

Wydział Elektryczny
Katedra Telekomunikacji i Aparatury Elektronicznej

Instrukcja do zajęć laboratoryjnych z przedmiotu:
Architektura i Programowanie Procesorów Sygnałowych

Numer ćwiczenia: 3

Temat ćwiczenia:

Obsługa i wykorzystanie timerów i portów wejścia/wyjścia.

Opracowali:

dr inż. Dariusz Jańczak
dr inż. Krzysztof Konopko

Białystok 2015

Temat: Obsługa i wykorzystanie timerów i portów wejścia/wyjścia.

1. Cel ćwiczenia

Celem ćwiczenia jest ugruntowanie wiedzy studentów i nabycie przez nich umiejętności z zakresu wykorzystania elementów architektury oraz podstawowych funkcji sygnałowych procesorów sygnałowych na przykładzie zmiennoprzecinkowego procesora TMS320C6713. W ramach zajęć studenci ugruntowują wiedzę na temat zasad obsługi portu I/O i timerów oraz zdobywają umiejętności konfiguracji i wykorzystania tych zasobów. Ponadto studenci nabywają umiejętność realizacji i wykorzystania synchronizacji metodą kontroli bitów stanu (pooling).

2. Zagadnienie do opracowania przed przystąpieniem do zajęć

Przed przystąpieniem do zajęć należy opracować następujące zagadnienia:

- ◆ architektura sygnałowego procesora zmiennoprzecinkowego TMS320C6713,
- ◆ obsługa portu I/O układu DSK procesora TMS320C6713,
- ◆ synchronizacja poprzez kontrolę bitów stanu (pooling), (testowanie, zerowanie i ustawianie bitów, skoki warunkowe),
- ◆ zasady obsługi timera (SPRU582).

3. Przebieg ćwiczenia

Ćwiczenia laboratoryjne prowadzone są w oparciu o zestaw TMS320C6713 DSP Starter Kit (DSK). DSK składa się ze sprzętu i oprogramowania umożliwiającego realizację procedur przetwarzania sygnałów w oparciu o procesor TMS320C6713.

3.1 Obsługa portu wejścia/wyjścia:

W ćwiczeniu należy zrealizować następujące procedury:

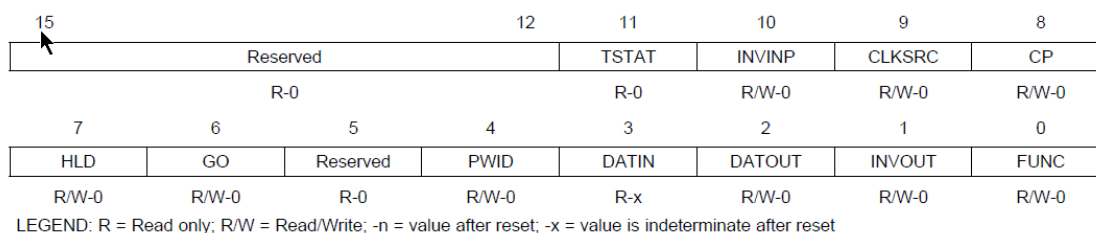
- ◆ napisać procedurę odczytującą i zapisującą wartość z portu I/O (najstarszy bajt pod adresem `0x90080000`),

0x90080000	SW3	SW2	SW1	SW0	LED3	LED2	LED1	LED0
	R 6	R 5	R 4	R 3	W/R 2	W/R 2	W/R 1	W/R 0

- ◆ w oparciu o port I/O napisać następujące procedury:
 - sygnalizującą diodami LED ustawienie przełączników (zmiana położenia przełącznika powoduje zgaszenie lub zapalenie odpowiadającej mu diody),
- ◆ wykorzystując port I/O napisać zrealizować jedną z następujących procedur:
 - sygnalizacja diodami LED w kodzie NKB liczby przełączników w określonej pozycji,
 - zliczającej modulo siedem liczby zmian położenia wybranego przełącznika,

3.1 Konfiguracja i wykorzystanie timerów:

Testując wartość bitu TSTAT (rej. CTL) (pooling) napisać i uruchomić procedurę zapalającą w odstępie sekundowym wybraną diodę.



Uwagi:

- zastosować wewnętrzne taktowanie (CLKSRC=1). Timer jest wówczas taktowany z częstotliwością 225MHz/4,
- ustawić tryb pracy zegarowej bit C/P=1.

3.2 Zadania złożone:

Wykonać zestaw zadań przydzielony grupie przez prowadzącego.

Przykładowe zadania:

- zapalanie w odstępie półsekundowym kolejnych diod (z wykorzystaniem timera0 i adresowania w trybie cyrkularnym),
- zliczanie czasu – wyświetlanie liczby sekund za pomocą diod LED (z wykorzystaniem timera1).
- zliczanie modulo osiem ilości sekund pomiędzy zmianą położenia określonego przełącznika i wyświetlanie czasu za pomocą diod LED (z wykorzystaniem timera0).

4. Sprawozdanie powinno zawierać:

- kody źródłowe wraz z opisem,
- wyniki działania procedur,
- analizę kodów źródłowych stosowanych procedur,
- uwagi i wnioski nasuwające się w trakcie wykonywania ćwiczenia.

5. Wymagania BHP

W trakcie realizacji programu ćwiczenia należy przestrzegać zasad omówionych we wstępie do ćwiczeń, zawartych w: „Regulaminie porządkowym w laboratorium” oraz w „Instrukcji obsługi urządzeń elektronicznych znajdujących się w laboratorium z uwzględnieniem przepisów BHP”. Regulamin i instrukcja są dostępne w pomieszczeniu laboratoryjnym w widocznym miejscu.

6. Literatura

1. Kowalski H. A., *Procesory DSP dla praktyków*, BTC, Legionowo, 2013
2. Dąbrowski A. (red.) *Przetwarzanie Sygnałów Przy użyciu Procesorów Sygnałowych*, Wyd. Politechniki Poznańskiej, Poznań 2000.
3. Smith S. W., *Cyfrowe przetwarzanie sygnałów: praktyczny poradnik dla inżynierów i naukowców*, Wydawnictwo BTC, Warszawa, 2007.
4. Texas Instruments, *TMS320C6000 DSP Peripherals Overview*, 2007.
5. Texas Instruments, *TMS320C6000 DSP 32-Bit Timer Reference Guide*, 2005.
6. Texas Instruments, *TMS320C67x DSP CPU and Instruction Set Reference Guide*, 2006.
7. Texas Instruments, *TMS320C6000 Programmer's Guide*, 2006.
8. Kehtarnavaz, N., *Real-Time Digital Signal Processing: Based on the TMS320C6000*, Newnes, 2005.
9. Welch T. B., Wright C.H.G., Morrow M.G., *Real-time Digital Signal Processing from Matlab to C with the TMS320C6x DSPs*, Taylor & Francis, 2012.

Dodatek 2.

```
; konfiguracja timera
CTL .set 01940000h
PRD .set 01940004h
dz_cz .set 56250000 ;dzielnik częstotliwości zeg. CPU do timera
```

```
.text
;
; ustawienie timera
; przerwania co 1s
MVKL.S2 CTL, B1 ; w B1 adres CTL
|| MVKL.S1 dz_cz, A10 ;dzielnik częstotliwości zeg. CPU do timera
MVKH.S2 CTL, B1 ; w B1 adres CTL
|| MVKH.S1 dz_cz, A10 ;dzielnik częstotliwości zeg. CPU do timera
MVK.S2 201h, B0 ;słowo sterujące
|| ADD.L1x 4, B1, A0 ; w A0 adres PRD
STW.D2 B0, *B1 ; zapis do CTL
|| STW.D1 A10, *A0 ; zapis do PRD
MVK.S2 201h, B0 ;słowo sterujące
STW.D2 B0, *B1 ; zapis do CTL
|| MVK.S2 02C1h, B0 ;słowo sterujące
STW.D2 B0, *B1 ; zapis do CTL - start timera
```