







# ЛЕКЦИЯ 8

## СИСТЕМА ЗА ВХОД/ИЗХОД

-  **Периферни устройства**
-  **Свързване на периферията**
-  **Портове за вход/изход**
-  **Видове В/И подсистеми**
-  **Протоколи за вход/изход**
-  **Драйвери за вход/изход**

## ВИДОВЕ ПУ

**Предназначението** на ПУ е да осигурят на главните (**ЦП** и **ОП**) **връзка с околния свят.**

**Главната разлика** между ПУ и главните, които са изцяло електронни, е че **ПУ имат и механични компоненти**, т. е. **те са бавни.**

**Видът на връзката** определя **4 вида ПУ:**

- ① входни** (**клавиатура**, скенер, четец и др.);
- ② изходни** (**екран**, **печат**, манипулатор и др.);
- ③ запомнящи** (**входно-изходни + носител**);
- ④ комуникационни** (**МО**дулатор-**ДЕМ**одулатор).

# ВЪНШНА ПАМЕТ

**Носителят**, който използват запомнящите ПУ (**ЗУ**), се нарича **външна памет** (**ВП**).

**Защо** е необходима **ВП** (нали има **ОП**)?

- ❶ **ОП** е енергозависима  $\Rightarrow$  трябва да има и **енергонезависимо помнене** на данни.
- ❷ големите обеми данни изискват разумен **компромис между цена и бързина**.
- ❸ **в ОП не може** да се поставят **всички програми**, които ЦП някога ще изпълни (**някои ще бъдат създадени едва утре**).

## ВИДОВЕ ЗУ

Запомнящите устройства се различават по **четири** показателя:

- ➊ **технология** за помнене;
- ➋ **механика** за достъп до носителя;
- ➌ **материал**, от който се изработва носителят;
- ➍ **взаимна връзка** между устройство и носител.

# ТЕХНОЛОГИИ ЗА ПОМНЕНЕ

**Технологията определя** как ще бъде изработен **запомнящият елемент**.

## ❶ **Магнитна** технология:

- 😊 възможен е **презапис** на данните;
- 😞 проблеми с магнитния **материал**.

## ❷ **Оптическа** технология:

- 😊 **икономична** (**повече** елементи **в см<sup>2</sup>**);
- 😊 бързо тиражиране чрез **щамповане**;
- 😞 **еднократен запис** (**преодолява се**).

# ВИД НА НОСИТЕЛЯ

**Механиката** за достъп до носителя **определя и вида на носителя.**

- ① **Лентови устройства (1 движение):**
  - 😊 **проста механика** (съвместимост, цена);
  - 😞 **загуба** на носителя (от спирането);
  - 😞 **голяма разлика във времето за достъп.**
- ② **Дискови устройства (2 движения):**
  - 😊 **(почти) еднакво време за достъп;**
  - 😞 **сложна механика** (по-скъпи);
  - 😞 **енергоемки** (постоянно въртене).

# МАГНИТНИ ДИСКОВЕ

Магнитните дискове се изработват от различен материал, който се покрива с тънък слой от железен окис.

## 1 Твърди дискове (алуминий):

😊 повече повърхности (дисков пакет);

😞 възможна повреда при удар.

## 2 Гъвкави дискове (пластмаса):

😊 евтин и огъващ се материал;

😞 само две повърхности (един диск).

# МЯСТО НА НОСИТЕЛЯ

**Взаимодействието** на носителя с устройството може да бъде **различно**.

## ① Със сменяем носител:

- 😊 ЗУ е с неограничен капацитет;
- 😊 носителят може и да се **пренася**;
- 😞 **сложно калибриране** (обем и скорост).

## ② С несменяем (капсулован) носител:

- 😊 **еднократно калибриране** в завода;
- 😊 **защита** на носителя (висока плътност);
- 😞 **носителят ограничава обема** на ЗУ;
- 😞 **за пренос на данни се демонтира ЗУ**.



## ПОДСИСТЕМА ЗА В/И

**Базовата организация** на компютърните системи **не се е променила** съществено.

От трите главни **подсистеми**, тази **за В/И** е претърпяла **най-големи промени**, поради появата и използването на **нови ПУ**.

Въпреки увеличаването на ПУ, **техниката за свързването** им към компютрите **остава стандартна** и за програмист, който познава основните принципи, **не е проблем** да изучи характеристиките на новопоявило се ПУ и **да програмира «общуването» на ЦП с него**.

## СВЪРЗВАНЕ КЪМ ЦП

Подсистемата за **В/И** се свързва към ЦП чрез шината за **В/И**, съдържаща адресни, даннови и управляващи линии (жици).

**Логически** (но не винаги и **физически**) шините за **В/И** и шините на ОП са различни.

Бавното действие на **ПУ** налага те да се свързват към шината на ЦП чрез посредник, наречен **контролер** (**периферен адаптер**).

Той **управлява ПУ** и «превежда» **сигналите**.

В някои системи се използват допълнителни **процесори за В/И**, наречени **канални**.

# ОСОБЕНОСТИ

**Подсистемата за вход и изход съдържа както ПУ, така и техните контролери.**

**Специализираните процесори за вход/изход (каналите) също са част от тази подсистема.**

**Контролерът надзирава ПУ в съответствие с командите на ЦП и преобразува данните от вътрешно представяне във формата на ПУ.**

**Границата между ПУ и контролер е размита.**

**От появата на ИС с ВСИ нормална практика е контролерът да се разработва по смесена апаратно-програмна технология.**

## ПОРТОВЕ ЗА В/И

За да осъществи своята работа по управление на ПУ **контролерът** трябва да **съхранява (буфера)** сигнали.

За целта неговата **апаратна част съдържа** определен брой **регистри**, достъпни и на ЦП.

Те се наричат **портове** (за В/И) за да не става объркване с регистрите на ЦП.

**Броят на портовете** зависи от **спецификата** на управляваното ПУ, а **адресите** им – от **конструктора** на КС. **Най-общо** портовете са три: **управление, данни и състояние.**

# ВИДОВЕ ПОДСИСТЕМИ

Конструкторите на ЦП използват **две схеми за пренос на данни** между регистрите на ЦП и **портовете за В/И:**

- ❶ **изолиран В/И:** шината за В/И и шината към ОП са **физически различни** и за пренос на данни между регистър на ЦП и В/И порт са необходими **нови МИ in** и **out**.
- ❷ **В/И по аналогия с обръщението към паметта:** ОП и В/И използват **обща шина** и за достъп до В/И порт могат да се използват **всички МИ (дори ADD и SUB!)**.

## В/И ПО АНАЛОГИЯ

- ☺ За първи път се появява в **PDP-11**.
- ☺ Използва се в **M6800, M6809, M68000** и др.
- ☺ Изолиран В/И (**Интел**) чрез запояване може да се трансформира във В/И по аналогия.
- ☺ **Не са необходими нови МИ** за В/И.
- ☺ Използват се **всички МИ**, а не само преноси.
- ☺ **Голям брой адреси** за портове.
- ☺ Структурата на **шината е по-проста**.
- ☹ Част от **адресите на ОП се губят** за портове.
- ☹ Контролерите разпознават **по-дълги адреси**.
- ☹ **Специализираните МИ** могат да са **по-бързи**.
- ☹ **Обединението** на шините може да попречи.

## ПОРТИЛИ КЛЕТКА?

**Проектантът на КС определя какво има на даден адрес при В/И по аналогия.**

**Когато на даден адрес има клетка ОП:**

- ① можем да **четем** и да **записваме** данни;
- ② винаги се чете **последното** записано.

**Когато на даден адрес има В/И порт:**

- ① възможно е **записът** да е **забранен**;
- ② възможно е да **не** може да се **чете** (**И**);
- ③ При всяко четене **данните** може да са **различни** (**В**).

# ПРОТОКОЛИ ЗА В/И

Създаването на **подпрограми за** общуване с **периферията** е на порядък **по-сложно** от създаването на **обикновени програми**.

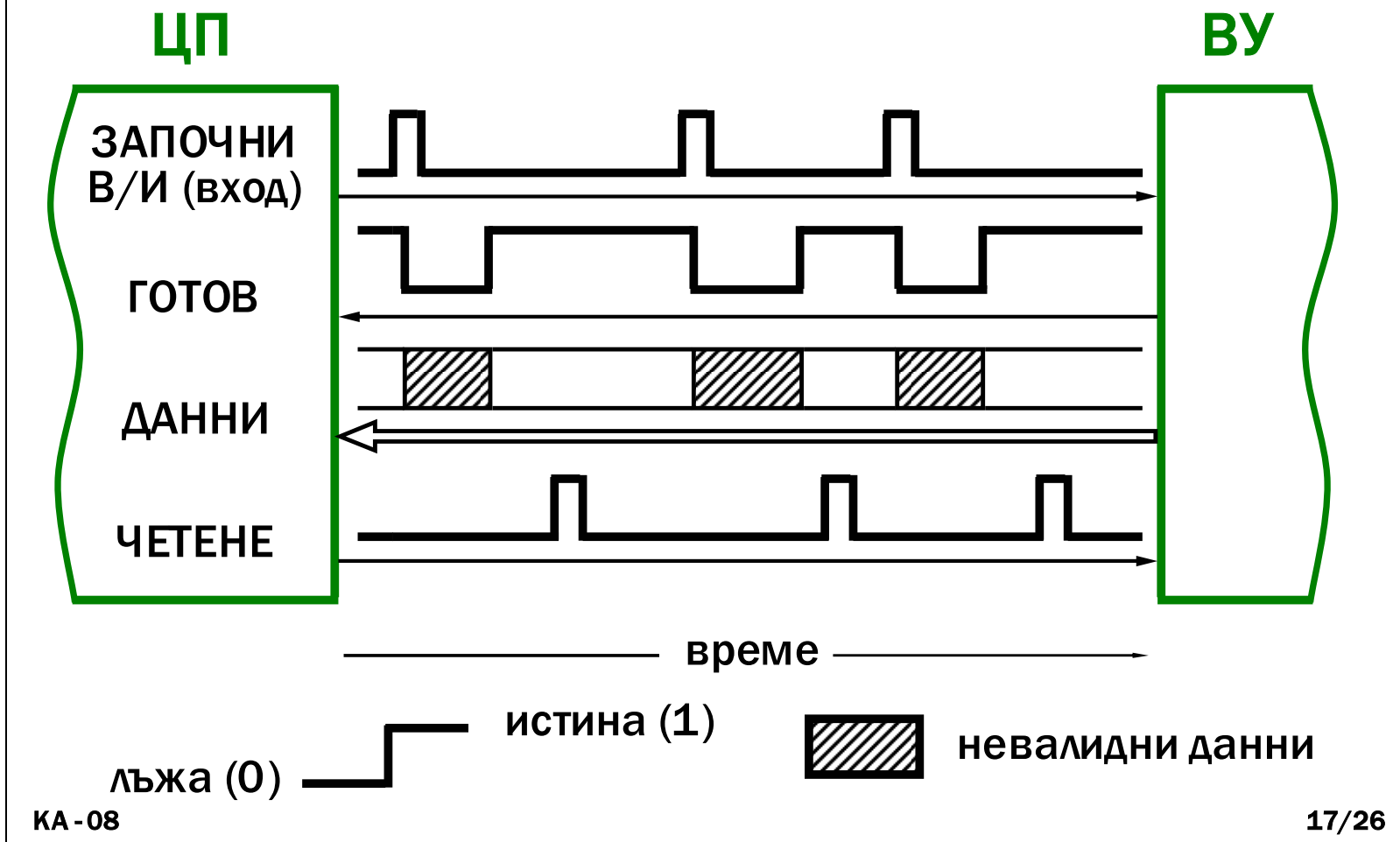
В общия случай устройствата за вход/изход използват **специфичен протокол за синхронизация**, наричан и **времедиаграма**.

**Спазването на времевите ограничения** е основната **причина за сложността** на подпрограмите за обмен с периферията.

**Протоколите са стандартизирани** като между **входния и изходния** има малка разлика.



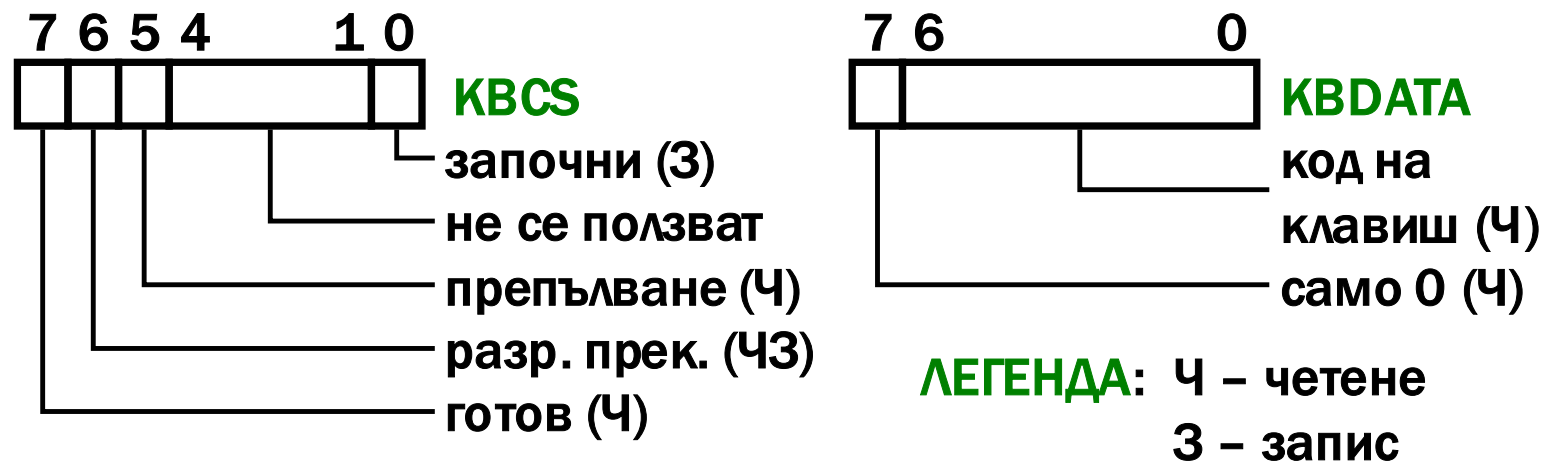
# ВХОДНА ОПЕРАЦИЯ



## ПРИМЕР: ВХОД

В този пример **портовете за управление и състояние са обединени**, т. е. те са на **един и същи адрес**, например **KBСS**.

Портът за **данни**, от който се чете въведен клавиш, е на **отделен адрес** – **KBDATA**.

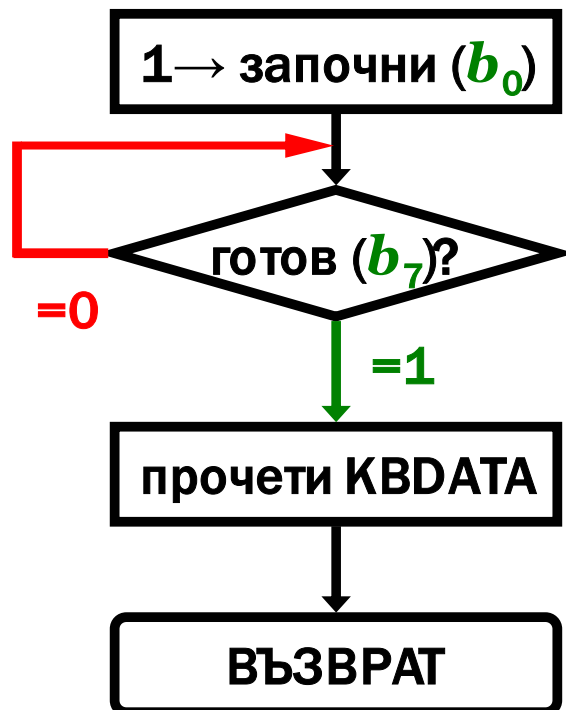


КА - 08

18/26

# ПРИМЕР: ППГ ЗА ВХОД

Въвеждането на данни може да бъде реализирано чрез следния алгоритъм:



```

KBDIN  LDA  KBCS
        ORA  #1
        STA  KBCS
  
```

```

KBDWT  TST  KBCS
        BPL  KBDWT
  
```

```

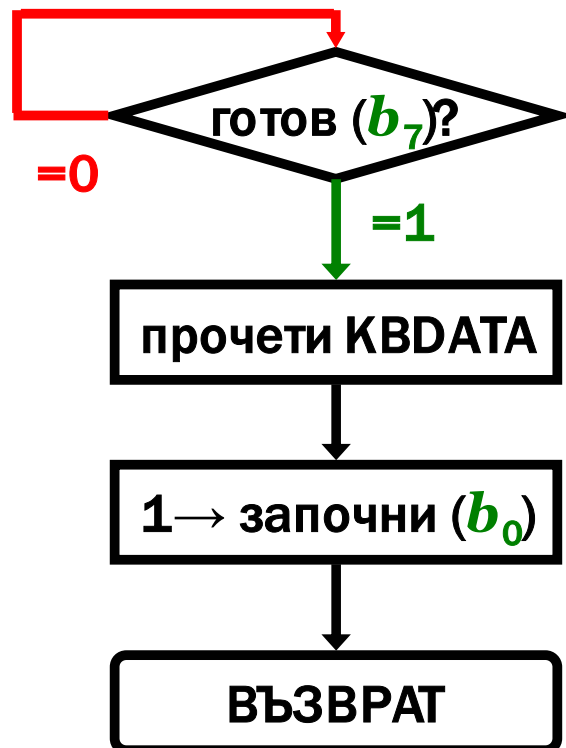
        LDA  KBDATA
        AND  #7FH
  
```

```

        RTS
  
```

# ПРИМЕР: ПАРАЛЕЛИЗЪМ

Ако незабавно стартираме операцията за ВХОД можем да постигнем паралелна работа:



```

KBDIN  TST  KBCS
        BPL  KBDIN

        LDA  KBDATA
        AND  #7FH

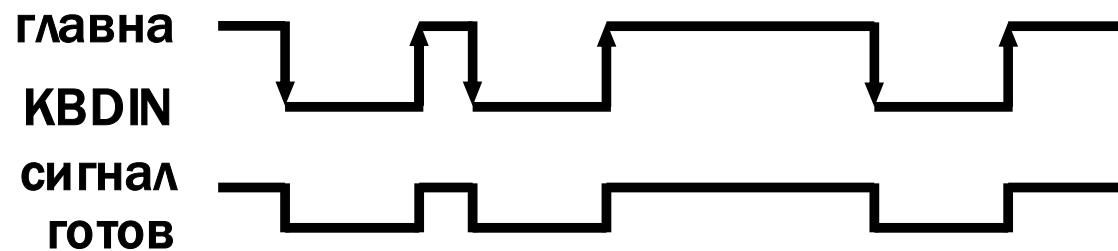
KBDIN  PUSHA
        LDA  KBCS
        ORA  #1
        STA  KBCS
        POPA

        RTS
  
```

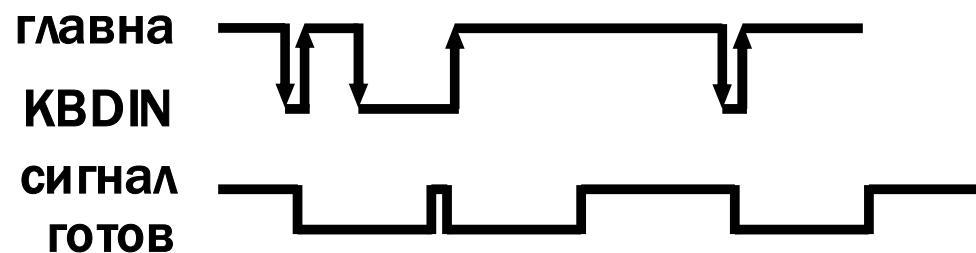
# СРАВНЕНИЕ

Ефектът от размяната се вижда при сравняване на времето на двата метода:

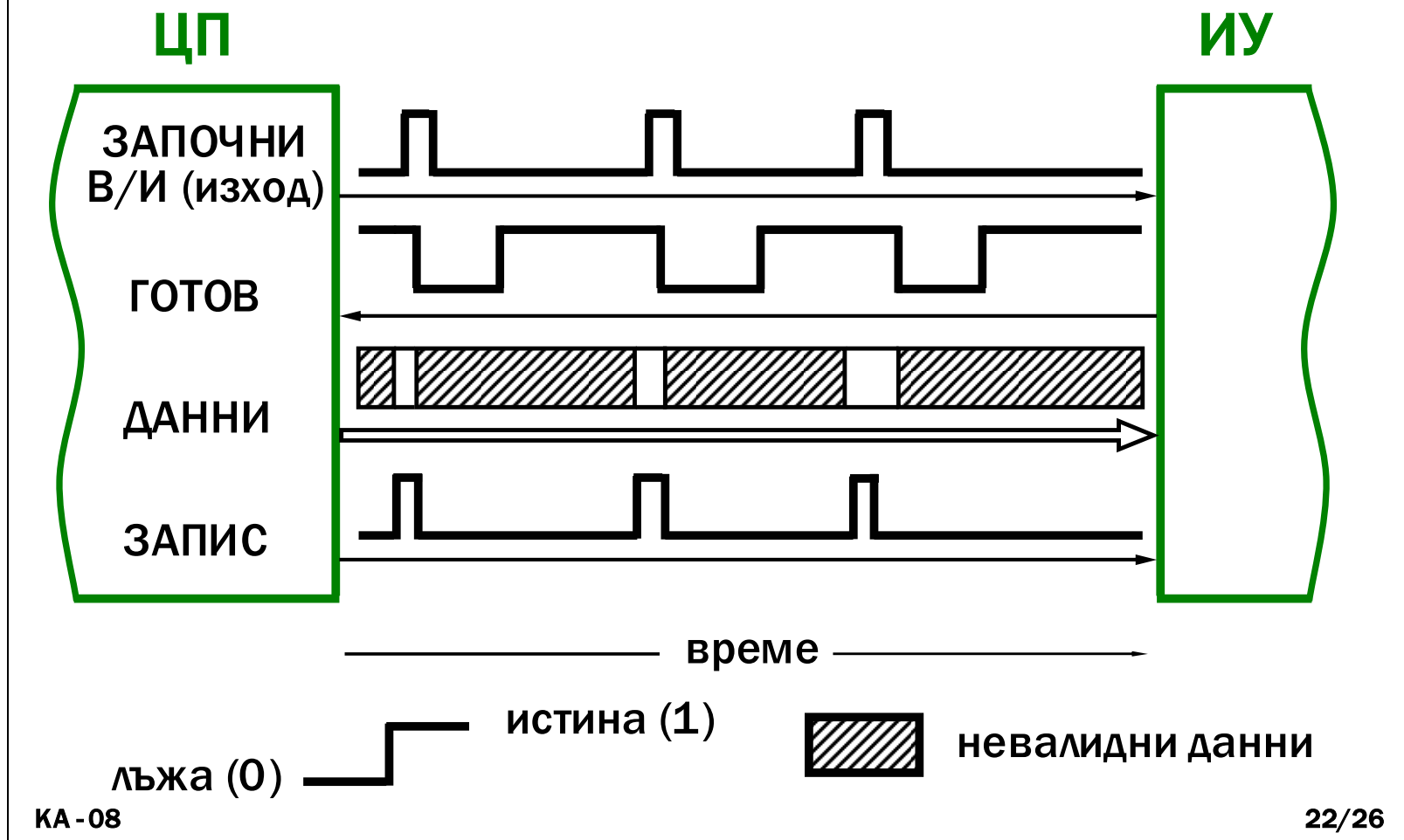
## ① несъвместен вход



## ② съвместен вход



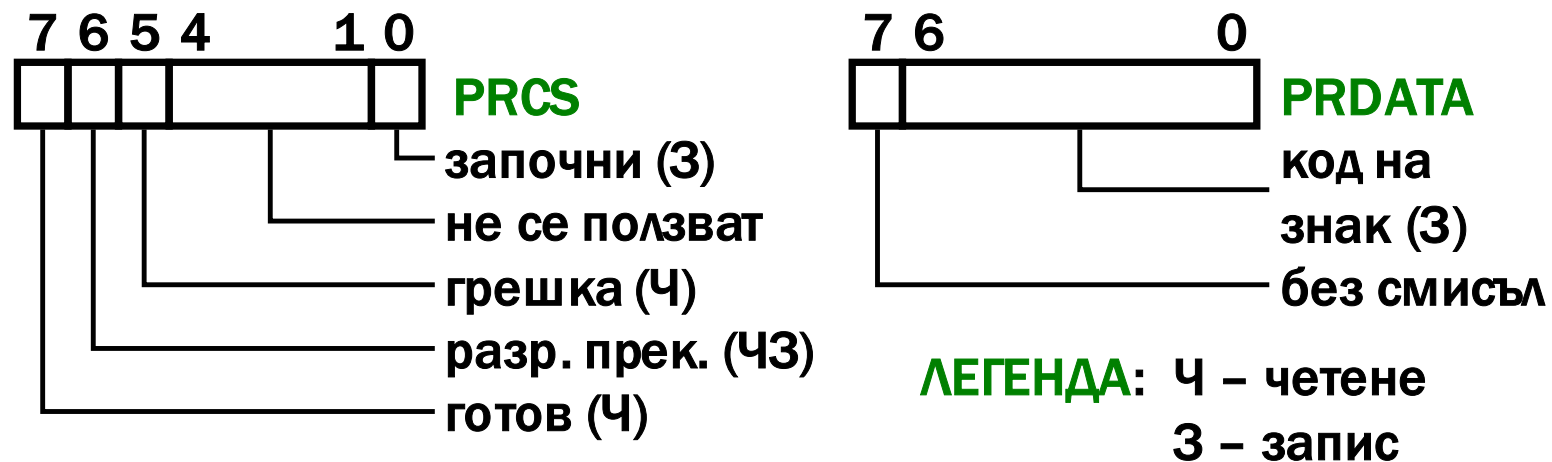
# ИЗХОДНА ОПЕРАЦИЯ



## ПРИМЕР: ИЗХОД

И в този пример **портовете за управление и състояние са обединени** и са на **един и същи адрес**, например **PRCS**.

Портът за **данни**, в който се записва знакът за печат, е на **отделен адрес** – **PRDATA**.

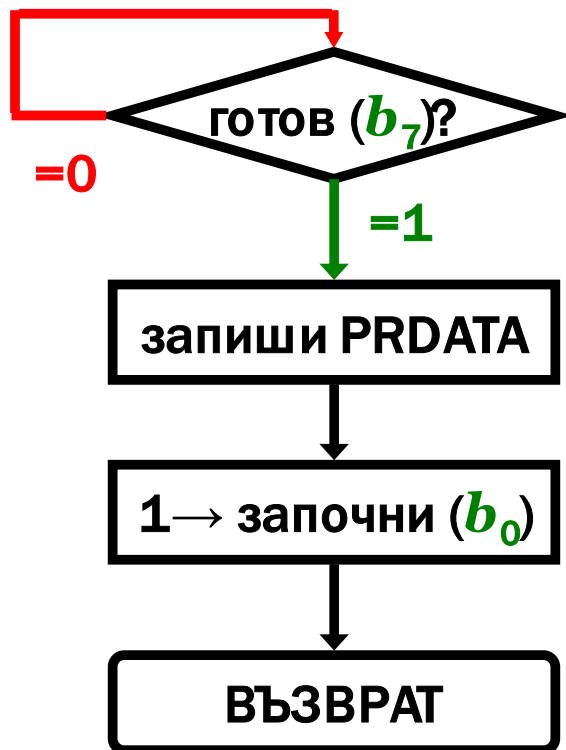


КА - 08

23/26

# ПРИМЕР: ППГ ЗА ИЗХОД

Изходните операции по естествен начин допускат паралелна работа:



```

PROUT  TST  PRCS
        BPL  PROUT

        STA  PRDATA

        LDA  PRCS
        ORA  #1
        STA  PRCS

        RTS
    
```



## ДРАЙВЕРИ ЗА В/И

Поради **времените изисквания** ППГ за общуване с периферията са на порядък **по-сложни** от изчислителните ППГ (напр. *sin*). Тези ППГ имат и едно голямо **преимущество**: веднъж съставени, те могат да се използват до изчезване на съответните ЦП и ПУ.

ППГ за В/И **укриват особеностите** на ПУ и са **прообразът на операционните системи**.

**Днес** е прието тези ППГ да **се наричат драйвери** за В/И. Обичайна практика е **драйверът за В/И** да получи като **параметър базовия адрес** на използваните **портове**.

**БЛАГОДАРЯ ВИ  
ЗА ВНИМАНИЕТО!**

**БЪДЕТЕ С МЕН И В  
СЛЕДВАЩАТА ЛЕКЦИЯ,  
КОЯТО ЩЕ НИ ОТВЕДЕ  
В НЕВЕРОЯТНИЯ СВЯТ НА  
СИСТЕМАТА  
ЗА ПРЕКЪСВАНЕ**