

РЕАЛІЗАЦІЯ СИСТЕМИ ПЕРЕДАЧІ ДАНИХ ДЛЯ БАГАТОКАНАЛЬНОЇ РАДІОТЕХНІЧНОЇ СИСТЕМИ НА FPGA З ВИКОРИСТАННЯМ LoRa

Вінницький національний технічний університет

Анотація

Удосконалено багатоканальну радіотехнічну систему для частотних перетворювачів фізичних величин з використанням радіовимірювальних сенсорів на основі транзисторної структури з від'ємним диференційним опором, завдяки інтеграції у систему LoRa модуля для передачі вимірних значень безпроводним шляхом.

Ключові слова: FPGA; ESP32; LoRa; ПЛІС; багатоканальна радіовимірювальна система; частотний перетворювач; транзисторна структура з від'ємним опором

Abstract

The multi-channel radio measurement system for frequency converters of physical quantities using radio measurement sensors based on a transistor structure with negative differential resistance has been improved, thanks to the integration into the LoRa system of a module for transmitting measured values wirelessly.

Keywords: ESP32; LoRa; FPGA; multi-channel radiomeasuring system; frequency transducer; transistor structure with negative resistance

Вступ

Розробка сучасних інформаційно-вимірювальних систем вимагає застосування гнучких та водночас високопродуктивних технологій. У цьому контексті значуще місце посідають системи на кристалі FPGA [1], які, завдяки своїй гнучкості та можливості реалізації високошвидкісної обробки даних, стають відмінним вибором для багатьох сфер застосування. Однак, для обробки та передачі даних на вищому рівні абстракції зазвичай потрібна інтеграція з більш високорівневими мікропроцесорними системами. Це спонукало до розробки архітектур, які комбінують можливості FPGA та мікроконтролерів, таких як ESP32 [2], для створення інтегрованих рішень. Ці тези фокусуються на системі, яка включає FPGA, що відповідає за первинну обробку даних і передає їх через UART до мікроконтролера ESP32. Після чого, ESP32 за допомогою модулю LoRa [3] передає дані до іншої ESP32, яка в свою чергу передає ці дані на персональний комп'ютер через UART для подальшої обробки та аналізу. У цій роботі буде описано покрокову реалізацію кожного із компонентів крім системи на FPGA.

Результати розробки та дослідження

Загальна структура розробленої системи зображена на рисунку 1. Дані, які генеруються FPGA, подальше передаються на модуль ESP32 через комунікаційний інтерфейс Universal Asynchronous Receiver/Transmitter (UART) із застосуванням швидкості передачі даних 115200 біт/с. Кожний пакет даних містить представлений у форматі JSON і має наступну структуру: масив із всіма датчиками, кожне представлення датчика містить два, поля ідентифікатор датчика і його значення.

```
sensors: [  
  {  
    id: <string with sensor ID>,  
    value: <single or multiple values of sensor, according to sensor type>  
  }, ...  
]
```

В архітектурі розробленої системи ESP32 відіграє важливу роль як міст між FPGA та LoRa модулем. Дані, які FPGA генерує [4], передаються на ESP32 через UART інтерфейс, і подальше на LoRa модуль для бездротової передачі до іншої ESP32 одиниці. Зв'язок між FPGA та ESP32, а також між ESP32 та LoRa модулем є одностороннім. Дані передаються без шифрування, що спрощує процес передачі, але водночас знижує рівень безпеки передачі даних.

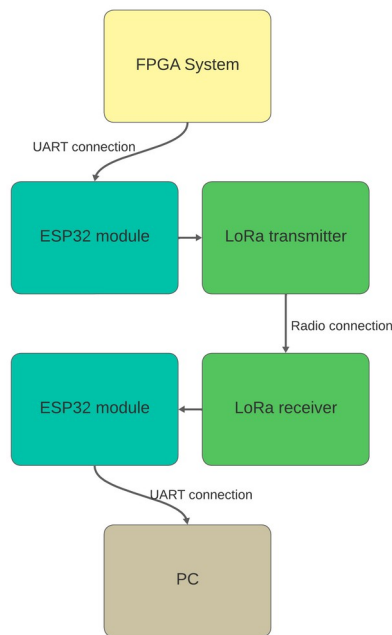


Рисунок 1 – Загальна структурна схема системи передачі даних від FPGA безпроводним шляхом

Код нижче ілюструє, як ESP32 може приймати дані з UART, обробляти пакети JSON та перенаправляти ці дані на LoRa модуль:

```
#include <ArduinoJson.h>
#include <HardwareSerial.h>
#include <LoRa.h>

// Піни для LoRa модуля
#define SCK 5 // GPIO5 -- SX1278's SCK
#define MISO 19 // GPIO19 -- SX1278's MISO
#define MOSI 27 // GPIO27 -- SX1278's MOSI
#define SS 18 // GPIO18 -- SX1278's CS
#define RST 14 // GPIO14 -- SX1278's RESET
#define DI0 26 // GPIO26 -- SX1278's IRQ(Interrupt Request)

HardwareSerial Serial2(2); // Оголошення апаратного серійного порту UART2

void setup() {
    Serial2.begin(115200, SERIAL_8N1, 16, 17); // Ініціалізація UART2 із швидкістю передачі 115200, 8 бітами
    даних, без біту парності та 1 бітом стопу
    Serial.begin(115200);
    // Налаштування пінів для LoRa модуля
    LoRa.setPins(SS, RST, DI0);
    if (!LoRa.begin(433E6)) { // Ініціалізація LoRa модуля на частоті 433 MHz
        Serial.println("LoRa init failed. Check your connections.");
    }
    // Налаштування параметрів LoRa
    LoRa.setSpreadingFactor(7);
    LoRa.setSignalBandwidth(125E3);
    LoRa.setCodingRate4(5);
}

void loop() {
    if (Serial2.available()) { // Перевірка на наявність даних на UART2
        String packet = Serial2.readStringUntil('\n'); // Читання даних з UART2 до символу нового рядка

        LoRa.beginPacket(); // Початок пакету LoRa
        LoRa.print(packet); // Надсилання даних пакету JSON через LoRa
        LoRa.endPacket(); // Завершення пакету LoRa
    }
}
```

У даному кодї використовується бібліотека ArduinoJson для обробки пакетів JSON. Бібліотека LoRa використовується для взаємодії з LoRa модулем та передачі даних. При наявності даних на UART, дані читаються, а потім перенаправляються на LoRa модуль для бездротової передачі. Цей код слугує основою для реалізації передачі даних від FPGA до ESP32 та подальшого їх перенаправлення на LoRa модуль.

На приймаючій стороні, ESP32 з LoRa модулем слухає радіо-частотний канал 433E6 для отримання вхідних пакетів. При отриманні пакету, ESP32 розбирає дані та витягує корисну інформацію.

```
#include <ArduinoJson.h>
#include <LoRa.h>
#include <HardwareSerial.h>

HardwareSerial PCSerial(1); // Апаратний серійний порт для комунікації з ПК

void setup() {
  LoRa.setPins(SS, RST, DIO); // Налаштування пінів для LoRa модулю
  if (!LoRa.begin(433E6)) { // Ініціалізація LoRa модулю на частоті 433 MHz
    while (true); // Цикл безкінечно, якщо ініціалізація не вдалася
  }
  // Налаштування параметрів LoRa
  LoRa.setSpreadingFactor(7);
  LoRa.setSignalBandwidth(125E3);
  LoRa.setCodingRate4(5);
  PCSerial.begin(115200, SERIAL_8N1, 16, 17); // Ініціалізація UART для комунікації з ПК
}

void loop() {
  int packetSize = LoRa.parsePacket(); // Початок парсингу пакету LoRa
  if (packetSize) { // Якщо пакет отримано
    String receivedPacket = "";
    while (LoRa.available()) {
      receivedPacket += (char)LoRa.read(); // Читання даних пакету
    }
    DynamicJsonBuffer jsonBuffer;
    JsonObject& root = jsonBuffer.parseObject(receivedPacket); // Розбір JSON пакету
    if (root.success()) { // Якщо розбір JSON вдався
      String id = root["id"];
      String data = root["value"]; // Витягування даних
      sendToPC(id, data); // Надсилання даних на ПК
    }
  }
}

void sendToPC(String id, String data) {
  PCSerial.println(id + "," + data); // Надсилання даних на ПК через серійний порт
}
```

Для передачі даних на ПК, ESP32 використовує UART інтерфейс. Дані можуть бути передані через серійний порт на ПК, де вони можуть бути зчитані та оброблені за допомогою спеціалізованого програмного забезпечення. У цьому прикладі, функція *sendToPC* використовується для надсилання даних на ПК через серійний порт. На ПК можна використовувати будь-яку програму для читання серійного порту (наприклад, Arduino IDE, PuTTY або будь-яке інше програмне забезпечення для роботи з серійним портом) для отримання та обробки даних.

Такий підхід дозволяє відділити процеси прийому та передачі даних, що полегшує відлагодження та розробку системи.

Висновки

Реалізовано систему передачі даних з FPGA на ПК з використанням безпроводної лінії зв'язку, через LoRa. Забезпечено односторонній зв'язок для передачі даних від FPGA до ESP32 з швидкістю 115200 біт/сек. Використано формат JSON для пакетування даних, який включає масив датчиків з їх назвами та поточними значеннями. Описано процес ініціалізації та налаштування параметрів LoRa модулю для передачі даних. Надано приклад коду для передачі даних з ESP32 на LoRa модуль. Описано процес прийому даних з LoRa, включаючи парсинг пакетів та витягування корисних даних. Описано

процес передачі даних з ESP32 на ПК через UART інтерфейс. Надано приклад коду для відправлення даних на ПК через серійний порт з використанням функції *sendToPC*.

Сформовано загальний опис системи передачі даних, яка починається з FPGA, проходить через ESP32 та LoRa модуль, і закінчується на ПК. Кожен компонент і кожен крок передачі даних детально описано, з врахуванням взаємодії між різними частинами системи.

СПИСОК ВИКОРИСТАНОЇ ЛІТЕРАТУРИ

1. What is an FPGA? [Електронний ресурс]: - Режим доступу: <https://www.xilinx.com/products/silicon-devices/fpga/what-is-an-fpga.html>
2. ESP32. [Електронний ресурс]: - Режим доступу <https://www.espressif.com/en/products/socs/esp32>
3. LoRa Alliance. [Електронний ресурс]: - Режим доступу: <https://lora-alliance.org/>
4. Осадчук О.В., Осадчук Я.О., Скощук В.К. Удосконалення багатоканальної радіотехнічної системи на FPGA для частотних перетворювачів фізичних величин підтримкою цифрових сенсорів // Вимірювальна та обчислювальна техніка в технологічних процесах. 2023, №2. –С.72-82. <https://vottp.khmnu.edu.ua/index.php/vottp/article/view/117>

Осадчук Олександр Володимирович — докт. техн. наук, проф., зав. кафедри інформаційних радіоелектронних технологій і систем, Вінницький національний технічний університет, osadchuk.av69@gmail.com

Скощук Валентин Костянтинович — аспірант кафедри інформаційних радіоелектронних технологій і систем, Вінницький національний технічний університет

Alexander Osadchuk — Doc. Tech. Sc., prof. Head of Department of Information Radio Engineering Technologies and Systems, Vinnytsia National Technical University, Vinnytsia, Ukraine, osadchuk.av69@gmail.com

Valentyn Skoshchuk – graduate student of the Department of Information Radio Engineering Technologies and Systems, Vinnytsia National Technical University, Vinnytsia, Ukraine