

Programmable logic with ProfiLux II and 3

1 Preface

The aquarium and terrarium computers of the ProfiLux II and 3 series allow (beginning with firmware 4.00) the creation of a so-called programmable logic.

The programmable logic allows the logical or time-dependent interpretation of the switching state of sockets, the result of this interpretation can in turn serve as a switching signal for a socket.

To be able to use the programmable logic of the ProfiLux, it is necessary to define first of all some terms.

In the Boolean switching algebra, there are only 2 states:

ON (is the same as TRUE, WAHR, 1 or HIGH)

OFF (is the same as FALSE, FALSCH, 0 or LOW)

In the ProfiLux these two states correspond to the switching states of the sockets.

A logical operation (or function) has one or more input states and one output state. The output state is determined by the input states and the operator. The operator determines how the input states shall be processed (linked).

2 Logical functions

A logical function is determined by the interrelationship input states to output state. Usually, this is made with logic tables.

2.1 NOT-Function

The NOT-function needs exactly one input. The output behaves always inversely to the input.

Input	Output
0	1
1	0

2.2 AND-Function

The AND-function needs 2 inputs. The output is only ON if both inputs are ON.

Input 1	Input 2	Output
0	0	0
1	0	0
0	1	0
1	1	1

2.3 OR-Function

The OR-function needs 2 inputs. The output is only ON if at least one input is ON.

Input 1	Input 2	Output
0	0	0
1	0	1
0	1	1
1	1	1

2.4 NAND-Function

The NAND-function needs 2 inputs. The output is only OFF if both inputs are ON.

Input 1	Input 2	Output
0	0	1
1	0	1
0	1	1
1	1	0

2.5 NOR-Function

The NOR-function needs 2 inputs. The output is only ON if both inputs are OFF.

Input 1	Input 2	Output
0	0	1
1	0	0
0	1	0
1	1	0

2.6 Equal-Function

The EQUAL-function needs 2 inputs. The output is then ON if both inputs have the same state.

Input 1	Input 2	Output
0	0	1
1	0	0
0	1	0
1	1	1

2.7 Unequal-Function

The UNEQUAL-function needs 2 inputs. The output is then ON if both inputs have a different state.

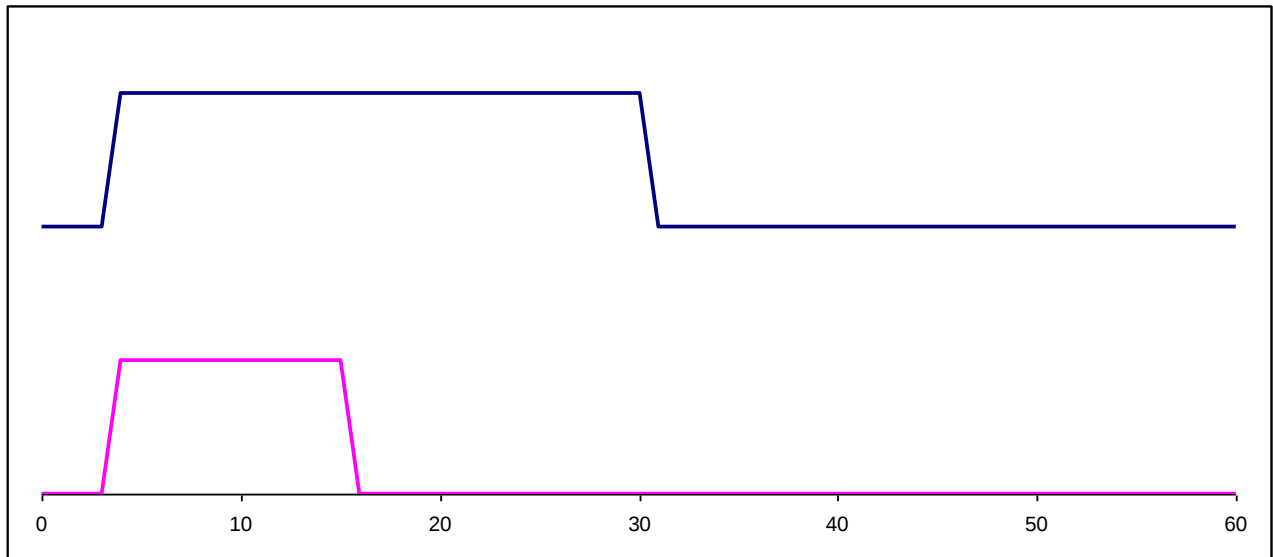
Input 1	Input 2	Output
0	0	0
1	0	1
0	1	1
1	1	0

3 Time functions

Besides the simple logical functions explained before, ProfiLux offers also time functions. All time functions have only one input. The output is switched dependently on the input state, the time function and the set time. These time functions are exemplified with time diagrams. Times between 1s and 28800s (= 8h) can be set.

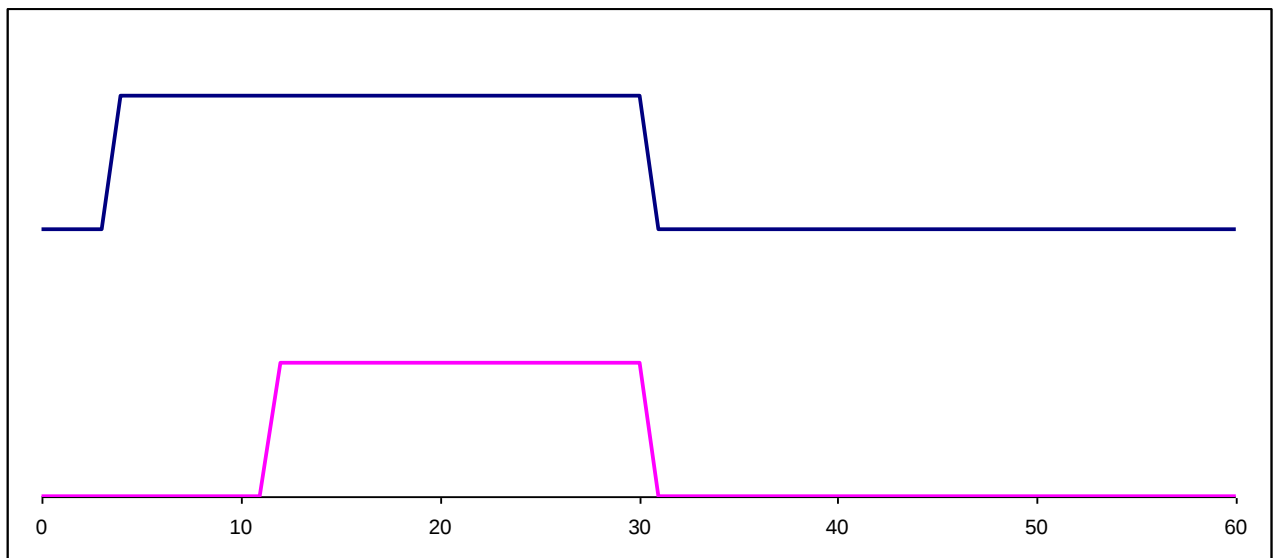
3.1 Pulse-Function

If the input changes from OFF to ON, the output is switched on. After the set time has expired (here in the example 12s), the output is switched off again, independent from the state of the input.



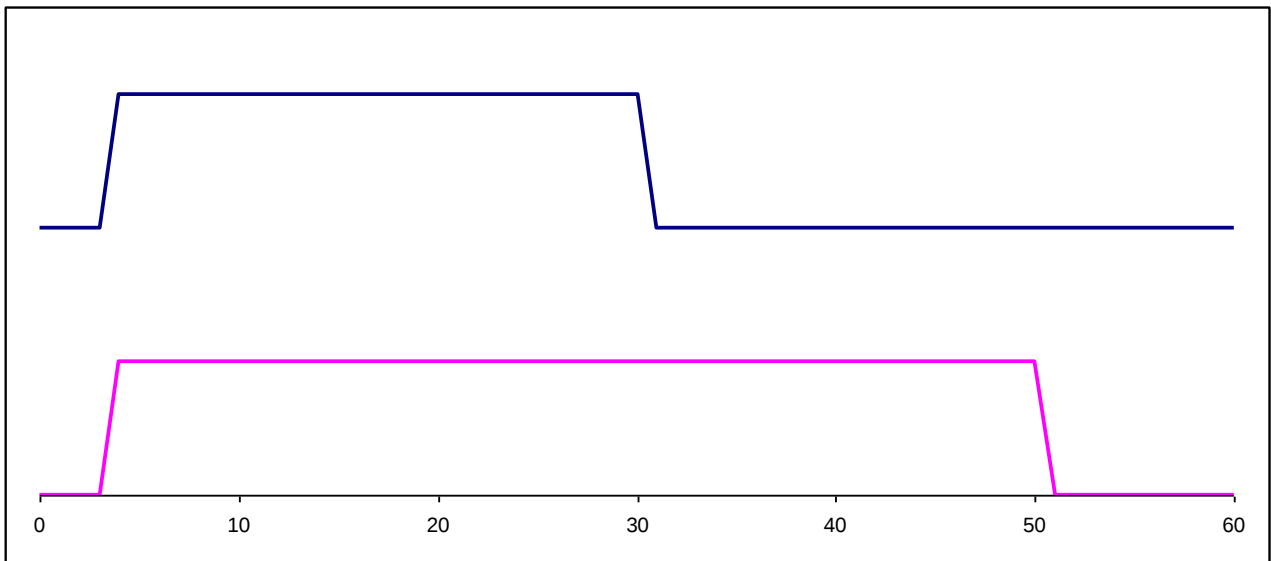
3.2 Delayed-On-Function

During switch-on, the output follows the input with the set delay time (here in the example 9s), during switch-off of the input, the output follows immediately.



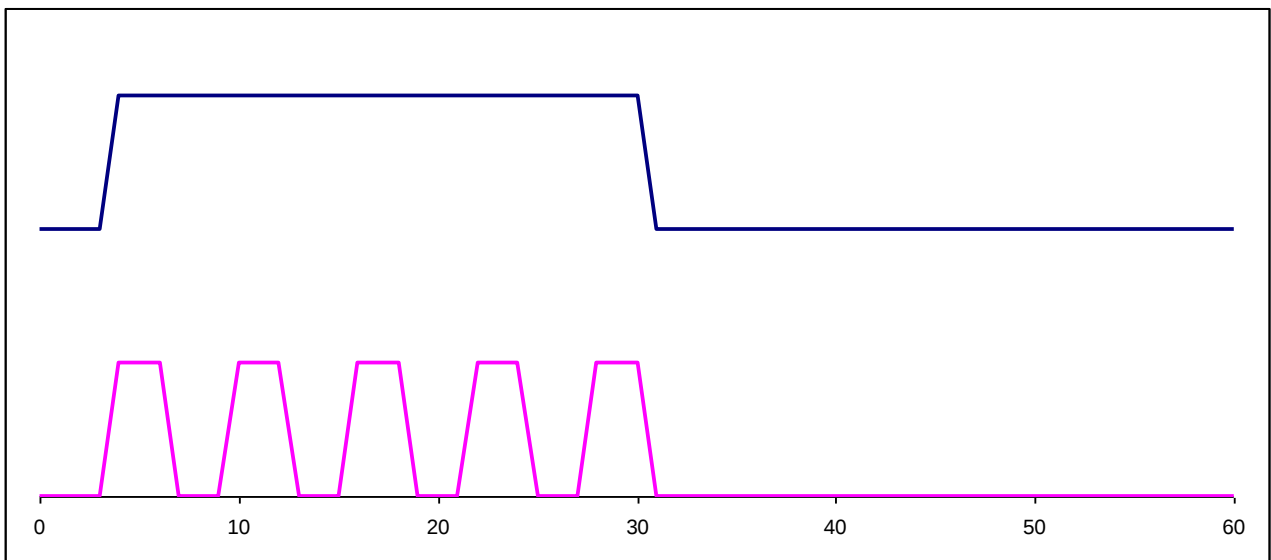
3.3 Delayed-Off-Function

The output follows the input immediately during switch-on, when the input is switched off, the output follows with the set delay time (here in the example 20s).



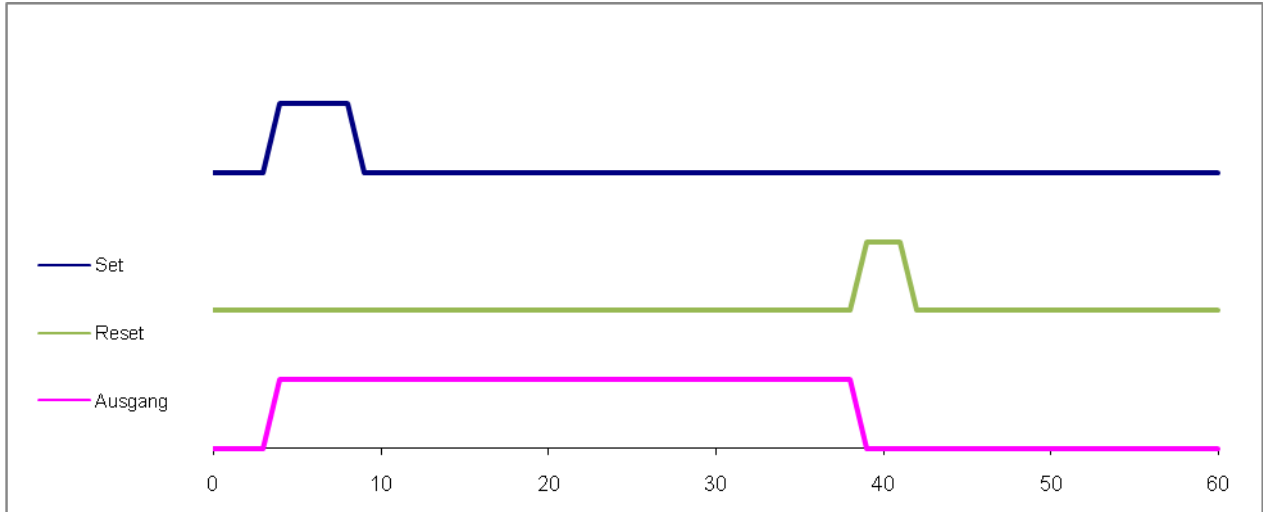
3.4 Cyclic-Function

As long as the input is switched on, the output switches alternately ON and OFF, the switching duration is determined by the set time (here in the example 3s). If the input is OFF, then also the output is OFF.



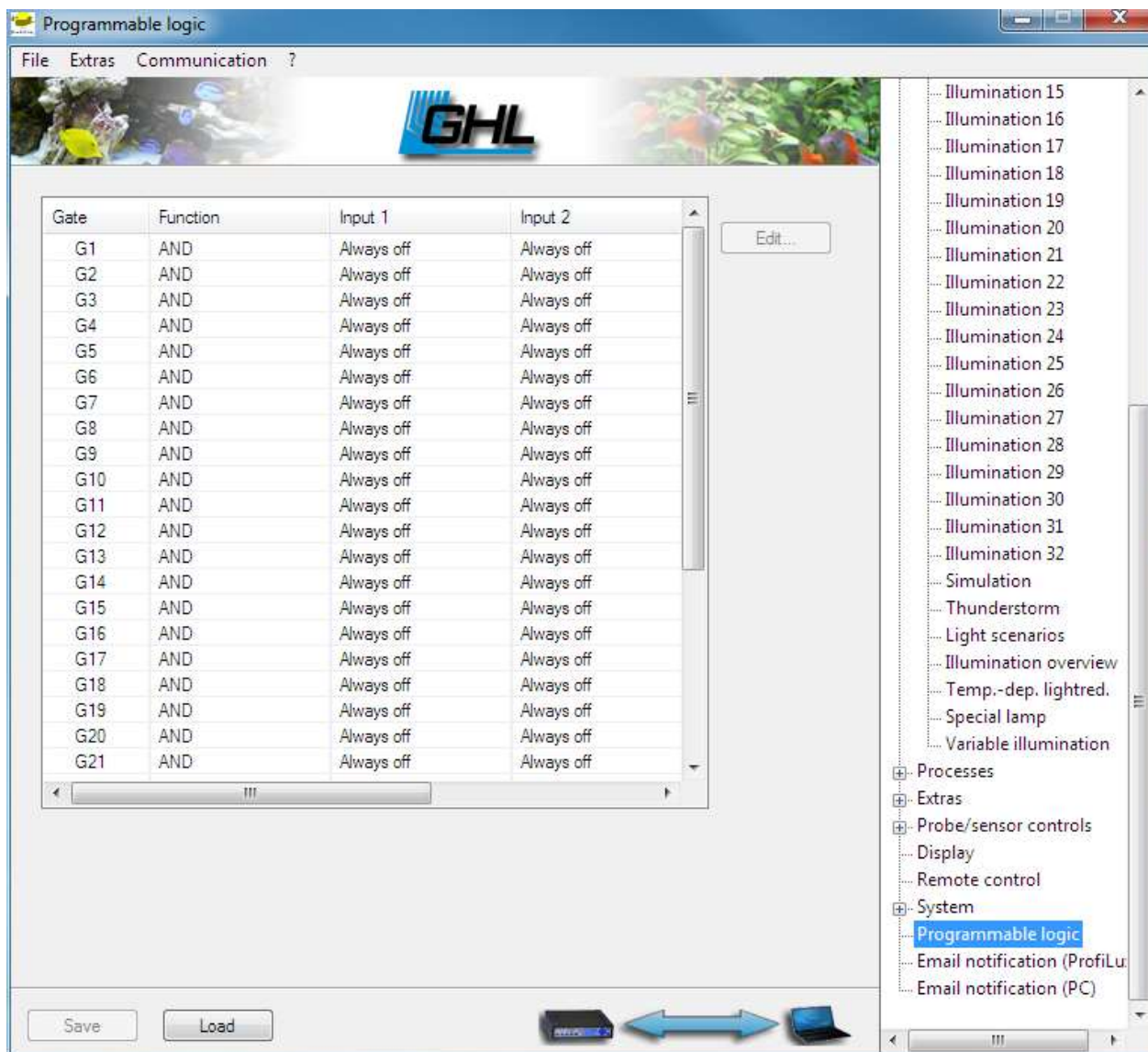
4 SR-Flipflop

This is a special function. The specialty is that the recent output state persists as long as both inputs are OFF. The output is switched ON if the Set-input is ON. Also if the Set-input is again OFF shortly afterwards, the output remains ON and that until the Reset-Input is ON, then the output is OFF again.

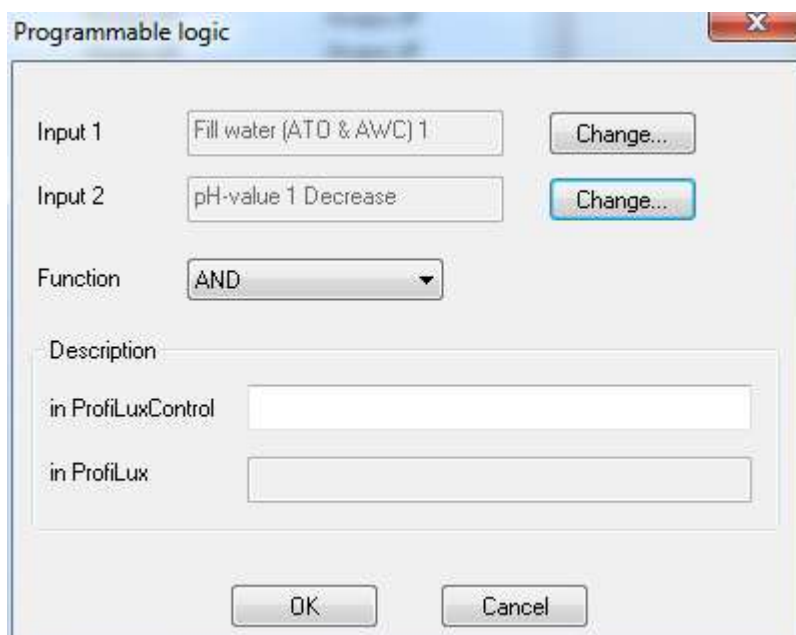


5 Use of the logical functions

In the menu *Programmable Logic* you can define up to 8 so-called *Gates*. A gate has one or 2 inputs and one output. Furthermore, for some functions (e.g. Delayed-off), the gates have also a settable time available.



Select a gate in the table and click on *Edit*.



Set first the desired logical function (here in the example AND), afterwards you can set the inputs resp. the times.

Here in the example the following function has been set for programmable logic x (x is a number from 1 to 8): *Filter AND Timer 3*

Programmable logic x is thus then active if *filter* and at the same time *timer 3* are active.

As a last step, set now the function *Programmable logic x* for the desired socket.