

Stanisław BOCIAN<sup>1</sup>  
Jarosław IWANOWSKI<sup>1</sup>

### **ZASTOSOWANIE MASZYNY STANOWEJ DO PROJEKTOWANIA UKŁADÓW WEJŚCIOWYCH WYKORZYSTUJĄC MIKROSYSTEM CYFROWY FIRMY CYPRESS Z PROGRAMEM PSoC EXPRESS**

*W artykule zaprezentowano przykładową realizację maszyny stanowej (automatu cyfrowego) z wykorzystaniem programu narzędziowego dla procesorów PSoC firmy Cypress – PSoC Express. Zamieszczono bloki logiczne programu wraz z komentarzami.*

### **APPLICATION THE FINITE STATE MACHINE FOR DESIGNING THE INPUT SYSTEMS USING THE DIGITAL MICROSYSTEM OF CYPRESS COMPANY WITH PSoC EXPRESS PROGRAM**

*The exemplary realization of the finite state machine (the digital automaton) with using the tool program for PSoC processors of Cypress company – PSoC Express is presented in this article. The logical blocks of program with the comments are included in it.*

## **1. WSTĘP**

Rozwój układów automatyki wymaga stosowania nowych rozwiązań mikroprocesorów, układów wejść i wyjść oraz układów komunikacyjnych. W procesie projektowania cyfrowych układów sterowania oprócz stosowania modeli specyfikacji formalnej bardzo ważne jest sprecyzowanie docelowej platformy realizacyjnej: sprzętowej, programowej i sprzętowo – programowej. Zalety i wady realizacji sprzętowej oraz programowej, rozpatruje się pod względem dwóch podstawowych kryteriów:

- czas reakcji – szybsze rozwiązanie sprzętowe,
- koszty realizacji – przyjmuje się niższe dla rozwiązań programowych, ze względu na niższą cenę zarówno samych układów, jak i narzędzi wspomagający proces projektowania.

---

<sup>1</sup>Stanisław BOCIAN Instytut Pojazdów Szynowych „TABOR” POLSKA; Poznań 61-055;Warszawska 181.

Telefon: + 48 (0)61 664 14 38; E-mail: Elektrotechnika @ tabor.com.pl

Jarosław IWANOWSKI Instytut Pojazdów Szynowych „TABOR” POLSKA; Poznań 61-055;Warszawska 181.

Telefon: + 48 (0)61 664 14 38; E-mail: Elektrotechnika @ tabor.com.pl

Własności te wydają się wzajemnie sprzeczne, dlatego należałoby zastosować jednocześnie obie metody stosując kompromis, łącząc zalety szybkości działania z niskimi kosztami produkcji. Funkcje takie realizują mikrosystemy cyfrowe. Układy takie buduje wiele firm, takie jak: ATMEL, CYPRESS i TRISCEND. Wspólną cechą tych układów jest posiadanie rdzenia mikroprocesorowego oraz programowalnej części sprzętowej (systemy o realizacji sprzętowo – programowej). Pod pojęciem mikrosystem cyfrowy rozumie się taki układ scalony, który w swej strukturze integruje (obok innych) właśnie te dwa elementy. Umożliwia to zmniejszenie rozmiarów gabarytowych produktu, zużycia energii, a nawet kosztów związanych z produkcją w porównaniu do rozwiązań bazujących na łączeniu osobno struktur sprzętowych i mikroprocesorowych.

## 2. PROGRAMOWANIE MIKROSYSTEMÓW CYFROWYCH PSoC FIRMY CYPRESS

W pracy przedstawiono strukturę rozproszonego sterowania w oparciu o mikrokontroler PSoC (Programmable System on Chip) firmy Cypress. Układ ten różni się od typowych mikrokontrolerów tym, że posiada programowalne peryferia nie tylko cyfrowe, ale także analogowe. Wybrano tę rodzinę układów ze względu na bogate możliwości programowego tworzenia układów analogowych z rekonfigurowanych bloków.

Tworzenie oprogramowania jest możliwe za pomocą dwóch narzędzi:

### **PSoC Designer**

Umożliwia tworzenie oprogramowania z wykorzystaniem zaimplementowanych bloków funkcjonalnych uzupełnionych o programy w Asemblerze (i/lub) języku C.

### **PSoC Express**

– Przeznaczony do tworzenia programu w oparciu o zaimplementowane bloki funkcjonalne we/wy oraz funkcje przejścia. Funkcje przejścia mogą być:

– kombinacyjne (Table Lookup)

– warunkowe (warunki w języku C):

    Status Encoder      If x1 then y1

                          If x2 then y2

                          If x3 then y3

    Priority Encoder    If x1 then y1

                          Else If x2 then y2

                          Else If x3 then y3

– stanowe (State Machine) – umożliwia tworzenie programu w oparciu o sporządzony wcześniej graf automatu.

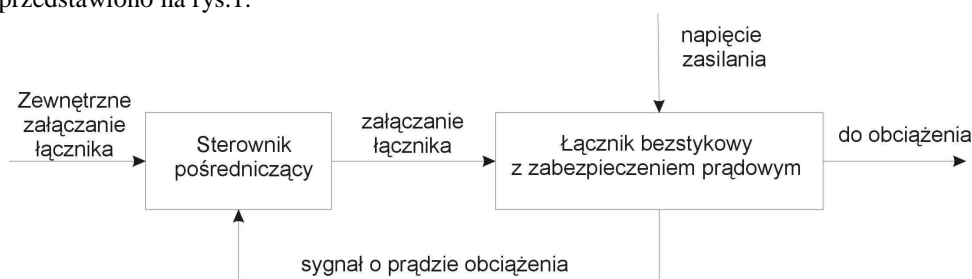
W pracy [1] przedstawiono przykładowo sposób realizacji programu PSoC Express. oraz przykładowy graf maszyny stanów. Opisano tam także strukturę wewnętrzną mikrosystemu cyfrowego PSoC firmy CYPRESS.

### 3. PROJEKT STEROWNIKA POŚREDNICZĄCEGO DO STEROWANIA ŁĄCZNIKÓW BEZSTYKOWYCH Z WEWNĘTRZNYM ZABEZPIECZENIEM PRĄDOWYM

Należy zaprojektować sterownik pośredniczący w sterowaniu łączników bezstykowych z wewnętrznym zabezpieczeniem prądowym tak, by na podstawie sygnału załączenia i pomiaru prądu obciążenia rozróżniał zadziałanie zabezpieczenia prądowego łącznika z powodu:

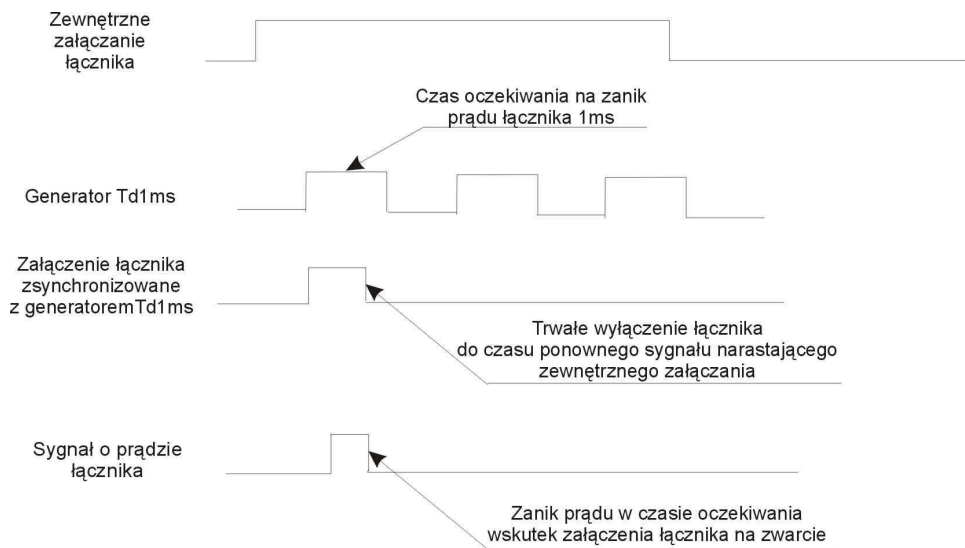
- załączenia w przypadku zwarcia,
- wyłączenia z powodu podniesienia napięcia zasilania wskutek komutacji innych łączników.

Schemat blokowy włączenia sterownika pośredniczącego w obwód sterowania przedstawiono na rys.1.



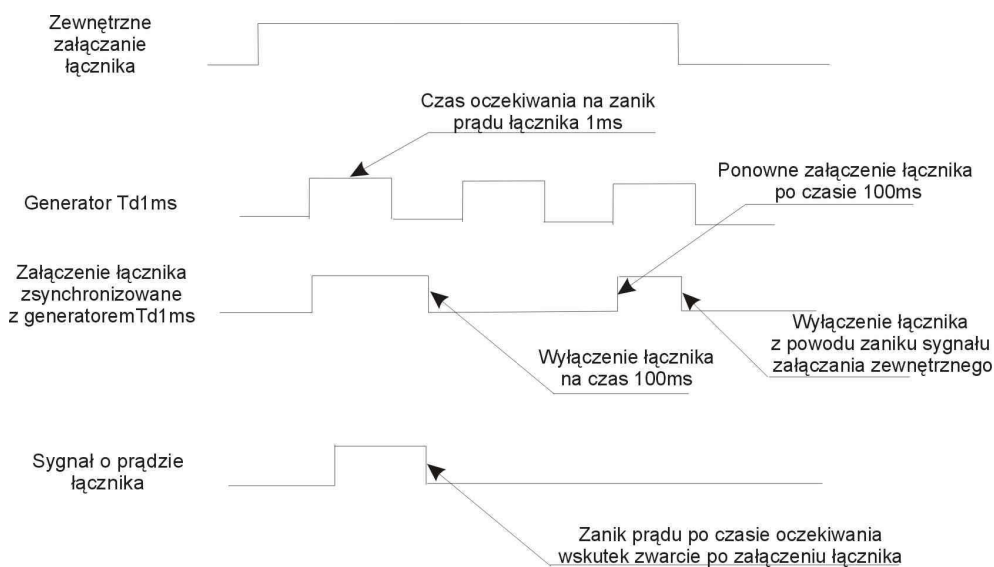
Rys.1. Schemat blokowy włączenia sterownika pośredniczącego w obwód sterowania

Rozwiązanie tego problemu zostanie obecnie szczegółowo opisane. Po podaniu sygnału zewnętrznego załączenia łącznika na wejście `wec_1` (rys.4) sterownika pośredniczącego następuje synchronizacja tego sygnału ze zboczem narastającym wewnętrznego generatora czasu oczekiwania  $Td1ms$  i podanie tego sygnału na wejście łącznika. Od tej chwili następuje odmierzenie czasu ( $Td1ms$ ), w którym spodziewane jest pojawienie się sygnału o prądzie obciążenia. Jeżeli w tym czasie prąd obciążenia pojawi się i zniknie to znaczy, że nastąpiło załączenie łącznika na zwarcie i wtedy sterownik pośredniczący wyłączy sterowanie łącznika na stałe (rys.2) do momentu pojawienia się sygnału zewnętrznego załączenia.



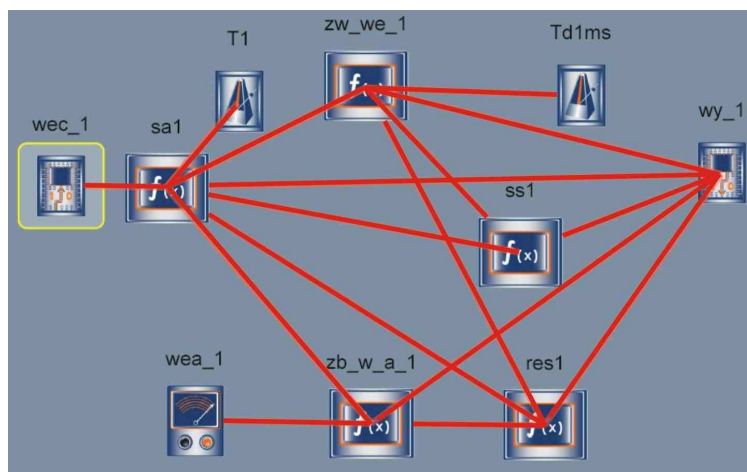
Rys.2. Diagram działania układu w przypadku załączenia na zwarcie

Jeżeli natomiast prąd obciążenia zniknie po czasie oczekiwania ( $T_{d1ms}$ ) to znaczy, że prąd łącznika wzrósł ponad wartość zabezpieczenia wskutek podniesienia napięcia zasilania. W tym przypadku sterownik pośredniczący po krótkiej przerwie załączy sterowanie ponownie.



Rys.3. Diagram działania układu w przypadku wystąpienia zwarcia po załączeniu

Schemat blokowy realizacji programu w środowisku programu PsoC Express przedstawiono na rys.4



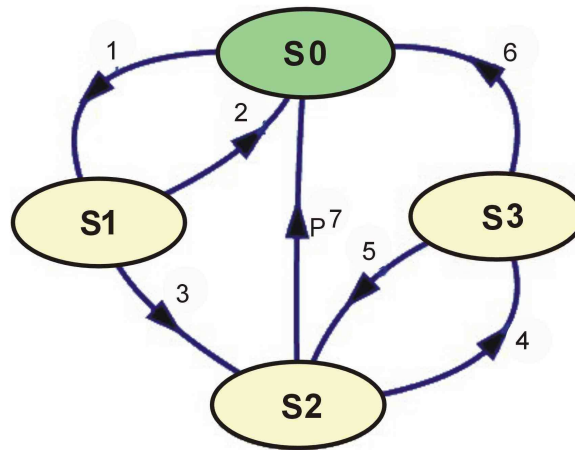
Rys.4. Schemat blokowy realizacji programu w środowisku programu PsoC Express

Oznaczenia elementów schematu blokowego.

- [1] wec\_1      wejście binarne sygnału zewnętrznego załączania łącznika
- [2] wea\_1      wejście analogowe sygnału pomiarowego prądu obciążenia
- [3] sa1        funkcja przejścia eliminacji drgań zbczy sygnału „wec\_1”
- [4] T1         timer opóźnienia zbczy sygnału „wec\_1” dla sa1
- [5] ss1        funkcja powtarzania odfiltrowanego sygnału „wec\_1”
- [6] zw\_we\_1   funkcja synchronizująca zbcze narastające ss1 z generatorem Td1ms
- [7] wy\_1      wyjście binarne załączenia łącznika
- [8] zb\_w\_a\_1   funkcja wykrywania zbcza opadającego prądu obciążenia
- [9] res\_1      funkcja formowania sygnału o wykryciu zbcza opadającego prądu obciążenia

Funkcja przejścia eliminacji drgań zbczy sygnału „wec\_1” została zrealizowana za pomocą automatu stanowego „sa1” przedstawionego na rys.5. Składa się z 4 stanów S0÷S3 i 7 warunków przejścia:

- 1      wec\_1==wec\_1\_High
- 2      wec\_1==wec\_1\_Low&&T1\_Status==T1\_Status\_\_Not\_Elapsed
- 3      wec\_1==wec\_1\_High&&T1\_Status==T1\_Status\_\_Elapsed
- 4      wec\_1==wec\_1\_Low
- 5      wec\_1==wec\_1\_High&&T1\_Status==T1\_Status\_\_Not\_Elapsed
- 6      wec\_1==wec\_1\_Low&&T1\_Status==T1\_Status\_\_Elapsed
- p7     zw\_we\_1==zw\_we\_1\_2&&zb\_w\_a\_1==zb\_w\_a\_1\_2



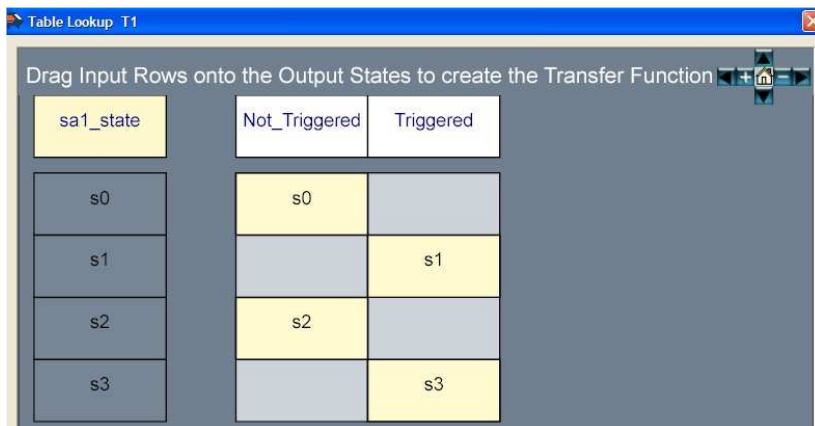
Rys.5. Graf automatu stanowego „sa1” obsługi wejścia cyfrowego „wec\_1”

Opis działania obsługi wejścia cyfrowego „wec\_1”.

Stan „s0” jest stanem początkowym i oznacza jednocześnie stan 0 wejścia „wec\_1” widziany przez dalszą część programu, stan „s2” oznacza zerowy stan 1 wejścia „wec\_1”. Stany „s1” i „s3” są przejściowymi stanami oczekiwania.

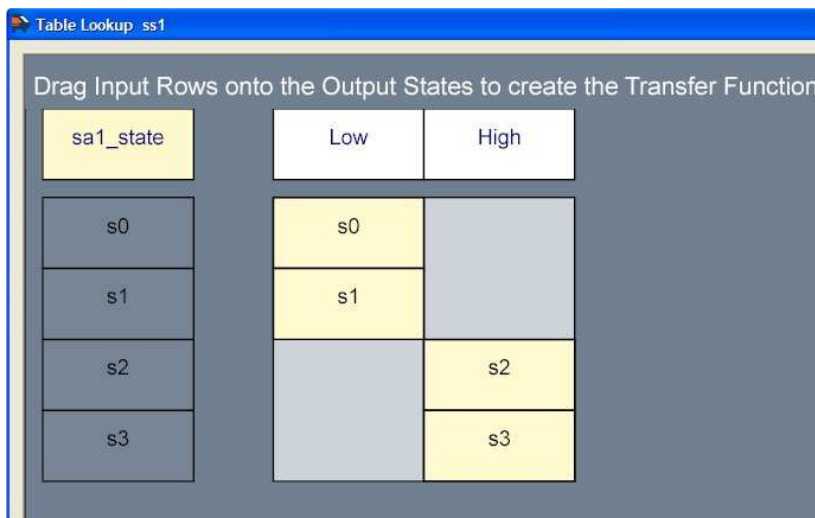
Automat przechodzi ze stanu „s0” do „s2”, gdy „wec\_1” przyjmuje stan 1 na czas większy od T1. W przeciwnym przypadku przyjmuje stan 0. Podobnie automat przechodzi ze stanu „s2” do „s0”, gdy „wec\_1” przyjmuje stan 0 na czas większy od T1. W przeciwnym przypadku przyjmuje stan 1. Jest to więc układ eliminujący drgania zboczy sygnału „wec\_1”. Ponadto automat przechodzi ze stanu „s2” do „s0” za sprawą warunku p7. Jest to wymuszenie zewnętrzne odłączające sygnał sterujący od układu do czasu ponownego pojawienia się zbocza narastającego na „wec\_1”. Warunek p7 oznacza, że zaniknął prąd obciążenia (zb\_w\_a\_1\_\_2) po upływie czasu oczekiwania Td1ms (zw\_we\_1\_\_2) i należy w związku z tym trwale i niezależnie od zewnętrznego sygnału załączenia łącznika odłączyć sygnał sterujący od układu do czasu ponownego pojawienia się zbocza narastającego na „wec\_1”.

Stany „s1” i „s2” automatu „sa1” wyzwalają generator monostabilny T1 (rys.6).



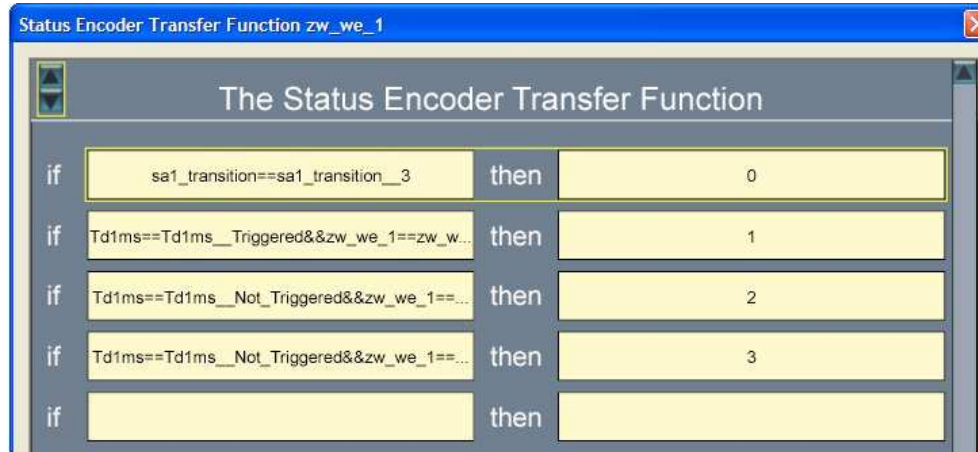
Rys.6. Wyzwalania timera T1

Na podstawie stanów automatu „sa1” funkcja przejścia „ss1” przyjmuje wartość logiczną „low”, gdy „wec\_1” = 0 lub trwa odmierzenie czasu T1 dla zbocza narastającego „wec\_1” oraz wartość logiczną „high”, gdy „wec\_1” = 1 lub trwa odmierzenie czasu T1 dla zbocza opadającego „wec\_1” (rys.7).



Rys.7. Przyporządkowania wartości logicznych funkcji „ss1”

Funkcja przejścia zw\_we\_1 synchronizuje zbocze narastające ss1 ze zboczem narastającym generatora Td1ms (rys.8)



Rys.8. Wartości funkcji zw\_we\_1

sa1\_transition==sa1\_transition\_3

Td1ms==Td1ms\_\_Triggered&&zw\_we\_1==zw\_we\_1\_\_3

Td1ms==Td1ms\_\_Not\_Triggered&&zw\_we\_1==zw\_we\_1\_\_1

Td1ms==Td1ms\_\_Not\_Triggered&&zw\_we\_1==zw\_we\_1\_\_0&&ss1==ss1\_\_High

Funkcja przejścia „zw\_we\_1” przyjmuje wartość „0”, gdy zostanie wykryte zbocze narastające sygnału z „wec\_1” czyli warunku nr 3 w automacie stanowym „sa1”.

Funkcja przejścia „zw\_we\_1” przyjmuje wartość „3”, gdy licznik odmierzający 1 ms „Td1ms” nie jest wyzwolony podczas, gdy „zw\_we\_1” ma wartość „0” i funkcja przejścia „ss1” przyjmuje wartość logiczną „high”. Jest to przejście ze stanu „zw\_we\_1\_0” i oznacza, że zostało wykryte i odfiltrowane zbocze narastające sygnału wejściowego i trwa on nadal, lecz generator Td1ms jest jeszcze w stanie „0”.

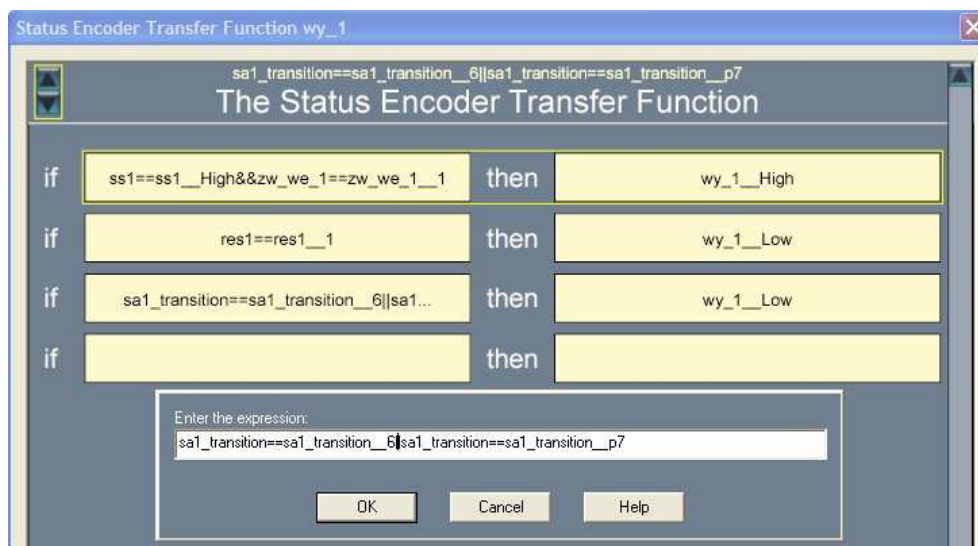
Funkcja przejścia „zw\_we\_1” przyjmuje wartość „1”, gdy zostanie wyzwolony licznik odmierzający 1 ms „Td1ms” podczas, gdy „zw\_we\_1” ma wartość „3”. Jest to przejście ze stanu „zw\_we\_1\_3” i oznacza, że zostało wykryte i odfiltrowane zbocze narastające sygnału wejściowego i trwa on nadal i generator Td1ms jest już w stanie „1”. Jest to powtórzenie sygnału „wec\_1” po odfiltrowaniu i zsynchronizowaniu z sygnałem wewnętrznego generatora.

Funkcja przejścia „zw\_we\_1” przyjmuje wartość „2”, gdy licznik odmierzający 1 ms „Td1ms” nie jest wyzwolony podczas, gdy „zw\_we\_1” ma wartość „1”. Jest to przejście ze stanu „zw\_we\_1\_1” i oznacza, że generator Td1ms odmierzył zadany interwał i jest w stanie „0”.

Wyjście wy\_1 (rys.9) przyjmuje wartość „1” w przypadku, gdy wejście wec\_1 po synchronizacji czyli ss1 ma wartość „1” i zostało wykryte i odfiltrowane zbocze narastające sygnału wejściowego i generator Td1ms jest w stanie odliczania czasu oczekiwania na zwarcie.

Warunki „6” i „p7” automatu sa1 powodujące przejście ze stanów „s2” do „s0” i „s3” do „s0” wymuszają na wy\_1 wartość „0”. Ten sam efekt daje wykrycie zbocza opadającego prądu obciążenia łącznika wykryte przez funkcję zb\_wa\_1 (rys.10) czyli res1\_1.



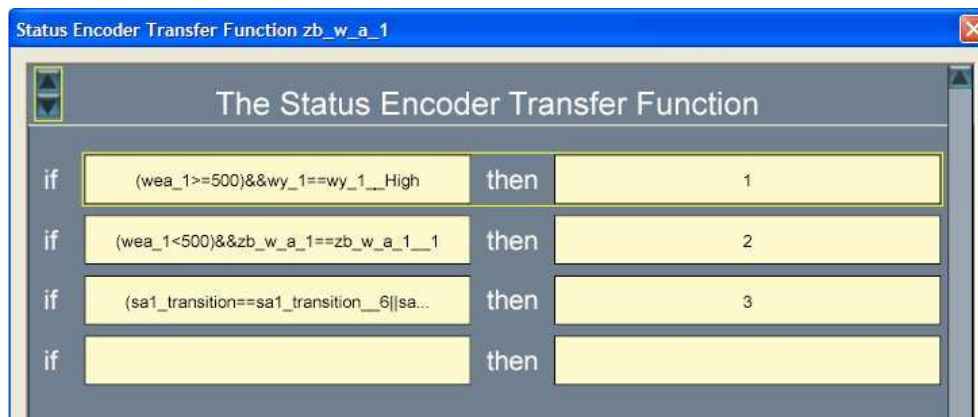


Rys.9. Wartości funkcji wy\_1

ss1==ss1\_High&&zw\_we\_1==zw\_we\_1\_1

res1==res1\_1

sa1\_transition==sa1\_transition\_6||sa1\_transition==sa1\_transition\_p7

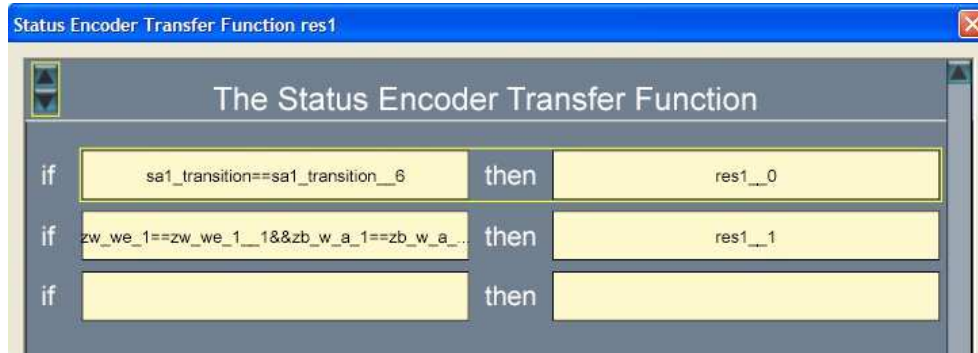


Rys.10. Wartości funkcji zb\_w\_a\_1

(wea\_1>=500)&&wy\_1==wy\_1\_High

(wea\_1<500)&&zb\_w\_a\_1==zb\_w\_a\_1\_1

(sa1\_transition==sa1\_transition\_6||sa1\_transition==sa1\_transition\_p7)

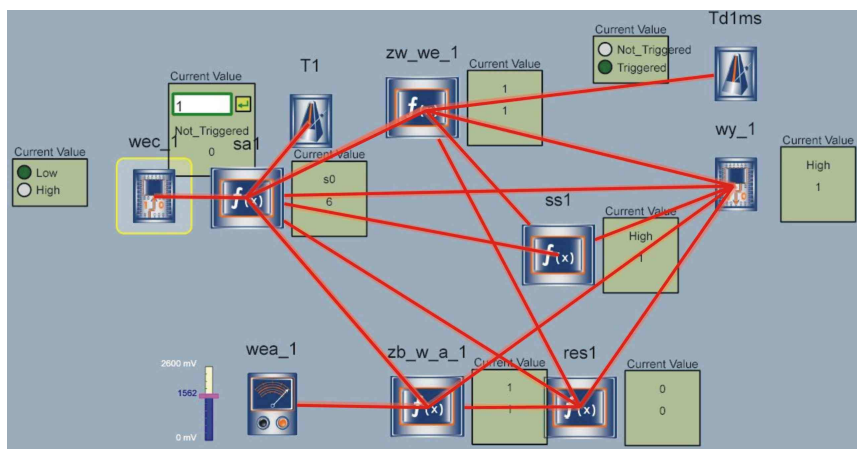


Rys.11. Wartości funkcji res1

sa1\_transition==sa1\_transition\_6

zw\_we\_1==zw\_we\_1\_\_1&&zb\_w\_a\_1==zb\_w\_a\_1\_\_2

Za pomocą programu narzędziowego PsoC Express można dokonać symulacji działania programu poprzez wymuszanie stanów logicznych i analogowych wejść, ręczne wyzwalanie działania timerów, zadawanie czasów trwania sygnałów z generatorów oraz obserwację stanów wyjść cyfrowych i funkcji przejścia. Na rys.12 przedstawiono okno symulacji opisywanego programu.



Rys. 12. Okno symulacji działania programu w PsoC Express

W oknach zamieszczonych obok bloków poszczególnych funkcji przedstawiono ich aktualny stan. Przykładowo automat „sa1” jest w stanie „s0” , do którego przeszedł za pomocą warunku przejścia nr 6.

### 3. WNIOSKI

Programowanie w środowisku programistycznym PSoC Express umożliwia szybką realizację projektu bez konieczności gruntownej znajomości języka C++. Całą strukturę programu generuje automatycznie kompilator. Dzięki temu użytkownik PSoC Express może poświęcić całą uwagę na precyzyjnym określeniu zadania sterowania. Możliwość wykorzystania do programowania narzędzia w postaci maszyny stanowej jest nieoceniona z punktu widzenia automatyka.

### 4. BIBLIOGRAFIA

- [1] Bocian S., Iwanowski J.: *Uniwersalny układ wejść – wyjść do zastosowania w pojazdach szynowych z wykorzystaniem mikrosystemu cyfrowego PSoC CY29466 firmy CYPRESS* Pojazdy Szynowe 3/2008.