Politechnika Białostocka

Wydział Elektryczny
Katedra Telekomunikacji i Aparatury Elektronicznej

Instrukcja do zajęć laboratoryjnych z przedmiotu:

Architektura i Programowanie Procesorów Sygnałowych

Kod: TS1C510210

Numer ćwiczenia: 7

Temat ćwiczenia:

Realizacja procedur generacji i modulacji sygnałów przy użyciu procesora sygnałowego

Opracowali:
dr inż. Dariusz Jańczak

Białystok 2017

Architektura i Programowanie Procesorów Sygnałowych

Temat: Realizacja procedur generacji i modulacji sygnałów przy użyciu procesora sygnałowego

1. Cel ćwiczenia

Celem ćwiczenia jest pogłębienie i ugruntowanie wiedzy studentów oraz nabycie przez nich umiejętności z zakresu współpracy procesora sygnałowego z układami wejścia/wyjścia, a także współpracy z przetwornikami A/C i C/A. Ćwiczenie prowadzone jest z wykorzystaniem modułu DSK TMS320C6713 i dostępnych w module przetworników TLV320AIC23. Studenci nabywają również umiejętność formułowania algorytmów i realizacji prostych zadań cyfrowego przetwarzania sygnałów takich jak: generacja sygnałów metodami algorytmicznymi i tablicowymi oraz modulacja AM. Dodatkowym aspektem dydaktycznym ćwiczenia jest zdobycie i pogłębienie umiejętności w zakresie opracowania dokumentacji dotyczącej realizowanego zadania inżynierskiego oraz umiejętności przygotowania tekstu zawierającego omówienie wyników realizacji tego zadania.

2. Zagadnienie do opracowania przed przystąpieniem do zajęć

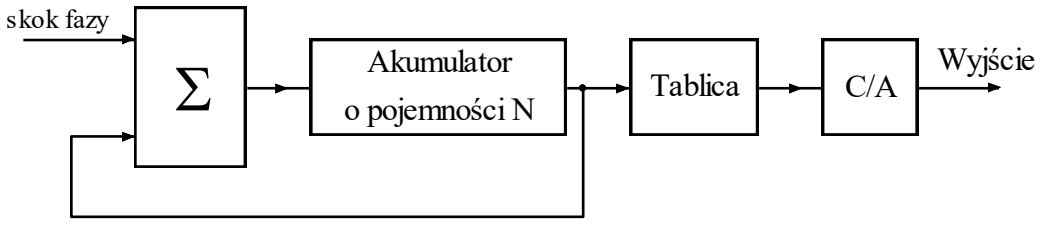
Przed przystąpieniem do zajęć należy opracować następujące zagadnienia:

- metody generacji sygnałów w systemach DSP,
- modulacja AM,
- wykorzystanie timera i systemu przerwań w zadaniach przetwarzania sygnałów,
- zasady współpracy procesora DSP z przetwornikami A/C i C/A,
- współpraca procesora TMS320C6713 z układem TLV320AIC23 (z przetwornikami A/C, C/A) na płycie DSK.

3. Przebieg ćwiczenia

3.1 Generacja przebiegu sinusoidalnego metodą tablicową

Napisać program generujący sygnał sinusoidalny metodą syntezy tablicowej. Podstawą tej metody jest tablica zawierająca *N* próbek jednego okresu funkcji sinus.



Rys. 1 Układ generatora z akumulatorem fazy

W każdym okresie próbkowania następuje dodanie do poprzedniej fazy (stanu akumulatora) kroku fazy *k* i wysłanie na wyjście próbki o fazie (numerze w tablicy)

równej uzyskanej sumie. Dodawanie odbywa się modulo *N*, a więc po dojściu do końca tablicy następuje powrót na jej początek (można wykorzystać adresowanie kołowe). Podstawowa częstotliwość uzyskanego w ten sposób przebiegu wynosi:

$$F_{WY} = \frac{k}{N} * F_S \qquad 0 < k < N/2$$

W ogólnym przypadku skok fazy *k* nie musi być liczbą całkowitą. W takiej sytuacji wysyłana jest próbka o numerze równym całkowitej części aktualnej fazy. Spowoduje to nieznaczny wzrost zawartości harmonicznych, ale uzyskujemy większą rozdzielczość generowanych częstotliwości, która wynosi:

$$\Delta F_{WY} = \frac{\Delta k}{N} * F_S$$

Plik ze zdefiniowanymi *N* próbkami sinusa (w obu kanałach) można wygenerować w Matlabie programem *gen_sin('sinus.dat',N),* a wynikowy plik *sinus.dat* można dołączyć do programu asemblerowego dyrektywą **.include**. Algorytm pracy programu generatora należy oprzeć na adresowaniu w trybie cyrkularnym. Synchronizacja okresu próbkowania przy wykorzystaniu przerwania od odbiornika portu szeregowego.

3.2 Generacja przebiegów okresowych:

Napisać program generacji wybranego sygnału np:

- prostokątny,
- piłokształtny,
- trójkątny,

Częstotliwość sygnału 500Hz. Kolejne próbki synchronizować przez timer.

3.3 Modulator amplitudowy

Zmodyfikować napisany w punkcie 3.1 program tak, aby generowana sinusoida była modulowana amplitudowo sygnałem z przetwornika A/C. W tym celu w każdym cyklu próbkowania należy:

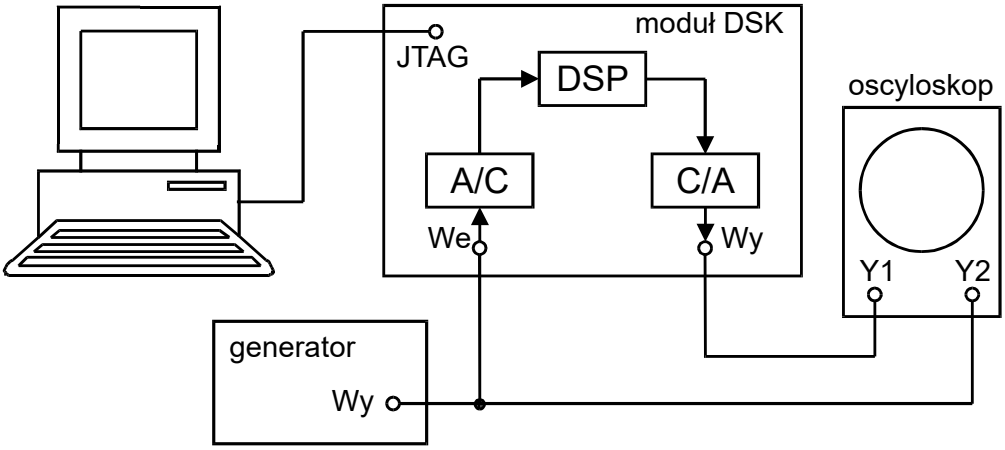
- pomnożyć próbkę wejściową przez współczynnik głębokości modulacji,
- do otrzymanego iloczynu dodać jedynkę,
- uzyskaną sumę przemnożyć przez aktualną wartość nośnej,
- rezultat wysłać do przetwornika C/A.

3.4 Generacja przebiegu sinusoidalnego metodą algorytmiczną

Napisać i uruchomić program generujący sygnał sinusoidalny metodą algorytmiczną. W tym celu można wykorzystać równanie różnicowe opisujące filtr, którego odpowiedź impulsowa będzie miała formę oscylacji niegasnących.

Realizacja ćwiczeń

Badania eksperymentalne należy przeprowadzić w układzie połączonym według schematu przedstawionego na Rys. 2. Amplituda napięcia podawanego z generatora na wejście układu A/C nie powinna przekraczać 2V.



Rys. 2. Schemat połączeń stanowiska laboratoryjnego DSP

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. Zieliński T. (red.), *Cyfrowe przetwarzanie sygnałów w telekomunikacji: podstawy, multimedia, transmisja*, PWN, Warszawa, 2014
2. Zieliński T., *Cyfrowe przetwarzanie sygnałów: od teorii do zastosowań*, WKŁ, Warszawa, 2009.
3. Smith S. W., *Cyfrowe przetwarzanie sygnałów: praktyczny poradnik dla inżynierów i naukowców*, Wydawnictwo BTC, Warszawa, 2007.
4. Kowalski H. A., *Procesory DSP w przykładach,* BTC, Legionowo, 2012.
5. Dąbrowski A. (red.) *Przetwarzanie Sygnałów Przy użyciu Procesorów Sygnałowych*, Wyd. Politechniki Poznańskiej, Poznań 2000.
6. Texas Instruments, *TMS320C6000 Optimizing Compiler User's Guide*, 2017.
7. Texas Instruments, *TMS320C6000 DSP Peripherals Overview*, 2009.
8. Texas Instruments, TMS320C6000 Chip Support Library API Reference Guide, 2004.
9. Texas Instruments, TMS320C67x DSP Library Programmer's Reference Guide, 2010.
10. Texas Instruments, *TLV320AIC23 Stereo Audio CODEC Data Manual*, 2004.

11. Kehtarnavaz, N., *Real-Time Digital Signal Processing: Based on the TMS320C6000*, Newnes, 2005.
12. 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.