



Podtrzymywanie danych w pamięci sterowników BXxx00

Poziom trudności: łatwy

Wersja dokumentacji: 1.7

Aktualizacja: 8.11.2013

Beckhoff Automation Sp. z o. o.

Wstęp

Dane po zaniku zasilania mogą być przechowywane w pamięci sterowników serii BXxx00 przez zmienne dwóch typów: **Retain** (2 kB) i **Persistent** (1 kB). Oba typy zmiennych są przechowywane w pamięci typu NOVRAM, która jest zapisywana w każdym cyklu. Zapis 1 kB zajmuje w przybliżeniu 1 ms. Sposób wywołania zmiennych typu **Persistent** różni się w zależności od wersji firmware'u na sterowniku.

Porównanie zmiennych retain i persistent

Poniżej przedstawione są czynności wykonywane na sterowniku oraz programie i zdolność przechowania wartości przez zmienne typu retain i persistent.

CZYNNOŚĆ	RETAIN	PERSISTENT
Restart programu lub sterownika	TAK	TAK
Zmiana programu bez zatrzymania (online change)	TAK	TAK
Zmiana programu z zatrzymaniem sterownika (load all)	NIE	TAK
Wgranie nowego programu	NIE	TAK
Reset programu	NIE	TAK
Reset All	NIE	NIE
Przywrócenie ustawień fabrycznych	NIE	NIE

Zmienne Retain

Zmienną typu Retain uzyskuje się poprzez odpowiednią deklarację

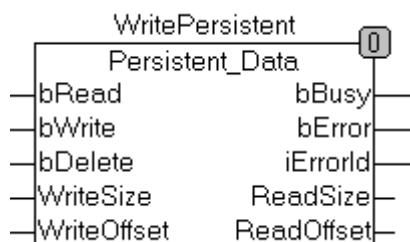
```

0001 PROGRAM MAIN
0002 VAR RETAIN
0003 (* Zmienne, typu RETAIN *)
0004   Remanent_Bool:  BOOL;
0005   Remanent_Integer: INT;
0006   Remanent_Word:   WORD;
0007   Remanent_String: STRING;
0008   Remanent_Byte:   BYTE;
0009   Remanent_Real:   REAL;
0010 END_VAR

```

Zmienne Persistent (BX FW > 1.17)

Wszystkie sterowniki z serii BX z firmware w wersji 1.17 lub wyższym, mają możliwość zapamiętywania zmiennych zaadresowanych w obszarze %M jako zmienne pamięci NOVDPRAM (nieulotne). Zmienne te zapisywane są poprzez wywołanie bloczka **Persistent_Data**. Poniżej przedstawiono wywołanie bloczka Persistent_Data.



Wejścia:

- **bWrite** – wejście zapisuje parametry zmiennych nieulotnych – rozmiar i przesunięcie obszaru – należy wywołać go **tylko raz** gdy zmieniamy parametry.
- **WriteSize** – rozmiar danych zapamiętywanych po zaniku zasilania (nie więcej niż 1kB).
- **WriteOffset** – początek danych zadeklarowanych w obszarze %M.

Na ftp.beckhoff.com/Poland/Pomoc znajduje się przykładowy program obsługi zmiennych Persistent – P_PERSISTENT.EXP.

Zmienne Persistent (BX FW < 1.17)

W sterownikach serii BX z firmware mniejszym niż 1.17 zmienną typu Persistent uzyskuje się poprzez odpowiednią deklarację

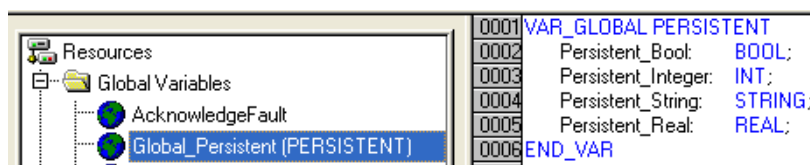
```

0011 VAR RETAIN PERSISTENT
0012 (* Zmienne, typu PERSISTENT *)
0013 Persistent_Bool:  BOOL;
0014 Persistent_Integer: INT;
0015 Persistent_Word:   WORD;
0016 Persistent_String: STRING;
0017 Persistent_Byte:   BYTE;
0018 Persistent_Real:   REAL;
0019 END_VAR

```

Uwagi:

1. Jeżeli w bloku funkcyjnym znajduje się chociaż jedna zmienna oznaczona jako retain lub persistent to do pamięci zostanie zapisana cała zawartość bloku funkcyjnego. Z tego względu zalecane jest umieszczenie wszystkich zmiennych globalnych **poza wnętrzami bloków funkcyjnych**, np. w oddzielnej zakładce zmiennych globalnych.



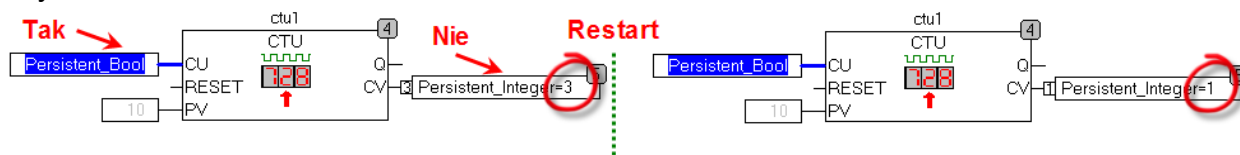
2. Wartość inicjalizacyjna (Initial Value) zmiennej PERSISTENT zostanie przypisana tylko w przypadku błędu odczytu zmiennych z pliku.

Przykład:

Persistent_Integer :INT := 15200;

3. Sprawdź czy zmienna persistent **nie jest nadpisywana** przez blok funkcyjny. W poniższym przykładzie zmienna Persistent_Bool jest wejściem a zmienna Persistent_Integer wyjściem bloku licznika. Po restarcie sterownika wartość wejściowa jest prawidłowo przechowana, natomiast wartość wyjściowa została w pierwszym cyklu nadpisana przez blok licznika.

Przykład:



4. Po wgraniu nowego projektu na sterownik, przed jego uruchomieniem sprawdź wartości zmiennych typu persistent. Może okazać się, że są one niezerowe – przechowują wartości z poprzedniego programu. Wykonaj komendę **Reset All**.