

Dennis SLAWIK*, Krzysztof TOMCZEWSKI*

STEROWNIK MIKROPROGRAMOWALNY NA BAZIE UKŁADU FPGA

W artykule przedstawiono koncepcję implementacji struktury sterownika programowalnego PLC w układzie FPGA. W ramach projektu opracowano centralną jednostkę sterującą, moduły wejść i wyjść binarnych, moduły wejść analogowych oraz interfejsy komunikacyjne. Opracowane moduły umożliwiają tworzenie różnych konfiguracji wejść-wyjść sterownika. W układzie FPGA zaimplementowano strukturę sprzętową, realizującą cykl pracy sterownika programowalnego PLC. Utworzone zostały również przykładowe moduły biblioteki użytkownika, umożliwiające tworzenie własnych programów. Biblioteki opracowano w języku VHDL. W układzie zaimplementowano interfejs UART umożliwiający komunikację z komputerem PC. Opracowany program komputerowy umożliwia ustawianie oraz monitorowanie stanów wejść i wyjść sterownika.

SŁOWA KLUCZOWE: układ programowalny, FPGA, sterownik PLC, VHDL.

1. WSTĘP

Cykl pracy sterownika PLC jest ustalany na etapie produkcji sprzętu i składa się z czterech podstawowych etapów. W pierwszym etapie następuje odczyt stanu sygnałów wejściowych, przy czym istotne jest, aby nastąpił on jednocześnie na wszystkich wejściach. Zapobiega to niebezpieczeństwu pojawienia się błędów związanych ze zmianą stanów wejść w trakcie ich odczytu i wprowadzenia do pamięci sterownika błędnych kombinacji.

W drugim etapie, w oparciu o wartości zapisane w rejestrze wejściowym i program zdefiniowany przez użytkownika ustalone są i kolejno zapisywane do bufora wyjściowego stany poszczególnych wyjść sterownika. W trakcie tego etapu stany fizycznych wyjść sterownika odpowiadają konfiguracji ustalonej w poprzednim cyklu sterowania. Dopiero po wykonaniu całego programu użytkownika i ustaleniu nowych stanów wszystkich wyjść, sterownik przechodzi do realizacji trzeciego etapu, w którym następuje jednoczesne przepisanie stanu bufora wyjściowego do rejestrów wyjściowych, sterujących elementami wykonawczymi. Następnie sterownik przechodzi do ostatniego etapu obejmującego realizację funkcji diagnostycznych i systemowych. Diagnostyka polega na porównaniu stanu fizycznych wyjść sterownika ze stanem bufora wyjściowego.

* Politechnika Opolska

W przypadku wystąpienia jakichkolwiek różnic sygnalizowane jest uszkodzenie sterownika i następuje wyłączenie wyjść. Funkcje systemowe obejmują m. in. komunikację z innymi jednostkami. Podczas czwartego etapu wysyłane są pakiety danych do innych modułów systemu sterowania. W przypadku współpracy sterownika z podłączonymi poprzez interfejsy szeregowo modułami rozszerzeń lub innymi sterownikami, otrzymywane od nich dane mogą być używane zgodnie z algorytmem zdefiniowanym w programie użytkownika do wyznaczenia wartości zmiennych wyjściowych w kolejnym cyklu sterowania [7].

Czas realizacji cyklu sterowania jest najczęściej przyjmowany stały i definiowany przez użytkownika z uwzględnieniem ograniczeń sprzętowych. Często zależy on jednak od złożoności programu użytkownika. Jeżeli program nie zostanie zakończony w założonym przez użytkownika czasie, wówczas następuje wydłużenie czasu trwania cyklu sterowania, do chwili zakończenia wszystkich etapów. Często producenci sterowników umożliwiają również ich pracę z maksymalną możliwą częstotliwością, zależną tylko od złożoności programu użytkownika. W tym trybie odczyt stanu wejść następuje bezpośrednio po zakończeniu realizacji funkcji diagnostycznych i systemowych.

Sterowniki PLC mogą posiadać również wejścia i wyjścia szybkie, które działają niezależnie od podstawowego cyklu pracy sterownika. Do wejść takich doprowadza się sygnały wymagające natychmiastowej reakcji sterownika. Wyjścia szybkie służą do sterowania szybkimi procesami, np. do generowania sygnałów PWM. Liczba wejść i wyjść szybkich w sterownikach PLC jest zazwyczaj bardzo ograniczona [3, 4, 5].

W sterownikach PLC stosuje się wyjścia przekaźnikowe lub tranzystorowe, przy czym wyjścia szybkie są wyjściami tranzystorowymi. Wyjścia przekaźnikowe są wyjściami wolniejszymi, ale zapewniają separację galwaniczną sterownika od układów wykonawczych. Z kolei wyjścia tranzystorowe są wyjściami szybszymi, ale pracują odnośnie wspólnego potencjału masy sterownika. Konieczne jest więc często stosowanie dodatkowej separacji galwanicznej w przypadku, gdy obciążenia pracują na różnych potencjałach.

Wszystkie operacje w sterownikach PLC realizuje mikrokontroler. Z tego względu wszystkie czynności realizowane są kolejno. Alternatywą dla mikrokontrolera może być układ programowalny z wyspecjalizowaną strukturą sprzętową. Zaletą takiego rozwiązania jest możliwość współbieżnej realizacji wielu zadań. Umożliwia to przyspieszenie realizacji cyklu pracy sterownika, a także jego znaczne uniezależnienie od wielkości programu [1, 7, 8, 9].

Czasy realizacji cyklu pracy sterowników PLC różnią się znacznie. Sterownik S7-1500 firmy Siemens, zależnie od złożoności programu oraz konfiguracji sterownika, cykl pracy realizuje od 250 μ s do 1ms. [3] Z kolei sterownik firmy Schneider Electric M221 posiada 4 wejścia szybkie pracujące z częstotliwością do 100 kHz. Podobną maksymalną częstotliwość odczytu wejść szybkich gwa-

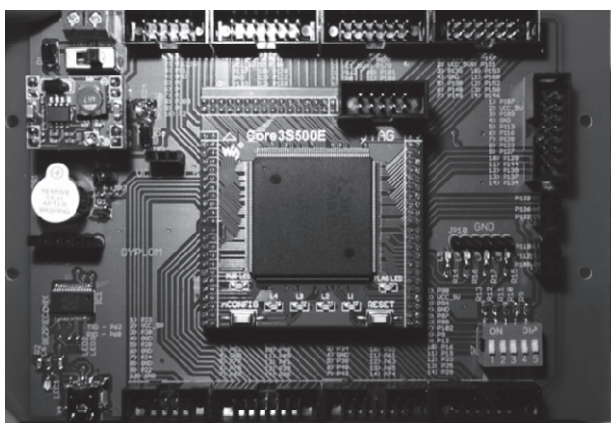
rantuje sterownik firmy Allan Bradley COMPACT 5000. [4] Cykl pracy sterownika firmy Wago CanOpen wynosi 3 ms przy obciążeniu 1000 instrukcji, przy 256 portach wejścia-wyjścia [5]. Świadczy to o tym, że sterowniki PLC realizują cykl pracy stosunkowo wolno. Czasy te są odpowiednie do sterowania większością procesów przemysłowych, ale w przypadku procesów o dużej dynamice mogą być niewystarczające.

2. STEROWNIK MIKROPROGRAMOWALNY NA BAZIE FPGA

2.1. Układ elektroniczny

Ideą realizacji sterownika mikroprogramowalnego na bazie układu FPGA jest przyspieszenie realizacji algorytmu i częściowe uniezależnienie go od złożoności programu użytkownika. Do realizacji projektu zastosowano układ Spartan 3E firmy Xilinx. Układy tej serii zawierają macierz bramek logicznych wraz z rozproszoną pamięcią oraz jednostki obliczeniowe zawierające mnożarki i sumatory, umożliwiające realizację obliczeń. Zależnie od wielkości w jednym układzie struktur tych jest od kilkunastu do kilkuset. Pozwala to na realizację wielu algorytmów obliczeniowych jednocześnie.

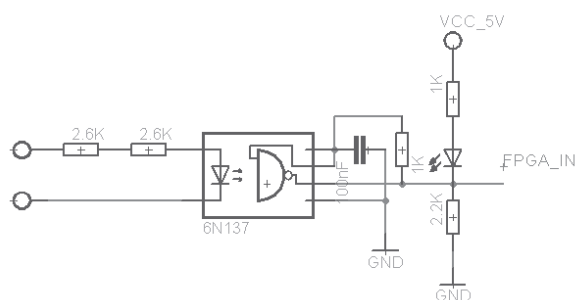
Model sterownika wykonano w postaci modułowej. Centralny moduł sterujący z układem Spartan 3E pokazano na rys. 1.



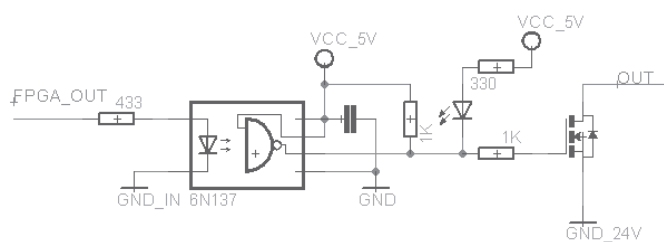
Rys. 1. Jednostka sterująca z układem Spartan 3E

Do modułu sterującego poprzez złącza IDC14 dołączane są moduły peryferyjne. Moduły te zaprojektowano w ten sposób, aby zapewniały separację galwaniczną układu sterowania od instalacji zewnętrznej. Rysunki 2 i 3 przedstawiają schematy obwodów wejściowych i wyjściowych. W wykonanych modu-

łach zgrupowano po 4 wejścia lub wyjścia binarne. Daje to dużą dowolność konfigurowania struktury sprzętowej sterownika.



Rys. 2. Schemat wejścia binarnego

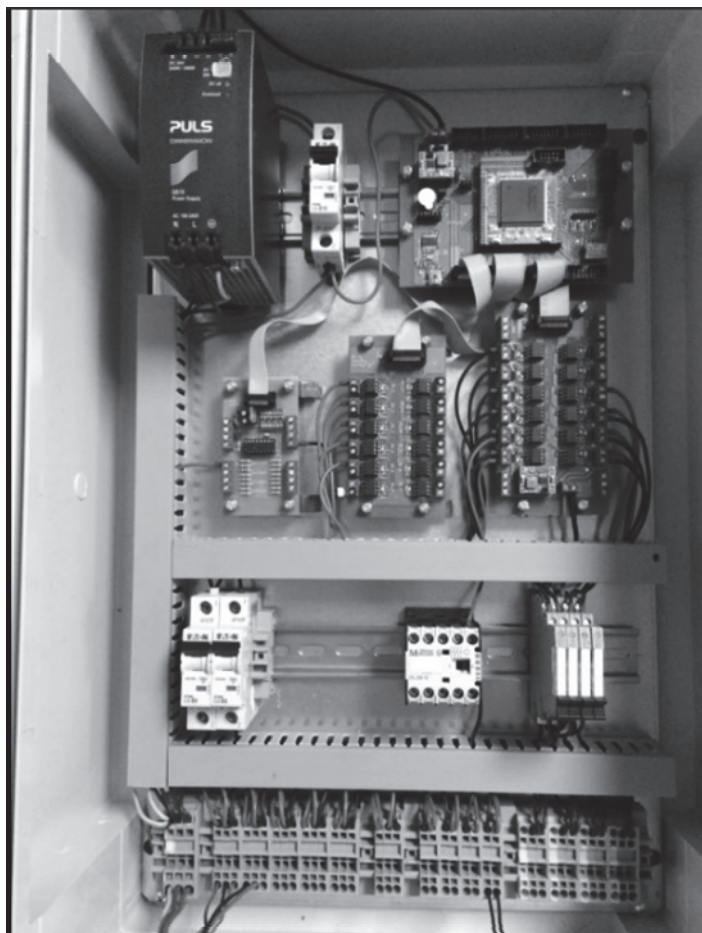


Rys. 3. Schemat wyjścia binarnego

Diagnostykę wyjść zrealizowano w ten sposób, że do wyjść binarnych dołączono wejścia, umożliwiające kontrolę fizycznych wyjść sterownika.

Dodatkowymi modułami są moduły wejść analogowych, podłączone do układu FPGA przez interfejs szeregowy SPI oraz konwerter UART - USB.

W ramach projektu wykonano sterownik pokazany na rys. 4. Jest to wersja ograniczona sprzętowo do potrzeb projektu.



Rys. 4. Sterownik mikroprogramowalny

2.2. Biblioteka konfiguracyjna i biblioteka użytkownika

Opracowana struktura wewnętrzna sterownika w układzie FPGA odzwierciedla konfigurację sprzętową. Przyjęto również, że sterownik mikroprogramowalny będzie odwzorowywała cykl pracy sterowników PLC. Z tego względu oprogramowanie układu FPGA podzielono na dwie części. Pierwsza z nich realizuje konfigurację sprzętową sterownika i cykl pracy zgodny z PLC. Drugą część stanowi biblioteka użytkownika, zawierająca elementy umożliwiające tworzenie programów w formie schematów blokowych, zbliżonych wyglądem do bloków w języku FBD. Moduły te można parametryzować stosując język VHDL. Poniżej przedstawiono fragment kodu modułu regulatora, w którym możliwa jest

parametryzacja rozdzielczości obliczeń, współczynnika skali przy obliczeniach stałoprzecinkowych, ustalanie ograniczeń minimalnych i maksymalnych wartości uchybu i sumy błędów członu całkującego. Dalsza część kodu tych modułów realizowana jest w oparciu o zadeklarowane wartości, podobnie jak pokazane poniżej deklaracje portów.

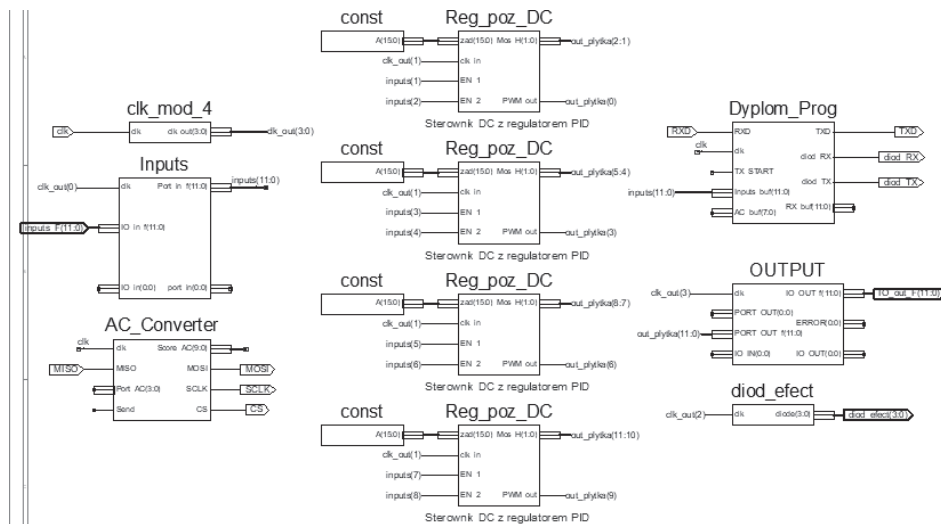
```
generic ( n : integer := 16; -- określenie n-bitowej rozdzielczości regulatora
         m : integer := 16; -- określenie m-bitowej rozdzielczość skalowania stałych regulatora
         uchyb_max : integer := 10000; -- maksymalna wartość uchybu
         uchyb_min : integer := (-10000); -- minimalna wartość uchybu
         I_max : integer := 10000; -- maksymalna wartość członu całkującego
         I_min : integer := (-10000)); -- minimalna wartość członu całkującego

Port ( clk : in STD_LOGIC;
      kier : out STD_LOGIC;
      kp_v : in std_logic_vector (m-1 downto 0);
      ki_v : in std_logic_vector (m-1 downto 0);
      kd_v : in std_logic_vector (m-1 downto 0);
      buf_z : in STD_LOGIC_VECTOR (n-1 downto 0);
      buf_rz : in STD_LOGIC_VECTOR (n-1 downto 0);
      buf_wys : out STD_LOGIC_VECTOR (n-1 downto 0));
end PI_0;
```

Użytkownik ma również możliwość tworzenia własnych modułów w języku VHDL oraz dołączania bibliotek standardowych lub własnych. Na rys. 5. pokazano strukturę sterownika mikroprogramowalnego zrealizowaną w programie graficznym środowiska ISE. Na rysunku tym zaznaczono obszar przeznaczony na program użytkownika. Wielkość okna edycji można dowolnie zmieniać. Program użytkownika powinien zostać dołączony do portów modułów biblioteki konfiguracyjnej. Opracowana biblioteka użytkownika zawiera między innymi bloki umożliwiające syntezę sterownika silnika prądu stałego. Środowisko projektowe umożliwia budowanie struktur hierarchicznych, dzięki czemu biblioteka zawiera również gotowy blok sterownika silnika prądu stałego, w postaci jednego symbolu. Zaletą układu FPGA jest możliwość wielokrotnego powielania takiego modułu w zakresie dostępnej struktury sprzętowej układu, bez istotnej zmiany prędkości realizacji algorytmu sterującego. Przykład takiego programu pokazano na rys. 6. Program umożliwia sterowanie 4 napędami z silnikami prądu stałego z prędkością zbliżoną do sterowania pojedynczym napędem [2, 6].



Rys. 5. Okno programu odzwierciedlające strukturę sprzętową sterownika



Rys. 6. Schemat układu sterowania czterema napędami z silnikami prądu stałego i regulatorami PID

Do celów porównawczych przedstawiono informacje na temat częstotliwości działania układu oraz wprowadzanych przez niego opóźnień, generowane przez środowisko ISE po zakończeniu syntezy układu z jednym i z czterema sterownikami.

Podsumowanie dla programu z 1 sterownikiem:

Minimum period: 14.911ns (Maximum Frequency: 67.065MHz)

Minimum input arrival time before clock: 21.338ns

Maximum output required time after clock: 3.597ns

Maximum combinational path delay: No path found

Podsumowanie dla programu z 4 sterownikami:

Minimum period: 16.742ns (Maximum Frequency: 59.730MHz)

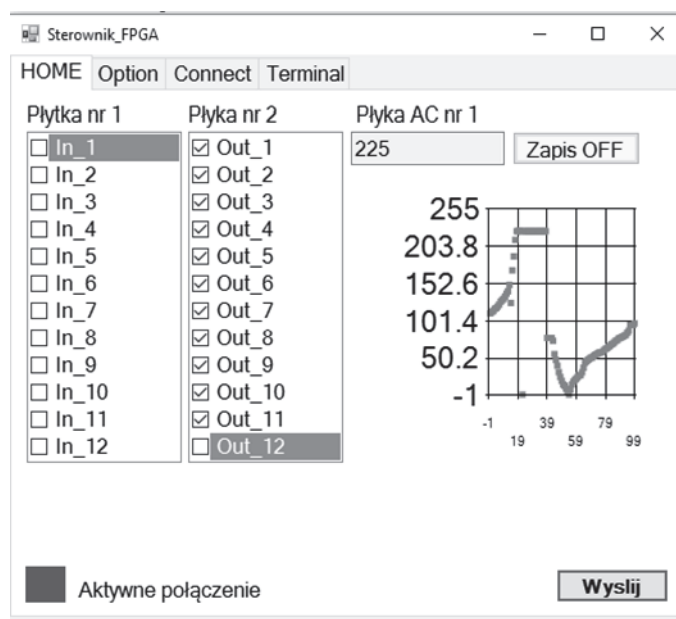
Minimum input arrival time before clock: 27.790ns

Maximum output required time after clock: 3.597ns

Maximum combinational path delay: No path found

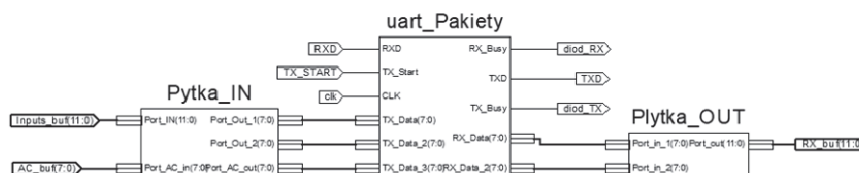
2.1. Komunikacja sterownika z komputerem

Sterowniki PLC umożliwiają łączenie ich poprzez interfejsy RS422, RS485 lub sieć Ethernet z innymi sterownikami lub komputerem w celu realizacji zdalnego sterowania i monitoringu. W ramach projektu opracowano prosty program komputerowy prezentujący możliwości komunikacji sterownika mikroprogramowalnego z komputerem. Projekt nie obejmował realizacji komunikacji z innymi sterownikami. Okno programu umożliwiającego wymuszanie stanów wyjść i monitorowanie pokazano na rys. 7.



Rys. 7. Okno programu do komunikacji ze sterownikiem

Po stronie sterownika opracowane zostały bloki realizujące interfejs UART oraz bloki łączące interfejs komunikacyjny z wejściami i wyjściami sterownika. Dzięki nim istnieje możliwość podłączenia do interfejsu USB dowolnych wejść i wyjść układu. Układ ten pokazano na rys. 8.



Rys. 8. Połączenie interfejsu UART z portami sterownika

2.4. Testy działania układu

W ramach badań układu elektronicznego wykonano pomiar opóźnienia sygnału na wyjściu fizycznym układu w stosunku do sygnału wejściowego. W tym celu w układzie FPGA zaimplementowano tylko elementy biblioteki realizującej strukturę sterownika PLC, bez programu użytkownika. Moduł wejść binarnych połączono bezpośrednio z modułem wyjść binarnych. Test polegał na pomiarze opóźnienia pomiędzy sygnałem na wejściu i na wyjściu sterownika. Pomiar wykonano analizatorem stanów logicznych. Wyniki pomiaru pokazano na rys. 8. Opóźnienie wynosi około 300 ns.

W obwodach wejść i wyjść sterownika zastosowano transoptory szybkie o czasie przełączania 75 ns, co stanowi znaczny udział w całkowitym czasie opóźnienia sygnału. Transoptory te wprowadzają połowę czasu propagacji sygnału przez sterownik.

4. PODSUMOWANIE

Opracowany układ realizuje cykl pracy sterownika PLC. Każde wejście i każde wyjście binarne może pracować z częstotliwością ponad 1 MHz, nawet przy rozbudowanych programach użytkownika. Wejścia analogowe pracują z maksymalną częstotliwością 18 kHz. Przeprowadzone testy świadczą o tym, że układ działa prawidłowo.

Opracowany układ umożliwia tworzenie różnych konfiguracji sprzętowych sterownika, dostosowanych do potrzeb realizowanego projektu. Każde wyjście sterownika może pracować jako wyjście szybkie. Dzięki sprzętowej implementacji algorytmu działania i możliwości współbieżnej realizacji wielu zadań, sterowniki oparte na układach FPGA mogą w wielu przypadkach znacznie przewyższać pod względem prędkości działania sterowniki PLC.

BIBLIOGRAFIA

- [1] Dhanashri Gawali, Sharma V. K., FPGA Based Micro-PLC Design Approach. Advances in Computing, Control, & Telecommunication Technologies, 2009.
- [2] Dinesh O. Dange, Rajan Mevekari, Dipankar D. Khartad, DC Motor Speed Control using LabVIEW FPGA Modeling, Control Algorithm Simulation & Implementation, International Journal for Scientific Research & Development| Vol. 3, Issue 05, 2015.
- [3] Dokumentacja techniczna firmy Siemens, S7-1500, ET 200SP, ET 200 pro Zyklus- und Reaktionszeiten, 09.2016.
- [4] Dokumentacja techniczna firmy Allen-Bradley, COMPACT 5000 I/O Modules and EtherNet/IP Adapters Specifications, 09.2017.
- [5] Dokumentacja techniczna firmy Wago, sterownik CANopen; 640/832 KB program/RAM; D-Sub; ekstrem, 28.01.2018.
- [6] Fares Sassi, Mehdi Abbes, Abdelkader Mami, FPGA Implementation of PID Controller, International Conference on Control, Engineering & Information Technology, Proceedings- Copyright IPCO-2014, pp.1-13.
- [7] Hajduk Z., Trybus B., Trybus L., Porównanie prototypu sterownika FPGA z klasycznymi sterownikami PLC. XVII Konferencja Automatyków Rytro 2013.
- [8] Handan Gürsoy, Mehmet Önder Efe, Control System Implementation on an FPGA Platform. IFAC-PapersOnLine, Volume 49, Issue 25, 2016, pp. 425-430.
- [9] Manish M. Patil, Shaila Subbaraman, Prashant S. Nilkund, IEC control specification to HDL synthesis: Considerations for implementing PLC on FPGA and scope for research, Control Automation and Systems (ICCAS), 2010.

FPGA BASED PROGRAMMABLE MICROCONTROLLER

The paper presents an implementation of a programmable logic controller in a FPGA based system. During the project a central control unit, digital and analog I/O modules, as well as communication interfaces module were designed. The modules allow for variable I/O configurations of the controller. The FPGA system implements a hardware structure, which performs the PLC work cycle. Additionally, a library in VHDL programming language was created allowing a user to create own programs. The system also implements the UART interface allowing communication with a PC. A sample computer program allows changing and monitoring of the controllers input and output ports.

(Received: 06.02.2018, revised: 08.03.2018)