

Jan MOCHA¹, Zbigniew ŁASKARZEWSKI²¹ POLITECHNIKA ŚLĄSKA, INSTYTUT ELEKTRONIKI² POLITECHNIKA ŚLĄSKA, INSTYTUT INFORMATYKI**Uniwersalny system testujący urządzenia elektroniczne****Mgr inż. Jan MOCHA**

Ukończył studia na Wydziale Automatyki, Elektroniki i Informatyki Politechniki Śląskiej w 2007 roku. Jest doktorantem w Instytucie Elektroniki Politechniki Śląskiej. Jego zainteresowania naukowe koncentrują się wokół układów logiki programowalnej, praktycznych aspektów wykorzystania elektroniki w aplikacjach przemysłowych i medycznych oraz zagadnień związanych z kompatybilnością elektromagnetyczną.



e-mail: jan.mocha@polsl.pl

Mgr inż. Zbigniew ŁASKARZEWSKI

Absolwent kierunku Informatyka na Wydziale Automatyki, Elektroniki i Informatyki Politechniki Śląskiej, obecnie doktorant w Instytucie Informatyki Politechniki Śląskiej. Jego zainteresowania związane są przede wszystkim z zagadnieniami bezprzewodowych sieci sensorowych, ale także problematyką komunikacji GSM, zastosowaniami i bezpieczeństwem algorytmów kryptograficznych, a także systemami wsparcia produkcji przemysłowej.



e-mail: zbigniew.laskarzewski@polsl.pl

Streszczenie

W artykule omówiono sposób działania aplikacji, funkcjonalność systemu, a także przeznaczenie uniwersalnego systemu testującego. Omówiono warstwową architekturę komunikacji pomiędzy częścią sprzętową uniwersalnego stanowiska testującego, aplikacją sterującą oraz systemem bazodanowym i wynikające z takiego podejścia zalety. Przedstawiono nowatorski sposób definiowania logiki testu w postaci wtyczek (ang. *plug-in*) w połączeniu z parametrami zdefiniowanymi w bazie danych.

Słowa kluczowe: uniwersalny system testowania, testowanie urządzeń elektronicznych, ATE, matryca wieloigłowa, testowanie funkcjonalne, SCPI, wtyczki.

Universal testing system for electronic devices**Abstract**

This paper presents universal testing system and its communication mechanisms. The system is dedicated for electronic devices testing and configuration during final stage of manufacturing. System construction was designed for quality control, efficient work, easy self-diagnostics, economic new device types deploying.

Introduction of this paper contains reference to other systems and popular methods of electronic devices testing [1, 2]. In the next clause system architecture [3] was under discussion. Easy in-circuit testing method based on special prepared testing program loaded to microcontroller of tested device was applied in the system. Physical connections of system elements were shown (Fig. 1). The third unit contains description of tester functionality [4] and functional block diagram (Fig. 2). The next part presents tiered architecture of communication between tester and control application (Fig. 3), SCPI based communication protocol [6], command exchange mechanisms between control application, tester and database server in command flow diagram (Fig. 4), aspects of test logic designing. Start of testing process is connected with loading plug-in library with test logic definition.

System emphasizes on minimal required staff skills which means both ease of application use and rapid test design process. Functionality of presented system is sufficient for testing electronic devices and makes possible to detect many potential damages.

Keywords: universal testing system, testing of electronic devices, ATE, bed of nails, functional testing, SCPI, plug-ins.

1. Wstęp

Testowanie jest nieodłącznym elementem procesu produkcji urządzeń elektronicznych, zaś urządzenia testujące (testery) stały się jednym z elementów linii technologicznej. Zgodnie z prawem dziesięciokrotności koszt wykrywania uszkodzeń oraz ich usuwania rośnie dziesięciokrotnie na każdym etapie produkcji [1].

W zależności od umiejscowienia systemu testowania na linii produkcyjnej wykorzystywane są różne metody testowania. Do końcowych testów urządzeń opuszczających linię produkcyjną wykorzystuje się testowanie funkcjonalne, bardzo często połączone z elementami testów wewnątrzobwodowych ICT (ang. *In-Circuit Testing*).

Wykorzystywane do testowania elektrycznego urządzenia, określane są jako systemy ATE (ang. *Automatic Test Equipment*) [2]. Są to systemy wysoko zautomatyzowane o bardzo dużej uniwersalności. Koszt zakupu oraz wykorzystywania systemów ATE jest za wysoki dla małych i średnich firm zajmujących się produkcją urządzeń elektronicznych. Dla małych i średnich firm istotna jest niska cena zakupu urządzenia testującego, połączona z niskimi kosztami adaptacji systemu do testowania nowych urządzeń. Równie istotna jest łatwość implementacji procedury testowej w nieskomplikowanym i przyjaznym dla użytkownika środowisku. Ze względu na małe serie produkcyjne, mniej istotna jest natomiast wydajność systemu.

W pracy opisano architekturę oraz mechanizmy wielopoziomowej komunikacji w opracowanym uniwersalnym systemie testującym. Opracowany system testujący służy do końcowych testów urządzeń opuszczających linię produkcyjną. Przeznaczony jest dla urządzeń analogowo-cyfrowych (mogących zawierać do trzech mikrokontrolerów) o małym lub średnim stopniu skomplikowania układowego i funkcjonalnego. Dedykowany jest w szczególności dla małych oraz średnich firm zajmujących się produkcją urządzeń spełniających powyższe kryteria.

2. Architektura systemu

Przyjęta metoda testowania opiera się na testach funkcjonalnych poszczególnych bloków testowanego urządzenia oraz współpracy pomiędzy poszczególnymi blokami – uproszczone testowanie funkcjonalne. Gdy specyfika testowanego urządzenia tego wymaga, przeprowadzane są testy funkcjonalne całego urządzenia. Możliwe jest również przeprowadzanie nieskomplikowanych testów ICT.

Przyjęty sposób testowania zakłada, że w czasie testu mikrokontroler testowanego urządzenia (o ile taki występuje), wykonuje dedykowany program testowy, który umożliwia jego komunikację z systemem testowym. Rozwiązanie takie pozwala na przejęcie kontroli nad testowanym urządzeniem przez system testowania. Ułatwia to testowanie bloków o bardzo różnorodnej funkcjonalności, bez potrzeby rozbudowywania zasobów sprzętowych testera, a także upraszcza proces tworzenia oraz obniża złożoność procedury testowej.

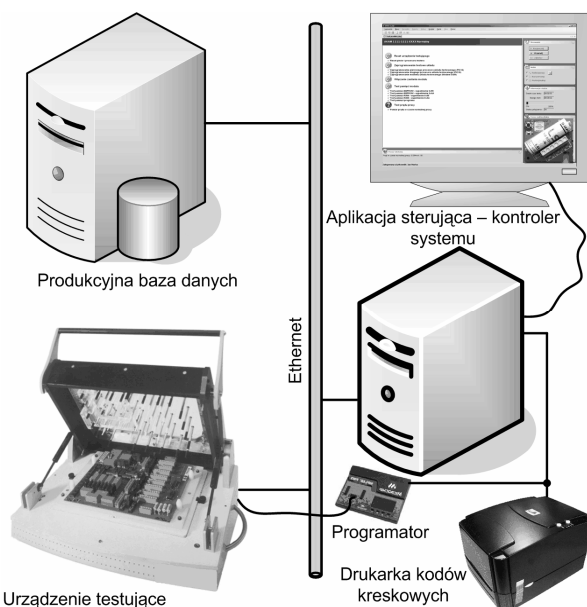
Jednym z głównych elementów opracowanego systemu testowego jest część sprzętowa – urządzenie testujące. Zadaniem części sprzętowej testera jest zadawanie wymuszeń dla obwodu testowanego oraz pobieranie odpowiedzi testowanego układu na wymuszenia. Zapewnia komunikację pomiędzy systemem testowania a testowanym urządzeniem oraz zawiera zasoby sprzętowe ułatwiające testowanie pewnych specyficznych bloków testowanych urządzeń [3, 4].

Jako interfejs pomiędzy testowanym urządzeniem, a testerem zastosowana została matryca wieloigłowa, tzw. łożo z gwoździami (ang. *bed of nails*). Zastosowanie matrycy wieloigłowej przy testowaniu mało i średnio złożonych sprzętowo urządzeń, zbudowanych

wanych głównie z elementów o niskim stopniu scalenia jest w pełni uzasadnione ekonomicznie. Punkty testowe dodawane są do urządzenia testowanego na poziomie jego projektu (ang. *Design For Testability*).

Kontrolerem systemu jest stacja robocza z dedykowaną aplikacją sterującą. Aplikacja sterująca wydaje komendy do wybranego testera podłączonego poprzez Ethernet do sieci lokalnej. Aplikacja pobiera z bazy danych niezbędne parametry do uruchamiania testu (np. wartość zadawanego napięcia), a także zapisuje wyniki testów. Zapisywanie w tzw. produkcyjnej bazie danych wszystkich kroków procedury testowej oraz ich wyników, umożliwia tworzenie raportów z pojedynczego testu, a także serii testów.

Komunikacja pomiędzy opisywanymi elementami systemu testowania odbywa się poprzez Ethernet. Dla celów diagnostycznych pozostawiono możliwość komunikacji z urządzeniem testującym poprzez interfejs RS-232. Bezpośrednio do stacji roboczej podłączony jest zewnętrzny programator mikrokontrolerów, a także drukarka kodów kreskowych – do oznaczania produkowanych urządzeń, a także ich opakowań. Budowa systemu testowego została przedstawiona na Rys. 1.



Rys. 1. Budowa systemu testowego
Fig. 1. Testing system structure

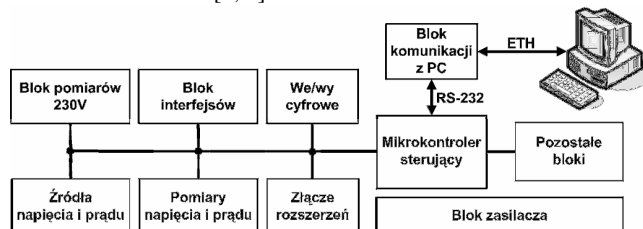
3. Budowa oraz możliwości części sprzętowej

Urządzenie testujące posiada zasoby sprzętowe, dzięki którym możliwe jest przeprowadzanie testów funkcjonalnych, uproszczonych testów funkcjonalnych oraz ICT. Zasoby funkcjonalne urządzenia testującego zawierają:

- regulowane źródła napięcia stałego,
- regulowane źródła prądu stałego typu *sink* (ang. *current sink*),
- źródła napięcia przemiennego o skokowo regulowanej wartości,
- wielowejściowy blok pomiaru napięcia stałego,
- bloki pomiaru prądu stałego pobieranego z regulowanych źródeł napięcia,
- blok komunikacji urządzenia testującego z mikrokontrolerem (-ami) testowanego urządzenia (RS-232),
- bloki interfejsowe dedykowane dla różnych urządzeń testowanych,
- wejścia-wyjścia cyfrowe mogące pracować z różnymi standardami poziomów logicznych,
- blok przyłączania sygnałów programujących z zewnętrznego programatora do mikrokontrolera (-ów) testowanego urządzenia,
- źródło napięcia sieci energetycznej (przemiennego) o skokowo regulowanej wartości z możliwością pomiaru wartości zadanego napięcia oraz pobieranie ze źródła prądu.

Pracą urządzenia testującego zarządza mikrokontroler sterujący, który komunikuje się z kontrolerem systemu poprzez interfejs ethernetowy. Do budowy bloku komunikacji poprzez Ethernet wykorzystano moduł konwertera protokołu Ethernet na RS-232 DIGI CONNECT EM [5]. Moduł DIGI został podłączony do wbudowanego w mikrokontroler sterujący sprzętowego modułu łącza szeregowego UART.

Uproszczony schemat blokowy urządzenia testującego został przedstawiony na Rys. 2, natomiast szczegółowo jego budowa została omówiona w [3, 4].

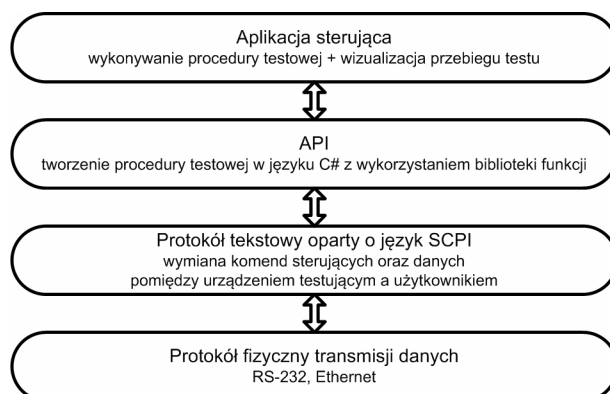


Rys. 2. Schemat blokowy części sprzętowej stanowiska testującego
Fig. 2. Block diagram of hardware part of testing station

4. Mechanizmy wymiany danych

Aplikacja sterująca uruchamiana jest na stacji roboczej z zainstalowanym środowiskiem uruchomieniowym MS .NET Framework 2.0. Jej działanie opiera się na współpracy z bazą danych MS SQL Server, w której przechowywane są m.in. konfiguracje sprzętowe testowanych urządzeń, ścieżki do oprogramowania mikrokontrolerów (testowego i docelowego), użytkownicy systemu, dostępne testy, rodzaje testów, parametry kroków testów, wyniki testów. Aplikacja pozwala na konfigurowanie systemu, a także na realizację testów.

Przebieg (logika) testów definiowany jest w dynamicznie dołączanych bibliotekach funkcji DLL, które po wybraniu typu urządzenia testowanego oraz jego docelowej konfiguracji ładowane są z wykorzystaniem mechanizmu wtyczek. Aplikacja sterująca realizuje test poprzez wywołanie metody *MainTest* wtyczki. Definiowanie logiki testu realizuje się poprzez implementację wtyczki testu w środowisku MS Visual Studio 2005, co daje możliwości oferowane przez środowisko MS .NET Framework 2.0. Językami implementacji mogą być: C#, C++, Visual Basic .NET, J#. Podczas implementacji do dyspozycji pozostają funkcje biblioteki API, która obejmuje pełną funkcjonalność testera – *TesterApi*, a także funkcje związane z pobraniem wartości parametru testu, zapisaniem wyniku testu do bazy danych, programowaniem mikrokontrolerów, drukowaniem naklejek, funkcjami związanymi z testami interaktywnymi, a także z wyświetlaniem multimediów. Funkcje *TesterApi* obejmują komunikację z testerem. Pojedyncza funkcja z biblioteki *TesterApi* realizuje wykonanie jednej lub kilku komend sterujących do testera. Użycie funkcji API jest łatwe ze względu na wbudowaną w bibliotekę pomoc w postaci mechanizmu Intellisense™.



Rys. 3. Warstwy implementacji komunikacji w aplikacji sterującej
Fig. 3. Communication implementation layers in control application

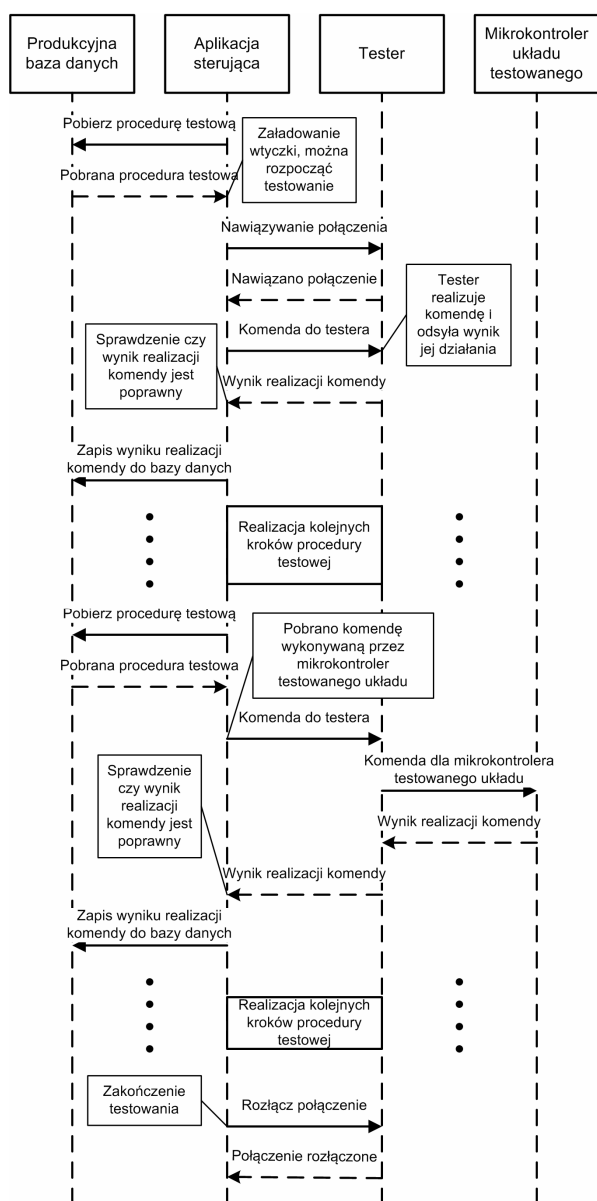
Komendy sterujące mają postać tekstową i bazują na standardzie SCPI (ang. *Standard Commands for Programmable Instruments*) [6], który został zdefiniowany dla zastosowań w urządzeniach kontrolno-pomiarowych. Komendy w łatwy sposób odzwierciedlają działanie funkcji – podzielone są na podsystemy m.in. takie jak:

- SOURce – podsystem sterowania źródłami napięć i prądów,
- READ – podsystem umożliwiający wykonywanie pomiarów,
- TRANsmit – podsystem umożliwiający wymianę komunikatów z urządzeniem testowanym.

Szczegółowy opis mechanizmu komunikacji oraz komend sterujących został zamieszczony w [6].

Komendy przesyłane są za pomocą łącza fizycznego Ethernet, choć system zakłada możliwość podłączenia poprzez łącze RS-232.

Standardowy obieg komunikatów związanych z uruchomieniem testu przedstawiony został na Rys. 4.



Rys. 4. Wielopoziomowa wymiana komunikatów w systemie testującym
Fig. 4. Multilevel commands exchange in the testing system

Parametry wszystkich kroków testów są przed testem pobierane z produkcyjnej bazy danych, zaś wyniki testów zapisywane są w niej natychmiast po ich otrzymaniu. Wyniki poszczególnych testów prezentowane są w postaci raportów, które pozwalają m.in. na porównanie wyników tego samego typu testu dla różnych testowanych urządzeń.

5. Podsumowanie

Funkcjonalność prezentowanego systemu testowania jest wystarczająca do przeprowadzania testów średnio skomplikowanych urządzeń elektronicznych i umożliwia wykrycie wielu możliwych defektów. Budowa testera umożliwia jego wydajną pracę, a także ułatwia diagnostykę jego działania.

Komunikacja pomiędzy kontrolerem systemu a urządzeniem testującym, a także bazą danych, odbywa się poprzez lokalną sieć Ethernet. Wybór Ethernetu jako interfejsu komunikacyjnego urządzenia testującego okazał się być trafny ze względu na niezawodność, skalowalność, a także dużą prędkość łącza i małe opóźnienia. Zaletę stanowi także brak przyporządkowania testera do stanowiska testowego, dzięki czemu jeden tester może być wykorzystywany przez wiele stacji roboczych z oprogramowaniem sterującym.

Wielowarstwowa architektura komunikacji z urządzeniem testującym pozwala na łatwe wydawanie komend bez użycia aplikacji sterującej – z wykorzystaniem standardowej aplikacji terminala (ze względu na zastosowanie komend tekstowych, których budowa bazuje na standardzie SCPI). Tekstowa postać komend ułatwiła także proces uruchamiania projektu części sprzętowej testera, a także implementację biblioteki funkcji testera – *TesterApi*.

Opisywany uniwersalny system testujący cechuje się bardzo dużą łatwością definiowania przebiegu testu. Implementacja testów w językach platformy MS .NET Framework 2.0 umożliwiła wykorzystanie biblioteki funkcji, co przyspiesza ich przygotowanie, a także zapewnia dużą ilość funkcji wynikających z zastosowania języka programowania wysokiego poziomu.

Graficzna aplikacja sterująca charakteryzuje się przyjaznością i dużą estetyką [6], w trakcie testu wyświetlane są multimedia ułatwiające przeprowadzenie testu, co powoduje, że użytkownikami systemu na linii produkcyjnej mogą być także osoby o małym doświadczeniu.

Opracowany system nie jest dedykowany do testów konkretnego typu urządzenia – zastosowanie macryc wieloigłowych oraz elastyczny mechanizm definiowania procedur testowych pozwala na łatwe przystosowywanie systemu do testowania różnych typów urządzeń. W ramach badań oraz wdrożenia opracowanego systemu, przeprowadzono kilka tysięcy testów kilku typów urządzeń telemetrycznych produkowanych przez firmę AIUT. Przeprowadzone próby potwierdzają przydatność praktyczną opracowanego systemu testowania.

Prace nad systemem realizowane były w ramach projektu celowego 6ZR9 2006 C/06757 zatytułowanego „Kompleksowy system wspomagający nadzór i podnoszenie jakości produkowanych urządzeń elektronicznych”. Projekt realizowany był we współpracy firmy AIUT sp. z o. o. z Gliwic oraz Instytutu Informatyki Teoretycznej i Stosowanej PAN w Gliwicach.

6. Literatura

- [1] A. Sowiński: *Automatyczne testowanie w mikroelektronice*. WKŁ, Warszawa 1991.
- [2] Agilent Technologies: materiały dotyczące systemu testowania ME-DALIST. www.agilent.com, styczeń 2009.
- [3] J. Mocha: *Uniwersalne stanowisko testujące*. Praca magisterska, Politechnika Śląska, Gliwice 2007.
- [4] J. Mocha, Z. Łaskarzewski: *Część sprzętowa uniwersalnego stanowiska testującego*. ELEKTRONIKA – konstrukcje, technologie, zastosowania, Wyd. SIGMA-NOT, 11/2008, ss.184-186.
- [5] DIGI: materiały dotyczące modułów DIGI CONNECT EM. www.digi.com, styczeń 2009.
- [6] Z. Łaskarzewski, J. Mocha: *Mechanizmy komunikacji z uniwersalnym urządzeniem testującym*. ELEKTRONIKA – konstrukcje, technologie, zastosowania, Wyd. SIGMA-NOT, 11/2008, ss.210-212.