

Mechatroniczne systemy wykonawcze, sensoryczne i sterujące

Wprowadzenie do układów FPGA

dr inż. Grzegorz Góra

D3-4.02

ggora@agh.edu.pl

<http://home.agh.edu.pl/~ggora/>

Katedra Robotyki i Mechatroniki

Wydział Inżynierii Mechanicznej i Robotyki

Akademia Górniczo-Hutnicza im. Stanisława Staszica w Krakowie

PLAN WYKŁADU

Część I

- 1. Zagadnienia przedmiotu oraz warunki zaliczenia***
- 2. Układy FPGA (budowa układu, zalety i wady, porównanie MCU i FPGA).***
- 3. Sposoby implementacji w programie Quartus Prime (Intel FPGA).***
- 4. Zastosowanie układów FPGA***
- 5. Układ edukacyjno-rozwojowy DE10-Lite.***
- 6. Przykładowe projekty realizowane z wykorzystaniem FPGA.***

Część II – Układy kombinacyjne

ZAGADNIENIA PRZEDMIOTU MSWSiS

Napędy

- silniki prądu stałego (DC);
- silniki krokowe (skokowe);
- napędy liniowe;
- napędy bezpośrednie (BLDC, PMSM);
- silniki synchroniczne AC;
- silniki asynchroniczne AC;
- silniki komutatorowe AC;
- przetworniki piezoelektryczne;
- sterowniki i stopnie mocy;
- napędy pneumatyczne i hydrauliczne.

Elektronika (technika cyfrowa)

- bramki i przerzutniki;
- projektowanie układów kombinacyjnych;
- projektowanie układów sekwencyjnych;
- mikroprocesory (budowa i architektura);
- implementacja w układach FPGA;
- soft-procesor Nios II.

Układy sensoryczne

- czujniki wielkości nieelektrycznych (klasyczne, MEMS);
- czujniki w systemach mechatronicznych;
- czujniki jako systemy mechatroniczne.

WARUNKI ZALICZENIA

Zaliczenie przedmiotu – ocena pozytywna z zajęć laboratoryjnych, projektowych i egzaminu:

zajęcia laboratoryjne (max. 60 punktów)

+

zajęcia projektowe (max. 40 punktów)

+

egzamin (napędy, elektronika, sensory).

Zaliczenie zajęć laboratoryjnych i projektowych - na podstawie ilości zdobytych punktów:

- (0%-49%) ocena dostateczna: 2.0
- (50%-60%) ocena dostateczna: 3.0
- (61%-70%) ocena ponad dostateczna: 3.5
- (71%-80%) ocena dobra: 4.0
- (81%-90%) ponad dobra: 4.5
- (91%-100%) ocena bardzo dobra: 5.0

Aktualnie zdobyte punkty (strona dr inż. Grzegorza Karpiela, zakładka Dydaktyka):

<http://home.agh.edu.pl/~gkarpiel/studenci.html>

Korekta punktowa w przypadku usprawiedliwionej nieobecności.

Ocena końcowa:

$$O_K = 0.35 \cdot O_{Lab} + 0.25 \cdot O_{Pro} + 0.4 \cdot O_{Egz}$$

O_{Lab} , O_{Pro} – oceny końcowe z zajęć laboratoryjnych i projektowych;

O_{Egz} – średnia arytmetyczna egzaminów z bloków: napędy, elektronika, sensory.

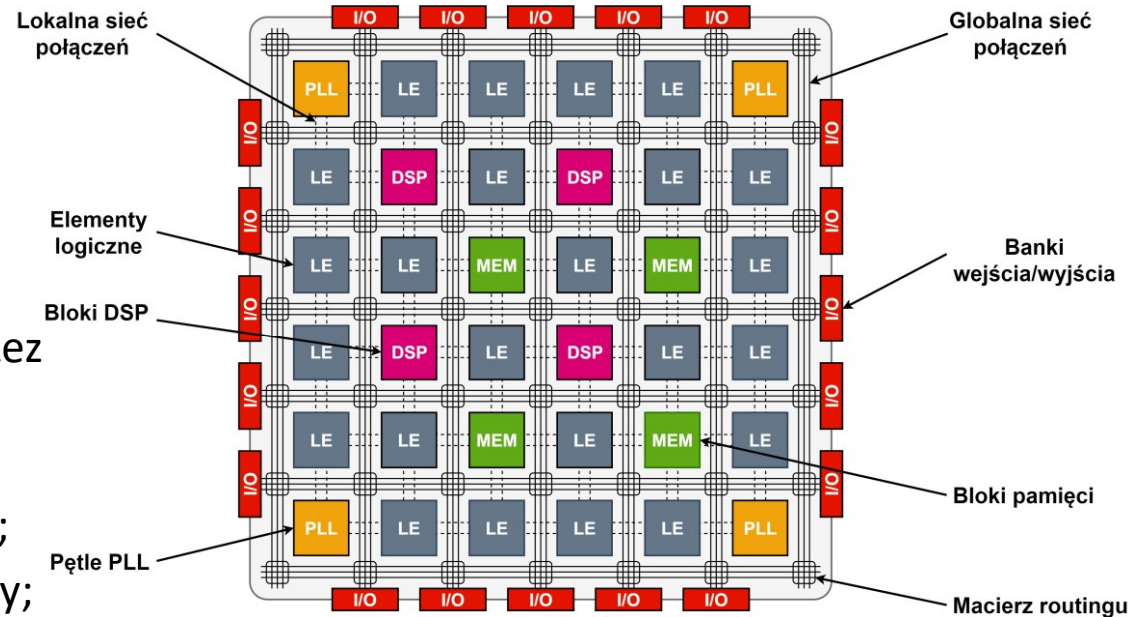
UKŁADY FPGA (ang. Field Programmable Gate Arrays)

Budowa układu FPGA:

- komórki LEs (ang. Logic Elements) / **ALM** (Adaptive Logic Modules) / **CLBs** (Configurable Logic Blocks);
- sieć połączeń – zapewnia komunikację lokalną z sąsiednimi LEs oraz komunikację w obrębie całego układu FPGA (tzw. połączenia globalne o różnym zasięgu);
- elementy dodatkowe (np. pamięć, mnożarki, pętle PLL, ADCs, itd.);
- inne (układ zasilania, porty wejścia/wyjścia).

Różnice pomiędzy układami FPGA (rodzinami):

- ilość elementów logicznych (ilość LEs);
- poziom złożoności LEs;
- szybkość układu (maksymalna częstotliwość pracy, wyrażona jako czas propagacji sygnału przez LEs od 1-9 ns);
- ilość i rodzaj wbudowanych elementów dodatkowych (pamięć, mnożarki, pętle PLL, itd.);
- ilość portów wejścia/wyjścia – rodzaj obudowy;
- wersja (commercial / industrial - zależna np. od zakresu temperatury pracy).

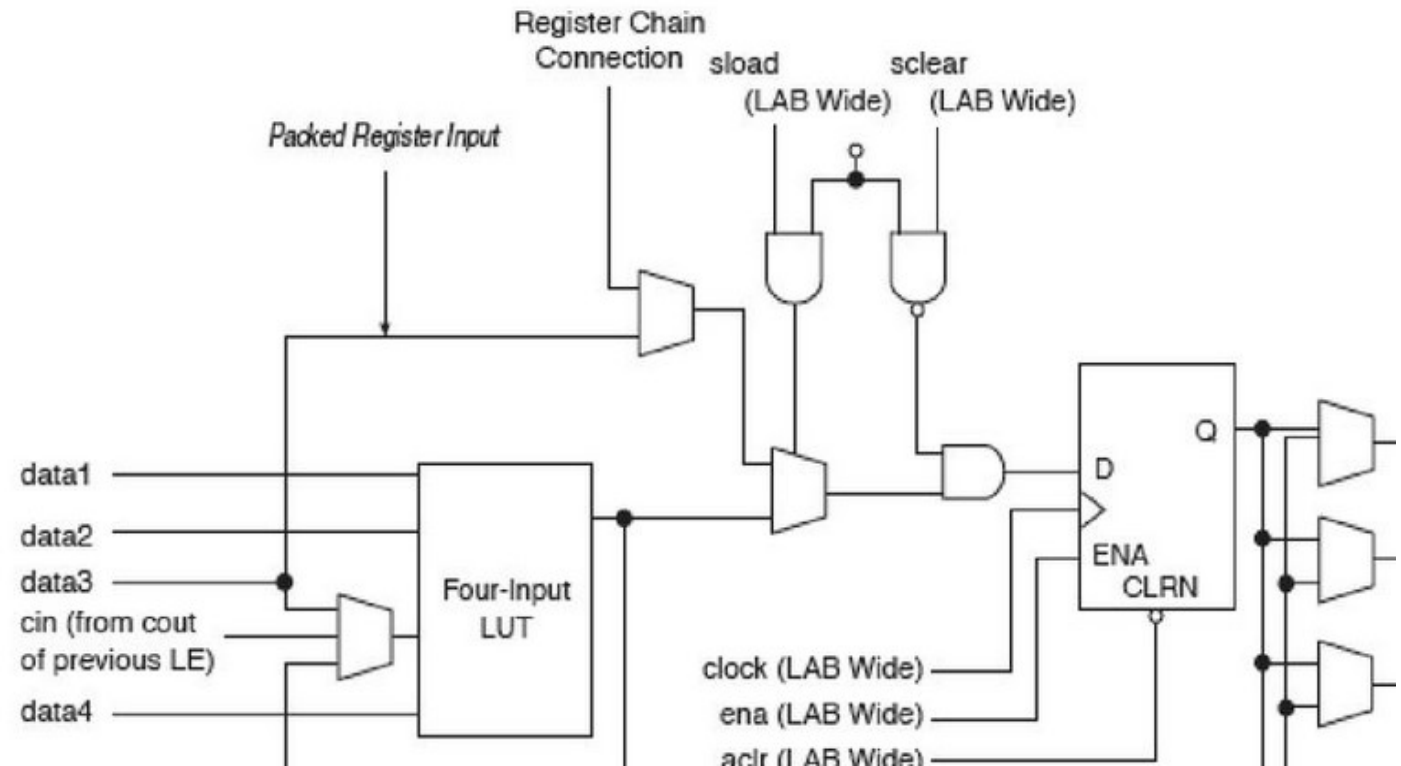


UKŁADY FPGA (ang. Field Programmable Gate Arrays)

Altera Cyclone III Logic Element (LE) – Normal Mode

Komórka LEs:

- bazuje na 4 bitowej tablicy Look Up Table (4 bitowa LUT pozwala na zaimplantowanie dowolnej 4 bitowej funkcji logicznej);
- przerzutnik pozwala na zatrzymanie sygnału (budowę układu sekwencyjnego);
- elementy dodatkowe do wewnętrznej konfiguracji i multipleksowania sygnału.



FPGA vs MCU: Podobieństwa i różnice

MCU



FPGA



Różnice

- Stała architektura narzucona przez producenta
- Język programowania wykorzystywany jako sekwencyjnie wykonywane polecenia
- Wykonywanie wielu zadań realizowane poprzez przełączanie się pomiędzy nimi
- Porty dedykowane, niewielka możliwość zmiany funkcjonalności : np. PWM, I2C, SPI
- Wbudowane przetworniki ADC i DAC

- Zmienna architektura konfigurowana przez użytkownika
- Język programowania wykorzystywany do wewnętrznej konfigurowania układu
- Zadania wykonywane jednocześnie i równolegle
- Porty ogólnego przeznaczenia dowolnie konfigurowane
- Najczęściej bez wbudowanych przetworników ADC i DAC

Podobieństwa

- Układ cyfrowy w formie układu scalonego
- Przeznaczony do realizacja zaimplementowanego algorytmu
- Programowany z wykorzystaniem dedykowanego języka

FPGA vs MCU: Podobieństwa i różnice

MCU



FPGA



Podobieństwa

- **Układ cyfrowy w formie układu scalonego**
standard napięciowy portów wejścia/wyjścia (GPIO) zazwyczaj +3V3.
- **Przeznaczony do realizacja zaimplementowanego algorytmu**
możliwość realizacji algorytmów: implementacja układów regulacji i sterowania, implementacja obliczeń numerycznych, obsługa czujników i przetworników o standardowych interfejsach (SPI, I2C, 1-Wire), obsługa urządzeń i układów peryferyjnych (np. wyświetlacze, stopnie mocy do napędów) komunikacja poprzez standardowe interfejsy (UART, Ethernet), itd.
- **Programowany z wykorzystaniem dedykowanego języka**
C / C++ / assembler – w przypadku mikrokontrolerów;
VHDL / (System) Verilog – w przypadku FPGA.

FPGA vs MCU: Podobieństwa i różnice

MCU



FPGA



Różnice

- Stała architektura narzucona przez producenta
- Język programowania wykorzystywany jako sekwencyjnie wykonywane polecenia
- Wykonywanie wielu zadań realizowane poprzez przełączanie się pomiędzy nimi
- Porty dedykowane, niewielka możliwość zmiany funkcjonalności : np. PWM, I2C, SPI
- Wbudowane przetworniki ADC i DAC

- Zmienna architektura konfigurowana przez użytkownika
- Język programowania wykorzystywany do wewnętrznej konfigurowania układu
- Zadania wykonywane jednocześnie i równolegle
- Porty ogólnego przeznaczenia dowolnie konfigurowane
- Najczęściej bez wbudowanych przetworników ADC i DAC

Podobieństwa

- Układ cyfrowy w formie układu scalonego
- Przeznaczony do realizacja zaimplementowanego algorytmu
- Programowany z wykorzystaniem dedykowanego języka

FPGA vs MCU: Podobieństwa i różnice

MCU

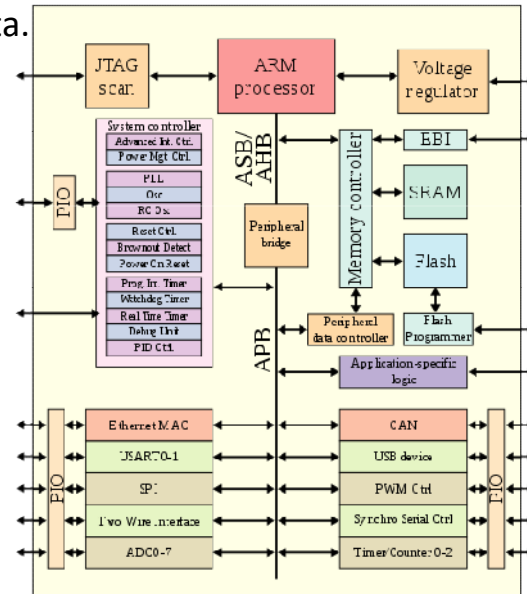


FPGA



Stała architektura narzucona przez producenta

- + gotowe moduły/układy peryferyjne, które wystarczy skonfigurować do działania;
- ograniczenie do elementów wbudowanych przez producenta.

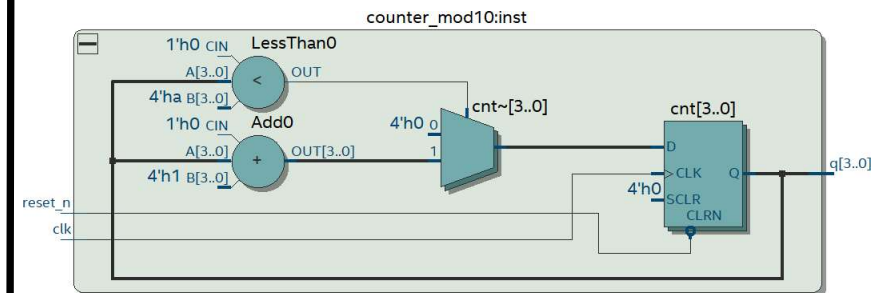


Zmienna architektura konfigurowana przez użytkownika

- + możliwość implementacji dowolnego modułu;
- zazwyczaj dłuższy czas implementacji.

Architektura fizyczna: LEs, sieć połączeń, itd.

Architektura sprzętowa: zdefiniowana przez użytkownika, np. licznik mod 10:



Różnice

FPGA vs MCU: Podobieństwa i różnice

MCU



FPGA



Różnice

- Stała architektura narzucona przez producenta
- **Język programowania wykorzystywany jako sekwencyjnie wykonywane polecenia**
- Wykonywanie wielu zadań realizowane poprzez przełączanie się pomiędzy nimi
- Porty dedykowane, niewielka możliwość zmiany funkcjonalności : np. PWM, I2C, SPI
- Wbudowane przetworniki ADC i DAC

- Zmienna architektura konfigurowana przez użytkownika
- **Język programowania wykorzystywany do wewnętrznej konfigurowania układu**
- Zadania wykonywane jednocześnie i równolegle
- Porty ogólnego przeznaczenia dowolnie konfigurowane
- Najczęściej bez wbudowanych przetworników ADC i DAC

Podobieństwa

- Układ cyfrowy w formie układu scalonego
- Przeznaczony do realizacja zaimplementowanego algorytmu
- Programowany z wykorzystaniem dedykowanego języka

FPGA vs MCU: Podobieństwa i różnice

MCU



FPGA



Język programowania wykorzystywany jako sekwencyjnie wykonywane polecenia

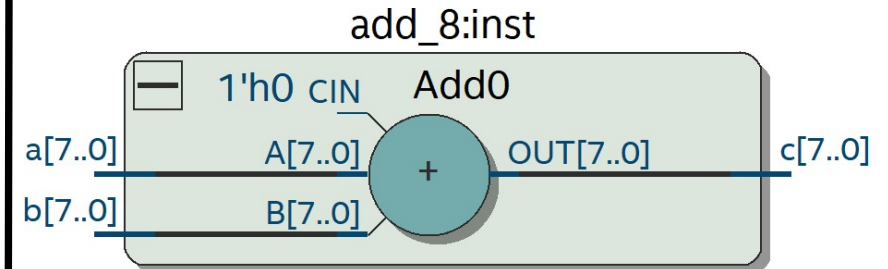
$c = a + b;$ // Język C

```

edit: C:\emu8086\MySource\Sum_Two_Array.asm
file edit bookmarks assembler emulator math ascii codes help
new open examples save compile emulate calculator convertor
01 DATA SEGMENT
02 NUM1 DB 36H,55H,27H,42H
03 NUM2 DB 38H,41H,29H,39H
04 RESULT DB 5 DUP (0)
05 ENDS
06
07 CODE SEGMENT
08 ASSUME DS:DATA CS:CODE
09 START:
10 MOV AX,DATA
11 MOV DS,AX
12
13 LEA SI,NUM1
14 LEA DI,NUM2
15 LEA BX,RESULT
16
17
18 MOV CX,5
19 LOOP1:
20 MOV AL,[SI]
21 ADD AL,[DI]
22 MOV [BX],AL
23 INC BX
24 INC SI
25 INC DI
26
27 LOOP LOOP1
28
29 MOV AH,4CH
30 INT 21H
31 ENDS
32 END START
33
34
35
line: 27 col: 33 drag a file
    
```

Język programowania wykorzystywany do wewnętrznej konfigurowania układu

$c \leq a + b;$ // Język VHDL



Różnice

FPGA vs MCU: Podobieństwa i różnice

MCU



FPGA



Różnice

- Stała architektura narzucona przez producenta
- Język programowania wykorzystywany jako sekwencyjnie wykonywane polecenia
- **Wykonywanie wielu zadań realizowane poprzez przełączanie się pomiędzy nimi**
- Porty dedykowane, niewielka możliwość zmiany funkcjonalności : np. PWM, I2C, SPI
- Wbudowane przetworniki ADC i DAC

- Zmienna architektura konfigurowana przez użytkownika
- Język programowania wykorzystywany do wewnętrznej konfigurowania układu
- **Zadania wykonywane jednocześnie i równolegle**
- Porty ogólnego przeznaczenia dowolnie konfigurowane
- Najczęściej bez wbudowanych przetworników ADC i DAC

Podobieństwa

- Układ cyfrowy w formie układu scalonego
- Przeznaczony do realizacja zaimplementowanego algorytmu
- Programowany z wykorzystaniem dedykowanego języka

FPGA vs MCU: Podobieństwa i różnice

MCU



FPGA



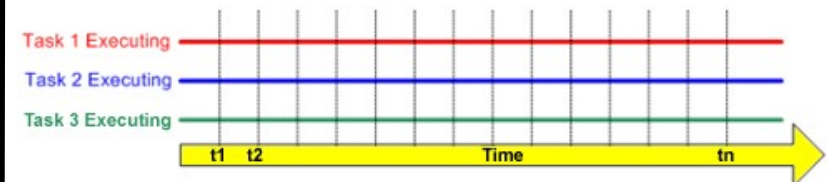
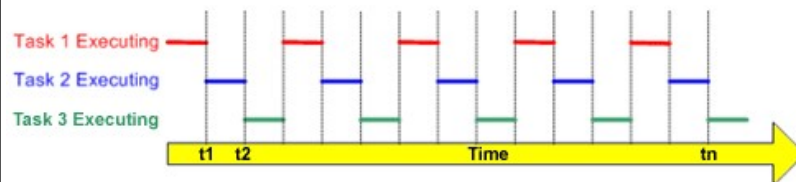
Wykonywanie wielu zadań realizowane poprzez przełączanie się pomiędzy nimi

- zwiększenie liczby obsługiwanych zadań ma wpływ realizację pozostałych;
- zazwyczaj konieczność nadawania priorytetów zadaniom;
- problemy w systemach czasu rzeczywistego z utrzymaniem stałej i wysokiej częstotliwości pracy;

Zadania wykonywane jednocześnie i równolegle

- za każde zadanie odpowiada inny „kawałek sprzętu”;
- dodanie nowych zadań nie wpływa na realizację pozostałych;
- dodanie nowych zadań powoduje wykorzystanie nowych zasobów układu FPGA, nie wpływa na szybkość realizacji już zaimplementowanych;
- programista ma możliwość „optymalizacji” pod kątem szybkości działania lub wykorzystania zasobów sprzętowych.

Różnice



FPGA vs MCU: Podobieństwa i różnice

MCU



FPGA



Różnice

- Stała architektura narzucona przez producenta
- Język programowania wykorzystywany jako sekwencyjnie wykonywane polecenia
- Wykonywanie wielu zadań realizowane poprzez przełączanie się pomiędzy nimi
- **Porty dedykowane, niewielka możliwość zmiany funkcjonalności : np. PWM, I2C, SPI**
- Wbudowane przetworniki ADC i DAC

- Zmienna architektura konfigurowana przez użytkownika
- Język programowania wykorzystywany do wewnętrznej konfigurowania układu
- Zadania wykonywane jednocześnie i równolegle
- **Porty ogólnego przeznaczenia dowolnie konfigurowane**
- Najczęściej bez wbudowanych przetworników ADC i DAC

Podobieństwa

- Układ cyfrowy w formie układu scalonego
- Przeznaczony do realizacja zaimplementowanego algorytmu
- Programowany z wykorzystaniem dedykowanego języka

FPGA vs MCU: Podobieństwa i różnice

MCU



FPGA



Różnice

- Stała architektura narzucona przez producenta
- Język programowania wykorzystywany jako sekwencyjnie wykonywane polecenia
- Wykonywanie wielu zadań realizowane poprzez przełączanie się pomiędzy nimi
- Porty dedykowane, niewielka możliwość zmiany funkcjonalności : np. PWM, I2C, SPI
- **Wbudowane przetworniki ADC i DAC**

- Zmienna architektura konfigurowana przez użytkownika
- Język programowania wykorzystywany do wewnętrznej konfigurowania układu
- Zadania wykonywane jednocześnie i równoległe
- Porty ogólnego przeznaczenia dowolnie konfigurowane
- **Najczęściej bez wbudowanych przetworników ADC i DAC**

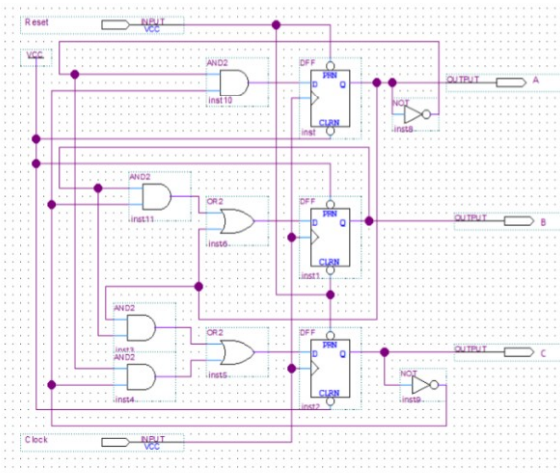
Podobieństwa

- Układ cyfrowy w formie układu scalonego
- Przeznaczony do realizacja zaimplementowanego algorytmu
- Programowany z wykorzystaniem dedykowanego języka

Quartus Prime – Sposoby implementacji

Schemat blokowy

- Podstawowe elementy logiczne (np. bramki logiczne, przerzutniki)
- Bloki sprzętowe generowane przy pomocy narzędzi pomocniczych (np. liczniki, multipleksery)
- Własne bloki sprzętowe (zaimplementowane w języku HDL)



Języki HDL

(ang. *Hardware Description Languages*)

Najczęściej używane języki:

- VHDL
- (System) Verilog
- System C

```

1 library ieee;
2 use ieee.std_logic_1164.all;
3 use ieee.numeric_std.all;
4
5 entity signed_adder is
6   port
7   (
8     aclr : in  std_logic;
9     clk  : in  std_logic;
10    a    : in  std_logic_vector;
11    b    : in  std_logic_vector;
12    q    : out std_logic_vector
13  );
14 end signed_adder;
15
16 architecture signed_adder_arch of signed_adder is
17   signal q_s : signed(a'high+1 downto 0); -- extra bit wide
18
19 begin -- architecture
20   assert(a'length >= b'length)
21     report "Port A must be the longer vector if different sizes!"
22     severity FAILURE;
23   q <= std_logic_vector(q_s);
24
25   adding_proc:
26   process (aclr, clk)
27     begin
28       if (aclr = '1') then
29         q_s <= (others => '0');
30       elsif rising_edge(clk) then
31         q_s <= ('0'&signed(a)) + ('0'&signed(b));
32       end if; -- clk'd
33     end process;
34
35 end signed_adder_arch;

```

Soft-processor

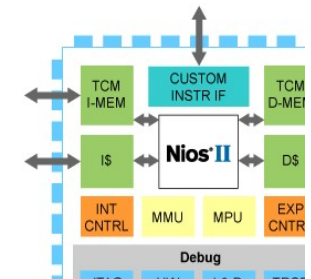
(np. *Nios II*)

Zalety:

- architektura zaprojektowana przez użytkownika;
- możliwość implementacji własnych układów peryferyjnych;
- możliwość podzielenia zadań na sprzętowe i programowe;
- możliwość implementacji kilku jednostek.

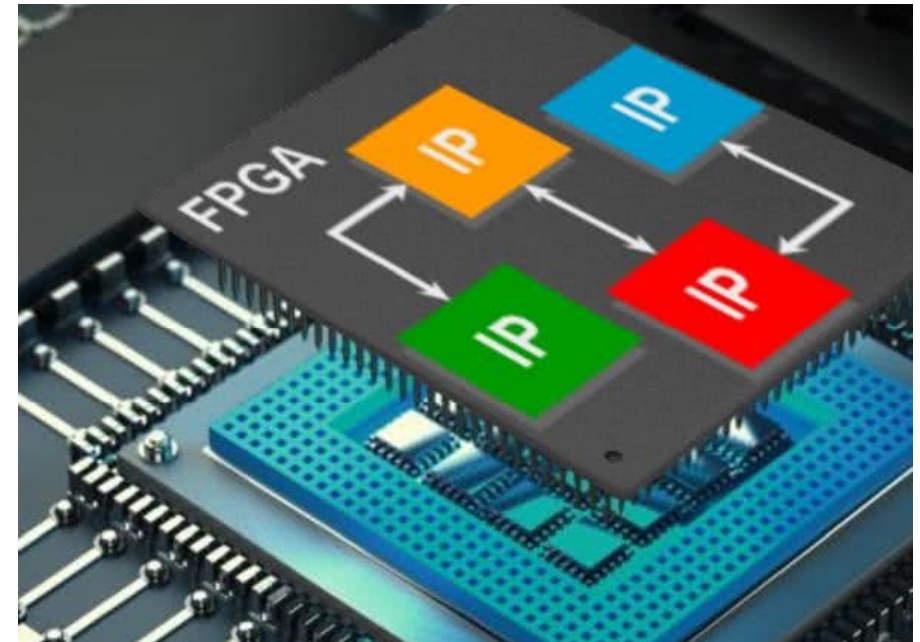
Wady:

- niższa częstotliwość pracy (taktowania);
- bardziej kosztowna platforma.



Zastosowanie układów FPGA

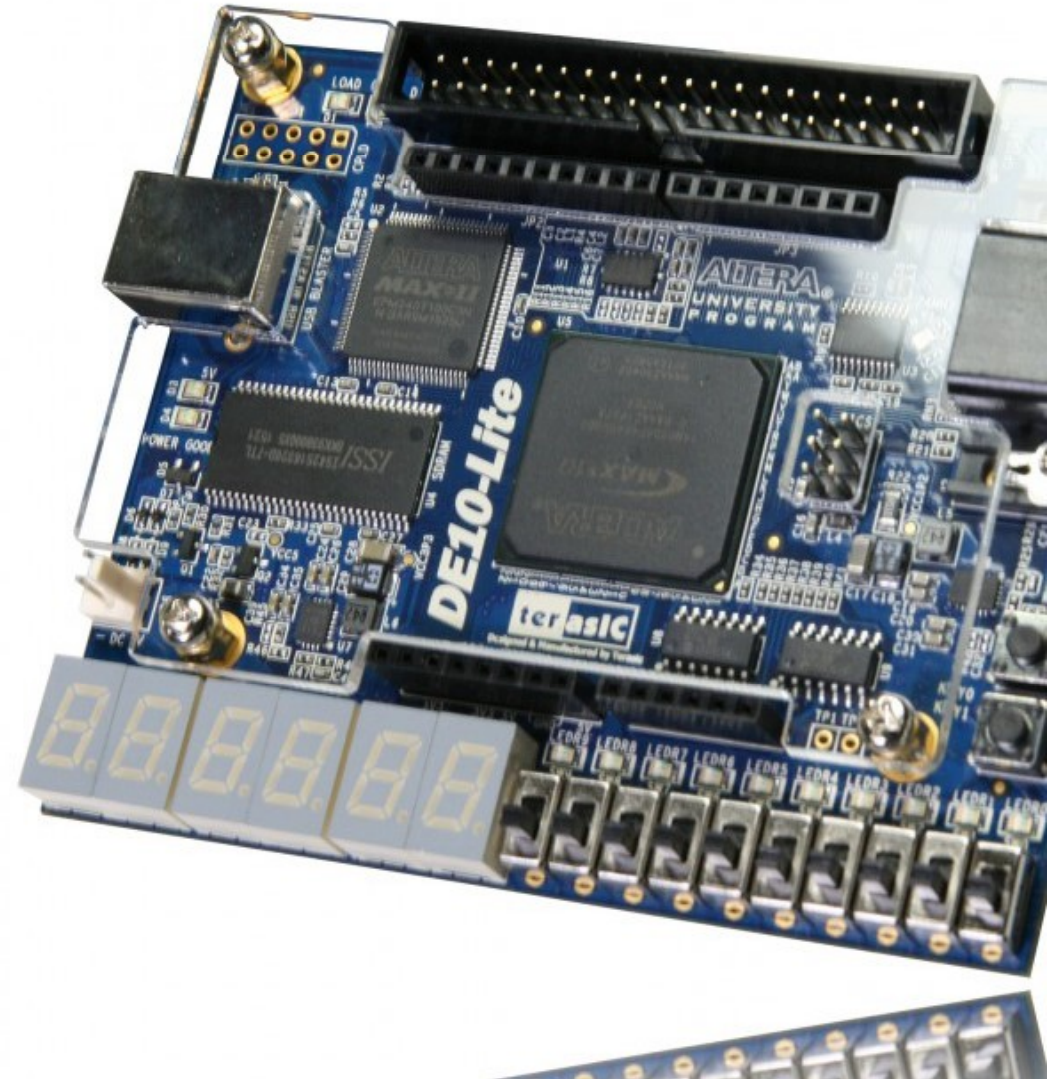
- Systemy Embedded
- Układy sterowania urządzeń mechatronicznych
- Przetwarzanie sygnałów (DSP)
- Implementacja obliczeń numerycznych
- Karty pomiarowe
- Systemy przetwarzania obrazów
- Aplikacje wojskowe
- Aplikacje sieciowe
- Power electronics
- Prototypowanie układów ASIC



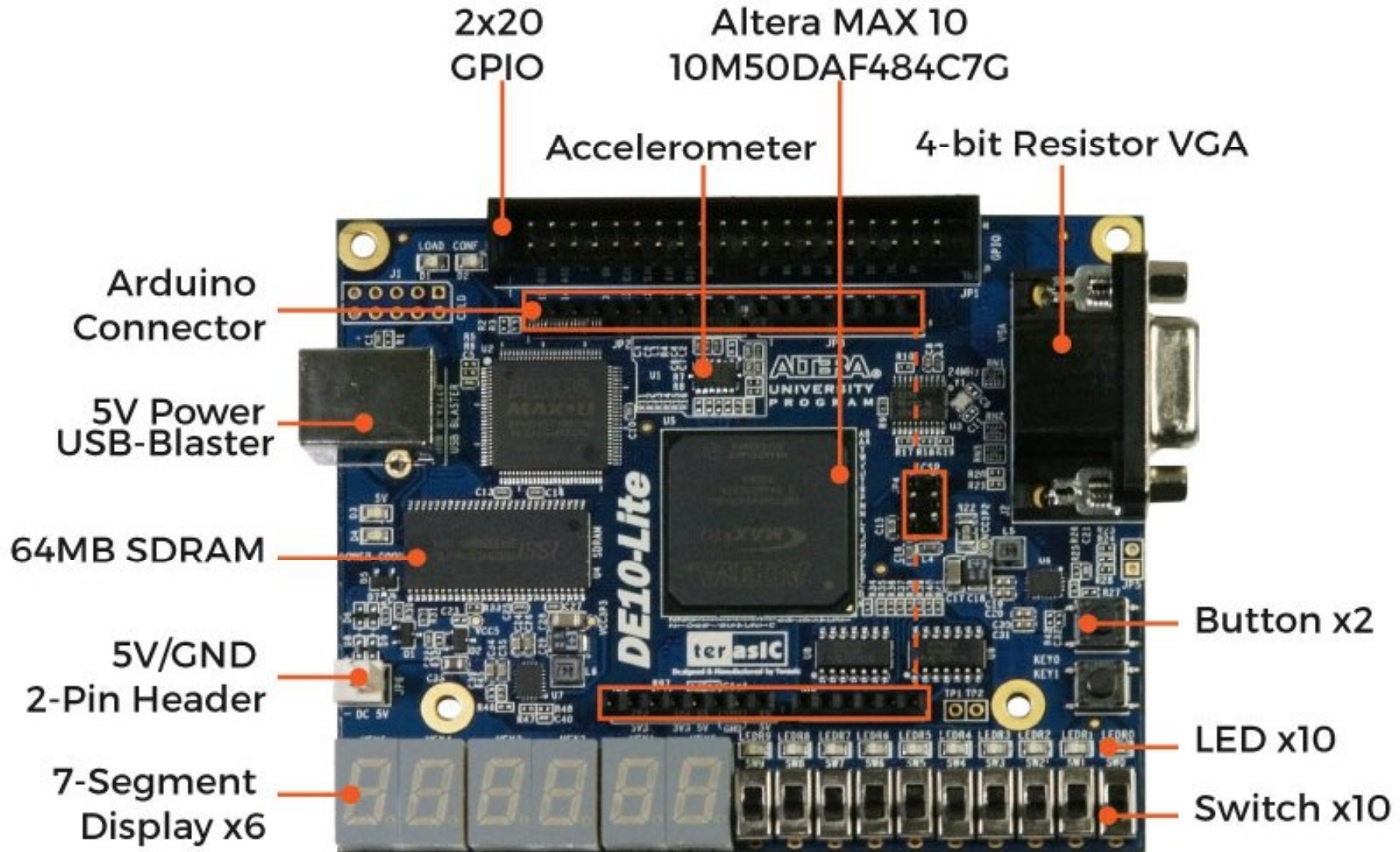
UKŁAD EDUKACYJNO-ROZWOJOWY DE10-LITE

FPGA

- MAX 10 10M50DAF484C7G
- 50 000 logicznych elementów programowalnych (LEs);
- 360 pinów GPIO (wejść/wyjść ogólnego przeznaczenia);
- 288 wbudowanych 9-bitowych mnożarek;
- 7 ns czas propagacji przez LEs (układy sekwencyjne ok. 100-120 MHz);
- 1.638 Kbit pamięci;
- 4 x PLL;
- wbudowane 2 x ADC.



UKŁAD EDUKACYJNO-ROZWOJOWY DE10-LITE



LITERATURA

- [1] <https://www.intel.com/>
- [2] <https://www.terasic.com.tw/en/>
- [3] <https://kamami.pl/>
- [4] <https://www.arm.com/>
- [5] <https://allaboutfpga.com/>

Video

- [A] https://www.youtube.com/watch?v=G-dKPe4Zz_M
- [B] <https://www.youtube.com/watch?v=CnSE-gF4jng>
- [C] <https://www.youtube.com/watch?v=IG2I56NKLQk>
- [D] https://www.youtube.com/watch?v=k_7IV0U2JhQ
- [E] https://www.youtube.com/watch?v=CXb_m8p3Yt4
- [F] <https://www.youtube.com/watch?v=ANxHyCAYGp0>
- [G] <https://www.youtube.com/watch?v=NBtGxF4UZRs>