

<https://doi.org/10.31891/2219-9365-2023-73-1-19>

УДК 621.317

Олександр ОСАДЧУК

Вінницький національний технічний університет

<https://orcid.org/0000-0001-6662-9141>

e-mail: [osadchuk.av69@gmail.com](mailto:osadchuk.av69@gmail.com)

Ярослав ОСАДЧУК

Вінницький національний технічний університет

<https://orcid.org/0000-0002-5472-0797>

e-mail: [osadchuk.av69@gmail.com](mailto:osadchuk.av69@gmail.com)

Валентин СКОЩУК

Вінницький національний технічний університет

e-mail: [skoschuk999@gmail.com](mailto:skoschuk999@gmail.com)

## ВИКОРИСТАННЯ ЯДРА NIOS II У БАГАТОКАНАЛЬНОМУ ЧАСТОТОМІРІ НА FPGA ДЛЯ РАДІОТЕХНІЧНОЇ СИСТЕМИ З ЧАСТОТНИМИ СЕНСОРАМИ ФІЗИЧНИХ ВЕЛИЧИН

У роботі представлено один із шляхів розширення функціональних можливостей багатоканальної системи паралельного вимірювання частоти на основі FPGA фірми Altera Cyclone IV, основною задачею якої є вимірювання інформативного параметру сенсорів фізичних величин з частотним виходом, за допомогою мікропроцесорного ядра NIOS II. Удосконалено багатоканальний універсальний вимірювальний прилад на основі FPGA, який має 12 вимірювальних каналів для сенсорів з частотним виходом. Частково перероблено блоки: лічильника імпульсів, обробки даних, UART передавача, згенеровано і налаштовано Nios II, повторно синтезовано схему. Інтеграція NIOS II у багатоканальний частотомір дозволить зробити систему більш гнучкою, додати попередню обробку, фільтрацію отриманих даних і змінювати кількість частотомірів, без зміни алгоритму обробки даних, що являється неможливим у попередній реалізації системи.

Ключові слова: NIOS II, FPGA, сенсор з частотним виходом, багатоканальний частотомір, частота, радіовимірювальні перетворювачі фізичних величин.

Oleksandr OSADCHUK, Iaroslav OSADCHUK, Valentyn SKOSHCHUK

Vinnitsia National Technical University

## USING OF THE NIOS II CORE IN A MULTI-CHANNEL FREQUENCY COUNTER ON FPGA FOR A RADIO ENGINEERING SYSTEM WITH FREQUENCY SENSORS OF PHYSICAL QUANTITIES

The paper presents one of the ways to expand the functionality of the multi-channel parallel frequency measurement system based on the Altera Cyclone IV FPGA, the main task of which is to measure the informative parameter of sensors of physical quantities with frequency output, using the NIOS II microprocessor core. An improved multi-channel universal measuring device based on FPGA, which has 12 measuring channels for sensors with frequency output. The blocks of the pulse counter, data processing, UART transmitter were partially reworked, Nios II was generated and configured, the circuit was re-synthesized. Integration of NIOS II into a multi-channel frequency meter will make the system more flexible, add pre-processing, filtering of received data and change the number of frequency meters, without changing the data processing algorithm, which is impossible in the previous implementation of the system. When introducing a frequency meter, a microprocessor system into the scheme, part of the tasks from the existing blocks are transferred to a more flexible component - the microprocessor. Such blocks as: data processing from counters, UART transmitter, completely lose their relevance, they will be implemented by means of the processor. The pulse counter unit will be partially redesigned for ease of use with a microprocessor system. Software has also been developed that allows to read data from frequency meters, process, convert and transfer them to the UART serial port. Since after the integration of the NIOS II core reading frequency and transfer to the UART port is done programmatically, it is possible to implement the algorithm that will convert the read frequency value into the physical value of the sensor.

Keywords: NIOS II, FPGA, sensor with frequency output, multichannel frequency meter, frequency, radio measuring transducers of physical quantities.

### Постановка проблеми у загальному вигляді

#### та її зв'язок із важливими науковими чи практичними завданнями

Поява FPGA дала змогу всі елементи фізично розмістити на одній інтегральній схемі, та перейти на якісно новий рівень, який пов'язаний із значним збільшенням їх ступеня інтеграції і підвищенням швидкодії [1-3]. Перевагою таких системи є можливість швидко змінювати внутрішню структуру і зв'язки між компонентами без втручання у апаратну частину, додавати нові функціональні блоки. Але є і недоліки, оскільки програмування здійснюється на досить низькому рівні, а саме рівні логічних елементів, то для побудови високорівневих алгоритмів обробки даних, фільтрів і аналізаторів, доводиться витратити велику кількість часу, для того щоб відобразити алгоритм у апаратній частині. Для такого роду задач необхідно мати інструменти відповідного рівня абстракції, такі як мікропроцесорні ядра, при роботі з якими не потрібно задумуватися про реалізацію: регістрів, арифметичних блоків, блоків пам'яті, шин передачі даних

і способу синхронізації вище згаданих блоків. Вирішенням цієї проблеми на FPGA була поява мікропроцесорного ядра NIOS II [2-6].

### Аналіз досліджень та публікацій

Сьогодні завдяки Інтернету, вбудованим веб-серверам та інформаційно-вимірювальним системам можна виконувати технологічні операції віддаленого моніторингу з дуже низькою ціною [3, 4]. Фактично, вбудовані інформаційно-вимірювальні системи мають дедалі більшу присутність у широкому діапазоні областей, пов'язаних із комерційною електронікою та промисловими застосуваннями [3-5]. Ці системи характеризуються пристроєм, призначеним для моніторингу мікросистемних мереж у режимі реального часу або для автоматичного виконання будь-якого завдання без втручання людини [4]. Зазвичай більшість із цих пристроїв реалізовано за допомогою ПК або мікроконтролерів, однак FPGA є життєздатною альтернативою для реалізації цих систем, оскільки вони додають нові функції до традиційних архітектур на основі мікропроцесорів або мікроконтролерів. Наприклад, технологія FPGA робить інформаційно-вимірювальну систему невеликих розмірів (портативними), гнучкими, реконфігурованими і перепрограмованими з такими перевагами, як хороше налаштування, економічна ефективність, інтеграція, доступність і розширюваність [5-7].

Ми можемо проектувати апаратне забезпечення, програмне забезпечення та ядро одночасно, що значно скорочує цикл проектування [8]. Технологія FPGA також забезпечує надзвичайно високу продуктивність обробки сигналів. Усі ці функції дозволяють нам реалізувати в одному пристрої вбудовану інформаційно-вимірювальну систему, яка виконується за допомогою програмного або жорсткого мікроконтролера всередині мікросхеми FPGA [8]. Цей мікроконтролер може взаємодіяти з IP-ядрами або модулями VHDL, які виконують певні апаратні засоби обробки та інші завдання.

У роботі [8] авторами запропоновано один з підходів до реалізації багатоканальної системи вимірювання частоти на FPGA. В якості мікросхеми для реалізації частотомірів використовується FPGA фірми Altera Cyclone IV EP4CE10F17C8. Для створення проекту під FPGA вибрано версію Quartus 15.1 [9]. Також розглядається створення основних блоків частотоміра, пояснюється їх призначення і принцип роботи. Після реалізації усіх необхідних блоків відбувається синтез схеми рис. 1.

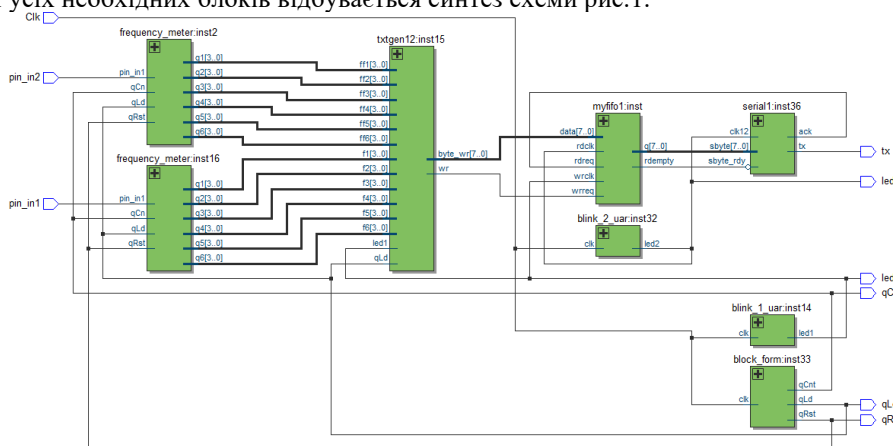


Рис. 1. Схема багатоканального частотоміра

Створюються наступні блоки:

1. Формувач сигналів керування – на виході послідовно створює три сигнали керування: «qRst», «qCnt», «qLd».
2. Лічильники імпульсів – використовують сигнали керування для запуску і вивантаження значень.
3. Обробник даних із лічильників – конвертує значення лічильників у ASCII символи і відправляє їх у чергу.
4. UART передавач – передає ASCII символи через послідовний порт до під'єданого пристрою.

На рис. 1 зображено поєднання блоків які використовуються, для отримання і передачі даних багатоканального частотоміра для вимірювання інформативних параметрів отриманих з радіовимірювальних перетворювачів, на основі реактивних властивостей напівпровідникових структур з від'ємним опором, які мають частотні виходи з повним збереженням у вихідному сигналі інформації про кількісне значення вимірюваної величини. На рис. 2 зображено результат роботи розробленої інформаційно-вимірювальної системи при роботі із радіовимірювальними сенсорами тиску.

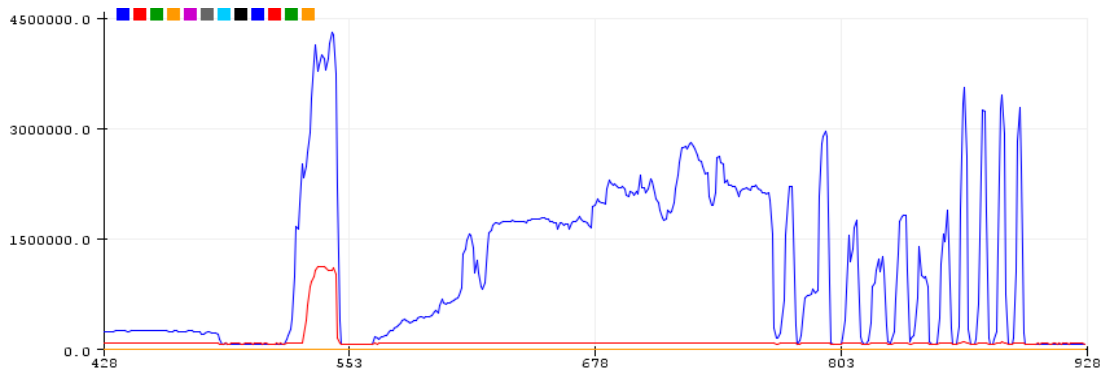


Рис. 2. Вимірювання значень радіовимірювальних сенсорів тиску з частотним виходом

### Формулювання цілей статті

Метою роботи є: інтеграція ядра NIOS II у багатоканальний частотомір на FPGA для радіотехнічної системи з частотними сенсорами фізичних величин [8], що дозволить зробити систему гнучкою, додати попередню обробку і фільтрацію отриманих даних, із частотомірів. Також це дозволить змінювати кількість частотомірів, без зміни алгоритму обробки даних, що являється неможливим у попередній реалізації інформаційно-вимірювальної системи [8].

### Теоретичні та експериментальні дослідження

Система з процесором Nios II - це еквівалент мікроконтролеру, який містить процесор, комбінацію додаткових підсистем і пам'яті в одному чіпі. Така система складається з ядра процесора Nios II і набору допоміжних підсистем на чіпі, (див. рис. 3).

Аналогічно сімейству мікроконтролерів, система з процесором Nios II використовує постійний набір інструкцій і модель програмування. Програмне ядро дозволяє не обмежуватися певною апаратною частиною, а розміщуватися в будь-яких чіпах сімейств Altera FPGA. Гнучкий набір допоміжних систем - одна з найголовніших відмінностей між процесорною системою Nios II і фіксованими мікроконтролерами. Оскільки процесор Nios II реалізується програмованою логікою, є можливість створити нестандартну процесорну систему з точним набором допоміжних підсистем, необхідних для вирішення задачі [2].

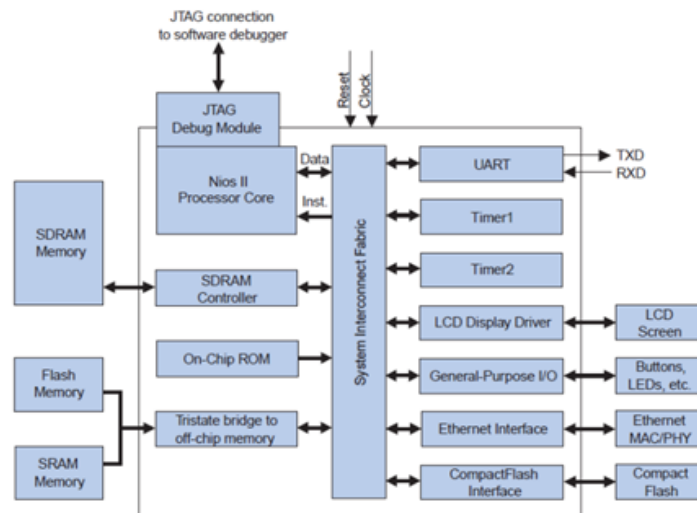


Рис. 3. Приклад мікропроцесорної системи Nios II

Функціональні модулі архітектури Nios II формують основний набір інструкцій [2]. Однак, вони не відображаються при апаратній реалізації. Архітектура описується набором інструкцій, без будь-якої апаратної реалізації. Функціональні модулі можуть бути апаратно реалізовані, імітовані програмно або повністю пропущені. Реалізація - це вибір з набору проєктів об'єднаних в специфічному ядрі Nios II. Кожна реалізація отримує на виході специфічні об'єкти, наприклад, ядра маленькі за розміром або високопродуктивні ядра. Це дозволяє адаптувати архітектуру Nios II відповідно до потреб поставлених задач. В кінцевій реалізації вибирається один з трьох компромісних варіантів: більша чи менша продуктивність, включення або виключення допоміжних функцій, апаратна реалізація або програмна емуляція [2].

Для інтеграція мікропроцесорної системи в існуючу схему частотоміра рис. 1 потрібно частково переробити блоки: лічильника імпульсів, обробки даних, UART передавача і повторно синтезувати схему. Першим етапом являється генерування і налаштування мікропроцесорної системи Nios II.

Генерація і налаштування мікропроцесорної системи відбувається за допомогою утиліти Qsys [2]. Першим елементом який потрібно налаштувати являється «Clock Source», він визначає робочу частоту системи, рис. 4. На платі разом з FPGA знаходиться кварцовий резонатор на 50 МГц, він буде використаний у якості вхідного генератора для розроблюваної мікропроцесорної системи [9, 10]. В елементі «Clock Source» виставляємо у якості робочої частоти 50 МГц, рис. 5.

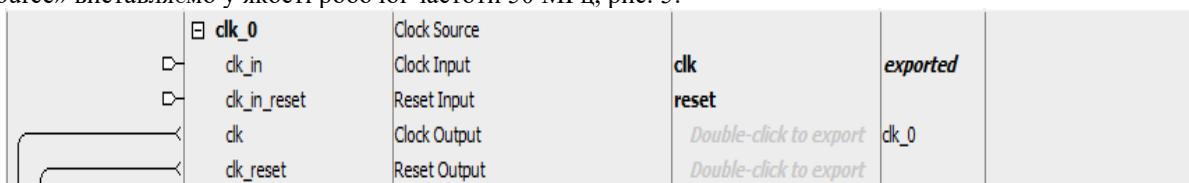


Рис. 4. Графічне відображення елемента «Clock Source»

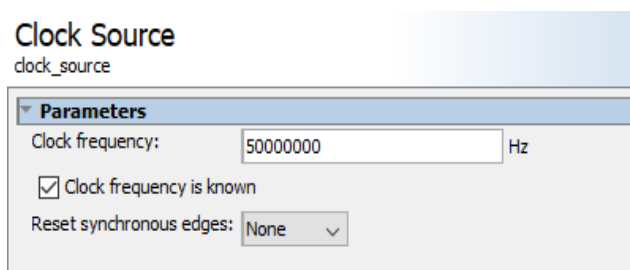


Рис. 5. Параметри налаштування елемента «Clock Source»

«On-Chip Memory» - використовується у якості блоку пам'яті RAM для мікропроцесорної системи, рис. 6. Розмір слова відповідає 32 бітам. Загальний розмір RAM пам'яті дорівнює 20 кБ, рис. 7.

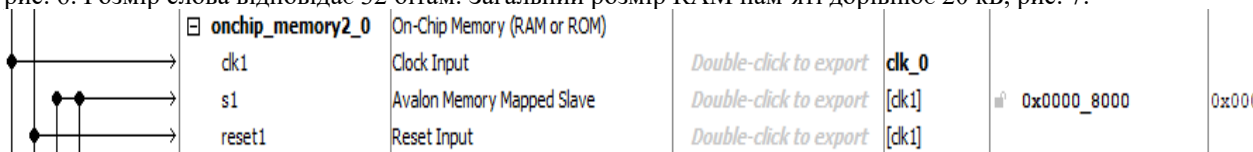


Рис. 6. Графічне відображення елемента «On-Chip Memory»

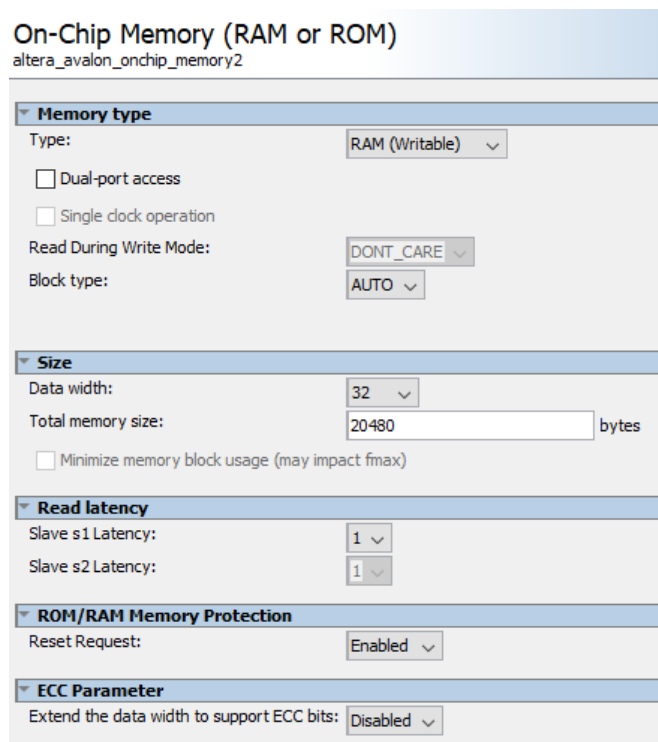


Рис. 7. Параметри налаштування елемента «On-Chip Memory»

«UART» - використовується для взаємодії із зовнішніми пристроями через послідовний порт, рис. 8. Основні налаштування: швидкість передачі 115200 Бод, автоматичний контроль цілісності пакета вимкнений, розмір даних в одному пакеті становить 8 біт, рис. 9.



Рис. 8. Графічне відображення інтерфейсу «UART»

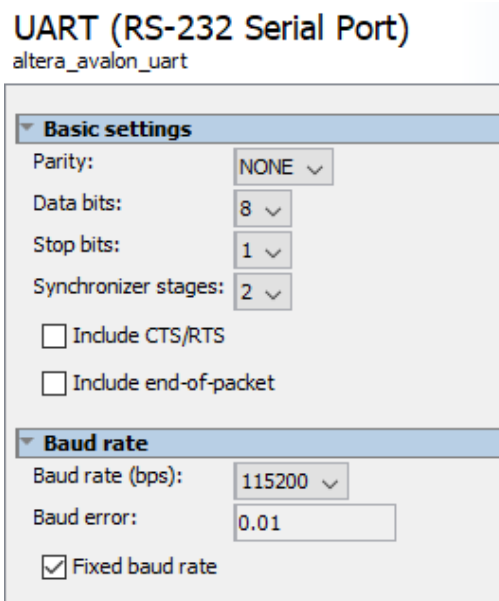


Рис. 9. Параметри налаштування інтерфейсу «UART»

«Counter» - використовується для підрахунку часу роботи системи і синхронізації операцій пов'язаних з вимірюваннями, рис. 10. Реалізація даного блоку відсутня у стандартній бібліотеці [9, 10], тому, він був реалізований самостійно, у якості інтерфейсу використовується вихідна 32-х розрядна шина, рис. 12. «Counter» розроблений у вигляді звичайного регістра, який змінює своє значення з кожним вхідним імпульсом від генератора. Для вимірювання часу, частота від генератор пропускається через подільник який визначає інтервал часу якому буде відповідати одне значення регістра, рис. 11. Для вимірювання 1мс необхідно виставити подільник у значення 1000. Оскільки частота основного резонатора 50 МГц, то одне значення регістра відповідає 50000 тактів генератора.

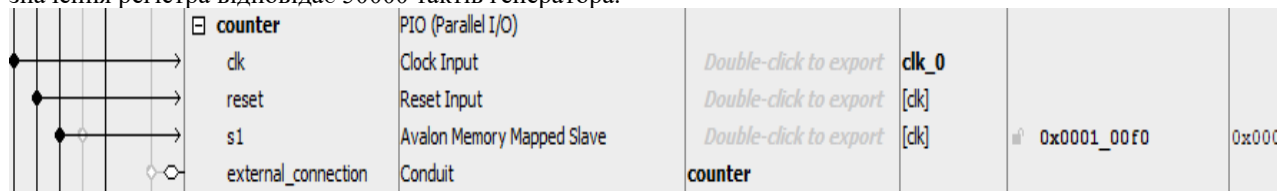


Рис. 10. Графічне відображення інтерфейсу для блока «Counter»

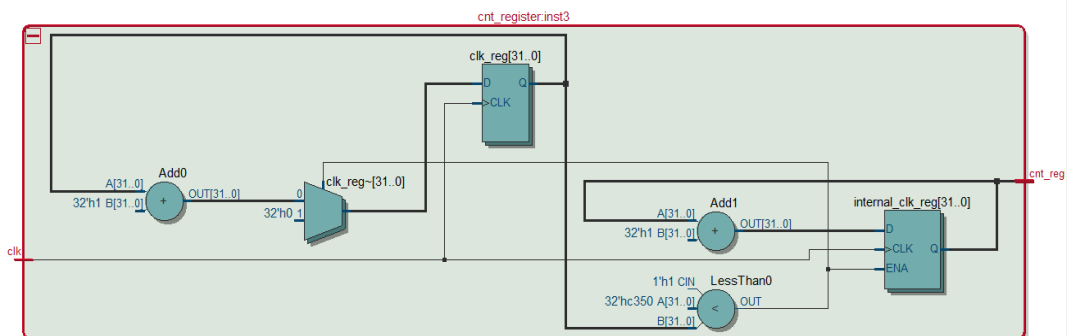


Рис. 11. Схематичне зображення лічильника часу

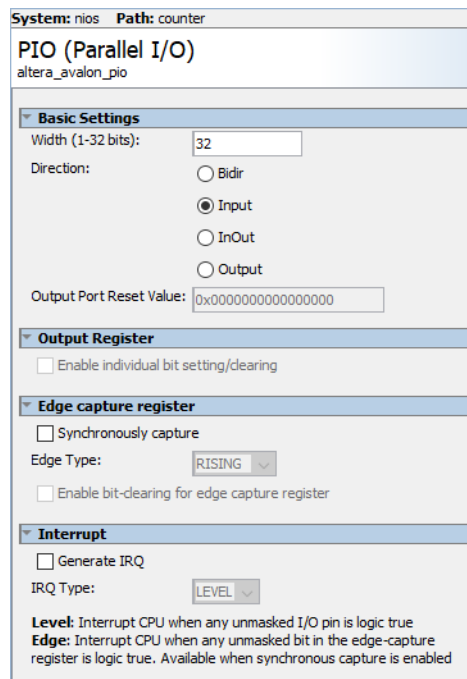


Рис. 12. Параметри налаштування інтерфейсу для блока «Counter»

«Sys IRQ» – інтерфейс для під'єднання зовнішніх джерел переривання, рис. 13. Підтримується до 8 зовнішніх джерел переривання, рис. 14.

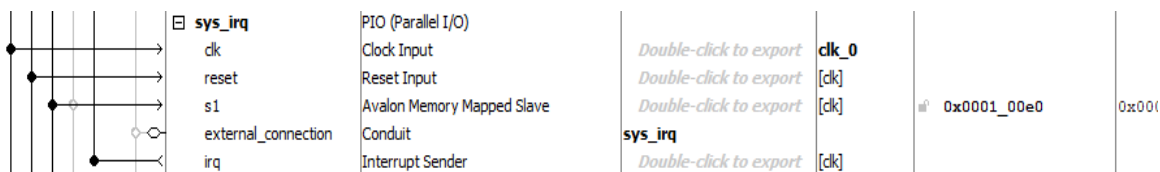


Рис. 13. Графічне відображення інтерфейсу «Sys IRQ»

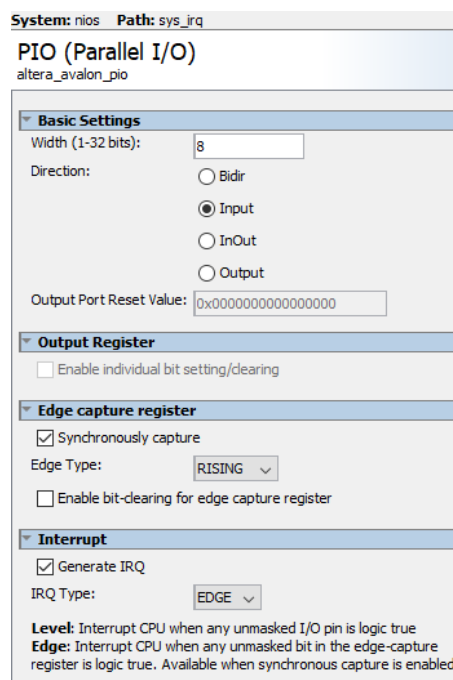


Рис. 14. Параметри налаштування інтерфейсу «Sys IRQ»

«FM» – інтерфейс для під'єднання блоку який реалізує вимірювання частоти, загалом підтримується до 12 частотомірів, рис. 15. Один блок під'єднується за допомогою 32-х розрядної вхідної шини, рис. 16.



fm_0	PIO (Parallel I/O)	clk_0	0x0001_00d0	0x0001
fm_1	PIO (Parallel I/O)	clk_0	0x0001_00c0	0x0001
fm_2	PIO (Parallel I/O)	clk_0	0x0001_00b0	0x0001
fm_3	PIO (Parallel I/O)	clk_0	0x0001_00a0	0x0001
fm_4	PIO (Parallel I/O)	clk_0	0x0001_0020	0x0001
fm_5	PIO (Parallel I/O)	clk_0	0x0001_0050	0x0001
fm_6	PIO (Parallel I/O)	clk_0	0x0001_0060	0x0001
fm_7	PIO (Parallel I/O)	clk_0	0x0001_0040	0x0001
fm_8	PIO (Parallel I/O)	clk_0	0x0001_0030	0x0001
fm_9	PIO (Parallel I/O)	clk_0	0x0001_0070	0x0001
fm_10	PIO (Parallel I/O)	clk_0	0x0001_0080	0x0001
fm_11	PIO (Parallel I/O)	clk_0	0x0001_0090	0x0001

Рис. 15. Графічне відображення інтерфейсу для блоків «FM»

System: nios Path: fm\_0

**PIO (Parallel I/O)**  
altera\_avalon\_pio

**Basic Settings**

Width (1-32 bits): 32

Direction:  Bidir  Input  InOut  Output

Output Port Reset Value: 0x0000000000000000

**Output Register**

Enable individual bit setting/clearing

**Edge capture register**

Synchronously capture

Edge Type: RISING

Enable bit-clearing for edge capture register

**Interrupt**

Generate IRQ

IRQ Type: LEVEL

**Level:** Interrupt CPU when any unmasked I/O pin is logic true  
**Edge:** Interrupt CPU when any unmasked bit in the edge-capture register is logic true. Available when synchronous capture is enabled

Рис. 16. Параметри налаштування інтерфейсу для блоків «FM»

«Nios2» – найважливіший компонент, ядро мікропроцесорної системи, рис. 17. Основними налаштуванням являється: мало-продуктивна версія ядра, для збільшення кількості вільних логічних елементів на FPGA, рис. 18; відключення компонентів для налагодження, рис. 19. Усі інші налаштування генеруються автоматично.

nios2_qsys_0	Nios II (Classic) Processor	clk_0	Double-click to export [clk]	
clk	Clock Input		Double-click to export [clk]	
reset_n	Reset Input		Double-click to export [clk]	
data_master	Avalon Memory Mapped Master		Double-click to export [clk]	
instruction_master	Avalon Memory Mapped Master		Double-click to export [clk]	
d_irq	Interrupt Receiver		Double-click to export [clk]	IRQ 0
custom_instruction_m...	Custom Instruction Master		Double-click to export [clk]	

Рис. 17. Графічне відображення інтерфейсу ядра «Nios2»

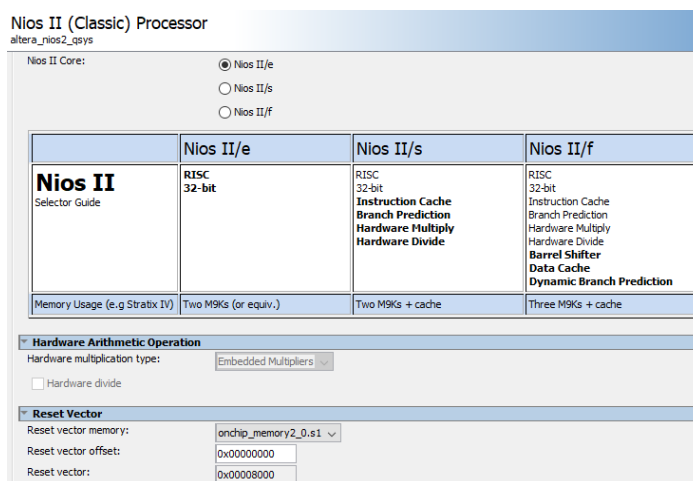


Рис. 18. Вибір продуктивності ядра мікропроцесорної системи

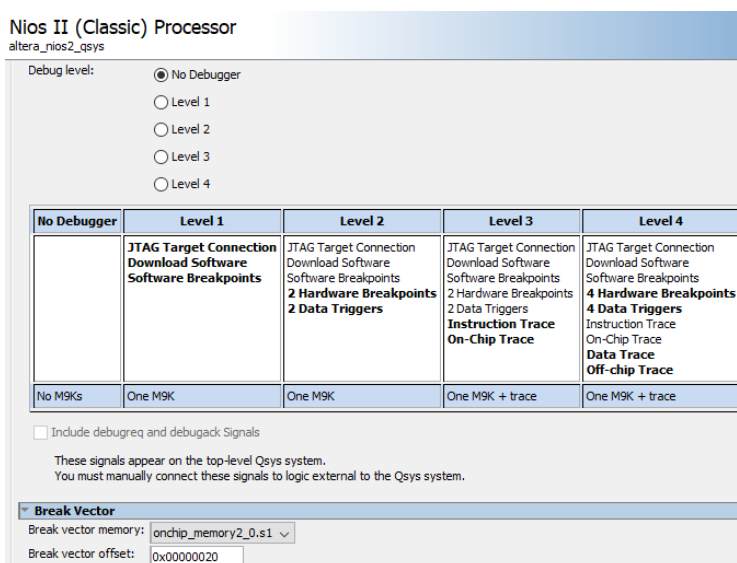


Рис. 19. Налаштування налагоджувального компонента

На рис. 20 зображений блок який містить всі компоненти згенерованої мікропроцесорної системи.

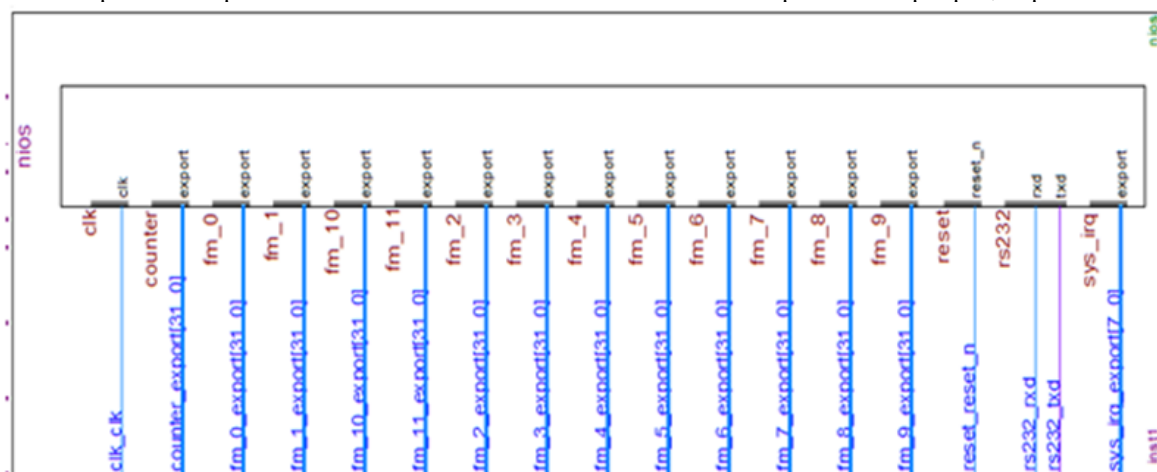


Рис. 20. Блок згенерований утилітою Qsys

При внесенні у схему частотоміра, частина задач з існуючих блоків переноситься на більш гнучкий компонент – мікропроцесор. Такі блоки як: обробка даних з лічильників, UART передавач, повністю втрачають свою актуальність, вони будуть реалізовані засобами процесора. Блок лічильника імпульсів буде частково перероблений, для зручності використання з мікропроцесорною системою.



Оновлений лічильник імпульсів представляє собою простий 32-х розрядний регістр який збільшує своє значення на одиницю при кожному вхідному імпульсі, у якості вихідного інтерфейсу використовується 32-х розрядна шина даних, рис. 21. Для опису та генерування лічильника імпульсів, код написаний на мові Verilog. Повна структурна схема з'єднань мікропроцесорної системи наведена на рис.22.

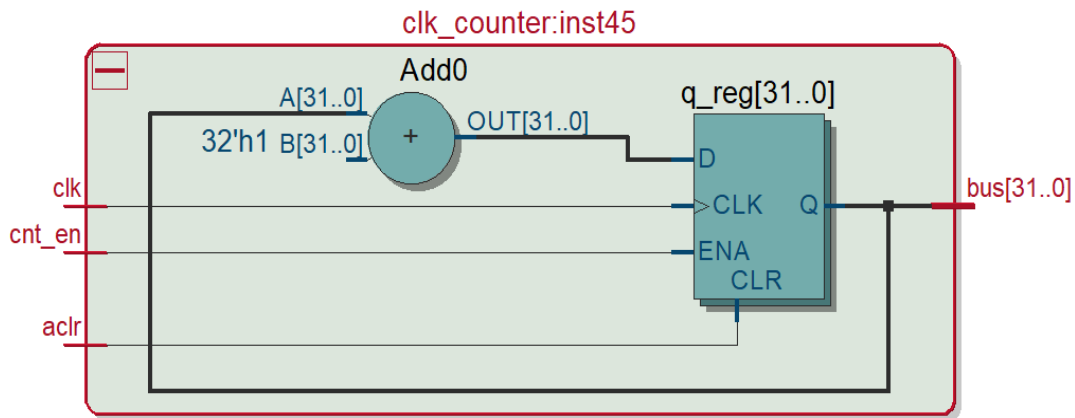


Рис. 21. Лічильник імпульсів

Блок керування і 12 лічильників були поміщені у окремий компонент для зручності використання, рис. 22. Інтерфейс отриманого компонента містить: вхід для зовнішнього тактового генератора, 12 вимірювальних входів, 12-ть вихідних шин для передачі результату роботи, інформаційні сигнали для визначення внутрішнього стану компонента, рис. 23.

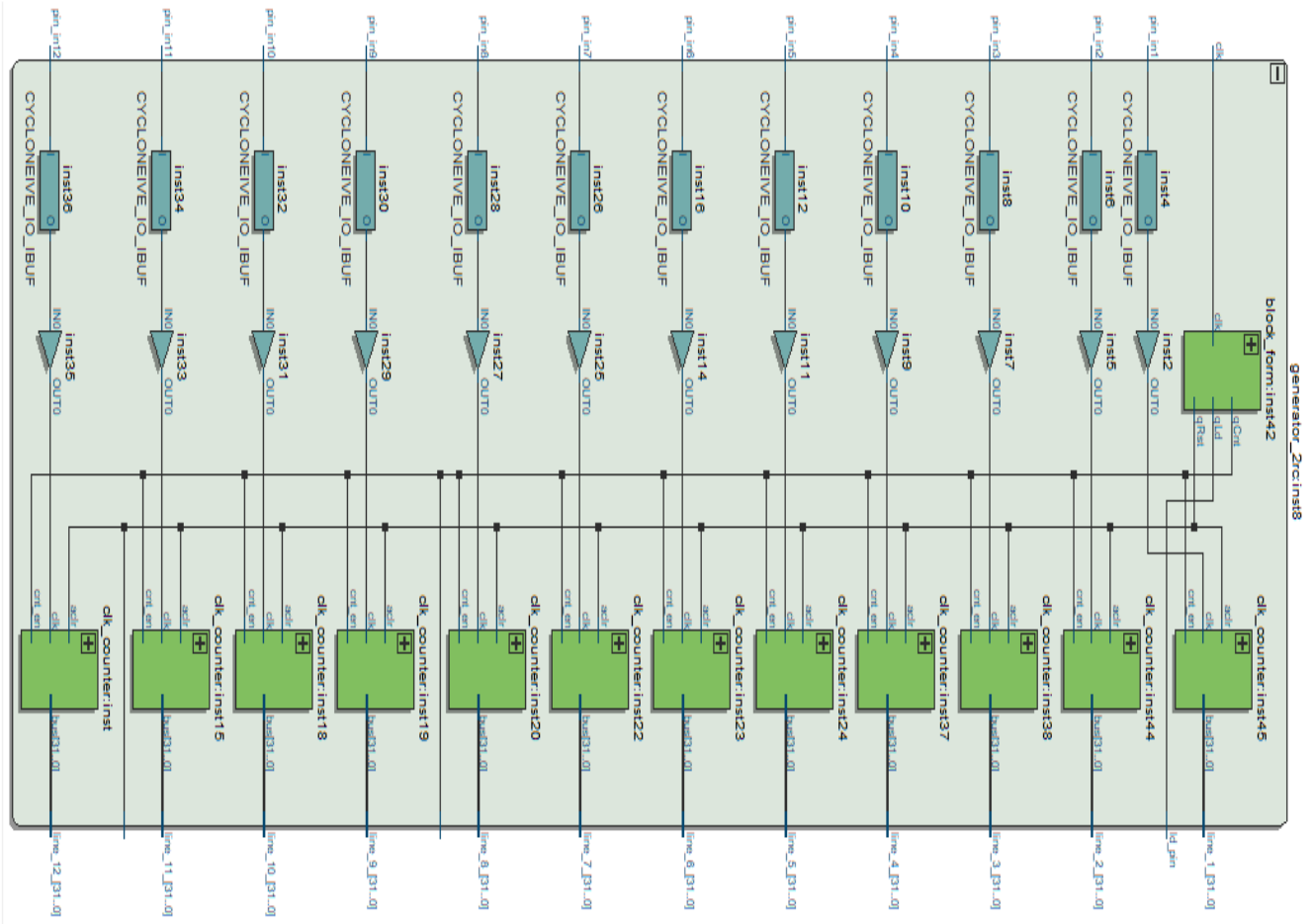


Рис. 22. Внутрішня структура компонента із частотомірами

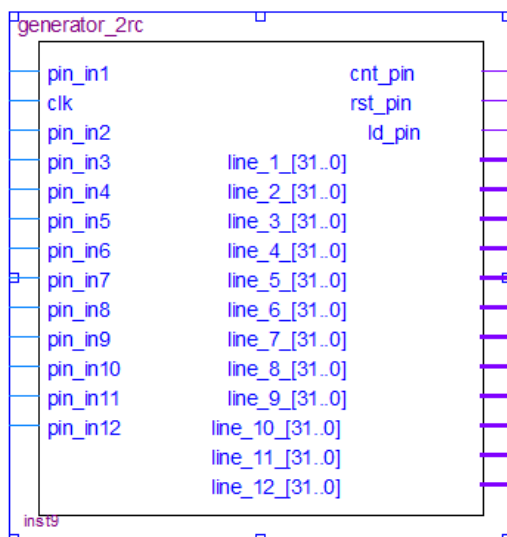


Рис. 23. Інтерфейс компонента із частотомірами

В кінцевому результаті було синтезовано схему рис. 24.

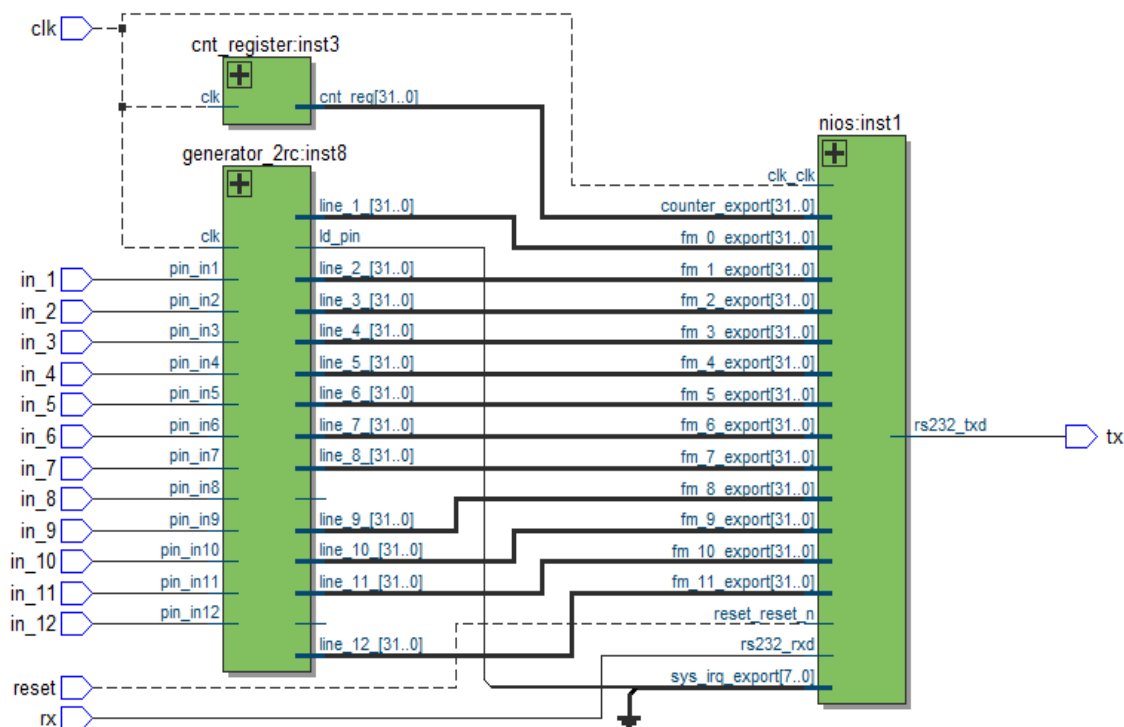


Рис. 23. Схема багатоканального частотоміра з використанням ядра NIOS II

Для отримання значень з частотомірів використовується розроблений інтерфейс. Коли частотоміри закінчують підрахунок, генерується переривання яке викликає у якості обробника функцію «fm\_irq\_handler», рис. 24.

```
void fm_irq_handler(void* context) {
    sys_irq_clear_interrupt_register(IRQ_MASK);
    alt_u8 i = 0x00;
    for (i = 0x00; i < CHANNEL_MAX; i++) {
        channels_data[i] = IORD_ALTERA_AVALON_PIO_DATA(channel_id_to_addr(i));
    }
}
```

Рис. 24. Обробник переривань для частотомірів

Функція «fm\_irq\_handler» зчитує значення частотомірів, які зберігаються в регістрах, у масив на 12 елементів, звідки можуть бути доступні для користувача у будь-який час, рис. 25. Кожний елемент масиву представляє значення з окремого частотоміра.

```
alt_32 fm_get_freq(alt_u8 channel_id) {  
    if (channel_id >= CHANNEL_MAX)  
        return RES_ERR;  
  
    return channels_data[channel_id];  
}
```

Рис. 25. Функція отримання значення з певного частотоміра

Для перевірки драйвера частотомірів до системи було під'єднано генератор частоти на 85 кГц до першого каналу. Основна функція спершу ініціалізує частотоміри, після чого послідовно викликає зчитування значень з частотомірів, реалізація функції зображена на рис. 26. Після зчитування отриманих значень відбувається формування повідомлення, у послідовності їх зчитування, кожне значення розділене символом «\b». Сформоване повідомлення відправляється на UART, до якого під'єднаний конвертор PL2303, який у свою чергу під'єднується до USB порту ПК. Перші 12 значень відповідають значенням частотомірів, рис. 27.

```
int main() {  
    sys_printf("Frequency meter initializing\r\n");  
    if (fm_init())  
        sys_printf("Error during frequency meter initializing\r\n");  
  
    /* Event loop never exits. */  
    while (1) {  
        alt_u8 i = 0x00;  
  
        for (; i < CHANNEL_MAX; i++)  
            data_printf(i == 0? "%d" : "\t%d", fm_get_freq(i));  
    }  
  
    return 0;  
}
```

Рис. 26. Зчитування значень з усіх драйверів і вивід отриманих значень на UART

Timestamp	Value	Value	Value	Value	Value	Value	Value	Value	Value	Value	Value
22:57:32.173>	8433	0	0	0	0	0	0	0	0	0	0
22:57:32.173>											
22:57:32.251>	8434	0	0	0	0	0	0	0	0	0	0
22:57:32.251>											
22:57:32.360>	8434	0	0	0	0	0	0	0	0	0	0
22:57:32.360>											
22:57:32.454>	8434	0	0	0	0	0	0	0	0	0	0
22:57:32.454>											
22:57:32.563>	8434	0	0	0	0	0	0	0	0	0	0
22:57:32.563>											
22:57:32.673>	8434	0	0	0	0	0	0	0	0	0	0
22:57:32.673>											
22:57:32.798>	8434	0	0	0	0	0	0	0	0	0	0
22:57:32.798>											
22:57:32.923>	8434	0	0	0	0	0	0	0	0	0	0
22:57:32.923>											
22:57:33.063>	8434	0	0	0	0	0	0	0	0	0	0
22:57:33.063>											
22:57:33.235>	8434	0	0	0	0	0	0	0	0	0	0
22:57:33.235>											

Рис. 27. Результат роботи процесорної системи з під'єднаними до неї датчиками

### Висновки з даного дослідження і перспективи подальших розвідок у даному напрямі

Синтезовано схему, для багатоканального радіотехнічного приладу на основі FPGA фірми Altera Cyclone IV, який має 12 вимірювальних каналів для сенсорів з частотним виходом і базується на гнучкому мікропроцесорному ядрі NIOS II. У якості вихідного інтерфейсу використовується широко розповсюджений цифровий протокол UART. Також для мікропроцесорного ядра було розроблено ПЗ яке дозволяє обробляти дані з частотомірів і передавати їх на UART порт. Спроектвано мікропроцесорну систему яка являється

гнучкою і дозволяє змінювати кількість вхідних сигналів від частотомірів, без зміни алгоритму обробки даних. Перероблено схему частотомірів для сумісності з інтерфейсом ядра NIOS II.

### Література

1. Кофанов В. Л. Лабораторний практикум з дослідження цифрових пристроїв на основі САПР MAX+PLUS II [Текст] : лабораторний практикум / В. Л. Кофанов, О. В. Осадчук, Д. В. Гаврілов. – Вінниця : УНІВЕРСУМ-Вінниця, 2006. – 200 с.
2. Nios II Processor Reference Handbook. – San Jose: Altera, 2016. – 260 с.
3. Borgonovo D. Application of the NIOS II processor-FPGA on the digital control of a single-phase PFC rectifier," / D. Borgonovo, M. L. Heldwein and S. A. Mussa // 2008 11th Workshop on Control and Modeling for Power Electronics, Zurich, Switzerland, 2008, pp. 1-7, doi: 10.1109/COMPEL.2008.4634702.
4. Safarpour M. An Embedded Programmable Processor for Compressive Sensing Applications," / M. Safarpour, I. Hautala and O. Silvén // 2018 IEEE Nordic Circuits and Systems Conference (NORCAS): NORCHIP and International Symposium of System-on-Chip (SoC), Tallinn, Estonia, 2018, pp. 1-5, doi: 10.1109/NORCHIP.2018.8573494.
5. Tayara H. A Real-Time Marker-Based Visual Sensor Based on a FPGA and a Soft Core Processor / Tayara H., Ham W., Chong K.T. // Sensors. 2016; 16(12):2139. <https://doi.org/10.3390/s16122139>
6. Magdaleno E. A FPGA Embedded Web Server for Remote Monitoring and Control of Smart Sensors Networks / Magdaleno E., Rodríguez M., Pérez F., Hernández D., García E. // Sensors. 2014; 14(1):416-430. <https://doi.org/10.3390/s140100416>
7. González D. A Low Cost Matching Motion Estimation Sensor Based on the NIOS II Microprocessor / González D., Botella G., Meyer-Baese U., García C., Sanz C., Prieto-Matías M., Tirado F. // Sensors. 2012; 12(10):13126-13149. <https://doi.org/10.3390/s121013126>
8. Осадчук О.В. Багатоканальний частотомір на програмованій логічній інтегральній схемі для радіовимірювальної системи з частотними сенсорами фізичних величин / Осадчук О.В., Осадчук Я.О., Скощук В.К. // Вісник Хмельницького національного університету, №6, 2021 (303) – С.186-194. DOI 10.31891/2307-5732-2021-303-6-186-194
9. Quartus Prime Standard Edition. URL: <https://fpgasoftware.intel.com/15.1/?edition=standard&platform=windows>
10. Cyclone IV Device Datasheet. URL: <https://www.intel.com/content/dam/www/programmable/us/en/pdfs/literature/hb/cyclone-iv/cyiv-53001.pdf>.

### References

1. Kofanov V. L. Laboratory workshop on the research of digital devices based on CAD MAX+PLUS II [Text]: laboratory workshop / V. L. Kofanov, O. V. Osadchuk, D. V. Gavrilov. – Vinnytsia: UNIVERSUM-Vinnytsia, 2006. – 200 p.
2. Nios II Processor Reference Handbook. – San Jose: Altera, 2016. – 260 с.
3. Borgonovo D. Application of the NIOS II processor-FPGA on the digital control of a single-phase PFC rectifier," / D. Borgonovo, M. L. Heldwein and S. A. Mussa // 2008 11th Workshop on Control and Modeling for Power Electronics, Zurich, Switzerland, 2008, pp. 1-7, doi: 10.1109/COMPEL.2008.4634702.
4. Safarpour M. An Embedded Programmable Processor for Compressive Sensing Applications," / M. Safarpour, I. Hautala and O. Silvén // 2018 IEEE Nordic Circuits and Systems Conference (NORCAS): NORCHIP and International Symposium of System-on-Chip (SoC), Tallinn, Estonia, 2018, pp. 1-5, doi: 10.1109/NORCHIP.2018.8573494.
5. Tayara H. A Real-Time Marker-Based Visual Sensor Based on a FPGA and a Soft Core Processor / Tayara H., Ham W., Chong K.T. // Sensors. 2016; 16(12):2139. <https://doi.org/10.3390/s16122139>
6. Magdaleno E. A FPGA Embedded Web Server for Remote Monitoring and Control of Smart Sensors Networks / Magdaleno E., Rodríguez M., Pérez F., Hernández D., García E. // Sensors. 2014; 14(1):416-430. <https://doi.org/10.3390/s140100416>
7. González D. A Low Cost Matching Motion Estimation Sensor Based on the NIOS II Microprocessor / González D., Botella G., Meyer-Baese U., García C., Sanz C., Prieto-Matías M., Tirado F. // Sensors. 2012; 12(10):13126-13149. <https://doi.org/10.3390/s121013126>
8. Osadchuk O.V. A multi-channel frequency counter on a programmable logic integrated circuit for a radio measuring system with frequency sensors of physical quantities / Osadchuk O.V., Osadchuk Ya.O., Skoshchuk V.K. // Bulletin of the Khmelnytskyi National University, No. 6, 2021 (303) – P.186-194. DOI 10.31891/2307-5732-2021-303-6-186-194
9. Quartus Prime Standard Edition. URL: <https://fpgasoftware.intel.com/15.1/?edition=standard&platform=windows>
10. Cyclone IV Device Datasheet. URL: <https://www.intel.com/content/dam/www/programmable/us/en/pdfs/literature/hb/cyclone-iv/cyiv-53001.pdf>.