

## ЛЕКЦІЯ 01

[1.1]

СУЧАСНА МІКРОПРОЦЕСОРНА ТЕХНІКА  
ЛЕКЦІЯ 1

СУДАКОВ О.О, РАДЧЕНКО С.П.  
«СУЧАСНА МІКРОПРОЦЕСОРНА ТЕХНІКА»

[1.2]

АКТУАЛЬНІСТЬ

- СЬОГОДНІ МІКРОПРОЦЕСОРИ ВИКОРИСТОВУЮТЬСЯ ДУЖЕ ШИРОКО
  - ПОБУТОВА ТЕХНІКА
  - ТРАНСПОРТ
  - ПРОМИСЛОВІСТЬ
  - НАУКА
  - ЗВ'ЯЗОК
  - МЕДИЧНА ТЕХНІКА
  - ...
- НЕОБХІДНО РОЗУМІТИ ОСНОВИ, ТЕРМІНОЛОГІЮ І ВМІТИ ВИКОРИСТОВУВАТИ МІКРОПРОЦЕСОРИ
- ФАХІВЦІ З МІКРОПРОЦЕСОРІВ - ПОТРІБНІ

[1.3]

ЗАДАЧІ КУРСУ

- **БАЗОВІ ЗНАННЯ ПРО МІКРОПРОЦЕСОРИ**
  - ТЕРМІНОЛОГІЯ
  - КЛАСИФІКАЦІЯ
- **ОСНОВИ РОБОТИ МІКРОПРОЦЕСОРІВ**
  - ПРИНЦИПИ ПОБУДОВИ
  - ХАРАКТЕРИСТИКИ
  - ТЕХНОЛОГІЇ
- **ВИКОРИСТАННЯ МІКРОПРОЦЕСОРНОЇ ТЕХНІКИ**
  - ПРОГРАМУВАННЯ МІКРОПРОЦЕСОРІВ
  - ПРАКТИЧНЕ ЗАСТОСУВАННЯ МІКРОПРОЦЕСОРІВ
  - ОЦІНКА ХАРАКТЕРИСТИК МІКРОПРОЦЕСОРНИХ СИСТЕМ

[1.4]

СТРУКТУРА КУРСУ

- **МОДУЛЬ 1 ОСНОВИ РОБОТИ МІКРОПРОЦЕСОРІВ.**
  - ЗАГАЛЬНІ ВІДОМОСТІ ПРО МІКРОПРОЦЕСОРИ
  - МІКРОКОНТРОЛЕРИ
  - ЦИФРОВІ МІКРОПРОЦЕСОРИ ЗАГАЛЬНОГО ПРИЗНАЧЕННЯ
  - КОНТРОЛЬНА**
- **МОДУЛЬ 2 МІКРОПРОЦЕСОРНІ СИСТЕМИ ТА ТЕХНОЛОГІЇ.**
  - МІКРО ЕОМ
  - СПЕЦІАЛІЗОВАНІ ЦИФРОВІ ПРОЦЕСОРНІ СИСТЕМИ
  - СПЕЦІАЛІЗОВАНІ АНАЛОГОВІ ПРОЦЕСОРИ
  - ФІЗИЧНІ ОСНОВИ ТЕХНОЛОГІЇ МІКРОПРОЦЕСОРІВ
  - КОНТРОЛЬНА**

■ ПЕРЕВІРКА САМОСТІЙНОЇ РОБОТИ

- ДЕМОНСТРАЦІЯ РОБОТИ ПРИСТРОЇВ, ВИКОНАНИХ СТУДЕНТАМИ

■ ЗАЛІК

[1.5]

САМОСТІЙНА РОБОТА

■ РОЗРОБКА ПРИСТРОЮ НА ОСНОВІ МІКРОПРОЦЕСОРА

- РОЗРОБКА ЕЛЕКТРОННОЇ СХЕМИ
- РОЗРОБКА ПРОГРАМИ
- МОДЕЛЮВАННЯ СХЕМИ І ПРОГРАМИ
- ВІДЛАДКА
- ТЕСТУВАННЯ

■ TI LAUNCHPAD ~30 ГРН [HTTP://TI.COM](http://ti.com)

■ ARDUINO 100-500 ГРН [HTTP://DX.COM](http://dx.com)

■ ПРОГРАМИ

- AVRSTUDIO – СИМУЛЯТОР AVR
- VMLAB – СИМУЛЯТОР AVR
- IAR WORKBENCH – СИМУЛЯТОР MSP430
- PROTEUS – СИМУЛЯТОР ЕЛЕКТРОННИХ СХЕМ

[1.6]

TI LAUNCHPAD [HTTPS://ESTORE.TI.COM/](https://estore.ti.com/)

- ПРОГРАМАТОР, ПЛАТФОРМА ДЛЯ НАЛАДКИ І НАВЧАННЯ MSP430

[1.7]

IAR WORKBENCH

[HTTP://WWW.TI.COM/LSDS/TI/TOOLS-SOFTWARE/DEVTOOLS.PAGE](http://www.ti.com/lstds/ti/tools-software/devtools.page)

- РОЗРОБКА ДЛЯ MSP430

[1.8]

ARDUINO [HTTP://DX.COM](http://dx.com)

- ПРОГРАМАТОР І ПЛАТФОРМА ДЛЯ НАВЧАННЯ І НАЛАДКИ AVR

[1.9]

VMLAB – [HTTP://WWW.AMCTOOLS.COM/](http://www.amctools.com/)

- СИМУЛЯТОР AVR

[1.10]

AVR STUDIO [HTTP://WWW.ATMEL.COM/TOOLS/ATMELSTUDIO.ASPX](http://www.atmel.com/tools/atmelstudio.aspx)

- РОЗРОБКА ПРОГРАМ ДЛЯ AVR

[1.11]

PROTEUS

- СИМУЛЯЦІЯ І НАЛАДКА ЕЛЕКТРОННИХ СХЕМ

[1.12]

ДОКУМЕНТАЦІЯ НА ПРОЦЕСОРИ

■ USER GUIDE – ІНСТРУКЦІЯ З ВИКОРИСТАННЯ

- [HTTP://WWW.ATMEL.COM/TOOLS/ATMELSTUDIO.ASPX?TAB=DOCUMENTS](http://www.atmel.com/tools/atmelstudio.aspx?tab=documents)

- [HTTP://WWW.TI.COM/LSDS/TI/MICROCONTROLLER/HOME.PAGE](http://www.ti.com/lstds/ti/microcontroller/home.page)

■ DATASHEET – ТЕХНІЧНІ ДАНІ

- [HTTP://WWW.TI.COM/LSDS/TI/MICROCONTROLLER/HOME.PAGE](http://www.ti.com/lstds/ti/microcontroller/home.page)

- [HTTP://WWW.ATMEL.COM/TOOLS/ATMELSTUDIO.ASPX?TAB=DOCUMENTS](http://www.atmel.com/tools/atmelstudio.aspx?tab=documents)

[1.13]

#### ВИЗНАЧЕННЯ МІКРОПРОЦЕСОРА

- **МІКРОПРОЦЕСОР** – ПРОЦЕСОР ВИКОНАНИЙ НА ІНТЕГРАЛЬНИХ МІКРОСХЕМАХ
- **ПРОЦЕСОР** - ЩОСЬ ДЛЯ КЕРУВАННЯ І ОБРОБКИ (ПРИСТРІЙ, ПРОГРАМА)
  - ЦЕНТРАЛЬНИЙ ПРОЦЕСОР КОМП'ЮТЕРА
  - ТЕКСТОВИЙ ПРОЦЕСОР (ПРОГРАМА)
  - СИГНАЛЬНИЙ ПРОЦЕСОР
- **ВУЗЬКЕ ЗНАЧЕННЯ** – ПРОГРАМНО-КЕРОВАНИЙ ПРИСТРІЙ УПРАВЛІННЯ ТА ОБРОБКИ ІНФОРМАЦІЇ
- **ПОХОДЖЕННЯ ТЕРМІНУ**
  - ПРОЦЕС – ПОСЛІДОВНІСТЬ ПОДІЙ У ЧАСІ
  - ПРОЦЕДУРА – ПОСТУП, ПРОСУВАННЯ
- **НЕ Є ПРОЦЕСОРАМИ** – ПРИСТРОЇ ДЕ НЕМАЄ ПОСТУПОВОСТІ
  - ЛОГІЧНІ СХЕМИ, ЕЛЕМЕНТИ ПАМ'ЯТІ, РАДІОПРИЙМАЧІ

[1.14]

#### ПРИКЛАД ПРОЦЕСОРНИХ СИСТЕМ

- МУЗИЧНА СКРИНЬКА
- ЕЛЕКТРОННИЙ ДЗВІНОК
- ПЕРСОНАЛЬНИЙ КОМП'ЮТЕР
- ВЕКТОРНО-КОНВЕЙЄРНИЙ СУПЕРКОМП'ЮТЕР
- АВТОСИГНАЛІЗАЦІЯ

[1.15]

#### ПЕРЕВАГИ МІКРОПРОЦЕСОРІВ

- УНІВЕРСАЛЬНІСТЬ
- ГНУЧКІСТЬ
- МАЛІ РОЗМІРИ
- ВАРТІСТЬ
- МАЛЕ ЕНЕРГОСПОЖИВАННЯ
- ПРОСТОТА РОЗРОБКИ ПРИСТРОЇВ
- НАДІЙНІСТЬ
- МОЖНА ВИКОРИСТОВУВАТИ ДЛЯ НЕТРАДИЦІЙНИХ ПОТРЕБ

[1.16]

#### КЛАСИФІКАЦІЯ МІКРОПРОЦЕСОРІВ

##### ЗА ПРИЗНАЧЕННЯМ

- ЦИФРОВІ – ДЛЯ ОБРОБКИ ДАНИХ У ЦИФРОВІЙ ФОРМІ
- АНАЛОГОВІ – ДЛЯ ОБРОБКИ ДАНИХ У АНАЛОГОВІЙ ФОРМІ
- УНІВЕРСАЛЬНІ (ЗАГАЛЬНОГО ПРИЗНАЧЕННЯ)

- СПЕЦІАЛІЗОВАНИ (ASIC)
- МІКРОКОНТРОЛЕРИ (MCU) - ЦИФРО-АНАЛОГОВІ  
МІКРОПРОЦЕСОРНІ СИСТЕМИ ДЛЯ КЕРУВАННЯ ПРИСТРОЯМИ
- МАТЕМАТИЧНІ
- СИГНАЛЬНІ (DSP, ASP)
- ДЛЯ ВБУДОВУВАНИХ СИСТЕМ (EMBEDDED)
- ДЛЯ СЕРВЕРІВ
- ГРАФІЧНІ
- АКСЕЛЕРАТОРИ
- КОМУТАЦІЙНІ

[1.17]

КЛАСИФІКАЦІЯ МІКРОПРОЦЕСОРІВ

**ЗА МІСЦЕМ В СИСТЕМІ**

- ЦЕНТРАЛЬНИЙ
- ПЕРИФЕРІЙНИЙ
- СПІВПРОЦЕСОР
- КОНТРОЛЕР (ПАМ'ЯТІ, ПЕРЕРИВАНЬ, МЕРЕЖЕВОГО АДАПТЕРА...)

[1.18]

КЛАСИФІКАЦІЯ ЦИФРОВИХ МІКРОПРОЦЕСОРІВ

- ЗА АРХІТЕКТУРОЮ
  - ПАРАЛЕЛЬНА
  - ВЕКТОРНА
  - КОНВЕСРНА
  - СКАЛЯРНА
  - СУПЕРСКАЛЯРНА
  - ГАРВАРДСЬКА
  - ФОН-НЕЙМАНІВСЬКА
  - З ДОВГИМ СЛОВОМ
- ЗА РОЗРЯДНІСТЮ
  - 4,8,16,32,64,128
- ЗА СИСТЕМОЮ КОМАНД
  - CISC, RISC

[1.19]

КЛАСИФІКАЦІЯ ЗА ФІЗИЧНИМИ ОСОБЛИВОСТЯМИ

- ТАКТОВА ЧАСТОТА
- ТЕХНОЛОГІЯ
- ЕНЕРГОСПОЖИВАННЯ
- ФІЗИЧНИЙ ПРИНЦИП

[1.20]

ІСТОРИЧНА ІНФОРМАЦІЯ

- З ДАВНОГО СВІТУ ДО НАШОГО ЧАСУ
  - ВОДЯНІ, МЕХАНІЧНІ ІНШІ ГОДИННИКИ З ВИКОНАВЧИМИ  
МЕХАНІЗМАМИ
  - АВТОМАТИЗОВАНИ АСТРОНОМІЧНІ ПРИЛАДИ

[1.21]

ІСТОРИЧНА ІНФОРМАЦІЯ

■ З 17 СТОРІЧЧЯ

- МЕХАНІЧНІ ОБЧИСЛЮВАЛЬНІ СИСТЕМИ (ЛЕЙБНІЦ, ПАСКАЛЬ, БЕБІДЖ)
  
- ТОЧНІ МЕХАНІЧНІ ГОДИННИКИ

[1.22]

ІСТОРИЧНІ ВІДОМОСТІ

■ З 19 СТОРІЧЧЯ

- АНАЛОГОВІ ЕЛЕКТРИЧНІ РЕЛЕЙНІ ТА ЕЛЕКТРОННІ ОБЧИСЛЮВАЛЬНІ МАШИНИ
- РОЗВИТОК ЗАСОБІВ КОМУНІКАЦІЙ

[1.23]

ІСТОРИЧНА ІНФОРМАЦІЯ

■ З 1940-Х РОКІВ

- РЕЛЕЙНІ І ЛАМПОВІ ЦИФРОВІ ОБЧИСЛЮВАЛЬНІ СИСТЕМИ
- ФЕРИТОВА ПАМ'ЯТЬ
- ПОЯВА ТРАНЗИСТОРІВ

[1.24]

ІСТОРИЧНА ІНФОРМАЦІЯ

■ З 1950-1960-Х Р

- ЕОМ НА ТРАНЗИСТОРАХ
  - СССР – МЭСМ, БЭСМ
  - США – КОРПОРАЦІЯ DEC, CDC, ПЕРШІ КОМП'ЮТЕРИ ІВМ
  - ПЕРШІ БАГАТОПРОЦЕСОРНІ КОМП'ЮТЕРИ З ВІРТУАЛЬНОЮ ПАМ'ЯТТЮ
- ПОЯВА МІКРОСХЕМ

[1.25]

ІСТОРИЧНІ ВІДОМОСТІ

■ З 1970-Х РОКІВ

- ПЕРШІ МІКРОПРОЦЕСОРИ INTEL, TI
- ПЕРШІ ВЕЛИКІ ІНТЕГРАЛЬНІ МІКРОСХЕМИ
- ВЕКТОРНО-КОНВЕЙЄРНІ СИСТЕМИ
- МАСИВНО-ПАРАЛЕЛЬНІ І БАГАТО-КОМП'ЮТЕРНІ СИСТЕМИ
- ПРОМИСЛОВІ СУПЕРКОМП'ЮТЕРИ CRAУ
- ЕЛЕКТРОННІ НАРУЧНІ ГОДИННИКИ, МІКРОКАЛЬКУЛЯТОРИ

[1.26]

ІСТОРИЧНІ ВІДОМОСТІ

■ З 1980-Х РОКІВ

- ПЕРСОНАЛЬНІ КОМП'ЮТЕРИ
- ІНТЕРНЕТ
- КЛАСТЕРНІ СИСТЕМИ
- ТРАНСП'ЮТЕРИ
- ВБУДОВУВАНІ СИСТЕМИ КЕРУВАННЯ

[1.27]

#### ІСТОРИЧНІ ВІДОМОСТІ

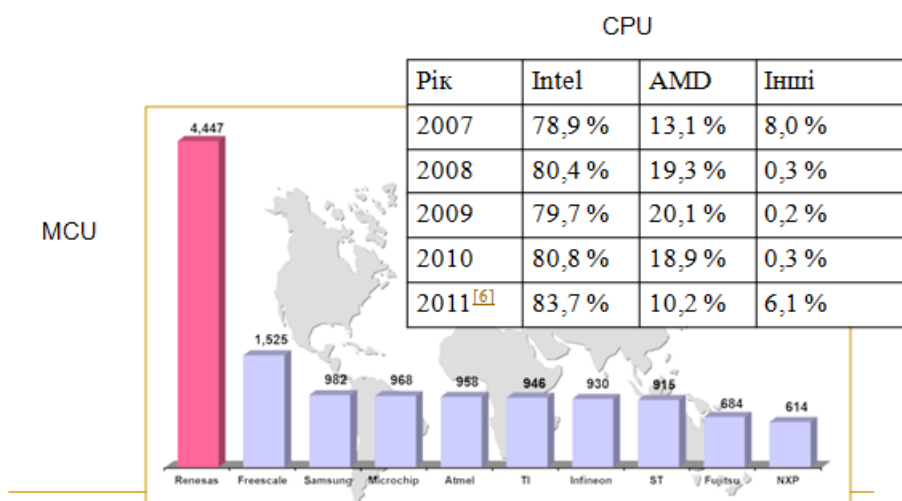
##### ■ З 1990-Х

- ЦИФРОВІ ГАДЖЕТИ
- ТОНКІ КЛІЄНТИ
- ВІРТУАЛІЗАЦІЯ
- БАГАТОЯДЕРНІСТЬ
- ...

[1.28]

#### ВИРОБНИЦТВО МІКРОПРОЦЕСОРІВ

### Виробництво мікропроцесорів



[1.29]

#### ПЕРСПЕКТИВИ РОЗВИТКУ МІКРОПРОЦЕСОРІВ

##### ■ НОВІ ІДЕЇ

- ОПТИЧНІ
- КВАНТОВІ
- НЕЙРО

##### ■ НОВІ АРХІТЕКТУРИ

- ПАРАЛЕЛІЗМ

##### ■ НОВІ ТЕХНОЛОГІЇ

- ЗМЕНШЕННЯ МАСШТАБУ
- ПІДВИЩЕННЯ ТАКТОВОЇ ЧАСТОТИ
- НОВІ МАТЕРІАЛИ

## ЛЕКЦІЯ 02

[2.1]

### ЦИФРОВЕ ПРЕДСТАВЛЕННЯ ІНФОРМАЦІЇ

#### ЛЕКЦІЯ 2

СУДАКОВ О.О, РАДЧЕНКО С.П.

«СУЧАСНА МІКРОПРОЦЕСОРНА ТЕХНІКА»

[2.2]

#### ІНФОРМАЦІЯ

- **ІНФОРМАЦІЯ** – БУДЬ-ЯКІ ДАНІ ПРО ЗОВНІШНІЙ СВІТ
  - Є ІНФОРМАЦІЯ – ЩОСЬ ВІДОМО; НЕВІДОМО - НЕМАЄ ІНФОРМАЦІЇ
  - ПРИЙШЛА ІНФОРМАЦІЯ – ЩОСЬ ВЗНАЛИ; НІЧОГО НЕ ВЗНАЛИ – НЕ ПРИЙШЛА ІНФОРМАЦІЯ
  
- **ІНФОРМАЦІЙНІ СИСТЕМИ** – ПРАЦЮЮТЬ З ІНФОРМАЦІЮ (БУДЬ-ЯКИМИ ДАНИМИ)
  - ОБРОБКА ІНФОРМАЦІЇ
  - ЗБЕРЕЖЕННЯ ІНФОРМАЦІЇ
  - ПЕРЕТВОРЕННЯ ІНФОРМАЦІЇ
  - ..
  
- **АНАЛОГОВЕ ПРЕДСТАВЛЕННЯ** – ІНФОРМАЦІЯ ПРЕДСТАВЛЯЄТЬСЯ ЗА ДОПОМОГОЮ ДЕЯКОЇ ФІЗИЧНОЇ ВЕЛИЧИНИ, ЯКА ПРИЙМАЄ НЕПЕРЕРВНІ ЗНАЧЕННЯ
  
- **ЦИФРОВЕ ПРЕДСТАВЛЕННЯ** – ІНФОРМАЦІЯ ПРЕДСТАВЛЯЄТЬСЯ ЗА ДОПОМОГОЮ ДЕЯКОЇ ФІЗИЧНОЇ ВЕЛИЧИНИ, ЯКА ЗМІНЮЄТЬСЯ ДИСКРЕТНО (НЕ НЕПЕРЕРВНОГО)
  
- **СИГНАЛ** – ФІЗИЧНІ ПРОЦЕСИ (АБО ДЕЯКІ ФІЗИЧНІ ВЕЛИЧИНИ ЦИХ ПРОЦЕСІВ), ЯКІ ПЕРЕНОСЯТЬ ІНФОРМАЦІЮ

[2.3]

#### АНАЛОГОВЕ ПРЕДСТАВЛЕННЯ

#### ІНФОРМАЦІЇ

- **ПАРАМЕТРИ ЗОВНІШНЬОГО СВІТУ ВЗАЄМНО-ОДНОЗНАЧНО ВІДОБРАЖАЮТЬСЯ НА ЗНАЧЕННЯ ДЕЯКИХ ФІЗИЧНИХ ВЕЛИЧИН СИГНАЛУ**
  - НАПРУГА НА КІНЦЯХ ПРОВІДНИКА ВІДПОВІДАЄ ЯКИМОСЬ ДАНИМ (ЕЛЕКТРИЧНИЙ СИГНАЛ)
  - ТИСК ПОВІТРЯ ВІДПОВІДАЄ ЯКИМОСЬ ДАНИМ (ЗВУКОВИЙ СИГНАЛ)
  
- **ПЕРЕДАЧА ІНФОРМАЦІЇ** – ПЕРЕМІЩЕННЯ ІНФОРМАЦІЇ У ПРОСТОРИ
  - НОВА ІНФОРМАЦІЯ – НЕПЕРЕДБАЧУВАНА ЗМІНА СИГНАЛУ
  - ПЕРЕДБАЧУВАНА ЗМІНА НЕ НЕСЕ НОВОЇ ІНФОРМАЦІЇ
  
- **МОДУЛЯЦІЯ** - ЗМІНА ЯКОГОСЬ ПАРАМЕТРА СИГНАЛУ ВІДПОВІДНО ДО ЗМІНИ ІНФОРМАЦІЇ

- БУДЬ-ЯКА ІНФОРМАЦІЙНА СИСТЕМА НА ТОМУ ЧИ ІНШОМУ ЕТАПІ ВИКОРИСТОВУЄ АНАЛОГОВЕ ПРЕДСТАВЛЕННЯ ІНФОРМАЦІЇ
  - ФІЗИЧНІ ВЕЛИЧИНИ НА ПРАКТИЦІ ЗДЕБІЛЬШОГО ЗМІНЮЮТЬСЯ НЕПЕРЕРВНИМ ЧИНОМ

[2.4]

#### ДИСКРЕТНЕ ПРЕДСТАВЛЕННЯ ІНФОРМАЦІЇ

- ДАНІ ПРЕДСТАВЛЯЮТЬСЯ У ВИГЛЯДІ ЗЛІЧЕНОГО НАБОРУ ЗНАЧЕНЬ ФІЗИЧНОЇ ВЕЛИЧИНИ
  - ЧЕРВОНИЙ, ЖОВТИЙ, ЗЕЛЕНИЙ
  - ВИСОКА-НИЗЬКА НАПРУГА
- ФІЗИЧНІ ВЕЛИЧИНИ ЗМІНЮЮТЬСЯ НЕПЕРЕРВНО – ДИСКРЕТНОЮ МОЖЕ ЛИШЕ ІНТЕРПРЕТАЦІЯ ДІАПАЗОНІВ ЗМІНИ ФІЗИЧНОЇ ВЕЛИЧИНИ І ЧАСОВИХ ІНТЕРВАЛІВ
- **ДИСКРЕТИЗАЦІЯ** - ПРЕДСТАВЛЕННЯ НЕПЕРЕРВНОГО У ЧАСІ СИГНАЛУ ЗА ДОПОМОГОЮ ДИСКРЕТНОГО НАБОРУ ВІДЛІКІВ
- **КВАНТУВАННЯ** – ПРЕДСТАВЛЕННЯ НЕПЕРЕРВНОГО ДІАПАЗОНУ ЗНАЧЕНЬ СИГНАЛУ У ВИГЛЯДІ ДИСКРЕТНОГО НАБОРУ РІВНІВ
- **МАНІПУЛЯЦІЯ** – МОДУЛЯЦІЯ ДИСКРЕТНИМИ ДАНИМИ (SK – SHIFT KEY)

[2.5]

#### ПАРАМЕТРИ СИГНАЛІВ

- ЕЛЕКТРИЧНА НАПРУГА, СТРУМ
  - ВИСОКИЙ-НИЗЬКИЙ РІВЕНЬ
- ЧАСТОТА
- ФАЗА
  - PSK
- ТРИВАЛІСТЬ (ШИРИНА) ІМПУЛЬСІВ
  - PWM
  - ДЕЛЬТА
  - ДЕЛЬТА-СИГМА
- АМПЛІТУДА КОЛИВАНЬ
  - QAM

[2.6]

#### ДАТЧИКИ

- **ДАТЧИК** (ДЖЕРЕЛО ДАНИХ, СЕНСОР) ВІД АНГЛ. DATA – ДАНІ
  - НЕ ВІД СЛОВА ДАВАТИ!
- ПРИСТРІЙ, ЯКИЙ ПЕРЕТВОРЮЄ ІНФОРМАЦІЮ ПРО ЗОВНІШНІЙ СВІТ У СИГНАЛ
  - ДАТЧИК ХОЛЛА – ПЕРЕТВОРЮВАЧ МАГНІТНЕ ПОЛЕ-ЕЛЕКТРИЧНА НАПРУГА
  - ПЕРЕТВОРЮВАЧ ТЕМПЕРАТУРА-ЧАСТОТА
  - МІКРОФОН
  - ФОТОДІОД
  - ЄМНІСНИЙ ДАТЧИК



- ОСНОВНІ ХАРАКТЕРИСТИКИ

- МАКСИМАЛЬНЕ ТА МІНІМАЛЬНЕ ЗНАЧЕННЯ ВХІДНИХ І ВИХІДНИХ ПАРАМЕТРІВ
- ДИНАМІЧНИЙ ДІАПАЗОН
- КРУТИЗНА ПЕРЕТВОРЕННЯ

[2.7]

ПРОБЛЕМА АНАЛОГОВОГО ПРЕДСТАВЛЕННЯ

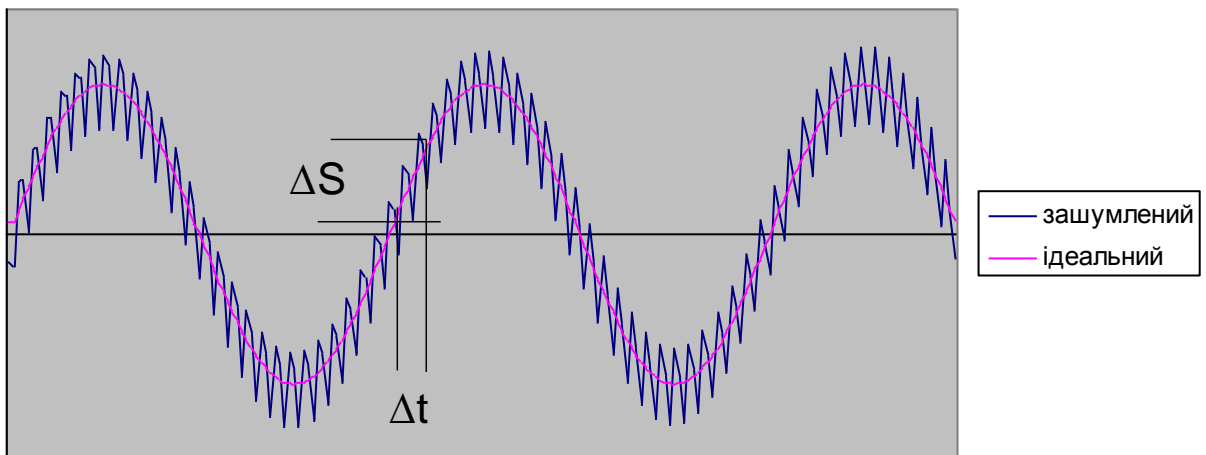
- НЕПЕРЕРВНИЙ СИГНАЛ МОЖЕ НЕСТИ НЕСКІНЧЕННУ КІЛЬКІСТЬ ІНФОРМАЦІЇ

- ВЗАЄМОДІЯ СИГНАЛУ ІЗ ЗОВНІШНІМ СЕРЕДОВИЩЕМ ПРИЗВОДИТЬ ДО НЕКОНТРОЛЬОВАНИХ ЗМІН – ШУМІВ

- ГАРАНТУВАНО (З ВИСОКОЮ ІМОВІРНІСТЮ) МОЖНА ВІДНОВИТИ ЛИШЕ ПОПАДАННЯ СИГНАЛУ В ІНТЕРВАЛ ЗНАЧЕНЬ  $\Delta S$  В ІНТЕРВАЛІ ЧАСУ  $\Delta t$  – ЕКВІВАЛЕНТНО ДИСКРЕТНОМУ СИГНАЛУ

$$N_q \approx \frac{S_{\max} - S_{\min}}{|\Delta S|} \approx \frac{\text{Signal}}{\text{Noise}} \quad \Delta t \approx \frac{|\Delta S|}{|dS/dt|} \approx \frac{\text{Noise}}{|dS/dt|}$$

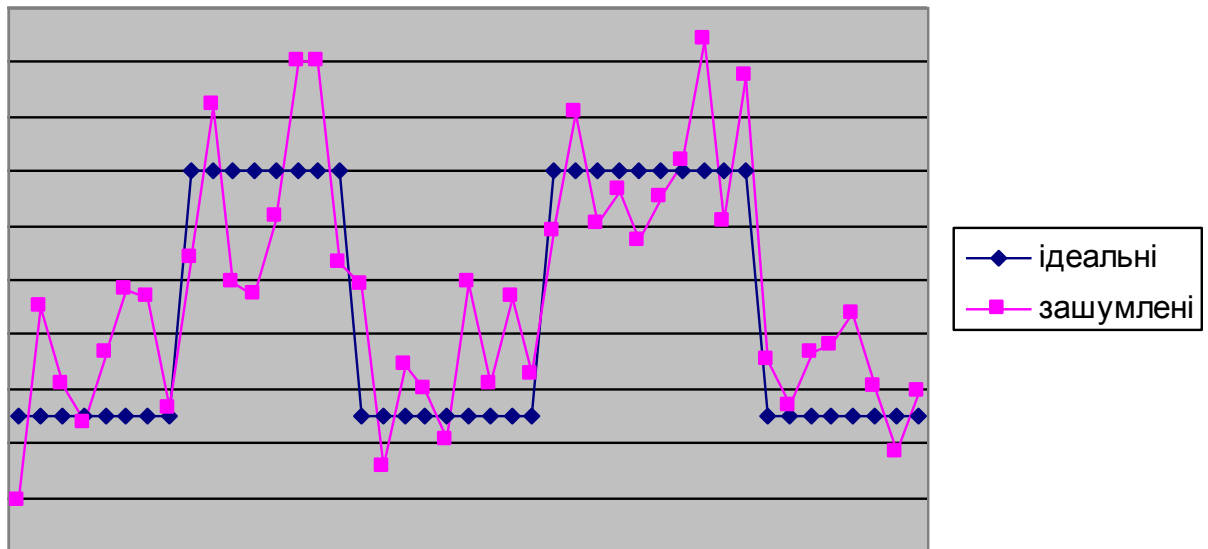
**МАКСИМАЛЬНИЙ ШУМ - МІНІМАЛЬНА КІЛЬКІСТЬ РІВНІВ КВАНТУВАННЯ 2**



[2.8]

ПЕРЕВАГИ ЦИФРОВОГО ПРЕДСТАВЛЕННЯ

- ПОВНЕ ВІДНОВЛЕННЯ СИГНАЛУ (ПІСЛЯ ПЕРЕДАЧІ, ЗАПИСУ, ОБРОБКИ)
- ПРОСТОТА І ТОЧНІСТЬ ВИКОНАННЯ ПЕРЕТВОРЕННЯ, ОБРОБКИ
- МОЖНА ВИКОРИСТОВУВАТИ НИЗЬКОЯКІСНІ СХЕМИ, КАНАЛИ ПЕРЕДАЧІ
- НЕДОЛІК – ПЕРЕДАЄТЬСЯ МЕНШЕ ІНФОРМАЦІЇ, НІЖ ПО АНАЛОГОВОМУ КАНАЛУ, НЕОБХІДНІСТЬ ПЕРЕТВОРЕННЯ АНАЛОГ-ЦИФРА-АНАЛОГ



[2.9]

#### ДИСКРЕТНЕ ПРЕДСТАВЛЕННЯ

- Є МАКСИМАЛЬНЕ І МІНІМАЛЬНЕ ЗНАЧЕННЯ СИГНАЛУ
- НЕПЕРЕРВНИЙ СИГНАЛ – НЕСКІНЧЕННИЙ НАБІР ЗНАЧЕНЬ
- ДИСКРЕТНИЙ СИГНАЛ - СКІНЧЕННИЙ НАБІР ЗНАЧЕНЬ (РІВНІВ КВАНТУВАННЯ  $NQ$ ) У ДИСКРЕТНІ МОМЕНТИ ЧАСУ  $i\Delta T$
- ДЛЯ ПРЕДСТАВЛЕННЯ ДИСКРЕТНОГО СИГНАЛУ ДОСТАТНЬО ЦІЛИХ ЧИСЕЛ ВІД 0 ДО  $NQ-1$
- ДЛЯ ЗАПИСУ ЦІЛИХ ЧИСЕЛ ЗАСТОСОВУЮТЬ СИСТЕМИ ЧИСЛЕННЯ

[2.10]

#### ПОЗИЦІЙНІ СИСТЕМИ ЧИСЛЕННЯ

- ЦІЛІ ЧИСЛА МОЖНА ЗАПИСАТИ ЗА ДОПОМОГОЮ РІЗНИХ СИСТЕМ ЧИСЛЕННЯ

$$x = a_n a_{n-1} \dots a_1 a_0 = \sum_{i=0}^n a_i b^i$$

- $N$ -КІЛЬКІСТЬ РОЗРЯДІВ,  $b > 0$  – ОСНОВА СИСТЕМИ ЧИСЛЕННЯ  $0 \leq a_i < b$  – ЦИФРА  $i$ -ГО РОЗРЯДУ ( $i$ -Ї ПОЗИЦІЇ)
- РОЗРЯД 0 – МОЛОДШИЙ (НАЙМЕНШ ЗНАЧУЩИЙ)
- РОЗРЯД  $n$  – СТАРШИЙ (НАЙБІЛЬШ ЗНАЧУЩИЙ)

$$n = \lfloor \log_b x \rfloor + 1$$

[2.11]

#### ПЕРЕВЕДЕННЯ ІЗ ОДНІЄЇ СИСТЕМИ ЧИСЛЕННЯ В ІНШУ (1 ПІДХІД)

- Є ДЕКІЛЬКА ПІДХОДІВ

$$x = a_n a_{n-1} \dots a_1 a_0 = \sum_{i=0}^n a_i b^i$$

- $A_i$  – ЧИСЛА ОДНІЄЇ СИСТЕМИ ЧИСЛЕННЯ В ІНШІЙ

■ В – ОСНОВА ОДНІЄЇ СИСТЕМИ ЧИСЛЕННЯ В ІНШІЙ  
ЗРУЧНО ДЛЯ ПЕРЕВЕДЕННЯ В ДЕСЯТКОВУ

$$10100011_2 = 1 \cdot 2^0 + 1 \cdot 2^1 + 0 \cdot 2^2 + 0 \cdot 2^3 + 0 \cdot 2^4 + 1 \cdot 2^5 + 0 \cdot 2^6 + 1 \cdot 2^7 = 163_{10}$$

■ КІЛЬКІСТЬ РОЗРЯДІВ

$$n_{10} = \lfloor \log_{10} 163 \rfloor + 1 = 2 + 1 = 3 \qquad n_2 = \lfloor \log_2 163 \rfloor + 1 = 7 + 1 = 8$$

[2.12]

ПЕРЕВЕДЕННЯ ІЗ ОДНІЄЇ СИСТЕМИ ЧИСЛЕННЯ В ІНШУ (2 ПІДХІД)

$$x_0 = x \Rightarrow a_0 = x_0 \bmod b$$

$$x_1 = \text{int}(x_0 / b) \Rightarrow a_1 = x_1 \bmod b$$

$$x_2 = \text{int}(x_1 / b) \Rightarrow a_2 = x_2 \bmod b$$

...

■ ЗРУЧНО ДЛЯ ПЕРЕВЕДЕННЯ З ДЕСЯТКОВОЇ

$$163_{10} \rightarrow b = 16$$

$$x_0 = 163 \Rightarrow a_0 = 163 \% 16 = 3$$

$$x_1 = \text{int}(163 / 16) = 10 \Rightarrow a_1 = 10 \% 16 = A$$

$$163_{10} = A3_{16}$$

■ КІЛЬКІСТЬ РОЗРЯДІВ

$$n_{16} = \lfloor \log_{16} 163 \rfloor + 1 = 1 + 1 = 2$$

[2.13]

КІЛЬКІСТЬ ІНФОРМАЦІЇ

■ ЧИМ ВИЩА ІМОВІРНІСТЬ ПОВІДОМЛЕННЯ – ТИМ МЕНШЕ В НЬОМУ ІНФОРМАЦІЇ (ШЕННОН 1948 Р.)

■ ІМОВІРНІСТІ ДВОХ НЕЗАЛЕЖНИХ ПОВІДОМЛЕНЬ ПЕРЕМНОЖУЮТЬСЯ, А КІЛЬКІСТЬ ІНФОРМАЦІЇ ПОВИННА СКЛАДАТИСЬ

$$M = \langle \log_b p \rangle = \int \frac{dp(x)}{dx} \log_b p(x) dx = \sum_i p_i \log_b p_i$$

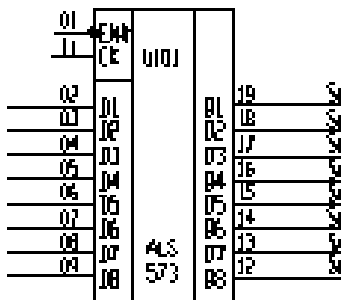
■ ІМОВІРНІСТЬ ВИЗНАЧАЄТЬСЯ КІЛЬКІСТЮ МОЖЛИВИХ КОМБІНАЦІЙ КОЛИ ВСІ РІВНІ КВАНТУВАННЯ РІВНОЙМОВІРНІ

$$M = - \sum_{i=0}^{N_q-1} \frac{1}{N_q} \log_b \frac{1}{N_q} = \log_b N_q = n_b$$

[2.14]

ВИКОРИСТАННЯ ДВІЙКОВОЇ СИСТЕМИ ЧИСЛЕННЯ

■ ОСНОВА ДВІЙКОВОЇ СИСТЕМИ В=2 ВІДПОВІДАЄ НАЙБІЛЬШІЙ ЗАВАДОСТІЙКОСТІ – МІНІМАЛЬНА КІЛЬКІСТЬ РІВНІВ КВАНТУВАННЯ



□ КОЖЕН РОЗРЯД – ОКРЕМИЙ ПРОВІДНИК

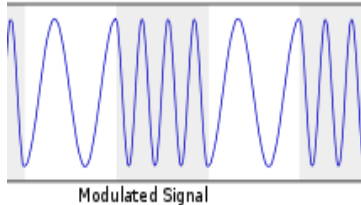
■ ЦИФРА(БІТ) – Є СИГНАЛ, АБО НЕМАЄ (VС)

■ ЦИФРА(БІТ) – ОКРЕМА ЧАСТОТА (FSK)

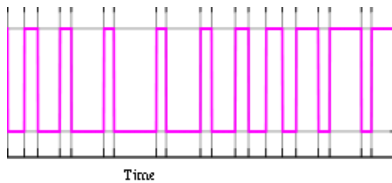
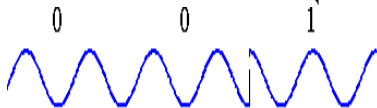
■ ЦИФРА(БІТ) – ПРОТИЛЕЖНІ ЗНАЧЕННЯ ФАЗИ 0 АБО 180 PSK



- КОЖЕН РОЗРЯД – ОКРЕМА ЧАСТОТА (FDM)
- ЦИФРА Є СИГНАЛ, АБО НЕМАЄ
- ЦИФРА ПРОТИЛЕЖНІ ЗНАЧЕННЯ ФАЗИ 0 АБО 180 (FSK)



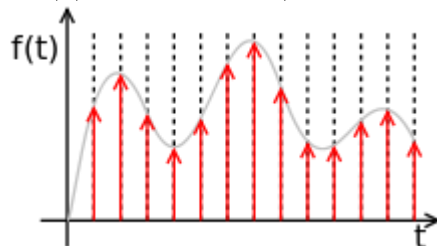
- КОЖЕН РОЗРЯД – ОКРЕМИЙ МОМЕНТ ЧАСУ (TDM)
- ЦИФРА (БІТ) Є СИГНАЛ, АБО НЕМАЄ (TSK)
- ЦИФРА – ПРОТИЛЕЖНІ ЗНАЧЕННЯ ФАЗИ 0 АБО 180 (PSK)
- ЦИФРА – ОКРЕМА ЧАСТОТА (FSK))
- ЦИФРА – ТРИВАЛІСТЬ ІМПУЛЬСУ (PWM)



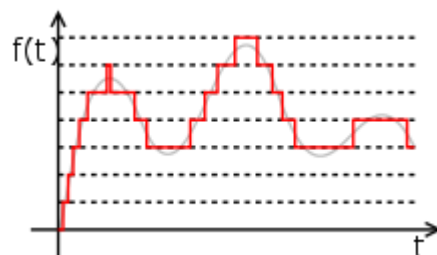
[2.15]

### АНАЛОГО-ЦИФРОВЕ І ЦИФРО-АНАЛОГОВЕ ПЕРЕТВОРЕННЯ

- АЦП (ADC)
- ДИСКРЕТИЗАЦІЯ

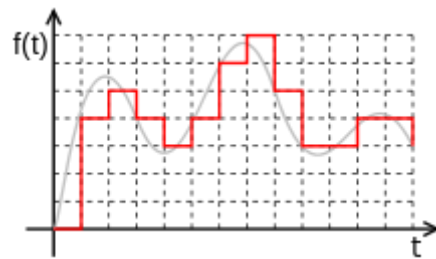
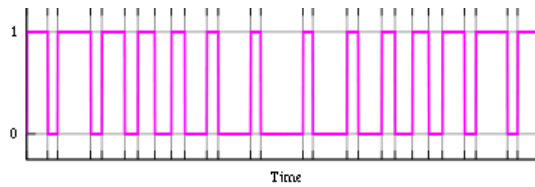


- КВАНТУВАННЯ

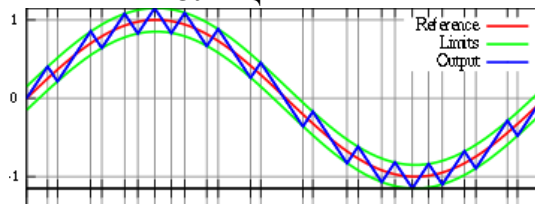


ЦАП (DAC)

- МОДУЛЯЦІЯ

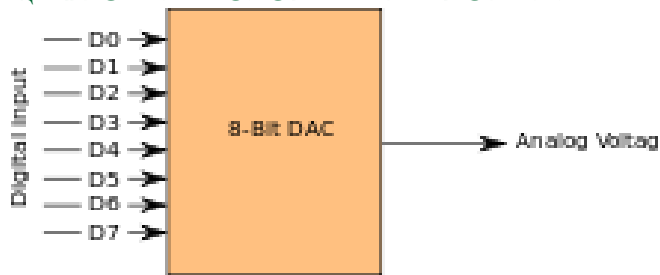


ІНТЕРПОЛЯЦІЯ



[2.16]

ЦИФРО-АНАЛОГОВЕ ПЕРЕТВОРЕННЯ



- R-2R МАТРИЦЯ (СХОДИНКОВИЙ)
  - МИТТЄВЕ ПЕРЕТВОРЕННЯ
  - СЕРЕДНЯ ЯКІСТЬ
  - ДЕШЕВА СХЕМА
  - ВЕЛИКА КІЛЬКІСТЬ РЕЗИСТОРІВ
- ШИРОТНО-ІМПУЛЬСНА МОДУЛЯЦІЯ
  - ВИСОКА ЯКІСТЬ
  - НИЗЬКА ЧАСТОТА ПЕРЕТВОРЕННЯ
- СИГМА-ДЕЛЬТА (ПЕРЕДИСКРЕТИЗАЦІЯ)
  - ВИСОКА ЯКІСТЬ
  - НИЗЬКА ЧАСТОТА
- ІНТЕРПОЛЯЦІЯ
  - ІНТЕГРАТОР
  - ФІЛЬТР НИЖНІХ ЧАСТОТ

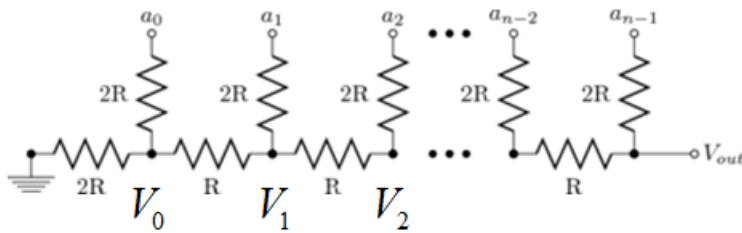
[2.17]

РЕЗИСТИВНІ МАТРИЦІ

- ПО ТЕОРЕМІ ТЕВЕНЕНА
  - ЗНАЧЕННЯ 0 – РЕЗИСТОРИ ПАРАЛЕЛЬНО – ОПІР R

□ ЗНАЧЕННЯ 1 – РЕЗИСТОРИ ПАРАЛЕЛЬНО ОПІР ДЖЕРЕЛА R

$$V_{out} = \frac{V_{ref} * a}{2^N}$$



[2.18]

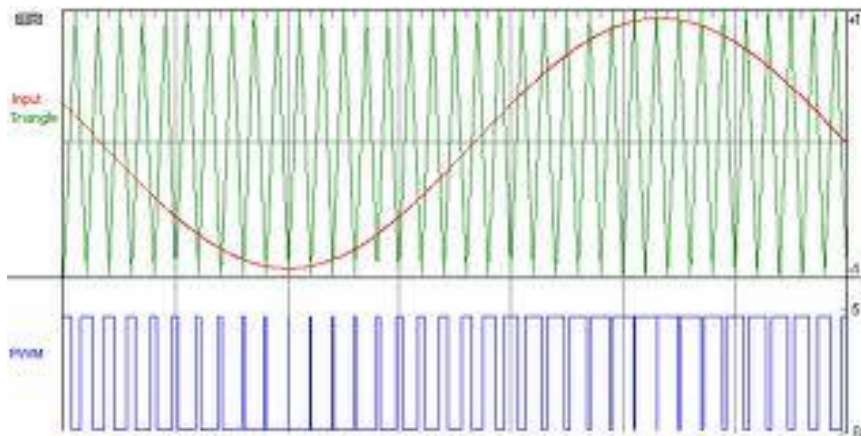
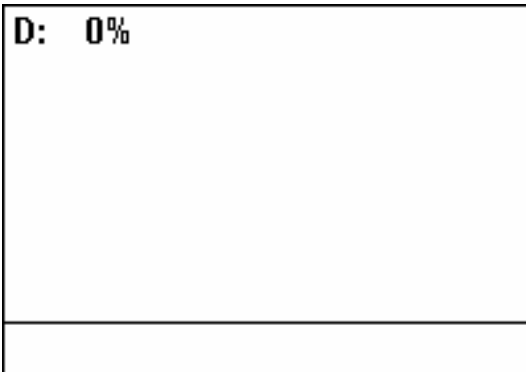
### ШИРОТНО-ІМПУЛЬСНА МОДУЛЯЦІЯ

■ ЗМІНЮЄТЬСЯ ШИРИНА ІМПУЛЬСА (СКВАЖНІСТЬ)

$D = T/T$

■ НА ВИХОДІ ІНТЕГРАТОРА ВСТАНОВИТЬСЯ ЗНАЧЕННЯ НАПРУГИ  $U_{REF} * D$

■ ЧАСТОТА ІМПУЛЬСІВ ЗНАЧНО ВИЩА, НІЖ ЧАСТОТА СИГНАЛУ



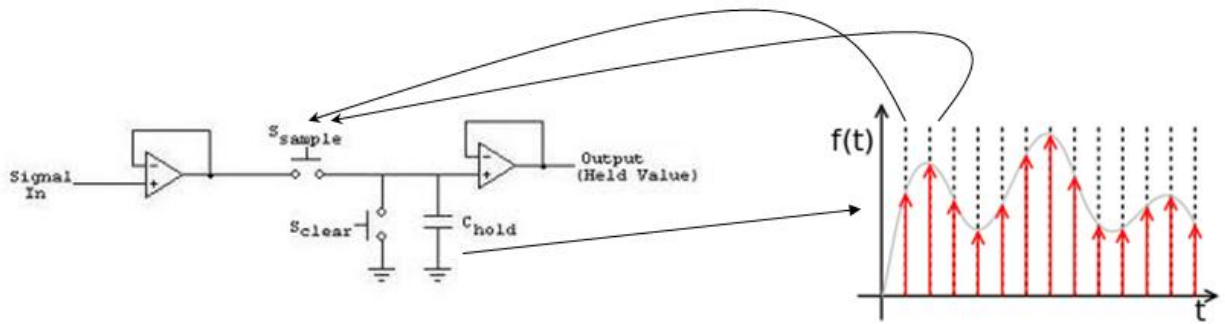
[2.19]

### АЦП - ДИСКРЕТИЗАЦІЯ (SAMPLE AND HOLD)

■ ПЕРІОД ДИСКРЕТИЗАЦІЇ  $\Delta T = T_S + T_Q$

■ ЧАС НАКОПИЧЕННЯ (ДИСКРЕТИЗАЦІЇ)  $T_S = 3-10 * R_{ВИХ} * C_{HOLD}$

■ ЧАС ПЕРЕТВОРЕННЯ (КВАНТУВАННЯ)  $T_Q$

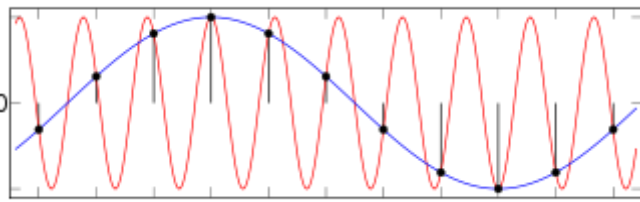


[2.20]

### ВИБІР ЧАСУ ДИСКРЕТИЗАЦІЇ

- ТЕОРЕМА КОТЕЛЬНИКОВА (ШЕННОНА)

$$\Delta t \leq \frac{1}{2f_{\max}}$$



- ALIASING – ОДНАКОВИЙ РЕЗУЛЬТАТ ДЛЯ РІЗНИХ СИГНАЛІВ КОЛИ ВЕЛИКИЙ ПЕРІОД ДИСКРЕТИЗАЦІЇ

- ДЛЯ ЗАПОБІГАННЯ

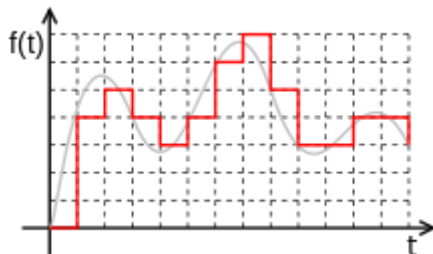
- ВЕЛИКИЙ  $C_{\text{HOLD}}$  МАЛИЙ  $\Delta t$
- ПЕРЕДИСКРЕТИЗАЦІЯ

$$\Delta t \leq \frac{1}{(5-10)f_{\max}}$$

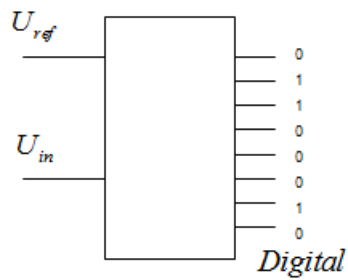
[2.21]

### КВАНТУВАННЯ (ПЕРЕТВОРЕННЯ)

- НЕОБХІДНО ЗАДАТИ РЕФЕРЕНТНУ НАПРУГУ  $U_{\text{REF}}$
- НЕПЕРЕРВНІ (МИТТЄВІ)
  - МАЛІ ЧАСИ НАКОПИЧЕННЯ  $\rightarrow 0$
  - МАЛІ ЧАСИ КВАНТУВАННЯ  $\rightarrow 0$
  - МАЛА КІЛЬКІСТЬ РІВНІВ (ДО 8 БІТ)
  - ДОРОГА СХЕМА
  - ШВИДКІ ПРОЦЕСИ – ВІДЕО, ОСЦИЛОГРАФИ.. NS



- SAR – SUCCESSIVE APPROXIMATION
  - НЕОБХІДНІТЬ SAMPLE AND HOLD
  - ТРИВАЛІ ЧАСИ ПЕРЕТВОРЕННЯ MS-MKS
  - СЕРЕДНЯ КІЛЬКІСТЬ РІВНІВ (8-12 БІТ)
  - МІКРОКОНТРОЛЕРИ



■ ДЕЛЬТА-СИГМА

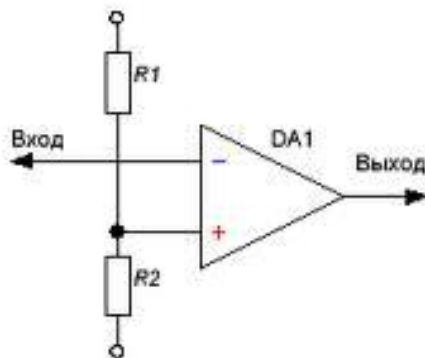
- ПЕРЕТВОРЕННЯ НАПРУГА-ЧАСТОТА
- ТРИВАЛІ ЧАСИ ПЕРЕТВОРЕННЯ MS
- ВЕЛИКА КІЛЬКІСТЬ РІВНІВ (8-12 БІТ)
- ВИСОКОЯКІСНА ОБРОБКА ЗВУКУ,



[2.22]

КОМПАРАТОР

- ОДНОБІТНИЙ АЦП
- НАПРУГА НА ВХОДІ ПОРІВНЮЄТЬСЯ З РЕФЕРЕНТНОЮ
  - $U_{BX} < U_{РЕФ}$  – НА ВИХОДІ 1
  - $U_{BX} > U_{РЕФ}$  – НА ВИХОДІ 0

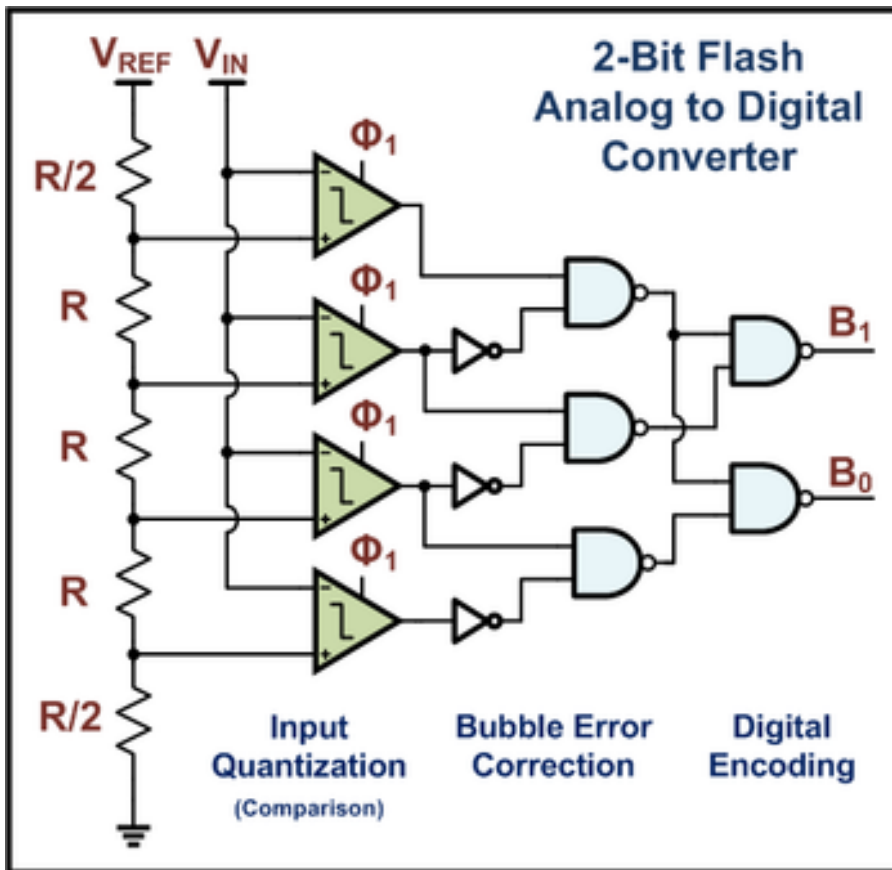


[2.23]

МИТТЄВІ ПЕРЕТВОРЮВАЧІ

- МАТРИЦЯ РЕЗИСТОРІВ ДІЛИТЬ РЕФЕРЕНТНУ НАПРУГУ НА РІВНІ КВАНТУВАННЯ
- КОМПАРАТОРИ ПОРІВНЮЮТЬ ВХІДНИЙ СИГНАЛ ІЗ ВСІМА РІВНЯМИ
- ЦИФРОВИЙ КОД ВІДПОВІДАЄ НАЙМЕНШОМУ РІВНЮ, НА ВИХОДІ ЯКОГО КОМПАРАТОР ВИДАЄ 1

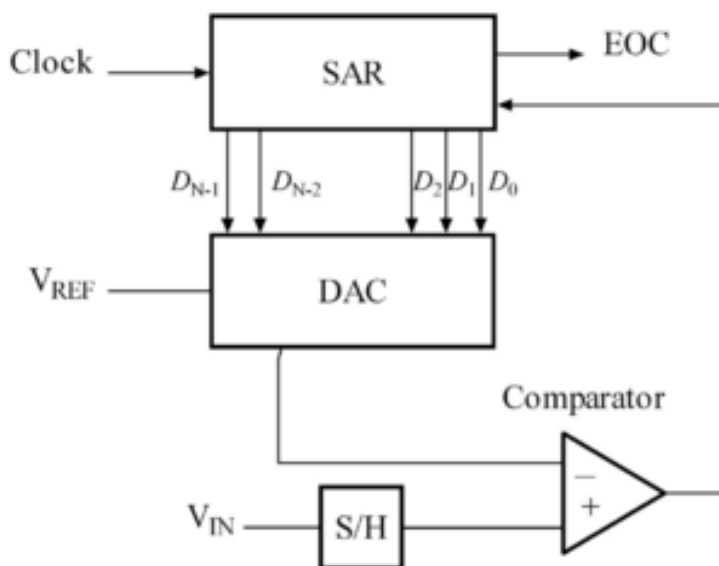




[2.24]

#### SAR – SUCCESSIVE APPROXIMATION REGISTER

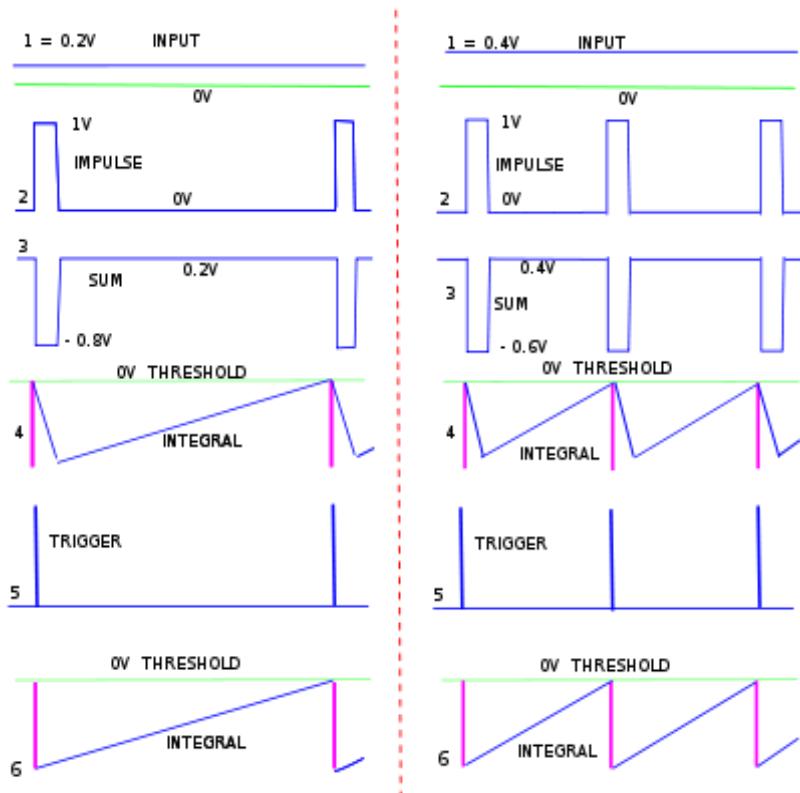
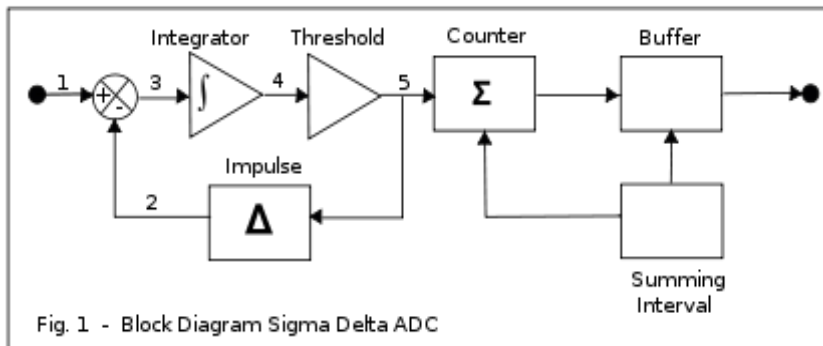
- ЦАП ВИДАЄ СИГНАЛ ВІД 0 ДО U<sub>REF</sub> НА ОСНОВІ ЗНАЧЕННЯ В РЕГІСТРІ SAR
- СИГНАЛ НА ВИХОДІ ЦАП ПОРІВНЮЄТЬСЯ З ВХІДНИМ КОМПАРАТОРОМ
  - UB<sub>X</sub> > U<sub>SAR</sub> 1
  - UB<sub>X</sub> < U<sub>SAR</sub> 0
- ЦЕЙ БІТ ЗАПИСУЄТЬСЯ У ВІДПОВІДНУ ПОЗИЦІЮ В РЕГІСТР SAR
- ДЛЯ N РОЗРЯДІВ ТРЕБА N ПОРІВНЯНЬ І N ТАКТІВ ГЕНЕРАТОРА



[2.25]

#### ДЕЛЬТА-СИГМА

- ВХІДНИЙ СИГНАЛ СКЛАДАЄТЬСЯ З КОРОТКИМ ВІД'ЄМНИМ ІМПУЛЬСОМ  $\Delta$  АМПЛІТУДИ UРЕФ
- НА ВИХОДІ ІНТЕГРАТОРА ЗБІЛЬШУЄТЬСЯ НАПРУГА ПОКИ НЕ СПРАЦЮЄ ТРИГЕР
- КОЛИ СПРАЦЬОВУЄ ТРИГЕР ГЕНЕРУЄТЬСЯ НОВИЙ ІМПУЛЬС  $\Delta$
- ЧИМ ВИЩА НАПРУГА НА ВХОДІ, ТИМ ШВИДШЕ НАРОСТАЄ НАПРУГА НА ВИХОДІ ІНТЕГРАТОРА І ЧАСТІШЕ СЛІДУЮТЬ ІМПУЛЬСИ
- ЛІЧИЛЬНИК  $\Sigma$  РАХУЄ ІМПУЛЬСИ ЗА ПЕВНИЙ ПРОМІЖОК ЧАСУ – ЦЯ КІЛЬКІСТЬ – КОД НА ВИХОДІ
- ЗА ЧАС  $\Delta T$  ТРЕБА ЗРОБИТИ 1 ПІДРАХУНОК
- ЧАСТОТА ІМПУЛЬСІВ ПОВИННА БУТИ ДУЖЕ ВЕЛИКОЮ



[2.26]

### ПРЕДСТАВЛЕННЯ ЦІЛИХ ЧИСЕЛ В ПРОЦЕСОРНИХ СИСТЕМАХ

- ДОДАТНІ ТА ВІД'ЄМНІ ЧИСЛА
  - ДОДАТНІ – СТАРШІЙ БІТ -0

- ВІД'ЄМНИ СТАРШИЙ БІТ – 1
- КОРОТКІ SHORT 16 БІТ
- ДОВГІ LONG 32 БІТ
- LONG LONG 64 БІТ
- ПОРЯДОК БАЙТІВ
  - LSB FIRST ПЕРШИЙ НАЙМЕНШ ЗНАЧУЩИЙ БАЙТ
  - MSB FIRST ПЕРШИЙ НАЙБІЛЬШ ЗНАЧУЩИЙ БАЙТ

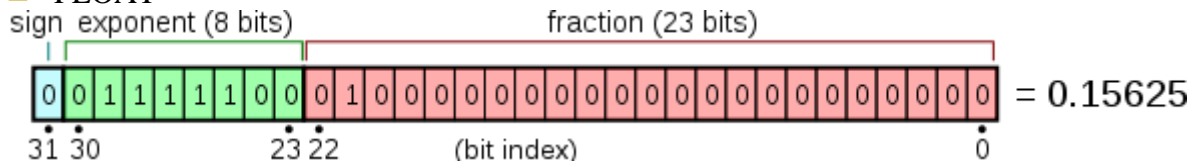
[2.27]

ЧИСЛА З ПЛАВАЮЧОЮ ТОЧКОЮ

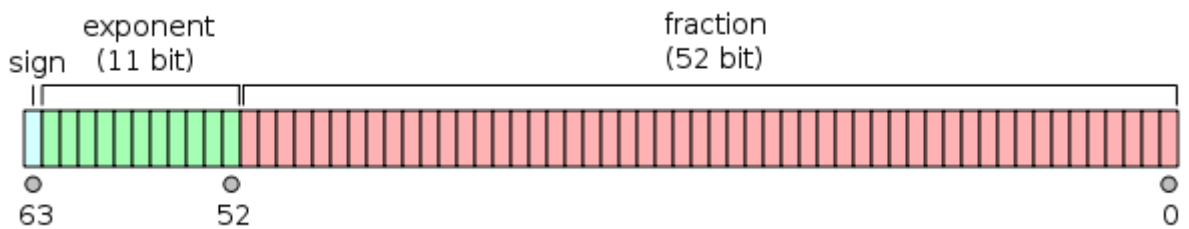
- ЕКСПОНЕНЦІАЛЬНЕ ПРЕДСТАВЛЕННЯ

- $A \cdot E^B$
- $ABS(A) < 10$

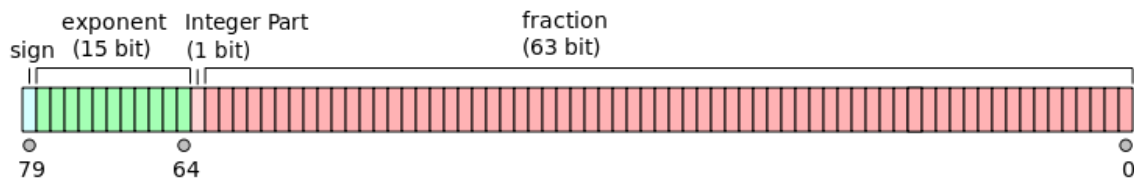
- FLOAT



- DOUBLE



- EXTENDED



[1.28]

ТЕКСТ

- МАСИВ, ЩО ЗАКІНЧУЄТЬСЯ НУЛЕМ

- КОЖЕН БАЙТ – СИМВОЛ
- КОЖНЕ СЛОВО СИМВО
- ОСТАННІЙ СИМВОЛ 0

## ЛЕКЦІЯ 03

[3.1]

ОСНОВИ ЦИФРОВОЇ СХЕМОТЕХНІКИ

ЛЕКЦІЯ 3

СУДАКОВ О.О, РАДЧЕНКО С.П.

«СУЧАСНА МІКРОПРОЦЕСОРНА ТЕХНІКА»

[2.2]

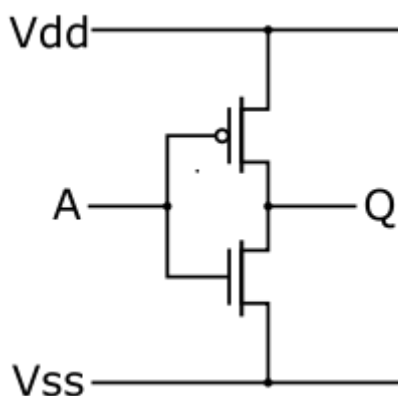
ЦИФРОВІ ІНФОРМАЦІЙНІ ПРИСТРОЇ

- ОБРОБКА ІНФОРМАЦІЇ У ЦИФРОВОМУ ВИГЛЯДІ
  - КЛЮЧІ, ВИМИКАЧІ, КОМУТАТОРИ
  - ЛОГІЧНІ ЕЛЕМЕНТИ
  - ГЕНЕРАТОРИ
  - ТРИГЕРИ
  - ЛІЧИЛЬНИКИ
  - ПЕРЕТВОРЮВАЧІ ЧАСТОТИ
  - РЕГІСТРИ
  - ЕЛЕМЕНТИ ПАМ'ЯТІ
  - ШИФРАТОРИ-ДЕШИФРАТОРИ
  - ФОРМУВАЧІ
  - МАТЕМАТИЧНІ ОПЕРАЦІЇ
  
- НАЙЧАСТІШЕ ВИКОРИСТОВУЮТЬ ДВІЙКОВЕ ПРЕДСТАВЛЕННЯ
  - ВИСОКИЙ РІВЕНЬ НАПРУГИ
  - НИЗЬКИЙ РІВЕНЬ НАПРУГИ

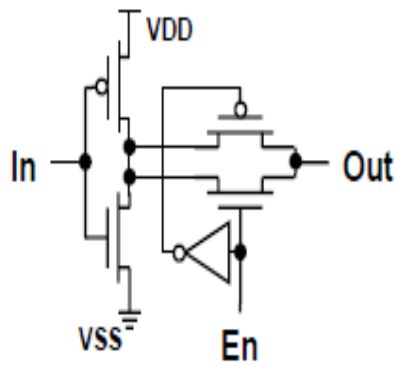
[3.3]

ЕЛЕКТРОННІ КМОП (CMOS) КЛЮЧІ (SWITCH)

- НАПРУГА НА ВИХОДІ ЗАЛЕЖИТЬ ВІД НАПРУГИ НА ВХОДІ
- 2 СТАНИ
  - ВИСОКИЙ РІВЕНЬ
  - НИЗЬКИЙ РІВЕНЬ



- 3 СТАНИ
  - ВИСОКИЙ
  - НИЗЬКИЙ
  - ЗАБОРОНЕНИЙ (ВИСОКИЙ ОПР, ПЛАВАЮЧИЙ, Z)

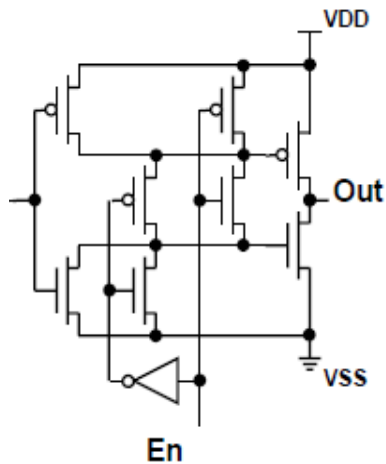


- VDD – ВИВІД СТОКА (DRAIN)
- VSS - ВИВІД ВИТОКУ (SOURCE)
- EN - ДОЗВІЛ

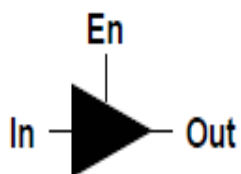
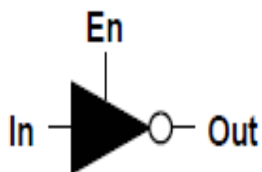
[3.4]

КМОП БУФЕР (ПОРТ, GATE)

- ПОСЛІДОВНО З'ЄДНАНІ КЛЮЧІ ДЛЯ ЗАБЕЗПЕЧЕННЯ ВИСОКОГО ВХІДНОГО ОПОРУ, НИЗЬКОГО ВИХІДНОГО, ВИСОКОЇ НАВАНТАЖУВАЛЬНОЇ СПРОМОЖНОСТІ
  - ІНВЕРТУЮЧІ
  - НЕІНВЕРТУЮЧІ

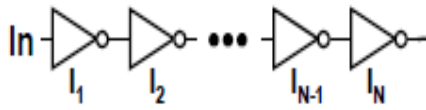


- ПІДТРИМКА ТРЕТЬОГО СТАНУ



- КОМУТАТОРИ

- ВИМИКАЧІ
- ПЕРЕМИКАЧІ

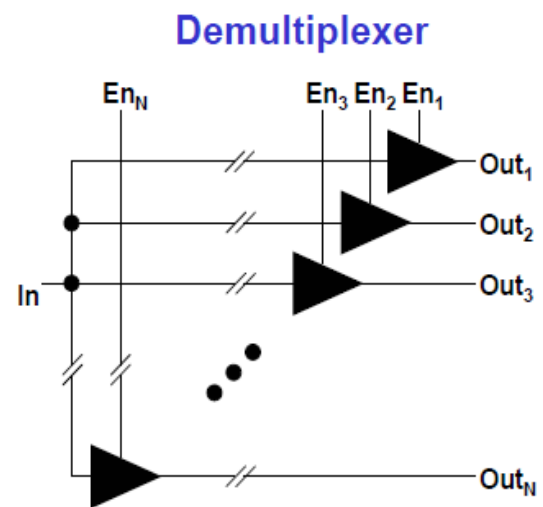
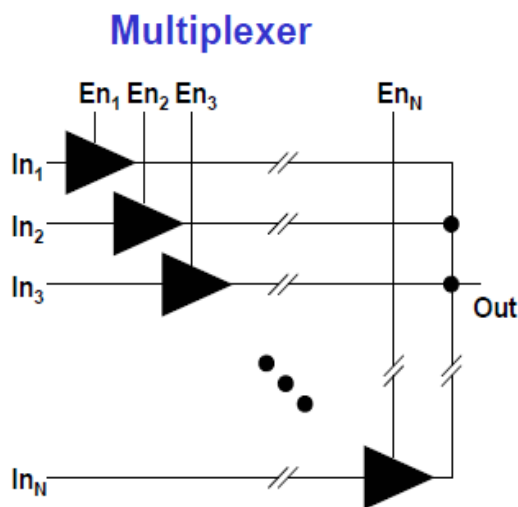


- ВИВОДИ МІКРОСХЕМ ПІДКЛЮЧЕНІ ЧЕРЕЗ ПОРТ ДО ЗОВНІШНЬОГО СВІТУ

[3.5]

### МУЛЬТИПЛЕКСОРИ, ДЕМУЛЬТИПЛЕКСОРИ

- КОМУТАТОР З БАГАТЬОХ ВХОДІВ НА 1 ВИХІД
- КОМУТАТОР З ОДНОГО ВХОДУ НА БАГАТО ВИХОДІВ
- РЕАЛІЗУЮТЬСЯ ЧЕРЕЗ БУФЕРИ

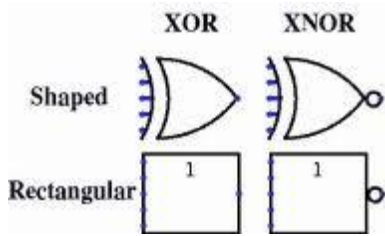


[3.6]

### ЛОГІЧНІ ЕЛЕМЕНТИ

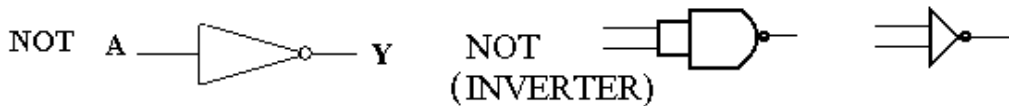
- ВИКОНАННЯ ЛОГІЧНИХ ОПЕРАЦІЙ
  - НЕ (NOT, INV)
  - АБО (OR, +)
  - І (AND, \*)
  - І-НЕ (NAND)
  - АБО-НЕ (NOR)
  - ВИКЛЮЧНЕ АБО (XOR)
  - ВИКЛЮЧНЕ АБО-НЕ (XNOR)
- РІВНІ СИГНАЛУ
  - ЛОГІЧНА 1 ЯК ПРАВИЛО ВИСОКИЙ РІВЕНЬ
  - ЛОГІЧНИЙ 0 ЯК ПРАВИЛО НИЗЬКИЙ РІВЕНЬ
- ОСНОВА БЛОК-СХЕМ ПРОЦЕСОРІВ І МІКРОКОНТРОЛЕРІВ – ТРЕБА ЗНАТИ
- ОСНОВА РОБОТИ БАГАТЬОХ ПРИСТРОЇВ

		Binary logic circuit	Used in this document	IEC approved
AND		AND		
NAND		NAND		
OR		OR		
NOR		NOR		
NOT		NOT (INVERTER)		



[3.7]

ЛОГІЧНІ ОПЕРАЦІЇ (NOT)



■ ІНВЕРСІЯ БІТА

NOT 1 = 0

NOT 0 = 1

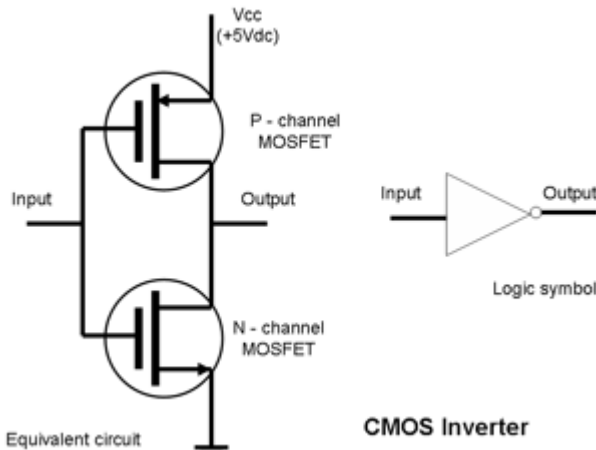
$\bar{1} = 0$

$\bar{0} = 1$

■ РЕАЛІЗУЄТЬСЯ

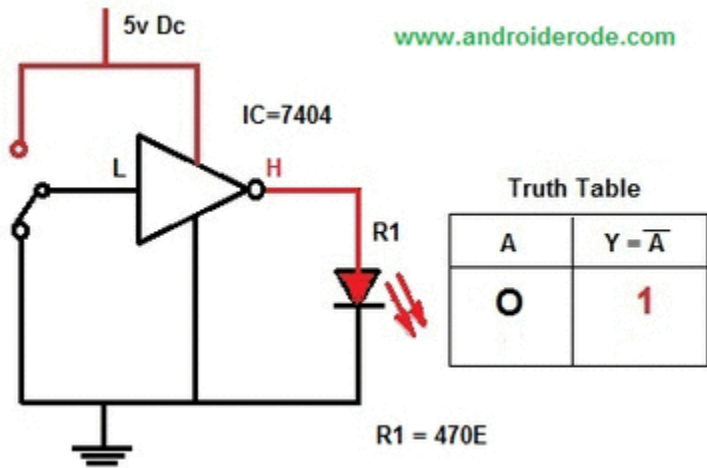
КМОП КЛЮЧ

КМОП GATE ІНВЕРТУЮЧИЙ



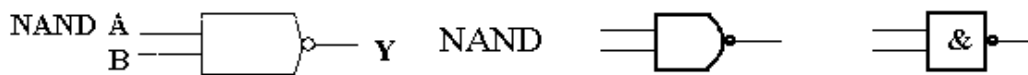
■ ВИКОРИСТАННЯ

□ ЗМІНА РІВНЯ НА ОБЕРНЕНИЙ



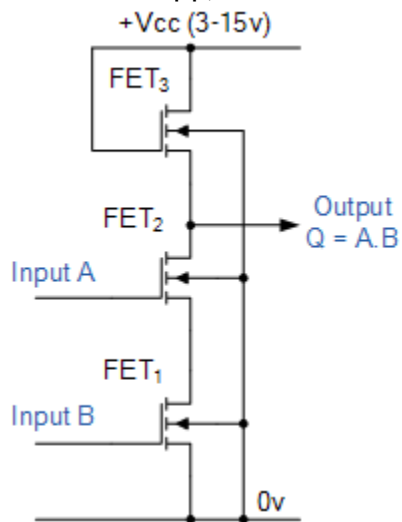
[3.8]

ЛОГІЧНІ ОПЕРАЦІЇ NAND



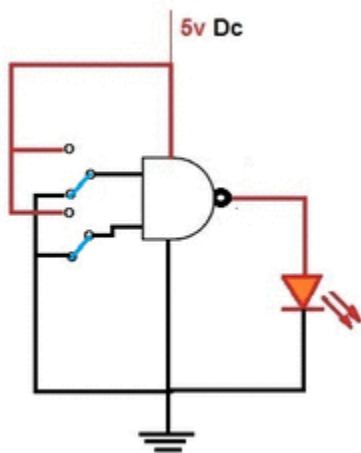
■ ЛОГІЧНА ОПЕРАЦІЯ І-НЕ

- 0 НА ВИХОДІ, ТІЛЬКИ КОЛИ ВСІ ВХОДИ 1
- 1 ВИХОДІ, КОЛИ 0 НА БУДЬ-ЯКОМУ ВХОДІ



■ БАЗОВИЙ ЕЛЕМЕНТ ДЛЯ КОНСТРУКЦІЇ ІНШИХ





Truth Table

A	B	$Y = \overline{A \cdot B}$
0	0	1

[3.9]

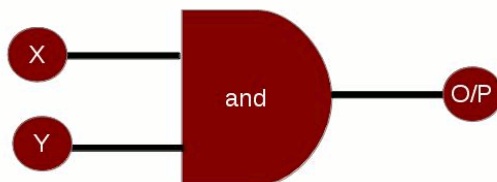
### ЛОГІЧНІ ОПЕРАЦІЇ (AND)



- ЛОГІЧНЕ І
  - 1 НА ВИХОДІ, КОЛИ НА ВСІХ ВХОДАХ 1
  - 0 НА ВИХОДІ, КОЛИ ХОЧА Б 1 ВХІД 0
- РЕАЛІЗАЦІЯ
  - NOT NAND
- ВИКОРИСТАННЯ
  - КЕРУВАННЯ ВИМИКАЧАМИ ДЛЯ ВВІМКНЕННЯ ПРИСТРОЇВ
  - ПЕРЕМНОЖЕННЯ СИГНАЛІВ
  - МНОЖЕННЯ І ДІЛЕННЯ ЧАСТОТИ
  - ВИДАЄ МІНІМУМ ВСІХ ВХІДНИХ СИГНАЛІВ

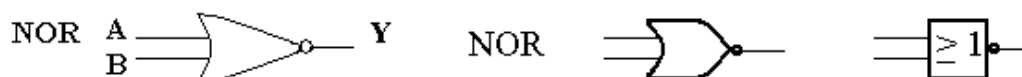
and Gate Animation

X	Y	O/P
0	0	0
0	1	0
1	0	0
1	1	1



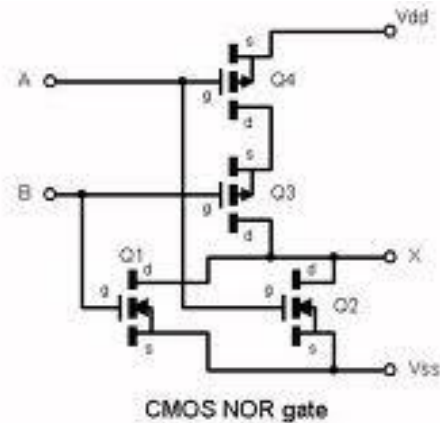
[3.10]

### ЛОГІЧНІ ЕЛЕМЕНТИ (NOR)



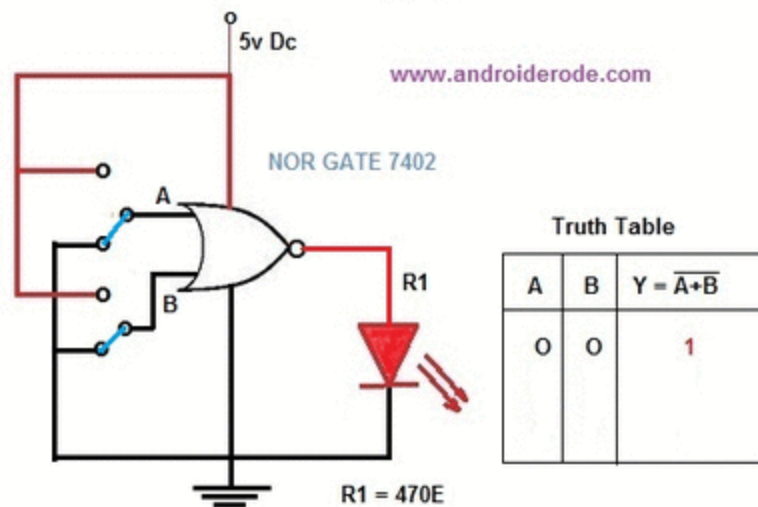
- АБО-НЕ

- ❑ НА ВИХОДІ 1 ЛИШЕ, КОЛИ НА ВСІХ ВХОДАХ 0
- ❑ КОЛИ ХОЧ НА ОДНОМУ ВХОДІ 1 НА ВИХОДІ 0



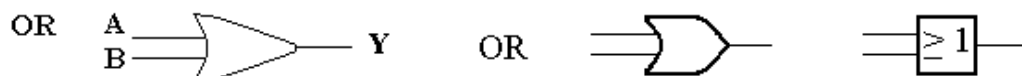
■ ЗАСТОСУВАННЯ

- ❑ БАЗОВИЙ ЕЛЕМЕНТ ДЛЯ СТВОРЕННЯ ІНШИХ



[3.11]

ЛОГІЧНІ ОПЕРАЦІЇ (OR)



■ ЛОГІЧНЕ ДОДАВАННЯ

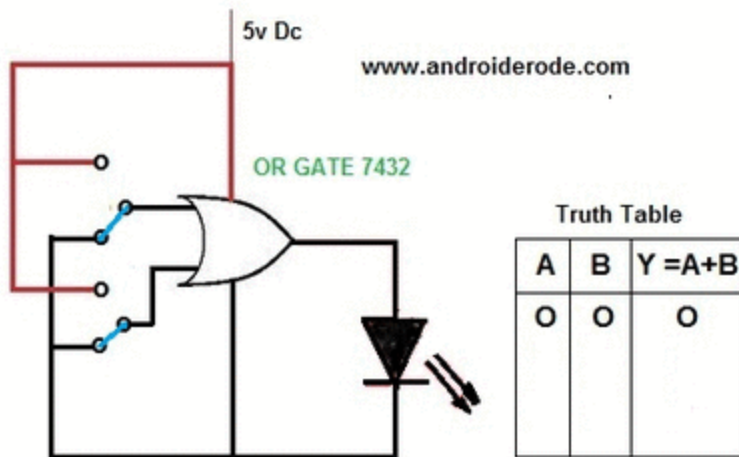
- ❑ 0 НА ВИХОДІ, ТІЛЬКИ КОЛИ НА ВСІХ ВХОДАХ 0
- ❑ 1 НА ВИХОДІ – КОЛИ ХОЧАБ НА ОДНОМУ ВХОДІ 1

■ РЕАЛІЗАЦІЯ

- ❑ NOR NOT

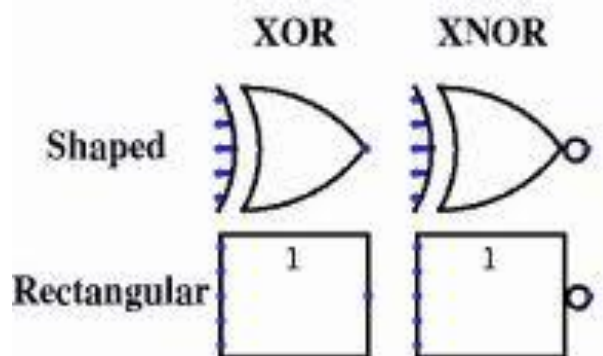
■ ЗАСТОСУВАННЯ

- ❑ КЕРУВАННЯ ВИМИКАЧАМИ
- ❑ ВИЗНАЧЕННЯ МАКСИМУМУ
- ❑ ДОДАВАННЯ СИГНАЛІВ



[3.12]

### ЛОГІЧНІ ОПЕРАЦІЇ XOR XNOR



- ВИКЛЮЧНЕ АБО
  - 1, КОЛИ НЕПАРНА КІЛЬКІСТЬ ОДНИЦЬ НА ВХОДАХ
  - 0 КОЛИ ПАРНА КІЛЬКІСТЬ ОДНИЦЬ НА ВХОДАХ
- РЕАЛІЗАЦІЯ
  - $A \text{ XOR } B = (A \text{ AND NOT } B) \text{ OR } (B \text{ AND NOT } A)$
- ВИКОРИСТАННЯ
  - ДОДАВАННЯ ЧИСЕЛ У ДВІЙКОВІЙ СИСТЕМІ
  - КОДУВАННЯ
  - ОБМІН МІЖ РЕГІСТРАМИ

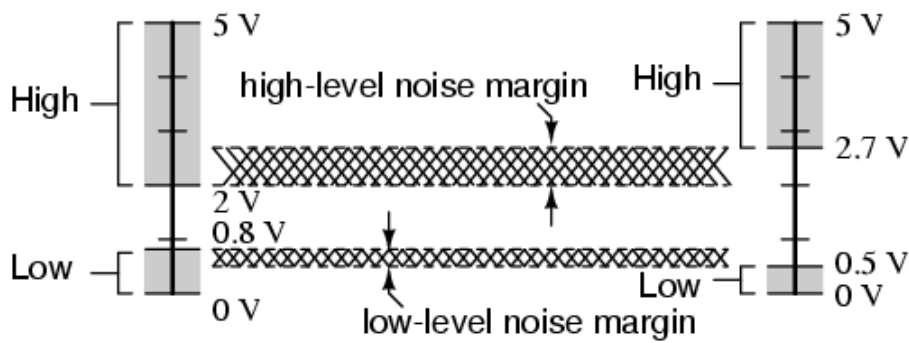
[3.13]

### ВИСОКИЙ І НИЗЬКИЙ РІВЕНЬ – РІЗНІ ЗНАЧЕННЯ В РІЗНИХ ТЕХНОЛОГІЯХ

- TTL (ТРАНЗИСТОРНО-ТРАНЗИСТОРНА ЛОГІКА)
  - НАПРУГА ЖИВЛЕННЯ 5В
  - ВИХІД
    - ВИСОКИЙ РІВЕНЬ 2.7-5 В
    - НИЗЬКИЙ РІВЕНЬ 0-0.5 В
  - ВХІД
    - ВИСОКИЙ РІВЕНЬ 2-5 В
    - НИЗЬКИЙ РІВЕНЬ 0-0.8 В
  - ВЕЛИКІ СТРУМИ СПОЖИВАННЯ – В МІКРОПРОЦЕСОРАХ НЕ ВИКОРИСТОВУЄТЬСЯ

Acceptable TTL gate input signal levels

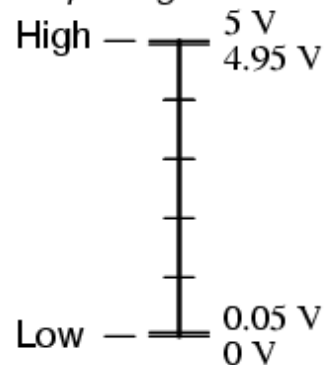
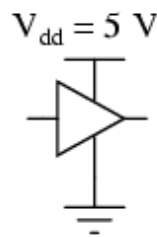
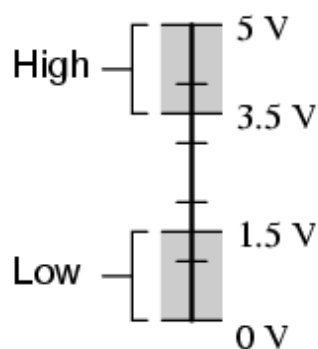
Acceptable TTL gate output signal levels



- CMOS (КМОП КОМПЛЕМЕНТАРНІ МОП)
  - КМОП 5 В
    - ВИСОКИЙ 3.5-5 В
    - НИЗЬКИЙ 0.05 – 1.5 В
  - КМОП 1.8-3.3 В
    - ВИСОКИЙ >1.1-2.5 В
    - НИЗЬКИЙ <0.35-0.5 В

Acceptable CMOS gate input signal levels

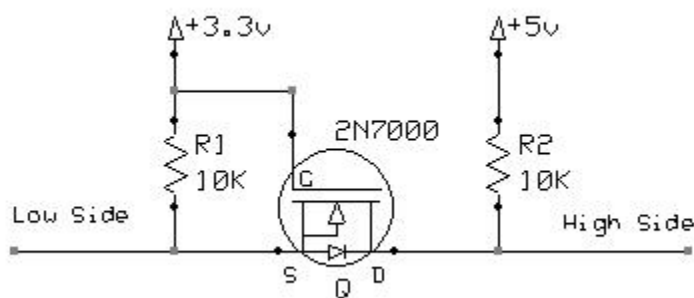
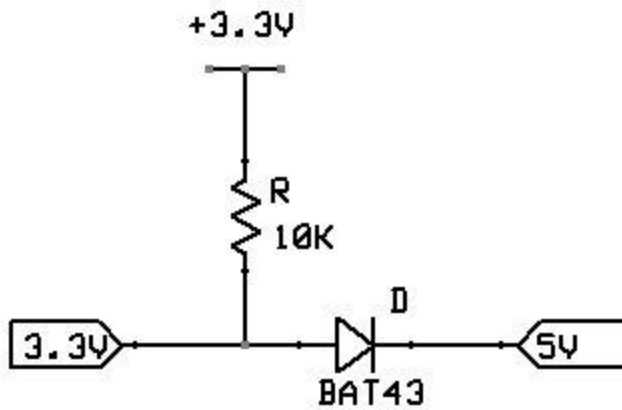
Acceptable CMOS gate output signal levels



[3.14]

ПЕРЕТВОРЮВАЧІ І ФОРМУВАЧІ РІВНІВ

- ПРИСТРОЇ 5 В МОЖУТЬ НЕ ПРАЦЮВАТИ НА 3.3
- ПРИСТРОЇ 3.3 МОЖУТЬ ЗГОРИТИ НА 5 В
- ДУЖЕ НЕ БАЖАНО НАПРЯМУ ПОДАВАТИ СИГНАЛ З МІКРОСХЕМ ОДНІЄЇ ТЕХНОЛОГІЇ НА ІНШУ!
- ТРЕБА ВИКОРИСТОВУВАТИ ПЕРЕТВОРЕННЯ РІВНІВ!
  - TTL<-> CMOS
  - 3.3<->5 В
  - 1.2<->15 В



[3.15]

#### ЕЛЕМЕНТИ ІЗ СТІЙКИМИ СТАНАМИ

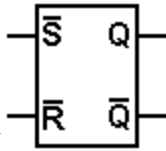
- ЛОГІЧНІ ЕЛЕМЕНТИ
  - СТАНИ НЕ СТІЙКІ
  - НЕ МОЖНА ЗАФІКСУВАТИ СТАН
- ЕЛЕМЕНТИ ПАМ'ЯТІ
  - СТІЙКІ СТАНИ
  - ВИХІД З ЯКОГОСЬ СТАНУ МОЖЛИВИЙ ШЛЯХОМ ПЕРЕХОДУ В ІНШИЙ СТАН
  - ПЕРЕХІД "МИТТЄВИЙ"
- ЕЛЕМЕНТИ ІЗ СТІЙКИМИ СТАНАМИ
  - ТРИГЕРИ
  - ЛІЧИЛЬНИКИ
  - РЕГІСТРИ
  - ЕЛЕМЕНТИ ПАМ'ЯТІ

[3.16]

#### ТРИГЕРИ (FLIP-FLOP, LATCH)

- ПРИСТРІЙ З ДВОМА СТАБІЛЬНИМИ СТАНАМИ
  - RS (RESET-SET, LATCH)
  - D (DELAY)
  - T (TOGGLE)
  - JK
- ПЕРЕКИДАННЯ МІЖ СТАНАМИ МИТТЄВЕ ЧЕРЕЗ ЗВОРОТНІЙ ЗВ'ЯЗОК

[3.17]



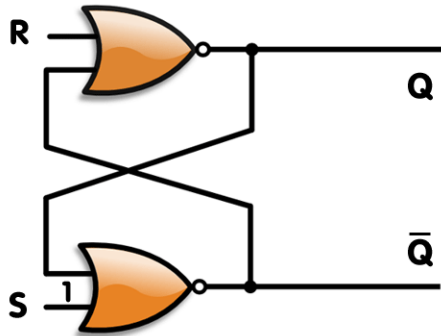
RS TA JK

- **RS**-ДВА ВХОДИ ДВА ІНВЕРСНІ ВИХОДИ
- ВСТАНОВЛЕННЯ (SET=1)
  - S=1 -> Q=1
  - R=0
- СКИДАННЯ (RESET=1)
  - R=1 -> Q=0
  - S=0
- ЗБЕРЕЖЕННЯ (УТРИМАННЯ, HOLD)
  - S=0, R=0, Q=CONST
- ЗАБОРОНЕНА КОМБІНАЦІЯ
  - S=1 R=1->Q-НЕВИЗНАЧЕНЕ

**JK** – МАЙЖЕ ТЕ Ж САМЕ, ЩО І RS  
ВІДМІННІСТЬ

J=1 K=1 -> ПЕРЕКИДАННЯ (FLIP-FLOP)

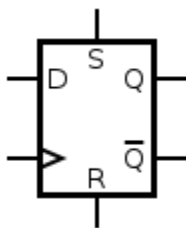
Q=NOT Q



[3.18]

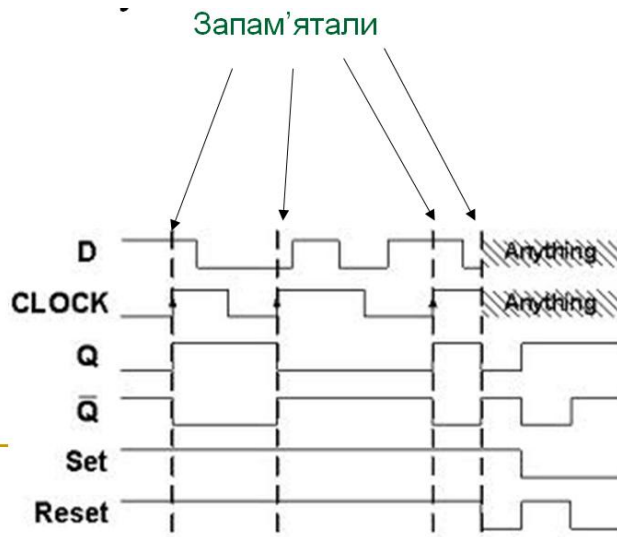
ТЕЖ САМЕ!!!

[3.19]



**D** ТРИГЕР (DELAY, DATA) FLIP-FLOP

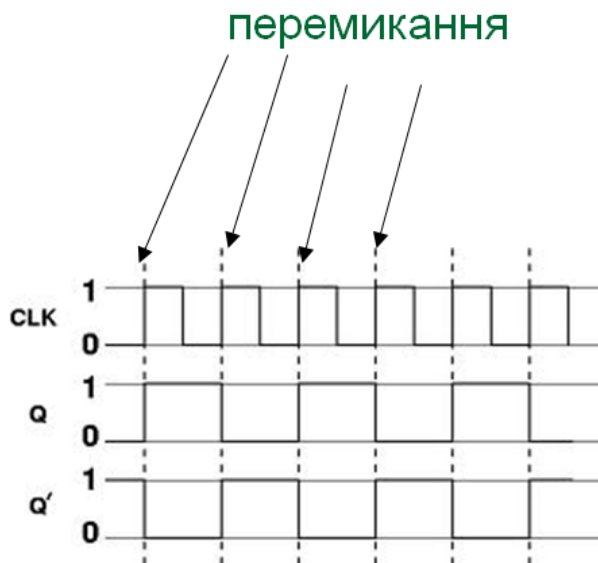
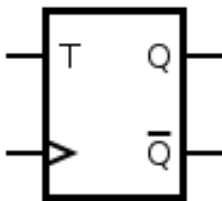
- ПЕРЕДАЄ ЗНАЧЕННЯ НА D ВХОДІ НА Q ВИХІД
- ЛИШЕ В МОМЕНТ ЗМІНИ СТАНУ ВХОДУ ">" , "CLOCK"
  - ЗА ФРОНОТОМ
  - ЗА СПАДОМ
- ОСНОВА
  - РЕГІСТРІВ
  - КОМІРОК ПАМ'ЯТІ



[3.20]

### T ТРИГЕР

- ТРИГЕР ПЕРЕМИКАННЯ (TOGGLE)
- ПЕРЕМИКАЄ СВІЙ СТАН НА ПРОТИЛЕЖНИЙ
  - ПО ФРОНТУ ІМПУЛЬСА НА ВХОДІ ">"
  - АБО ПО СПАДУ ІМПУЛЬСА НА ВХОДІ ">"
- ВХІД Т
  - $T=1$  ПЕРЕМИКАННЯ ДОЗВОЛЕНЕ
  - $T=0$  ПЕРЕМИКАННЯ НЕДОЗВОЛЕНЕ
- ОСНОВА
  - ЛІЧИЛЬНИКІВ
  - ПОДІЛЬНИКІВ ЧАСТОТИ



[3.21]

### ТРИГЕРИ З ЗАБОРОНОЮ (GATED LATCH)

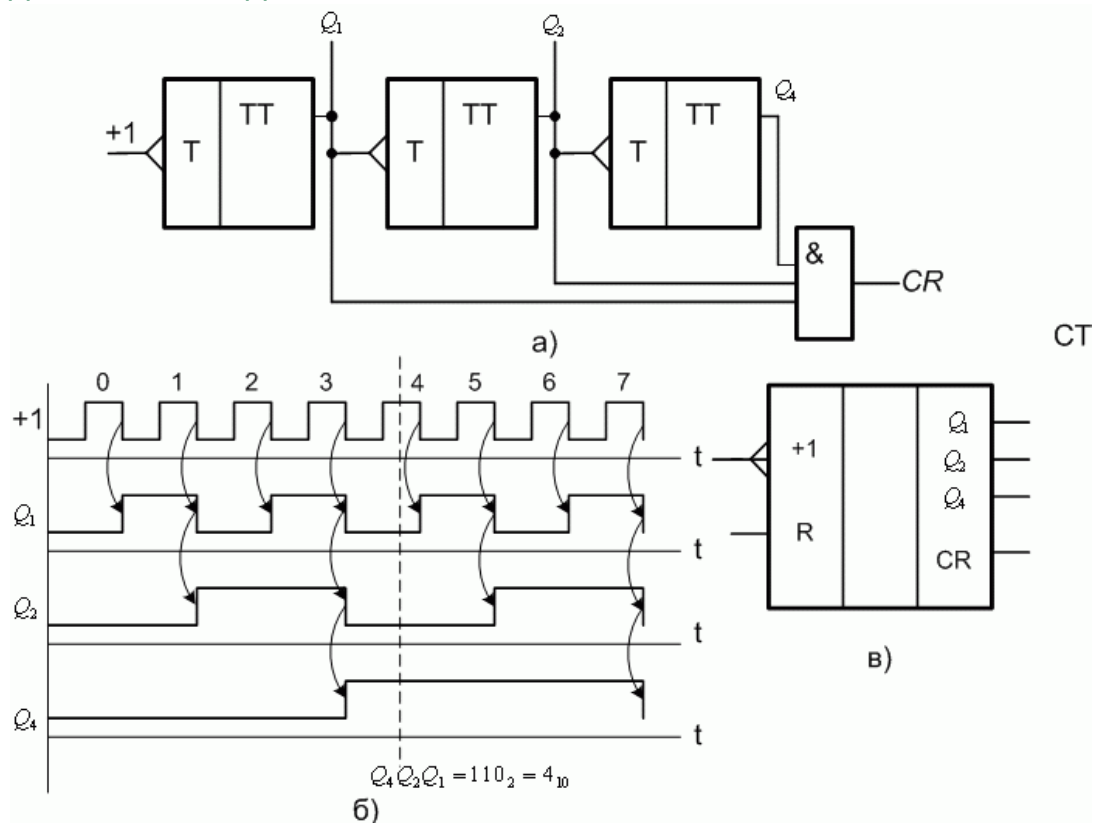
- SYNCHRONOUS RS, D, JK, T
- ЩЕ ОДИН ВХІД E
  - E=1 ПРИСТРІЙ ВИКОРИСТОВУЄТЬСЯ
  - E=0 ПРИСТРІЙ ЗБЕРІГАЄ ОСТАННІЙ СТАН

[3.22]

### ЛІЧІЛЬНИКИ, ПОДІЛЬНИКИ ЧАСТОТИ (COUNTER, DIVIDER)

- ЛІЧІЛЬНИКИ- ПРИСТРОЇ, НА ВИХОДІ ЯКИХ КІЛЬКІСТЬ ІМПУЛЬСІВ
- ПОДІЛЬНИК ЧАСТОТИ
  - НА ВИХОДІ ЧАСТОТА В N РАЗІВ МЕНША ЗА ВХІДНУ
- РЕАЛІЗАЦІЯ – ПОСЛІДОВНІСТЬ Т- ТРИГЕРІВ
- ТИПИ
  - СИНХРОННІ
  - АСИНХРОННІ

### ДВІЙКОВИЙ КОД КІЛЬКОСТІ ІМПУЛЬСІВ

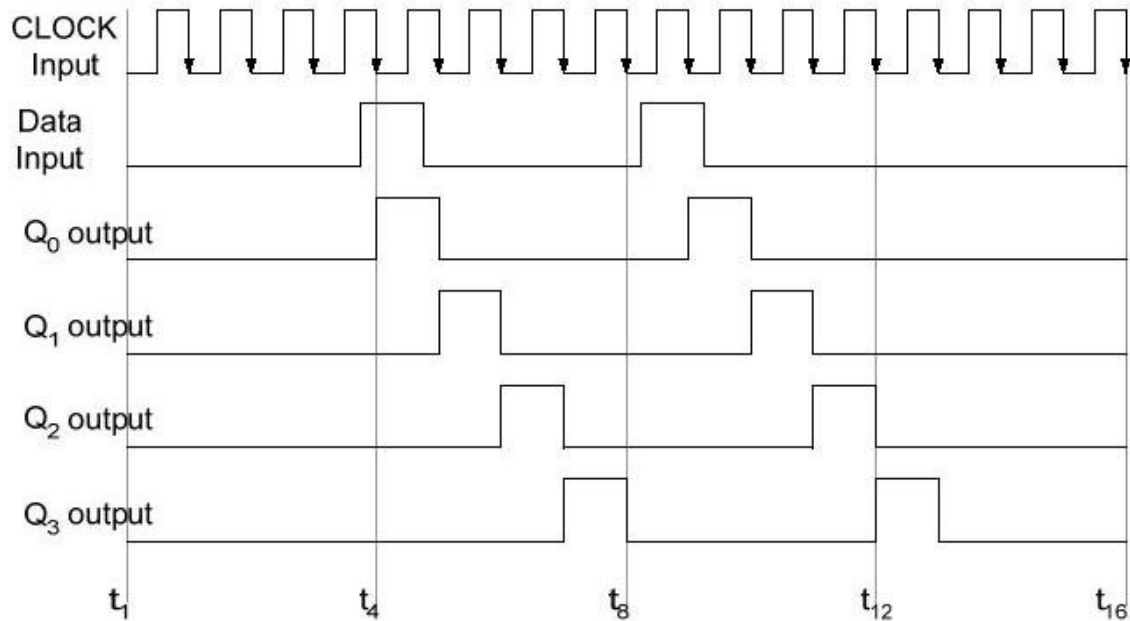
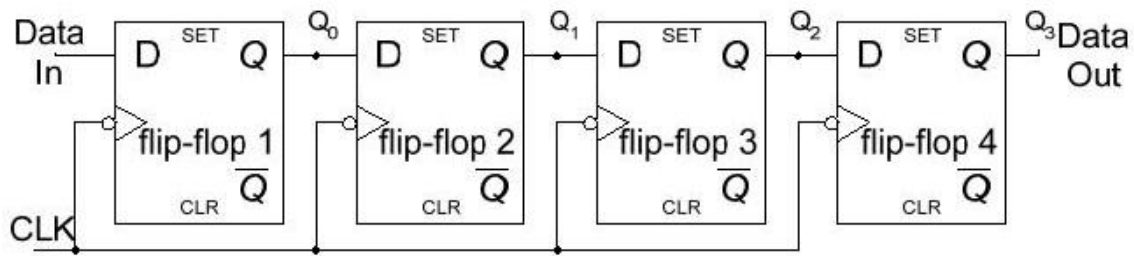


[3.23]

### РЕГІСТРИ

- ЗАПАМ'ЯТОВУЮЧИЙ ПРИСТРІЙ
  - ПАРАЛЕЛЬНІ
  - ЗСУВОВІ
- МНОЖЕННЯ ЧИСЕЛ

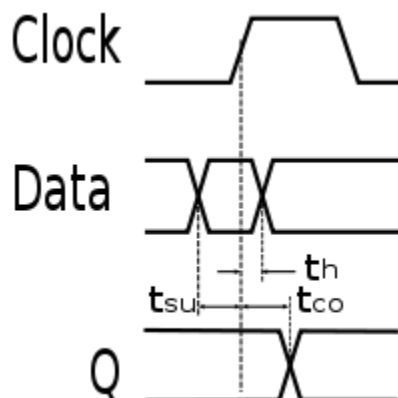




[3.24]

### ЧАСОВІ ПАРАМЕТРИ ПРИСТРОЇВ ІЗ СТІЙКИМИ СТАНАМИ

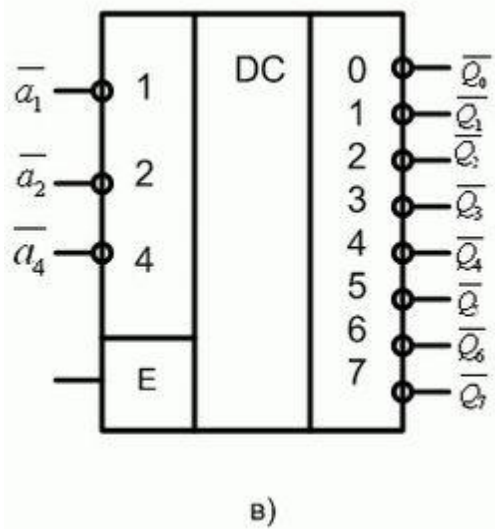
- ПЕРЕМИКАННЯ ТРИГЕРІВ ВИМАГАЄ ЧАСУ
- ЦЕ ВИЗНАЧАЄ ШВИДКІСТЬ РОБОТИ ПРОЦЕСОРНИХ ПРИСТРОЇВ
- ПАРАМЕТРИ
  - ЧАС ВСТАНОВЛЕННЯ ( $t_{su}$ ) – СКІЛЬКИ ТРЕБА ТРИМАТИ ВХІДНИЙ СИГНАЛ, ДО ТАКОВОГО ІМПУЛЬСУ, ЩОБ ТРИГЕР СПАРЦЮВАВ
  - ЧАС УТРИМАННЯ ( $t_{hu}$ ) – СКІЛЬКИ ТРЕБА ТРИМАТИ ВХІДНИЙ СИГНАЛ, ДО ТАКОВОГО ІМПУЛЬСУ, ЩОБ ТРИГЕР СПАРЦЮВАВ
  - ЧАС ЗАТРИМКИ ( $t_{co}$ ) ЧАС МІЖ ВСТАНОВЛЕННЯМ ВХОДУ І ЗМІНОЮ ВИХОДУ
  - ЧАС НАРОСТАННЯ ФРОНТУ



[3.25]

## ШИФРАТОРИ-ДЕШИВРАТОРИ

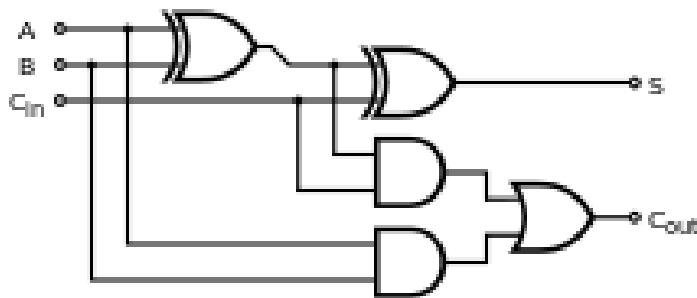
- ПРИСТОЇ, ЯКІ ОДНОМУ ЧИСЛУ СТАВЛЯТЬ У ВІДПОВІДНОСТЬ ІНШЕ ЗА ПЕВНИМ ЗАКОНОМ
  - ДВІЙКОВЕ-ДЕСЯТКОВЕ
  - ДВІЙКОВЕ-СЕМИСЕГМЕННИЙ ІНДИКАТОР



[3.26]

## АРИФМЕТИЧНІ ОПЕРАЦІЇ

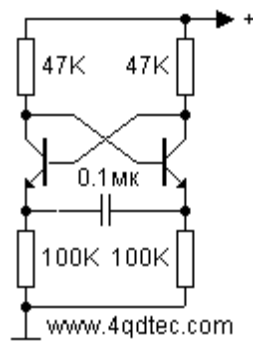
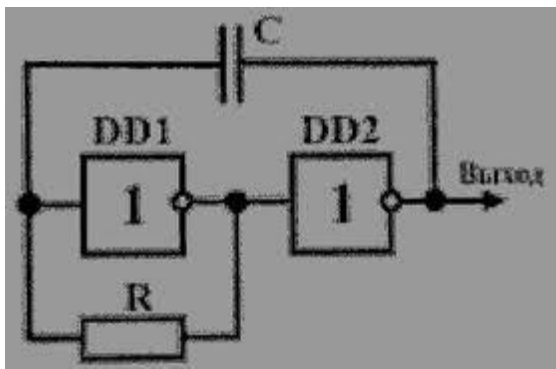
- ДОДАВАННЯ
  - СУМАТОР (ADDER)
  - ВХОДИ A B CIN(CARRY)
  - ВИХОДИ S COUT(CARRY)
- МНОЖЕННЯ
  - ЛОГІЧНИЙ ЗСУВ
  - БІНАРНЕ МНОЖЕННЯ



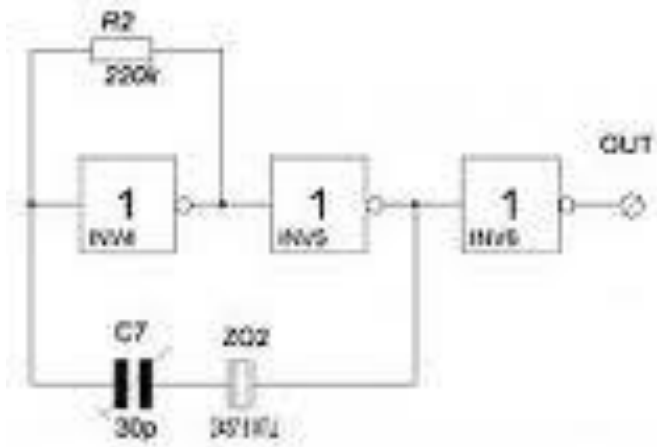
[3.27]

## ГЕНЕРАТОРИ КОЛИВАНЬ

- СИСТЕМНИЙ ГОДИННИК ВАЖЛИВА ЧАСТИНА ПРОЦЕСОРНОЇ СИСТЕМИ
  - СТАБІЛЬНА ЧАСТОТА
  - МОЖЛИВІСТЬ ПЕРЕСТРОЮВАННЯ ЧАСТОТИ



[www.4qdtex.com](http://www.4qdtex.com)



## ЛЕКЦІЯ 04

[4.1]

СКЛАДОВІ КОМПОНЕНТИ МІКРОПРОЦЕСОРІВ [2.2]  
ЦИФРОВІ ІНФОРМАЦІЙНІ ПРИСТРОЇ

### ЛЕКЦІЯ 4

СУДАКОВ О.О, РАДЧЕНКО С.П.

«СУЧАСНА МІКРОПРОЦЕСОРНА ТЕХНІКА»

[4.2]

#### КОМБІНАЦІЙНІ ТА ПОСЛІДОВНОСНІ СХЕМИ

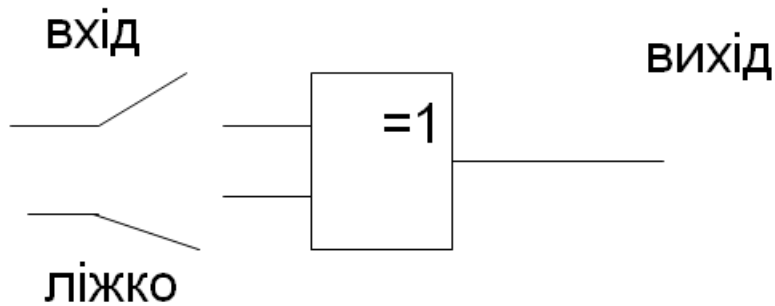
- КОМБІНАЦІЙНА – РЕЗУЛЬТАТ НА ВИХОДІ ЗАЛЕЖИТЬ ВІД ПОТОЧНОГО ЗНАЧЕННЯ ПАРАМЕТРІВ НА ВХОДАХ
  - КОМУТАТОРИ
  - ЛОГІЧНІ СХЕМИ
  - ГЕНЕРАТОРИ
  - ШИФРАТОРИ -ДЕШИФРАТОРИ
- ПОСЛІДОВНОСНА – РЕЗУЛЬТАТ НА ВИХОДІ ЗАЛЕЖИТЬ ВІД ВХОДІВ, ЧАСУ, СТАНУ
  - ТРИГЕРИ
  - ЛІЧИЛЬНИКИ
  - РЕГІСТРИ
  - ПАМ'ЯТЬ
  - ПРОЦЕСОРИ

[3.3]

#### КОМБІНАЦІЙНА СХЕМА (ВИМИКАЧ СВІТЛА)

- В КІМНАТІ 2 ВИМИКАЧІ
  - 1-Й БІЛЯ ВХОДУ
  - 2-Й БІЛЯ ЛІЖКА
- ЗРОБИТИ СХЕМУ ЯКА ВМИКАЄ І ВИМИКАЄ СВІТЛО В КІМНАТІ
  - ВМИКАЄШ ПРИ ВХОДІ
  - ВИМИКАЄШ З ЛІЖКА
  - ВМИКАЄШ З ЛІЖКА
  - ВИМИКАЄШ ПРИ ВХОДІ

ВХІД	ЛІЖКО	ВИХІД
0	0	0
1	0	1
0	1	1
1	1	0

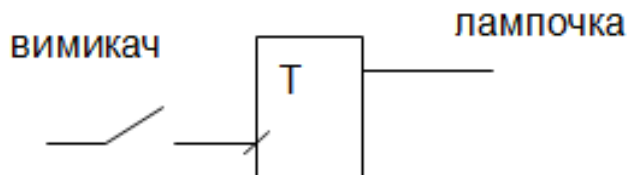


ВИХІД= (ВХІД AND NOT ЛІЖКО) OR  
(NOT ВХІД AND ЛІЖКО)=ВХІД XOR ЛІЖКО

[4.4]

#### ПОСЛІДОВНІСНА СХЕМА

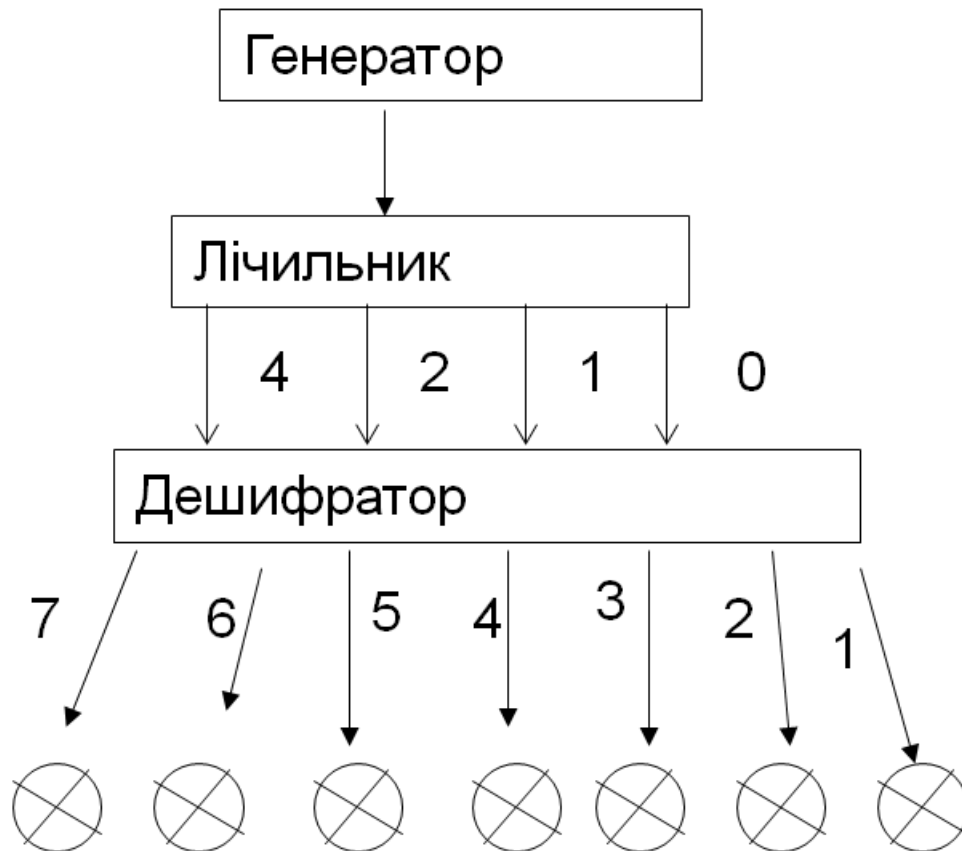
- ОДИН ВИМИКАЧ І ОДНА ЛАМПОЧКА
  - ВИМИКАННЯ ВИМАКАЧА НІЧОГО НЕ МІНЯЄ
  - ВМИКАННЯ ВИМИКАЧА ВИМИКАЄ ЛАМПОЧКУ, КОЛИ ВОНА ГОРИТЬ І ВМИКАЄ – КОЛИ НЕ ГОРИТЬ



[4.5]

#### БІЖУЧІ ВОГНІ

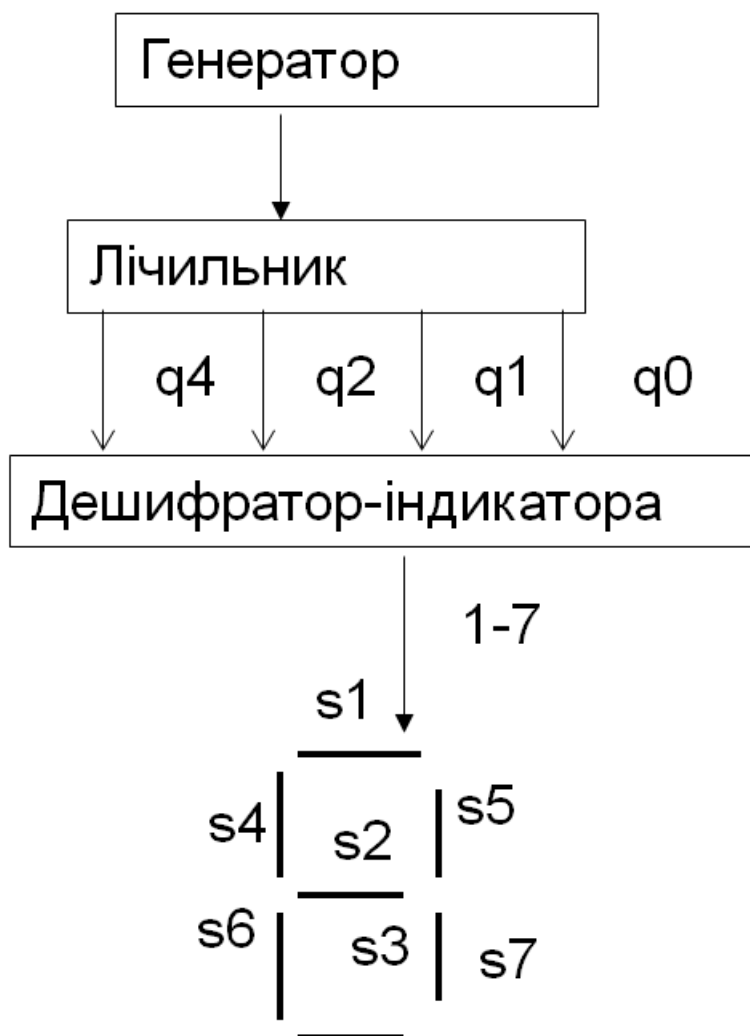
- СИСТЕМА З ПОСЛІДОВНИМ ВИКОНАННЯМ
- ПРИМІТИВНИЙ ПРОЦЕСОР
  - ЛІЧИЛЬНИК РАХУЄ ІМПУЛЬСИ ГЕНЕРАТОРА
  - ВИДАЄ НА ВИХОДІ ДВІЙКОВИЙ КОД
  - ДЕШИФРАТОР ЗАПАЛЮЄ ЛАМПУ З ВІДПОВІДНИМ НОМЕРОМ
- КОМПОНЕНТИ
  - ГЕНЕРАТОР – СИСТЕМНИЙ ГОДИННИК (SC)
  - ЛІЧИЛЬНИК-ПРОГРАМНИЙ ЛІЧИЛЬНИК (PC)
  - ДЕШИФРАТОР – ПАМ'ЯТЬ КОМАНД
  - ЛАМПОЧКИ- ВИКОНУВАЛЬНИЙ ПРИСТРІЙ
- ВИХІД ЛІЧИЛЬНИКА
  - ШИНА АДРЕС
  - АДРЕСА КОМАНДИ



[4.6]

#### ДЕШИФРАТОР

- ДЕШИФРАТОР – КОМБІНАЦІЙНА СХЕМА
  - КОЖНОМУ ВХІДНОМУ КОДУ СПІВСТАВЛЯЄТЬСЯ ВИХІДНИЙ КОД
  - АНАЛОГ ПРОГРАМИ І ПАМ'ЯТІ
- КОЖНІЙ АДРЕСІ МОЖНА СПІВСТАВИТИ
  - ПРИСТРІЙ
  - КОМАНДУ
  - ДАНІ
- ПОСТІЙНА ПАМ'ЯТЬ



0: S1 & S4 & S6 & S3 & S7 & S5

S1= (NOT Q0 & NOT Q1 & NOT Q2 & NOT Q4) OR ..

S2=

[4.7]

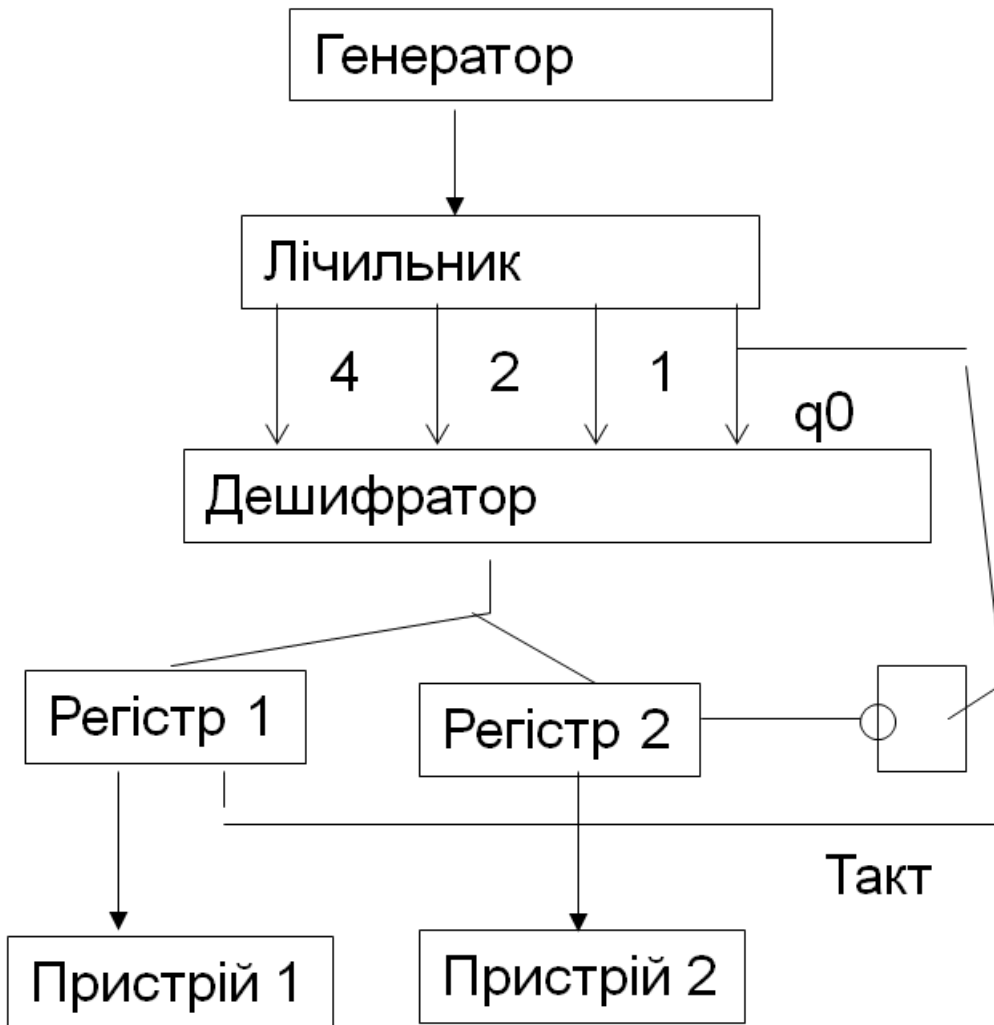
#### КЕРУВАННЯ ДЕКІЛЬКОМА ПРИСТРОЯМИ

##### ■ ЗАДАЧА

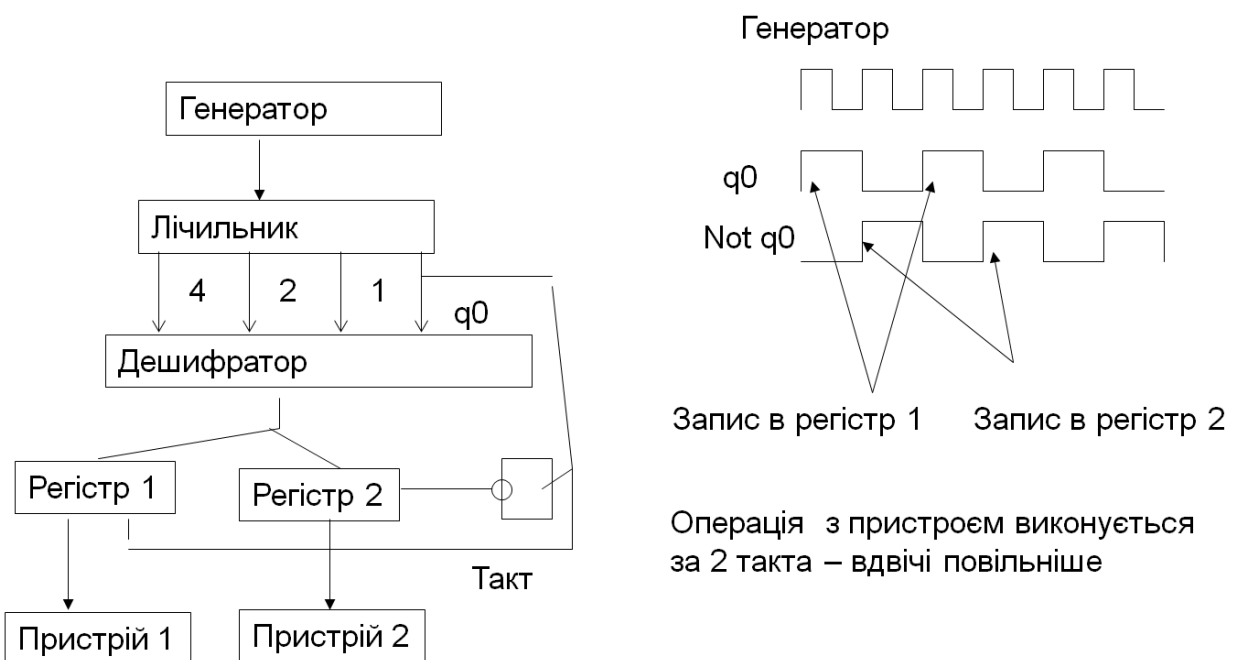
- ПРИСТРІЙ 1 – КОЖНА ПАРНА КОМАНДА
- ПРИСТРІЙ 2 – КОЖНА НЕПАРНА КОМАНДА

##### ■ РЕАЛІЗАЦІЯ

- ДЕШИФРАТОР ВИДАЄ НА ВИХІД КОМАНДИ
- РЕГІСТР 1 ЗАПАМ'ЯТУЄ НЕПАРНІ КОМАНДИ
- РЕГУСТР 2 ЗАПАМ'ЯТУЄ ПАРНІ КОМАНДИ



[4.8]  
РОБОТА СХЕМИ





[4.9]

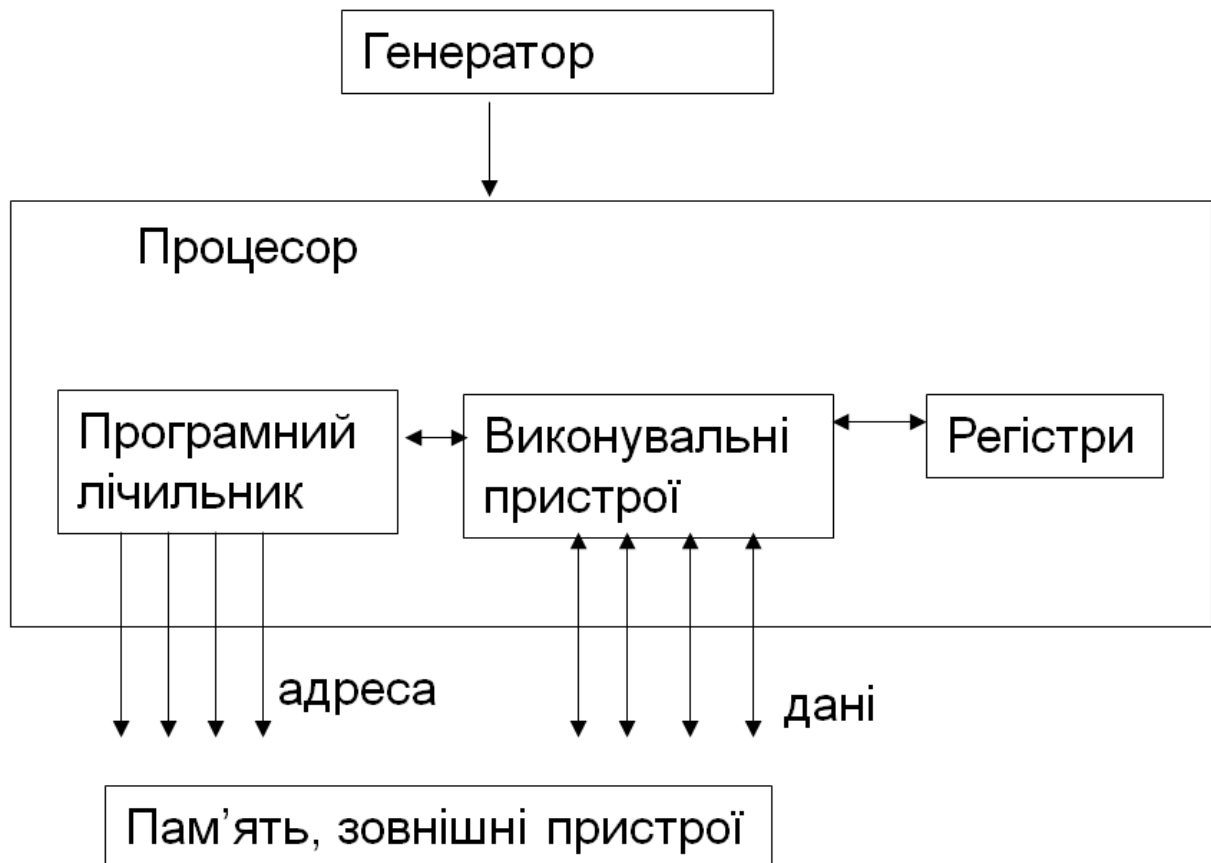
#### ЗАСТОСУВАННЯ РЕГІСТРІВ

- ПЕРЕДАЧА ДАНИХ РІЗНИМ ПРИСТРОЯМ ПО ОДНИМ ПРОВОДАМ
  - ТАКТ1 – ВИДАЧА ДАНИХ І ЗАПИС В РЕГІСТР ПРИСТРОЮ 1
  - ТАКТ2 – ВИДАЧА ДАНИХ І ЗАПИС В РЕГІСТР ПРИСТРОЮ 2
  
- ВИКОНАННЯ МАТЕМАТИЧНИХ ОПЕРАЦІЙ З ДЕКІЛЬКОМА ПОРЦІЯМИ ДАНИХ
  - ТАКТ1 – ВИДАЧА ДАНИХ1 І ЗАПИС В РЕГІСТР1
  - ТАКТ2 - ВИДАЧА ДАНИХ2 І ЗАПИС В РЕГІСТР 2
  - ТАКТ3 ДАНІ1+ДАНІ2 І ЗАПИС В РЕГІСТР 3
  - ВИКОНАННЯ КОМАНДИ – 3 ТАКТИ
  
- ОПЕРАТИВНА ПАМ'ЯТЬ
  - ТАКТ 1 - ПО ПРОВОДАМ АДРЕСИ – ДОЗВІЛ ЗАПИСУ-ЧИТАННЯ В ПЕВНИЙ РЕГІСТР
  - ТАКТ 1 ПО ПРОВОДАМ КЕРУВАННЯ – КЕРУВАННЯ ЗАПИСОМ-ЧИТАННЯМ
  - ТАКТ 1 ПО ПРОВОДАМ ДАНИХ – ПЕРЕДАЧА, АБО ПРИЙОМ ДАНИХ
  - ТАКТ 2 ПО ПРОВОДАМ ТАКТУВАННЯ – ЗАПИС АБО ЧИТАННЯ
  
- ЗБІЛЬШЕННЯ КІЛЬКОСТІ ФУНКЦІЙ ЗА РАХУНОК ЗМЕНШЕННЯ ШВИДКОСТІ

[4.10]

#### СХЕМА ПРОЦЕСОРА

- ПРОЦЕСОР
  - ПРОГРАМНИЙ ЛІЧИЛЬНИК
  - ВИКОНУВАЛЬНІ ПРИСТРОЇ (АЛП)
  - РЕГІСТРИ
  
- ЯК ПРАЦЮЄ
  - НА ШИНУ АДРЕСИ ВИДАЄТЬСЯ АДРЕСА КОМАНДИ
  - ПАМ'ЯТЬ ВИДАЄ КОМАНДУ ЧИ ДАНІ ЗА ЦІЮ АДРЕСОЮ
  - ВИКОНУВАЛЬНИЙ ПРИСТРІЙ ВИКОНУЄ РОБОТУ
  - РЕЗУЛЬТАТИ ЗАПИСУЮТЬСЯ РЕГІСТРИ АБО В ПАМ'ЯТЬ



[4.11]

АРИФМЕТИКО-ЛОГІЧНИЙ ПРИСТРІЙ (АЛУ, ALU, ПРИСТРІЙ КЕРУВАННЯ, ФУНКЦІОНАЛЬНИЙ ПРИСТРІЙ)

- КЕРУВАННЯ РОБОТОЮ І ВИКОНАННЯ ФУНКЦІЙ
  - ДЕШИФРАТОР КОМАНД
  - БУФЕРНІ РЕГІСТРИ (ДЛЯ ТИМЧАСОВОГО ЗБЕРІГАННЯ КОМАНД І ДАНИХ)
  - АРИФМЕТИЧНІ ПРИСТРОЇ (ДОДАВАННЯ, ВІДНІМАННЯ, МНОЖЕННЯ)
- РОЗРЯДНІСТЬ АЛУ
  - МАКСИМАЛЬНА КІЛЬКІСТЬ БІТІВ ЯКІ МОЖУТЬ БУТИ ОБРОБЛЕНІ ЗА ОДИН ТАКТ
  - 2,4,8,16,32,64
- СИСТЕМА КОМАНД
  - ЯКІ КОМАНДИ МОЖЕ ВИКОНАТИ ПРОЦЕСОР
  - ADD, MUL, AND, XOR, NOR...

[4.12]

РЕГІСТРИ ПРОЦЕСОРА

- **ПРОГРАМНИЙ ЛІЧИЛЬНИК** (РЕГІСТР АДРЕСИ, ВКАЗІВНИК ІНСТРУКЦІЇ, INSTRUCTION POINTER, IP, PROGRAM COUNTER, PC)
  - АДРЕСА НАСТУПНОЇ КОМІРКИ ПАМ'ЯТІ
- **РЕГІСТРИ ЗАГАЛЬНОГО ПРИЗНАЧЕННЯ** (GPR, РЗП)
  - A, B, C, D - АРГУМЕНТИ І РЕЗУЛЬТАТИ ВИКОНАННЯ КОМАНД
- **АКУМУЛЯТОРИ** (A)
  - РЕГІСТРИ, ЯКІ ПО ЗАМОВЧАННЮ ВИКОРИСТОВУЮТЬСЯ ДЛЯ ОПЕРАЦІЇ
  - КОМАНДА **A=A ADD B** ВИГЛЯДАЄ ПРОСТІШЕ **ADD B**

- **РЕГІСТР СТАНУ** (STATUS REGISTER, SR, РЕГІСТР ПРАПОРЦІВ, ОЗНАК)
  - ЧИ БУЛО ПЕРЕНЕСЕННЯ, ЧИ БУВ РЕЗУЛЬТАТ ОПЕРАЦІЇ 0, ...
- **ІНДЕКСНІ РЕГІСТРИ**
  - СПРОЩЕННЯ АДРЕСАЦІЇ ПАМ'ЯТІ
- **РЕГІСТРИ СТЕКУ** (МАГАЗИННОЇ ПАМ'ЯТІ, BP, SP)
  - СПРОЩЕННЯ АДРЕСАЦІЇ ПАМ'ЯТІ

[4.13]

#### СИСТЕМА КОМАНД ПРОЦЕСОРА

- **СИСТЕМА КОМАНД** (НАБІР ІНСТРУКЦІЙ)
  - ЯКІ ДІЇ МОЖЕ ВИКОНУВАТИ ПРОЦЕСОР
    - АРИФМЕТИЧНІ ОПЕРАЦІЇ
    - ЛОГІЧНІ ОПЕРАЦІЇ
    - ЗАПИС-ЗЧИТУВАННЯ ДАНИХ З ПАМ'ЯТІ ЧИ ІНШИХ ПРИСТРОЇВ
    - ВИКЛИК ПІДПРОГРАМ ...
  - НАБІР КОМАНД ЖОРСТКО "ЗАШИТИЙ" В ДЕШИФРАТОРІ КОМАНД ПРОЦЕСОРА
  - КОЖНА КОМАНДА МАЄ МАШИННИЙ КОД (УНІКАЛЬНУ БІТОВУ КОМБІНАЦІЮ)
  - КОМАНДИ ЗБЕРІГАЄТЬСЯ В ЗОВНІШНІЙ ПАМ'ЯТІ І ЗАВАНТАЖУЮТЬСЯ В ПРОЦЕСОР
  - ДЕЯКІ КОМАНДИ МАЮТЬ АРГУМЕНТИ, ТОДІ ТРЕБА ЗАВАНТАЖИТИ КОМАНДУ І АРГУМЕНТИ
  - МОВА АСЕМБЛЕРА – КОЖНА КОМАНДА ПРОЦЕСОРА – ОДНА ОКМАНДА МОВИ
- РІЗНІ КОМАНДИ ВИКОНУЮТЬСЯ ВПРОДОВЖ РІЗНОГО ЧАСУ
  - **КОМАНДНИЙ ЦИКЛ** – ПОСЛІДОВНІСТЬ ДІЙ ВИКОНАННЯ КОМАНДИ
  - **МІКРОКОМАНДА** – ЕТАП ВИКОНАННЯ КОМАНДИ
  - РЕГІСТР-РЕГІСТР – ШВИДКІ КОМАНДИ (МАЛО ОБМІНУ ІЗ ЗОВНІШНІМИ ПРИСТРОЯМИ)
  - РЕГІСТР-ПАМ'ЯТЬ/ ПРИСТРІЙ – ПОВІЛЬНІШІ КОМАНДИ (БІЛЬШЕ ОБМІНУ ІЗ ЗОВНІШНІМИ ПРИСТРОЯМИ)
  - КОМАНДИ БЕЗ АРГУМЕНТІВ – ШВИДШІ
  - ЧИМ БІЛЬШЕ АРГУМЕНТІВ (1,2,3...) ТИМ ПОВІЛЬШІШІ

[4.14]

#### ШИНА (BUS)

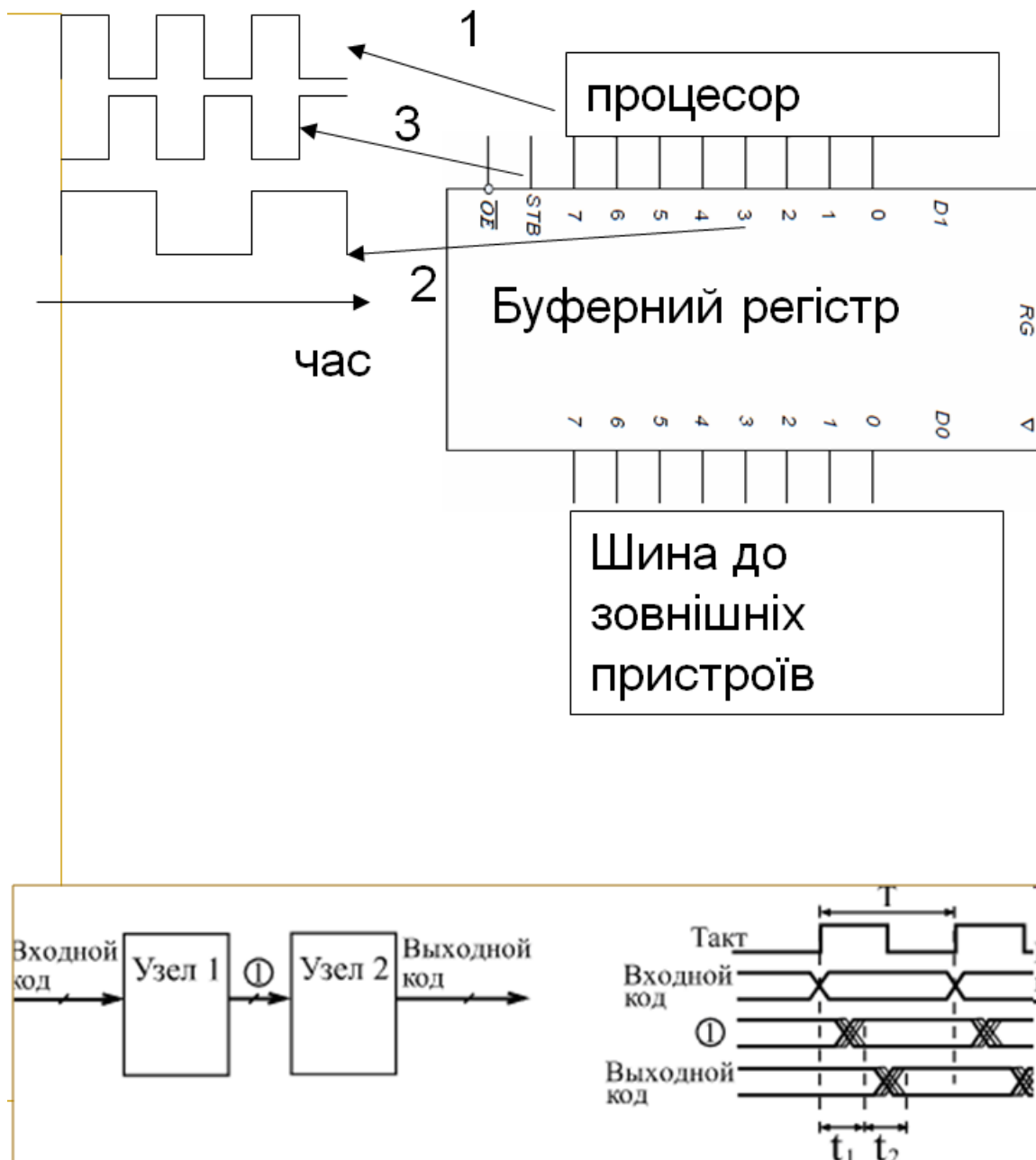
- **ШИНА** - НАБІР ПРОВОДІВ СПІЛЬНОГО ПРИЗНАЧЕННЯ
  - ШИНА АДРЕСИ – ПЕРЕДАЧА АДРЕСИ КОМАНДИ ЧИ ДАНИХ В ПАМ'ЯТІ
  - ШИНА ДАНИХ – ПЕРЕДАЧА САМОЇ КОМАНДИ ЧИ ДАНИХ
  - ШИНА КЕРУВАННЯ – КЕРУВАННЯ ПРИСТРОЯМИ
  - ВНУТРІШНІ ШИНИ ПРОЦЕСОРА
- ДО ШИН ПРИЄДНАНО БАГАТО ПРИСТРОЇВ
  - ВНУТРІШНІ ПРИСТРОЇ ПРОЦЕСОРА
  - ПАМ'ЯТЬ
  - ЗОВНІШНІ ПРИСТРОЇ
- РОЗРЯДНІСТЬ ШИНИ – КІЛЬКІСТЬ ПРОВОДІВ (БІТІВ)
- КОНТРОЛЕР ШИНИ – ПРИСТРІЙ ЯКИЙ КЕРУЄ ОБМІНОМ ПО ШИНИ
  - БУФЕРНИЙ РЕГІСТР – РЕГІСТР ЗАПИСУ-ЗЧИТУВАННЯ ДАНИХ НА ШИНИ
  - ШИННИЙ ФОРМУВАЧ – ПОРТ З ВИСОКОЮ НАВАНТАЖУВАЛЬНОЮ ЗДАТНІСТЮ
- ШИНИ МАЮТЬ СВОЇ ПРОТОКОЛИ ПЕРЕДАЧІ ІНФОРМАЦІЇ
  - ПРОТОКОЛ – НАБІР ПРАВИЛ, ЯКИМИ КЕРУЮТЬ ПРИСТРОЇ ДЛЯ ПРАВИЛЬНОГО ОБМІНУ (1 ПЕРЕДАЄ – ІНШІ ПРИЙМАЮТЬ ТОЩО)

[4.15]

КОНТРОЛЕР ШИНИ, БУФЕРНІ РЕГІСТРИ, ФОРМУВАЧІ

- БАГАТО ПРИСТРОЇВ ПІДКЛЮЧЕНІ ДО ШИНИ
  - ПРОЦЕСОР НЕ ВИТРИМАЄ ТАКИЙ СТРУМ
  - МОЖУТЬ БУТИ ЗАТРИМКИ ЧЕРЕЗ РІЗНІ ШВИДКІСТЬ
- БУФЕРНИЙ РЕГІСТР
  - ВИСОКА НАВАНТАЖУВАЛЬНА СПРОМОЖНІСТЬ
- ЗАПИС-ЗЧИТУВАННЯ ДАНИХ
  - ТАКТОВИЙ ІМПУЛЬС 1 – ВИСТАВЛЕННЯ ДАНИХ ПРОЦЕСОРОМ НА ШИНУ
  - ДАНІ ДОСТУПНІ ВПРОДОВЖ ТАКТУ 2
  - ТАКТОВИЙ ІМПУЛЬС 3 (ІНВЕРСНИЙ ЧЕРЕЗ ПІВПЕРІОДА) – ЗАПИС В РЕГІСТР І НА ШИНУ
  - ДАНІ ПЕРЕДАЮТЬСЯ З ЗАТРИМКОЮ НА 1 ТАКТ
  - ВІДКЛЮЧЕННЯ - ПЕРЕВЕДЕННЯ В СТАН ВИСОКОГО ВИХІДНОГО ОПОРУ (HOLD, УТРИМАННЯ)
- Є ДВОНАПРАВЛЕНІ, ПРОГРАМОВАНІ

## Тактові імпульси



[4.16]

### РОЗРЯДНІСТЬ

- ШИНА АДРЕСИ – НАЙБІЛЬША РОЗРЯДНІСТЬ
  - 8,16,32,64,128
- ШИНА ДАНИХ
  - 8,16,32,64
- ШИНА КЕРУВАННЯ
  - ЧАСТО СУМІЩЕНА З ДАНИМИ ЧИ АДРЕСОЮ
- РОЗРЯДНІСТЬ ПРОЦЕСОРА
  - РОЗРЯДНІСТЬ ШИНИ ДАНИХ 8,16,32
  - РОЗМІР МАШИНОГО СЛОВА – ПОРЦІЯ ДАНИХ З ЯКОЮ ЕФЕКТИВНО МОЖЕ ПРАЦЮВАТИ ПРОЦЕСОР 8,16,32,64
  - МАКСИМАЛЬНА АДРЕСОВАНА ОБЛАСТЬ – РОЗРЯДНІСТЬ ЛІЧИЛЬНИКА КОМАД  $2^8, 2^{16}, 2^{32}, 2^{64}$
  - МАКСИМАЛЬНИЙ ОБ'ЄМ ПАМ'ЯТІ  $2^8, 2^{16}, 2^{32}, 2^{64}, 2^{128}$
  - БАЙТ – ОДИНИЦЯ ПЕРЕДАЧІ, АДРЕСАЦІЇ І ЗБЕРЕЖЕННЯ ДАНИХ

[4.17]

#### ПРИКЛАДИ КОМАНД ПРОЦЕСОРА

- БЕЗ АРГУМЕНТІВ
  - HLT –ЗУПИНИТИ РОБОТУ
  - MOV A,B ПЕРЕСИЛКА З РЕГІСТРА В В А (A=B)
  - ADD B ДОДАТИ ДО АКУМУЛЯТОРА ВМІСТ РЕГІСТРА В
- З ОДНИМ АРГУМЕНТОМ
  - MVI A,<B> ПЕРЕСИЛКА ЗНАЧЕННЯ <B> В РЕГІСТР А (A=<B>)
- З ОДНИМ АРГУМЕНТОМ ВЕЛИКОЇ ДОВЖИНИ
  - LDA AB ЗАВАНТАЖИТИ АКУМУЛЯТОР ЗНАЧЕННЯМ В ОБЛАСТІ ПАМ'ЯТІ З АДРЕСОЮ АВ
  - CALL AB ВИКЛИКАТИ ПІДПРОГРАМУ ЗА АДРЕСОЮ ОБЛАСТІ ПАМ'ЯТІ АВ
- КОЖНА ВИМАГАЄ ПЕВНУ КІЛЬКІСТЬ МАШИННИХ ТАКТІВ І ЦИКЛІВ

[4.18]

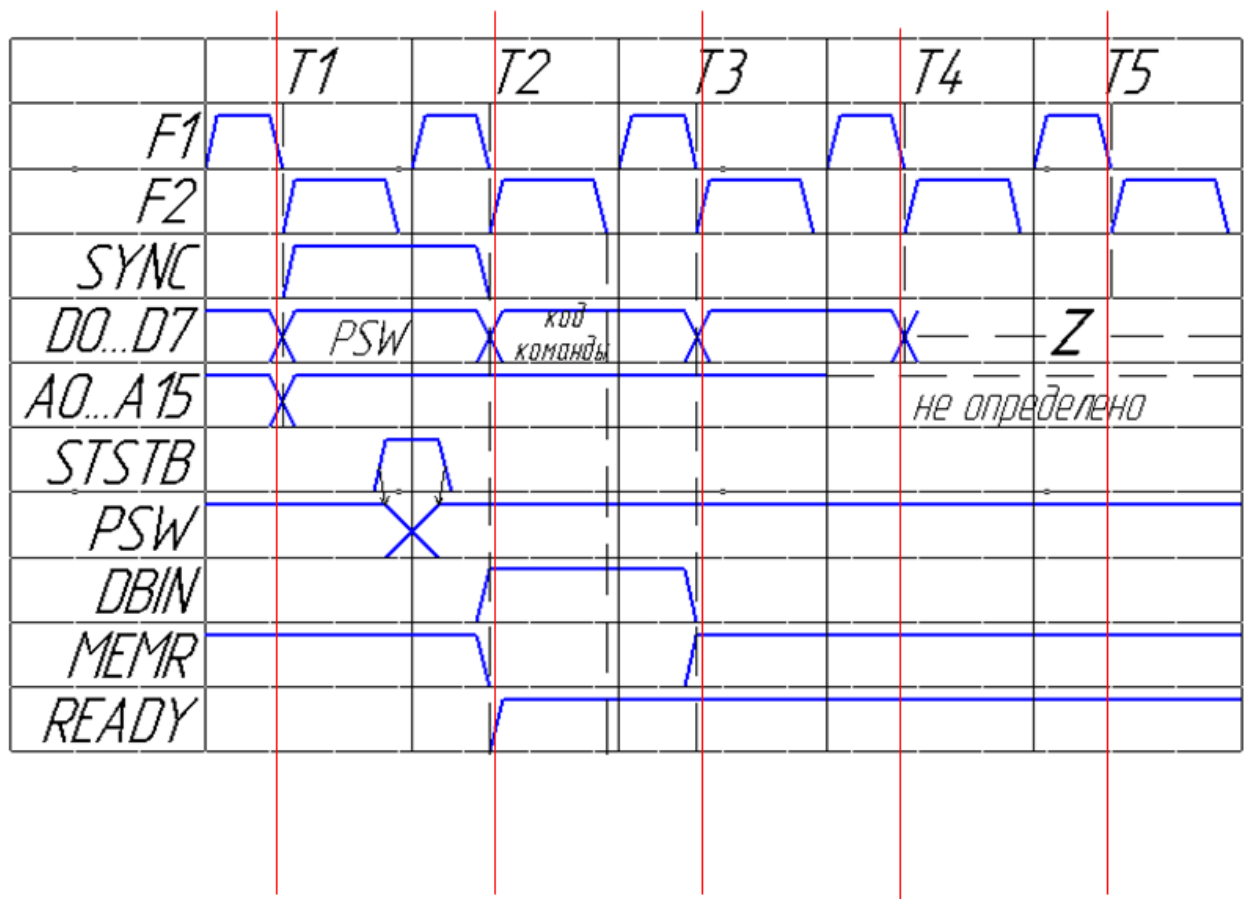
#### ТАКТ, КОМАНДНИЙ ЦИКЛ, МАШИННИЙ ЦИКЛ, МІКРОКОМАНДА

- ЧАС ВИКОНАННЯ КОМАНДИ – КОМАНДНИЙ ЦИКЛ
- КОЖНА ПЕРЕСИЛКА ДАНИХ ПО ШИНІ – МАШИННИЙ ЦИКЛ
  - КОМАНДНИЙ ЦИКЛ 1-7..10 МАШИННИХ ЦИКЛІВ
  - ПЕРЕСИЛКА ПО ШИНІ ДАНИХ 1-2 БАЙТИ
- МАШИННИЙ ЦИКЛ
  - ДЕКІЛЬКА МІКРОКОМАНД
- МІКРОКОМАНДА – ДІЇ ЗА ОДИН ТАКТ
  - ТАКТ 1. ВИДАЧА НА ШИНУ ДАНИХ БАЙТА СТАНУ, НА ШИНУ АДРЕСИ – АДРЕСИ КОМАНДИ, НА ШИНУ КЕРУВАННЯ СИГНАЛУ СИНХРОНІЗАЦІЇ
  - ТАКТ 2. ОЧІКУВАННЯ ГОТОВНОСТІ ЗОВНІШНЬОГО ПРИСТРОЮ ЧИ ПАМ'ЯТІ
  - ТАКТ 3 ОТРИМАННЯ КОМАНДИ
  - .....
  - ТАКТ N ВИКОНАННЯ

[4.19]

#### ПРИКЛАД МАШИННОГО ЦИКЛА

- ТАКТ1 –ВИДАЧА АДРЕСИ КОМАНДИ І СЛОВА СТАНУ
  - ІНФОРМАЦІЯ ПРО ТЕ, ЩО ПРОЦЕСОР “ХОЧЕ” РОБИТИ
- ТАКТ 2 – ПЕРЕВІРКА ГОТОВНОСТІ ЗОВНІШНІЙ ПРИСТРОЇВ
  - ПЕРЕВІРЯЄ ПОКИ НЕ БУДЕ СИГНАЛУ READY
- ТАКТ 3 – ЧИТАЄ КОМАНДУ ПО ШИНІ АДРЕСИ
- ТАКТ 4 – ВИКОНАННЯ КОМАНДИ
- МОЖЕ БУТИ БАГАТО ЦИКЛІВ І ТАКТІВ В ЗАЛЕЖНОСТІ ВІД КОМАНДИ
  - 1-7 ЦИКЛІВ 3-25 ТАКТІВ



В СУЧАСНИХ ПРОЦЕСОРАХ  
ЗА ТАКТ МОЖЕ ВИКОНУВАТИСЬ ДЕКІЛЬКА ОПЕРАЦІЙ

[4.20]

#### ОСОБЛИВОСТІ РОБОТИ З ПАМ'ЯТТЮ

- КОЖЕН ЕЛЕМЕНТ ДАНИХ (БАЙТ, СЛОВО) МАЄ АДРЕСУ В ПАМ'ЯТІ
  - ПРИ ВИСТАВЛЕННЯ АДРЕСИ НА ШИНУ АДРЕСИ НА ШИНІ ДАНИХ КОНТРОЛЕР ПАМ'ЯТІ ВИДАЄ ДАНІ З ЦІЄЇ КОМІРКИ
- ПРЯМА АДРЕСАЦІЯ
  - АДРЕС ПАМ'ЯТІ Є ПАРАМЕТРОМ КОМАНДИ
  - ПОВІЛЬНА ПЕРЕДАЧА ПО ШИНІ
- НЕПРЯМА АДРЕСАЦІЯ
  - АДРЕСА ПАМ'ЯТІ ЗНАХОДИТЬСЯ В РЕГІСТРІ
  - ШВИДКА ПЕРЕДАЧА ПО ШИНІ
- ІНДЕКСНА АДРЕСАЦІЯ
  - АДРЕСА ПАМ'ЯТІ ЗНАХОДИТЬСЯ В РЕГІСТРІ
  - В ІНШОМУ РЕГІСТРІ ЗНАХОДИТЬСЯ ЗМІЩЕННЯ
  - ЕФЕКТИВНО ОПТИМІЗУЄТЬСЯ
- СТЕК
  - В РЕГІСТРІ ЗНАХОДИТЬСЯ АДРЕСА
  - ПРИ ЗАПИСІ ДО АДРЕСИ ДАНІ ПЕРЕСИЛАЮТЬСЯ І АДРЕСА ЗМЕНШУЄТЬСЯ НА 1
  - ПРИ ЗЧИТУВАННІ ДАНІ ПЕРЕСИЛАЮТЬСЯ І АДРЕСА ЗБІЛЬШУЄТЬСЯ НА 1
  - ЕФЕКТИВНА ДЛЯ ВИКЛИКІВ ПІДПРОГРАМ

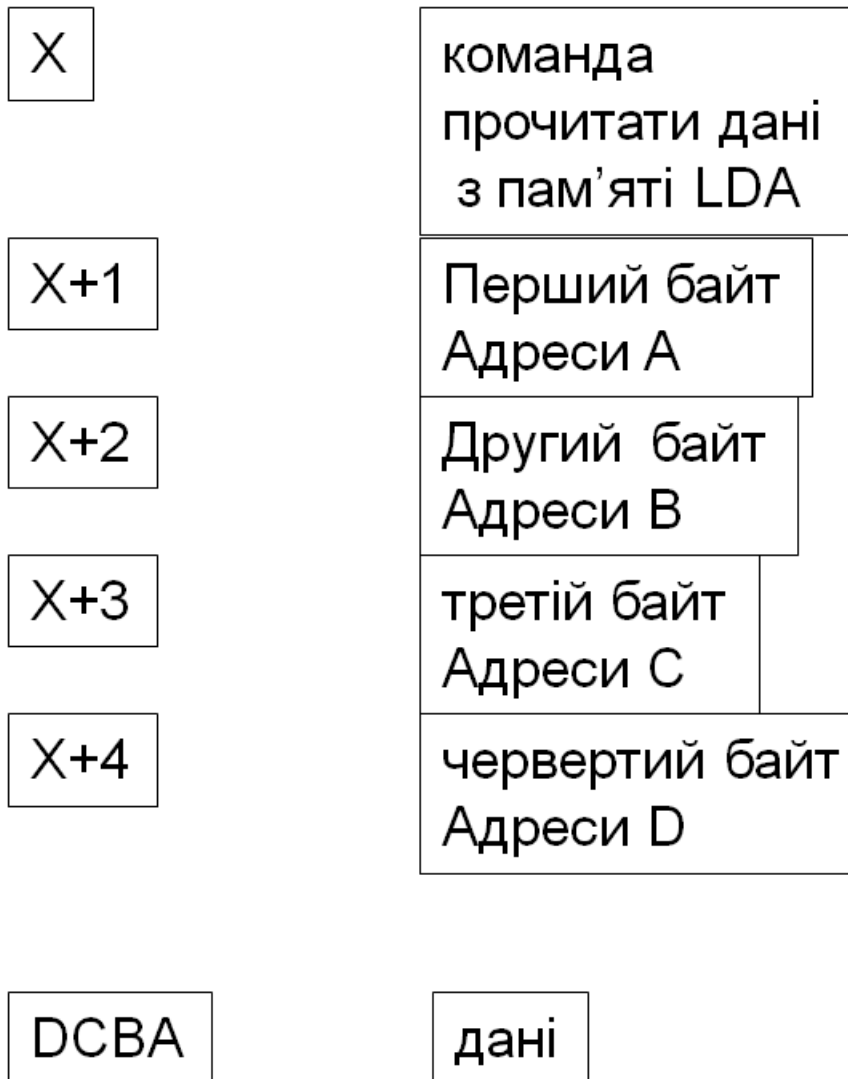
[4.21]

#### ПРЯМА (БЕЗПОСЕРЕДНЯ) АДРЕСАЦІЯ

- АДРЕСА ВКАЗУЄТЬСЯ ЯК ПАРАМЕТР КОМАНДИ

- LDA DCBA – ЗАВАНТАЖИТИ В АККУМУЛЯТОР ДАНІ, ЯКІ ЗБЕРІГАЮТЬСЯ В ПАМ'ЯТІ ЗА АДРЕСОЮ DCBA
- ЗАСТОСОВУЄТЬСЯ РІДКО
  - 5 ЦИКЛІВ ЧИТАННЯ КОМАНДИ
  - 1 ЦИКЛ ЧИТАННЯ ДАНИХ
  - ДУЖЕ ПОВІЛЬНО (~20 ТАКТІВ)
  - АБСОЛЮТНІ АДРЕСИ НЕ ДАЮТЬ МОЖЛИВІСТЬ ПЕРЕМІЩУВАТИ ДАНІ В ПАМ'ЯТІ

Адреси пам'яті                      вміст

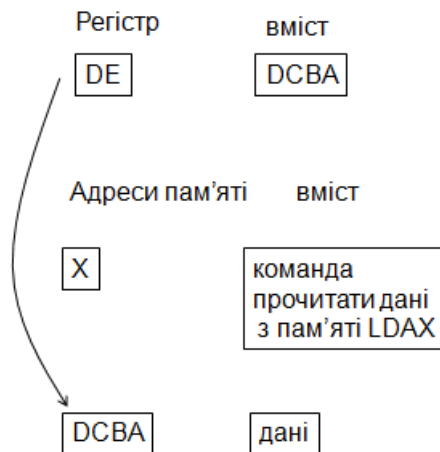


[4.22]

#### НЕПРЯМА (РЕГІСТРОВА) АДРЕСАЦІЯ

- АДРЕСА ЗБЕРІГАЄТЬСЯ В РЕГІСТРІ
- LDAХ ЗАВАНТАЖИТИ В АККУМУЛЯТОР ДАНІ, ЯКІ ЗБЕРІГАЮТЬСЯ ЗА АДРЕСОЮ ПАМ'ЯТІ, ЯКА ЗАПИСАНА В РЕГІСТРІ DE
- ЗАСТОСОВУЄТЬСЯ ЧАСТО
  - 1 ЦИКЛ ЧИТАННЯ КОМАНДИ
  - 1 ЦИКЛ ЧИТАННЯ ДАНИХ
  - ШВИДКО (~5 ТАКТІВ)
  - ЛЕГКО ПЕРЕМІЩУВАТИ ДАНІ В ПАМ'ЯТІ

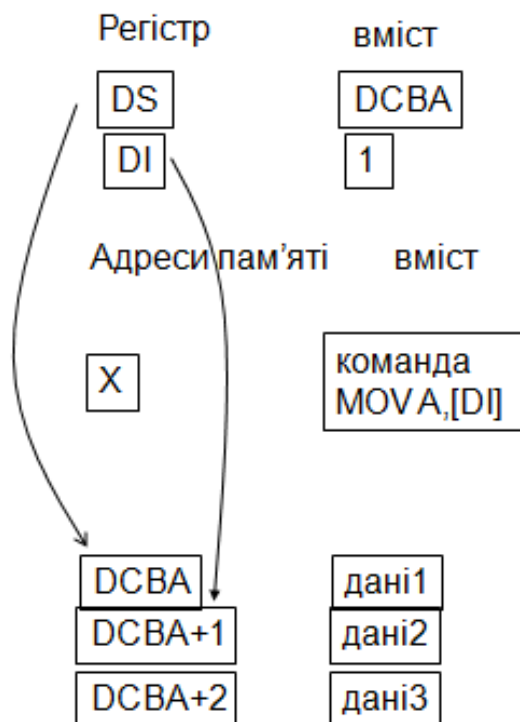




[4.23]

### ІНДЕКСНА АДРЕСАЦІЯ – СПЕЦІАЛЬНА ФОРМА НЕПРЯМОЇ

- АДРЕСА ЗБЕРІГАЄТЬСЯ В ОДНОМУ РЕГІСТРІ, ЗМІЩЕННЯ – В ІНШОМУ
- MOV A,[DI] ЗАВАНТАЖИТИ ДАНІ, ЯКІ МІСТЯТЬСЯ В ПАМ'ЯТІ ЗА АДРЕСОЮ, ЯКА РІВНА СУМІ ЗНАЧЕНЬ В РЕГІСТРІ DS ТА DI
- ШИРОКО ЗАСТОСОВУЄТЬСЯ
  - 1 ЦИКЛ ЧИТАННЯ КОМАНДИ
  - 1 ЦИКЛ ЧИТАННЯ ДАНИХ
  - ШВИДКО (~5 ТАКТИВ)
  - ЕФЕКТИВНО ПРАЦЮВАТИ З МАСИВАМИ (КОЖНА НАСТУПНА КОМАНДА 3 ТАКТИ)



[4.24]

### ПРЯМА ВІДНОСНА АДРЕСАЦІЯ – СПЕЦІАЛЬНА ФОРМА ПРЯМОЇ

- В ПАРАМЕТРІ КОМАНДИ ВКАЗУЄТЬСЯ ЗМІЩЕННЯ ВІДНОСНО АДРЕСИ КОМАНДИ
- ЧАСТО ЗАСТОСОВУЄТЬСЯ

- 2 ЦИКЛИ ЧИТАННЯ КОМАНДИ
- 1 ЦИКЛ ЧИТАННЯ ДАНИХ
- МОЖН АПЕРЕМІЩУВАТИ ПРОГРАМУ В ПАМ'ЯТІ

Адреси пам'яті      вміст

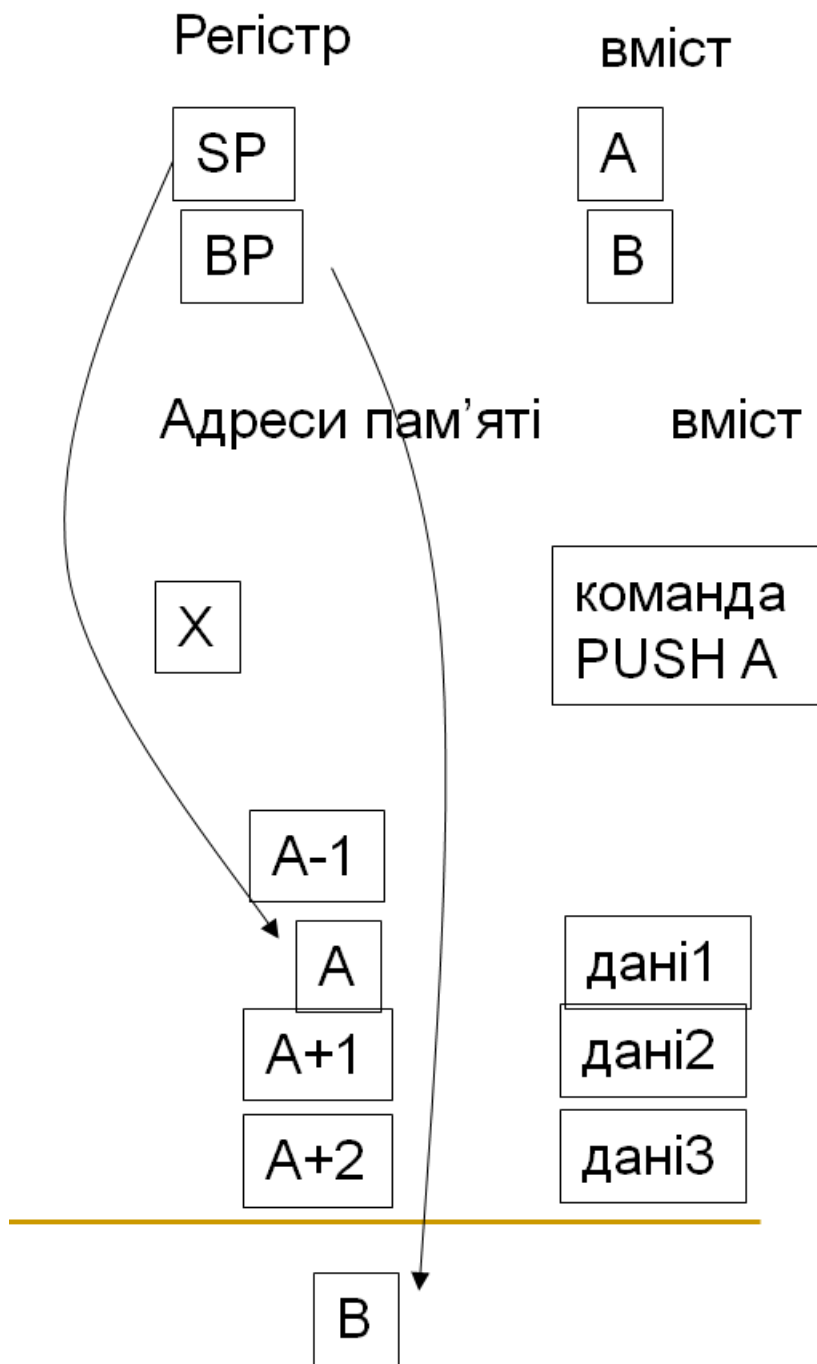
X	команда прочитати дані з пам'яті LDA
X+1	Зміщення A

X+A	дані
-----	------

[4.25]

#### СТЕК - СПЕЦІАЛЬНА ФОРМА НЕПРЯМОЇ АДРЕСАЦІЇ

- ОСТАННІМ ЗАЙШОВ-ПЕРШИМ ВИЙШОВ
- В РЕГІСТРІ SP (STACK POINTER) ЗБЕРІГАЄТЬСЯ АДРЕСА ПАМ'ЯТІ ДЕ ЗНАХОДЯТЬСЯ ОСТАННІ ЗАПИСАНІ ДАНІ
- ЗАПИС ДАНИХ PUSH A
  - ЗНАЧЕННЯ SP ЗМЕНШУЄТЬСЯ НА1
  - ДАНІ З A ЗАПИСУЮТЬСЯ ЗА НОВИМ ЗНАЧЕННЯМ SP (У ВЕРШИНУ СТЕКА)
- ЧИТАННЯ ДАНИХ POP A
  - ДАНІ ЗА АДРЕСОЮ SP ЗЧИТУЮТЬСЯ В РЕГІСТР A
  - ЗНАЧЕННЯ SP ЗБІЛЬШУЄТЬСЯ НА1
- РЕГІСТР BP-ОСНОВА СТЕКА ([SP]>=[BP])

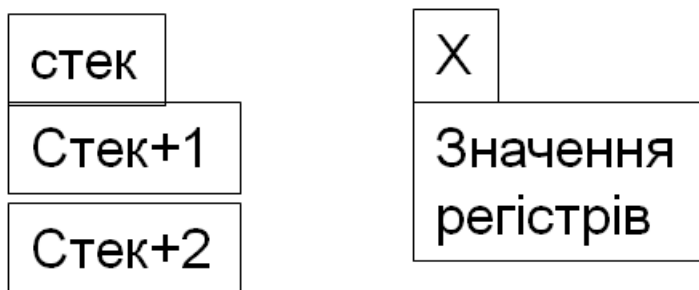
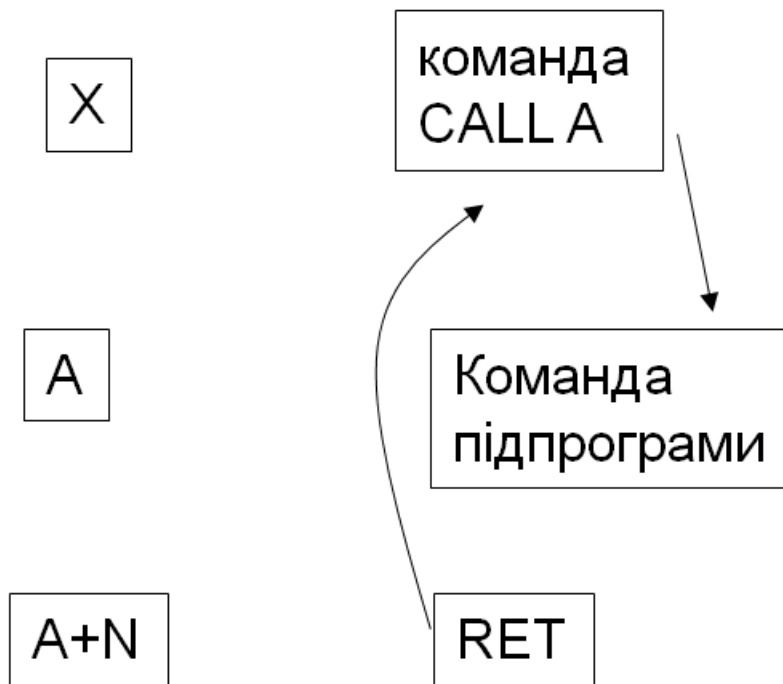


[4.26]

#### ВИКЛИК ПІДПРОГРАМ

- ПІДПРОГРАМА – ЧАСТИНА ПРОГРАМИ, ЯКА МОЖЕ ВИКЛИКАТИСЬ З РІЗНИХ ЧАСТИН ПРОГРАМИ
- КОМАНДА ВИКОНАННЯ ПІДПРОГРАМИ CALL ADDR
  - ЗАПИС В СТЕК ПОТОЧНОГО ЗНАЧЕННЯ КОМАНДНОГО ЛІЧИЛЬНИКА
  - ЗАПИС В СТЕК РЕГІСТРА СТАНУ І ІНШИХ РЕГІСТРІВ
  - ПЕРЕХІД ЗА АДРЕСОЮ ADDR
- КОМАНДА ПОВЕРНЕННЯ З ПІДПРОГРАМИ RET
  - ВІДНОВЛЕННЯ ІЗ СТЕКУ РЕГІСТРА СТАНУ І ІНШИХ РЕГІСТРІВ
  - ВІДНОВЛЕННЯ КОМАНДНОГО ЛІЧИЛЬНИКА

Адреси пам'яті      вміст



[4.27]

#### ОБЛАСТІ ПАМ'ЯТІ

- ЧАСТО ОБЛАСТЬ ПАМ'ЯТІ ДІЛЯТЬ НА ФУНКЦІОНАЛЬНІ ЧАСТИНИ
- СЕГМЕНТ КОДУ (CODE SEGMENT, CS)
  - АДРЕСИ ПАМ'ЯТІ В ЯКИХ ЗБЕРІГАЮТЬСЯ КОМАНДИ ПРОЦЕСОРА
- СЕГМЕНТ ДАНИХ (DATA SEGMENT DS)
  - АДРЕСИ ПАМ'ЯТІ В ЯКИХ ЗБЕРІГАЮТЬСЯ ДАНІ
- СТЕК (STACK SEGMENT, SS)
  - АДРЕСИ ПАМ'ЯТІ В ЯКИХ ЗНАХОДИТЬСЯ СТЕК
- КУЧА
  - АДРЕСИ ПАМ'ЯТІ, ЯКІ ВИКОРИСТОВУЮТЬСЯ ДЛЯ ТИМЧАСОВОГО ВИДІЛЕННЯ ПАМ'ЯТІ

[4.28]

#### АРХІТЕКТУРИ ЕОМ – КОНЦЕПЦІЯ ПОБУДОВИ

- ФОН-НЕЙМАНІВСЬКА АРХІТЕКТУРА
  - ❑ ПРОЦЕСОР ВІДДІЛЕНИЙ ВІД ПАМ'ЯТІ
  - ❑ СИСТЕМА КОМАНД НЕ МІНЯЄТЬСЯ
  - ❑ ОДНА Й ТА Є САМА ПАМ'ЯТЬ МОЖЕ ВИКОРИСТОВУВАТИСЬ ДЛЯ КОМАНД, ДАНИХ, СТЕКУ, КУЧІ, ТОЩО
  - ❑ ПАМ'ЯТЬ АДРЕСУЄТЬСЯ ПОСЛІДОВНО
  - ❑ ПРОГРАМА ВИКОНУЄТЬСЯ ПОСЛІДОВНО
  - ❑ НАДЗВИЧАЙНО ПОШИРЕНА І УНІВЕРСАЛЬНИЙ ПІДХІД
- ГАРВАРДСЬКА АРХІТЕКТУРА
  - ❑ КОМАНДИ І ДАНІ ЗБЕРІГАЮТЬСЯ В РІЗНИХ ОБЛАСТЯХ ПАМ'ЯТІ І ПЕРЕДАЮТЬСЯ ПО РІЗНИМ КАНАЛАМ
  - ❑ МОДИФІКОВАНА ГАРВАРДСЬКА АРХІТЕКТУРА – В СЕРЕДИНІ І ЗЗОВНІ ПРОЦЕСОРА РІЗНІ ШИНИ АДРЕС І ДАНИХ
  - ❑ ПЕРЕВАГИ: ПАМ'ЯТЬ ПРОГРАМ МОЖЕ БУТИ БІЛЬШОЮ І ДЕШЕВШОЮ НІЖ ДАНИХ, МОЖЛИВА ПАРАЛЕЛЬНА ПЕРЕДАЧА КОМАНД І ДАНИХ
  - ❑ НЕДОЛІКИ – НЕ УНІВЕРСАЛЬНА
- ГІБРИДНІ АРХІТЕКТУРИ
  - ❑ РІЗНИЙ КЕШ КОМАНД І ДАНИХ, АЛЕ СПІЛЬНА ОПЕРАТИВНА ПАМ'ЯТЬ

## ЛЕКЦІЯ 05

[5.1]

### ПЕРЕДАЧА ДАНИХ МІЖ КОМПОНЕНТАМИ МІКРОПРОЦЕСОРНИХ СИСТЕМ

#### ЛЕКЦІЯ 5

СУДАКОВ О.О, РАДЧЕНКО С.П.

«СУЧАСНА МІКРОПРОЦЕСОРНА ТЕХНІКА»

[5.2]

#### ПРОБЛЕМА ВЗАЄМОДІЇ КОМПОНЕНТІВ ПРОЦЕСОРНИХ СИСТЕМ

- ВЕЛИКА КІЛЬКІСТЬ ПРИСТРОЇВ ВЗАЄМОДІЄ МІЖ СОБОЮ І З ПРОЦЕСОРОМ
- ВСІ ЦІ ПРИСТРОЇ ТРЕБА З'ЄДНАТИ МІЖ СОБОЮ
- ЗАСТОСОВУЮТЬСЯ СПІЛЬНІ ШИНИ (BUS, BIDIRECTIONAL UNIVERSAL SWITCH )
- ЗАСТОСУВАТИ ОКРЕМІ ШИНИ ДЛЯ КОЖНОГО ПРИСТРОЮ - НЕ МОЖЛИВО
- СПІЛЬНІ ШИНИ – КОНФЛІКТ ПРИ ДОСТУПІ
- ЗАСТОСОВУЮТЬСЯ РІЗНІ ШИНИ І РІЗНІ ПРОТОКОЛИ
  - ПАРАЛЕЛЬНІ
  - ПОСЛІДОВНІ
  - СИНХРОННІ
  - АСИНХРОННІ

[5.3]

#### СПЕЦІАЛЬНІ СТАНИ ПРОЦЕСОРА

- СТАН **ЧИТАННЯ-ЗАПИСУ ПАМ'ЯТІ (MEMORY READ WRITE)**
  - ПО ШИНІ АДРЕСИ ПЕРЕДАЮТЬСЯ АДРЕСИ ПАМ'ЯТІ
  - ПО ШИНІ ДАНИХ ПЕРЕДАЮТЬСЯ КОМАНДИ АБО ДАНІ
  - СПЕЦІАЛЬНІ СИГНАЛИ КЕРУВАННЯ ВКАЗУЮТЬ НА ДОСТУП ДО ПАМ'ЯТІ
- СТАН **ЧИТАННЯ-ЗАПИСУ ПОРТІВ ВВЕДЕННЯ ВИВЕДЕННЯ (PORT IN/OUT)**
  - ПО ШИНІ АДРЕСИ ПЕРЕДАЮТЬСЯ НОМЕР ПОРТА ВВЕДЕННЯ-ВИВЕДЕННЯ
  - ПО ШИНІ ДАНИХ ПЕРЕДАЮТЬСЯ КОМАНДИ АБО ДАНІ
  - СПЕЦІАЛЬНІ СИГНАЛИ КЕРУВАННЯ ВКАЗУЮТЬ НА ДОСТУП ДО ЗОВНІШНІХ ПРИСТРОЇВ
- СТАН **ЗУПИНКИ (STOP)**
  - ПРОЦЕСОР НІЧОГО НЕ ВИКОНУЄ
  - СПЕЦІАЛЬНІ СИГНАЛИ КЕРУВАННЯ ВКАЗУЮТЬ НА ЗУПИНКУ
- СТАН **УТРИМАННЯ (HOLD)**
  - ПРОЦЕСОР НІЧОГО НЕ ПЕРЕДАЄ ПО ШИНІ І ПЕРЕВОДИТЬ ВИВОДИ ШИН СТАН ВИСОКОГО ОПОРУ
  - СПЕЦІАЛЬНІ СИГНАЛИ КЕРУВАННЯ ВКАЗУЮТЬ НА УТРИМАННЯ
- СТАН **ПЕРЕРИВАННЯ (INTERRUPT)**
  - ПРОЦЕСОР ПЕРЕРИВАЄ ВИКОНАННЯ ЗА ЗАПИТОМ ЗОВНІШНЬОГО ПРИСТРОЮ
  - СПЕЦІАЛЬНІ СИГНАЛИ КЕРУВАННЯ ВКАЗУЮТЬ НА ПЕРЕРИВАННЯ
- СТАН **СКИДАННЯ (RESET)**
  - ПОЧАТКОВИЙ СТАН
- СТАН **ВВІМКНЕННЯ**
  - ВІДРАЗУ ПІСЛЯ ВВІМКНЕННЯ ЖИВЛЕННЯ

[5.4]

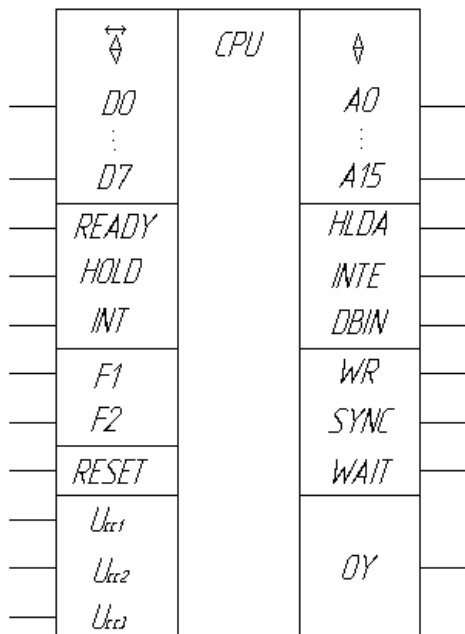
#### СТАН ВВІМКНЕННЯ (POWER ON)

- РЕГІСТРИ ПРОЦЕСОРА НЕВИЗНАЧЕНІ

- ПРОЦЕСОР НЕ МОЖНА ВИКОРИСТОВУВАТИ ПОКИ ВІН НЕ БУДЕ ПЕРЕВЕДЕНИЙ У СТАН ІНІЦІАЛІЗАЦІЇ
- ПРИ ВВІМКНЕННІ ЖИВЛЕННЯ ДЕЯКІ ПРОЦЕСОРИ ВМІЮТЬ ЦЕ ОБРОБЛЯТИ
  - ПЕРЕРИВАННЯ ПО ВВІМКНЕННЮ ЖИВЛЕННЯ
- ДЕЯКІ ПРОЦЕСОРИ ВИМАГАЮТЬ ІНІЦІАЛІЗАЦІЇ
  - ШТУЧНО ПОДАТИ ВИСОКИЙ ЧИ НИЗЬКИЙ РІВЕНЬ НА ВИВІД RESET
- ДЕЯКІ ПРОЦЕСОРИ ПРИ ЗБОЯХ ЧИ ЗАНИЖЕННІ ЖИВЛЕННЯ ВИКОНУЮТЬ ІНІЦІАЛІЗАЦІЮ
  - BROWN OUT RESET

[5.5]

- СТАН ІНІЦІАЛІЗАЦІЇ (СКИДАННЯ, RESET) ВИНИКАЄ ПРИ ПОДАЧІ СИГНАЛУ НА ВИВІД RESET
- ВСІ РЕГІСТРИ, КРІМ РЕГІСТРІВ ЛІЧИЛЬНИКА КОМАНД (IP, CODE SEGMENT) НЕ ЗМІНЮЮТЬСЯ
- РЕГІСТРИ ЛІЧИЛЬНИКА КОМАНД (IP, CS) ВСТАНОВЛЮЮТЬСЯ У ПЕВНЕ ЗНАЧЕННЯ (RESET VECTOR)
- ВИКОНАННЯ ПОЧИНАЄТЬСЯ З МАШИННОЇ ІНСТРУКЦІЇ ЗА ЦІЄЮ АДРЕСОЮ

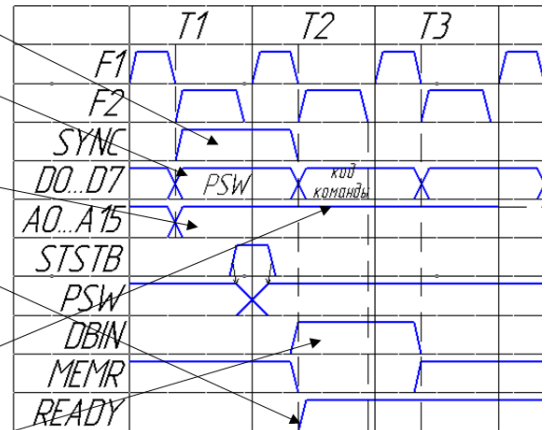


[5.6]

#### ЦИКЛ ЧИТАННЯ (ВИБІРКА)-ЗАПИС ПАМ'ЯТІ (MEMORY)

- ТАКТ 1
  - ПРОЦ. СИГНАЛ СИНХРОНІЗАЦІЇ
  - ПРОЦ. СЛОВО СТАНУ НА ШИНУ ДАНИХ
  - ПРОЦ. СПЕЦІАЛЬНИЙ БІТ ВКАЗУЄ ЧИТАННЯ/ЗАПИС ПАМ'ЯТІ
  - ПРОЦ. АДРЕСА КОМАНДИ
- ТАКТ 2 ПЕРЕВІРКА ГОТОВНОСТІ
  - ПРОЦ. ПЕРЕВІРКА СИГНАЛУ READY
  - ПРОЦ. ПЕРЕВІРКА СИГНАЛІВ ПЕРЕРИВАННЯ, ЗАХОПЛЕННЯ
  - ОЧІКУВАННЯ ПОКИ НЕМАЄ ЦИХ СИГНАЛІВ
- ТАКТ 3
  - ПАМ'ЯТЬ. ВИДАЄ ВМІСТ ПАМ'ЯТІ НА ШИНУ ДАНИХ
  - ПАМ'ЯТЬ. ВИДАЄ СИГНАЛ ЗАПISУ
- АНАЛОГІЧНІ ЦИКЛИ МОЖУТЬ ПОСТОРЮВАТИСЬ

- Такт 1
  - Проц. Сигнал синхронізації
  - Проц. Слово стану на шину даних
  - Проц. Спеціальний біт вказує читання/запис пам'яті
  - Проц. Адреса команди
- Такт 2 перевірка готовності
  - Проц. Перевірка сигналу ready
  - Проц. Перевірка сигналів переривання, захоплення
  - Очікування поки немає цих сигналів
- Такт 3
  - Пам'ять. Видає вміст пам'яті на шину даних
  - Пам'ять. Видає сигнал запису
- Аналогічні цикли можуть посторюватись



[5.7]

### ЦИКЛ ЧИТАННЯ ПОРТУ ВВОДУ ВИВОДУ (IO)

- ВІДРІЗНЯЄТЬСЯ ВІД ЦИКЛА ЧИТАННЯ ПАМ'ЯТІ
  - БІТАМИ У СЛОВІ СТАНУ
  - НА ШИНУ АДРЕСИ ВИДАЄТЬСЯ НЕ АДРЕСА ПАМ'ЯТІ А НОМЕР ПОРТА
  - ДАНІ ПЕРЕДАЮТЬСЯ НЕ В ПАМ'ЯТЬ А ВЗОВНІШНІЙ ПРИСТРІЙ
- САМА КОМАНДА ВВОДУ- ВИВОДУ ОДНОГО БАЙТА ТРИВАЄ ДЕКІЛЬКА ЦИКЛІВ
  - ЦИКЛ 1 - ВИБІРКА КОМАНДИ
  - ЦИКЛ 2 – ВИБІРКА НОМЕРА ПОРТУ
  - ЦИКЛ 3 – ЧИТАННЯ ПОРТУ ВВОДУ ВИВОДУ
- ПРИСТРІЙ ПОВИНЕН ВІДСЛІДКУВАТИ СИГНАЛИ SYNC, ШИНУ ДАНИХ, ШИНУ АДРЕСИ, ВИДАВАТИ СИГНАЛ ГОТОВНОСТІ

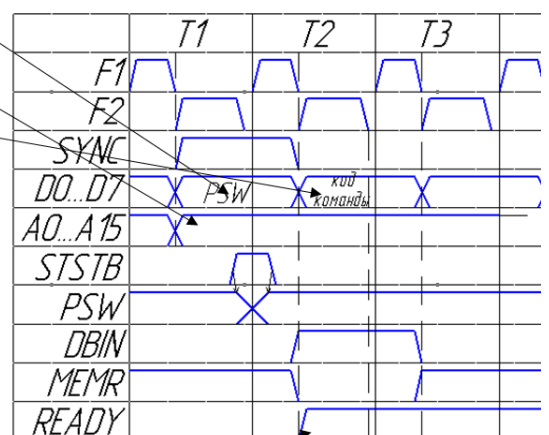
- Відрізняється від циклу читання пам'яті

- бітами у слові стану
- На шину адреси видається не адреса пам'яті а номер порта
- Дані передаються не в пам'ять а зовнішній пристрій

- Сама команда вводу- виводу одного байта триває декілька циклів

- Цикл 1 - вибірка команди
- Цикл 2 – вибірка номера порту
- Цикл 3 – читання порту вводу виводу

- Пристрій повинен відслідковувати сигнали SYNC, шину даних, шину адреси, видавати сигнал готовності



[5.8]

- СТАН ЗУПИНКИ (STOP,HALT)
  - ПРОЦЕСОР ЗУПИНЯЄ РОБОТУ
  - ПРИ ВИКОНАННІ КОМАНДИ HLT



- ЗУПИНЯЄТЬСЯ В НАСТУПНОМУ ЦИКЛІ ПІСЛЯ ВИДАЧІ СЛОВА СТАНУ, ДЕ ВСТАНОВЛЕНИЙ СПЕЦІАЛЬНИЙ БІТ ЗУПИНКИ
- ЗАПУСТИТИ ПРОЦЕСОР МОЖНА
  - ПЕРЕРИВАННЯМ
  - ПЕРЕВЕДЕННЯМ В СТАН ЗАХОПЛЕННЯ
  - ПЕРЕВЕДЕННЯМ В СТАН RESET
- ВИКОРИСТОВУЄТЬСЯ ДЛЯ НАЛАГОДЖЕННЯ ПРОГРАМ, ЕКОНОМІЇ ЕЛЕКТРОЕНЕРГІЇ

[5.9]

#### СТАН ЗАХОПЛЕННЯ (УТРИМАННЯ, HOLD)

- ПЕРЕВЕДЕННЯ ВСІХ ВИВОДІВ ШИН У СТАН ВИСОКОГО ВИХІДНОГО ОПОРУ
  - ЗДІЙСНЮЄТЬСЯ В ТАКТІ ПЕРЕВІРКИ БУДЬ-ЯКОГО ЦИКЛУ
  - ДЛЯ ПЕРЕВЕДЕННЯ В HOLD ЗОВНІШНІЙ ПРИСТРІЙ ПОДАЄ СИГНАЛ НА ВХІД HOLD
  - ПІСЛЯ ПЕРЕВЕДЕННЯ В СТАН HOLD ПРОЦЕСОР НА ВИВОДІ ANOLD ВСТАНОВЛЮЄ СИГНАЛ ПЬДТВЕРДЖЕННЯ ЗАХОПЛЕННЯ
  - ПІСЛЯ ЦЬОГО ЗОВНІШНІ ПРИСТРОЇ МОЖУТЬ ПЕРЕДАВАТИ ДАНІ ПО ШИНІ
- ЗАСТОСОВУЄТЬСЯ ДЛЯ ОБМІНУ ПО ШИНІ БЕЗ УЧАСТІ ПРОЦЕСОРА
  - ПРЯМИЙ ДОСТУП ДО ПАМ'ЯТІ (ПДП, DIRECT MEMORY ACCESS, DMA)
- ЗОВНІШНІЙ ПРИСТРІЙ ЗАПИСУЄ/ЧИТАЄ ДАНІ В ПАМ'ЯТЬ БЕЗ УЧАСТІ ПРОЦЕСОРА
  - ШВИДКО ВИКОНУЄТЬСЯ ВВІД-ВИВІД
  - НЕ СПОЖИВАЄТЬСЯ ЕНЕРГІЯ ПРИЦЕСОРОМ

[5.10]

#### КОНТРОЛЕР ПДП

- СПРОЩУЄ ОБМІН ПДП
- МАЄ ДЕКІЛЬКА КАНАЛІВ
- КОЖЕН КАНАЛ – БУФЕРНІ РЕГІСТРИ
  - АДРЕСА
  - ДАНІ
  - КЕРУВАННЯ
  - СТАН
  - ЛІЧИЛЬНИК
- ПРИ ЗАПИТІ НА ПДП ВІД ПРИСТРОЮ
  - ЗАПИС В РЕГІСТРИ НОМЕР КАНАЛУ, АДРЕСУ ПОЧАТКУ АБО КІНЦЯ, КІЛЬКІСТЬ БАЙТ
  - ЗАПИТ НА ПДП
  - ПРИ ПЬДТВЕРДЖЕННІ ЗАХОПЛЕННЯ ШИНИ ПРИСТРІЙ ПЕРЕДАЄ ДАНІ КОНТРОЛЕРУ
  - КОНТРОЛЕР ПЕРЕДАЄ ДАНІ В ПАМ'ЯТЬ



[5.11]  
**СТАН ПЕРЕРИВАННЯ (INTERRUPT)**

- ПЕРЕРИВАННЯ – ПРОЦЕСОР ПЕРЕРИВАЄ ВИКОНАННЯ ПРОГРАМИ І ВИКОНУЄ ЗАПИТ ВІД ЗОВНІШНЬОГО ПРИСТРОЮ
  - ПРОЦЕСОР ПОВИНЕН ДОЗВОЛИТИ ПЕРЕРИВАННЯ КОМАНДА ІЕ
  - ДЛЯ ЗАБОРОНИ ПЕРЕРИВАННЯ КОМАНДА ІD
  - В РЕЖИМІ ПДП ПЕРЕРИВАННЯ ЗАБОРОНЕНІ
  - З РЕЖИМУ STOP ВИВОДИТЬСЯ ПЕРЕРИВАННЯМ
  - ПРИСТРІЙ ВИДАЄ НА ВИВІД ПРОЦЕСОРА INT СИГНАЛ
  - ПРОЦЕСОР В КІНЦІ КОЖНОГО ЦИКЛА ПЕРЕВІРЯЄ ЗАПИТ НА ПЕРЕРИВАННЯ
  - ПРИ НАЯВНОСТІ ЗАПИТУ НА ПЕРЕРИВАННЯ ПРОЦЕСОР ВИДАЄ СИГНАЛ ПІДТВЕРДЖЕННЯ НА ВИВІДІ INTA
  - ПРОЦЕСОР ЧЕКАЄ НА КОМАНДУ НА ШИНІ ДАНИХ
  - ПРИ ОТРИМАННІ КОМАНДИ ПОЧИНАЄТЬСЯ ОБРОБКА ПЕРЕРИВАННЯ

[5.12]

- ПІСЛЯ ПІДТВЕРДЖЕННЯ ПЕРЕРИВАННЯ ПРОЦЕСОР ЗЧИТУЄ НА ШИНІ ДАНИХ КОД КОМАНДИ, АБО ЧИСЛО – ІНДЕКС ВЕКТОРА ПЕРЕРИВАНЬ (НОМЕР ПЕРЕРИВАННЯ)
  - ВЕКТОР ПЕРЕРИВАНЬ – МАСИВ АДРЕС ЗА ЯКИМИ ЗНАХОДЯТЬСЯ ПІДПРОГРАМИ ОБРОБКИ ПЕРЕРИВАНЬ
  - ПЕРЕРИВАННЯ 5 – ЕЛЕМЕНТ ВЕКТОРА 5
- ВІДБУВАЄТЬСЯ ВИКЛИК ПІДПРОГРАМИ ЗА АДРЕСОЮ, ЯКА ВІДПОВІДАЄ НОМЕРУ ПЕРЕРИВАННЯ У ВЕКТОРІ (АБО ВИКОНАННЯ КОМАНДИ ЯКУ ЗЧИТАЛИ З ШИНИ)
- ПІСЛЯ ЗАВЕРШЕННЯ ОБРОБКИ ВІДБУВАЄТЬСЯ ВИХІД ІЗ ПІДПРОГРАМИ
- ШИРОКО ЗАСТОСОВУЄТЬСЯ ДЛЯ ОБРОБКИ ЗАПИТІВ ВІД ЗОВНІШНІХ ПРИСТРОЇВ
- МОЖНА ПРОГРАМНО ЗМІНЮВАТИ КОД ОБРОБНИКА, ДОЗВОЛЯТИ І ЗАБОРОНЯТИ ПЕРЕРИВАННЯ

[5.13]

**КОНТРОЛЕР ПЕРЕРИВАНЬ**

- СПРОЦУЄ ВЗАЄМОДІЮ ІЗ ЗОВНІШНІМИ ПРИСТРОЯМИ

- ПРИЙМАЄ ЗАПИТИ ВІД ПРИСТРОЇВ ПО ВИВОДАМ ПЕРЕРИВАНЬ (НОМЕР ВИВОДА НОМЕР ПЕРЕРИВАННЯ)
- ВСТАНОВЛЮЄ ПРІОРИТЕТНІСТЬ ПЕРЕРИВАНЬ
  - КОЛИ ОБРОБЛЯЄТЬСЯ ПЕРЕРИВАННЯ З МЕНШИМ НОМЕРОМ – ПЕРЕРИВАННЯ З БІЛЬШИМ НОМЕРОМ НЕ ОБРОБЛЯЄТЬСЯ
  - ТАЙМЕР - НАЙПРІОРИТЕТНІШЕ

[5.14]

#### СПЕЦІАЛЬНІ ПЕРЕРИВАННЯ

- НЕШТАТНІ СТАНИ ПРОЦЕСОРА МОЖУТЬ ГЕНЕРУВАТИ ПЕРЕРИВАННЯ
  - ПЕРЕРИВАННЯ RESET
- НЕМАСКОВАНЕ ПЕРЕРИВАННЯ – ПЕРЕРИВАННЯ ЯКЕ МАЄ НАЙВИЩИЙ ПРІОРИТЕТ І ЗАБОРОНЯЄТЬСЯ, ЧИ ДОЗВОЛЯЄТЬСЯ ОКРЕМО
  - ОКРЕМИЙ ВИВІД ПРОЦЕСОРА (NON-MASKABLE INTERRUPT, NMI)
  - ГЕНЕРУЄТЬСЯ ПРИ ФАТАЛЬНИХ ПОМИЛКАХ (ПАМ'ЯТЬ, ТОЩО)
- ПРОГРАМНІ ПЕРЕРИВАННЯ
  - ВИКЛИК ВЕКТОРА ПЕРЕРИВАНЬ

[5.15]

#### ШИНА ISA

- INDUSTRIAL STANDARD ARCHITECTURE
- ВИКОРИСТОВУЄ ПОРТИ ВВОДУ ВИВОДУ, ПЕРЕРИВАННЯ, КАНАЛИ ПДП
- ПРИСТРІЙ МАЄ
  - ДІАПАЗОН ПОРТІВ ВВОДУ-ВИВОДУ IO RANGE
  - НОМЕР ПЕРЕРИВАННЯ IRQ
  - НОМЕР КАНАЛУ DMA
  - ДІАПАЗОН АДРЕС ПАМ'ЯТІ DMA
- ЛІНІЇ (62(ХТ)-98(АТ))
  - АДРЕСА 20(ХТ)-24(АТ) ЛІНІЙ
  - ДАНІ 8(ХТ)-16(АТ) ЛІНІЙ
  - ПЕРЕРИВАННЯ ДО 11 ЛІНІЙ
  - КЕРУВАННЯ
- ШВИДКІСТЬ 4-20 МГЦ
- ЗАРАЗ ВИКОРИСТОВУЄТЬСЯ РІДКО В ОСНОВНОМУ ДЛЯ СЕРВЕРІВ
- ПРОБЛЕМА
  - НЕМОЖЛИВО СПІЛЬНО ВИКОРИСТОВУВАТИ ПЕРЕРИВАННЯ
  - ОБМЕЖЕНА ШВИДКІСТЬ І КІЛЬКІСТЬ ПРИСТРОЇВ
  - ПРОБЛЕМИ З PLUG-AND-PLAY

[5.16]

#### ШИНА PCI (PERIPHERAL COMPONENTS INTERCONNECT)

- ШИНА НЕЗАЛЕЖНА ВІД ТИПУ ПРОЦЕСОРА
- ШВИДКІСТЬ 33, 66 ТА 133 МГЦ 32 ТА 64 БІТИ
- АВТОКОНФІГУРАЦІЯ
  - ПРИСТРОЇ ВИЗНАЧАЮТЬСЯ АВТОМАТИЧНО
- ВЗАЄМОДІЯ З ІНШИМИ ПРИСТРОЯМИ ТА ШИНАМИ ЗА ДОПОМОГОЮ МОСТІВ
  - «ПІВНІЧНИЙ МІСТ» – ВЗАЄМОДІЯ З ПРОЦЕСОРОМ, ПАМ'ЯТТЮ, ШВИДКИМИ ПРИСТРОЯМИ
  - «ПІВДЕННИЙ МІСТ» – ВЗАЄМОДІЯ З ПОВІЛЬНИМИ ПРИСТРОЯМИ ВВОДУ ВИВОДУ
  - PCI-PCI МІСТ
  - PCI-ISA МІСТ
- МОСТИ СКЛАДНІ КОНТРОЛЕРИ, ЯКІ ЗАБЕЗПЕЧУЮТЬ РОБОТУ ШИНИ
- ПЕРЕДАЧА ПО ШИНІ БЕЗ УЧАСТІ ПРОЦЕСОРА
- МОЖЛИВА ПАРАЛЕЛЬНА ПЕРЕДАЧА ДАНИХ І РОБОТИ ПРОЦЕСОРА

[5.17]

## ПОСЛІДОВНІ ШИНИ

- ПАРАЛЕЛЬНІ ШИНИ ШВИДКІ, АЛЕ Є ПРОБЛЕМИ
  - НАВОДКИ ВІД СУСІДНІХ ЛІНІЙ
  - НЕОБХІДНІСТЬ ОДНАКОВОСТІ ПАРАМЕТРІВ ЛІНІЙ
  - НЕОБХІДНІСТЬ ГАРАНТУВАННЯ СИНХРОННОСТІ ПЕРЕДАЧІ
  - ВЕЛИКА ІМОВІРНІСТЬ ВИХОДУ З ЛАДУ
  - ДОРОГО
  - МАЛА ДАЛЬНІСТЬ ПЕРЕДАЧІ
- ПОСЛІДОВНІ ШИНИ
  - ПРОСТІ
  - МОЖНА МАСШТАБУВАТИ ШВИДКІСТЬ
  - ПРАЦЮЮТЬ НА ВЕЛИКІЙ ВІДСТАНІ І ВЕЛИКІЙ ШВИДКОСТІ
  - ДЕШЕВІ
- ДУЖЕ ЧАСТО ЗАСТОСОВУЮТЬСЯ
  - UART
  - SPI
  - I<sup>2</sup>C

[5.18]

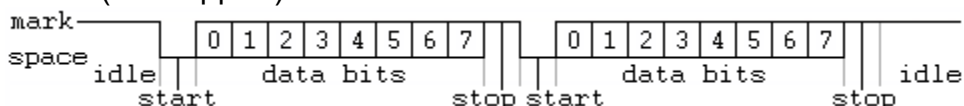
### UART (UNIVERSAL ASYNCHRONOUS RECEIVER TRANSCEIVER)

- ПРОТОКОЛ ПЕРЕДАЧІ ПОСЛІДОВНОЮ ШИНОЮ
- ПОТРІБНО 2-3 ПРОВІДИ (ОДНО ДВОНАПРАВЛЕНА ПЕРЕДАЧА)
  - АСИНХРОННИЙ – ПРИЙМАЧ І ПЕРЕДАВАЧ НЕ ПОВИННІ МАТИ СПІЛЬНОГО ТАКТОВОГО ГЕНЕРАТОРА
  - УНІВЕРСАЛЬНИЙ – ПІДХОДИТЬ ДЛЯ БУДЬ-ЯКИХ ДАНИХ
  - МОЖЕ ПРИЙМАТИ І ПЕРЕДАВАТИ
- ДАНІ ПРЕДСТАВЛЯЮТЬСЯ У ВИГЛЯДІ ПОТОКУ БІТІВ (СЕРІАЛІЗУЮТЬСЯ) І НАЗАД (ДЕСЕРІАЛІЗУЮТЬСЯ)
- ЗАСТОСОВУЄТЬСЯ ДЛЯ КЕРУВАННЯ ПРИСТРОЯМИ
  - КОНСОЛІ
  - МОДЕМИ
  - ПРОГРАМАТОРИ
- НЕДОЛІКИ
  - ПОВІЛЬНИЙ (СТАНДАРТ ДО ~100 КБАЙТ/С)
  - ВЕЛИКІ НАКЛАДНІ ВИТРАТИ (ПЕРЕДАЄТЬСЯ БАГАТО НЕПОТРІБНОЇ ІНФОРМАЦІЇ)

[5.19]

### ПЕРЕДАЧА-ПРИЙОМ

- ДАНІ ПЕРЕДАЮТЬСЯ ПО ОДНОМУ БІТУ
- ПРИЙМАЧ І ПЕРЕДАВАЧ НАЛАШТОВУЮТЬСЯ НА ОДНАКОВУ ШВИДКІСТЬ І ОДНАКОВІ ОСОБЛИВОСТІ ПРОТОКОЛУ ПЕРЕДАЧІ
- БІТИ КОДУЮТЬСЯ РІВНЕМ 0- НИЗЬКИЙ 1 –ВИСОКИЙ
- ПЕРЕДАЧА КОЖНОГО БІТУ ВІДБУВАЄТЬСЯ ВПРОДОВЖ ОДНАКОВОГО ФІКСОВАНОГО ПРОМІЖКУ ЧАСУ
  - ЧАС ПЕРЕДАЧІ БІТА = 1/(ШВИДКІСТЬ ПЕРЕДАЧІ)
- ДАНІ ПЕРЕДАЮТЬСЯ ПОРЦІЯМИ ПО 1 БАЙТУ
  - 1БАЙТ = 8-9 БІТ (НАЛАШТОВУ)
- ПЕРЕД ПОРЦІЄЮ ДАНИХ МОЖУТЬ БУТИ СТАРТОВІ БІТИ (ЗАЖДИ 0)
- ПІСЛЯ ПЕРЕДАЧІ МОЖУТЬ БУТИ БІТИ ПАРНОСТІ І СТОПОВІ БІТИ (ЗАВЖДИ 1)

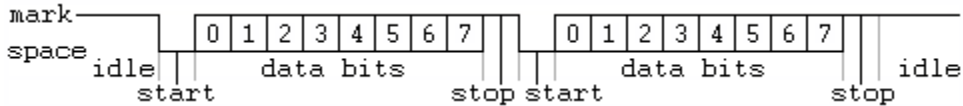


[5.20]

### UART – ПЕРЕДАВАЧ

- В РЕЖИМІ КОЛИ НИЧОГО НЕ ПЕРЕДАЄТЬСЯ ПЕРЕДАВАЧ ВИДАЄ НА ЛІНІЮ ВИСОКИЙ РІВЕНЬ (1)
- ПЕРЕДАЧА БАЙТА ПОЧИНАЄТЬСЯ З ПЕРЕДАЧІ СТАРТОВОГО БІТУ (0)
  - ПЕРЕДАВАЧ ПЕРЕМИКАЄТЬСЯ З ВИСОКОГО НА НИЗЬКИЙ РІВЕНЬ
- ПЕРЕДАВАЧ ЧЕКАЄ ВПРОДОВЖ ЧАСУ ПЕРЕДАЧІ БІТУ

- ПЕРЕДАВАЧ ПЕРЕКИДАЄ НА ЛІНІЮ У ВИСОКИЙ ЧИ НИЗЬКИЙ СТАН В ЗАЛЕЖНОСТІ ВІД ЗНАЧЕННЯ БІТІВ ДАНИХ, ЯКІ ПЕРЕДАЮТЬСЯ І ЧЕКАЄ ВПРОДОВЖ ЧАСУ ПЕРЕДАЧІ БІТІВ
  - ТАК 8 АБО 9 РАЗІВ
- ПІСЛЯ ПЕРЕДАЧІ ОСТАННЬОГО БІТА З БАЙТА ДАНИЙ ПЕРЕДАВАЧ ПЕРЕДАЄ СТОПОВИЙ БІТ
  - ПЕРЕКИДАЄ ЛІНІЮ У ВИСОКИЙ СТАН І ЧЕКАЄ ЧАС ПЕРЕДАЧІ БІТУ
- НАСТУПНИЙ БАЙТ ПЕРЕДАЄТЬСЯ АНАЛОГІЧНО



[5.21]

#### UART-ПРИЙМАЧ

- ПРИЙМАЧ ЧЕКАЄ ПЕРЕПАДУ З ВИСОКОГО НА НИЗЬКИЙ РІВЕНЬ
- КОЛИ ТАКИЙ ПЕРЕПАД ТРАПЛЯЄТЬСЯ ПРИЙМАЧ ЧЕКАЄ ПОЛОВИНУ ЧАСУ ПЕРЕДАЧІ БІТУ
  - ЯКЩО ВСЕ ЩЕ НУЛЬ (СТАРТОВИЙ БІТ), ТО ПРИЙМАЄ ДАЛІ
- ПРИЙМАЧ ЧЕКАЄ ЧАС ПЕРЕДАЧІ БІТУ І ЗЧИТУЄ ПОТОЧНЕ ЗНАЧЕННЯ БІТУ
  - ПОВТОРЮЄ ТАК 8-9 РАЗІВ І ЗАПИСУЄ ПРИЙНЯТІ ДАНІ
- ПРИЙМАЧ ЧЕКАЄ ЧАС ПЕРЕДАЧІ БІТУ І ПЕРЕВІРЯЄ НАЯВНІСТЬ 1
- ЯКЩО СТАРТОВИЙ АБО СТОПОВИЙ БІТ НЕПРАВИЛЬНІ – ПОМИЛКА
- ДАЛІ ВСЕ ПОВТОРЮЄТЬСЯ



[5.22]

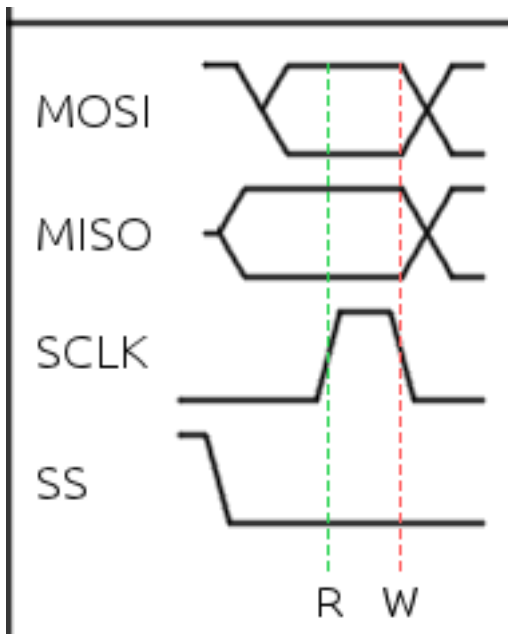
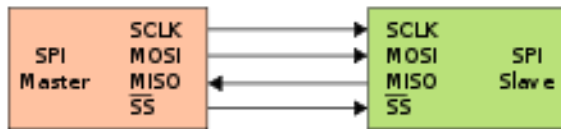
#### ШВИДКІСТЬ ПЕРЕДАЧІ

- ПРИЙМАЧ І ПЕРЕДАВАЧ ПОВИННІ МАТИ ТАКТОВІ ГЕНЕРАТОРИ З ПРИБЛИЗНО ОДНАКОВОЮ ЧАСТОТОЮ
  - ТОЧНІСТЬ ТАКА, ЩО ЗА ПЕРІОД ПЕРЕДАЧІ ДАНИХ НЕ ПОВИННА НАБІГТИ ПОХИБКА БІЛЬША, НІЖ 0.5 ЧАСУ ПЕРЕДАЧІ БІТА
  - 0.5БІТ/9.5 БІТ~5% (РЕАЛЬНО 1-3 %)
  - ЦЕ ОБМЕЖУЄ МАКСИМАЛЬНУ ШВИДКІСТЬ, БО СКЛАДНІШЕ ЗАБЕЗПЕЧИТИ СТАБІЛЬНІСТЬ
- ШВИДКІСТЬ ПЕРЕДАЧІ (БОД, BAUD RATE)
  - КІЛЬКІСТЬ ФІЗИЧНИХ БІТІВ ЗА СЕКУНДУ
  - ШВИДКІСТЬ ПЕРЕДАЧІ КОРИСНИХ ДАНИХ МЕНША
- Є СТАНДАРТНІ ШВИДКОСТІ
  - 300, 600, 1200, 2400, 4800, 9600, 19200, 38400, 57600, 115200, 230400, 460800, 921600
- Є СТАНДАРТИ НА РІВНІ НАПРУГ І ПРОТОКОЛИ
  - RS-232, RS-485, LIN, IRDA

[5.23]

#### ШИНА SPI (ДО 50 МГц)

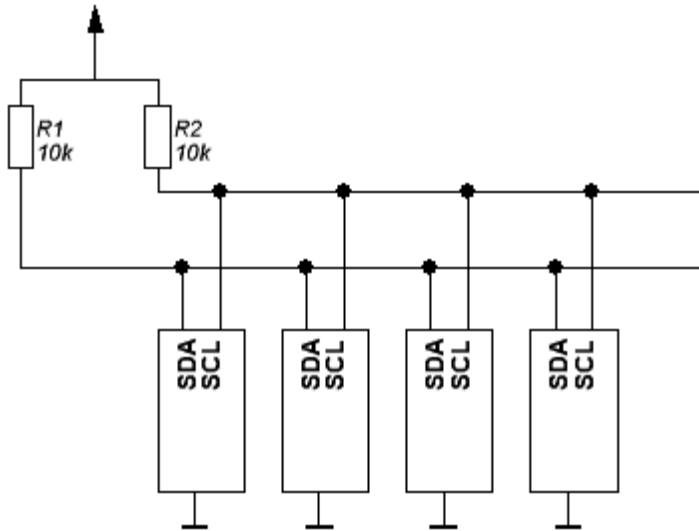
- 1 ГОЛОВНИЙ - БАГАТО ВЕДЕНИХ
  - 4 ПРОВОДИ
- ГОЛОВНИЙ ВИДАЄ ТАКТОВИЙ СИГНАЛ НА ВСІ ВЕДЕНІ ПРИСТРОЇ
- ЗА ДОПОМОГОЮ ВИВІДУ SS=0 ГОЛОВНИЙ ВИБИРАЄ ТІ ПРИСТРОЇ З ЯКИМИ ВІД ВЗАЄМОДІЄ
- ПЕРЕДАЧА ДАНИХ ДВОНАПРАВЛЕНА
  - ГОЛОВНИЙ З ВИВІДУ MOSI НА ВИВІД MISO ВЕДЕНОГО
  - ВЕДЕНИЙ З ВИВІДУ MISO НА ВИВІД MOSI ГОЛОВНОГО
- ВИВЕДЕННЯ ДАНИХ ПО СПАДУТАКТОВОГО ІМПУЛЬСУ
- ВВЕДЕННЯ ДАНИХ ПО ФРОНТУ ТАКТОВОГО ІМПУЛЬСУ



[5.24]  
2

#### I<sup>2</sup>C - INTER-INTEGRATED CIRCUIT (10 КБІТ/С-3.4 МБІТ/С)

- БАГАТО ГОЛОВНИХ І БАГАТО ВЕДЕНИХ ПРИСРОЇВ
- 3 ПРОВІДИ
  - SDA – ДАНІ
  - SCL- ТАКТ
- ВІДКРИТИЙ КОЛЕКТОР, АБО СТІК
  - НИЗЬКИЙ РІВЕНЬ НА ШИНІ ВИГРАЄ
- ТАКТОВИЙ СИГНАЛ ВИДАЄТЬСЯ ЛИШЕ ГОЛОВНИМИ, А ЧИТАЄТЬСЯ ВСІМА
- ДАНІ ВИДАЮТЬСЯ ВСІМА, ЧИТАЮТЬСЯ ВСІМ
- АРБІТРАЖ – КОЛИ ДЕКІЛЬКА ОДНОЧАСНО
  - ПОКИ ВСІ ДАНІ ОДНАКОВІ – НОРМАЛЬНО
  - ВІДКЛЮЧАЄТЬСЯ ТОЙ, ХТО БАЧИТЬ НЕ СВОЇ ДАНІ



[5.25] 2

- ІС – ПЕРЕДАЧА -ПРИЙОМ ЗАПИС НА ШИНУ – ПРИ НИЗЬКОМУ ТАКТІ
- ЧИТАННЯ ШИНИ – ПРИ ВИСОКОМУ ТАКТІ
- СТАРТ – ПЕРЕХІД СТАРТУ І ДАНИХ З ВИСОКОГО В НИЗЬКЕ ЗНАЧЕННЯ
- СТОП – ПРИ ВИСОКОМУ ТАКТІ ВІДБУВАЄТЬСЯ ЗАПИС НА ШИНУ СПОЧАТКУ НИЗЬКОГО А ПОТІМ ВИСОКОГО ЗНАЧЕННЯ
- МОЖНА РЕГУЛЮВАТИ ШВИДКІСТЬ ЗМІНЮЮЧИ ТРИВАЛІСТЬ НУЛЯ НА ТАКТОВОМУ ПРОВІДІ
- ПОТУЖНИЙ ПРОТОКОЛ, ШИРОКО ЗАСТОСОВУЄТЬСЯ



[5.26]

#### ШИНА USB UNIVERSAL SERIAL BUS

- USB 1.X, 2.X, 3.X
  - 4 ПРОВІДИ
  - ШВИДКІСТЬ 1,5 МБІТ/С- 4,8 ГБІТ/С).
  - СТРУМ ДО 0.5 А
  - ДАЛЬНІСТЬ 3-5 М
- ГАРЯЧЕ ПІДКЛЮЧЕННЯ-ВІДКЛЮЧЕННЯ
  - СПОЧАТКУ ВІДКЛЮЧЮЮТЬСЯ ДВА ПРОВІДИ ДАНИХ, ПОТІМ ДВА ПРОВІДИ ЖИВЛЕННЯ
- ІНІЦІАТОРОМ ЗАВЖДИ Є ХОСТ – КОМП'ЮТЕР ТОЩО
- ДАНЫ ПЕРЕДАЮТЬСЯ СПОСОБОМ СХОЖИМ НА UART
  - ВИКОРСТОВУЄТЬСЯ ДЕКІЛЬКА РІВНІВ СИГНАЛУ

## ЛЕКЦІЯ 06

### ЗАГАЛЬНА ІНФОРМАЦІЯ ПРО МІКРОКОНТРОЛЕРИ

## ЛЕКЦІЯ 6

СУДАКОВ О.О, РАДЧЕНКО С.П.

«СУЧАСНА МІКРОПРОЦЕСОРНА ТЕХНІКА»

[6.2]

МІКРОКОНТРОЛЕР (MICRO CONTROL UNIT, MCU,  $\mu$ C, UC)

- **МІКРОКОНТРОЛЕР** – АНАЛОГО-ЦИФРОВИЙ ОДНОКРИСТАЛЬНИЙ КОМП'ЮТЕР ПРИЗНАЧЕНИЙ ДЛЯ КЕРУВАННЯ ПРИСТРОЯМИ
- **МІСТИТЬ**
  - **ПРОЦЕСОР**
  - ЕНЕРГОЗАЛЕЖНУ ПАМ'ЯТЬ
  - ЕНЕРГОНЕЗАЛЕЖНУ ПАМ'ЯТЬ
  - **ПЕРИФЕРІЮ**
- **ПЕРИФЕРІЯ**
  - ГЕНЕРАТОРИ ТАКОВИХ СИГНАЛІВ
  - ТАЙМЕРИ (ЛІЧИЛЬНИКИ, ВАРТОВІ ТОЩО)
  - КОМПАРАТОРИ
  - ПІДСИЛЮВАЧІ
  - ЦИФРОВІ ПОРТИ ВВОДУ-ВИВОДУ (GPIU)
  - АЦП
  - ЦАП
  - ДАТЧИКИ (ТЕМПЕРАТУРИ, ТИСКУ ТОЩО)
  - КОНТРОЛЕРИ ПДП (DMA)
  - ПОСЛІДОВНІ ШИНИ (UART, SPI, I2C ТОЩО)
  - КОНТРОЛЕР ІНДИКАТОРА ...

[6.3]

ЗАСТОСУВАННЯ MCU

- ВБУДОВАВАНІСИСТЕМИ (EMBEDDED SYSTEMS)
- КЕРУВАННЯ ПРИСТРОЯМИ
  - ПРАЛЬНА МАШИНА (ПРОГРАМА ПРАННЯ)
  - МІКРОХВИЛЬОВА ПІЧКА (ПРОГРАМА ПРИГОТУВАННЯ)
  - ДВИГУН АВТОМОБІЛЯ (КЕРУВАННЯ РОЗПОДІЛОМ ЗАПАЛЮВАННЯ, ІНЖЕКТОРАМИ)
  - ПІДВІСКА І ТРАНСМІСІЯ АВТОМОБІЛЯ (АВТОБЛОКУВАННЯ ДИФЕРЕНЦІАЛА, АВТОРОЗПОДІЛ МОМЕНТУ НА ПЕРЕДНЮ ТА ЗАДНЮ ОСІ ...)
  - МЕДИЧНІ ПОРТАТИВНІ ПРИСТРОЇ (ВИМІРЮВАЧІ ТИСКУ, ТЕМПЕРАТУРИ, РІВНЯ ГЛЮКОЗИ, СЕРЦЕВІ СТИМУЛЯТОРИ ...)
  - ДОЗИМЕТРИ
  - ВИМІРЮВАЛЬНІ ПРИСТРОЇ (МІЛЬТИМЕТРИ, ОСЦИЛОГРАФИ)
  - ЛАБОРАТОРНЕ ОБЛАДНАННЯ (СПЕКТРОМЕТРИ, ТЕРЕЗИ, ...)
  - ТОМОГРАФИ (КЕРУВАННЯ ПЕРИФЕРІЄЮ)
- ЯКИМ ЧИНОМ КЕРУЄ
  - ПРОЦЕСОР ВМИКАЄ РІЗНІ ПЕРИФЕРІЙНІ ПРИСТРОЇ
  - ПЕРИФЕРІЙНІ ПРИСТРОЇ ЗАПИСУЮТЬ СИГНАЛИ З ДАТЧИКІВ
  - ПРОЦЕСОР "АНАЛІЗУЄ" СИГНАЛИ З ДАТЧИКІВ
  - ПРОЦЕСОР ЗМІНЮЄ РЕЖИМ РОБОТИ ПЕРИФЕРІЙНИХ ПРИСТРОЇВ, ЩОБ РЕАГУВАТИ НА ЗМІНУ ВХІДНИХ ДАНИХ

[6.4]

ТИПИ ТА ХАРАКТЕРИСТИКИ MCU

- **АРХІТЕКТУРА (ЯДРО)**
  - СВОЯ СИСТЕМА КОМАНД
  - СВОЇ ОСОБЛИВОСТІ КЕРУВАННЯ ПЕРИФЕРІЄЮ
- **НАБІР ПЕРИФЕРІЇ (ПРИЗНАЧЕННЯ)**



- ШИРОКИЙ ДІАПАЗОН (GPIO, ADC, DAC,... )
- **ДІАПАЗОН ТАКТОВИХ ЧАСТОТ**
  - ТАКТОВОГО ГЕНЕРАТОРА (0...)
  - ПРОЦЕСОРА (0-300 МГц)
  - ПЕРИФЕРІЇ (0-...)
  - ПРОДУКТИВНІСТЬ (MFLOPS, MIPS)
- **РОЗРЯДНІСТЬ**
  - ПРОЦЕСОРА (8-16 БІТ)
  - ПЕРИФЕРІЇ (8-32 БІТ)
- **НАПРУГА ЖИВЛЕННЯ**
  - 5,3,3,1,8,1,2,0,9 В
- **ПОТУЖНІСТЬ**
  - Вт ( $10^{-7}$  -  $10^0$ ) ДЛЯ ПОТУЖНИХ Вт/МГц (ДО  $10^0$ )

[6.5]

#### ДЕЯКІ АРХІТЕКТУРИ MCU

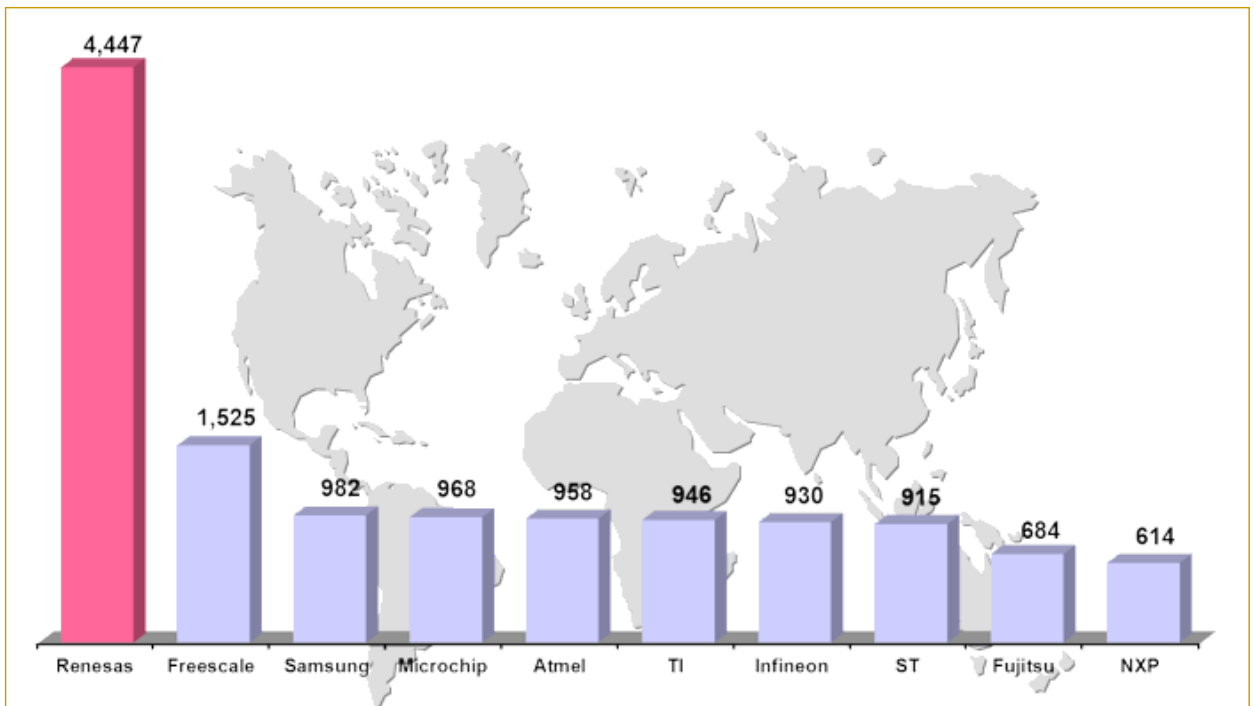
(ЯК ПРАВИЛО СТОСУЄТЬСЯ ПРОЦЕСОРА)

- **ARM** – (ACORN RISC MACHINE)
  - РОЗРОБНИК АРХІТЕКТУРИ ARM HOLDINGS
  - БАГАТО ВИРОБНИКІВ ПРОЦЕСОРІВ
  - ЯК ПРАВИЛО МІКРОКОНТРОЛЕРИ З ПОТУЖНИМ ПРОЦЕСОРОМ
- **MIPS** (MICROPROCESSOR WITHOUT INTERLOCKED PIPELINE STAGES)
  - ПОТУЖНІ ПРОЦЕСОРИ
  - ДЕКІЛЬКА ВИРОБНИКІВ
- **INTEL 8051**
  - КЛАСИЧНА АРХІТЕКТУРА МІКРОКОНТРОЛЕРІВ, ДЕКІЛЬКА ВИРОБНИКІВ
- **ATMEL AVR** (ALF-EGIL BOGEN AND VEGARD WOLLAN RISC)
  - ПЕРШІ ВИКОРИСТАЛИ FLASH ПАМ'ЯТЬ
- **TEXAS INSTRUMENTS MSP430** (MIXED SIGNAL PROCESSOR)
  - НАДНИЗЬКА ПОТУЖНІСТЬ СПОЖИВАННЯ
- **POWERPC** (PERFORMANCE OPTIMIZATION WITH ENHANCED RISC – PERFORMANCE COMPUTING)
- **PARALLAX PROPPELLER** (ДЛЯ ВІДЕОІГОР)
- **V850** (RENESAS)
- **MOTOROLA 6800 I 68000** (M68K)

[6.6]

#### ВИРОБНИКИ МІКРОКОНТРОЛЕРІВ

- **RENESAS**
  - 32BIT : RX, V850, SUPERH, R32C, H8SX
  - 16BIT: RL78, H8, M16C, R8C, 78K0R
  - 8BIT : 78K0
- **FREESCALE**
  - 8,16,32 БІТ MOTOROLA 6800 I 68000
- **SAMSUNG**
- **MICROCHIP**
  - PIC 8,12,13,16,24,32 БІТ
- **ATMEL**
  - AVR 8 I 32 БІТ
- **TEXAS INSTRUMENTS**
  - ARM TMS320 C2000 32 БІТ
  - MSP430 16 БІТ

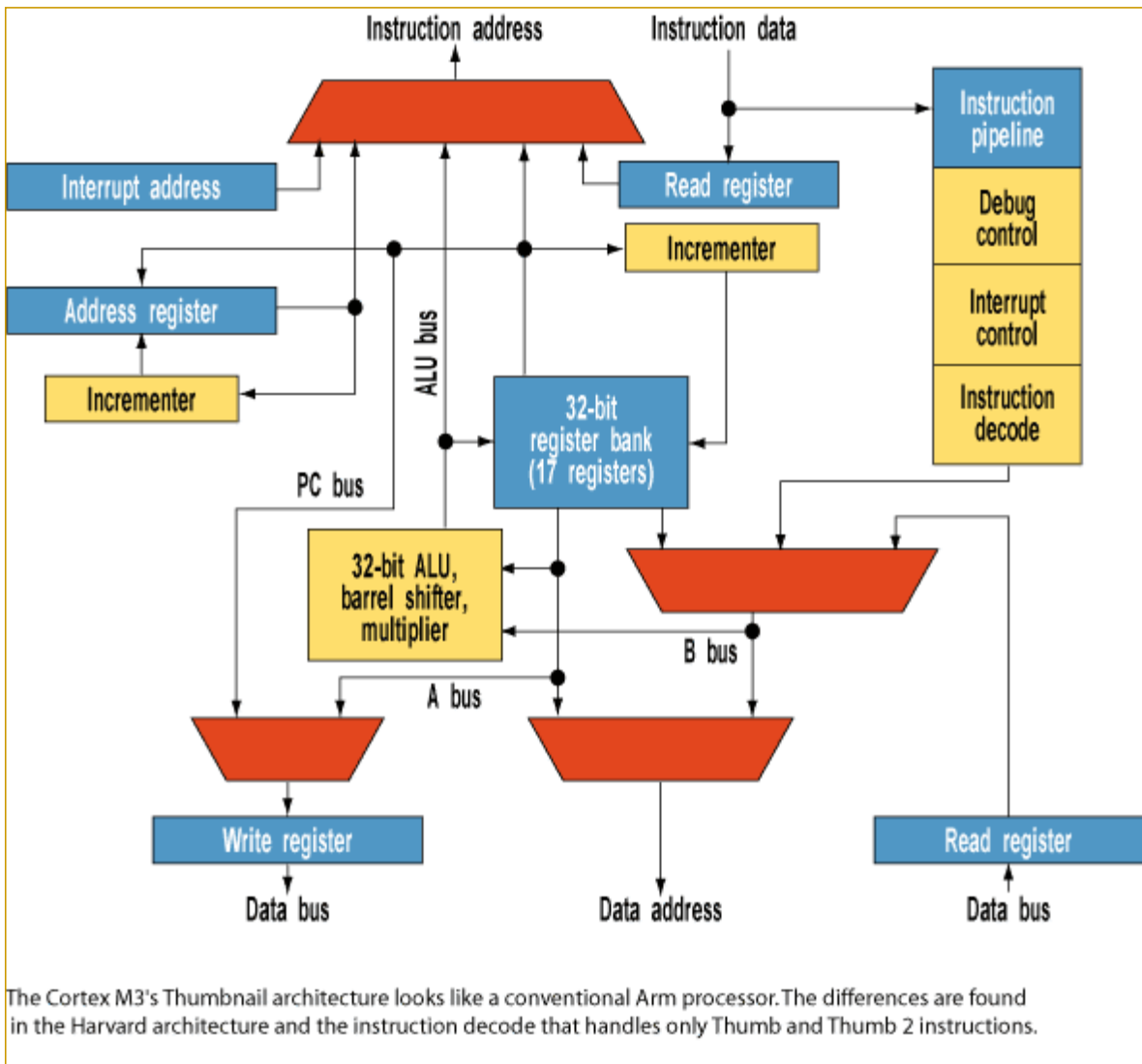


[6.7]

ARM-ВИСОКОПРОДУКТИВНІ МІКРОКОНТРОЛЕРИ ДЛЯ ШИРОКОГО КОЛА

ЗАДАЧ [HTTP://INFOCENTER.ARM.COM](http://infocenter.arm.com)

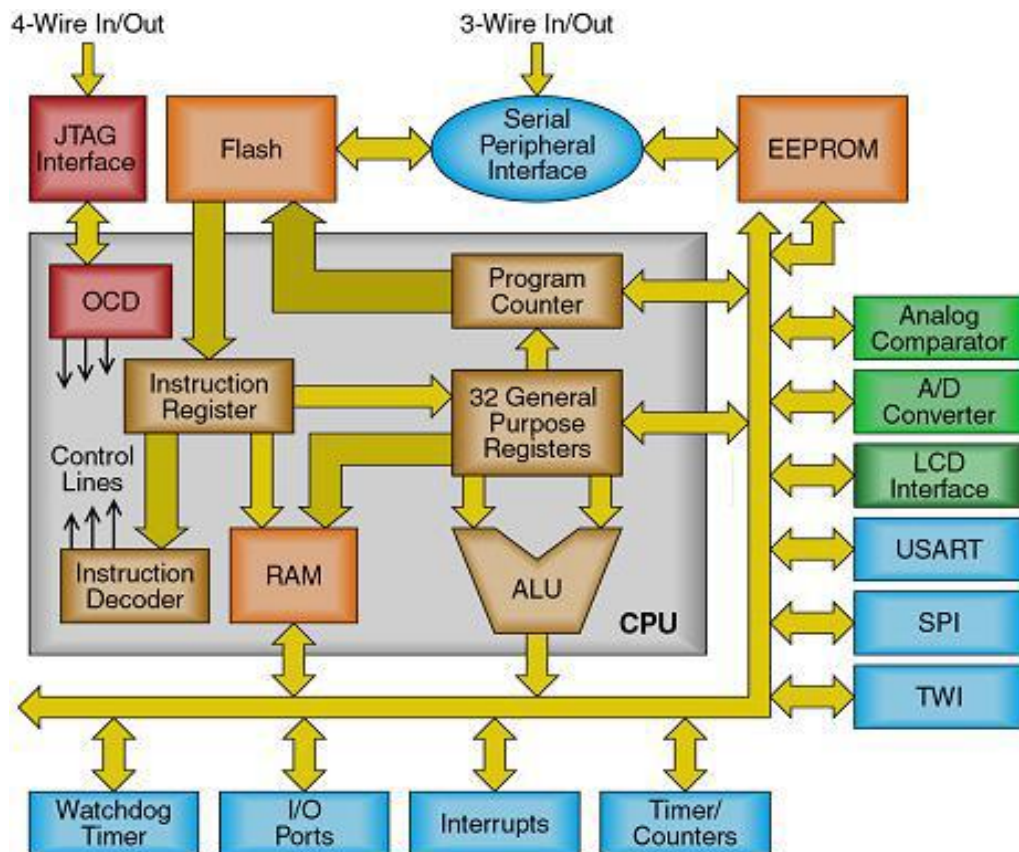
- НАЙБІЛЬШЕ ВИРОБНИКІВ
- RISC 32 БІТИ
- 17 РЕГІСТРІВ ЗАГАЛЬНОГО ПРИЗНАЧЕННЯ
- ГАРВАРДСЬКА АРХІТЕКТУРА
- ОКРЕМИЙ КОНТРОЛЕР ПЕРЕРИВАНЬ



[6.8]

AVR-МІКРОКОНТРОЛЕРИ ДЛЯ ШИРОКОГО КОЛА ЗАДАЧ [HTTP://ATMEL.COM](http://atmel.com)

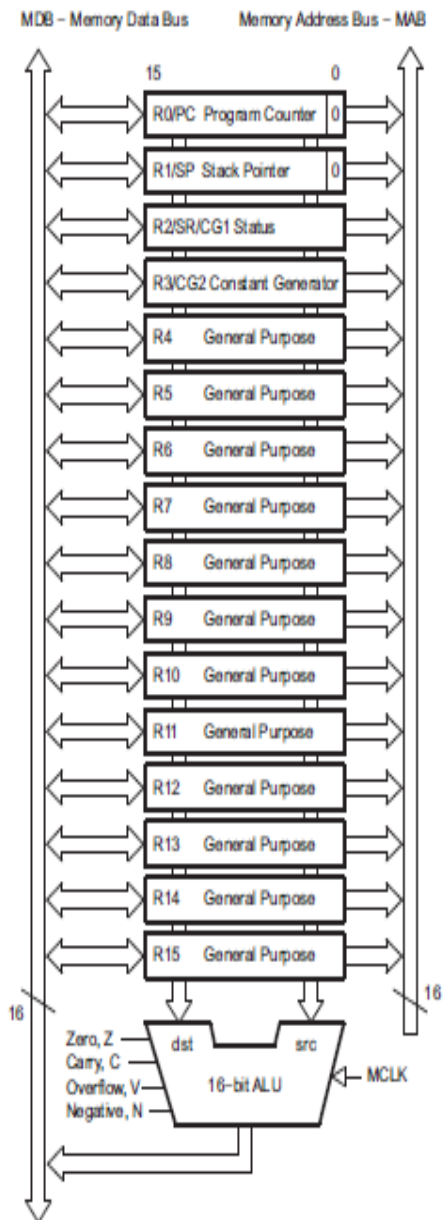
- ВИРОБНИК ATMEL
  - ATMEGA
  - ATTINY
  - ..
- RISC ПРОЦЕСОР
  - 8 АБО 32 БІТИ
- 16-32 РЕГІСТРА ЗАГАЛЬНОГО ПРИЗНАЧЕННЯ
- ГАРВАРДСЬКА АРХІТЕКТУРА



[6.9]

#### MSP430-МІКРОКОНТРОЛЕРИ З НАДНИЗЬКИМ ЕНЕРГОСПОЖИВАННЯМ

- ВИРОБНИК TEXAS INSTRUMENTS
  - [HTTP//TI.COM](http://ti.com)
- RISC ПРОЦЕСОР 16 БІТ
- 16 РЕГІСТРІВ 16 БІТ
- ФОННЕЙМАНІВСЬКА АРХІТЕКТУРА
- ДЛЯ МОБІЛЬНИХ ПРИСРОІВ
  - СЕРЦЕВІ СТИМУЛЯТОРИ
  - ДОЗИМЕТРИ
  - ВИМІРЮВАЧІ ГЛЮКОЗИ



[6.10]

## ОСНОВНІ КОМПОНЕНТИ МІКРОКОНТРОЛЕРІВ

- **ГЕНЕРАТОРИ**
  - ГЕНЕРАЦІЯ ТАКОВИХ СИГНАЛІВ РІЗНОГО ПРИЗНАЧЕННЯ
- **ПРОЦЕСОР**
  - RISC – ПРОСТІ ОДНОТИПНІ КОМАНДИ
  - ОПТИМІЗАЦІЯ ДЛЯ ОБРОБКИ ПЕРЕРИВАНЬ
  - МІНІМАЛЬНІ ЗАТРИМКИ
  - МАЛА ПРОДУКТИВНІСТЬ
- **ПАМ'ЯТЬ**
  - ОПЕРАТИВНА (STATIC RAM)
  - ПОСТІЙНА (ROM)
  - ПЕРЕПРОГРАМОВАНА (FLASH, EEPROM)
- **ГРІО** - ЦИФРОВИЙ ВВІД-ВИВІД
  - – ВИВЕДЕННЯ ВИСОКОГО, АБО НИЗЬКОГО РІВНЯ НА «НІЖКИ» МІКРОСХЕМИ
  - ВВЕДЕННЯ ВИСОКОГО, АБО НИЗЬКОГО РІВНЯ З «НІЖОК» МІКРОСХЕМИ
- **ТАЙМЕРИ**
  - ВИМІРЮВАННЯ ЧАСОВИХ ІНТЕРВАЛІВ
- **КЕРУВАННЯ ЖИВЛЕННЯМ**
  - РЕАКЦІЯ НА ВТРАТУ ЖИВЛЕННЯ
- **АНАЛОГОВІ ПРИСТРОЇ**

- ❑ АЦП, КОМПАРАТОРИ, ПІДЧИЛЮВАЧІ

[6.11]

#### ПРОЦЕСОР МІКРОКОНТРОЛЕРА-ВИМОГИ

- **ВИКОНАННЯ ПРОСТИХ ПРОГРАМ**
  - ❑ ОСНОВНА ФУНКЦІОНАЛЬНІСТЬ ЛЕЖИТЬ НА ПЕРИФЕРІЇ
  - ❑ ВЕЛИКОГО ОБ'ЄМУ ПАМ'ЯТІ НЕ ТРЕБА
  - ❑ ВИСОКА ПРОДУКТИВНІСТЬ НЕ ОБОВ'ЯЗКОВА
- **ШВИДКА РЕАКЦІЯ НА ПОДІЇ** ВІД ПЕРИФЕРІЇ
  - ❑ МАЛІ ЗАТРИМКИ В ОБРОБЦІ ПЕРЕРИВАНЬ
  - ❑ ВЕЛИКА КІЛЬКІСТЬ РЕГІСТРІВ ЗАГАЛЬНОГО ПРИЗНАЧЕННЯ
- **ПРОСТИЙ НАБІР КОМАНД**, ЯКІ ШВИДКО ВИКОНУЮТЬСЯ
  - ❑ RISC – REDUCED INSTRUCTION SET
  - ❑ КОМАНДА ВИМАГАЄ МАЛО ТАКТИВ
- **НАЙЧАСТІШЕ ГАРВАРДСЬКА АРХІТЕКТУРА**
  - ❑ КОМАНДИ І ДАНІ В РІЗНІЙ ПАМ'ЯТІ І ПЕРЕДАЮТЬСЯ ПО РІЗНИМ ШИНАМ
  - ❑ МЕНШЕ ТАКТИВ НА ОБРОБКУ КОМАНД З АРГУМЕНТАМИ
  - ❑ **НЕДОЛІКИ: ВАЖКО ПЕРЕПРОГРАМОВУВАТИ ПРОГРАМУ І ДАНІ**
- **ШВИДКА ЗУПИНКА І ВІДНОВЛЕННЯ РОБОТИ**
  - ❑ РОБОТА З РІЗНОЮ ТАКТОВОЮ ЧАСТОТОЮ

[6.12]

- **РЕЖИМИ РОБОТИ ПРОЦЕСОРА ПРОГРАМУВАННЯ ПЕРИФЕРІЇ**
  - ❑ ЗАПУСК ПЕРИФЕРІЙНИХ ПРИСТРОЇВ НА ВИКОНАННЯ
- **ОБРОБКА ПОДІЙ**
  - ❑ ОПИТУВАННЯ (POLLING)
  - ❑ ПЕРЕРИВАННЯ (INTERRUPTING)
- **ОПИТУВАННЯ (СИНХРОННИЙ РЕЖИМ)**
  - ❑ ПРОЦЕСОР ПОСТІЙНО ПРАЦЮЄ І ПЕРЕВІРЯЄ СТАН ПРИСТРОЇВ
  - ❑ ВИСОКА ПРОДУКТИВНІСТЬ
  - ❑ ПРОСТОТА ПРОГРАМУВАННЯ
  - ❑ ВИСОКІ ЕНЕРГОЗАТРАТИ
- **ПЕРЕРИВАННЯ (АСИНХРОННИЙ РЕЖИМ)**
  - ❑ ПРОЦЕСОР ЗУПИНЯЄТЬСЯ, АБО ВИКОНУЄ ОПИТУВАННЯ
  - ❑ КОЛИ НАСТУПАЄ ПОДІЯ РОБОТА ПРОЦЕСОРА ПЕРЕРИВАЄТЬСЯ І ВИКЛИКАЄТЬСЯ ФУНКЦІЯ-ОБРОБНИК ПОДІЇ
  - ❑ СКЛАДНІШЕ ПРОГРАМУВАННЯ
  - ❑ НИЗЬКІ ЕНЕРГОЗАТРАТИ
  - ❑ ВИСОКА ШВИДКІСТЬ РЕАКЦІЇ НА ПОДІЇ
  - ❑ НИЖЧА ПРОДУКТИВНІСТЬ

```
VOID SETUP()
```

```
{  
  PINMODE(PIN, OUTPUT);  
  ATTACHINTERRUPT(0, BLINK..);  
}
```

```
VOID LOOP()
```

```
{  
  DIGITALWRITE(PIN, STATE);  
}
```

```
VOID BLINK()
```

```
{  
  STATE = !STATE;  
}
```

- **Програмування периферії**
  - Запуск периферійних пристроїв на виконання
- **Обробка подій**
  - Опитування (polling)
  - Переривання (interrupting)
- **Опитування (синхронний режим)**
  - Процесор постійно працює і перевіряє стан пристроїв
  - Висока продуктивність
  - Простота програмування
  - Високі енергозатрати
- **Переривання (асинхронний режим)**
  - Процесор зупиняється, або виконує опитування
  - Коли настає подія робота процесора переривається і викликається функція-обробник події
  - Складніше програмування
  - Низькі енергозатрати
  - Висока швидкість реакції на події
  - Нижча продуктивність

```

void setup()
{
  pinMode(pin, OUTPUT);
  attachInterrupt(0, blink..);
}

void loop()
{
  digitalWrite(pin, state);
}

void blink()
{
  state = !state;
}

```

[6.13]

#### КЕРУВАННЯ ПЕРИФЕРІЄЮ

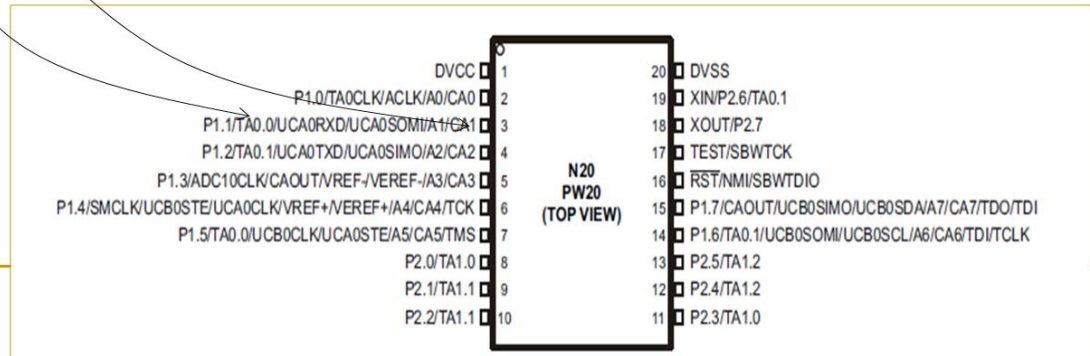
- ПІДКЛЮЧЕННЯ/ВІКЛЮЧЕННЯ ПРИСТРОЇВ
  - В СЕРЕДИНІ МІКРОСХЕМИ
  - ДО ЗОВНІШНІХ ВИВОДІВ
  - Є СПЕЦІАЛЬНІ РЕГІСТРИ 1-ВІМКНЕНО 0-ВІМКНЕНО
  - ВИКОНУЄТЬСЯ ПРОЦЕСОРОМ ЗАПИСОМ В РЕГІСТРИ
- ПРОГРАМУВАННЯ ПРИСТРОЇВ
  - ВИБІР ТА ЗМІНА РЕЖИМУ РОБОТИ
  - ЗАПИС ВЕКТОРІВ ПЕРЕРИВАНЬ
  - ЗАПУСК ПРИСТРОЇВ НА ВИКОНАННЯ
  - ОПИТУВАННЯ І ОБРОБКА ПЕРЕРИВАНЬ

[6.14]

#### ПРИКЛАД – ПІДКЛЮЧЕННЯ ДО ВИВОДІВ

- КОЖЕН ВИВІД МІКРОСХЕМИ МОЖЕ БУТИ ПІДКЛЮЧЕНИЙ ДО РІЗНИХ ВИВОДІВ РІЗНИХ ПРИСТРОЇВ
  - ТАЙМЕР
  - ПОРТ ЦИФРОВОГО ВВОДУ-ВИВОДУ
  - КОМПАРАТОР
  - АЦП
- ПОВНА ІНФОРМАЦІЯ – В ДОКУМЕНТАЦІЇ НА МІКРОКОНТРОЛЕР
  - DATASHEET
  - USER GUIDE

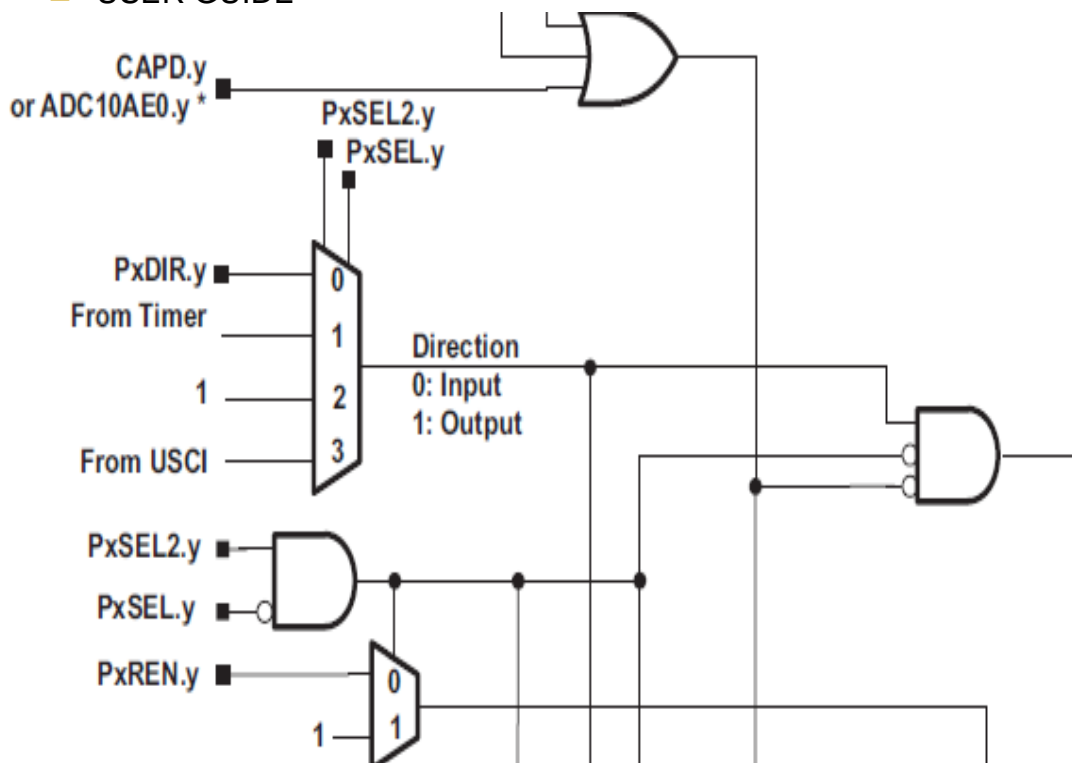
- Кожен вивід мікросхеми може бути підключений до різних виводів різних пристроїв
  - Таймер
  - Порт цифрового вводу-виводу
  - Компаратор
  - АЦП
- Повна інформація – в документації на мікроконтролер
  - Datasheet
  - User guide



[6.15]

#### ПРИКЛАД – ПІДКЛЮЧЕННЯ В СЕРЕДИНІ МІКРОСХЕМИ

- РІЗНІ ПРИСТРОЇ МОЖУТЬ БУТИ З'ЄДНАНІ МІЖ СОБОЮ
- ПОВНА ІНФОРМАЦІЯ В ДОКУМЕНТАЦІЇ НА МІКРОКОНТРОЛЕР
- DATASHEET
- USER GUIDE



[6.16]

#### КЕРУВАННЯ ПІДКЛЮЧЕННЯМИ

- Є СПЕЦІАЛЬНІ РЕГІСТРИ
  - ЧАСТО З ВІДОБРАЖЕННЯМ НА ПАМ'ЯТЬ



- ЗАПИС БІТІВ У КОМІРКУ ПАМ'ЯТІ ЗМІНЮЄ СТАН ПІД'ЄДНАННЯ ЧИ ВІД'ЄДНАННЯ ВИВОДУ
- АДРЕСИ КОМІРОК Є В ДОКУМЕНТАЦІЇ
- ЦІ АДРЕСИ МАЮТЬ ВИЗНАЧЕНІ ІМЕНА
- ПРИКЛАД
  - РЕГІСТР **P1DIR** – НІЖКА НА ВВІД ЧИ НА ВИВІД
    - **P1DIR.0=0** (БІТ0=0) “НІЖКА” 1 (**P1.0**) ДЛЯ ВВОДУ ДАНИХ
  - РЕГІСТР **P1SEL** - ДО ЯКОГО ПРИСРОЮ ПІДКЛЮЧАТИ «НІЖКУ»
    - **P1SEL.0=1** (БІТ0=1) “НІЖКА” 1 (**P1.0**) ПІДКЛЮЧЕНА ДО ВХОДУ ТАЙМЕРА

[6.17]

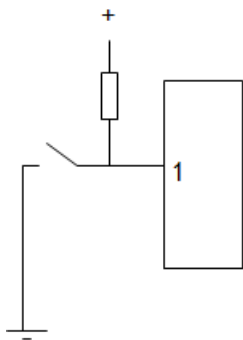
#### ПРОГРАМУВАННЯ ПРИСТРОЇВ

- КОЖЕН ПРИСТРІЙ МАЄ РЕГІСТРИ ВІДОБРАЖЕНІ НА ПАМ'ЯТЬ
  - ЗАПИС БІТІВ У КОМІРКУ ПАМ'ЯТІ ЗМІНЮЄ РЕЖИМ РОБОТИ ПРИСТРОЮ
  - ЧИТАННЯ БІТІВ – ОДЕРЖАННЯ РЕЗУЛЬТАТУ
- ПРИКЛАД АЦП
  - REF2\_5V =1** – ОПОРНА НАПРУГА АЦП 2.5 В
  - SAMPCON =1** ЗАПУСК АЦП
  - ADC10MEM** – МІСТИТЬ РЕЗУЛЬТАТ АНАЛОГО-ЦИФРОВОГО ПЕРЕТВОРЕННЯ

[6.18]

#### ОПИТУВАННЯ ПРИСТРОЇВ

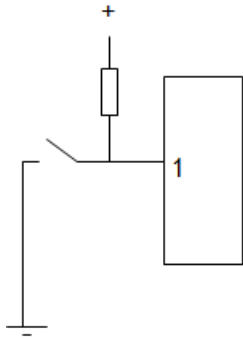
- ПЕРІОДИЧНО ПЕРЕВІРЯЄТЬСЯ СТАН ПРИСТРОЮ
  - НІЖКИ МІКРОКОНТРОЛЕРА
  - РЕГІСТРА
- ПРИКЛАД
  - ОЧІКУВАННЯ НАТИСНЕННЯ НА КНОПКУ
  - КНОПКА ПРИЄДНАНА ДО НІЖКИ 1
  - WHILE (P1IN & 0X01);



[6.19]

#### ПЕРЕРИВАННЯ

- РЕЄСТРАЦІЯ ФУНКЦІЇ –ОБРОБНИКА ПЕРЕРИВАННЯ
- ДОЗВІЛ ПЕРЕРИВАННЯ
- ОЧІКУВАННЯ ПЕРЕРИВАННЯ
- ПРИКЛАД – КНОПКА
  - РЕЄСТРУЄМО ФУНКЦІЮ BLINK ДЛЯ ОБРОБКИ ПЕРЕРИВАННЯ  
ATTACHINTERRUPT(0, BLINK..);
  - ДОЗВОЛЯЄМО ПЕРЕРИВАННЯ P1IE=1
  - ЗУПИНЯЄМО РОБОТУ HLT



[6.20]

#### МАСКОВАНІ, НЕМАСКОВАНІ, ПРІОРИТЕТНІ

- КОЛИ ПІД ЧАС ВИКОНАННЯ ОБРОБНИКА ПЕРЕРИВАННЯ ПРИХОДИТЬ ІНШЕ ПЕРЕРИВАННЯ?
  - МАСКОВАНІ – ЗАБОРОНЯЮТЬСЯ ОДНІ ПЕРЕРИВАННЯ ПІД ЧАС ВИКОНАННЯ ІНШИХ
  - НЕМАСКОВАНІ – НЕ МОЖУТЬ БУТИ ЗАБОРОНЕНІ ІНШИМИ ПЕРЕРИВАННЯМИ (ЗАБОРОНЯЮТЬСЯ ОКРЕМО)
  - БІЛЬШ ВИСОКОПРІОРИТЕТНІ МАСКУЮТЬ БІЛЬШ НИЗЬКОПРІОРИТЕТНІ
- ВИХІД З ЛАДУ ГЕНЕРАТОРА – ВИМАГАЄ НЕГАЙНОЇ РЕАКЦІЇ – НЕМАСКОВАНЕ ПЕРЕРИВАННЯ
- ТАЙМЕР – ПРІОРИТЕТНІШИЙ НІЖ КНОПКА

[6.21]

#### ТАКТОВІ ГЕНЕРАТОРИ (SYSTEM CLOCK)

- ГЕНЕРУВАННЯ ТАКТОВИХ КОЛИВАНЬ
- ДЕКІЛЬКА ГЕНЕРАТОРІВ ДЛЯ ПРОЦЕСОРА І ПЕРИФЕРІЇ
  - ВИСОКОЧАСТОТНІ – ДЛЯ ВИСОКОЇ ПРОДУКТИВНОСТІ
  - НИЗЬКОЧАСТОТНІ – ДЛЯ ЕНЕРГОЗБЕРЕЖЕННЯ
  - ВИСОКОСТАБІЛЬНІ (КВАРЦЕВІ)
- КЕРУВАННЯ ЧАСТОТОЮ В ШИРОКИХ МЕЖАХ
  - ДЛЯ ВИСОКОЧАСТОТНИХ
- ПІДКЛЮЧЕННЯ ГЕНЕРАТОРІВ ДО РІЗНИХ ПРИСТРОЇВ
- ЧАСТО НЕОБХІДНО НАЛАШТОВУВАТИ ГЕНЕРАТОРИ ПЕРЕД РОБОТОЮ

[6.22]

#### КОНТРОЛЬ ЖИВЛЕННЯ

- ЩО РОБИТИ ПРИ ВВІМКНЕННІ МІКРОКОНТРОЛЕРА?
  - ЗАПУСТИТИ ГЕНЕРАТОР
  - ВИКЛИКАТИ ПЕРЕРИВАННЯ ПО ПЕРЕЗАВАНТАЖЕННЮ
- ЩО РОБИТИ ПРИ СТРИБКАХ ЖИВЛЕННЯ
  - BROWN OUT RESET (BOW) – ПЕРЕЗАВАНТАЖИТИ ПРОЦЕСОР
- ЯК ЕКОНОМИТИ ЕЛЕКТРОЕНЕРГІЮ
  - ВИМИКАТИ НЕПОТРІБНІ ПРИСТРОЇ
  - ЗМЕНШУВАТИ ЧАСТОТУ ТАКТОВОГО ГЕНЕРАТОРА ПІД ЧАС “СНУ”

[6.23]

#### GPIO (GENERAL PURPOSE INPUT OUTPUT)

- ЦИФРОВІ ПОРТИ ВВОДУ-ВИВODУ
  - ЗАПИС 0 АБО 1 НА “НІЖКИ” МІКРОКОНТРОЛЕРА

- ЗЧИТУВАННЯ ПОТОЧНОГО СТАНУ “НІЖОК” МІКРОКОНТРОЛЕРА
- ВИКЛИК ПЕРЕРИВАННЯ ПРИ ЗМІНІ СТАНУ “НІЖОК” МІКРОКОНТРОЛЕРА
- ВВІМКНЕННЯ РЕЗИСТОРІВ “ПІДТЯГУВАННЯ”
  - ВИВІД ЧЕРЕЗ РЕЗИСТОР ПІД’ЄДНУЄТЬСЯ ДО + АБО –
- ВИКОРИСТОВУЄТЬСЯ
  - “СПІЛКУВАННЯ” ІЗ ЗОВНІШНІМИ ПРИСТРОЯМИ
  - ШИМ І СИГМА-ДЕЛЬТА ЦАП

[6.24]

#### ТАЙМЕРИ

- ТАЙМЕР – ЛІЧИЛЬНИК ТАКОВИХ ІМПУЛЬСІВ
  - СКІЛЬКИ ІМПУЛЬСІВ ЗА ПЕВНИЙ ЧАС
  - СКІЛЬКИ ІМПУЛЬСІВ МІЖ ПЕВНИМИ ПОДІЯМИ
  - ЗАПИС ЗНАЧЕННЯ ЛІЧИЛЬНИКА В РЕГІСТР ПРИ НАСТАННІ ПЕВНОЇ ПОДІЇ
  - ЗМІНА ЗНАЧЕННЯ НА “НІЖЦІ” МІКРОСХЕМИ ПРИ ДОСЯГНЕННЯ ТАЙМЕРОМ ПЕВНОГО ЗНАЧЕННЯ
- ПРОЦЕСОР НЕ МОЖЕ ТОЧНО ВІДМІРЯТИ ЧАСОВІ ІНТЕРВАЛИ
  - РІЗНІ КОМАНДИ МАЮТЬ РІЗНИЙ ЧАС ВИКОНАННЯ
  - НАБЛИЖЕНО МОЖНА ЗДІЙСНЮВАТИ ЗАТРИМКУ ЦИКЛОМ
  - ТАЙМЕР – НЕЗАЛЕЖНИЙ ВІД ПРОЦЕСОРА ПРИСТРІЙ КОНТРОЛЮ ЧАСУ

[6.25]

#### РЕЖИМИ ТАЙМЕРІВ

- РЕЖИМ ЗАХОПЛЕННЯ (CAPTURE)
  - ПРИ НАСТАННІ ПЕВНОЇ ПОДІЇ ЗНАЧЕННЯ ЛІЧИЛЬНИКА ЗАПИСУЄТЬСЯ В СПЕЦІАЛЬНИЙ РЕГІСТР
  - ВИКЛИКАЄТЬСЯ ПЕРЕРИВАННЯ
  - ДЛЯ ТОЧНОГО СЛІДКУВАННЯ ЗА ЗОВНІШНІМИ ПОДІЯМИ
- РЕЖИМ ПОРІВНЯННЯ (COMPARE)
  - ПРИ ДОСЯГНЕННІ ТАЙМЕРОМ ЗНАЧЕННЯ В СПЕЦІАЛЬНОМУ РЕГІСТРІ ЗМІНЮЄТЬСЯ НАПРУГА НА ВИВОДІ ТАЙМЕРА
  - ВИКЛИКАЄТЬСЯ ПЕРЕРИВАННЯ
  - ДЛЯ ТОЧНОГО КЕРУВАННЯ ЗОВНІШНІМИ ПРОЦЕСАМИ

[6.26]

#### ПЕРЕРИВАННЯ ТАЙМЕРІВ

- ПО ПЕРЕПОВНЕННЮ
  - ДЛЯ ВРАХУВАННЯ ХОДУ ЧАСУ ПРОЦЕСОРОМ
- ПО ЗАХОПЛЕННЮ
  - ДЛЯ ЗБЕРЕЖЕННЯ ПРОЦЕСОРОМ МОМЕНТУ ЧАСУ НАСТАННЯ ПОДІЇ
- ПО ПОРІВНЯННЮ
  - ДЛЯ ПОВІДОМЛЕННЯ ПРОЦЕСОРУ, ЩО ЗАПЛАНОВАНА ДІЯ ВИКОНАНА
- АСИНХРОННИЙ ПРОЦЕСОР МОЖЕ КЕРУВАТИ СИНХРОННИМИ ПОДІЯМИ

[6.27]

#### ВИКОРИСТАННЯ ТАЙМЕРІВ

- ВИМІРЮВАННЯ ТРИВАЛОСТІ І ЧАСТОТИ СИГНАЛІВ
  - СКІЛЬКИ ІМПУЛЬСІВ ТАКОВОГО ГЕНЕРАТОРА ПРОХОДИТЬ ЗА ПЕРІОД СИГНАЛУ

- ГЕНЕРУВАННЯ ІМПУЛЬСІВ ЗАДАНОЇ ТРИВАЛОСТІ І ЧАСТОТИ
  - ЗМІНА СТАНУ “НІЖКИ” ПРИ НАРАХУВАННЯ ПЕВНОЇ КІЛЬКОСТІ ІМПУЛЬСІВ ТАКТОВОГО ГЕНЕРАТОРА
- КЕРУВАННЯ
  - КРОКОВИМИ ДВИГУНАМИ
  - ЦАП (ШИМ І СИГМА-ДЕЛЬТА)
  - СХЕМА СПІВПАДІНЬ
- ВИМІРЮВАННЯ
  - ЄМНОСТІ