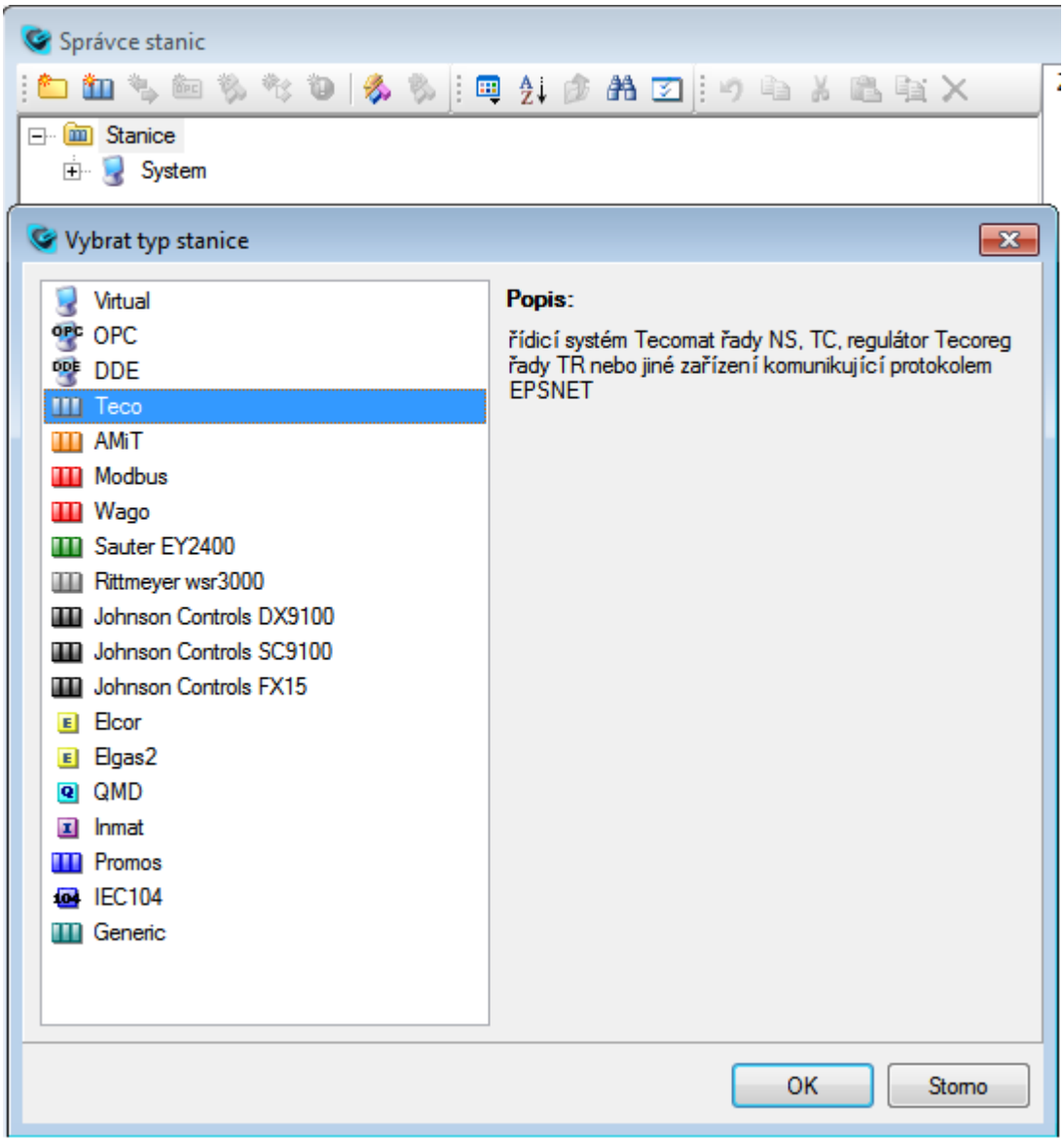
The following assumes that a PLC is programmed by the above mentioned development tools and the generated PUB file (file with a .PUB extension), which contains a list of public PLC program tags, is available for further work in the **Reliance Design** environment.

If you are not acquainted with the Reliance Design development environment, we highly recommend that you read through the documentation: Reliance 4 - First Steps, © 2011, GEOVAP, spol. s r.o.

It is recommended that communication with a PLC is set before we start creating visualization screens.

1. Open the Device Manager by clicking on *Managers -> Device Manager* in the main menu

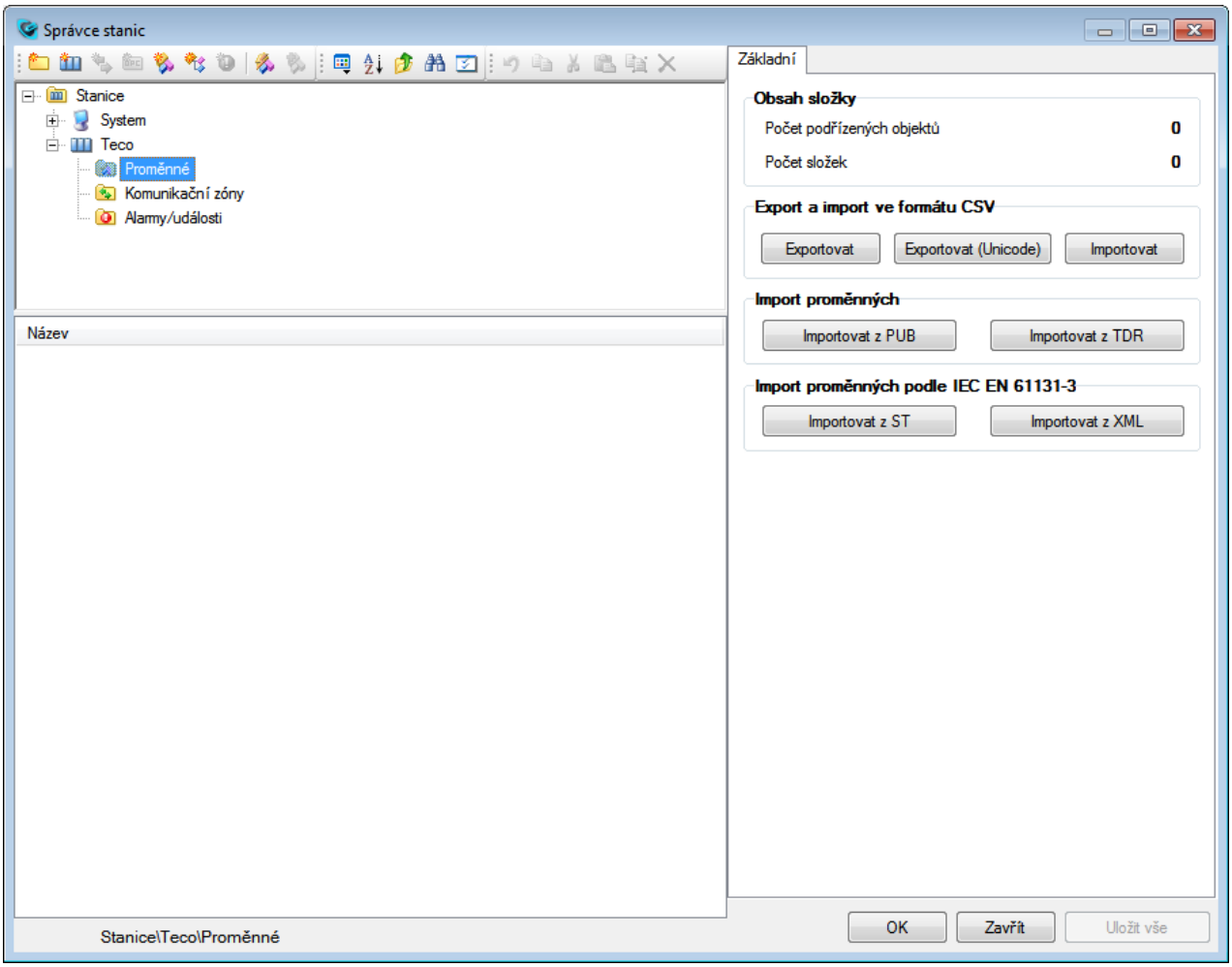
2. Tecomat® PLCs are used for **CFox**, therefore, select the *Teco* device and confirm - picture No. 6



Picture No. 6 - Device Manager

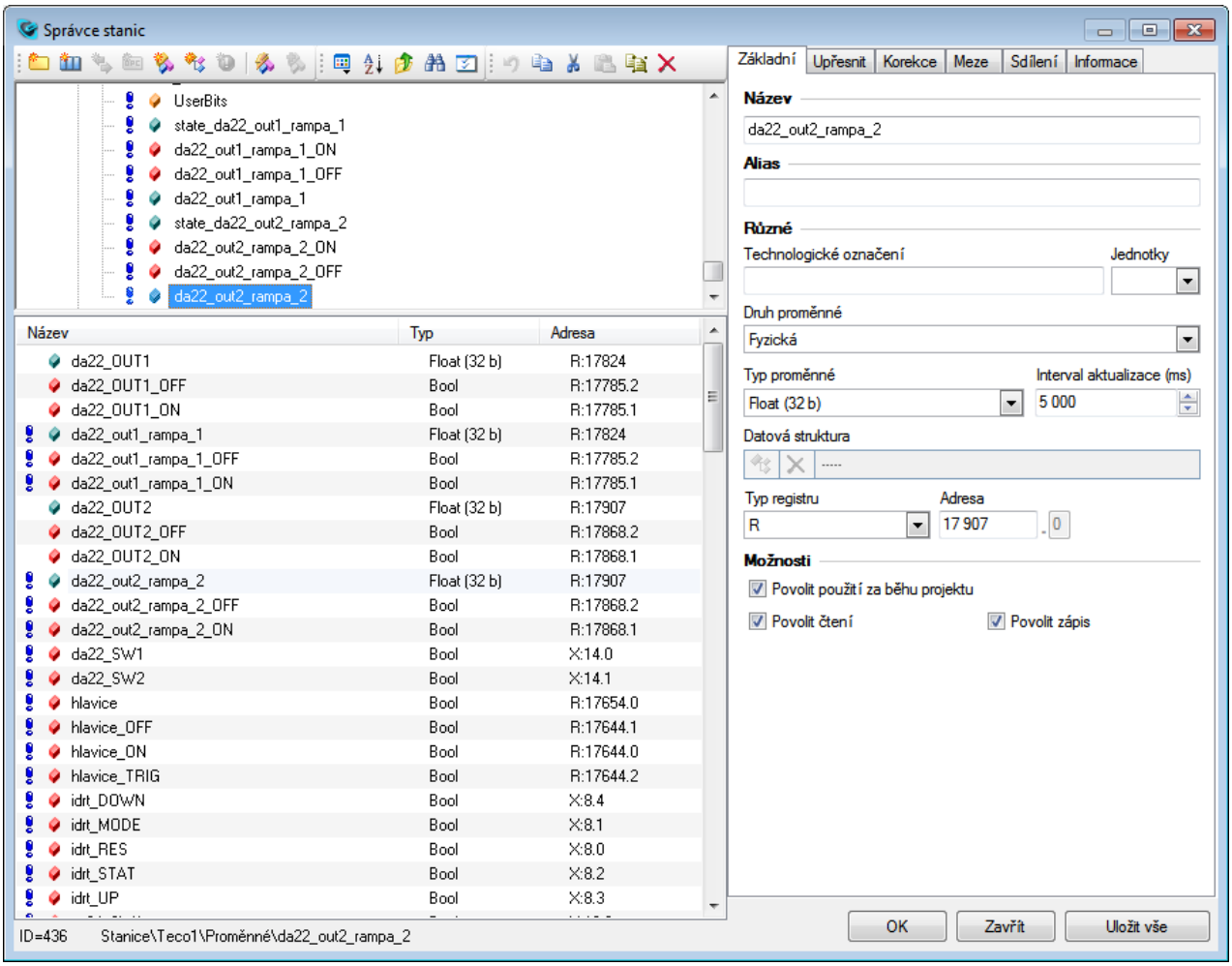
3. Select *Tags* located in the Teco object

4. In the *Basic* tab, which is located in the right side of the window, import tags from a PUB file by clicking on the *Import from PUB* command - picture No. 7



Picture No. 7 - Device Manager

5. The *Import from PUB* command is designed for selecting a PUB file and importing tags
6. After the import is finished, new tags appear in the *Tags* file - picture No. 8



Picture No. 8 - Device Manager - List of imported tags

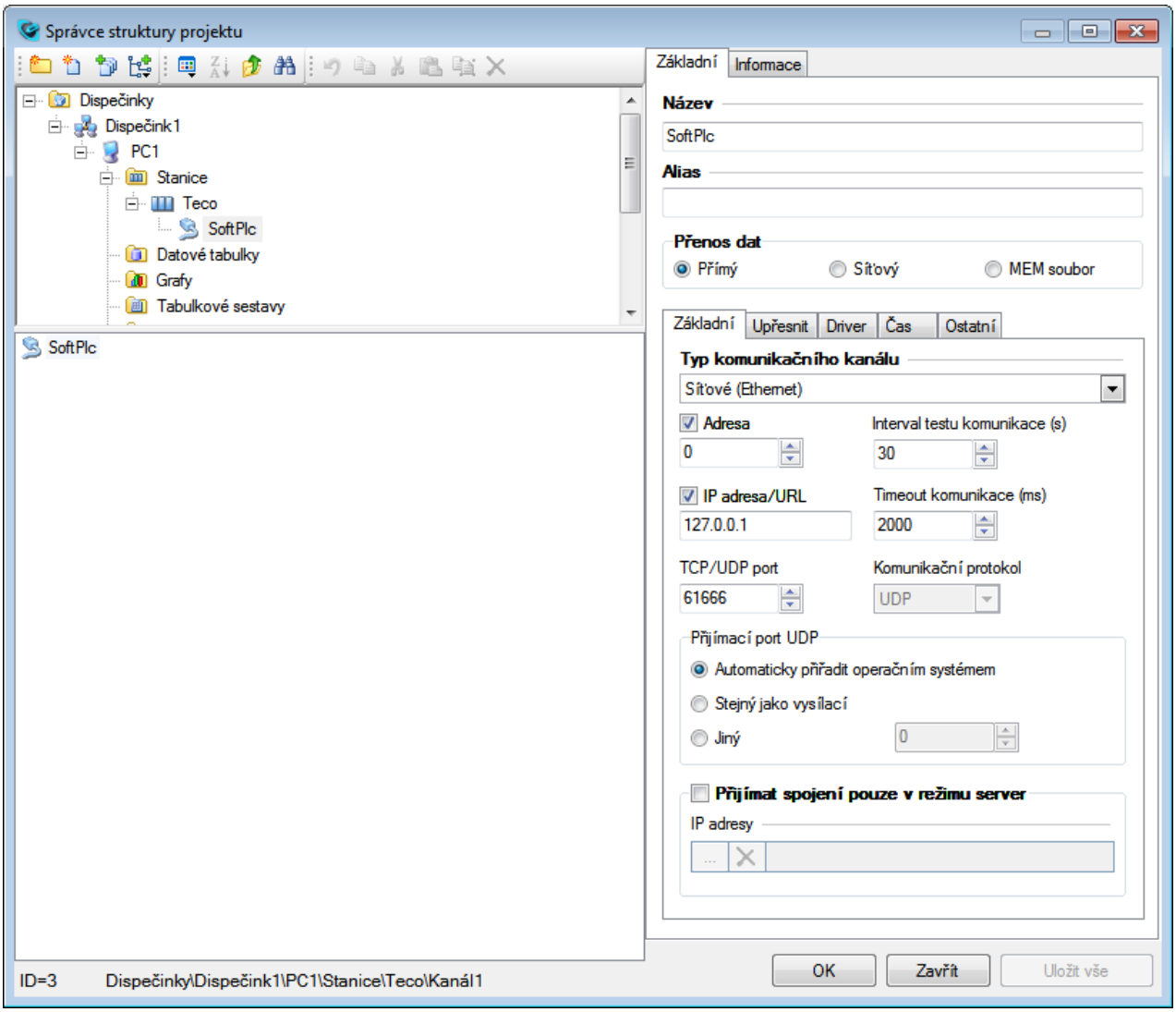
7. At any time in the process, it is possible to revert and make any necessary changes in the *Tags* file.
8. A number of properties can be assigned to each tag - picture No. 8 - for a detailed description, see the following documentation: *Reliance 4 - Development Environment, © 2011, GEOVAP, spol. s r.o.*
9. Now, the defined tags are ready to be used in our project
10. When the modifications are carried out, users are asked to confirm that the new Teco PLC device should be connected to a computer.

11. The Project Structure Manager opens automatically to connect the device to a computer and the ready changes only need to be confirmed (connection can of course be carried out later).

5.2 Creating a Visualization Project

If you want to know in detail how to create a visualization project, read the following documentation: *Reliance 4 - First Steps, © 2011, GEOVAP, spol. s r.o., Reliance 4 - Development Environment, © 2011, GEOVAP, spol. s r.o.* Below is the description of how to create a visualization project with the Teco PLC (CFox).

In the Project Structure Manager (the *Managers -> Project Structure Manager* command in the main menu), it is possible to configure settings for communication with the PLC Tecomat® device. These settings can be configured after selecting a *Channel*-type object (there is a channel named *SoftPlc* in picture No. 9).



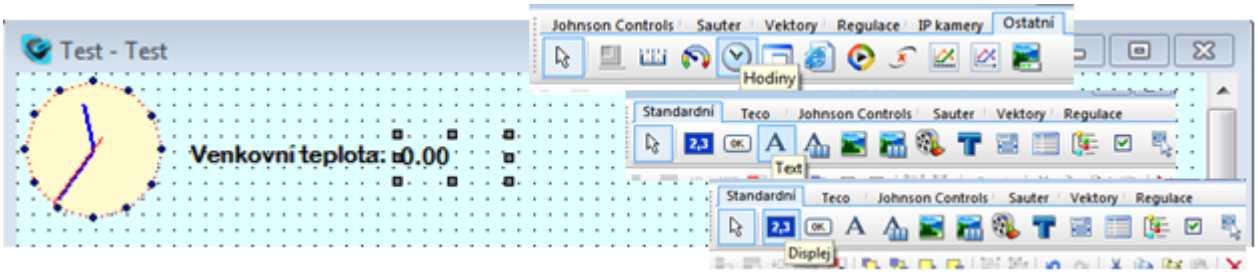
Picture No. 9 - Project Structure Manager

After configuring the communication settings in the *Project Structure Manager*, you can start creating the user interface. The basic components of the user interface are visualization windows, into which graphic objects (components) can be pasted. While the first visualization window can be created when building a project, another one can be set up later in the *Window Manager*. Components are pasted into the windows by selecting them in the *Component Palette* (picture No. 10). Each component's icons suggest the purpose of a particular component. To paste a component into a window, simply click on the component in the palette to select it, then click on the window area to place it. You may configure the size, location, and graphic settings of each component at your discretion.

Picture No. 10 - Component Palette

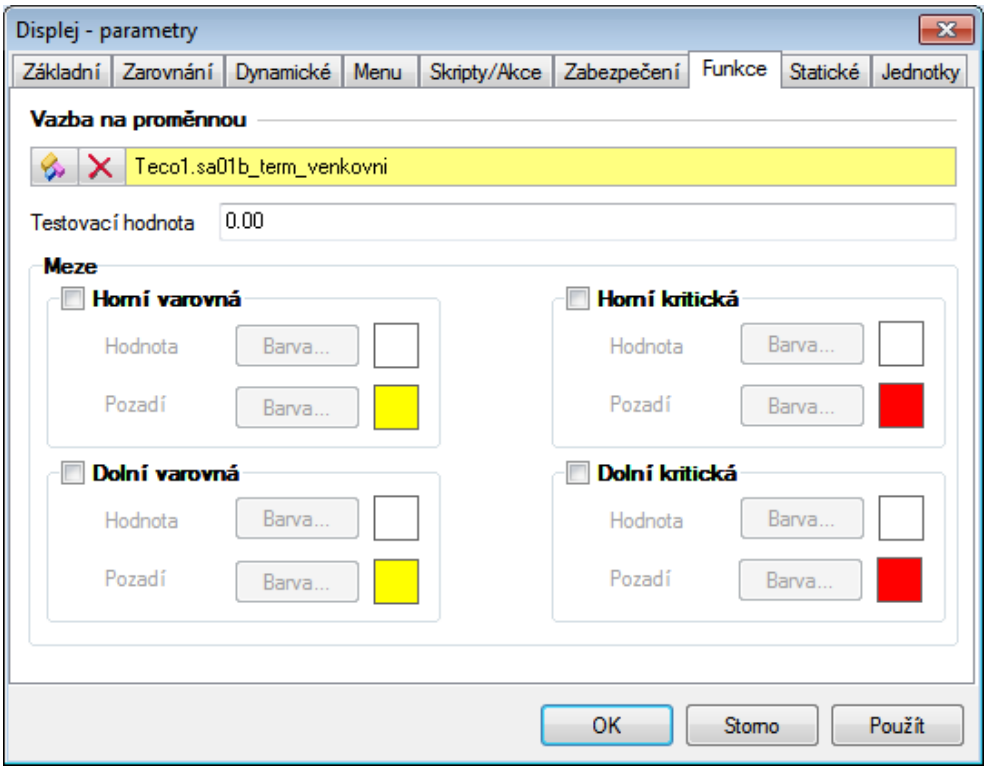
Enter time information into a visualization window using the *Clock* component - picture No. 11. This data is very useful for visualization users (it is quite obvious that date/time information can also be entered using a simple script, but it is not intended to be shown in this example).

By pasting the *Text* component, enter static text "Outside temperature" and, using the *Display* component, enter the value of outside temperature (picture No. 11).



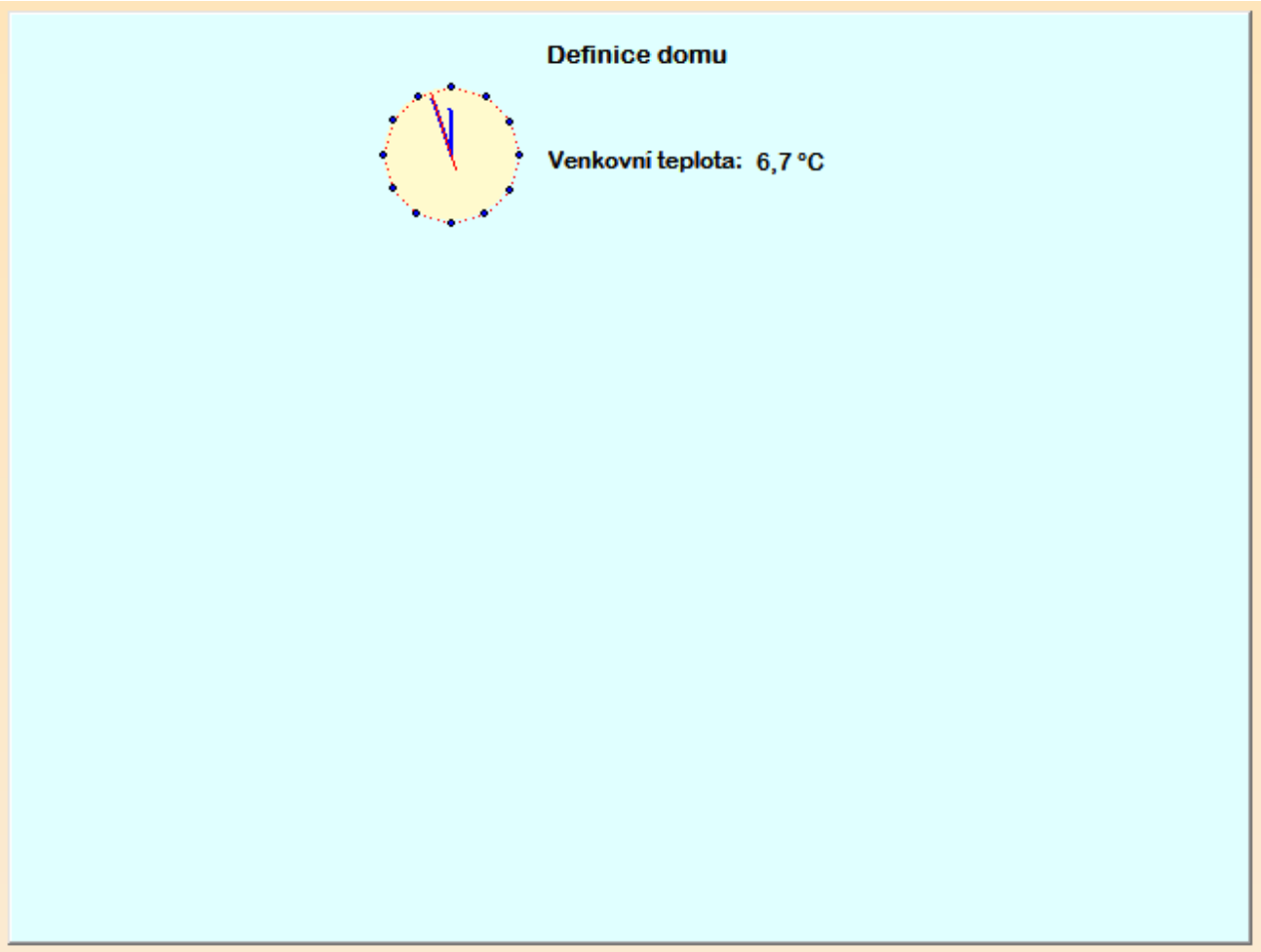
Picture No. 11 - Text and Display components

To use the Display components, it is necessary to define the link to a tag (picture No. 12), which contains the value of outside temperature.



Picture No. 12 - Display component - Properties - Functions (Link to tag)

The application can now be launched in the *runtime module*. You can also check whether the entered time and the assigned tag display the correct data (picture No. 13). To start the runtime module, use the *Project -> Start* command in the Reliance Design main menu (F9).

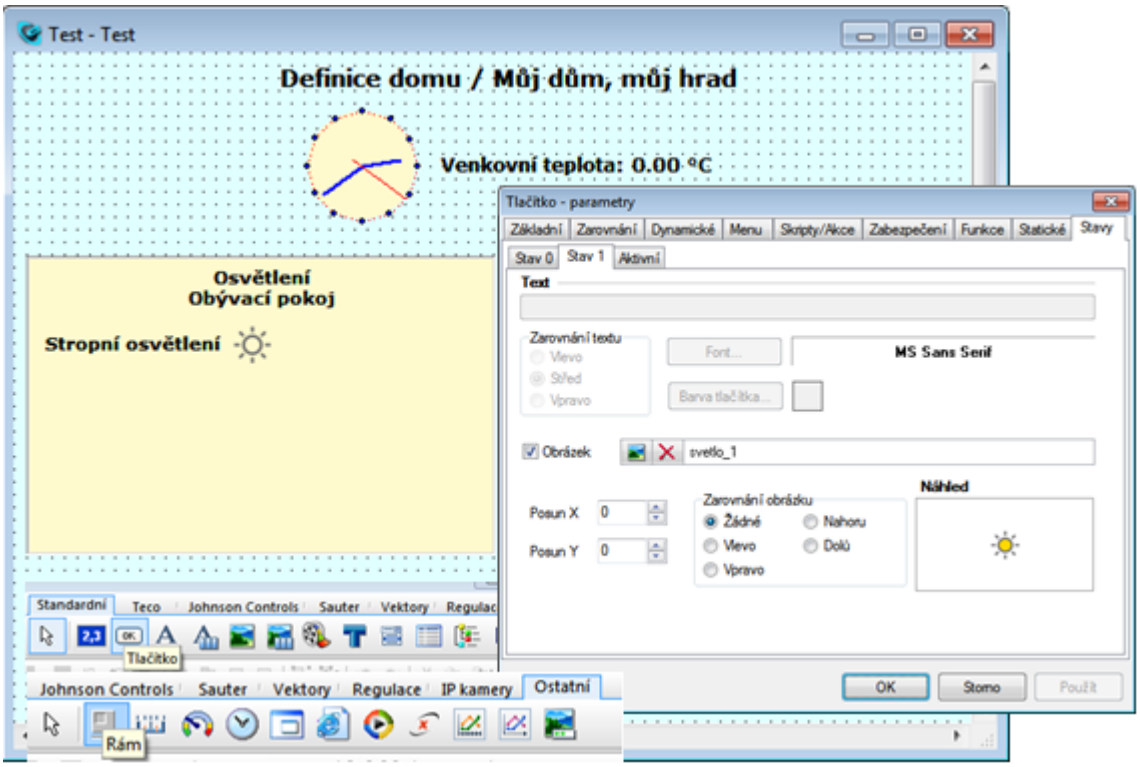


Picture No. 13 - Visualization of time and outside temperature

5.2.1 Visualization and Control of Lighting

To display and control lighting (lighting circuits), additional components must be added to the visualization window. Next, pictures either from the Reliance 4 graphical library or that of the user are imported into the project via the *Picture Manager*. *You can find more information about the Picture Manager in the following documentation: Reliance 4 - Development Environment, © 2011, GEOVAP, spol. s r.o.*

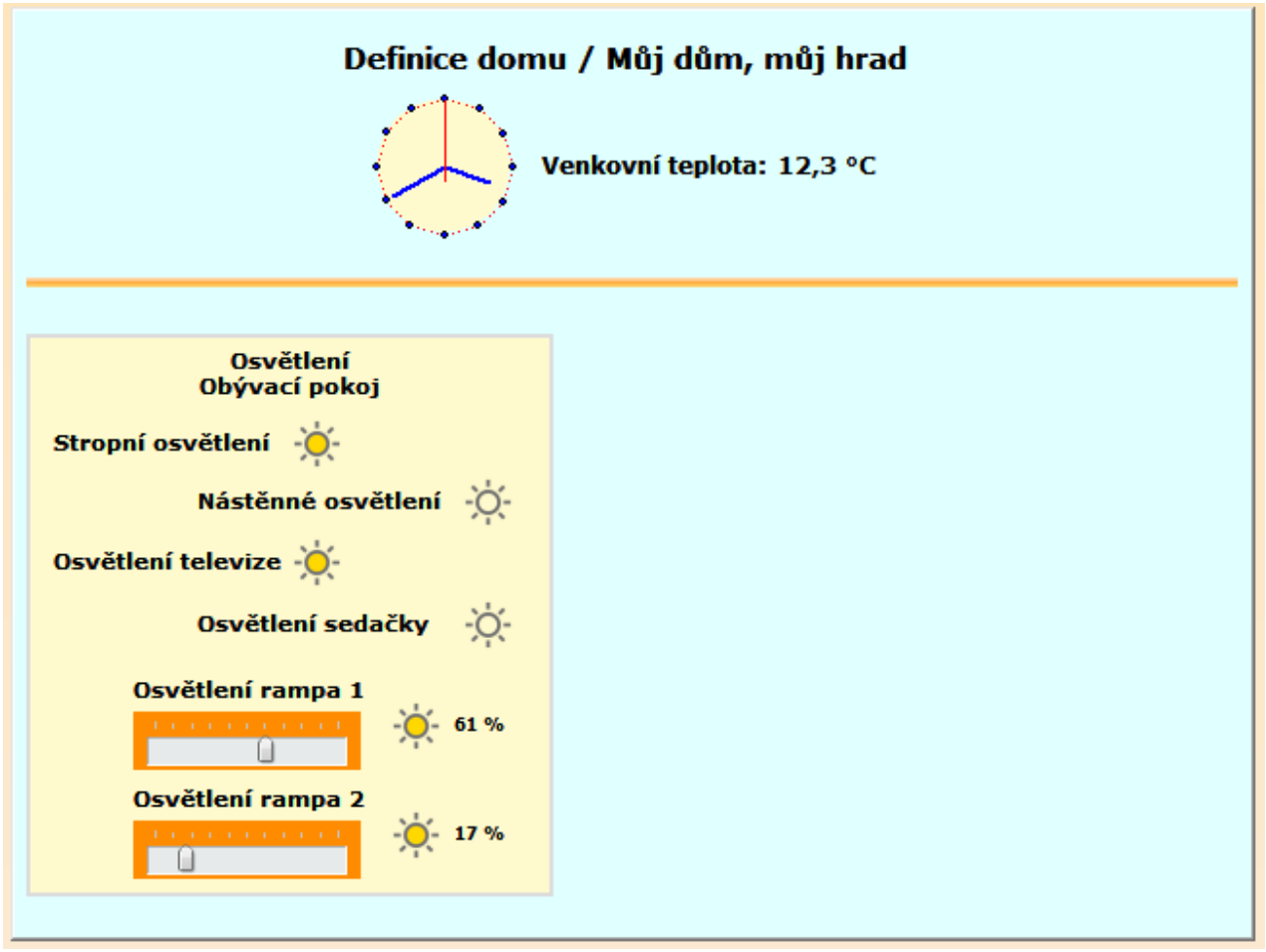
For a proper overview, individual parts should be graphically divided with the *Bevel*component (picture No. 14). Furthermore, add a *Text* component on top of the Bevel component with the corresponding description (see picture No. 11).



Picture No. 14 - Visualization and control of lighting

To display and control a particular lighting circuit, add a *Button* component, adapt it graphically and define the link to a tag (picture No. 14). In the same way, add the components for all lighting circuits which are defined in the FoxTool development environment project. Using the *Scrollbar* component, you can carry out settings while the *Display* component helps you to visualize the level of lighting. The added and modified components can be easily duplicated using the *Duplicate* command (Ctrl+D) in the window's popup menu and their properties (e.g., link to a tag) can then be changed.

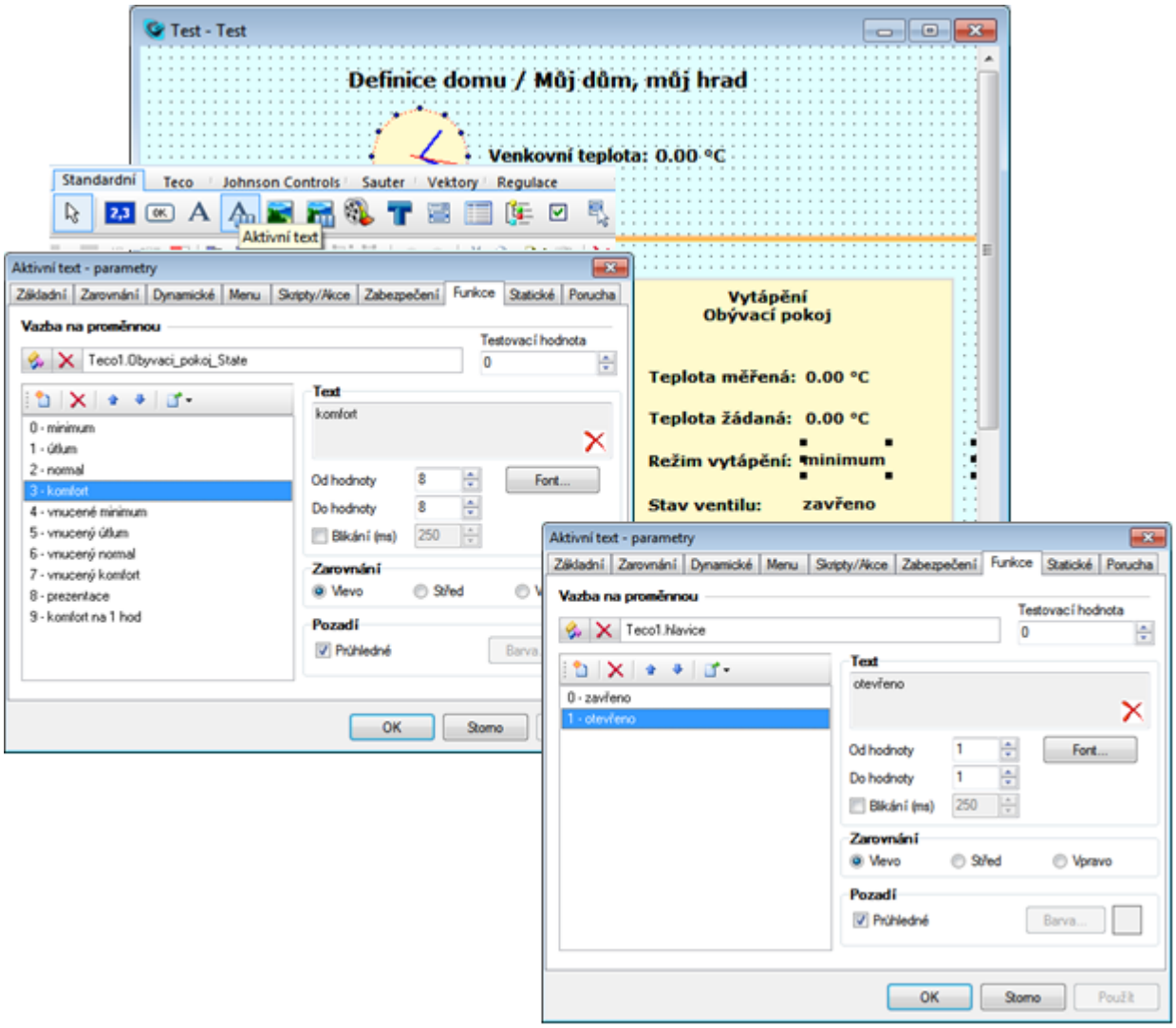
At any time, you can run (using the *Project -> Run* command) the application and check whether the correct data is displayed (picture No. 15).



Picture No. 15 - Visualization and control of lighting

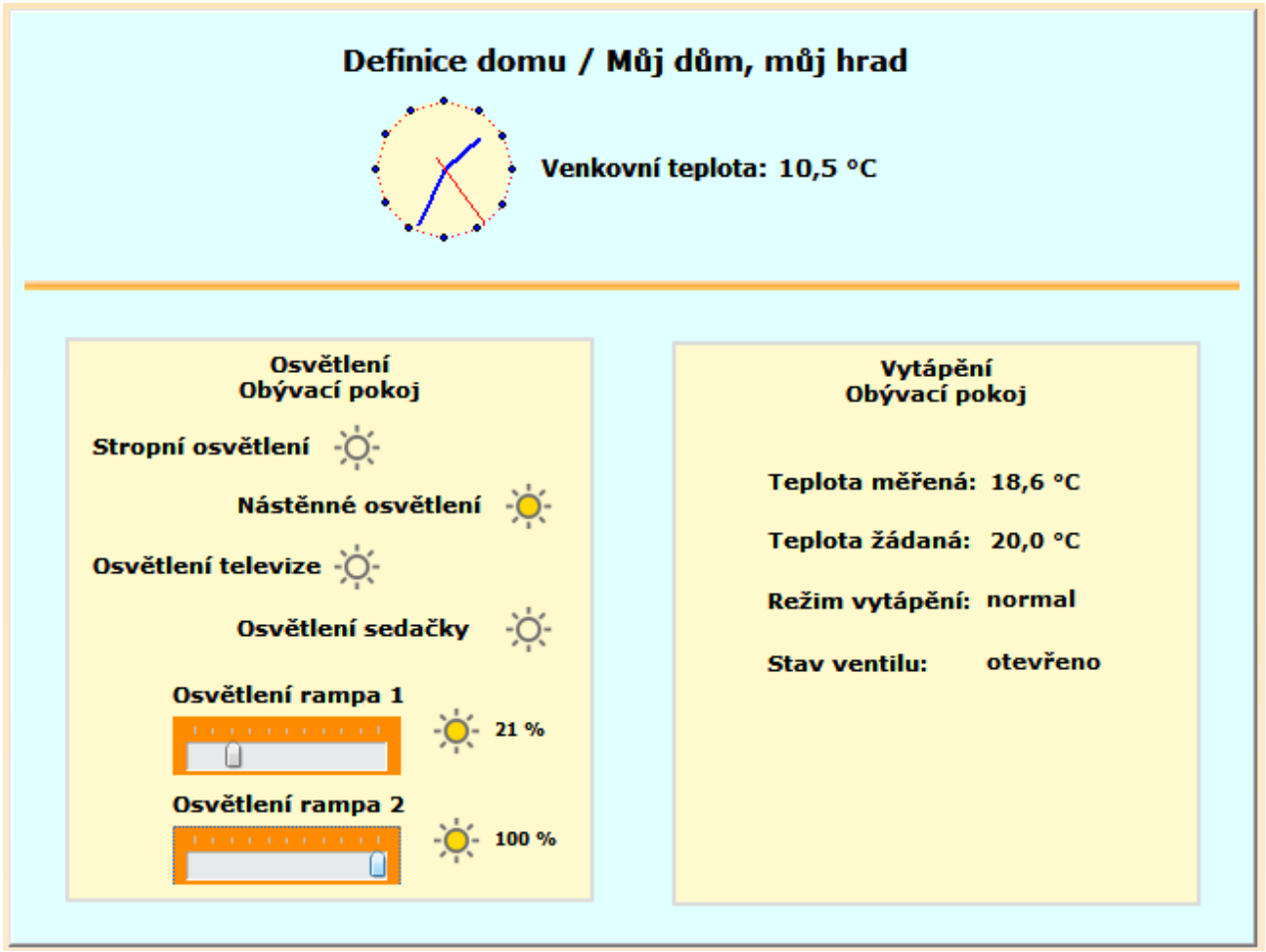
5.2.2 Visualization and Control of Heating Circuit, Use of Time Program

To visualize and control a heating circuit and for time program purposes, additional components must be added. These additional components are not only the known *Bevel,Text,* and *Display,* but also the *Active Text* component (picture No. 16). As shown in the left screenshot, this component will display the current heating circuit mode, which is defined in the FoxTool development environment. You can now define the link to a corresponding tag and enter the displayed text, or you can adapt the component graphically. In the right screenshot, the component displays the head status - again, define the link to a tag, or you can adapt the component graphically.



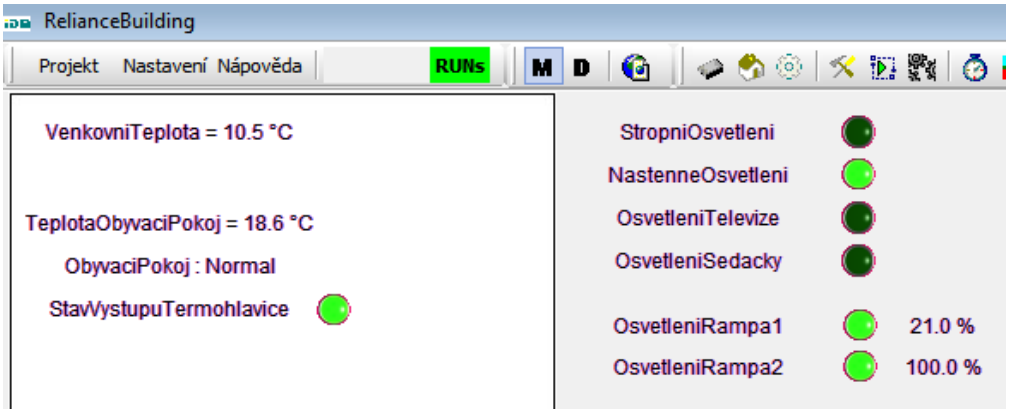
Picture No. 16 - Visualization and control of a heating circuit

At any time, you can run (using the *Project -> Run* command) the application and check whether the correct data is displayed (picture No. 17).



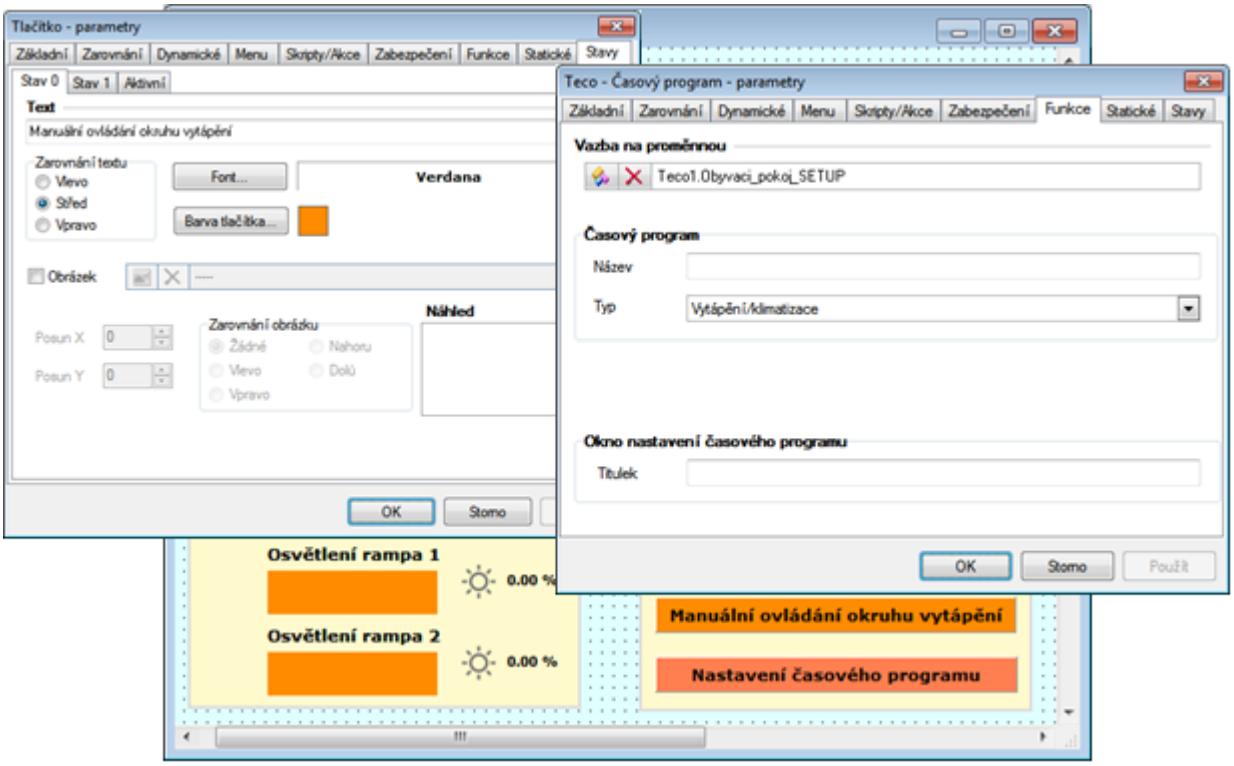
Picture No. 17 - Visualization and control of a heating circuit

If you compare the runtime data to the FoxTool development environment visualization, you will see that the displayed values and states are identical (picture No. 18 including the lighting visualization status).



Picture No. 18 - Data and status display in the FoxTool development environment

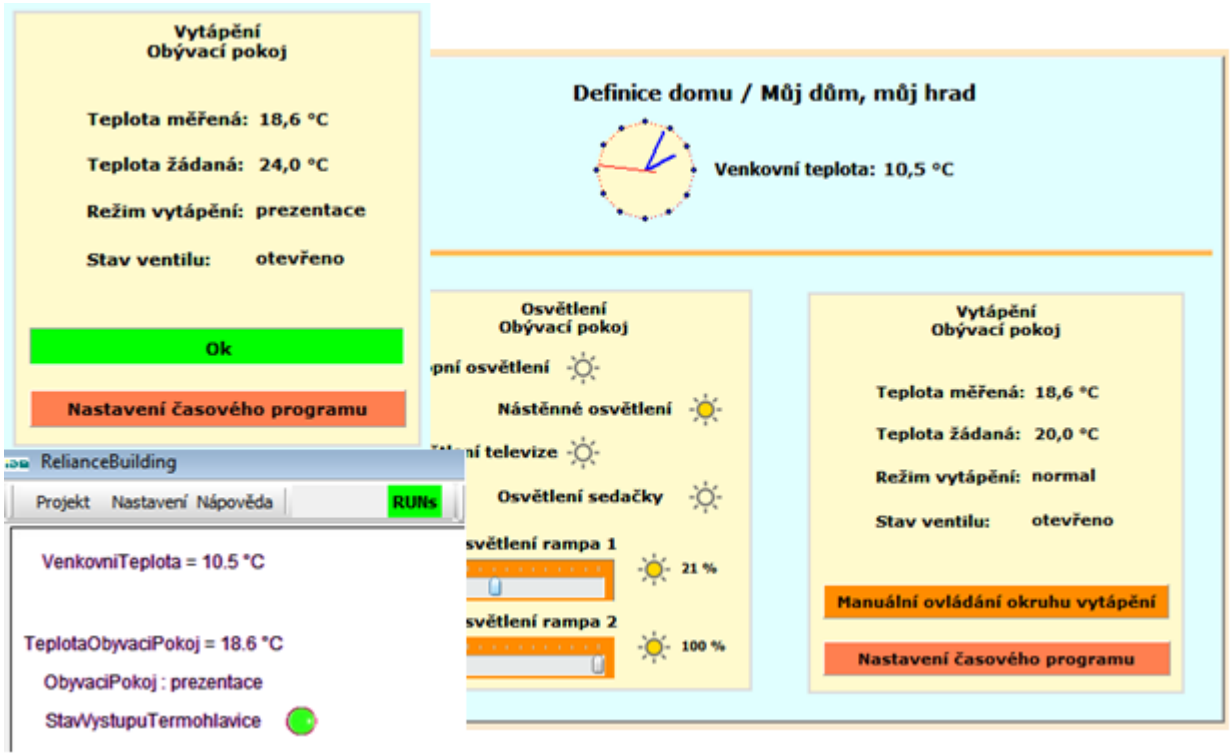
To manually intervene in the heating circuit automatic control, e.g., using a thermostat (RCM2-01, IDRT2-01, IART2-01), it is essential to provide manual control of the circuit. This manual setting will make it possible to switch between the modes Presentation, Forced Comfort and Control by Time Program (similarly, we can visualize and control any time program mode). Now, select and paste a *Button* component (picture No.14). Enter the text "Manual control of the heating circuit" (picture No. 19) and define the link to a tag, or you can adapt the component graphically.



Picture No. 19 - Visualization and control of a heating circuit

The *Teco - Time Program* component is designed for working with time programs in the Tecomat® PLC (picture No. 19). Paste this component into the visualization window, define the link to a corresponding tag, or you can adapt the component graphically. This component is intended for setting a selected time program (i.e., without interfering with the PLC user program runtime).

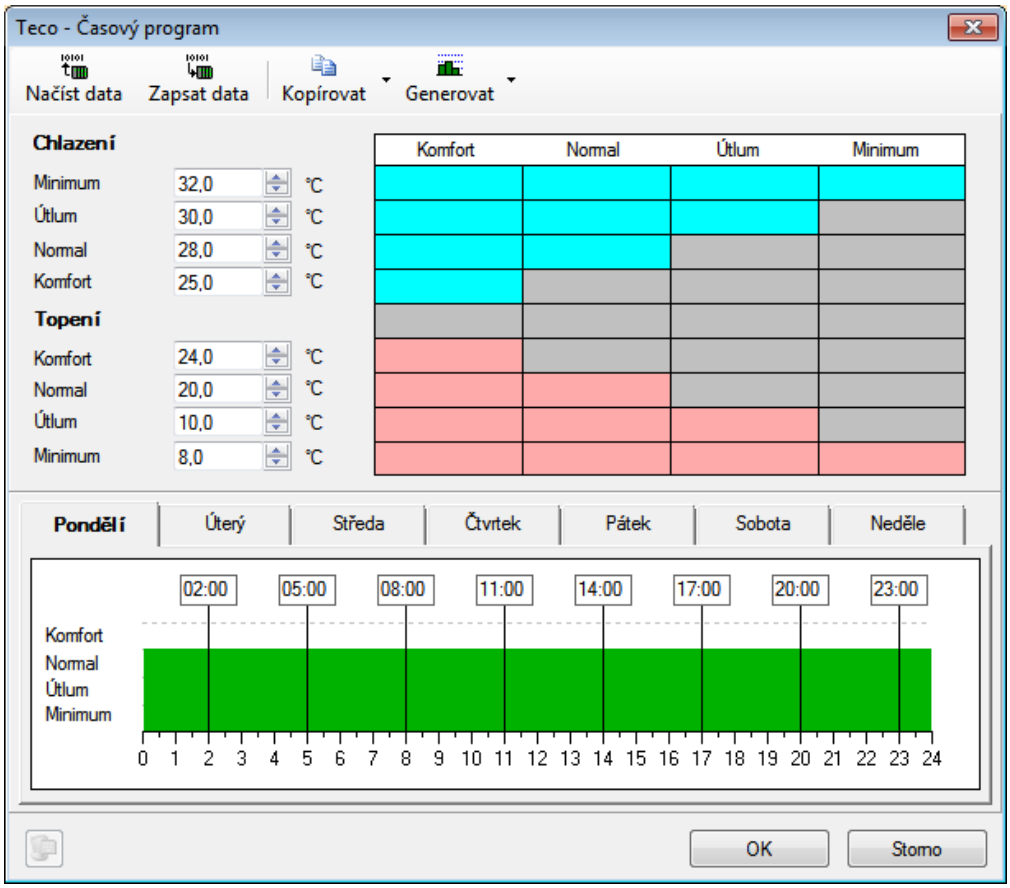
Run the application (the *Project -> Start* command) and check the functions of the added components (picture No. 20).



Picture No. 20 - Visualization and control of a heating circuit

By pressing the "Manual control of the heating circuit" button, "Required temperature" and "Heating mode" can be modified (picture No. 20). Concurrently, the modified values will appear in the FoxTool development environment (picture No. 20).

By pressing "Time program settings", it is possible to display the "Teco - Time Program" window. This window is nearly identical to the Time/Week Program Manager in the FoxTool environment (picture No. 21). In this window, you can change time program settings for a selected heating circuit. It is possible to change temperatures in each mode, timestamps, and corresponding modes in time intervals.



Picture No. 21 - Time program settings

6 Conclusion

If you are interested in learning how to develop a visualization project in a visualization (SCADA/HMI) system, or how to connect the system to a PLC, but you consider it too complicated to install the software, reconfigure your PC and work in a development environment, there's nothing to fear. Try the **Reliance** SCADA/HMI system.

In this manual, we have shown you that creating a simple project in the **Reliance** development environment and its interconnection with the **CFox** applications are user-friendly and fast. Our example simply describes how to visualize and control lighting and heating systems. There are, of course, more systems you can visualize and control, including data archiving, database creation, etc.

If you would like to know more about this topic, do not hesitate and try the **Reliance** system. Working in the **Reliance Design** development environment is:

- clear
- intuitive
- user-friendly
- rapid (RAD - rapid application development)

Among the main priorities of the system are its stability, reliability, user-friendliness, and the speed of development. As a result, a robust, reliable and easily scalable system, optimized even for large applications, is available and ready to be used.

When using **Reliance Design**, you will appreciate:

- mastering working with the development environment is speedy
- modern design, themes
- manager and wizard system makes creating a visualization easy
- basic functions are configured instead of being programmed
- additional functions can be programmed in the VBScript language
- easy extensibility of visualizations

- detailed project diagnostics (analyzing a visualization project, detecting possible problems, selecting unused components, significant time savings during debugging and putting into operation)
- fast and quality technical support from the Reliance Team
- examples and the FAQ section available on the Reliance website www.reliance.cz

Thousands of Reliance installations have been successfully deployed to control system processes. Among the customers who rely on the Reliance system daily are worldwide renowned companies: RWE Group, E.ON, Coca Cola, Panasonic, Heineken, Pilsner Urquell, Budweiser Budvar, Groupe Soufflet, Aliachem, Olympus, KGHM copper mines, Toyoda, Mercedes-Benz, Toyota, Peugeot, Citroën, Pirelli, Tesco, Migros, Karbosan, Vileda, Phoenix-Zeppelin ...

You too can become a satisfied customer and member of the Reliance Team!

7 Recommended Documentation

1. Reliance 4 - First Steps, © 2011, GEOVAP, spol. s r.o.
2. Reliance 4 - Development Environment, © 2011, GEOVAP, spol. s r.o.
3. Starting with the MOSAIC Environment, 7th edition, February 2008
4. Programmable 32 bit Tecomat PLC Serial Communication, 16th edition, March 2009