ul. Konstantynowska 79/81
95-200 Pabianice
tel/fax 42-2152383, 2270971
e-mail: fif@fif.com.pl

Programmable time relay

PCS-533

User manual

v. 1.0.1



Information concerning the safe use of the relay is marked with symbols. All information and recommendations bearing these symbols should be strictly obeyed.

	Risk of electric shock.
	Potentially dangerous situation that could lead to a danger for the operating personnel or damage to the relay.
Information concerning the design, operation and maintenance of the time relay.	
	Important information, valuable tip.
	Practical advice, problem solution.
	An example of use or operation.

Table of contents

Introduction 5
Installation 5
Description of terminals 5
Specification 5
Connection schemes 6
Indications 8
PCS33 Configurator 10
Welcome screen 10
Main menu 11
Program editor 12
Edit program list 15
Scroll program list 15
Edit entries 16
Delete entries 17
Change the order of entries 17
Command list 18
Reset 18
Stop 19
Pause 21
Return to 22
Output 23
Input A 25
Input B 25
Input A – special function 27
Input B – special function 27
Templates 29
Template list 30
01) Delayed switching on 30
02) Delayed switching off 31
03) Generate a single pulse with rising edge 32
04) Generate a single pulse with trailing edge 33
05) Switch-off delay with support function 35
06) Generate a single pulse after a preset time 36
07) Cyclic delayed switching on 37

08) Cyclic delayed switching off....38
09) Switching on of the contact for the time t1 with the button pressed and for the time t2 with the button released....39
10) Generate a preset number of pulses....40
11) Generate pulses for a preset period of time....41
History of changes....43
Warranty....44

Introduction

PCS533 is a programmable time relay that allows the implementation of simple programs for controlling the output relay as a function of time and in conjunction with signals directed to control inputs A and B. The operation mode of the controller is prepared via graphical application running on the Android phone. NFC interface built-in the phone is used to transfer the program (*).

Installation

Description of terminals



1-3) Power supply - 12-265V AC/DC
4) Control input A
6) Control input B
10) Relay output – contact normally closed (NC)
11) Relay output – common contact (COM)
12) Relay output – contact normally open (NO)

Fig. 1) Description of terminals

	**Please note**: No galvanic isolation between the power supply (terminals 1 - 3) and the control inputs (terminals 4 and 6). In case of a mains power this means that a dangerous potential can emerge on the terminals 4 and 6.

Specification

- Power supply: 12 – 265 V AC/DC
- Control inputs
  - Number of inputs: 2
  - Control voltage: 12 – 265 V AC/DC
  - Triggered with:
    - level L or N for AC power supply
    - level + for DC power supply
  - Input circuit triggering current < 1 mA
- Output

  - Single relay with NO/NC contact
  - Acceptable load
    - 16 A for 250 V AC voltage in load class AC-1
    - 16 A for 24 V DC voltage
    - 1 A for >24 V DC voltage
- Relay type: Freely programmable
- Number of program steps: 200
- Timers: from 0.1 to 24h
- Controller state indication: 2 x LED
- Power consumption:
  - 0.8 W with relay on
  - 0.4 W in standby mode
- Operating temperature: -25 … + 50 degrees
- Thermal protection: 70 degrees
- Terminal: Screw terminals, max. 2.5 mm^2
- Installation: on DIN rail. Size 1S
- Ingress protection: IP20

Connection schemes

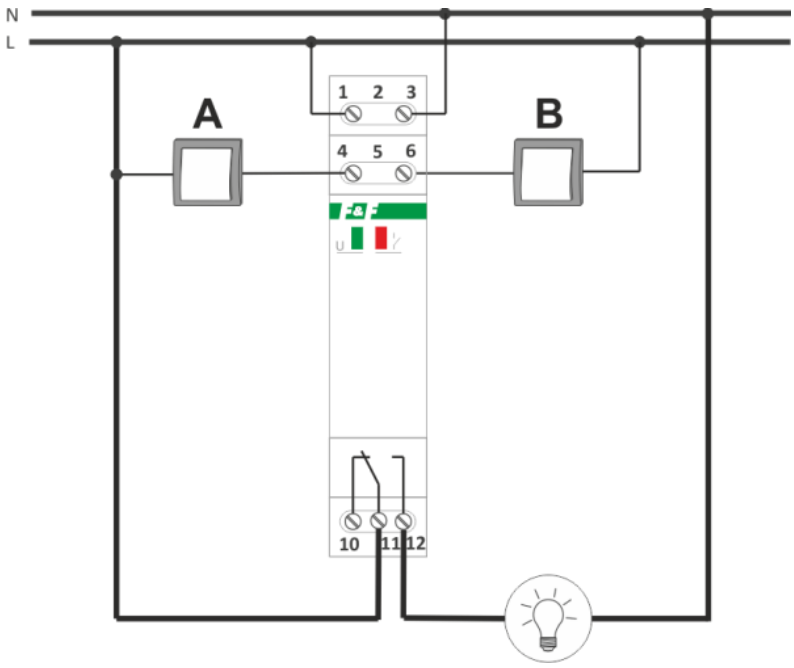


Fig. 2) Alternating voltage power supply (control with level L)

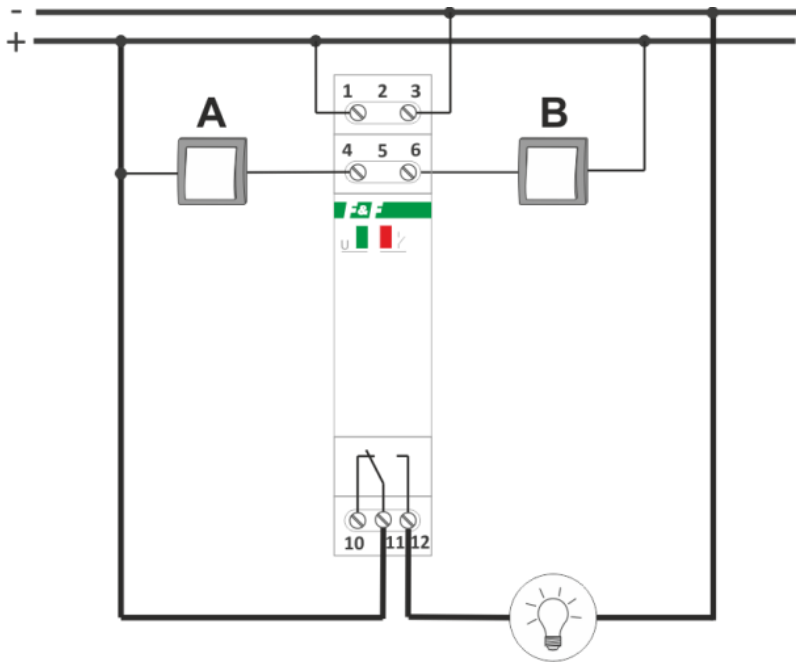


Fig. 3) Direct voltage power supply (control with level +)

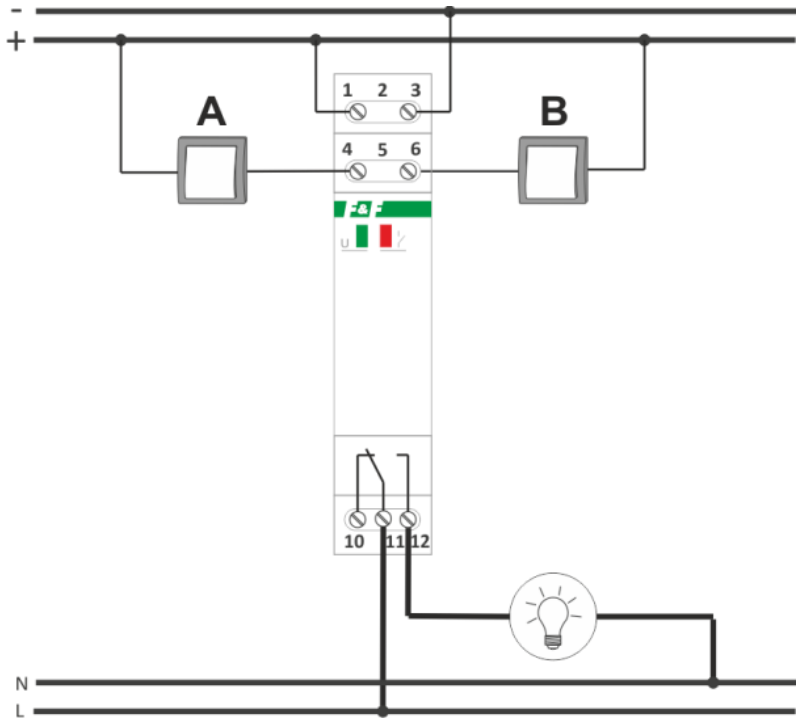


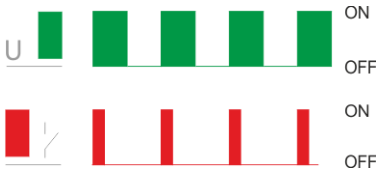
Fig. 4) Separated power supply circuit/control and receiver

Indications

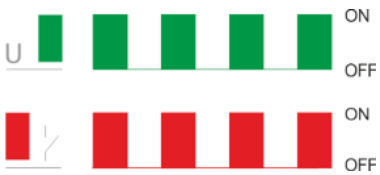
The PCS533 relay is equipped with two signaling LEDs. During normal operation the green LED indicates the presence of power supply voltage. The red LED indicates the switching on of the relay. Flashing of one or both controls indicates the emergency situation. The list of possible errors is shown in the following table.

<table>
<tr><td>
</td><td>No program

Flashing green LED and red LED off indicates that there is no proper program to execute saved in the memory of the controller.

Please note: Due to the way the program is transferred from phone to controller, it may happen that the whole program will not be properly copied to the memory of the controller. In this case, when the NFC connection is broken, the controller tries to load new program and in case of an error it indicates it by flashing green LED. Similar verifying procedure is performed when the controller activates.</td></tr>
<tr><td>
</td><td>Disruption of the timer signal
In case of a strong electromagnetic interference the operation of the quartz generator that is used for measuring the time may be disrupted. In that case the controller will switch to the secondary clock of lesser accuracy. In that situation, move the controller away from the source of interference.</td></tr>
<tr><td>
</td><td>Disruption of the memory readout
Strong electromagnetic interference may also lead to disruption of communication with EEPROM memory that stores the program configuration. The relay is equipped with procedures for restoring the communication with memory, but if such errors occur it is recommended to move the relay away from the source of interference.</td></tr>
<tr><td>
</td><td>Exceeding the temperature

Alternate flashing controls indicate the maximum temperature inside the module. Until the temperature does not drop to a safe value, the relay will be switched off and the program execution stopped.

Temperature alarm may occur in the following cases:
1. Too high load of the relay contacts. In this case reduce the load, for example through the use of an additional contactor through which the load will be switched on.
2. Too high ambient temperature. In this case improve ventilation in the distribution box or increase spacing between modules to improve heat dissipation.
3. Too high power supply voltage (above 265 V)
4. Improperly tightened terminals of the controller power supply (1-3) or load (10-12)</td></tr>
<tr><td>
</td><td>General error

General error of the controller is signaled by the simultaneous flashing of both control lights. It indicates the faulty operation of the controller or erroneous entry in the configuration memory.</td></tr>
</table>

	 The most common cause of a general error is erroneous entries in the controller program memory. Usually they appear in case of an incorrect transfer of the program from the phone to the controller (for example caused by the communication break during data copying). In the case of a general error in each case again copy the program from the phone to the controller.

PCS33 Configurator

All operations related to the preparation of the program, copying from/to the controller and managing the programs is realized through the **PCS533 Configurator** application. This application is designed for phones and tablets running on Android (version at least 4.0). In addition, to communicate with the controller the phone or tablet must be equipped with near field communication (**NFC**) module. The PCS533 Configurator app can be downloaded free of charge via the Google Play store.

	Google Play store performs initial verification and blocks the ability to install the app on devices that do not meet the application requirements (for example lack of NFC module or unsupported version of Android). **In this case, the attempt to search the app in the store will fail and the message about the lack of searched application will be displayed.**

Welcome screen

After starting the PCS533 Configurator, a window is displayed that allows you to read the program from the PCS controller. (

Fig. 5



Fig. 5) Welcome screen

According to the instructions on the screen, when you bring the phone closer to the PCS533 controller, the connection will be made automatically and the program from the controller memory will be copied to the application.

	NFC (Near Field Communication) is used for connecting the controller and the phone and has a small operational range. Therefore, due to the small size of the antenna in the controller and equally small NFC antennas in phones, difficulties in the connection making may occur. In this case you need to experimentally locate the antenna in the phone by slowly moving the phone over the front elevation of the controller until the connection is made (indicated by displaying the download progress bar, among other things).

In case the NFC module in the phone is disabled when the application starts, the appropriate message will be displayed on the screen (Fig. 6) that redirects to the phone settings where you can enable the NFC module.

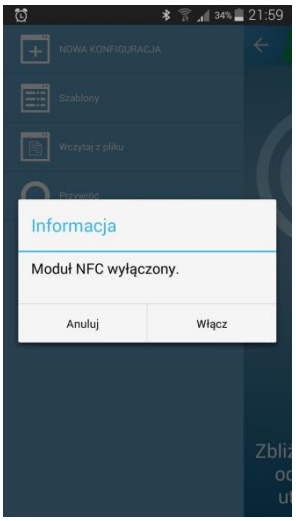


Fig. 6) Message about disabled NFC module

Main menu

Main menu is opened by clicking on the icon located in the upper left corner of the application (Fig. 7).

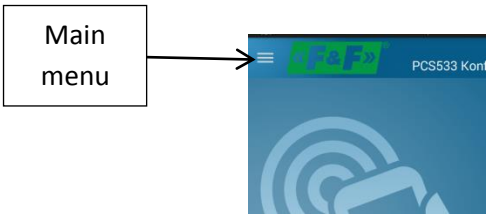


Fig. 7) Opening main menu

The look of the main menu is shown in Fig. 8 and it is composed of the following commands:

- **Continue editing** – Return to the edition of the current program. This command does not appear if there is no program in the memory of the configurator.
- **New configuration** – create new, empty program file.
- **Templates** – a list of pre-defined operating programs. For use as a ready solutions or as a base to create your own program.
- **Load from file** – read the previously created program from the phone memory.
- **Restore** – the ability to run any program saved in the memory of the controller.

- **Information** – a brief description of the application.

The details of each command are described below.

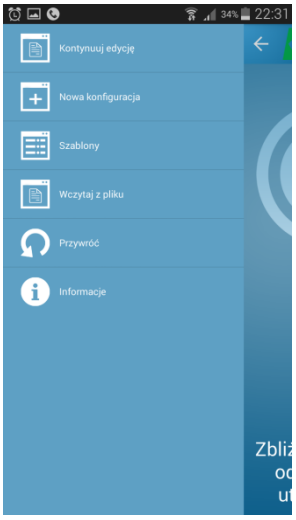


Fig. 8) Main menu

Program editor

Program editor is the main view of the PCS533 Configurator app and it is designed for the creating and editing of the program of the controller operation.

If any program is loaded to the application at the opening of the editor window, a message will be displayed with a question what to do with the previous entry (Continue or start over?
Start over - Continue

Fig. 9



Previous entry was interrupted.
Continue or start over?
Start over - Continue

Fig. 9) Configuration override

Selecting **Start over** option removes the previous entry and opens new configuration. **Continue** option stops the execution of the command and leaves the previous configuration.

Example screen of the program editor is shown in Fig. 10. It consists of the following elements:

- Header at the top of the screen If the program is linked with the particular controller, the **DEV** parameter displays the identifier of the device type (PCS533) and the **ID** parameter - the unique identification number of the controller**.**

	By default the identifier of the clock is presented as a serial number of the controller. It is also possible to give the device other name. This is done using the button with a pencil symbol that is located on the right side of the header. After entering your own name, it will appear whenever you connect with a given controller.

- Workspace in the middle part of the screen with a list of commands to perform.

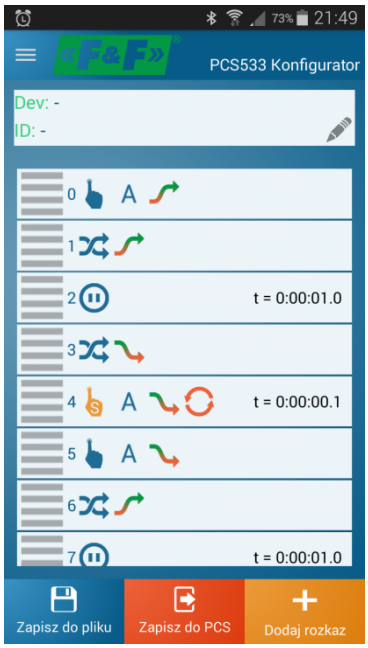


Fig. 10) Program window

- Bar with the function keys on the bottom of the screen:
  - **Save to file**. This command allows you to save the current list of programs to a file. When you press the button, a dialog box appears that allows you to enter a file name.

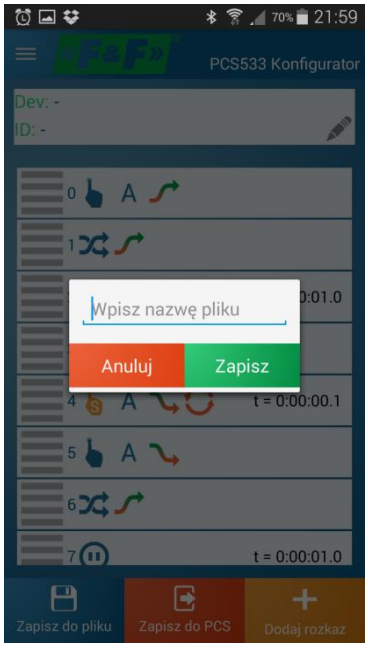


Fig. 11) Saving program to a file

The file saved in this way will be available from the Load from file command that is located in the main menu of the application.

	Please note: The application does not verify whether a file with the entered name already exists. If you will use the same name, the new configuration will be overwritten in place of the previous one without displaying any additional warning.

- **Save to PCS** – a command to copy the current program to the PCS533 controllers. When you press the button, a message box will appear asking you to bring the phone closer to the device in order to save data.

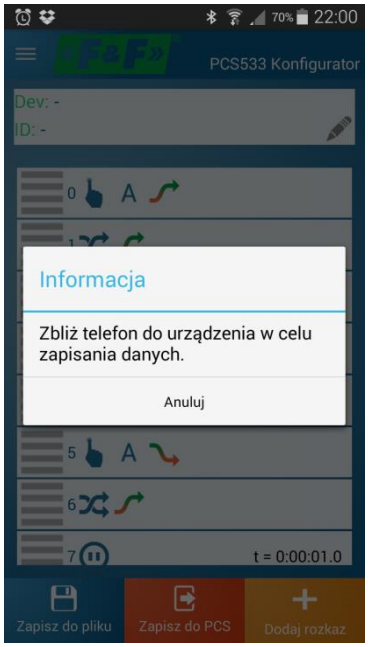


Fig. 12) Save to PCS

Data copying will start automatically once the phone and controller are connected. When the copy is done, a message box will appear with information about correct or invalid saving of the program. If the copying process was correct, you can program other controllers without taking any action in the application - you just need to bring the phone closer to the next controller. To stop further entries, press the Cancel button displayed on the screen. The new program in the controller will begin to execute immediately after the connection between phone and controller is terminated.

	If the controllers are in close proximity to each other, the transmission may be interfered, for example part of the program may be copied to one device, and part to the other. In such situation it is recommended to spread controllers apart for the time of programming or for example draw out the programmed controller higher. The controllers may also be programmed with the power supply voltage off, so the power failure does not prevent the interference in such cases.

- **Add command** - a command that displays a window with a list of commands that can be added to the program (Fig. 13). A detailed description of those commands and their parameters is presented later in this manual.

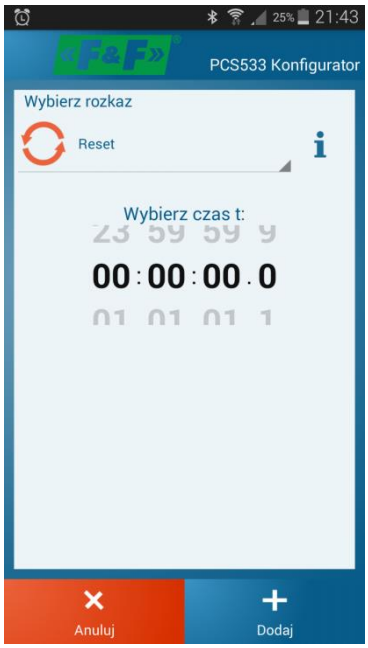


Fig. 13) Adding/editing command

Edit program list

Program of the PCS533 controller can contain up to 200 steps executed according to the order of appearance on the list.

Scroll program list

To scroll the list of the programs press the right side of the program list (Fig. 14) and then move the view to the previous or next program steps by moving the list up-down.

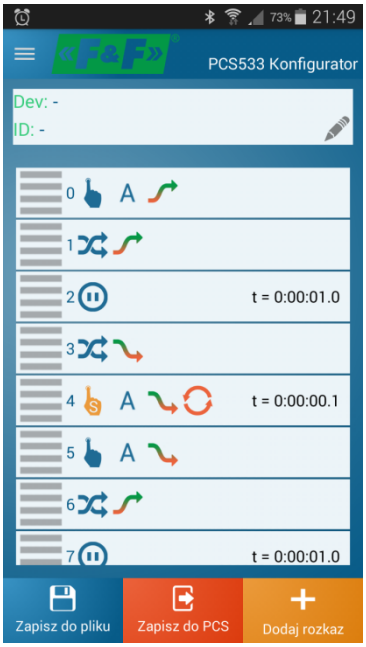


Fig. 14) Scroll program list

Edit entries

To enter the edit mode of the selected step, briefly click the command entry. You will then see the command edit window - examples (depending on the edited command) are shown in Fig. 15.

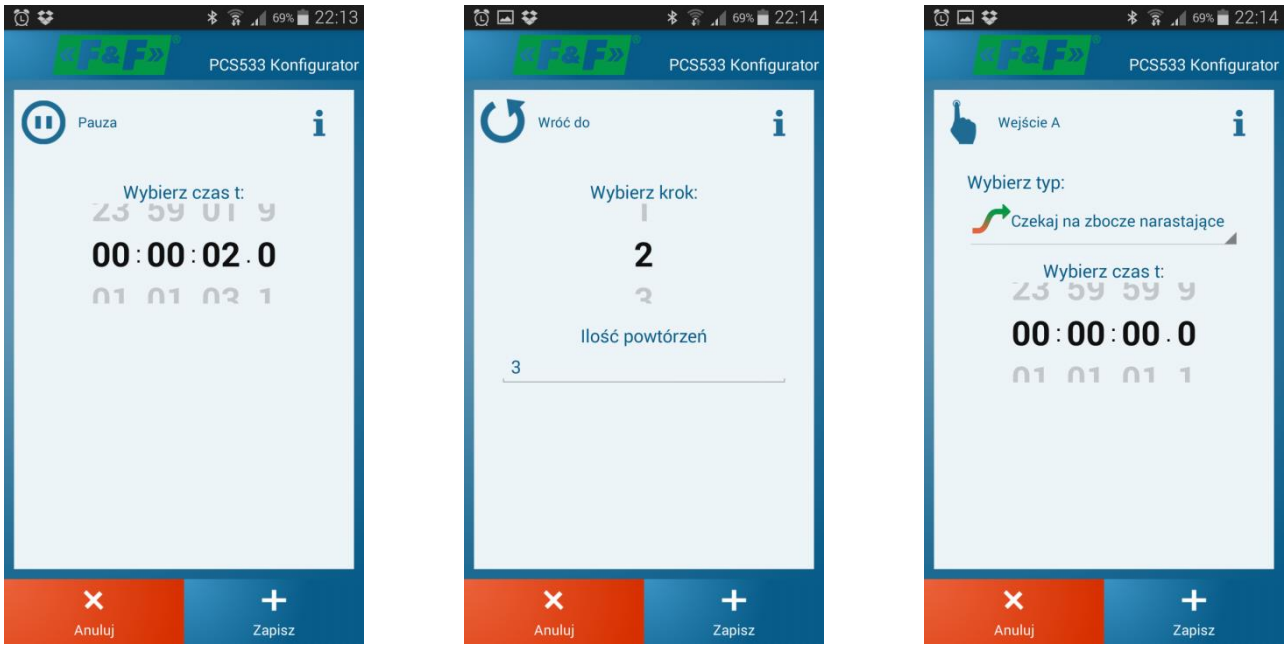


Fig. 15) Command edit window

Edit window consists of three main parts. The upper one contains the name of the command and icon symbolizing it - in examples from Fig. 15 they are respectively: **Pause, Return to** and **Input A**. On the right side of the name of the command is located the **i** button - it displays a window with detailed information concerning the selected command (the example for the **Return to** command is shown in Fig. 16).

Information

A jump to a preset program step is carried out. We assume that the jump can be done only to the existing step. The Counter parameter determines how many times the jump will be done. If the Counter=0, the number of returns is unlimited. If the Counter>0 then the program will execute a preset number of steps. **Please note**: the Return to commands cannot be in loop.

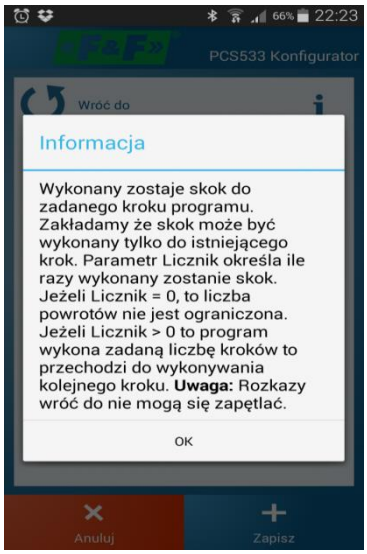


Fig. 16) Information for Return to command

Delete entries

To delete a selected program entry, press and hold it and then move it to the left or right.

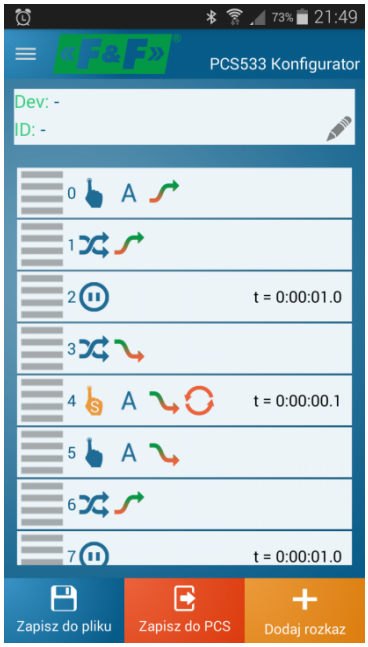


Fig. 17) Deleting program entry

.

	Please note: When you remove program entries, you should pay close attention and remember that it may change the way the program operates. This is particularly true for the jump commands which may be placed in the further part of the program. As a result of the removal of the step, the jump will not return to the assumed point of the program.

	After removing the command at the bottom of the screen the „Restore removal" message will be displayed (Fig. 18). It allows you to recover accidentally erased command.

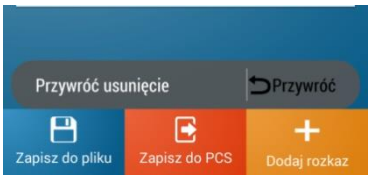


Fig. 18) Restoring deleted command

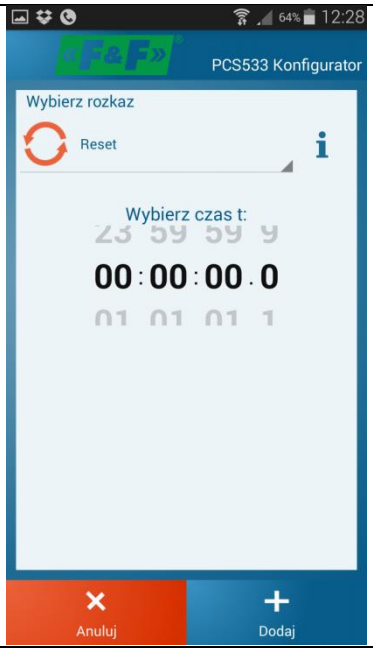
Change the order of entries

The order of the commands on the list determines the order of their implementation by the controller. By holding down the symbol located on the left side of the entry you can move the command to the new position in the program.

	Please note: Changing the order of entries may result in undesirable behavior of the program - this is particularly true for the jump instructions that after such change will not return to the assumed point of the program.

Command list

	Reset



The **Reset** command:

- aborts the current program;
- switches off the relay;
- resets the internal timers and counters of the program loop;
- deletes the settings of the special inputs;
- causes the return to the first step of the program and starts the execution of the program from the beginning.

The execution of the **Reset** command may be delayed for the time specified with parameter **Select time t** in range from 0.0 (immediate execution of command) to 24 hours in increments of 0.1 s. Countdown to execution of **Reset** command does not stop the execution of the further part of the program. The **Reset** command will be executed after the preset time regardless of the currently executed step of program. If during the countdown to execute the Reset command another **Reset** command will appear then it will replace the previous command (countdown to reset will start from the beginning according to the time set in the new command).

	Stop



The **Stop** command stops the execution of the program. The execution of the **Stop** command may be delayed for the time specified with parameter **Select time t** in the range from 0.0 (immediate execution of command) to 24 hours in increments of 0.1 s. Countdown to execution of **Stop** command does not stop the execution of the further part of the program. The **Stop** command will be executed after the preset time regardless of the currently executed step of program. If during the countdown to execute the Reset command another **Stop** command will appear then it will replace the previous command (countdown to reset will start from the beginning according to the time set in the new command>.

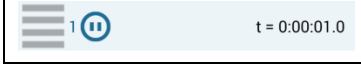
Operation of the controller can be restarted after the **Stop** command by:

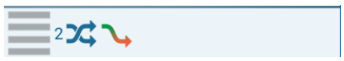
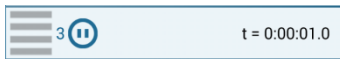
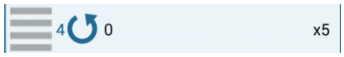
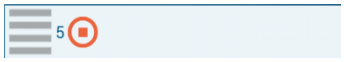
- execution of previously defined delayed **Reset** command;
- triggering the special input to which the **Reset** command was earlier assigned;
- reconnecting the controller power supply.

In each of these cases, the program execution will start from the beginning.

Example

When you turn on the power supply, the relay will activate in cycle of 1 second ON/1 second OFF. The cycle will be repeated five times and then the program will be stopped with the **Stop** command. Next power-up will start the cycle from the beginning.

Symbol of the command	Parameters	Description of operation
	Command: Output Mode: On Time: 0 s (permanent switching on)	First **Output** command sets the relay in **ON** position. Time of the switching on is set to 0 which means the relay will stay on until the next **Output** command will appear.
	Command: Pause Time: 1 s	Stops the execution of the program for 1 second.

	Command: Output Mode: Off Time: 0 s	After a one second delay the next **Output** command will be executed that sets the relay in **Off** position. Run time is set to 0 which means the relay will stay on until the next **Output** command will appear.
	Command: Pause Time: 1 s	Stops the execution of the program for 1 second.
	Command: Return to Select step: 0 Number of repetitions: 5	The **Return to** command undoes the execution of the program to step 0 (the beginning of the program). This cycle will be repeated five times and then the next step of the program will be executed.
	Command: Stop Time: 0 s	Stop of the program execution. The next start of the program will take place after the next power-up of the controller.

	Pause



The **Pause** command is used to temporarily stop the execution of the program. Time of the interval can be set using the **Select time t** selector in range from 0.1 second to 24 hours in increments of 0.1 second.

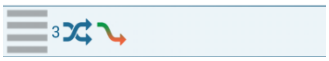
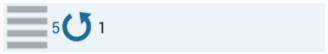
Please note: The **Pause** command:

- does not stop the countdown to execute the **Stop** and **Reset** commands;
- does not stop the countdown to the switch on/switch off of the relay that is set with the **Output** command;
- does not block the implementation of the functions performer by the special inputs A and B.

Example

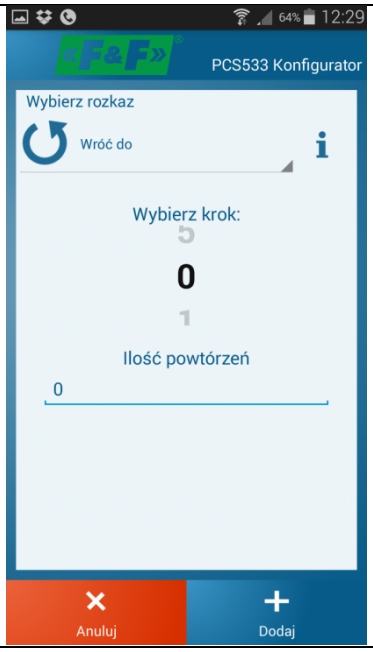
Sequence 2s ON, 3s OFF is activated ten seconds after the power supply is turned on. The sequence is repeated until the power supply is turned off.

	Command: Pause Time: 10 s	Execution of the program is paused for 10 seconds after the power supply is turned on.
	Command: Output Mode: On Time: 0 s	After the ten seconds pause the **Output** command will be executed that sets the relay in **ON** state. Time is set to 0 which means the relay will stay on until the next **Output** command will appear.
	Command: Pause Time: 2 s	Pause stops the execution of the program and the activated relay for duration of 2 seconds.

	Command: Output Mode: Off Time: 0 s	After the execution of the previous **Pause** command the Output command will be processed that sets the relay in **OFF** state. Time is set to 0 which means the relay will stay on until the next **Output** command will appear.
	Command: Pause Time: 3 s	Stop the execution of the program and hold the relay in OFF state for 3 seconds.
	Command: Return to Step: 1 Number of repetitions: 0	After the three seconds delay the next Return to command will undo the execution of the program to step 1 (reactivation of the relay).The number of repetitions in Return to command is set to 0 which means that the sequence of commands 1 - 4 will be cyclically repeated until the power supply is off.



Return to

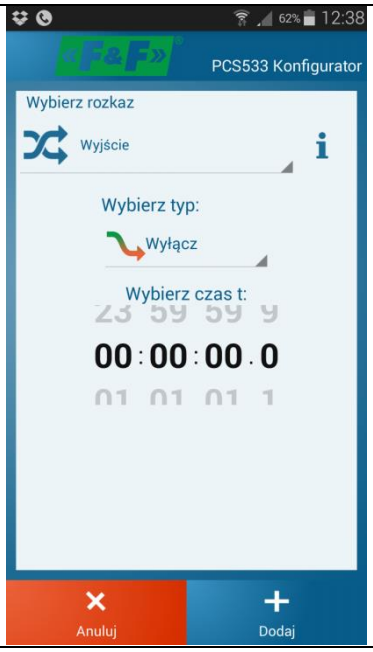


The **Return to** command allows you to move the point of program execution to previously set step, thanks to which it is possible to cyclically repeat groups of commands.

This command has two parameters:

- **Number of a step** - a step to which the execution of the program will be taken back;
- **Number of repetitions** - parameter that indicates how many times the Repeat to command will be repeated. If the number of repetitions will be set to zero then the command will be repeated unlimited number of times. A number greater than zero indicates how many times the given step will be repeated.

	Output



The **Output** command is designed for direct control of the status of the relay output. The command consists of two parameters: **Select type -** in which the action of the relay is set and **Select time t** - which defines the time of the switching on/off of the relay.

Select type - selection of the relay action

	Switch off	Setting the relay in to **OFF** position
	Switch on	Setting the relay in to **ON** position
	Opposite state	Switching the relay to the opposite position - from the **OFF** position to position **ON**, and from the **ON** position to position **OFF**.

Select time t

This parameter allows you to activate the relay for a given time regardless of the currently executed program. If the time is set to zero, the relay will be switched on permanently (until the next **Output** command appears). If the time is set to a value greater than zero the relay will be set in a given state, the time countdown will start and the controller will proceed to process next steps of the program. After a specified time, regardless of the currently executed step of the program, the state of the relay will be switched to the opposite position.

Execution of the next **Output** command interrupts the execution of the earlier one - new action of the relay is set and the countdown from the new command starts.

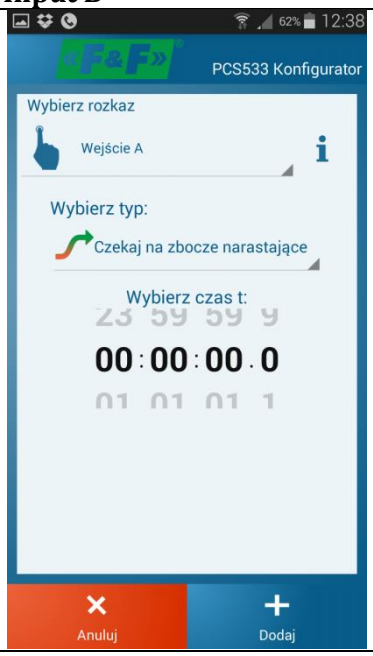
Warning: The pair of commands **Output** (to set the state of the relay) and **Pause** (to stop the execution of the program for a given time) is better suited to implement the time control of the relay.

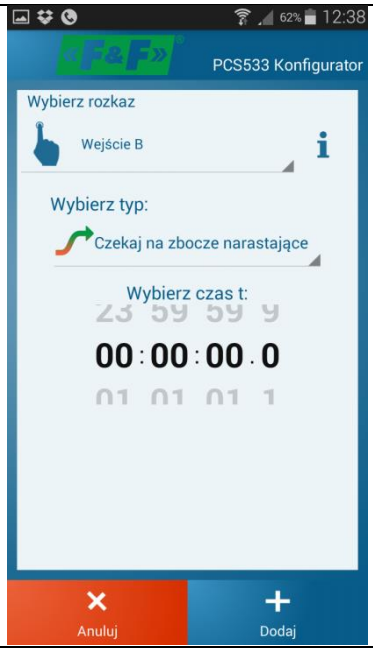
Example

The relay will be set to ON position for 1 minute. If during this time button A will be pressed then the relay will switch off at the moment the button is pressed.

Command	Command: Output Mode: Switch on Time: 1 min	This command sets the relay in the ON state for a time of 1 minute and then the controller will perform the next step.
Input A	Command: Input A Mode: Wait for a rising edge Time: 0 s	The controller waits for the pressing of the button A while at the same time the time set in the previous step is counted down. Pressing the button will cause the next step of the program to be executed. If the preset time of the switching of the relay passes during the waiting for the pressing of a button then the relay will remain off, but the program will still be waiting for the pressing of a button.
Output	Command: Output Mode: Switch off Time: 0 s	Command to set the relay in the OFF mode. If the button A is pressed before the run time of the relay elapses then this command will turn off the relay earlier. If the button is pressed after the preset time is up then the relay will be turned off earlier and this command will not introduce any change.

	Input A **Input B**





The **Input A**/**Input B** command stops the execution of the program until the event that was defined in the command appears at the output. The type of the event is selected using the **Select type** parameter. The following options are available here:

	Wait for rising edge	The controller detects the pressing of a button. If after the pressing the button remains pressed for a predetermined time then the controller executes the next step of the program.
	Wait for trailing edge	The controller detects the button release. If the button remains released pressed for a time set with the **Select time** parameter the controller proceeds to execute the next step of the program.
	Wait for any edge	The controller responds to the emergence of any edge at the signal at the input. If the button is pressed (rising edge) and remains pressed for a preset time then the controller proceeds to execute the next step of the program. If the button is released (trailing edge) and remains released for a preset time then the controller proceeds to execute the next step of the program.

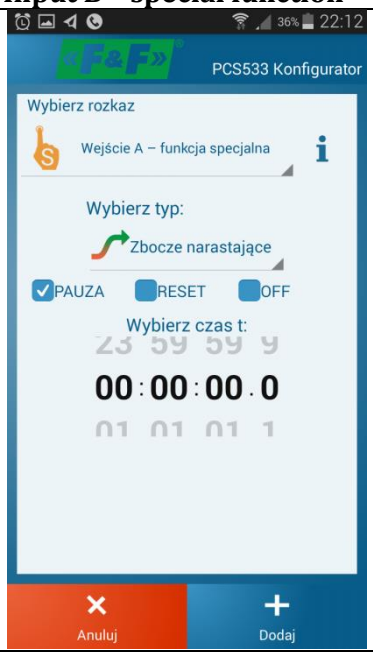
	Wait for low level	If a low level is maintained for a preset time at the input of the controller (button released) then the controller proceeds to execute the next step of the program. Warning: Timing starts at the emergence of the **Input** command.
	Wait for high level	If a high level is maintained for a preset time at the input of the controller (button pressed) then the controller proceeds to execute the next step of the program. Warning: Timing starts at the emergence of the **Input** command.

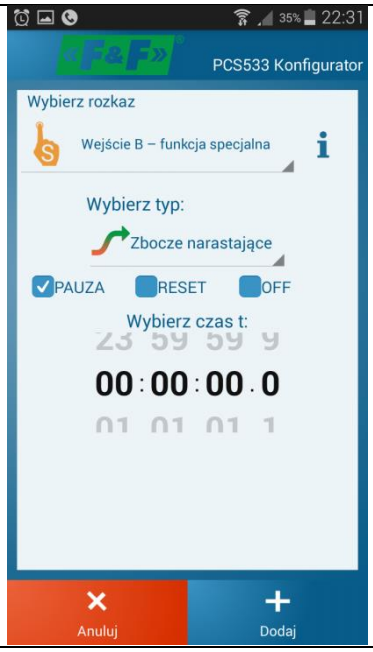
The **Select time t** parameter defines the preset time for which the input must maintain a certain level for a command to be executed and for a controller to execute the next step of the program. The time of the delay can be adjusted from 0.0 (immediate reaction) to 24 hours.

Warning: Execution of the **Input A/Input B** command:

- does not stop the countdown to execute **STOP** and **RESET** commands;
- does not stop the measuring of the switch on/switch off time of the relay that is set by the **Output** command;
- does not block the implementation of functions performer by the special inputs A and B.

	Input A – special function **Input B – special function**





Special functions of inputs A and B allow you to define the events at the input which will be handled at any time of program execution. This means that the same command does not stop the execution of the next part of the program, but the subsequent emergence of a specific event (for example a pressing of a button) will execute one of the defined actions:

PAUSE	Stop the execution of the program. Subsequent fulfillment of the condition of the special function resumes the operation of the program.
RESET	Execute the **Reset** command and start processing the program from the beginning.
OFF	Disable special function for the selected input.

Type of the event is selected using the **Select type** parameter. The following options are available here:

	Wait for rising edge	The controller detects the pressing of a button. If after the pressing the button remains pressed for a predetermined time then the controller executes the preset action (**Pause** or **Reset**).
	Wait for trailing edge	The controller detects the button release. If the button remains released pressed for a time set with the **Select time** parameter the controller proceeds to execute the preset action (**Pause** or **Reset**).
	Wait for any edge	The controller responds to the emergence of any edge at the signal at the input. If the button is pressed (rising edge) and remains pressed

		for a preset time then the controller proceeds to execute the preset action (**Pause** or **Reset**). If the button is released (trailing edge) and remains released for a preset time then the controller proceeds to execute the preset action (**Pause** or **Reset**).
	Wait for low level	If a low level is maintained for a preset time at the input of the controller (button released) then the controller proceeds to execute the preset action (**Pause** or **Reset**). Warning: Timing starts at the emergence of the **Input** command.
	Wait for high level	If a high level is maintained for a preset time at the input of the controller (button pressed) then the controller proceeds to execute the preset action (**Pause** or **Reset**). Warning: Timing starts at the emergence of the **Input** command.

The **Select time t** parameter defines the preset time for which the input must maintain a certain level for a command to be executed and for a controller to execute the next step of the program. The time of the delay can be adjusted from 0.0 (immediate reaction) to 24 hours.

Templates

Templates are pre-defined programs to implement a number of basic functions that are required of time controllers. These programs are presented in the form of a list containing a graphical diagram of operation and a brief description of each functionality (Fig. 19).

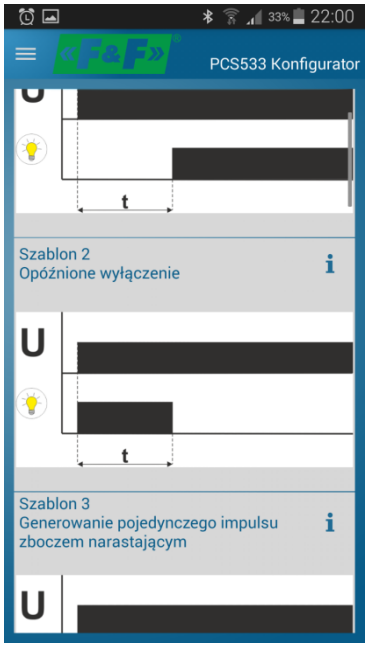


Fig. 19) Templates

Detailed information on the program carried out by the template can be displayed by pressing the „**i**" symbol that is located next to the description of the template (Fig. 20).

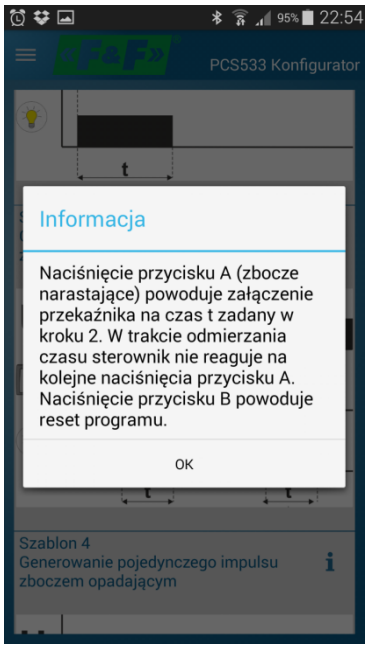
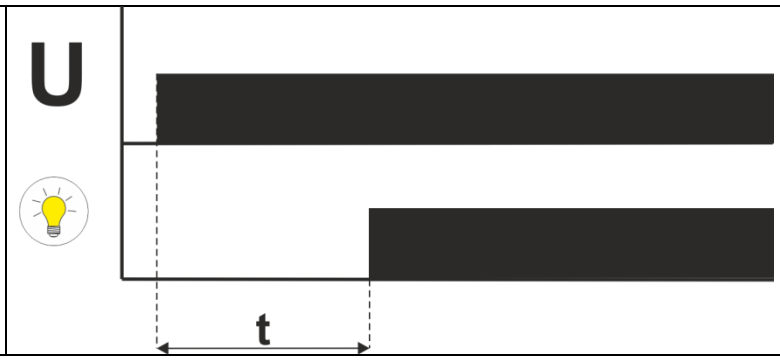


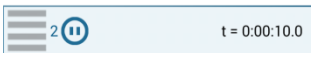
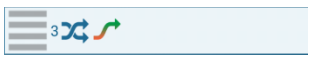
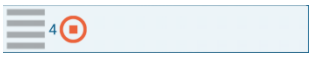
Fig. 20) Description of the exemplary template

After selecting a template, its content will be copied to the program editor from where it can be copied directly to the controller or it can be modified according to your needs.

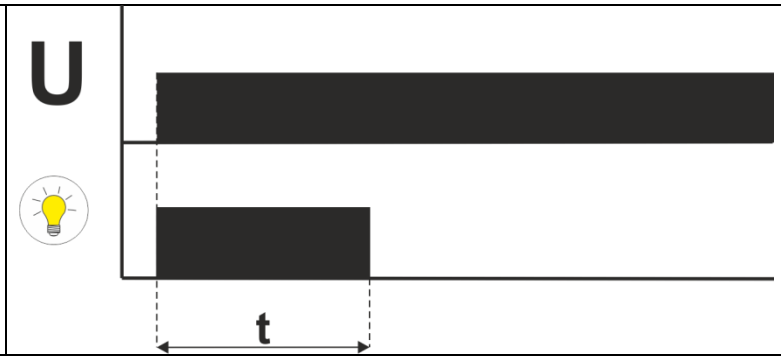
Template list

01) Delayed switching on	

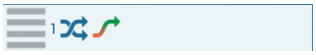
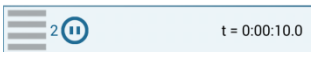
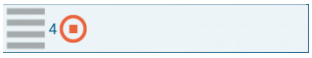
After switching on the power supply the controller measures the preset time t and then switches on the relay and stops the execution of the program. Pressing the button **B** resets the program.

	Command: **Input B – special function** Type: Rising edge Action: Reset Time: 0 s	The **Reset** function is assigned to the input B. Pressing at any time the button connected to input B will reset the controller and the program will start again from the beginning.
	Command: **Input** Type: Turn on Time: 0 s	Set the relay in the off position. Time 0 - the relay will remain off until the next **Output** command will change it. Warning: The command may be considered redundant in this case, because after the reset of the controller the relay is by default set in the off position.
	Command: **Pause** Time: 10 s	Pause suspends the execution of the rest of the program for a preset time (default - 10 s). After this time the next step of the program is executed. Warning: The time that is set here determines the delay of the switching on of the relay. If necessary, set the time to the desired value.
	Command: **Input** Type: Turn on Time: 0 s	Set the relay in the on position. Time is set to zero so the relay will remain on.
	Command: **Stop** Time: 0 s	Stop the execution of the program. The relay stays on because of the command from the previous step. Special function **Reset** that is assigned to button B is still active. Pressing the B button will reset the

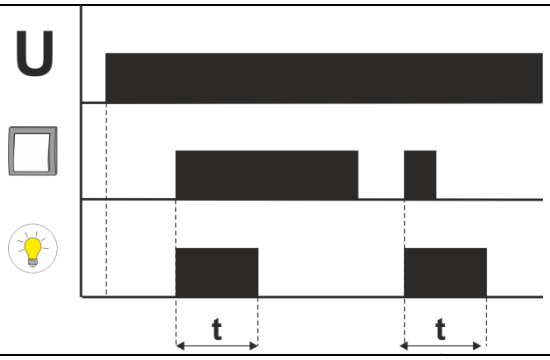
		controller and start the execution of the program from the beginning. Another way of resetting the controller is to turn off the power supply.

02) Delayed switching off	

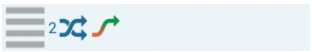
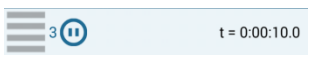
After turning on the power supply the controller switches on the relay. When the preset time t will elapse the relay will be switched off and the execution of the program will stop. Pressing button **B** resets the program.

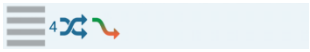
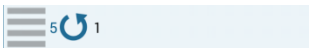
	Command: **Input B – special function** Type: Rising edge Action: Reset Time: 0 s	The **Reset** function is assigned to the input B. Pressing at any time the button connected to input B will reset the controller and the program will start again from the beginning.
	Command: **Output** Type: Turn on Time: 0 s	Set the relay in the on position. Time 0 - the relay will remain on until the next **Output** command will change it
	Command: **Pause** Time: 10 s	Pause suspends the execution of the rest of the program for a preset time (default - 10 s). After this time the next step of the program is executed. Warning: The time that is set here determines the delay of the switching off of the relay. If necessary, set the time to the desired value.
	Command: **Output** Type: Turn off Time: 0 s	Set the relay in the off position. Time is set to zero so the relay will remain off.
	Command: **Stop** Time: 0 s	Stop the execution of the program. The relay stays off because of the command from the previous step. Special function **Reset** that is assigned to button B is still active. Pressing the B button will reset the controller and start the execution of

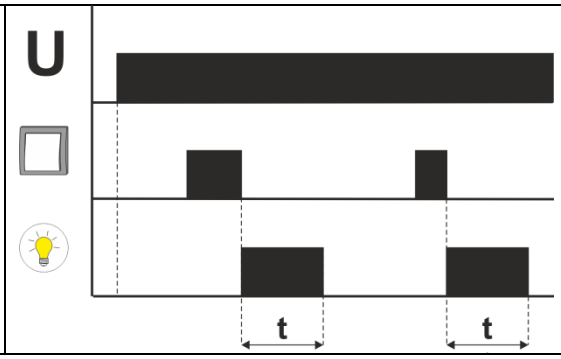
		the program from the beginning. Another way of resetting the controller is to turn off the power supply.

03) Generate a single pulse with rising edge	

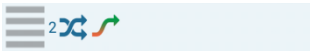
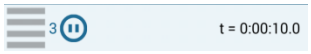
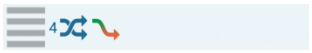
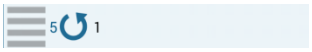
Pressing the **A** button (rising edge) will switch on the relay for a preset time **t**. During that time the relay does not react to the subsequent pressings of the button **A**. After the preset time elapses the relay switches off and the cycle starts from the beginning - the relay waits for the next pressing of the button **A**. Pressing the button **B** at any time interrupts the sequence and the relay returns to waiting for a pressing of the button **A**.

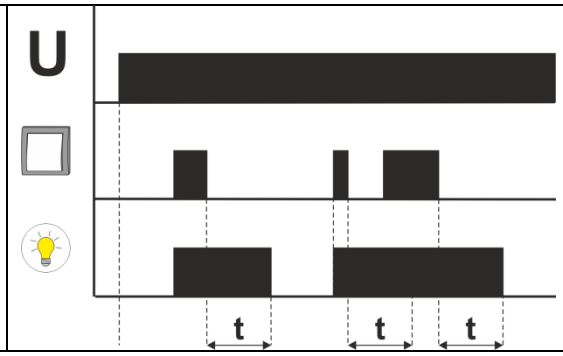
	Command: **Input B – special function** Type: Rising edge Action: Reset Time: 0 s	The **Reset** function is assigned to the input B. Pressing at any time the button connected to input B will reset the controller and the program will start again from the beginning.
	Command: **Input A** Type: Wait for a rising edge Time: 0 s	The relay waits for the pressing of a button A. The time is set to 0 seconds, so the relay will react directly to pressing. Up to this point the execution of the program is stopped.
	Command: **Output** Type: Turn on Time: 0 s	Set the relay in the on position. Time 0 - the relay will remain on until the next **Output** command will change it.
	Command: **Pause** Time: 10 s	Pause suspends the execution of the rest of the program for a preset time (default - 10 s). After this time the next step of the program is executed. Warning: The time that is set here determines the delay of the switching on of the relay. If necessary, set the time to the desired value.

	Command: **Output** Type: Turn off Time: 0 s	Set the relay in the off position. Time is set to zero so the relay will remain off.
	Command: **Return to** Select step: 1 Number of repetitions: 0	Execution of the program is transferred to step 1 - waiting for the pressing of a button A. The number of repetitions is set to 0 which means that this cycle can be repeated indefinitely. Warning: Despite the fact that the **Return to** command skipped the step 0 that establishes the special function for the input B, this command, once entered, remain active all the time.

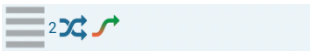
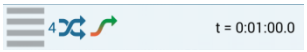
04) Generate a single pulse with trailing edge	

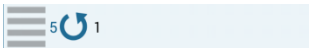
Releasing the **A** button (trailing edge) will switch on the relay for a preset time **t**. During that time the relay does not react to the subsequent pressing/releasing of the button **A**. After the preset time elapses the relay switches off and the cycle starts from the beginning - the relay waits for the next releasing of the button **A**. Pressing the button **B** at any time interrupts the sequence and the relay returns to waiting for a releasing of the button **A.**

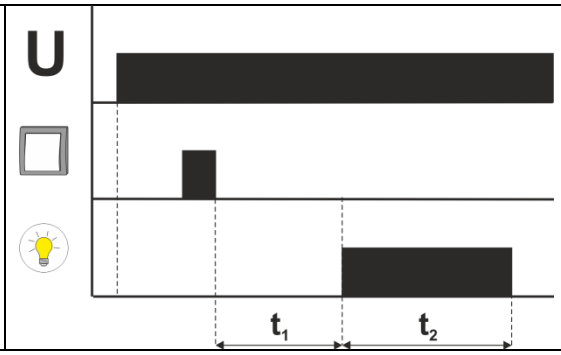
	Command: **Input B – special function** Type: Rising edge Action: Reset Time: 0 s	The **Reset** function is assigned to the input B. Pressing at any time the button connected to input B will reset the controller and the program will start again from the beginning.
	Command: **Input A** Type: Wait for a trailing edge Time: 0 s	The relay waits for the pressing of a button A. The time is set to 0 seconds, so the relay will react directly to pressing. Up to this point the execution of the program is stopped.
	Command: **Output** Type: Turn on Time: 0 s	Set the relay in the on position. Time 0 - the relay will remain on until the next **Output** command will change it.
	Command: **Pause** Time: 10 s	Pause suspends the execution of the rest of the program for a preset time (default - 10 s). After this time the next step of the program is executed. Warning: The time that is set here determines the delay of the switching on of the relay. If necessary, set the time to the desired value.
	Command: **Output** Type: Turn off Time: 0 s	Set the relay in the off position. Time is set to zero so the relay will remain off.
	Command: **Return to** Select step: 1 Number of repetitions: 0	Execution of the program is transferred to step 1 - waiting for the pressing of a button A. The number of repetitions is set to 0 which means that this cycle can be repeated indefinitely. Warning: Despite the fact that the **Return to** command skipped the step 0 that establishes the special function for the input B, this command, once entered, remain active all the time.

05) Switch-off delay with support function	

Pressing the **A** button will switch on the relay. During the release of the button the countdown of the preset time starts and then the relay will be switched off. If the button will be pressed again during the countdown the cycle will start from the beginning. Pressing button **B** at any time interrupts the sequence and the relay returns to waiting for a pressing of button **A**.

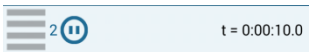
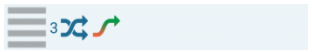
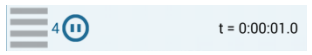
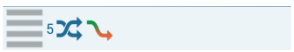
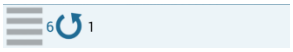
	Command: **Input B – special function** Type: Rising edge Action: Reset Time: 0 s	The **Reset** function is assigned to the input B. Pressing at any time the button connected to input B will reset the controller and the program will start again from the beginning.
	Command: **Input A** Type: Wait for a rising edge Time: 0 s	The relay waits for the pressing of a button A. The time is set to 0 seconds, so the relay will react directly to pressing. Up to this point the execution of the program is stopped.
	Command: **Output** Type: Turn on Time: 0 s	Set the relay in the on position. Time 0 - the relay will remain on until the next **Output** command will change it.
	Command: **Input A** Type: Wait for a trailing edge Time: 0 s	The relay waits for the releasing of a button A. Up to this point the execution of the program is stopped and the relay that was switched-on in the previous step will remain on.
	Command: **Output** Type: Turn on Time: 1 min	Set the relay in the on position for 1 minute. Warning: Setting the time in this step does not stop the execution of the program, which means that the program will return to the next waiting for a pressing of a button A. If the button will not be pressed during the preset time the relay will switch off.

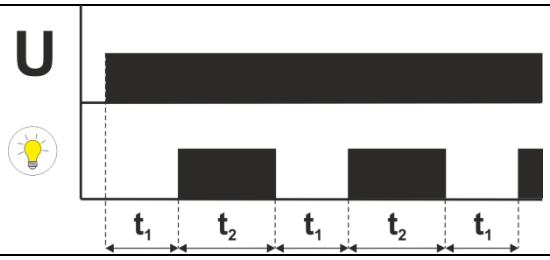
	Command: **Return to** Select step: 1 Number of repetitions: 0	Execution of the program is transferred to step 1 - waiting for the pressing of a button A. The number of repetitions is set to 0 which means that this cycle can be repeated indefinitely. Warning: Despite the fact that the **Return to** command skipped the step 0 that establishes the special function for the input B, this command, once entered, remain active all the time.

06) Generate a single pulse after a preset time	

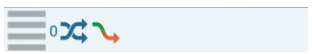
The trailing edge at the output **A** (button release) starts counting down time t_1. After that time the relay switches on for a time t_2. During the $t_1 - t_2$ cycle the subsequent pressings of a button **A** will be ignored. Pressing button **B** at any time interrupts the sequence and the relay returns to waiting for a pressing of button **A**.

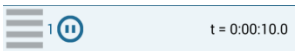
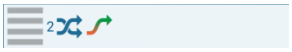
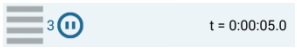
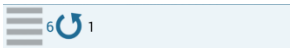
	Command: **Input B – special function** Type: Rising edge Action: Reset Time: 0 s	The **Reset** function is assigned to the input B. Pressing at any time the button connected to input B will reset the controller and the program will start again from the beginning.
	Command: **Input A** Type: Wait for a trailing edge Time: 0 s	The relay waits for the pressing of a button A. The time is set to 0 seconds, so the relay will react directly to pressing. Up to this point the execution of the program is stopped.

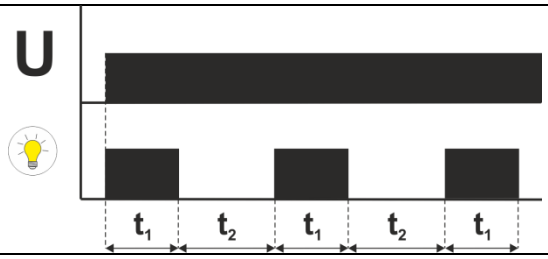
	Command: **Pause** Time: 10 s	Pause suspends the execution of the rest of the program for a preset time (default - 10 s). After this time the next step of the program is executed.
	Command: **Output** Type: Turn on Time: 0 s	Set the relay in the on position. Time 0 - the relay will remain on until the next **Output** command will change it.
	Command: **Pause** Time: 1 min	Pause suspends the execution of the rest of the program for a preset time (default - 1 minute). After this time the next step of the program is executed.
	Command: **Output** Type: Turn off Time: 0 s	Set the relay in the off position. Time 0 - the relay will remain off until the next **Output** command will change it.
	Command: **Return to** Select step: 1 Number of repetitions: 0	Execution of the program is transferred to step 1 - waiting for the pressing of a button A. The number of repetitions is set to 0 which means that this cycle can be repeated indefinitely. Warning: Despite the fact that the **Return to** command skipped the step 0 that establishes the special function for the input B, this command, once entered, remain active all the time.

07) Cyclic delayed switching on	

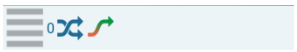
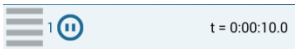
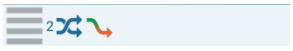
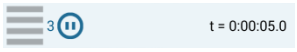
After switching on the power supply the program starts to cyclically execute the following operations: turn off the relay for a time $\mathbf{t_1}$, turn on the relay for a time $\mathbf{t_2}$.

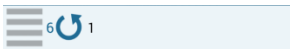
	Command: **Output** Type: Turn off Time: 0 s	Set the relay in the off position. Time 0 - the relay will remain off until the next **Output** command will change it.

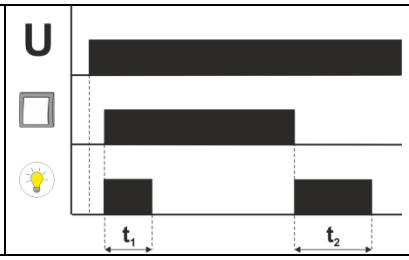
	Command: **Pause** Time: 10 s	Pause t_1 suspends the execution of the rest of the program for a preset time (default - 10 s). After this time the next step of the program is executed.
	Command: **Output** Type: Turn on Time: 0 s	Set the relay in the on position. Time 0 - the relay will remain on until the next **Output** command will change it.
	Command: **Pause** Time: 5 s	Pause t_2 suspends the execution of the rest of the program for a preset time (default - 5 seconds). After this time the next step of the program is executed.
	Command: **Return to** Select step: 1 Number of repetitions: 0	Execution of the program is transferred to step 1 - waiting for the pressing of a button A. The number of repetitions is set to 0 which means that this cycle can be repeated indefinitely.

08) Cyclic delayed switching off	

After switching on the power supply the program starts to cyclically execute the following operations: turn on the relay for a time t_1, turn off the relay for a time t_2.

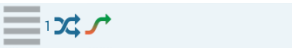
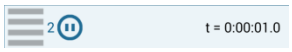
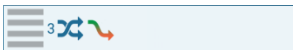
	Command: **Output** Type: Turn on Time: 0 s	Set the relay in the on position. Time 0 - the relay will remain on until the next **Output** command will change it.
	Command: **Pause** Time: 10 s	Pause t_1 suspends the execution of the rest of the program for a preset time (default - 10 s). After this time the next step of the program is executed.
	Command: **Output** Type: Turn off Time: 0 s	Set the relay in the off position. Time 0 - the relay will remain off until the next **Output** command will change it.
	Command: **Pause** Time: 5 s	Pause t_2 suspends the execution of the rest of the program for a preset time (default - 5 seconds). After this time the next step of the program is executed.

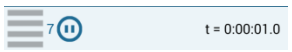
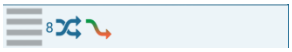
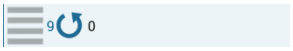
	Command: **Return to** Select step: 1 Number of repetitions: 0	Execution of the program is transferred to step 1 - waiting for the pressing of a button A. The number of repetitions is set to 0 which means that this cycle can be repeated indefinitely.

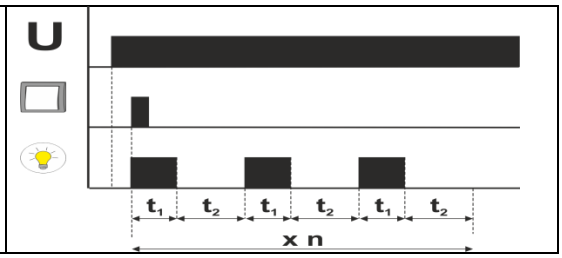
09) Switching on of the contact for the time t1 with the button pressed and for the time t2 with the button released	

Pressing the **A** button (rising edge) will switch on the relay for the time $\mathbf{t_1}$. Releasing the button (trailing edge) will switch on the relay for the time $\mathbf{t_2}$. If the trailing edge occurs before the end of the $\mathbf{t_1}$ timing, then the switching on for the time $\mathbf{t_1}$ will not occur. If the subsequent pressing will take place during the timing of time $\mathbf{t_1}$, then the whole next cycle will be skipped.

.

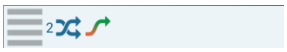
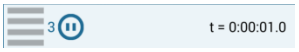
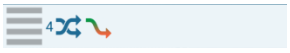
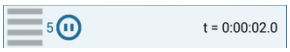
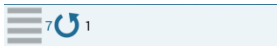
	Command: **Input A** Type: Wait for a rising edge Time: 0 s	The relay waits for the pressing of a button A. The time is set to 0 seconds, so the relay will react directly to pressing. Up to this point the execution of the program is stopped.
	Command: **Output** Type: Turn on Time: 0 s	Set the relay in the on position. Time 0 - the relay will remain on until the next **Output** command will change it.
	Command: **Pause** Time: 1 s	Pause $\mathbf{t_1}$ suspends the execution of the rest of the program for a preset time (default - 1 s). After this time the next step of the program is executed.
	Command: **Output** Type: Turn off Time: 0 s	Set the relay in the off position. Time 0 - the relay will remain off until the next **Output** command will change it.
	Command: **Input A – special function** Type: Low level Action: Reset Time: 0.1 s	This step will be responsible for breaking the cycle in case the button is released before time $\mathbf{t_1}$ elapses. If the low level at the output **A** will sustain for 0.1 s the program will be restarted and its execution will start from the beginning.

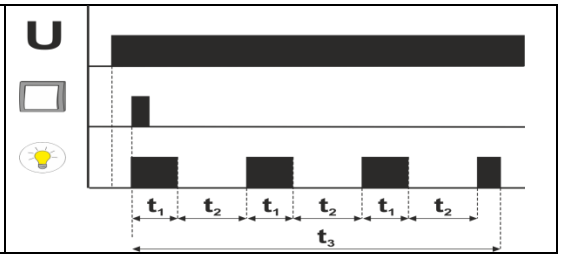
	Command: **Input A** Type: Wait for a trailing edge Time: 0 s	If a high level is still present at the output A then the program awaits for the release of a button A. Up to this point the execution of the program is stopped.
	Command: **Output** Type: Turn on Time: 0 s	Set the relay in the on position. Time 0 - the relay will remain on until the next **Output** command will change it.
	Command: **Pause** Time: 2 s	Pause t_2 suspends the execution of the rest of the program for a preset time (default - 2 s). After this time the next step of the program is executed.
	Command: **Output** Type: Turn off Time: 0 s	Set the relay in the off position. Time 0 - the relay will remain off until the next **Output** command will change it.
	Command: **Return to** Select step: 0 Number of repetitions: 0	Execution of the program is transferred to step 0 - waiting for the pressing of a button **A**. The number of repetitions is set to 0 which means that this cycle can be repeated indefinitely.

10) Generate a preset number of pulses	

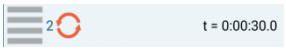
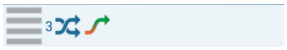
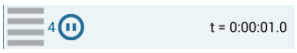
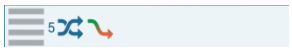
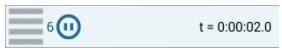
Pressing the **A** button (rising edge) generates the n-chain of pulses. We have defined the time of the switching on of the relay and the interval between them. Pressing the **B** button will reset the controller and start the execution of the program from the beginning.

	Command: **Input B – special function** Type: Rising edge Action: Reset Time: 0 s	The **Reset** function is assigned to the input B. Pressing at any time the button connected to input B will reset the controller and the program will start again from the beginning.
	Command: **Input A** Type: Wait for a trailing edge Time: 0 s	The relay waits for the pressing of a button A. The time is set to 0 seconds, so the relay will react directly to pressing. Up to this point the execution of the program is stopped.

	Command: **Output** Type: Turn on Time: 0 s	Set the relay in the on position. Time 0 - the relay will remain on until the next **Output** command will change it.
	Command: **Pause** Time: 1 s	Pause t_1 suspends the execution of the rest of the program for a preset time (default - 1 s). After this time the next step of the program is executed.
	Command: **Output** Type: Turn off Time: 0 s	Set the relay in the off position. Time 0 - the relay will remain off until the next **Output** command will change it.
	Command: **Pause** Time: 2 s	Pause t_2 suspends the execution of the rest of the program for a preset time (default - 2 s). After this time the next step of the program is executed.
	Command: **Return to** Select step: 2 Number of repetitions: 3	Execution of the program is transferred to step 2, which means that the next cycle of the switching on of the relay is starting. The number of repetitions is set to 3 which means that the relay will switch on and off three times and then the program will proceed to the next step.
	Command: **Return to** Select step: 1 Number of repetitions: 0	Execution of the program is transferred to step 1 - waiting for the pressing of a button A. The number of repetitions is set to 0 which means that this cycle can be repeated indefinitely.

11) Generate pulses for a preset period of time	

Pressing the **A** button (rising edge) generates the chain of pulses where the relay is on for the time t_1 and off for the time t_2. This whole cycle is repeated for the time t_3 and then the program starts from the beginning and waits for the next pressing of the button **A**. Pressing the B button will reset the controller and start the execution of the program from the beginning.

	Command: **Input B – special function** Type: Rising edge Action: Reset Time: 0 s	The **Reset** function is assigned to the input B. Pressing at any time the button connected to input B will reset the controller and the program will start again from the beginning.
	Command: **Input A** Type: Wait for a rising edge Time: 0 s	The relay waits for the pressing of a button A. The time is set to 0 seconds, so the relay will react directly to pressing. Up to this point the execution of the program is stopped.
	Command: **Reset** Time: 30 s	Setting the Reset command causes that after 30 seconds (time t_3) from executing that command the controller will be reset and the execution of the program will start from the beginning.
	Command: **Output** Type: Turn on Time: 0 s	Set the relay in the on position. Time 0 - the relay will remain on until the next **Output** command will change it.
	Command: **Pause** Time: 1 s	Pause t_1 suspends the execution of the rest of the program for a preset time (default - 1 s). After this time the next step of the program is executed.
	Command: **Output** Type: Turn off Time: 0 s	Set the relay in the off position. Time 0 - the relay will remain off until the next **Output** command will change it.
	Command: **Pause** Time: 2 s	Pause t_2 suspends the execution of the rest of the program for a preset time (default - 2 s). After this time the next step of the program is executed.
	Command: **Return to** Select step: 3 Number of repetitions: 0	Execution of the program is transferred to step 3, which means that the next cycle of the switching on of the relay is starting. The number of repetitions is set to 0 which means that the cycle will repeat itself until elapses the time of the reset t_3 that was defined in step 2.

History of changes

Warranty

1. The relay comes with a 24 month warranty. The term of this warranty begins on the date of purchase of the product.
2. The warranty is valid only with a proof of purchase.
3. The notification of the complaint must be made at the place of purchase or directly at the manufacturer:

F&F Filipowski sp. j.
ul. Konstantynowska 79/81
95-200 Pabianice
Tel. (42) 227-09 71
e-mail: dztech@fif.com.pl

4. Written information about the nature of the fault and the circumstances of its occurrence must be attached to the notification of the complaint.
5. F&F Filipowski sp. j. commits itself to review the complaints in accordance with Polish law.
6. The choice of the form of settling the customer complaint: replacement of the product for the product free from defects, repair or refund belongs to the manufacturer.
7. Warranty does not cover:
    a. Mechanical and chemical damages;
    b. Damages resulting from improper use or the use inconsistent with the user manual;
    c. Damages incurred after the sale as a result of accidents or other events for which nor the producer, nor the place of sale are responsible, for example damages in transit.
8. Warranty does not cover actions that user should perform in accordance with the user manual, for example installing multi-meter, building electrical installation, installing other required electrical protection.
9. Warranty does not limit the buyer's rights arising from the nonconformity of goods with the contract.