

УДК 519.876.5

Поморова О.В, д.т.н., проф., Тітова В.Ю., к.т.н, доцент, Медзатий Д.М., к.т.н, доцент
Хмельницький національний університет

ДОСВІД ВИКОРИСТАННЯ ОБЧИСЛЮВАЛЬНОГО ПРИСТРОЮ DE1-SOC У НАВЧАЛЬНОМУ ПРОЦЕСІ ТА НАУКОВИХ ДОСЛІДЖЕННЯХ КАФЕДРИ СИСТЕМНОГО ПРОГРАМУВАННЯ ХНУ

Поморова О.В, Тітова В.Ю., Медзатий Д.М. Досвід використання обчислювального пристрою DE1-SOC у навчальному процесі та наукових дослідженнях кафедри системного програмування ХНУ. У статті пропонується використання плати DE1-SOC, яка побудована за сучасною технологією ПЛІС, в якості учбового стенду для навчального процесу кафедри системного програмування Хмельницького національного університету. Плата являє собою автономний зовнішній пристрій, який підключається до ПК за допомогою розташованих на ній портів та інтерфейсів. Її використання в навчальному процесі дозволяє студентам краще зрозуміти принципи роботи сучасних обчислювальних пристроїв, проектувати власні пристрої та тестувати їх роботу за допомогою ресурсів плати.

Ключові слова: комп'ютерна інженерія, програмовані логічні інтегральні схеми, DE1-SOC, навчальний процес.

Поморова О.В, Тітова В.Ю., Медзатий Д.М. Опыт использования вычислительного устройства DE1-SOC в учебном процессе и научных исследованиях кафедры системного программирования ХНУ.

В статье предлагается использование платы DE1-SOC, которая построена по современной технологии ПЛИС, в качестве учебного стенда для учебного процесса кафедры системного программирования Хмельницкого национального университета. Плата представляет собой автономное внешнее устройство, которое подключается к ПК с помощью расположенных на ней портов и интерфейсов. Ее использование в учебном процессе позволяет студентам лучше понять принципы работы современных вычислительных устройств, проектировать собственные устройства и тестировать их работу с помощью ресурсов платы.

Ключевые слова: компьютерная инженерия, программируемые логические интегральные схемы, DE1-SOC, учебный процесс.

Pomorova O.V., Titova V.Yu., Medzaty D.M. Experience of using the computing device DE1-SOC for the educational process and scientific research at the System Programming department of KhNU. Using the board DE1-SOC, which is built on a modern CPLD technology, as a training bench for the educational process at the System Programming department of Khmelnytsky National University is proposed in this paper. The board is a standalone external device that connects to a PC by ports and interfaces located on it. Using the board provides to students better understanding the work principles of modern computing devices, designing own devices and testing their work by using the board resources.

Keywords: Computer Engineering, Complex Programmable Logic Device, DE1-SOC, educational process.

Вступ. На сьогоднішній день, комп'ютерна інженерія це - технічні (апаратні) засоби та системне програмне забезпечення комп'ютерних систем і мереж універсального і спеціального призначення, а також їх компоненти. Зміст діяльності фахівців з комп'ютерної інженерії полягає в розробці апаратно-програмних засобів сучасних і перспективних інформаційних технологій, розробці та застосуванні комп'ютерних систем і мереж загального та спеціального призначення, їх системного програмного забезпечення, спеціалізованих комп'ютерних систем і мереж з оптимізованими параметрами, інтегрованих комп'ютерних систем, технічних засобів захисту інформації в комп'ютерних системах і мережах. Оволодіння зазначеними навичками та компетенціями неможливе без використання у процесі навчання сучасних комп'ютерних засобів.

Одним з перспективних напрямків сучасних технологій розробки апаратно-програмних засобів є технологія програмованих логічних інтегральних схем (ПЛІС). Її сучасний розвиток дозволив не тільки замінити ПЛІС мікросхеми малого та середнього ступеня інтеграції у спеціалізованих цифрових та комп'ютерних пристроях, але й дозволив використовувати ПЛІС у навчальному процесі в якості учбових стендів.

Серед переваг такого використання можна виділити наступні [1-2]:

- підтримка реалізації складних цифрових електронних схем та пристроїв, які за своєю потужністю не поступаються сучасним процесорам;
- простота реалізації зазначених схем та пристроїв за рахунок використання систем автоматизованого проектування (САПР), які пропонують розробники ПЛІС.

Учбові стенди на ПЛІС можуть бути реалізовані в трьох варіантах:

- як автономний зовнішній пристрій, програмування якого здійснюється за допомогою персонального комп'ютера (ПК) через спеціальний інтерфейс JTAG;
- як плата розширення ПК з інтерфейсом шини PCI, PCI Express, AGP або інших;

- як автономний зовнішній пристрій, який підключається до ПК за допомогою послідовних або паралельних інтерфейсів (USB, LPT тощо).

Перевагою першого варіанту є те, що такий стенд може працювати автономно без ПК. Перевагою другого – можливість обміну даними з ПК в режимі реального часу. Третій варіант поєднує в собі переваги двох попередніх, а тому надає більш широкі можливості для навчального процесу.

Одним з прикладів ПЛІС, яка відноситься саме до третього класу, є обчислювальний пристрій DE1-SoC від Terasic Technologies. Він використовується в якості учбового стенду на кафедрі системного програмування Хмельницького національного університету.

Зазначений обчислювальний пристрій містить двоядерний процесор ARM Cortex-A9, інтегровану оперативну пам'ять та множини інтерфейсів шин та портів, які використовуються для підключення сучасної периферії. Його структура наведена на рис. 1.

Даний стенд використовується в рамках вивчення студентами предмету «Архітектура комп'ютерів». Предмет розрахований на два семестри, та складається з 15 лабораторних робіт.

Метою лабораторних робіт першого семестру є вивчення можливостей мови програмування VHDL, яка є базовою мовою при розробці апаратури сучасних обчислювальних цифрових систем [3].

Для цього студентам даються завдання з проектування комбінаційних схем, простих обчислювальних пристроїв, зокрема таймерів, лічильників адрес та інших. Наявні на платі засоби візуалізації: світлодіоди та цифрові дисплеї дозволяють наочно перевірити правильність функціонування спроектованих пристроїв та зрозуміти принципи їх роботи.

Метою лабораторних робіт другого семестру є набуття студентами навичок у розробці:

- апаратних засобів обміну даними між процесором і зовнішніми пристроями, обробки переривань програми та прямого доступу до пам'яті;
- мікропроцесорів та мікроконтролерів на базі мікропроцесорних комплектів ВІС.

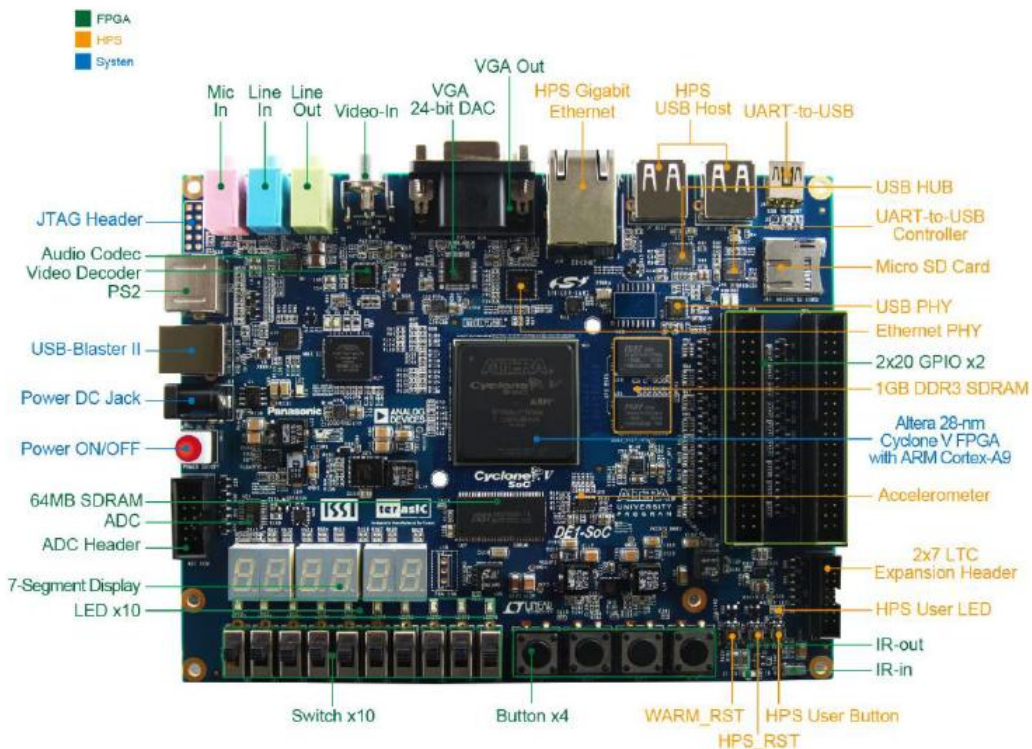


Рис. 1. Структура обчислювального пристрою DE1-SoC, який використовується в якості учбового стенду

Для цього студентам надаються різноманітні завдання, серед яких можна виділити:

- реалізацію арифметично-логічного пристрою (АЛП), який виконує арифметичні та логічні операції, задані в індивідуальному завданні, та встановлює задані у завданні прапорці. Ресурси плати дозволяють протестувати роботу реалізованого АЛП та краще зрозуміти принцип його функціонування;

- реалізацію основної та стекової пам'яті заданого обсягу, доступ до яких здійснюється за алгоритмами, заданими в індивідуальному завданні. В ході виконання лабораторної роботи студенти мають змогу використовувати інтегровану на плату пам'ять та тестувати різні способи доступу до неї;

- реалізацію послідовного та паралельного портів, які функціонують за заданими в індивідуальному завданні алгоритмами. Ресурси плати дозволяють використати реалізовані порти для підключення зовнішніх пристроїв та візуалізації передачі інформації з них;

- реалізацію керуючого автомату, який виконує дії або обчислення згідно індивідуального завдання, та мікропроцесора за архітектурою, заданою в індивідуальному завданні.

Набуття навичок в реалізації подібних автоматів та мікропроцесорів дозволить студентам проектувати та програмувати мікроконтролери та мікропроцесори в їх майбутній професійній діяльності.

Реалізація усіх зазначених пристроїв здійснюється за допомогою САПР Quartus II 15.0. Дана програма відрізняється зручним інтерфейсом, який значно спрощує процес проектування та розробки будь-яких пристроїв та дозволяє не тільки виконувати завдання в межах лабораторних робіт, але й створювати повноцінні апаратні реалізації пристроїв, які можуть бути використані в сучасному виробництві.

Приклад виконання лабораторних робіт за допомогою САПР Quartus II 15.0 наведено на рис. 2-5. За допомогою програмного коду студенти можуть створювати власні пристрої: прості, такі як мультиплексор (рис. 2) або декодер, та складні, такі як арифметично-логічний пристрій (рис. 3) або керуючий автомат (рис. 4). За допомогою графічних файлів існує можливість поєднувати пристрої між собою, створюючи цифрові схеми на їх основі (рис. 5).

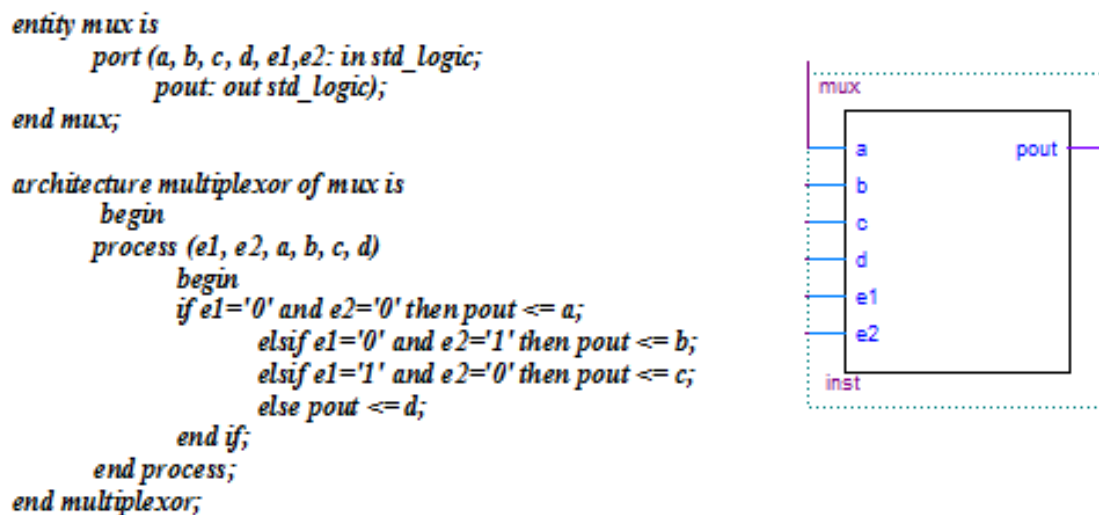


Рис. 2. Приклад створення пристрою, який працює за принципом мультиплексора

Окрім навчального процесу, плата використовується як стенд для наукових досліджень та студентських проектів.

На сьогоднішній день студентські колективи працюють над наступними проектами:

- реалізація на базі ПЛІС логічних відеоігор та гральних автоматів;
- реалізація на базі ПЛІС системи оповіщення;
- реалізація на базі ПЛІС системи керування електричною підстанцією.

```

process ( reset, op_code, in1, in2)
  variable tmp_out: std_logic_vector (15 downto 0);
begin
  if reset = '1' then
    tmp_out:= "0000000000000000";
  case op_code is
    when op_add =>
      out1<=in1 + in2;
      flag_ov<= 'Z';
    when op_mul =>
      tmp_out:= in1 * in2;
      out1<=tmp_out(7 downto 0);

    if (tmp_out (15 downto 8) /= "00000000") then
      flag_ov<= '1';
    else
      flag_ov<= '0'; end if;
    when op_rol =>
      tmp_out (7 downto 0):= in1;
      out1(7 downto 1)<= tmp_out (6 downto 0);
      out1(0)<= tmp_out (7);
      flag_ov<= 'Z';
    when others =>
      out1<= "ZZZZZZZZ"; flag_ov<= 'Z';
    end case;
  else out1<="00000000"; flag_ov<= 'Z'; end if;
end process;
end behaviour;

```

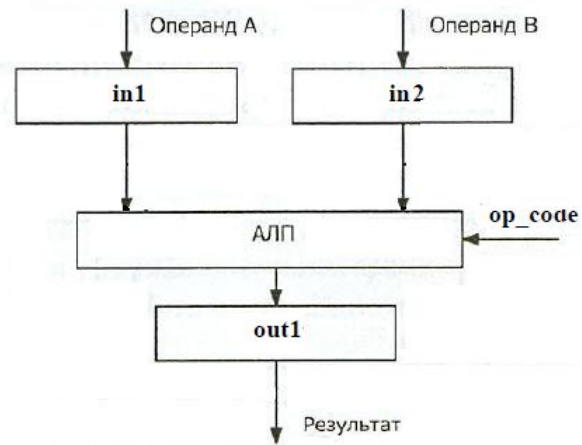


Рис.3. Приклад створення арифметичного-логічного пристрою, який виконує операції додавання, множення та циклічного зсуву.

На основі обчислювального пристрою виконується магістерська робота, присвячена апаратній реалізації природничих алгоритмів, таких як: алгоритм рою бджіл, мурашиний алгоритм, генетичний алгоритм, алгоритм зграї вовків та інші, та дисертаційна робота, присвячена виявленню поліморфних вірусів в операційних системах та комп'ютерних мережах. У цій роботі обчислювальний пристрій виконує роль плати-прискорювача роботи персонального комп'ютера, яка містить спеціальне програмне забезпечення та здійснює моніторинг програмного коду на наявність у ньому тіл вірусів.

Висновки. Сучасні навчальні стенди на основі ПЛІС дозволяють значно розширити можливості навчального процесу при підготовці студентів технічних спеціальностей, зокрема спеціальності комп'ютерна інженерія.

Студенти використовують такі стенди при виконанні лабораторних та практичних робіт, оволодіваючи навичками розробки сучасних програмно-апаратних засобів.

На основі стендів проводяться наукові та дослідницькі роботи, що дозволяє перевірити теоретичні розробки на практиці та отримати більш якісні наукові результати.

Також сучасні ПЛІС використовуються у якості плат-прискорювачів, виконуючи емуляцію таких програмних засобів, як антивіруси, штучні імунні системи, системи моніторингу ресурсів та вивільняючи ресурси комп'ютера для опрацювання поточних задач користувача.

Напрямом подальших робіт є розроблення нових лабораторних робіт, спрямованих на розробку студентами різноманітних обчислювальних та керуючих пристроїв, та на розширення тематик наукових і магістерських робіт, які можуть бути виконані на основі обчислювального пристрою DE1-SoC.


```

process (start, stan_next) -- реєстр поточного стану
-- якщо сигнал запуску дорівнює 0, автомат знаходиться в 000 стані
-- якщо сигнал запуску дорівнює один автомат переходить зі стану в стан
begin
if start='0' then
STAN<="000"; else stan<=stan_next; end if;
end process;

process (stan) -- логіка переходів
-- визначає яке значення отримає поточний стан наступної миті
begin
if stan="000" then stan_next<="001"; end if;
if stan="001" then stan_next<="010"; end if;
if stan="010" then stan_next<="011"; end if;
if stan="011" then stan_next<="100"; end if;
if stan="100" then stan_next<="101"; end if;
if stan="101" then stan_next<="110"; end if;
if stan="110" then stan_next<="111"; end if;
end process;

process (stan) -- логіка виходу
-- кожному поточному стану автомату у відповідність ставить певна дія або множина дій
variable m:std_logic_vector (0 to 4);
begin
if stan="000" then y1<='0'; y2<='0'; y3<='0'; y4<='0'; y5<='0'; end if;
if stan="001" then m(0):=a1; m(1):=a2; m(2):=a3; m(3):=a4; m(4):=a5; end if;
if stan="010" then m(0):=not m(0); end if;
if stan="011" then m(1):=not m(1); end if;
if stan="100" then m(2):=not m(2); end if;
if stan="101" then m(3):=not m(3); end if;
if stan="110" then m(4):=not m(4); end if;
if stan="111" then y1<=m(0); y2<=m(1); y3<=m(2); y4<=m(3); y5<=m(4); end if;
end process;
    
```

Рис.4. Приклад створення керуючого автомату, який перетворює двійкове число в обернений код

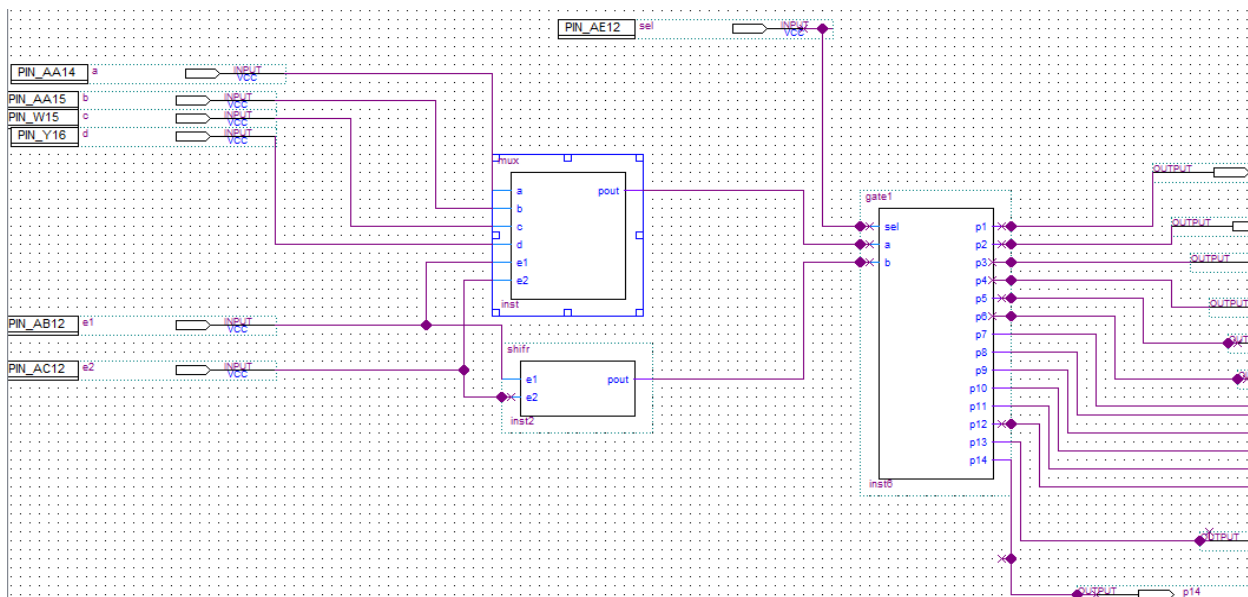


Рис. 5. Поєднання створених пристроїв на схемі та їх прив'язка до ніжок плати

1. Стешенко В.Б. ПЛИС фирмы ALTERA: проектирование устройств обработки сигналов Москва: Додэка, 2000. — 128 с.
2. Грушвицкий Р.И., Мурсаев А.Х., Угрюмов Е.П. Проектирование систем на микросхемах программируемой логики СПб.: БХВ-Петербург, 2002. — 608 с.
3. Поляков А. К. Языки VHDL и Verilog в проектировании цифровой аппаратуры. — М.: СОЛОН-Пресс, 2003. — 320 с.
4. DeSoC User Manual / - http://courses.cs.washington.edu/courses/cse467/15wi/docs/DE1_SoC_User_Manual.pdf