

Własności energetyczne szeregowego interfejsu komunikacyjnego SPI

Streszczenie. W artykule przedstawiono wyniki badań rzeczywistych własności energetycznych szeregowego interfejsu komunikacyjnego SPI. Do badań wybrano interfejs mikrokontrolera Silicon Labs C8051F902 z grupy mikrokontrolerów o niskim zużyciu energii. Opracowano szereg programów w wielu wariantach służących do wnikliwego zbadania zachowania się interfejsu w różnych konfiguracjach programowych oraz sprzętowych. Przebadano wpływ źródła taktowania mikrokontrolera, szybkości transmisji danych, napięcia zasilania układu, porównano wbudowane moduły interfejsu SPI oraz zmian ustawień portów I/O na własności czasowo-energetyczne interfejsu SPI.

Abstract. The real energetic properties of SPI serial communication interface have been presented in this paper. The interface of Silicon Labs microcontroller C8051F902 was selected for research from the group of microcontrollers with low power consumption. The group of programs in different versions for accurate testing of interface in different programming and hardware configurations were elaborated. The influence of microcontroller timing source, speed of data transmission and power supply on energetic properties of SPI interface was investigated. The internal modules of SPI interface and modifications of I/O ports sets were also analysed. **Energetic properties of SPI serial communication interface**

Słowa kluczowe: SPI, szeregowy interfejs komunikacyjny, układ o niskim zużyciu energii, mikrokontroler.

Keywords: SPI, serial communication interface, low power consumption circuit, microcontroller

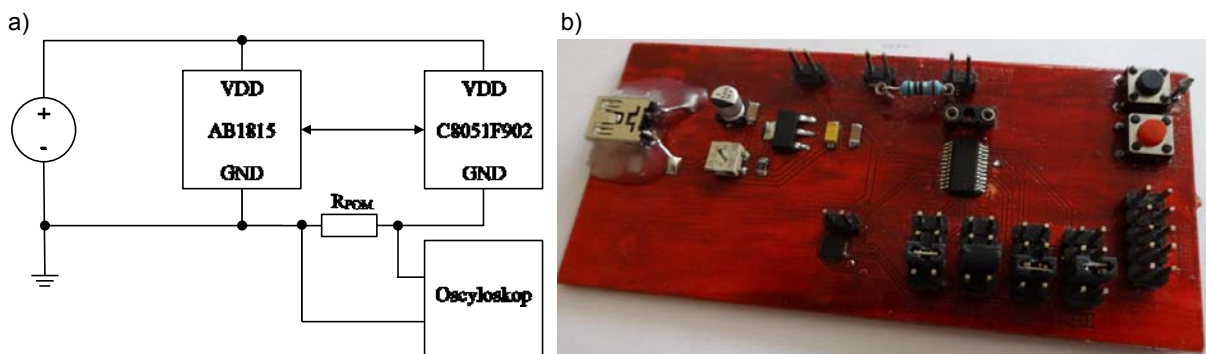
Wprowadzenie

Zasilanie bateryjne elektronicznych urządzeń mobilnych lub zasilanie ich energią odzyskiwaną na wiele sposobów z otoczenia, skłania do stosowania mikroprocesorów o niskim zużyciu energii. Chociaż mikrokontrolery te mają przeważnie wysoką sprawność energetyczną, to ich peryferia (np. pamięci zewnętrzne, zegary RTC, czujniki) mogą w znaczny sposób zwiększać zużycie energii przez urządzenie. Energia zużywana przez urządzenia składa się z energii potrzebnej do ich rzeczywistej pracy i energii potrzebnej do komunikacji z peryferiami. Składniki te są trudne do wyodrębnienia i jednoznacznego określenia ich wartości. W niniejszym opracowaniu podjęto próbę scharakteryzowania pod tym kątem interfejsu SPI stosowanego w systemach wbudowanych. Zaprojektowano stanowisko do pomiaru zużycia energii przez mikrokontroler typu C8051F902 firmy Silicon Laboratories podczas komunikowania się z zegarem czasu rzeczywistego RTC AB1815 firmy ABRACON w różnych konfiguracjach sprzętowych i programowych.

Ostatecznym celem pracy jest wyznaczenie rzeczywistych parametrów energetycznych wybranych układów I/O systemu mikroprocesorowego (SPI, IIC, ADC) o deklarowanym niskim zużyciu energii i zastosowanego w aplikacji o niekonwencjonalnym rozwiązaniu zasilania systemowego w postaci układu odzysku energii z pól elektromagnetycznych systemów RFID oraz systemów teletransmisyjnych powszechnego użytku.

Koncepcja układu pomiarowego

W założeniu stanowisko do badania własności energetycznych interfejsu SPI składa się z mikrokontrolera C8051F902 [1], który jest układem badanym, układu symulującego obciążenie linii komunikacyjnych w postaci zegara czasu rzeczywistego AB1815 [2] oraz układu zasilającego. Głównym elementem pomiarowym jest rezystor R_{POM} , na którym za pomocą oscyloskopu mierzony jest spadek napięcia (rys. 1). Dzięki ograniczeniu przebiegu w czasie, można wyliczyć zużycie energii przez mikrokontroler na wysłanie pakietu danych (wzór 1) [3].



Rys. 1. Schemat układu pomiarowego (a) oraz widok układu (b) do pomiaru własności energetycznych szeregowego interfejsu komunikacyjnego SPI

$$(1) \quad E = \int_{T_0}^{T_1} \frac{V_{DD} V_{R_{POM}}}{R_{POM}} dt$$

Układ zasilano z przenośnego źródła energii o napięciu 5 V i po uwzględnieniu spadku napięcia na stabilizatorze uzyskano napięcia zasilania układu C8051F902 od 1,25 ÷ 3,8 V, co było wystarczające do celów pomiarowych. Zastosowano rezystor pomiarowy o wartości rezystancji 5,045 Ω, którego wartość zweryfikowano mostkiem RLC.

Pomiary wykonywano przy pomocy cyfrowego oscyloskopu Tektronix TDS 2001C. Posiada on bardzo przydatną funkcję wyznaczania wartości RMS przebiegu z przedziału ograniczonego kursorami CursorRMS, która daje wyniki dokładniejsze niż uzyskiwane oscyloskopem MSO 4104 z funkcją TrueRMS i CyclicRMS [4]. Ta druga nie sprawdza się zresztą w przypadku okresowych przebiegów złożonych, jakie obserwuje się w badanym układzie. Weryfikacji metody dokonano poprzez wyznaczenie wartości skutecznej prądu w R_{POM} przy zasilaniu go

napieciem o przebiegu prostokątnym i porównanie wyników pomiaru oscyloskopem z funkcją CursorRMS z obliczeniami tego prądu z odpowiednich proporcji w dziedzinie czasu dla elementów przebiegu uzyskanych z pomiarów na ekranie oscyloskopu MSO 4104.

Przeprowadzenie badań

Wszystkie programy napisano w języku asemblera mikrokontrolerów rodziny 8051. Jest to język, którego główną zaletą jest możliwość pełnej optymalizacji wykorzystania wszelkich zasobów mikrokontrolera, a w szczególności możliwość kontroli każdego pojedynczego cyklu maszynowego. W niniejszej pracy jest to szczególnie ważne. Do celów pomiarowych opracowano trzy główne typy programów wraz z ich modyfikacjami dla poszczególnych przypadków. Wykorzystano sprzętowe przerwanie pochodzące od modułu SPIn.

Program z restartem programowym przedstawiono w

tablicy 1. Kolejno wykonywane są następujące zadania: zostaje wyłączony watchdog (brak zaburzeń ustalonego cyklu programowego), następuje konfiguracja źródła taktowania, modułu SPI, portów oraz przerwań. Po wpisaniu danych do rejestru SPInDAT rozpoczyna się transmisja. W programie z resetem programowym (tabl. 1 i rys. 2) po zakończeniu wysyłania danych następuje programowy reset mikrokontrolera. W programie z resetem programowym i trybem IDLE (tabl. 2 i rys. 3a) w czasie wysyłania danych mikrokontroler przechodzi w tryb IDLE, a po zakończeniu wysyłania danych następuje reset mikrokontrolera. Ten przypadek najbardziej odpowiada trybowi pracy w docelowej aplikacji układu [5]. W wersji ze skokiem programowym (tabl. 3 i rys. 3b) podczas wysyłania danych mikrokontroler przechodzi w tryb IDLE a po zakończeniu wysyłania danych następuje powrót na początek programu głównego.

Tabela 1. Fragment kodu programu z restartem programowym dla SPI0 podprogramem obsługi przerwania dla SYSCLOCK 20 MHz i oscylatora o niskim zużyciu energii przy wysłaniu jednego bajtu danych

```

Main:
    clr NSS0MD0          ; zerowanie linii NSS
    mov SPI0DAT,        #0B5h      ; dane do SPI0DAT
Main2:
    jnb P0.7,          Main2      ; oczekiwanie zakończenia trans.
    clr P0.7           ; zerowanie pinu P0.7
    orl RSTSRC,        #010h      ; zresetowanie mikrokontrolera

SPI0_INT:
    clr SPIF0           ; zerowanie flagi przerwania
    setb NSS0MD0       ; ustawianie linii NSS
    setb P0.7          ; ustawienie pinu P0.7
    reti               ; powrót do programu głównego

```

Tabela 2. Fragment zmodyfikowanego kodu programu z restartem programowym oraz trybem uśpienia

```

Main:
    clr NSS0MD0          ; zerowanie linii NSS
    mov SPI0DAT,        #0B5h      ; dane do SPI0DAT
Main2:
    jnb P0.7,          Main3      ; skok do Main3 jeśli nie ustawiony
    clr P0.7           ; zerowanie pinu P0.7
    orl RSTSRC,        #010h      ; zresetowanie mikrokontrolera
Main3:
    orl PCON,          #001h      ; przejście w tryb uśpienia
    sjmp               Main2      ; skok do Main2

```

Tabela 3. Fragment zmodyfikowanego kodu programu ze skokiem do etykiety Start oraz trybem uśpienia

```

Main:
    clr NSS0MD0          ; zerowanie linii NSS
    mov SPI0DAT,        #0B5h      ; dane do SPIDAT
Main2:
    jnb P0.7,          Main3      ; skok do Main3 jeśli nie ustawiony
    clr P0.7           ; zerowanie pinu P0.7
    ljmp Start         ; skok do początku programu głównego
Main3:
    orl PCON,          #001h      ; przejście w tryb uśpienia
    sjmp               Main2      ; skok do Main2

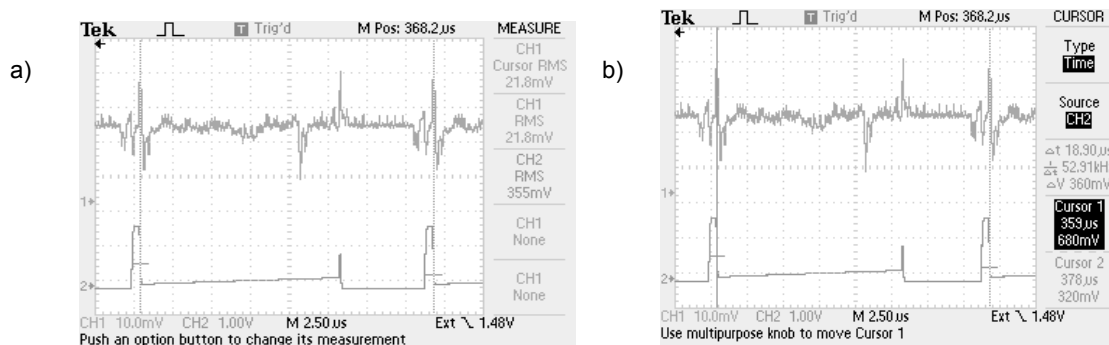
```

W zmodyfikowanych programach wprowadza się zmiany w ustawieniach szybkości taktowania mikrokontrolera, jego źródła oraz szybkości pracy interfejsu SPI. Dodatkowo sprawdzany był wpływ ustawienia niewykorzystywanych wyprowadzeń mikrokontrolera jako cyfrowych lub analogowych na pobór prądu przez układ.

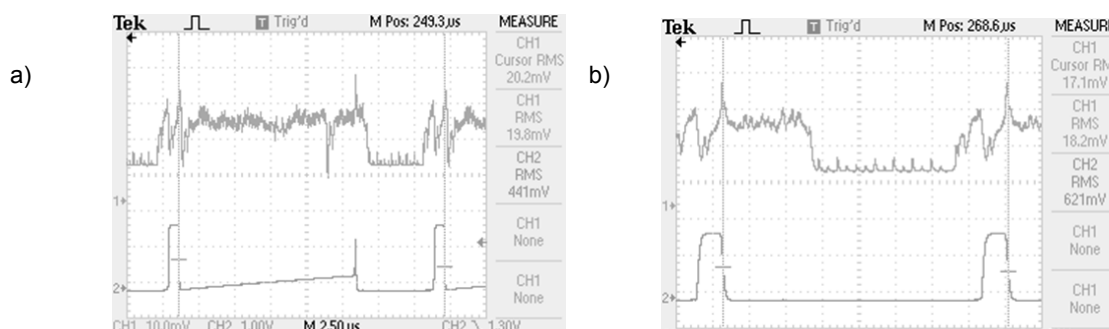
Jak się okazało, konsumpcja energii przez wbudowane moduły SPI0 i SPI1 podczas wykonywania różnych programów jest porównywalna, z czego wynika, że obydwa moduły SPI posiadają identyczną budowę wewnętrzną (brak takich danych w nocie katalogowej).

Porównując przebiegi z rysunków 2 i 3a można zauważyć wyraźny spadek poboru prądu w momencie przejścia w tryb IDLE, co pozwala zaoszczędzić około 12 nJ podczas wykonywania jednego cyklu programu.

Podczas badania wpływu źródła taktowania układu C8051F902 na ilość energii zużytej przy poszczególnych cyklach programu korzystano z wewnętrznych oscylatorów: precyzyjnego, pracującego z maksymalną możliwą częstotliwością 24,5 MHz oraz oscylatora o niskim zużyciu energii o maksymalnej częstotliwości taktowania 20 MHz.



Rys. 2. Przebieg prądu badanego układu i jego wartość skuteczna (a) oraz czas cyklu (b) dla transmisji danych przez moduł SPI0 podczas wykonywania programu z resetem programowym



Rys. 3. Przebieg prądu badanego układu i jego wartość skuteczna dla modułu SPI0 podczas wykonywania programu z resetem programowym (a) oraz z cyklicznym skokiem do początku programu głównego (b) z włączonym trybem IDLE

Tabela 4. Parametry czasowo-energetyczne interfejsu SPI dla modułu SPI0, taktowanego oscylatorem precyzyjnym w różnych konfiguracjach programowych

Typ programu:	Cykl	I, mA	Czas, µs	Energia, nJ
Restart programowy	Cały	4,80	17,20	148,44
	Aktywny	5,71	4,80	49,30
Restart programowy, tryb IDLE	Cały	4,24	17,30	132,03
	Aktywny	3,96	5,00	35,66
Skok do etykiety Start, tryb IDLE	Cały	4,34	6,32	49,36
	Wysyłanie	3,35	3,10	18,68

Tabela 5. Parametry czasowo-energetyczne interfejsu SPI dla modułu SPI0, taktowanego oscylatorem o niskim zużyciu energii w różnych konfiguracjach programowych

Typ programu:	Cykl	I, mA	Czas, µs	Energia, nJ
Restart programowy	Cały	4,42	18,3	145,54
	Aktywny	4,54	6	49,00
Restart programowy, tryb IDLE	Cały	4	18,5	133,27
	Aktywny	3,15	6,2	35,16
Skok do etykiety Start, tryb IDLE	Cały	3,39	7,8	47,57
	Wysyłanie	2,64	3,88	18,40

Tabela 6. Parametry czasowo-energetyczne interfejsu SPI dla modułu SPI0, taktowanego oscylatorem precyzyjnym podczas wysyłania 100-bajtowej tablicy danych

Typ programu:	Cykl	I, mA	Czas, µs	Energia, nJ	Energia na 1B, nJ
Restart programowy, tryb Idle – wyjścia analogowe	Cały	3,47	480	2995,69	29,96
	Aktywny	3,43	464	2862,74	28,63

Restart programowy, tryb Idle – wyjścia cyfrowe	Cały	3,45	480	2978,58	29,79
	Aktywny	3,41	464	2846,20	28,46

Tabela 7. Parametry czasowo-energetyczne interfejsu SPI dla modułu SPI0, taktowanego oscylatorem o niskim zużyciu energii podczas wysyłania 100-bajtowej tablicy danych

Typ programu:	Cykl	I, mA	Czas, µs	Energia, nJ	Energia na 1B, nJ
Restart programowy, tryb Idle – wyjścia analogowe	Cały	2,74	600	2952,90	29,53
	Aktywny	2,68	584	2811,67	28,12
Restart programowy, tryb Idle – wyjścia cyfrowe	Cały	2,72	600	2931,50	29,32
	Aktywny	2,68	588	2830,93	28,31

Używany w pracy parametr „cały cykl” oznacza całkowity czas trwania programu, „cykl aktywny” oznacza czas trwania programu po wznowieniu pracy od resetu mikrokontrolera, natomiast „cykl wysyłania” jest czasem wysyłania danych przez interfejs SPI. Wersje programu określone terminem „wyjścia analogowe/cyfrowe” oznaczają w jaki sposób zostały skonfigurowane nieużywane wyprowadzenia mikrokontrolera. Porównując tablice 4 i 5 można zauważyć skrócenie czasu trwania cyklu podczas taktowania mikrokontrolera oscylatorem precyzyjnym, pomimo dodatkowej instrukcji ORL, za pomocą której włączany jest oscylator precyzyjny. Jest to okupione wzrostem poboru prądu i większym zużyciem energii (w każdym przypadku) przy taktowaniu oscylatorem precyzyjnym. Różnica jest jednak nieznaczna i w przypadkach wymaganej dużej precyzji czynności

uzależnionych od stabilności generatora, jego użycie nie spowoduje istotnego wzrostu kosztów energetycznych dla maksymalnych częstotliwości tych generatorów.

Podczas wysyłania większej ilości danych (Tabl. 6 i 7) wartość pobieranego prądu spada dla całego cyklu. Jest to spowodowane tym, że mikrokontroler większość czasu trwania całego cyklu jest w trybie IDLE. Według noty katalogowej układ C8051F902 [1] w tym trybie podczas taktowania oscylatorem precyzyjnym pobiera prąd o wartości 2,1 mA (nie wliczając pracy dodatkowych modułów), natomiast podczas pracy z oscylatorem o niskim zużyciu energii - 1,6 mA. Większy pobór prądu jest spowodowany cyklicznym wyprowadzaniem mikrokontrolera z trybu IDLE po ukończeniu wysyłania jednego bajtu na czas umieszczenia nowych danych w buforze nadawczym oraz oczywiście pracą samego modułu SPIn. Pomiar energii zużywanej w zależności od szybkości taktowania linii SCK zaczyna się od wartości SYSCLK podzielonej przez 4. Jest to największa możliwa szybkość pracy interfejsu SPI, gwarantująca poprawny odczyt danych

[1]. Podczas taktowania wewnętrznym oscylatorem o niskim zużyciu energii, największa możliwa szybkość przesyłania danych to 5 Mb/s. Zmniejszenie szybkości wysyłania danych obniża pobór prądu, jednak wydłuża czas całego cyklu i ostatecznie powoduje większe zużycie energii, mimo uaktywniania trybu IDLE.

Tabela 8. Wpływ szybkości transmisji na własności czasowo-energetyczne interfejsu SPI dla modułu SPI0. Typ programu z restartem programowym oraz trybem IDLE. Pozostałe nieużywane wyprowadzenia skonfigurowane jako cyfrowe

Szybkość transmisji, Mb/s	5	2	1	0,4	0,2	0,1	0,04
I, mA	4,16	3,92	3,65	3,11	2,72	2,38	2,08
Czas, μ s	16,00	18,50	22,60	35,00	55,60	97,00	220,00
Energia, nJ	119,83	130,63	148,30	195,97	271,65	415,12	823,82

Tabela 9. Wpływ napięcia zasilania na własności czasowo-energetyczne badanego układu wykorzystujący do transmisji danych moduł SPI0. Cały cykl programu z restartem programowym oraz trybem Idle

U, V	1,8	2	2,2	2,4	2,6	2,8	3	3,2	3,4	3,6
I, mA	3,94	3,96	3,98	3,98	3,92	3,85	4,14	4,18	4,20	4,24
Czas, μ s	18,5	18,5	18,5	18,5	18,5	18,5	18,5	18,5	18,5	18,5
Energia, nJ	131,45	146,82	162,22	176,90	188,78	199,25	230,27	247,61	264,74	282,51

Na koniec zbadano wpływ zmian napięcia zasilania w przedziale 1,8 V ÷ 3,6 V na stopień zużycia energii przez układ C8051F902 (tabl. 9). Nasuwa się spodziewany wniosek co do wzrostu jej wartości. Z zależności (1) można wykazać, że przy dwukrotnym wzroście wartości napięcia zasilania V_{DD} zwiększy się dwukrotnie wartość energii E , jednak dodatkowy wzrost poboru prądu prowadzi do jeszcze większego wzrostu zużycia energii we wszystkich badanych przypadkach sprzętowych i programowych.

Podsumowanie

Przedmiotowy interfejs został przebadany w różnych konfiguracjach sprzętowych i programowych, dzięki czemu zbadano wpływ poszczególnych ustawień na własności czasowo-energetyczne. Porównano ze sobą wbudowane moduły interfejsu SPI (SPI0 i SPI1), zbadano wpływ zmiany źródła taktowania mikrokontrolera, szybkości transmisji danych oraz przeprowadzono wpływ zmiany wartości napięcia zasilającego na czas trwania programu i zużycie energii. Otrzymane wyniki mają w dużej części swoje odzwierciedlenie w danych katalogowych układu, chociaż w poszczególnych przypadkach uzyskano bardziej precyzyjne i jednoznaczne dane dla konkretnych konfiguracji systemu. Nie potwierdzono także spadku zużycia energii po ustawieniu nieużywanych wyprowadzeń jako wejść analogowych.

Uzyskane w pracy dane, w połączeniu z informacjami na temat energetycznych własności innych modułów peryferyjnych układu C8051F902, pozwolą na optymalizację struktury funkcjonalnej oraz scenariuszy pracy modelu opracowywanego autonomicznego, półpasywnego identyfikatora RFID z funkcją odzysku energii z pól elektromagnetycznych systemów RFID oraz systemów teletransmisyjnych powszechnego użytku [5].

Prace zrealizowano przy wsparciu Narodowego Centrum Badań i Rozwoju (NCBR) w ramach grantu PBS1/A3/3/2012 (U-8795/G/PBS).

LITERATURA

- [1] Dane katalogowe rodziny układów C8051F91x-C8051F90x firmy Silicon Labs
<http://www.silabs.com/Support%20Documents/TechnicalDocs/C8051F91x-90x.pdf>
- [2] Dane katalogowe rodziny układów RTC ABRACON AB18XX
<http://www.abracon.com/PrecisionTiming/AB18XX.pdf>
- [3] Johnson D., Implementing serial bus interfaces using general purpose digital instrumentation, *IEEE Instrumentation Measurement Magazine*, vol. 13, no. 4, 8-13, August 2010
- [4] Rydzewski J., Pomiary oscyloskopowe, WNT, Warszawa, 2007
- [5] Jankowski-Mihułowicz P., Kalita W., Skoczylas M., Węglarski M., Modelling and Design of HF RFID Passive Transponders with Additional Energy Harvester, *International Journal of Antennas and Propagation*, ISSN 1687-5869, doi:10.1155/2013/242840, Vol. 2013, Article ID 242840, 1-10, 2013

Autorzy: dr inż. Kazimierz Kamuda, Politechnika Rzeszowska, Wydział Elektrotechniki i Informatyki, ul. Powstańców Warszawy 12, 35-959 Rzeszów, E-mail: kazik@prz.edu.pl; dr inż. Dariusz Klepacki, Politechnika Rzeszowska, Wydział Elektrotechniki i Informatyki, ul. Powstańców Warszawy 12, 35-959 Rzeszów, E-mail: dklepa@prz.edu.pl; dr inż. Wiesław Sabat, Politechnika Rzeszowska, Wydział Elektrotechniki i Informatyki, ul. Powstańców Warszawy 12, 35-959 Rzeszów; mgr inż. Mariusz Skoczylas, Politechnika Rzeszowska, Wydział Elektrotechniki i Informatyki, ul. Powstańców Warszawy 12, 35-959 Rzeszów; inż. Krzysztof Maj, Politechnika Rzeszowska, Wydział Elektrotechniki i Informatyki, ul. Powstańców Warszawy 12, 35-959 Rzeszów.