

ОСОБЛИВОСТІ РОЗРОБКИ ПРОГРАМНИХ ЗАСОБІВ ПОКОМПОНЕНТНОГО ДІАГНОСТУВАННЯ ЦИФРОВИХ ПРИСТРОЇВ В УМОВАХ ВИРОБНИЦТВА

В статті розглядаються особливості розробки програмних засобів для систем покомпонентного діагностування схем цифрових пристроїв. В результаті аналізу літератури було запропоновано структурну схему програмних засобів систем внутрішньосхемного пошуку несправностей, яка являється найбільш ефективною для широкого класу цифрових пристроїв в умовах їх серійного виробництва.

Ключові слова: діагностування, тестування, структура, компонент, комутація, генераційні процеси.

S. I. PEREVOZNIKOV, N. O. BILICHENKO, V. S. OZERANSKYI
Vinnytsia national technical university

FEATURES OF THE SOFTWARE COMPONENT-BY-COMPONENT DIAGNOSIS OF DIGITAL DEVICES IN A PRODUCTION ENVIRONMENT

Abstract – the aim of the research – discusses features of the design software for systems of component-by-component of diagnostics digital devices.

The analysis of modern methods and systems in a production environment exercise testing digital devices of varying complexity. The choice of methods and tools based on the economic feasibility of their use.

As a result of the experience gained from leading manufacturers of digital equipment was introduced software block diagram of the directional troubleshooting that is most effective for a wide class of digital devices in a production environment.

Keywords: diagnostic, testing, structure, component, switching, generation processes.

Вступ

Поряд з існуючими методами і способами автоматичного пошуку несправностей в цифрових пристроях (ЦП) особливе місце займають системи покомпонентного діагностування (СПД). Привабливість використання подібних засобів відома: вона найбільш успішна в умовах середнього і багатосерійного виробництва ЦП при достатній кількості внутрішніх вузлів пристроїв, до яких здійснюється конструктивний доступ (контактні площадки для голчатого адаптера з боку СПД). Аналіз сучасних фірм-виробництв тестового обладнання показав, що ефективність стратегій діагностування досягається при обґрунтованому підборі різних методів тестування на всіх технологічних етапах виготовлення ЦП.

Аналіз досліджень та публікацій

Слід відмітити, що системи покомпонентного (пофрагментного) діагностування представляють собою функціональне об'єднання і взаємодію апаратного та програмного забезпечення. При цьому апаратна частина якомога більше спрямована на прискорення процесу тестування і аналізу отриманих результатів. Програмне забезпечення має специфічну можливість підготувати об'єкт діагностування у «найкращий» спосіб завдяки можливості деструктуризації схеми ЦП засобами тестування (шляхом організації введення штучних каналів передачі інформації між внутрішніми вузлами об'єкту). Для таких систем ефективність зростає зі зростом контактних вузлів.

В основі програмування процесів пошуку підструктур тестування лежить бібліотека стандартних і нестандартних тестів ОД. Сучасні бібліотеки тестів спрямовані на розв'язання як специфічних задач, так і на вирішення загальних питань тестового пошуку несправностей, наприклад, розбиття початкової структури схеми ЦП, на множину підструктур, упорядкування або композиція фрагментів схем із елементів для їх подальшого тестування. Але, при цьому, слід зазначити, що незалежно від складностей та властивостей ЦП, а також від можливостей самих засобів, існують загальні принципи організації програмного простору для реалізації процедур пошуку несправностей.

Постановка задачі

Досвід фірм-виробників сучасних засобів діагностування показав, що неможливо в повній мірі формалізувати весь процес побудови програмної підтримки в системах такого класу. Це приводить до об'єднання різних методів побудови системи з метою визначення ефективної структури СПД для широкого класу об'єктів дослідження.

Пошук більш універсальної структури зумовлено рядом обставин, серед яких найбільш впливовими є:

- спрощення інтерфейсу користувача з СПД;
- спрощення підготовки програм діагностування ЦП;
- підвищення ефективності процедур діагностичних експериментів на етапі моделювання програм пошуку несправностей.

Отже, необхідно сформулювати таку структуру програмних засобів СПД, яка б задовольняла вищезгаданим вимогам.

Розроблення програмних засобів систем діагностування ЦП

Спрямованість програмного забезпечення СПД ґрунтується на ефективному сполученні експлуатаційних характеристик і розробки автоматизованих процедур пошуку несправностей у ЦП. При цьому враховується також природне підвищення кваліфікації програмістів та адаптація програм до поточних апаратних ресурсів СПД, що з часом швидко змінюються [1]. Сукупність складових процедур, що включаються у СПД, складає систему програмного забезпечення, структурну схему якої представлено на рис. 1.

Отже, усе програмне забезпечення СПД можна умовно поділити на складові:

- системне програмне забезпечення;
- спеціалізоване програмне забезпечення, яке включає в себе банк знань.

В свою чергу, системне програмне забезпечення являє собою:

- операційні системи (призначені для управління ЕОМ);
- тестування

функціональний модуль СПД (призначені для тестового контролю виконання функцій внутрішніх блоків СПД);

- протоколи обміну інформацією між центральною ЕОМ і спеціалізованим обладнанням (програмування драйверів для взаємозв'язку нестандартного обладнання з центральною ЕОМ).

Спеціалізоване програмне забезпечення представляє собою

комплекс програмних модулів, який призначено для автоматизації процесу програмування, налагодження робочої програми діагностування (РПД), написання прикладних програм обробки інформації. Вони дозволяють на основі компілятора мови покомпонентного діагностування (МПД) сформувати з первинного опису структури, будь-якого фрагмента схеми ЦП, її повне представлення. Слід зазначити, що з повного опису МПД-програми генерується код РПД-програми, який потім записується у відповідну бібліотеку РПД-програм апаратних засобів. Код такої програми є основою для ретрансляції «назад» у МПД-програму, а також для формування масивів, необхідних для графічного відображення компонентів ЦП та їх зв'язків у підсхемах, що тестуються апаратними засобами [2].

Інструментарій відлагодження програм тестування представляє собою відомі комп'ютерні засоби для написання, відлагодження і запуску програм пошуку несправностей у схемах ЦП, що досліджуються.

Бібліотека стандартних і нестандартних тестів включає в себе готові програми тестування елементів (компонентів) як без порушення внутрішньої структури ЦП, так і з її зміною за рахунок введення тимчасових додаткових зв'язків з боку СПД, які в наступному програмно усуваються.

На вибір стратегії тестування впливають особливості об'єднання елементів у фрагменти, формування програм їх тестування а також фізичні особливості конструктивної побудови схеми ЦП. Пошук їх раціонального сполучення призводить до аналізу і вибору найбільш раціональної стратегії діагностування, яка задовольняє умовам і фізичним обмеженням виробництва цифрових пристроїв.

Система підготовки програм тестування має у своєму складі мовне забезпечення, що спрямоване на моделювання і відлагодження діагностичних процедур, коли об'єкт дослідження тестується у стендовому режимі.

Особливість мовного забезпечення таких СПД пояснюється існуванням операторів, які фіксують, наприклад, тільки зміни рівнів тестових сигналів в конкретній контрольній точці схеми ЦП.

Таким чином, мовне забезпечення має власну ієрархію:

- мови, що описують взаємодію між окремими етапами діагностування;
- мови, що описують сам процес тестового пошуку несправностей у ЦП.

Так, на старшому рівні ієрархії знаходяться мови, за допомогою яких здійснюється діалог оператора з СПД. При цьому розрізняють такі категорії користувачів:

- оператор системи – користувач, що безпосередньо контролює процес діагностування у цеху

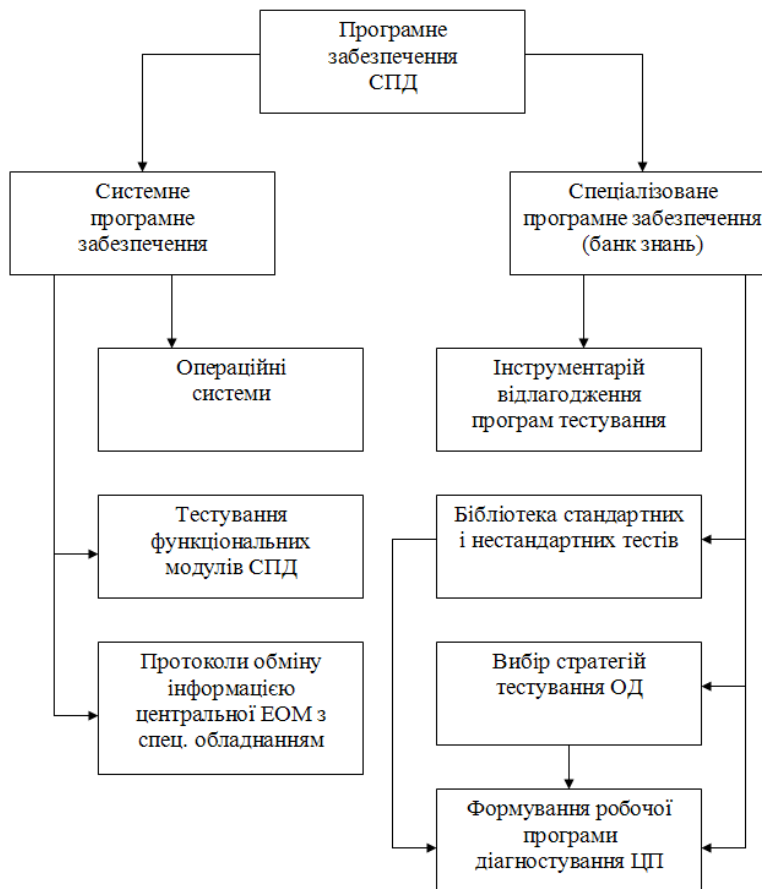


Рис. 1. Структурна схема програмного забезпечення СПД

підприємства;

- розробник програм тестування ЦП, який відлагоджує самі програми діагностування ОД;
- системний програміст, що стежить за розподілом обчислювальними ресурсами СПД,
- оператор ремонту несправних плат [3].

Розглянемо, наприклад, мову CPD-2 [3], призначену для опису програм векторного тестування цифрових схем. Вона має виразність, забезпечує простоту використання, а також гнучкість, що дозволяє адекватно описувати різні топологічні особливості (або зміни складу) фрагментів ЦП. Структура операторів такої мови містить в собі наступні лексеми:

<i>I1</i> – позначення фрагмента схеми	- ‘ <i>F</i> ’,	<i>I11</i> – роздільник тестових рядків	- ‘/’,
<i>I2</i> – ознака входів комутаторів	- ‘ <i>W</i> ’,	<i>I12</i> – ознака закінчення тесту для фрагмента схеми	- ‘:’,
<i>I3</i> – ознака виходів комутаторів	- ‘ <i>R</i> ’,	<i>I13</i> – ознака закінчення програми тестування всієї схеми	- ‘!!’,
<i>I4</i> – ознака забезпечення електричних умов діагностування фрагмента схеми	- ‘ <i>S</i> ’,	<i>I14</i> – позначення маски (частина номерів виводів комутаторів)	- ‘ <i>M</i> ’,
<i>I5</i> – ознака подачі тестового вектора на фрагмент схеми	- ‘ <i>T</i> ’,	<i>I15</i> – ознака з’єднання фрагментів схеми (контурів) по входах	- ‘ <i>Vi</i> ’,
<i>I6</i> – ознака інвертування розрядів у тестовому слові	- ‘ <i>P</i> ’,	<i>I16</i> – ознака з’єднання фрагментів схеми (контурів) по виходах	- ‘ <i>Vo</i> ’,
<i>I7</i> – позначення номерів виводів комутаторів	- (1÷999)	<i>I17</i> – ознака подання блокуючого сигналу	- ‘ <i>BG</i> ’,
<i>I8</i> – ознака входів фрагмента схеми, що тестується	- ‘ <i>Z</i> ’,	<i>I18</i> – ознака зняття блокуючого сигналу	- ‘ <i>ZG</i> ’,
<i>I9</i> – ознака виходів фрагмента схеми, що тестується	- ‘ <i>E</i> ’,	<i>I19</i> – ознака наявності генерації в контурі	- ‘ <i>G</i> ’,
<i>I10</i> – роздільник чисел у структурі оператора	- ‘;’,	<i>I20</i> – ознака наявності x-стану в контурі	- ‘ <i>X</i> ’.

Складання програм діагностування цифрових фрагментів має свої особливості. Так, з метою організації діагностування схеми ЦП генераторним способом додатково введені ряд лексем *I15* – *I20*.

I15 – ознака з’єднання сформованих контурів по входах після встановлення початкових умов для комутації внутрішніх вузлів схеми ЦП.

I16 – ознака з’єднання сформованих контурів по виходах після встановлення електричних умов комутації внутрішніх вузлів схеми ЦП.

I17 – ознака подання сигналу для блокування самогенерації у сформованих контурах.

I18 – ознака зняття сигналу для блокування самогенерації у сформованих контурах (власне запуск процесу самогенерації в контурах).

Лексема *I19* застосовується для відображення особливостей протікання процесу генерації у контурах на швидкодіючих елементах. Вона дозволяє засобом СПД використовувати часові узгодження з об’єктом діагностування.

Лексема *I20* застосовується для відображення особливостей протікання процесу самогенерації у контурах на елементах, що мають невисоку швидкодію. Вона дозволяє засобом СПД використовувати тільки вимірювання напруги у будь-яких вузлах з’єднання сформованих контурів, при чому значення напруги повинно бути в межах 1,5 ÷ 1,7 В.

Опис лексем *I1* – *I14* детально розглянуто в [4].

Розглянемо приклади програм тестування фрагментів схем, описаних мовою CPD-2.

Приклад 1.

1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22
F I W I5 R 42 T M 42 / Z I5 E 42 / Z BG Vi ZG E G ;

де	1 – позначення фрагмента схеми,	13 – позначення виходів комутаторів, у яких повинна бути логічна '1' після подачі тестового вектора,
	2 – номер фрагмента,	14 – номер виходу фрагмента схеми,
	3 – позначення входу комутатора,	15 – роздільник тестових векторів,
	4 – номер входу комутатора,	16 – позначення входів фрагмента,
	5 – позначення виходів комутатора,	17 – подання блокуючого сигналу,
	6 – номер виходу комутатора,	18 – позначення з’єднання входів контурів,
	7 – позначення тестового вектора,	19 – зняття блокуючого сигналу,
	8 – позначення маски,	20 – позначення виходів фрагмента,
	9 – номер виводу комутатора,	21 – позначення вимірів генерацій цих процесів,
	10 – роздільник тестового вектора,	22 – кінець тестування фрагмента схеми.
	11 – позначення входу, куди подається сигнал логічної '1',	
	12 – номер входу фрагмента схеми,	

Приклад 2.

F15 W 20,21, 34, R 42, 34 S 4 M 8 /20, 21/ 21, 34/ T M 42/ 32, 42 / ;

де *M 8* – позначення числа повторів пари векторів, що слідує за маскою в тестовому запису / 20, 21/ 21, 34/.

Таким чином, замість 8 рядків записуються тільки 2.

При наявності несправностей в ОД на системний монітор виводиться діагностичне повідомлення списку несправних елементів. Інформацію про такі елементи можна деталізувати, наприклад, повідомляючи тип та координати несправності.

Невід'ємною складовою системного програмного забезпечення є операційна система (ОС). Основною задачею ОС є введення РПД в пам'ять центральної ЕОМ (при включенні обладнання), а також «запуск» програм управління в заданому режимі функціонування для послідовного виконання тестів сформованих компонентів розбиттів.

Операційна система виконує такі основні функції:

- запуск керуючої програми;
- завантаження робочої програми діагностування об'єкту дослідження;
- періодичне підключення системи самоконтролю засобів діагностування;
- незалежний контроль працездатності усіх модулів СПД;
- збір та обробка інформації про відмови і збої самої СПД.
- послідовна активізація підпрограм виконання тестів;
- формування масивів несправностей;
- графічне відображення несправних компонентів та детальну інформацію про них;
- активізація підсистеми обробки результатів тестового опитування компонентів ЦП та процедур

можливого уточнення (контрольний повтор тесту);

Достовірність функціонування СПД ґрунтується на періодичному тестовому опитуванні всіх основних модулів системи. Така операція, як правило, здійснюється перед початком режимів діагностування об'єктів дослідження. При цьому враховується час тестування та пріоритет контролю складових модулів СПД. Деякі модулі проходять таку перевірку частіше (тобто мають вищий пріоритет), наприклад, контактний адаптер, блок аналогових вимірів. Інші модулі, у яких пріоритет нижчий, перевіряються через більш тривалий час. До цієї купи відносять модуль комутації каналів, модуль аналогових вимірів, тощо.

На рис. 2 представлено структурну схему системи самоконтролю для засобів внутрішньосхемного діагностування цифрових пристроїв.

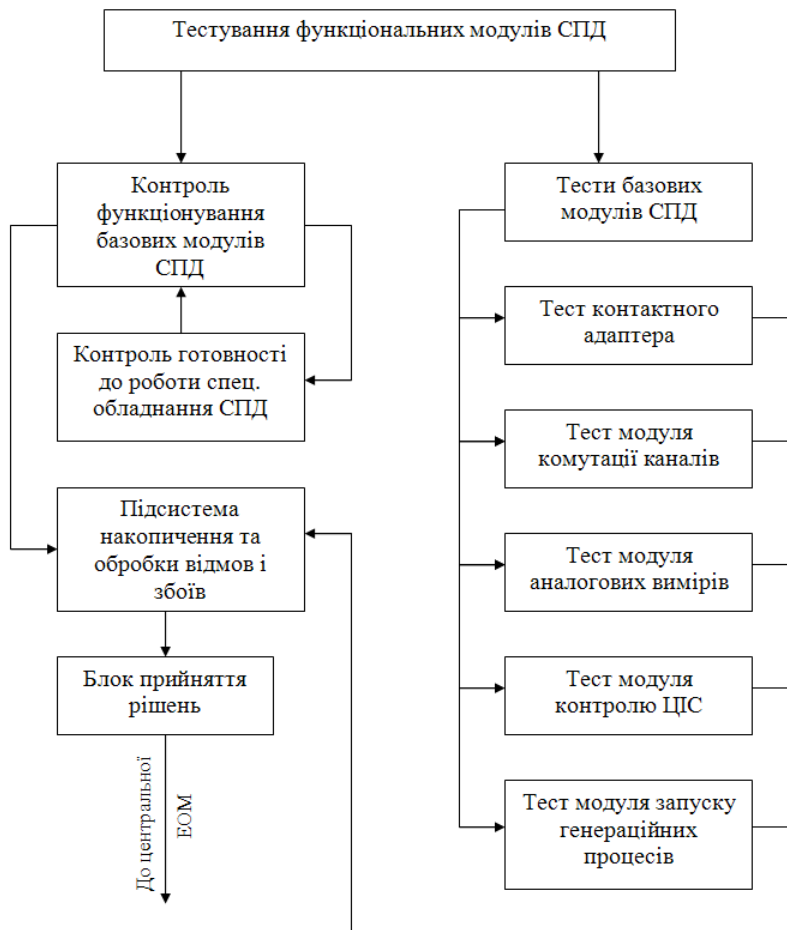


Рис. 2. Структурна схема тестування основних модулів СПД

Тестування функціональних модулів СПД можна поділити на дві основні групи:

- контроль працездатності базового обладнання СПД;
- безпосередні тести складових модулів СПД.

Тести складових модулів СПД розподіляють на:

- тест контактного адаптера (опитується кожна немаскована голка);
- тест модуля комутації каналів (перевіряються всі електронні ключі);
- тест модуля аналогових вимірів (перевіряється правильність балансування еталонного мосту);
- тест модуля контролю ЦПС (опитується пристрій управління модуля контролю ЦПС);
- тест модуля запуску генерацій цих процесів (опитується основні складові модуля).

Застосування такої системи самоконтролю значно підвищує вірогідність функціонування СПД в умовах середнього та крупносерійного виробництва. При цьому можна виділити функції, що покладені на неї:

- виявлення раптових відмов (особливо збоїв, що можуть привести, наприклад, до пропуску несправного фрагмента схеми або навпаки);
- виявлення поступових відмов під час тривалого функціонування СПД.

Особливим режимом самоконтролю СПД є перевірка її готовності. При цьому виявляються поступові відмови (вихід параметрів елементів за припустимі межі) основних модулів системи діагностування. Накопичення та обробка інформації про відмови та збої модулів дозволяє швидко локалізувати місце несправності СПД у режимі самоконтролю. Включення режиму самоконтролю здійснюється автоматично із заздалегідь обраним періодом.

Прикладною програмою для СПД є РПД-програма. РПД являє собою набір тестів, призначених для діагностування компонентів ЦП. Первісним джерелом для РПД є трансляція МПД-програми.

Висновки

Запропонований підхід щодо формування структури програмного забезпечення СПД сприяє скороченню загального часу пошуку несправностей у схемах цифрових пристроїв за рахунок особливостей здійснення фізичних процесів, які протікають в комутованих ланцюгах схем. При цьому в кожному вимірі (згідно алгоритмів тестування) міститься інформація про окремий ланцюг штучної структури, що дає можливість прискорити локалізацію спрямованого пошуку несправностей.

Управління такими процесами спрямованого пошуку несправностей виконується відповідним програмним забезпеченням. Слід відмітити, що порядок з'єднання елементів і отримання зворотних реакцій контролюється програмними засобами в реальному часі (але в статичному режимі).

Література

1. Очкуров М.А. Стратегії прискореного діагностування цифрових схем / С.І. Перевозніков, М.А. Очкуров, В.С. Озеранський // Інформаційні технології та комп'ютерна інженерія. – 2008. – № 1 (11). – 12 с.
2. Аналіз композиційного підходу формування штучних фрагментів цифрових схем, як об'єктів внутрішньосхемного діагностування / [Перевозніков С.І., Біличенко Н.О., Очкуров М.А., Озеранський В.С.] – Інформаційні технології та комп'ютерна інженерія. – 2007. – № 2 (9). – 6 с.
3. Перевозніков С.І. Особливості формування компонентних структур тестування для систем внутрішньосхемного пошуку несправностей цифрових пристроїв / С.І. Перевозніков, В.С. Озеранський, Л.В. Крупельницький // Вісник Вінницького політехнічного інституту. – 2012. – № 12. – С. 62–71.
4. Перевозніков С.І. Формування компонентних структур тестування цифрових пристроїв на основі їх графового представлення / С.І. Перевозніков, В.С. Озеранський, І.Р. Арсенюк // Інформаційні технології та комп'ютерна інженерія. – 2012. – № 2 (24). – С. 56–61.

References

1. Ochukov M.A., Perevoznikov S.I., Ozeranskyi V.S. Strategii pryskorenogo diagnostuvannya tsyfrovyyh shem. Informatsijni tehnologii ta komp'yuterna inzheneriya. – 2008. – №1 (11). – 12 c. [in Ukrainian]
2. Perevoznikov S.I., Bilichenko N.O., Ochukov M.A., Ozeranskyi V.S. Analiz kompozytsijnogo pidhodu formyvannya shtuchnyh fragmentiv tsyfrovyyh shem, yak obyektiv vnytrishn'oshemnogo diagnostyvannya. Informatsijni tehnologii ta komp'yuterna inzheneriya. – 2007. – №2(9). – 6 c. [in Ukrainian]
3. Perevoznikov S.I., Ozeranskyi V.S., Krupelnytskyj L.V. Osoblyvosti formyvannya komponentnyh struktur testuvannya dlya system vnytrishn'oshemnogo poshuku nespravnostej tsyfrovyyh prystroiv. Visnyk Vinnytskogo politechnogo instytutu. – 2012. - №12. – С. 62 – 71. [in Ukrainian]
4. Perevoznikov S.I., Ozeranskyi V.S., Arsenyuk I.R. Formyvannya komponentnyh struktur testuvannya tsyfrovyyh prystroiv na osnovi ih grafovogo predstavlenya. Informatsijni tehnologii ta komp'yuterna inzheneriya. – 2012. - №2(24). – С. 56–61. [in Ukrainian]

Рецензія/Peer review : 5.11.2013 р.

Надрукована/Printed : 7.2.2014 р.

Рецензент: к.т.н., доцент кафедри програмного забезпечення
Вінницького національного технічного університету Майданюк В. П.