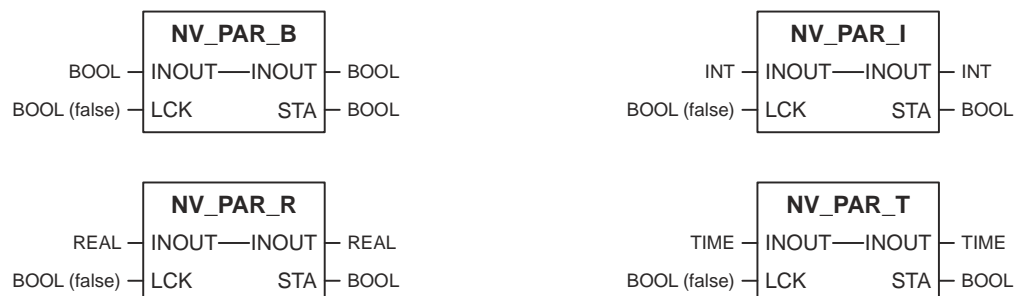


NV_PAR – zapis do pamięci nieulotnej (FLASH)

Blok zapisujący w pamięci nieulotnej (FLASH) zmienioną wartość.

Interfejs



```
FUNCTION_BLOCK NV_PAR_T
(*$COMMENT Zapis do pamięci nieulotnej (FLASH) wartości typu TIME*)
VAR_IN_OUT
    INOUT (*$COMMENT Zmienna do zapisu/odczytu*): TIME;
END_VAR
VAR_INPUT
    LCK (*$COMMENT Blokada zapisu*): BOOL:=FALSE;
END_VAR
VAR_OUTPUT
    STA (*$COMMENT Status *): BOOL;
END_VAR
```

Opis wejść i wyjść

INOUT – wejście-wyjście bloku, zmiana wartości aktualizuje zapis w FLASH
LCK – blokada zapisu wartości INOUT do pamięci FLASH
STA – status (FALSE =OK., TRUE=błąd zapisu/odczytu)

Działanie bloku

Blok monitoruje wejście INOUT i w przypadku zmiany wartości wykonuje zapis do pamięci FLASH. Zapis nie jest realizowany, gdy wejście LCK jest ustawione na TRUE (blokada zapisu). Aktualna wartość zapisana w pamięci FLASH jest zawsze dostępna na wyjściu INOUT (przy sprawnej pamięci). Ponieważ INOUT jest typu IN_OUT (wejście-wyjście), zmienna podstawiana na wejście bloku jest każdorazowo aktualizowana po wykonaniu obliczeń. Jeżeli zapis do sprawnej pamięci FLASH nie jest możliwy z powodu aktywnej blokady, wtedy zmienna podawana na wejście INOUT jest nadpisywana wartością zapamiętaną w pamięci FLASH (zmienna „dołączona” do INOUT przyjmuje wartość z FLASH). W przypadku wystąpienia błędu zapisu/odczytu (np. uszkodzona pamięć lub brak miejsca w pamięci) wyjście STA przyjmuje wartość TRUE. Każdorazowo przy wpisaniu programu do sterownika pamięć FLASH jest inicjowana domyślną wartością zmiennej przypisanej do wejścia INOUT.

Jeżeli wartość zmiennej wejściowej bloku jest ustawiana z oprogramowania SCADA, wówczas może zająć potrzeba zablokowania zapisu do pamięci FLASH (LCK:=TRUE) na czas zsynchronizowania wartości zmiennych sterownika i oprogramowania SCADA (odczyt przez SCADA wartości zapisanych w FLASH). Zablokowanie zapisu może być również stosowane dla uniknięcia przypadkowych

zapisów wywołanych błędami operatora (wprowadzanie zmian wyłącznie w określonym trybie pracy).

Uwaga!

Zapis do pamięci nieulotnej po wywołaniu bloku nie jest dokonywany bezpośrednio do fizycznej pamięci FLASH, ale do specjalnego obszaru pamięci RAM. Asynchroniczny podsystem, niezależny od maszyny wirtualnej (podobnie jak podsystem komunikacyjny), synchronizuje zawartość tego specjalnego obszaru z pamięcią FLASH po każdym cyklu, dbając o integralność przechowywanych danych i przedłużenie żywotności pamięci.

Status pamięci nieulotnej można sprawdzić przy pomocy funkcji NV_PAR_STATUS.

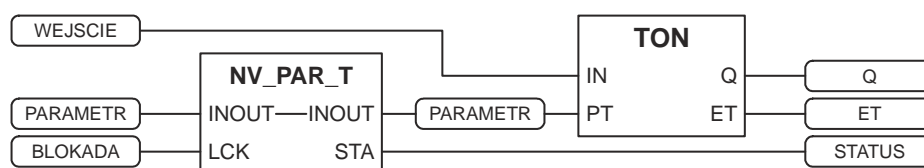
NV_PAR_STATUS — BYTE

Funkcja służy do kontroli statusu pamięci nieulotnej. Wyjście pokazuje kod ostatniego błędu pamięci nieulotnej zgodnie z poniższą tabelą. Funkcja ta wewnętrznie korzysta z zaawansowanej funkcji sterującej FLASH_CTL_CMD.

Kod błędu	Znaczenie
0	Brak błędu.
1	Próba odczytu zmiennej, która nie została wcześniej zapisana (nie znaleziono zmiennej).
2	Próba odczytu lub zapisu zmiennej, której rozmiar różni się od poprzednio zapisanej.
3	Pamięć zapełniona, próba zapisu zmiennej, której rozmiar przekracza pozostały wolny obszar.
4	Zbyt częste zapisy do pamięci nieulotnej.

Przykład

Monitorowanie i zapis do pamięci FLASH zmiennej PARAMETR będącej wejściem bloku TON. Wznowienie zasilania lub przekłamanie w pamięci RAM nie powoduje utraty wartości zapisanej uprzednio do pamięci FLASH.



Realizacja 1 – blokada zapisu

```

PROGRAM TEST_FLASH_1
VAR_EXTERNAL (*$AUTO*)
    WEJSCIE : BOOL;          (* Wejście bloku TON *)
    PARAMETR: TIME;          (* Zmienna zapisywana do FLASH *)
    BLOKADA : BOOL:=TRUE;    (* Blokada zapisu do FLASH *)
    Q       : BOOL;          (* Wyjście Q bloku TON *)
    ET      : TIME;          (* Wyjście ET bloku TON *)
    STATUS  : BOOL;          (* Status zapisu pamięci FLASH *)
END_VAR;

VAR
    NV_PAR1 : NV_PAR_T;
    TON1    : TON;

```

```

END_VAR;

(* Wejście INOUT musi być skojarzone ze zmienną! *)
NV_PAR1(INOUT:=PARAMETR, LCK:=BLOKADA); (* PARAMETR zapisywany we FLASH *)
STATUS := NV_PAR1.STA;                  (* Status zapisu FLASH *)

TON1(IN:=WEJSCIE, PT:=PARAMETR);        (* Opóźnione załączenie (TON) *)
Q := TON1.Q;                            (* Wyjście Q bloku TON *)
ET := TON1.ET;                          (* Wyjście ET bloku TON *)

END_PROGRAM

```

W momencie wznowienia zasilania zmienna BLOKADA przyjmuje wartość TRUE, co blokuje możliwość dokonania zapisu do pamięci FLASH. Jednocześnie na wyjściu INOUT pojawia się wartość odczytana z tej pamięci. W czasie, gdy BLOKADA=TRUE, zmienna PARAMETR nadpisywana jest wartością odczytaną z FLASH w celu uniknięcia zapisu pamięci domyślną wartością zmiennej (inną niż w pamięci FLASH) po zdjęciu blokady. Chcąc po pewnym czasie zmienić wartość mieszczącą się w pamięci FLASH, należy najpierw zlikwidować blokadę (BLOKADA=FALSE). Gdy BLOKADA=FALSE, wtedy każdorazowa zmiana wartości zmiennej PARAMETR zapisywana jest do pamięci FLASH i przekazywana na wyjście INOUT. Ponieważ po wykonaniu obliczeń wewnątrz bloku NV_PAR1 PARAMETR=NV_PAR1.INOUT, dlatego wejście PT bloku TON1 może być skojarzone ze zmienną PARAMETR. Rozwiązanie może być stosowane również w przypadku zadawania wartości parametrów z systemu SCADA.

Realizacja 2 – bez blokady zapisu

```

PROGRAM TEST_FLASH_2
VAR_EXTERNAL (*$AUTO*)
    WEJSCIE : BOOL;          (* Wejście bloku TON *)
    PARAMETR: TIME;          (* Zmienna zapisywana do FLASH *)
    Q        : BOOL;          (* Wyjście Q bloku TON *)
    ET       : TIME;          (* Wyjście ET bloku TON *)
    STATUS   : BOOL;          (* Status zapisu pamięci FLASH *)
END_VAR;

VAR
    NV_PAR1 : NV_PAR_T;
    TON1     : TON;
END_VAR;

(* Wejście INOUT musi być skojarzone ze zmienną! *)
NV_PAR1(INOUT:=PARAMETR);      (* PARAMETR zapisywany we FLASH *)
STATUS := NV_PAR1.STA;         (* Status zapisu FLASH *)

TON1(IN:=WEJSCIE, PT:=PARAMETR); (* Opóźnione załączenie (TON) *)
Q := TON1.Q;                   (* Wyjście Q bloku TON *)
ET := TON1.ET;                 (* Wyjście ET bloku TON *)

END_PROGRAM

```

Przy wznowieniu zasilania blok NV_PAR1 odczytuje pamięć FLASH i nadpisuje zmienną PARAMETR. Wejście LCK domyślnie ustawione jest na FALSE (brak blokady), więc każda zmiana wartości zmiennej PARAMETR będzie zapisywana do pamięci FLASH. Rozwiązanie nie wymaga żadnej interwencji użytkownika, ale jest

niezalecane przy korzystaniu z sytemu SCADA, z uwagi na możliwość nadpisania zmiennej `PARAMETR` domyślną wartością pochodzącą z zewnętrznego systemu przy jego uruchamianiu.