

# Flashing Rate and Clock Generator in SIPROTEC 4 Relays

## Implementation in the CFC

### Foreword

In many cases, the customer requires flashing LEDs, for example, to draw the operator's attention to a particular status. This feature is mostly necessitated by safety-critical requirements. For example, it can be used to indicate a change from a safe to a critical state. This change could be caused by the operator:

- Keyswitch turned to LOCAL
- Control command output

or also by the process:

- Measured values are at limit value
- Protection function has been blocked

Applications are also possible in which a clock generator (analog flashing rate generator) outputs a pulse at regular intervals that can be used, for example, to synchronize other relays.

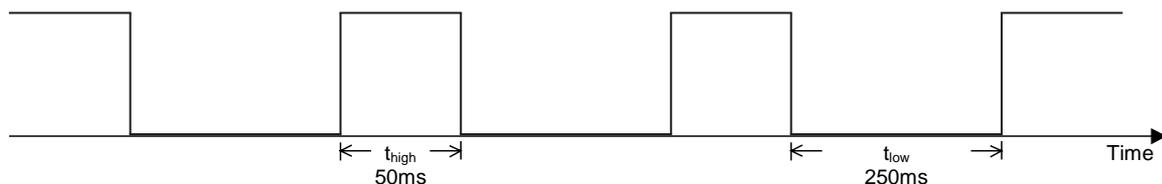
In the SIPROTEC 4 relays, no such flashing mechanism is available. The LEDs and binary outputs can only be routed to "Latched" or "Unlatched", but not to "Flashing".

However, this missing functionality can be compensated for using the programmable logic, a highlight feature of the SIPROTEC 4 family. This paper describes two solutions; you must use one or the other depending on the type of triggering signal:

- Short pulse (fleeting indication): e.g. pressing a function key on the relay
- Signal pending for a long time: e.g. "keyswitch to LOCAL" or "measured values above limit value"

### CFC and timers

The flashing mechanism uses a signal that switches from "low" to "high" at regular intervals:



The time block, which is the only block that processes the time, can be used in this case. Two are available for this with device firmware 4.30 and higher. But for typical flashing the normal timer block is sufficient. It actually consists of two timers.



#### Inputs:

With **S**, the "timer 1" is started and runs until the time parameterized in **T1** has elapsed. If the starting signal **S** is reset, "timer 2", whose runtime is set by **T2**, is started. On a signal to **R**, all running timers are stopped.

#### Outputs:

As long as "timer 1" is running, **QT1** is "high". The same goes for "timer 2" and **QT2**.<sup>1</sup>

<sup>1</sup> As a logic combination ( $Q=(S+T2) * T1 * R$ ), **Q** is not required very often and is not relevant to this task.

## CFC and priority class

As is stated in the help for the block, the timer only runs in certain priority classes. Why?

There are 4 priority classes:

### 1. Fast PLC (PLC\_Bearb)

- Priority higher than the protection function
- Event-controlled: Start of the class on change in input signal
- Suitable for blocking protection functions
- No more than about 12 blocks can be placed

### 2. Slow PLC (PLC1\_Bearb)

- Priority lower than the protection function
- Event-controlled: Start of the class on change in input signal
- Preferred class for event-controlled applications (e.g. flashing mechanism)
- No more than about 30 blocks can be placed

### 3. Switchgear interlocking (SFS\_Bearb)

- Priority lower than the protection function
- Event-controlled: Start of the class on output of a (switching) command
- Suitable for interlock conditions
- No more than about 100 blocks can be placed

### 4. Measured value processing (MW\_Bearb)

- Priority lower than the protection function
- Cyclically controlled: Start of the class at regular intervals or if the processor is not off-loaded
- Suitable for processing measured values (e.g. limit value comparison, implementation of protection functions such as "Reverse Power ANSI 32" or "Power Factor ANSI 55")
- No more than about 100 blocks can be placed

Because the timer can be started at any point in time, it can only be placed in the PLC classes. In the other levels, its execution would not be guaranteed.

The CFC chart that must be created for the flashing mechanism must therefore be set accordingly:

- 1 Create a new CFC chart.
- 2 Open that CFC chart.
- 3 Select in the menu "Edit" the entry "Run Sequence..." (a window "Run Sequence" opens).
- 4 Select "PLC1\_Bearb" (i.e. the slow PLC class).
- 5 Select the entry "Predecessor for Installation" in menu "Edit" of this window.
- 6 Close this window.

The new CFC chart will now definitely run in the correct priority class.

