

Systemy wbudowane - wykład 12

Przemek Błażkiewicz

30 czerwca 2018

1 / 100

Układy w systemach wbudowanych

Komponenty dyskretne: ?

2 / 100

Układy w systemach wbudowanych

Komponenty dyskretne:

- jednostki obliczeniowe (CPU);

3 / 100

Układy w systemach wbudowanych

Komponenty dyskretne:

- jednostki obliczeniowe (CPU);
- jednostki pomocnicze (ko-procesory);

4 / 100

Notes

Notes

Notes

Notes

Układy w systemach wbudowanych

- Komponenty dyskretne:
- jednostki obliczeniowe (CPU);
 - jednostki pomocnicze (ko-procesory);
 - akceleratory;

Notes

Układy w systemach wbudowanych

- Komponenty dyskretne:
- jednostki obliczeniowe (CPU);
 - jednostki pomocnicze (ko-procesory);
 - akceleratory;
 - pamięci;

Notes

Układy w systemach wbudowanych

- Komponenty dyskretne:
- jednostki obliczeniowe (CPU);
 - jednostki pomocnicze (ko-procesory);
 - akceleratory;
 - pamięci;
 - kontrolery magistrali, mostki (bridge);

Notes

Układy w systemach wbudowanych

- Komponenty dyskretne:
- jednostki obliczeniowe (CPU);
 - jednostki pomocnicze (ko-procesory);
 - akceleratory;
 - pamięci;
 - kontrolery magistrali, mostki (bridge);
 - magistrale;

Notes

Komponenty dyskretne:

- jednostki obliczeniowe (CPU);
- jednostki pomocnicze (ko-procesory);
- akceleratory;
- pamięci;
- kontrolery magistrali, mostki (bridge);
- magistrale;
- kontrolery we/wy (I/O);

9 / 100

(Główne) jednostki obliczeniowe

Notes

10 / 100

(Główne) jednostki obliczeniowe

- funkcjonalność procesora
- technologia wykonania układu procesora

11 / 100

(Główne) jednostki obliczeniowe

- funkcjonalność procesora
 - ogólnego przeznaczenia;
- technologia wykonania układu procesora

Notes

Notes

Notes

12 / 100

(Główne) jednostki obliczeniowe

- funkcjonalność procesora
 - ogólnego przeznaczenia;
 - programowalno-dedykowany;
- technologia wykonania układu procesora

13 / 100

Notes

(Główne) jednostki obliczeniowe

- funkcjonalność procesora
 - ogólnego przeznaczenia;
 - programowalno-dedykowany;
 - specjalizowany (custom, single-operation);
- technologia wykonania układu procesora

14 / 100

Notes

(Główne) jednostki obliczeniowe

- funkcjonalność procesora
 - ogólnego przeznaczenia;
 - programowalno-dedykowany;
 - specjalizowany (custom, single-operation);
- technologia wykonania układu procesora
 - CMOS

15 / 100

Notes

(Główne) jednostki obliczeniowe

- funkcjonalność procesora
 - ogólnego przeznaczenia;
 - programowalno-dedykowany;
 - specjalizowany (custom, single-operation);
- technologia wykonania układu procesora
 - CMOS
 - FPGA

16 / 100

Notes

(Główne) jednostki obliczeniowe

- funkcjonalność procesora
 - ogólnego przeznaczenia;
 - programowalno-dedykowany;
 - specjalizowany (custom, single-operation);
- technologia wykonania układu procesora
 - CMOS
 - FPGA
 - CPLD

17 / 100

Notes

(Główne) jednostki obliczeniowe

- funkcjonalność procesora
 - ogólnego przeznaczenia;
 - programowalno-dedykowany;
 - specjalizowany (custom, single-operation);
- technologia wykonania układu procesora
 - CMOS
 - FPGA
 - CPLD
 - ...

18 / 100

Notes

Procesor ogólnego przeznaczenia

Charakterystyka:

Notes

19 / 100

Procesor ogólnego przeznaczenia

Charakterystyka:

- uniwersalny procesor o szerokim zakresie typowych operacji;
- zazwyczaj architektura von Neumana (pamięć programu, pamięć danych);
- w oparciu o rejestry i dużą pamięć (zatem cache, zatem zarządca pamięci);
- potokowość, przepustowość, szeroka magistrala → szybkość działania;

Notes

20 / 100

Procesor ogólnego przeznaczenia

Charakterystyka:

- uniwersalny procesor o szerokim zakresie typowych operacji;
- zazwyczaj architektura von Neumana (pamięć programu, pamięć danych);
- w oparciu o rejestry i dużą pamięć (zatem cache, zatem zarządca pamięci);
- potokowość, przepustowość, szeroka magistrala → szybkość działania;

Zalety:

21 / 100

Notes

Procesor ogólnego przeznaczenia

Charakterystyka:

- uniwersalny procesor o szerokim zakresie typowych operacji;
- zazwyczaj architektura von Neumana (pamięć programu, pamięć danych);
- w oparciu o rejestry i dużą pamięć (zatem cache, zatem zarządca pamięci);
- potokowość, przepustowość, szeroka magistrala → szybkość działania;

Zalety:

- szybka produkcja (łatwiejsze programowanie);
- łatwa dostępność;
- wysoka elastyczność - jedna aplikacja, wiele możliwości.

22 / 100

Notes

Procesor ogólnego przeznaczenia

Charakterystyka:

- uniwersalny procesor o szerokim zakresie typowych operacji;
- zazwyczaj architektura von Neumana (pamięć programu, pamięć danych);
- w oparciu o rejestry i dużą pamięć (zatem cache, zatem zarządca pamięci);
- potokowość, przepustowość, szeroka magistrala → szybkość działania;

Zalety:

- szybka produkcja (łatwiejsze programowanie);
- łatwa dostępność;
- wysoka elastyczność - jedna aplikacja, wiele możliwości.

koszt...

23 / 100

Notes

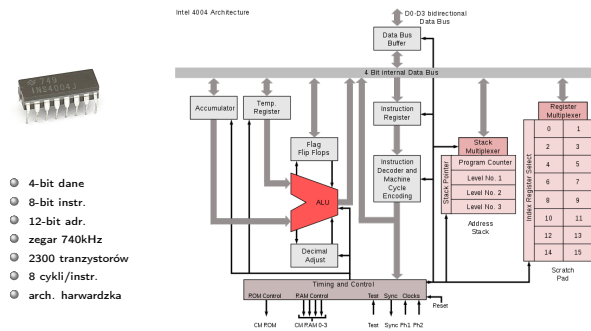
Intel 4004



- 4-bit dane
- 8-bit instr.
- 12-bit adr.
- zegar 740kHz
- 2300 tranzystorów
- 8 cykli/instr.
- arch. harwardzka

24 / 100

Notes



25 / 100

Procesor programowo-dedykowany

Pewne operacje są typowe dla pewnych dziedzin/zastosowań.

Notes

Notes

26 / 100

Procesor programowalno-dedykowany

Pewne operacje są typowe dla pewnych dziedzin/zastosowań.

Filtry to często obliczenie:

$$y = \sum_{k=0}^{M-1} A_k X_k$$

Operacja MAC \rightarrow *multiply and add.*

Notes

27 / 100

Procesor programowo-dedykowany

Pewne operacje są typowe dla pewnych dziedzin/zastosowań.

Filtry to często obliczenie:

$$y = \sum_{k=0}^{M-1} A_k X_k$$

Operacja MAC \rightarrow *multiply and add.*

Niech $X_m = b_{m(N-1)}b_{m(N-2)} \dots b_m 1 b_m 0$:

$$\begin{aligned} y &= \sum_{k=0}^{M-1} A_k \left[b_{k(N-1)} 2^{N-1} + \sum_{n=0}^{N-2} b_{kn} 2^n \right] \\ &= \sum_{k=0}^{M-1} A_k b_{k(N-1)} 2^{N-1} + \sum_{n=0}^{N-2} \left[\sum_{k=0}^{M-1} A_k b_{kn} \right] 2^n \end{aligned}$$

Notes

28 / 100

Procesor programowalno-dedykowany

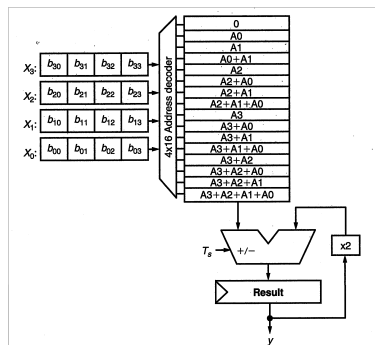
$$y = \sum_{k=0}^{M-1} A_k b_{k(N-1)} 2^{N-1} + \sum_{n=0}^{N-2} \left[\sum_{k=0}^{M-1} A_k b_{kn} \right] 2^n$$

29 / 100

Notes

Procesor programowalno-dedykowany

$$y = \sum_{k=0}^{M-1} A_k b_{k(N-1)} 2^{N-1} + \sum_{n=0}^{N-2} \left[\sum_{k=0}^{M-1} A_k b_{kn} \right] 2^n$$



30 / 100

Notes

Procesor dedykowany

Procesory dedykowane, np. DSP (*digital signal processing*).

- przyspieszone operacje (np. MAC, wektorowe) – dedykowane układy mnożące oraz operacje w asemblerze; specjalne tryby adresowania;

31 / 100

Notes

Procesor dedykowany

Procesory dedykowane, np. DSP (*digital signal processing*).

- przyspieszone operacje (np. MAC, wektorowe) – dedykowane układy mnożące oraz operacje w asemblerze; specjalne tryby adresowania;
- architektura harwardzka i potokowość;

32 / 100

Notes

Procesory dedykowane, np. DSP (*digital signal processing*).

- przyspieszone operacje (np. MAC, wektorowe) – dedykowane układy mnożące oraz operacje w asemblerze; specjalne tryby adresowania;
- architektura harwardzka i potokowość;
- trudniej programować, wolniejsze w ogólnych zastosowaniach...

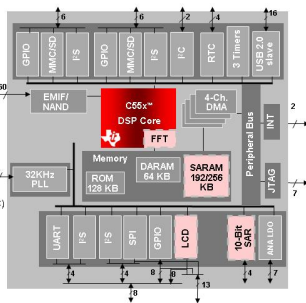
33 / 100

Texas Instruments TMS320C5504 DSP system-on-chip

TMS320C5504/5 Block Diagram and Deltas

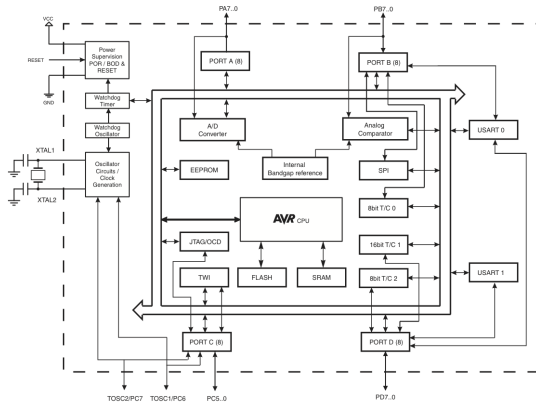
C5504 Features

- Core**
 - Dual MAC, C55x CPU with JTAG disable option
 - 1.0V @ 60MHz, 1.3V @ 100MHz (20MHz available)
 - Dynamic Voltage and Frequency Scaling
- Memory**
 - 256-KB On-Chip Memory: 64-KB DARAM, 192-KB SARAM
 - 128-KB ROM
- Peripherals**
 - Four serial busses offering combinations of PS, UART, SPI, MMC/SD, and GPIO
 - High speed USB 2.0
 - Multi-master and Slave PC with 7 or 10-bit addressing modes
 - Three 32-bit timers with watchdog functionality
 - Four 4-Channel DMAs
 - 16-bit EMIF with asynchronous SRAM, NAND (with 4-bit ECC)
 - Low power PLL (0.7mA) with 32-KHz crystal oscillator
 - Real-time clock with 32-KHz crystal input, 1 analog LDO
- I/O**
 - 1.8V, 2.5V, 2.8V, 3.3V
 - Separate VO supplies for EMIF and serial interface
- Package**: 196-pin 10x10mm BGA with 0.65mm pitch
- C5505 Additional Features**:
 - 320-KB On-Chip Memory: 64-KB DARAM, 256-KB SARAM
 - 1024-point FFT Coprocessor
 - LCD Bridge muxed with PS, SPI, UART, and GPIO
 - 4ch 10-bit SAR ADC



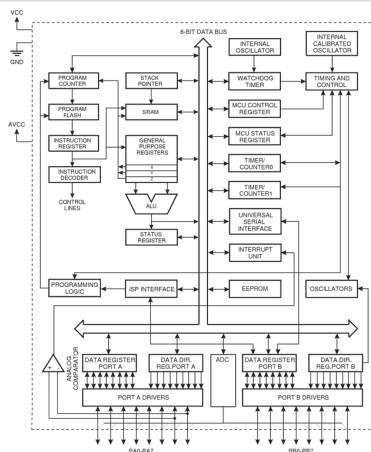
34 / 100

Mikrokontrolery AVR – ATtiny, ATmega, X-Mega...



35 / 100

Mikrokontrolery AVR – ATtiny, ATmega, X-Mega...



36 / 100

Procesor specjalizowany

Mały układ wykonujący jedną, (prostą) czynność.

Notes

Procesor specjalizowany

- Mały układ wykonujący jedną, (prostą) czynność.
- timer/watchdog/counter;

Notes

Procesor specjalizowany

- Mały układ wykonujący jedną, (prostą) czynność.
- timer/watchdog/counter;
 - UART/USART

Notes

Procesor specjalizowany

- Mały układ wykonujący jedną, (prostą) czynność.
- timer/watchdog/counter;
 - UART/USART
 - kontroler LCD, silnika krokowego...

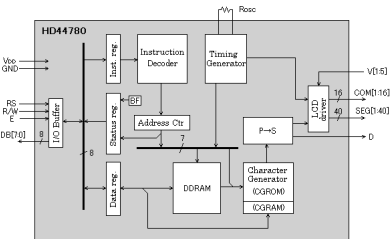
Notes

Mały układ wykonujący jedną, (prostą) czynność.

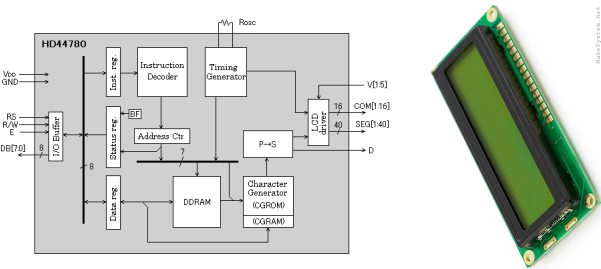
- timer/watchdog/counter;
- UART/USART
- kontroler LCD, silnika krokowego...

Gotowe procesory vs. budowane własnym sumptem.

HD44780 - kontroler LCD 2x16 znaków



HD44780 - kontroler LCD 2x16 znaków



HD44780 - kontroler LCD 2x16 znaków

Code											Description	Execution Time (max) (when t_{oc} or f_{oc} is 270 kHz)
Instruction	RS	R/W	DB7	DB6	DB5	DB4	DB3	DB2	DB1	DB0		
Clear display	0	0	0	0	0	0	0	0	0	1	Clears entire display and sets DDRAM address 0 in address counter.	
Return home	0	0	0	0	0	0	0	0	1	—	Sets DDRAM address 0 in address counter. Also returns display from being shifted to original position. DDRAM contents remain unchanged.	1.52 ms
Entry mode set	0	0	0	0	0	0	0	1	I/D	S	Sets cursor move direction and specifies display shift. These operations are performed during data write and read.	37 μ s
Display on/off control	0	0	0	0	0	0	1	D	C	B	Sets entire display (D) on/off, cursor on/off (C), and blinking of cursor position character (B).	37 μ s
Cursor or display shift	0	0	0	0	0	1	S/C	R/L	—	—	Moves cursor and shifts display without changing DDRAM contents.	37 μ s
Function set	0	0	0	0	1	DL	N	F	—	—	Sets interface data length (DL), number of display lines (N), and character font (F).	37 μ s
Set CGRAM address	0	0	0	1	ACG	ACG	ACG	ACG	ACG	ACG	Sets CGRAM address. CGRAM data is sent and received after this setting.	37 μ s
Set DDRAM address	0	0	1	ADD	ADD	ADD	ADD	ADD	ADD	ADD	Sets DDRAM address. DDRAM data is sent and received after this setting.	37 μ s

COordinate Rotation Dlgital Computer

Zestaw parametryzowanych algorytmów aproksymujących różne funkcje matematyczne (sin, cos, cosh, atan, pierwiastek...)

45 / 100

Notes

COordinate Rotation Dlgital Computer

Zestaw parametryzowanych algorytmów aproksymujących różne funkcje matematyczne (sin, cos, cosh, atan, pierwiastek...)

- operacje dodawania i przesunięcia, prosta arytmetyka

46 / 100

Notes

COordinate Rotation Dlgital Computer

Zestaw parametryzowanych algorytmów aproksymujących różne funkcje matematyczne (sin, cos, cosh, atan, pierwiastek...)

- operacje dodawania i przesunięcia, prosta arytmetyka
- efektywny dla wieloelementowych wektorów we/wy

47 / 100

Notes

CORDIC – obrót wektora (x_s, y_s) o θ

- $x_f = x_s \cos \theta - y_s \sin \theta$
 $y_f = x_s \sin \theta + y_s \cos \theta$

48 / 100

Notes

CORDIC – obrót wektora (x_s, y_s) o θ

- $x_f = x_s \cos \theta - y_s \sin \theta$
- $y_f = x_s \sin \theta + y_s \cos \theta$
- $\theta = \sum_{i=0}^{\inf} \alpha_i$

Notes

Notes

CORDIC – obrót wektora (x_s, y_s) o θ

- $x_f = x_s \cos \theta - y_s \sin \theta$
- $y_f = x_s \sin \theta + y_s \cos \theta$
- $\theta = \sum_{i=0}^{\inf} \alpha_i$
- $x_{i+1} = x_i \cos \alpha_i - y_i \sin \alpha_i$
- $y_{i+1} = y_i \sin \alpha_i + y_i \cos \alpha_i$

Notes

CORDIC – obrót wektora (x_s, y_s) o θ

- $x_f = x_s \cos \theta - y_s \sin \theta$
- $y_f = x_s \sin \theta + y_s \cos \theta$
- $\theta = \sum_{i=0}^{\inf} \alpha_i$
- $x_{i+1} = x_i \cos \alpha_i - y_i \sin \alpha_i$
- $y_{i+1} = y_i \sin \alpha_i + y_i \cos \alpha_i$
- $x_{i+1} = \cos(\alpha_i)(x_i - y_i \tan(\alpha_i))$
- $y_{i+1} = \cos(\alpha_i)(y_i + x_i \tan(\alpha_i))$

Notes

CORDIC – obrót wektora (x_s, y_s) o θ

- $x_f = x_s \cos \theta - y_s \sin \theta$
- $y_f = x_s \sin \theta + y_s \cos \theta$
- $\theta = \sum_{i=0}^{\inf} \alpha_i$
- $x_{i+1} = x_i \cos \alpha_i - y_i \sin \alpha_i$
- $y_{i+1} = y_i \sin \alpha_i + y_i \cos \alpha_i$
- $x_{i+1} = \cos(\alpha_i)(x_i - y_i \tan(\alpha_i))$
- $y_{i+1} = \cos(\alpha_i)(y_i + x_i \tan(\alpha_i))$
- $\alpha_i = \tan^{-1}(\sigma_i 2^{-i})$

CORDIC – obrót wektora (x_s, y_s) o θ

Notes

CORDIC – obrót wektora (x_s, y_s) o θ

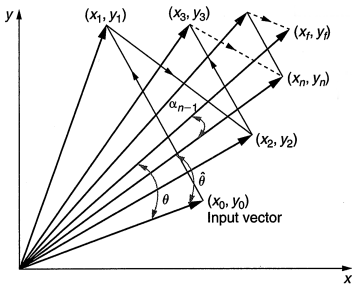
Notes

- $x_f = x_s \cos \theta - y_s \sin \theta$
- $y_f = x_s \sin \theta + y_s \cos \theta$
- $\theta = \sum_{i=0}^{\inf} \alpha_i$
- $x_{i+1} = x_i \cos \alpha_i - y_i \sin \alpha_i$
- $y_{i+1} = y_i \sin \alpha_i + x_i \cos \alpha_i$
- $x_{i+1} = \cos(\alpha_i)(x_i - y_i \tan(\alpha_i))$
- $y_{i+1} = \cos(\alpha_i)(y_i + x_i \tan(\alpha_i))$
- $\alpha_i = \tan^{-1}(\sigma_i 2^{-i})$
- $x_{i+1} = \cos(\alpha_i)(x_i - \sigma_i y_i 2^{-i})$
- $y_{i+1} = \cos(\alpha_i)(y_i + \sigma_i x_i 2^{-i})$

$$i = 0$$
$$x_0 = x_s$$
$$y_0 = y_s$$
$$z_0 = \theta$$
$$x_{i+1} = x_i - \sigma_i y_i 2^{-i}$$
$$y_{i+1} = y_i + \sigma_i x_i 2^{-i}$$
$$z_{i+1} = z_i - \sigma_i \tan^{-1}(2^{-i})$$
$$\sigma_i = \begin{cases} 1 & \text{if } z_i \geq 0 \\ -1 & \text{if } z_i < 0 \end{cases}$$

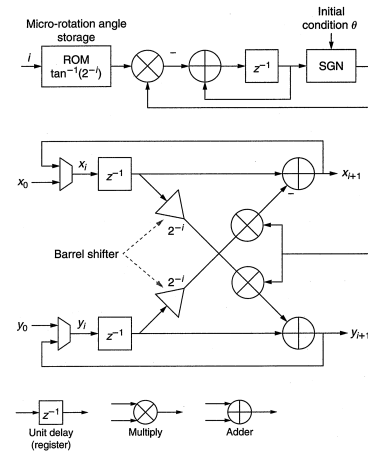
CORDIC

Notes



CORDIC

Notes

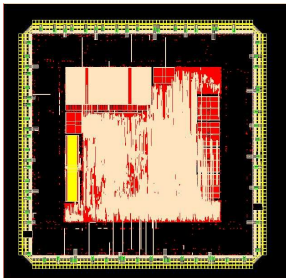


Procesory "programowalne"

– modyfikowalny hardware: FPGA/CPLD

57 / 100

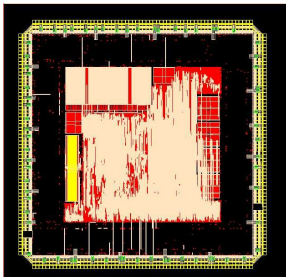
LEON2-FT - ESA i jej wkład w podbój kosmosu



Leon – a new recipe for chips - www.ESA.int

58 / 100

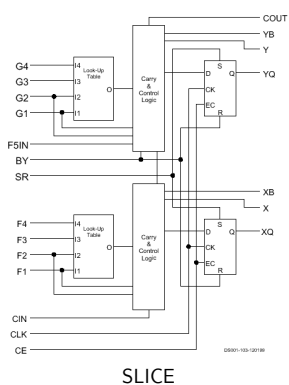
LEON2-FT - ESA i jej wkład w podbój kosmosu



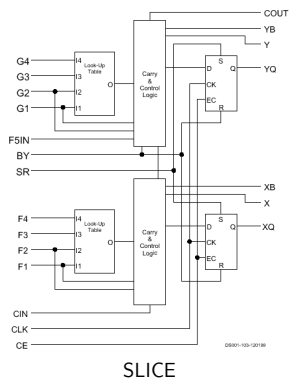
"The next stage was then to physically translate that code to create a test board hosting a programmable chip called a 'field programmable gate array' (FPGA) where the LEON design could be put to work and tested."

Leon – a new recipe for chips - www.ESA.int

59 / 100



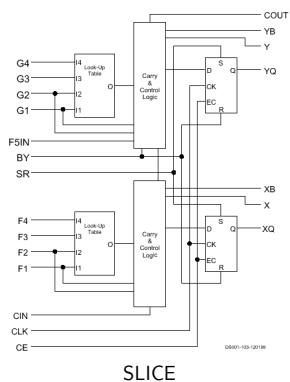
60 / 100



- wchodzą w skład CLB (configurable logic block)

Notes

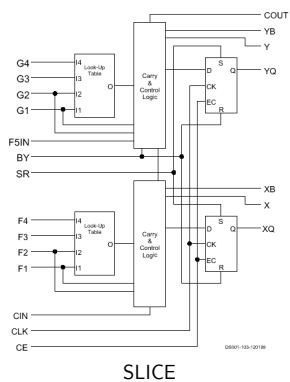
61 / 100



- wchodzą w skład CLB (configurable logic block)
- dodatkowa "drobnica" (carry chain, OR, XOR ...)

Notes

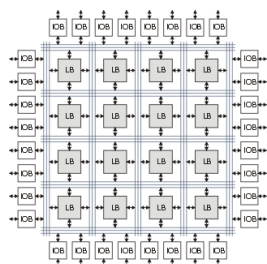
62 / 100



- wchodzą w skład CLB (configurable logic block)
- dodatkowa "drobnica" (carry chain, OR, XOR ...)
- CLB mają szybkie połączenia z sąsiadami

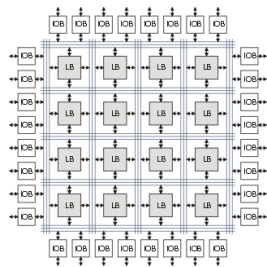
Notes

63 / 100



Notes

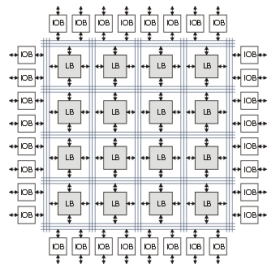
64 / 100



- łączę co dwa, co 4, co 8, wszystkie w koło

65 / 100

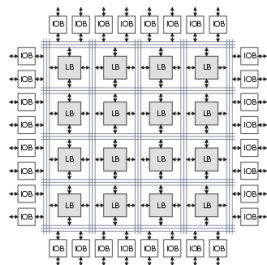
Notes



- łączę co dwa, co 4, co 8, wszystkie w koło
- układ mnożący (w Virtexie 18x18 bitów)

66 / 100

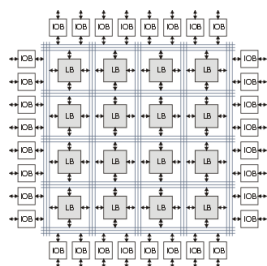
Notes



- łączę co dwa, co 4, co 8, wszystkie w koło
- układ mnożący (w Virtexie 18x18 bitów)
- rozsiane bloki pamięci RAM (po np. 18kBit)

67 / 100

Notes

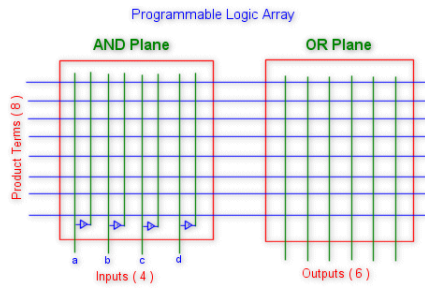


- łączę co dwa, co 4, co 8, wszystkie w koło
- układ mnożący (w Virtexie 18x18 bitów)
- rozsiane bloki pamięci RAM (po np. 18kBit)
- peryferia plus układ kompensacji zegara

68 / 100

Notes

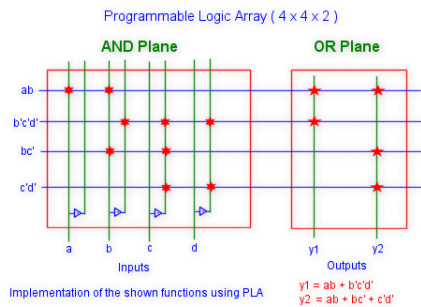
SPLD - dwie płaszczyzny, AND, OR



69 / 100

Notes

SPLD - dwie płaszczyzny, AND, OR



70 / 100

Notes

SPLD – (Simple) Programmable Logic Device

- stałe bramki, konfigurowalne połączenia

Notes

71 / 100

SPLD – (Simple) Programmable Logic Device

- stałe bramki, konfigurowalne połączenia
- FPLA, GAL – reprogramowalne

Notes

72 / 100

SPLD – (Simple) Programmable Logic Device

- stałe bramki, konfigurowalne połączenia
- FPLA, GAL – reprogramowalne
- GAL, PAL – stałe łączenia bramek OR

73 / 100

Notes

SPLD – (Simple) Programmable Logic Device

- stałe bramki, konfigurowalne połączenia
- FPLA, GAL – reprogramowalne
- GAL, PAL – stałe łączenia bramek OR
- dysjunkcyjna postać normalna

74 / 100

Notes

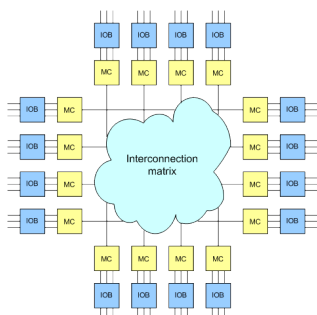
SPLD – (Simple) Programmable Logic Device

- stałe bramki, konfigurowalne połączenia
- FPLA, GAL – reprogramowalne
- GAL, PAL – stałe łączenia bramek OR
- dysjunkcyjna postać normalna
- dobre gdy iloczynny wielu we "kompresowane" do alternatywy

75 / 100

Notes

CPLD – Complex Programmable Logic Device



- LB - logic blocks (kilka)
- microcells - ~ 8-16 w LB
- MC połączone w LB
- LB połączone w CPLD

76 / 100

Notes

CPLD – po środku

PLD

- nieulotnie programowalne
- bloki we/wy bezpośrednio do LB
- dalej podstawą suma-produktów

FPGA

- zwiększona złożoność operacji
- dużo bramek
- podzespoły ogólnego przeznaczenia

77 / 100

CPLD – za/przeciw

- przewidywalne opóźnienia czasowe **krótsze niż FPGA**

78 / 100

CPLD – za/przeciw

- przewidywalne opóźnienia czasowe **krótsze niż FPGA**
 - krytyczne, "high-performance" aplikacje nadzorcze

79 / 100

CPLD – za/przeciw

- przewidywalne opóźnienia czasowe **krótsze niż FPGA**
 - krytyczne, "high-performance" aplikacje nadzorcze
- **niedrogie i oszczędne** energetycznie

80 / 100

Notes

Notes

Notes

Notes

- **przewidywalne** opóźnienia czasowe **krótsze niż FPGA**
 - krytyczne, "high-performance" aplikacje nadzorcze
- **niedrogie i oszczędne** energetycznie
 - wymagające finansowo, bateryjne systemy mobilne (sensory)

81 / 100

Notes

- **przewidywalne** opóźnienia czasowe **krótsze niż FPGA**
 - krytyczne, "high-performance" aplikacje nadzorcze
- **niedrogie i oszczędne** energetycznie
 - wymagające finansowo, bateryjne systemy mobilne (sensory)
 - proste aplikacje np. 1-z-N, dekodowanie adresów etc.

82 / 100

Notes

- **przewidywalne** opóźnienia czasowe **krótsze niż FPGA**
 - krytyczne, "high-performance" aplikacje nadzorcze
- **niedrogie i oszczędne** energetycznie
 - wymagające finansowo, bateryjne systemy mobilne (sensory)
 - proste aplikacje np. 1-z-N, dekodowanie adresów etc.
- **mniej wszechstronna** struktura

83 / 100

Notes

- **przewidywalne** opóźnienia czasowe **krótsze niż FPGA**
 - krytyczne, "high-performance" aplikacje nadzorcze
- **niedrogie i oszczędne** energetycznie
 - wymagające finansowo, bateryjne systemy mobilne (sensory)
 - proste aplikacje np. 1-z-N, dekodowanie adresów etc.
- **mniej wszechstronna** struktura
- **ograniczona** ilość wejść dla microcell

84 / 100

Notes

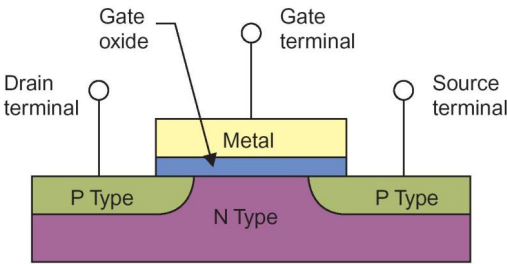
Tranzystor polowy - MOSFET

Metal-On-Silicon, Field-Effect, Transistor → MOSFET

Notes

Tranzystor polowy - MOSFET

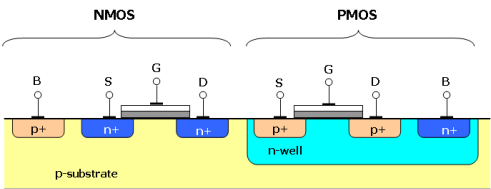
Metal-On-Silicon, Field-Effect, Transistor → MOSFET



Notes

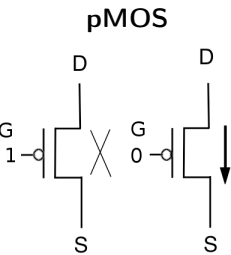
Tranzystor polowy - MOSFET

Metal-On-Silicon, Field-Effect, Transistor → MOSFET



Notes

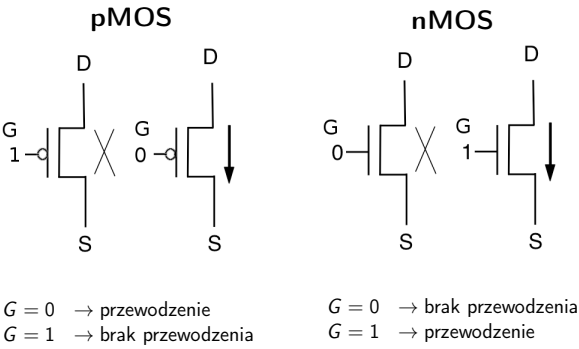
MOSFET - działanie



$G = 0 \rightarrow$ przewodzenie
 $G = 1 \rightarrow$ brak przewodzenia

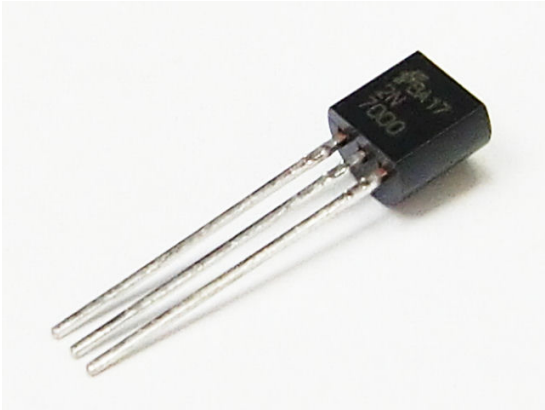
Notes

MOSFET - działanie



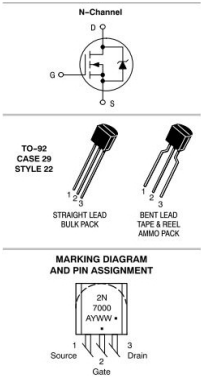
Notes

MOSFET 2N7000



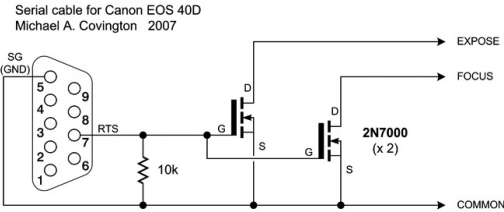
Notes

MOSFET 2N7000



Notes

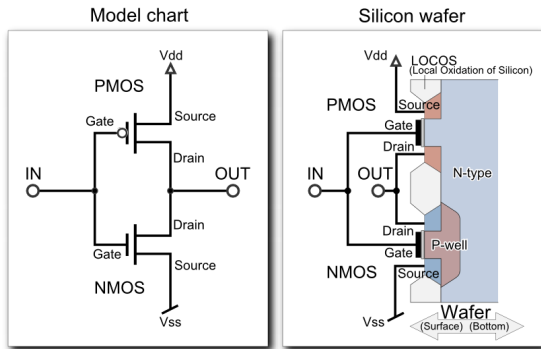
MOSFET 2N7000



Notes

Bramki logiczne CMOS - inverter

CMOS inverter



93 / 100

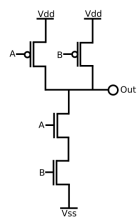
Notes

Bramki logiczne CMOS - NAND, NOR

Notes

94 / 100

Bramki logiczne CMOS - NAND, NOR

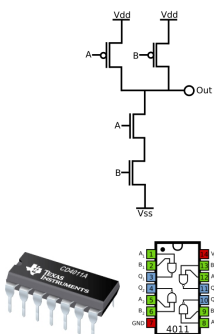


95 / 100

Notes

Bramki logiczne CMOS - NAND, NOR

Notes



96 / 100

Bramki logiczne CMOS - NAND, NOR

Notes

97 / 100

Bramki logiczne CMOS - NAND, NOR

Notes

98 / 100

Notes

99 / 100

Notes

Rzeczy do zapamiętania

- procesory specjalizowane vs. programowalne vs. programowalne-dedykowane;
- CORDIC;
- budowa, porównanie FPGA, (S/C)PLD;
- technologia CMOS: bramki IVN, NAND, NOR;

Do przeczytania

- http://www.esa.int/Our_Activities/ ⇒ LEON – a new recipe for chips.

100 / 100