

РЕАЛІЗАЦІЯ ЦИФРОВИХ ФІЛЬТРІВ НА ESP32 У СЕРЕДОВИЩІ ARDUINO IDE

Мачоніс Т.С.

Науковий керівник – к.т.н., проф. Зубков О.В.

Харківський національний університет радіоелектроніки, каф. МТС

м. Харків, Україна

e-mail: tomas.machonis@nure.ua

The work analyzes tasks and electronic devices that require the implementation of digital filtering. A review of the characteristics of popular ESP32 modules has been completed. The coefficients of three low-pass filters were synthesized using a specialized utility in Matlab. Using the developed digital filter program, the filtering time, access to the analog-to-digital and digital-to-analog converter in the Arduino IDE were measured, and the microprocessor core load was calculated. Disadvantages of the Arduino IDE have been identified when working with the internal peripherals of the ESP32 module.

Цифрова фільтрація сигналів в мікроконтролерах, зокрема ESP32, відіграє ключову роль у обробці даних з датчиків та забезпеченні стабільності та точності вимірювань. Одним з основних завдань цифрової фільтрації є зменшення впливу шумів на сигнал. ESP32 може використовувати цифрові фільтри для підтримки стійкості вимірювань у шумних середовищах. Іноді вимірювання з датчиків можуть мати різкі зміни або високочастотні складові. Цифрові фільтри дозволяють плавно згладити ці зміни та забезпечити більш стабільні вимірювання. Аналогові датчики температури, вологості і т.д. можуть бути чутливими до пульсацій у мережі живлення, що призводить до пульсацій результатів вимірювань. Цифрові фільтри допомагають вирішити цю проблему, згладжуючи коливання результатів. У якості приклада можна навести вимірювання прискорення за допомогою акселерометра, при якому може виникати велика кількість шумів. Цифрові фільтри використовуються для зменшення цього шуму та витягнення корисних сигналів.

Модулі ESP32 мають вбудовані цифрові інтерфейси SPI, I2C за допомогою яких можна підключати цифрові датчики (наприклад, акселерометр), а також вбудований 12 розрядний аналого-цифровий (АЦП) перетворювач за допомогою якого можна підключити аналогові датчики.

Двоядерний 32-розрядний LX6 мікропроцесор модуля ESP32 працює на тактовій частоті до 240МГц. Підтримка операцій з плаваючою комою та достатньо висока тактова частота дозволяє цьому мікропроцесору реалізувати функцію цифрової фільтрації сигналів. Компанія виробник модулів ESP32 навіть надає бібліотеку з сукупністю методів цифрової фільтрації, але вона доступна тільки для розробки програмного забезпе-

чення у Visual Studio Code. Найбільш популярним середовищем розробки програмного забезпечення зараз є Arduino IDE. Основа популярності цієї IDE – її простота використання та написання програми, а недоліком є обмеженість використання апаратних можливостей мікропроцесора. Саме тому метою дослідження була ефективність реалізації цифрових фільтрів на ESP32 у середовищі Arduino IDE.

Для аналізу були обрані 3 фільтри з кінцевою імпульсною характеристикою (FIR). Частота дискретизації аналогового сигналу була завдана 10кГц. Фільтри мали граничну частоту смуги пропускання 500Гц та полосу затримання від 2,5кГц, але відрізнялись величиною придушення частот у межах полоси затримання. Перший фільтр мав величину придушення 40дБ, другий – 87дБ, третій – 154дБ. Відповідно вони відрізнялись кількістю коефіцієнтів: в першому фільтрі їх було 13, в другому – 32, у третьому – 64. Синтез фільтрів був виконаний з використанням спеціалізованої утиліти Matlab (Filter Builder).

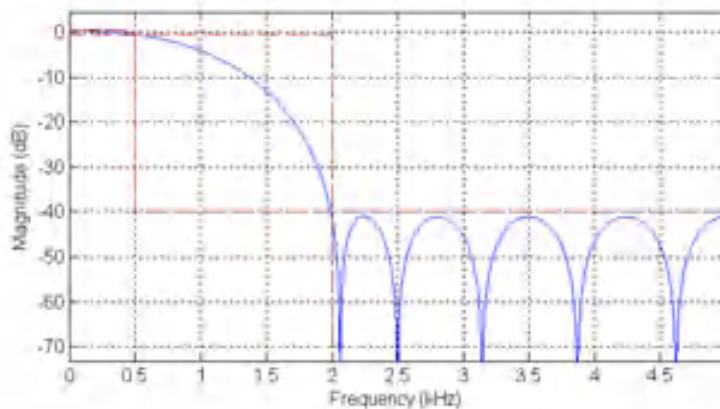


Рисунок 1. АЧХ першого фільтра низьких частот

Вигляд амплітудно-частотної характеристики (АЧХ) першого фільтра наведено на рисунку 1.

У якості апаратної платформи було обрано модуль ESP-WROOM-32 з тактовою частотою 240МГц. В мікропроцесорі існують вбудовані: АЦП та цифро-аналоговий перетворювач (ЦАП). За допомогою АЦП в дослідженнях зовнішній аналоговий

сигнал від генератора подавався на вхід АЦП, а кодові комбінації, що відповідають відлікам аналогового сигналу подавались на вхід цифрового фільтра. Результат фільтрації (відліки з виходу фільтру) подавались на ЦАП для зворотного перетворення у вихідний аналоговий сигнал. У якості цифрового осцилографа, спектроаналізатора та генератора аналогових сигналів використовувався USB мультіаналізатор Instrustar ISDS 220B.

Для реалізації синхронної роботи АЦП, цифрового фільтра та ЦАП використовувався апаратний таймер модуля ESP32. В обробнику передання від таймера зчитувалося поточне значення результату перетворення АЦП, виконувалась програма цифрової фільтрації та результат фільтрації завантажувалася у ЦАП. Поточне значення кількості нарахованих тактових імпульсів в апаратному таймері використовувалось для аналізу часу, який потрібен на отримання даних від АЦП, реалізацію операцій фільтрації та видачу результату фільтрації у ЦАП.

В результаті досліджень було виявлено, що в середовищі Arduino IDE при використанні стандартної функції доступу до АЦП `analogRead()` необхідно 60мкс на отримання нового результату вимірювань. При використанні функції `adc1_get_raw()` з бібліотеки фірми розробника модулів потрібно значно менше часу – 41мкс. Для завантаження результатів фільтрації у ЦАП з використанням функції `dac_output_voltage()` потрібно 1мкс. За результатами вимірювань у ESP32 на виконання усіх операцій алгоритму фільтрації із 13 коефіцієнтами фільтра потрібно 2мкс, у фільтрі з 32 коефіцієнтами – 3.1 мкс, у фільтрі з 64 коефіцієнтами – 5 мкс.

Висновки. Виходячи з періоду вимірювань 100мкс цифрова фільтрація на ESP32 реалізується достатньо швидко і, навіть, швидкодія роботи фільтрів з великою кількістю коефіцієнтів (64 коефіцієнта) достатньо висока – на фільтрацію використовується до 5% часу одного ядра. Однак середовище Arduino IDE не дозволяє швидко отримувати нові результати від АЦП та працювати з ЦАП. Значення часу доступу до АЦП та ЦАП досягають 60% та 1% відповідно. Використання функції доступу до АЦП від розробника ESP32 дозволяє зменшити час доступу до АЦП на 19% від загальної швидкодії ядра, але все одно це дуже поганий результат. Вирішенням цієї проблеми може бути лише написання програм у Visual Studio з додатковим фрейворком від компанії Espressif – розробника модулів.

Список використаних джерел:

1. Zubkov O., Svyd I., Vorgul O. Features of the Digital Filters Implementation on STM32 Microcontrollers III International Scientific and Practical Conference Theoretical and Applied Aspects of Device Development on Microcontrollers and FPGAs. 2021. P. 6-8.
2. Zena Ez Dallalbashi MatLab Based Design and Implemetation of Digita Filter International Journal of Computer Science and Network Security. 2020. Vol. 20. No. 1. P. 91–101.
3. Ratik Mittal, Vijay K. Jain Implementing Digital Filters and DSP Micro-Controller for Estimating the Frequency of a Time-Domain Signal. 2017. P. 1-3.
4. Muharrem Çelebi Digital Filter Design Based on ARDUINO and Its Applications, Medical Technologies Congress. 2020. P. 263-266.