Pulse counters

Project for PR200-24.2

Project overview

The example explains the use of different pulse counters. The project contains 1 data processing block and 1 screen.

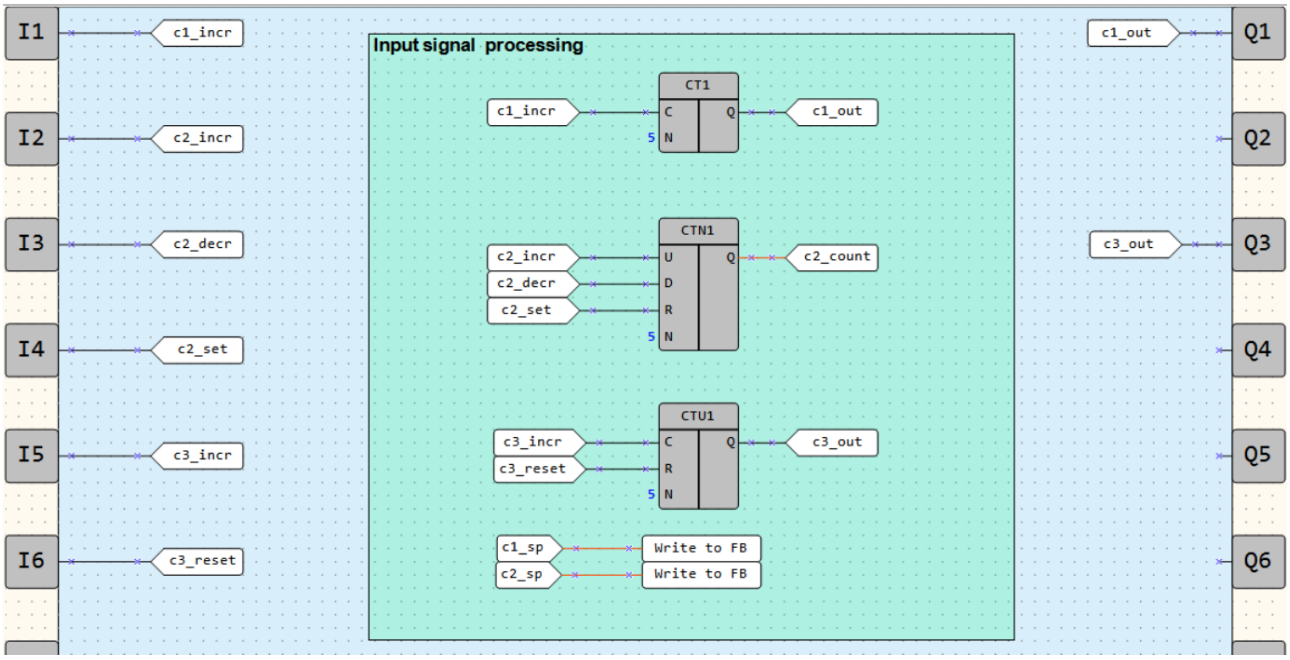

Fig. 1. Program workspace

Data processing block:

- Input signal processing

Table 1. Device inputs/outputs

Name	Type	Description
I1	BOOL	1st counter / incremental count (count up)
I2	BOOL	2nd counter / incremental count (count up)
I3	BOOL	2nd counter / decremental count (count down)
I4	BOOL	2nd counter / reset
I5	BOOL	3rd counter / incremental count (count up)
I6	BOOL	3rd counter / reset
Q1	BOOL	1st output
Q3	BOOL	3rd output

Table 2. Project variables

Name	Type	Description
c1_incr	BOOL	1st counter / incremental count (count up)
c1_sp	INT	1st counter / setpoint
c1_out	BOOL	1st counter / output

c2_incr	BOOL	2nd counter / incremental count (count up)
c2_decr	BOOL	2nd counter / decremental count (count down)
c2_sp	INT	2nd counter / setpoint
c2_set	BOOL	2nd counter / set to setpoint
c2_count	INT	2nd counter / count result
c3_incr	BOOL	3rd counter / incremental count (count up)
c3_reset	BOOL	3rd counter / reset to zero
c3_out	BOOL	3st counter / output

There are three types of counters used in the project:

- CT – threshold counter with self-reset
- CTN – universal counter
- CTU – threshold counter

The CT counter (Fig. 2) counts the pulses until the counted value is equal to setpoint *N* specified in the Property Box. The output *Q* is then set to *TRUE*, the counter is reset and starts counting from zero.

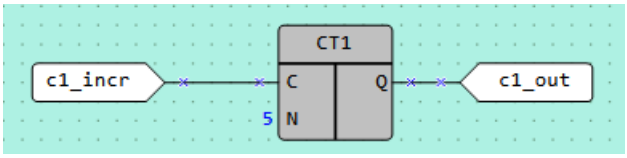

Fig. 2. CT counter

The *CTN* counter (Fig. 3) provides counting in both positive and negative directions. The setpoint *N* is also specified in the Property Box, but the output *Q* returns the number of counted pulses.

If the variable *c2_set = TRUE* is applied to the input *R*, the counter output value *Q* becomes equal to the setpoint value on the input *N*. This feature is required to use countdown or count up from a fixed value.

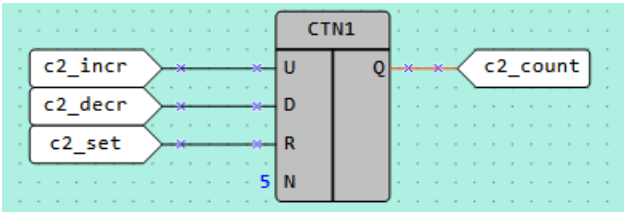

Fig. 3. CTN counter

The *CTU* counter counts pulses up to the setpoint on the input *N*, then the output *Q* becomes *TRUE* and remains until *TRUE* is received on the input *R* and the counting starts from zero.

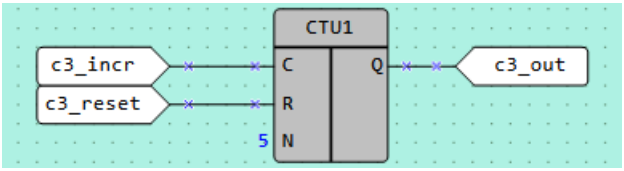

Fig. 4. CTU counter

For the *CT* and **CTN** counters, the setpoint can also be set via *WriteToFB* blocks, i.e. via variables which, for instance, can be written using the display and the function buttons.

Fig. 5. Write setpoints to counters

Screen

Table 3. Function buttons

Function buttons	Action
⩛	Scroll down through screen rows
⩚	Scroll up through screen rows

The first screen row shows the number of counted pulses of the 2nd counter (*CTN*). The 2nd and 3rd rows show the setpoints for the 1st and the 2nd counter respectively.

Initially the first two rows of the first screen are displayed (Fig. 6).

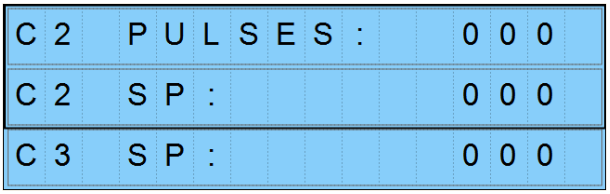

Fig. 6. Screen