

Krzysztof CZYŻ, Tomasz ZAJĄC
POLITECHNIKA ŚLĄSKA, INSTYTUT AUTOMATYKI,
ul. Akademicka 16, 44-100 Gliwice

Układ aktywnej redukcji hałasu pracujący pod kontrolą systemu uClinux

Dr inż. Krzysztof CZYŻ

Zatrudniony na stanowisku adiunkta w Instytucie Automatyki Politechniki Śląskiej. Posiada szerokie doświadczenie praktyczne z zakresu cyfrowego przetwarzania sygnałów poparte udziałem w wielu pracach badawczych nad: własnościami algorytmów filtracji cyfrowej, sterowania adaptacyjnego, systemami wbudowanymi oraz systemami operacyjnymi czasu rzeczywistego.



e-mail: krzysztof.czyz@polsl.pl

Mgr inż. Tomasz ZAJĄC

Absolwent wydziału Automatyki, Elektroniki i Informatyki Politechniki Śląskiej. Jego zainteresowania obejmują cyfrowe przetwarzanie sygnałów, systemy telekomunikacyjne oraz systemy operacyjne czasu rzeczywistego.



e-mail: tomasz.zajac@post.pl

Streszczenie

Idea Aktywnej Redukcji Hałasu (ARH) jest związana z redukcją niepożądanego niskoczęstotliwościowego dźwięku, w rezultacie tworzona jest przestrzenna strefa cisy wokół wybranego mikrofonu. Artykuł opisuje praktyczną implementację systemu ARH pracującego pod kontrolą systemu operacyjnego uClinux na dwurdzeniowym procesorze sygnałowym Blackfin firmy Analog Devices. System operacyjny uClinux okazuje się być tanim, dynamicznym środowiskiem rozwojowym dla procesorów sygnałowych Blackfin, dodatkowo wykorzystując wielordzeniowe procesory jest możliwe pokonanie głównej wady Linux'a mianowicie braku funkcjonalności systemu operacyjnego czasu rzeczywistego niezbędnej do zastosowań przetwarzania sygnałów. Można to osiągnąć poprzez oddzielenie kodu który musi być wykonywany w czasie rzeczywistym i uruchamianie go na osobnym rdzeniu podczas gdy drugi rdzeń jest wykorzystywany przez systemem operacyjny Linux. Zastosowanie systemu operacyjnego Linux w implementacji układu ARH przynosi szereg korzyści zwłaszcza zwiększa możliwości rozwoju i debugingu oprogramowania, Linux daje również dostęp do ogromnej ilości otwartego oprogramowania. W prezentowanym rozwiązaniu w oparciu o system operacyjny Linux stworzono aplikację integrującą funkcje web-serwera z układem aktywnej redukcji hałasu, dynamicznie tworzone strony internetowe tworzą graficzny interfejs użytkownika systemu ARH dostępny poprzez przeglądarkę internetową.

Słowa kluczowe: aktywna redukcja hałasu, przetwarzanie sygnałów, systemy sterowania, systemy wbudowane.

Active Noise Control running under uClinux

Abstract

The idea of ANC is concerned with attenuation of unwanted low frequency sound (noise). As a result, a three-dimensional zone of quiet is created around the selected microphones. The paper describes ANC system implementation running under the uClinux system on dual core Analog Devices Blackfin DSP. The uClinux is proving to be a low-cost, dynamic development environment. On multi-core Blackfin DSPs, it is possible to overcome the main disadvantage of Linux - lack of hard-real time functionality indispensable for signal processing applications. It can be achieved by separation of the real-time signal processing code and handling it on a dedicated core, while the other core is running Linux. Such a solution offers advantages to designers, most notably by opening up a wide range of applications, drivers and protocols which often are open-source or free software. In most cases, only a compilation or some minor tweaking is necessary to get that software up and running. The advantages for using Linux in the ANC system are ease of use, better system control, easy possibility of creating web based user interfaces and improved debugging capabilities. In this particular application, use of Linux enables a Web server integration into the ANC system. Dynamically created web pages can be accessed by any web browser and become the graphical user interface to the ANC system.

Keywords: active noise control, control systems, digital signal processing, discrete-time systems, embedded systems.

1. Wprowadzenie

Tradycyjne metody redukcji hałasu takie jak np. izolacja akustyczna są metodami pasywnym, szczególnie efektywnymi dla hałasu o częstotliwości powyżej 500 Hz [7]. Jest to spowodowane tym, że grubość izolacji akustycznej powinna być odwrotnie proporcjonalna do częstotliwości redukowanego hałasu. Dlatego metody pasywne są uważane za nieefektywne i często zbyt kosztowne w zastosowaniu do redukcji hałasu niskoczęstotliwościowego.

W przeciwieństwie do metod pasywnych, aktywne metody Redukcji hałasu (ARH) są bardzo efektywne dla zakłóceń niskoczęstotliwościowych z zakresu częstotliwości poniżej 500 Hz. Zastosowanie układów ARH dla wyższych częstotliwości jest ograniczone głównie rozmiarami tworzonych stref cisy, których rozmiar jest odwrotnie proporcjonalny do częstotliwości redukowanego zakłócenia.

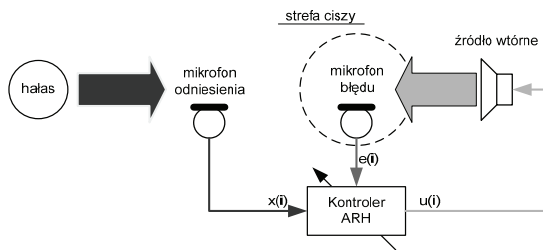
Ideę układów kompensacji stosowanych na potrzeby aktywnej redukcji hałasu niskoczęstotliwościowego przedstawiono na rys. 1. Pomieszczenie zakłócone jest hałasem (zakłóceniem) generowanym przez źródło pierwotne, hałas pierwotny powinien być stłumiony za pomocą źródła wtórnego (głośnik sterujący). Hałas pierwotny jest mierzony za pomocą mikrofonu odniesienia z wyprzedzeniem, zanim dotrze poprzez przestrzeń akustyczną do mikrofonu błędu. Sygnały z mikrofonu odniesienia i błędu są wykorzystywane następnie do strojenia adaptacyjnego kompensatora oraz do monitorowania jakości działania układu ARH. Wypracowany przez kompensator sygnał sterowania pobudza głośnik sterujący, który generuje falę akustyczną. Dociera ona do mikrofonu błędu wpływając na hałas pierwotny. Zadaniem algorytmu adaptacyjnego jest taki dobór wag kompensatora, aby zminimalizować wartość średniokwadratową sygnału mierzonego przez mikrofon błędu [4, 7]. W rezultacie czego wokół mikrofonu błędu powstaje lokalna przestrzenna strefa cisy.

Celem artykułu jest zaprezentowanie praktycznej implementacji systemu ARH na dwurdzeniowym procesorze sygnałowym pracującym pod kontrolą systemu operacyjnego uClinux. System operacyjny uClinux jest systemem operacyjnym zorientowanym na zastosowania wbudowane. Tworzenie systemu wbudowanego na bazie Linuxa przynosi szereg korzyści, zwłaszcza poprzez umożliwienie dostępu do szerokiej gamy aplikacji, sterowników i protokołów dostępnych w formie otwartego oprogramowania [6].

Zastosowanie systemu operacyjnego Linux podczas implementacji układu ARH przyniosło szereg korzyści zwłaszcza zwiększa możliwości rozwoju i debugingu oprogramowania, w oparciu o dostępne aplikacje Linuxa stosunkowo szybko można stworzyć elementy graficznego interfejsu użytkownika, czy systemu przekazywania poprzez Internet sygnałów pochodzących z działającego układu sterowania. W prezentowanym rozwiązaniu w oparciu o system operacyjny Linux stworzono aplikację integrującą funkcje web-serwera z układem aktywnej redukcji hałasu, dynamicznie tworzone strony internetowe tworzą graficzny interfejs użytkownika systemu ARH dostępny poprzez przeglądarkę internetową.

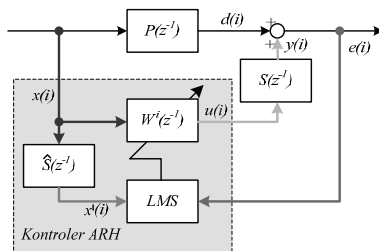
2. Układ aktywnej redukcji hałasu

Układ ARH ze sprzężeniem w przód używa sygnałów pochodzących z mikrofonów odniesienia i błędu do dostosowania parametrów kontrolera. Sygnał sterujący generowany przez kontroler podawany jest na głośnik wtórny, fala akustyczna generowana przez kontroler propaguje się poprzez ścieżkę wtórną do mikrofonu błąd gdzie wpływa na poziom pierwotnego hałasu. Celem algorytmu adaptacji jest dostosowanie parametrów kontrolera w taki sposób, aby zminimalizować średniokwadratową wartość sygnału pochodzącego z mikrofonu błędu. W wyniku redukcji hałasu powstaje wokół mikrofonu błędu lokalna przestrzenna strefa ciszy (rys. 1). Schemat blokowy adaptacyjnego układu ARH ze sprzężeniem w przód jest pokazany na rys. 2.



Rys. 1. Schemat funkcjonalny jednokanałowego adaptacyjnego układu aktywnej redukcji hałasu

Fig. 1. Scheme of a single channel adaptive ANC system



Rys. 2. Schemat blokowy adaptacyjnego układu ARH ze sprzężeniem w przód

Fig. 2. Block diagram of a feedforward ANC system

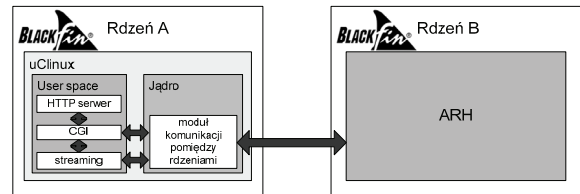
W przedstawionym systemie jako kontroler używany jest filtr adaptacyjny $W(z^{-1})$. Sygnał sterujący $u(i)$ jest otrzymywany w wyniku filtracji sygnału odniesienia $x(i)$ poprzez filtr $W(z^{-1})$, następnie sygnał ten pobudza głośnik generujący. Przedstawiony system ARH został stworzony na bazie adaptacyjnego algorytmu Fx-LMS (Filtered-x Least Mean Squares), w algorytmie tym współczynniki filtra $W(z^{-1})$ wyznaczone są na podstawie sygnału błędu $e(i)$ oraz sygnału odniesienia $x(i)$ (lub jego oszacowania) przefiltrowanego przez model ścieżki wtórnej $S(z^{-1})$. Współczynniki filtra cyfrowego $W(z^{-1})$ mogą być strojne za pomocą różnych modyfikacji algorytmu LMS [2, 3, 4, 5]. Zachowanie adaptacyjnego algorytmu Fx-LMS, a w szczególności jego zbieżność, jest kontrolowana przez odpowiedni dobór parametru kroku adaptacji, jednak warto podkreślić że nie jest to zadanie łatwe i wymaga najczęściej podejścia „prób i błędów” [7, 8]. Dlatego zaprezentowany układ rozbudowano o graficzny interfejs użytkownika wspomagający proces parametryzacji algorytmu sterującego.

3. Implementacja układu ARH

Procesory z rodziny Analog Devices Blackfin są procesorami zaprojektowanymi specjalnie dla potrzeb obliczeń i przetwarzania sygnałów. Procesory te łączą 32-bitowy zestaw instrukcji typu RISC z dwoma 16-bitowymi jednostkami mnożąco-sumacyjnymi (MAC). Taka architektura umożliwia bardzo wydajne wykonywanie kodu związanego z przetwarzaniem sygnałów jak i aplikacji ogólnego przeznaczenia zwłaszcza działających pod kontrolą popularnych systemów operacyjnych. Procesor BF561 dodatkowo rozszerza funk-

cyjność procesorów z rodziny Blackfin oferując dwa symetryczne rdzenie pracujące maksymalnie z częstotliwością 600 MHz [1].

Implementacja systemu ARH składa się z kilku aplikacji wykonywanych na obu rdzeniach procesora Blackfin BF561. Algorytm ARH musi być wykonany w czasie rzeczywistym, dlatego został on zrealizowany na drugim rdzeniu procesora (Core B). Natomiast na pierwszym rdzeniu (Core A) działa system operacyjny uClinux wraz z aplikacjami odpowiedzialnymi za interfejs użytkownika i rejestrację danych. Wymiana danych między rdzeniami odbywa się poprzez dedykowany moduł jądra Linuksa oraz zestaw funkcji odpowiedzialnych za wymianę danych na drugim rdzeniu. Interfejs użytkownika jest dostępny poprzez przeglądarkę internetową i jest oparty na asynchronicznych żądaniach serwera HTTP obsługiwanych przez programy CGI. Schemat wymiany danych pomiędzy elementami systemu zaprezentowano na rys. 3.



Rys. 3. Schemat ścieżek wymiany danych pomiędzy poszczególnymi elementami systemu ARH

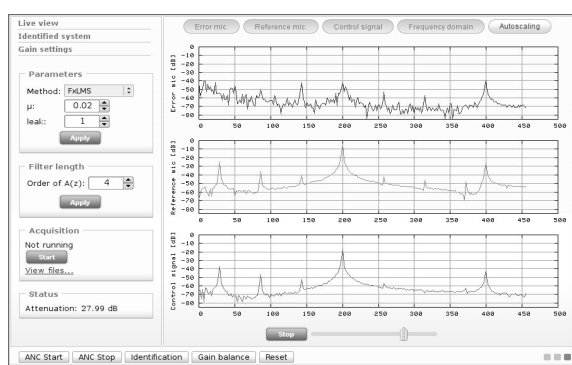
Fig. 3. Scheme of data exchange paths between elements of the ANC system

Aplikacja realizująca w czasie rzeczywistym algorytm sterowania systemu ARH działa z pominięciem systemu operacyjnego jednak jej uruchomienie jest inicjowane z poziomu rdzenia pracującego pod kontrolą systemu operacyjnego Linux. Rozwiązanie takie jest podyktowane koniecznością zagwarantowania pracy systemu ARH ze stałym okresem próbkowania wynoszącym 2 kHz oraz wyznaczenia wartości kolejnego sterowania w czasie krótszym niż okres próbkowania. Po uruchomieniu aplikacja inicjuje niezbędny sprzęt, a następnie przechodzi do nieskończonej pętli realizującej automat stanu oczekując na polecenia przekazywane przez oprogramowanie graficznego interfejsu użytkownika. W chwili gdy zarejestrowana zostanie zdefiniowana liczba próbek sygnałów odniesienia i błędu przez przetworniki analogowo-cyfrowe generowane jest przerwanie, próbki są decymowane i przekazane do algorytmu sterowania. Algorytm sterowania został zoptymalizowany w assemblerze, dzięki czemu udało się zwiększyć 97-krotnie szybkość obliczeń względem funkcji napisanej w czystym języku C. Sygnał sterujący jest wyznaczany poprzez przefiltrowanie sygnału odniesienia przez filtr $W(z^{-1})$, wyznaczenie wartości pojedynczego elementu filtra wykonywane jest przez procesor w przeciągu dwóch cykli zegarowych. W podobnym czasie wyznaczana jest przez procesor korekta pojedynczego współczynnika filtra adaptacyjnego. Następnie sygnał sterujący jest interpolowany i przekazywany do przetwornika cyfrowo-analogowego.

System operacyjny Linux dzieli pamięć operacyjną na dwa odrębne regiony: przestrzeń jądra oraz przestrzeń użytkownika. Przestrzeni pamięci jądra systemu jest miejscem w którym przechowywany jest kod jądra systemu operacyjnego, za pośrednictwem tej przestrzeni system dostarcza również swoje usługi jako sterownik urządzeń. Kod wykonywany w przestrzeni jądra ma bezpośredni dostęp do urządzeń wejścia-wyjścia. Natomiast przestrzeń pamięci użytkownika w systemie Linux jest przestrzenią pamięci, w której uruchamiane są aplikacje użytkownika. Aplikacje te również do swojego poprawnego działania wymagają dostępu do urządzeń wejścia-wyjścia, jednakże bezpośredni dostęp do urządzeń wejścia-wyjścia nie jest możliwy i dobywa się za pośrednictwem usług jądra systemu. W celu zapewnienia komunikacji między rdzeniami stworzono dedykowany moduł jądra ICC (Inter-Core Communication), umożliwia on dwukierunkową komunikację między rdzeniami przez cykliczne przekazywanie struktur zawierających dane pochodzące z układu ARH oraz zestawy komend wysłanych do układu ARH przez programy z przestrzeni użytkownika. Wymiana danych odbywa się w opar-

ciu o pamięć L2 współdzieloną przez oba rdzenie. Gdy bufor zawiera dane, generowane jest przerwanie do danego rdzenia i dane kopiowanie są do odpowiedniego bufora wewnętrznego.

Aplikacja tworząca interfejs użytkownika jest zlokalizowana w przestrzeni użytkownika systemu Linux. Została stworzona na podstawie dwóch ogólnodostępnych programów: serwera Boa oraz programu Gnuplot. Serwer Boa to prosty i szybki serwer HTTP rozpowszechniany na licencji GNU GPL. Boa jest jedno-wątkowym serwerem HTTP, oznacza to, że w przeciwieństwie do tradycyjnych serwerów WWW, nie używa funkcji fork() do tworzenia swojej kopii dla obsługi wielu połączeń. Obsługa wielu połączeń jest realizowana poprzez mechanizm wewnętrznego przełączania, funkcja fork() jest wykorzystywana wyłącznie do wywoływania programów CGI. Natomiast Gnuplot jest programem umożliwiającym tworzenie wykresów, sposób tworzenia wykresu jest określany za pomocą argumentów przekazywanych do programu. Gnuplot obsługuje również eksport wykresu do wielu popularnych formatów plików graficznych.



Rys. 4. Interfejs użytkownika systemu ARH - przebiegi sygnałów przetwarzanych przez procesor DSP

Fig. 4. ANC system web based user interface live view subpage

Wymiana danych pomiędzy modułem jądra ICC a interfejsem użytkownika oparta jest na programach CGI działających wewnątrz serwera HTTP. Dodatkowo w tle w przestrzeni użytkownika działa program odpowiedzialny za gromadzenie danych pochodzących z układu ARH. Zadanie to nie mogło być zrealizowane w oparciu o programy CGI ponieważ rejestracja danych może być aktywna przez długie okresy czasu, podczas gdy procesy działające jako programy CGI, zostaną zamknięte po zerwaniu połączenia z serwerem lub po upływie określonego czasu. Gdy użytkownik uaktywni funkcję rejestracji danych program pobiera dane z kontrolera ARH działającego na drugim rdzeniu za pośrednictwem wcześniej opisanego modułu jądra ICC. Dane zapisywane są do pliku tymczasowego, po zakończeniu rejestracji plik jest kopiowany do katalogu z którego użytkownik może go pobrać lub usunąć.

Programy CGI są również używane do kontroli i parametryzacji układu ARH. Za ich pomocą wysyłane są podstawowe komendy, takie jak startu czy zatrzymania układu ARH, polecenia identyfikacji modeli poszczególnych torów, zmiany parametrów czy wzmocnień dla układu codeca audio. Co więcej, programy CGI są odpowiedzialne za tworzenie na bieżąco wykresów sygnałów błędów, odniesienia oraz sterowania układu ARH, w tym przypadku programy generują plik zawierający parametry wykresu i wywołany jest program Gnuplot, a w wyniku jego działania tworzony jest plik graficzny zawierający konkretny wykres.

4. Interfejs użytkownika

Interfejsem użytkownika dla układu aktywnej redukcji hałasu jest okno przeglądarki internetowej, przykładowy wygląd interfejsu został zaprezentowany na rys. 4. Okno podzielone jest na trzy obszary: po lewej stronie prezentowane są parametry systemu ARH oraz aktualnie osiągnięty poziom redukcji hałasu, na dole okna umiejscowione zostały przyciski sterujące działaniem systemu. Pozostała część okna jest używana przez wtyczki generujące dynamiczną zawartość taką jak

wykresy przebiegu w czasie sygnałów przetwarzanych przez procesor DSP. Interfejs użytkownika składa się z stron: podglądu na bieżąco sygnałów, identyfikacji oraz parametryzacji układu.

Strona podglądu sygnałów umożliwia obserwację na bieżąco przebiegu w czasie sygnałów błędów, odniesienia i sygnału sterującego. Wszystkie obserwowane sygnały mogą być prezentowane również w postaci gęstości widmowych mocy. Bezpośrednio nad wykresami znajdują się przyciski sterujące sposobem w jaki wyświetlane są przebiegi. Na stronie identyfikacji modeli znajdują się kontrolki wspomagające identyfikację modelu odpowiedzi impulsowej ścieżki wtórnej $S_k(z^{-1})$, zidentyfikowany model może być zapisany na dysk komputera użytkownika. Parametry układu ARH prezentowane są po lewej stronie okna. Interfejs użytkownika umożliwia wybór algorytmu sterowania, jego parametryzację, użytkownik może zmienić wielkość kroku adaptacji oraz długość filtra adaptacyjnego. Rejestracja sygnałów jest kontrolowana przez przyciski znajdujące się pod parametrami układu ARH, po naciśnięciu przycisku rozpoczyna się nagrywanie sygnałów, a wybrane sygnały mogą być zapisane na dysku komputera użytkownika w postaci plików.

5. Podsumowanie

System operacyjny *uClinux* okazuje się być tanim, dynamicznym środowiskiem rozwojowym dla procesorów sygnałowych Blackfin, dodatkowo wykorzystując wielordzeniowe procesory jest możliwe pokonanie głównej wady Linux'a mianowicie braku funkcjonalności systemu operacyjnego czasu rzeczywistego niezbędnej do zastosowań przetwarzania sygnałów. Można to osiągnąć poprzez oddzielenie kodu który musi być wykonywany w czasie rzeczywistym i uruchamianie go na osobnym rdzeniu podczas gdy drugi rdzeń jest wykorzystywany przez system operacyjny Linux. Takie rozwiązanie przynosi szereg korzyści, zwłaszcza poprzez umożliwienie dostępu do szerokiej gamy aplikacji, sterowników i protokołów dostępnych w formie otwartego oprogramowania, którego wykorzystanie w większości przypadków sprowadza się jedynie kompilacji i/lub drobnych modyfikacji. W prezentowanym rozwiązaniu w oparciu o system operacyjny Linux stworzono aplikację integrującą funkcje web-serwera z układem aktywnej redukcji hałasu, dynamicznie tworzone strony internetowe tworzą graficzny interfejs użytkownika systemu ARH dostępny poprzez przeglądarkę internetową.

Praca naukowa finansowana ze środków budżetowych na naukę w roku 2010 jako projekt badawczy N N514 232037.

6. Literatura

- [1] Analog Devices, Analog Devices Home Page, Blackfin Processors, 2010.
- [2] Czyż K.: Active Noise Control Systems with Nonuniform Signal Sampling. Jacek Skalmierski Computer Studio, 2007.
- [3] Douglas S.C.: An Efficient Implementation of the Modified Filtered-X LMS Algorithm. Department of Electrical Engineering, University of Utah, Salt Lake City, 1997.
- [4] Elliott S.J., Boucher C.C.: Interaction Between Multiple Feed-forward Active Control Systems. IEEE Transactions on Speech and Audio Processing, 2(4), 1994.
- [5] Elliott S.J., Sutton T.J.: Performance of Feedforward and Feedback Systems for Active Noise Control. IEEE Transactions on Speech and Audio Processing, 4(3), 1996.
- [6] Hennerich M.: *uClinux* as an Embedded OS on a DSP. Embedded Systems Design, 2004.
- [7] Kuo S.M., Morgan D.R.: Active Noise Control Systems. Algorithms and DSP Implementations. J. Wiley & Sons, New York, 1996.
- [8] Rupp M., Sayed A.H.: Robust FxLMS Algorithms with Improved Convergence Performance. IEEE Transactions on Speech and Audio Processing, 6(1), 1998.