

ЛЕКЦИЯ 9

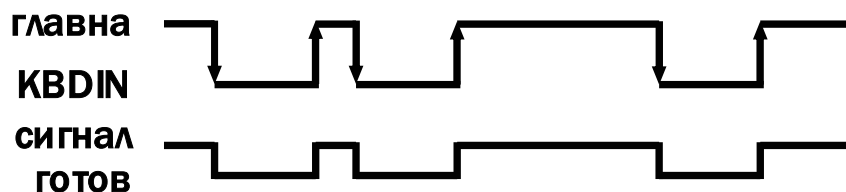
СИСТЕМА ЗА ПРЕКЪСВАНЕ

- ⌚ **Необходимост и същност**
- ⌚ **Поява и типове**
- ⌚ **Промяна на цикъла на УУ**
- ⌚ **Видове прекъсвания**
- ⌚ **Приоритети на прекъсванията**
- ⌚ **Примери**

КА - 09

1/20

НЕОБХОДИМОСТ



Когато в ОП има разположена втора главна програма, ЦП би могъл да изпълнява тази втора главна програма, вместо да изчаква пасивно появата на сигнал «ГОТОВ» в KBDIN.

Проблемът е как при изпълнение на тази втора програма ще стане ясно, че е дошло време за възобновяване на първата, т. е. че операцията по В/И е завършила с «ГОТОВ»?

КА - 09

2/20

СЪЩНОСТ

ЦП има един или няколко **електрически входа**, на които **В/И устройства** поставят активен **сигнал при възстановяване** на своята **готовност** (т. е. **«готов» от 0 става 1**). Така ПУ предават **заявка за обслужване**, която изисква от **ЦП временно да спре** изпълнението на **текущата програма** за да изпълни специален програмен **участък за обслужване**, чрез което да обърне внимание **на** нужното устройство за **В/И**. Впоследствие **прекъснатата програма продължава нормално** своята **работата**.

КА - 09

3/20

ПОЯВА И ТИПОВЕ

Асинхронен В/И чрез прекъсване се появява през **1958 г. в IBM – 709**.

Макар, че **изпълняваните програми не забелязват**, че изпълнението им е било спирано, **за някои това не е бива да става**. За целта **ЦП има специален управляващ бит**, наречен **маска (флаг) на прекъсванията**.

Заявки за прекъсване, които се влияят от тази маска, се наричат **маскируеми**.

Прекъсванията, които не се влияят от флага (неотложните), се наричат **немаскируеми**.

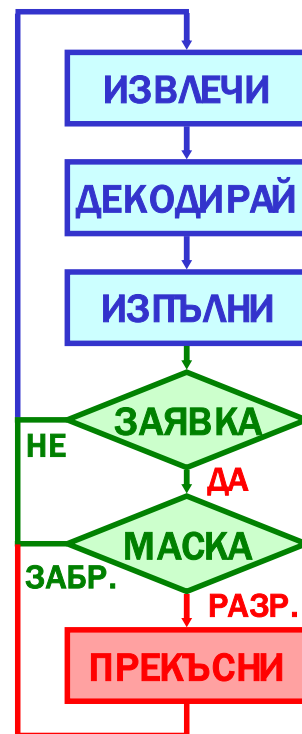
КА - 09

4/20

НОВ ЦИКЪЛ НА УУ

Не е рационално работата на ЦП да се прекъсва при изпълнение на МИ.

За да може ЦП да реагира на заявки за прекъсване стандартният цикъл на УУ **извечи–декодирай–изпълни** се допълва с нова фаза за проверка, включваща и реализация на прекъсването.



КА - 09

5/20

ДЕЙСТВИЕ НА ЦП

Прекъснатата програма трябва да може да продължи правилно след прекъсването.

Действията на ЦП при прекъсване са:

- ❶ Запазва жизнено важните параметри на програмата: като минимум **ПБ** и **РУ**.
- ❷ Забранява маскируемите прекъсвания.
- ❸ Установява нова стойност в **ПБ**.

Регистрите на ЦП също са важни за изпълнението на прекъснатата програма.

Тяхното запазване и възстановяване става от ППГ за обслужване на В/И устройство.

КА - 09

6/20

ВИДОВЕ ПРЕКЪСВАНИЯ

Създадени за облекчаване на операциите по В/И прекъсванията започват да **се използват и за други цели.**

Така днес са известни следните **ВИДОВЕ:**

- ① за вход/изход (възникват **първи**).
- ② програмни (при изпълнение на **МИ**).
- ③ външни (**друг ЦП**, таймер и др.).
- ④ за начално установяване (**reset**).
- ⑤ по контрол (**грешки**) на програмата.
- ⑥ по контрол (**грешки**) на апаратурата.

КА - 09

7/20

ПРИМЕРИ ЗА ГРЕШКИ

① На програмата:

- 💡 опит за **делене на 0**.
- 💡 **препълване** и машинна 0 при ПЗ.
- 💡 **недействителен КОП**.
- 💡 **нереализиран КОП**.
- 💡 **привилегирована МИ** в обикновен режим.
- 💡 опит за **нарушаване на защита на ОП**.
- 💡 **некоректен адрес** на МИ.

② На апаратурата:

- 💡 нарушение на **контрола по четност на ОП**.
- 💡 **неизправен ЦП** или друга част на КС.

КА - 09

8/20

ПРИОРИТЕТИ

Когато ЦП има **няколко** електрически **входа** за заявяване на **прекъсване** обичайна практика е **тези входове** да бъдат **наредени по приоритет (M6809)**.

Отделните входове могат да имат **отделни маски за забрана** на прекъсванията, **но** се практикува **и друга схема** с определяне на **приоритет на ЦП (PDP-11)**: заявка с по-висок приоритет от този на ЦП се обслужва, а заявка с по-нисък се отлага. Дори **при един вход ПУ** може да **се разделят по приоритет с допълнителна апаратура**.

КА - 09

9/20

ПРОГРАМЕН ПРИОРИТЕТ

Не е възможно две В/У устройства **едновременно да заявят прекъсване**.

При забранени прекъсвания заявките **не се обслужват** и **могат да се натрупат** няколко.

Такива **заявени, но все още не обслужени, прекъсвания** се наричат **«висящи»**.

Първата част на обслужването е **откриване** на В/И устройство **чрез проверка на сигнал «готов»**, когато ПУ има право да прекъсва.

Редът за проверка дава **две програмни приоритетни схеми: фиксирана и кръгова**.

КА - 09

10/20

ФИКСИРАН ПРИОРИТЕТ

При програмните схеми устройствата (чрез битовете в порт управление на контролера) се проверяват едно след друго, като първият отговорил печели (т. е. той има предимство).

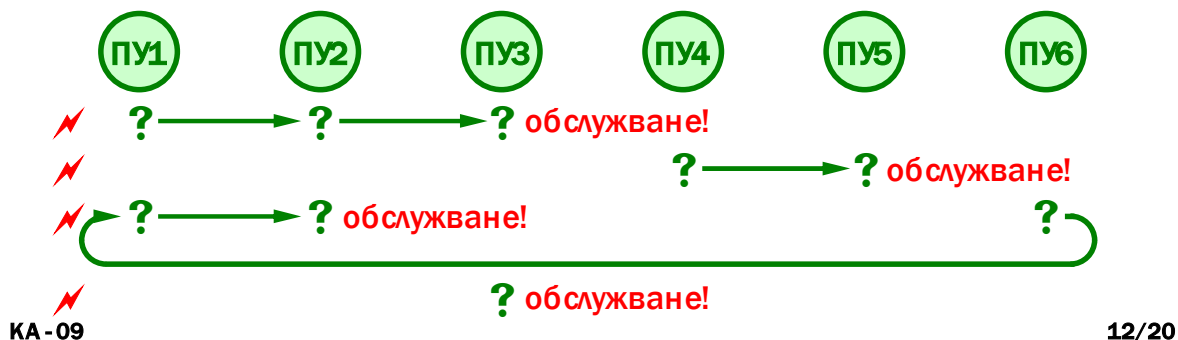
При фиксиран приоритет запитванията започват от едно и също устройство и редът на обхождане винаги е еднакъв.



КРЪГОВ ПРИОРИТЕТ

Както и при фиксирания приоритет **редът на обхождане е предварително определен, но в него първото у-во е и след последното.**

Главната **разлика** се състои в това, че всеки път започваме от устройството, което следва обслуженото при предишното обхождане.



ПРИМЕР 1: IBM 360

ЦП има 16 32-битови РОП и 24-битов адрес. Вместо ПБ има 64-битов регистър, наречен **Дума за Състояние на Програмата (ДСП) – Program Status Word (PSW)**:

байт	0		1		2		3	
	м-ки В/И		ОП	реж.	код на прекъсване			
	4		5		6		7	
	Д	У	прг.	програмен брояч				

Реж.: ASCII/EBCDIC, контрол ап., пауза/изч., суп./задача.

Прг.: маски на прекъсване по контрол на програмата.

КА - 09

13/20

IBM 360 (продължение)

Реакцията на ЦП при прекъсване е **запазване на ДСП** на определен адрес в ОП, наречен **стара ДСП**, и **зареждане на нова ДСП** от друг адрес на ОП, наречен **нова ДСП**.

ВИД ПРЕКЪСВАНЕ	СТАРА	НОВА
Външно	24	88
Програмно (SVC)	32	96
Контрол на програмата	40	104
Контрол на апаратурата	48	112
Вход/изход	56	120

КА - 09

14/20

ПРИМЕР 2: M6809

Този ЦП има **три входа** за заявяване на прекъсване: **немаскируеми (NMI)**, **бързи (FIRQ)** и **обичайни (IRQ)** прекъсвания.

Регистрите на ЦП са малко: четири 8-битови (2 акумулатора **A** и **B**, страничен **DPR** и **ПУ**) и пет 16-битови (**ПБ**, 2 индексни **X** и **Y** и 2 стекови **S** – системен и **U** – потребителски).

Регистърът на условията има вида:



КА - 09

15/20

M6809 (продължение)

При прекъсване M6809:

- ❶ **Запазва** в системния стек (**S**):
 - ⦿ **ПБ** и **ПУ** (къс стек) при бързо прекъсване (**FIRQ**);
 - ⦿ **всички регистри** на ЦП без **S** (пълн стек).
- ❷ **Забранява прекъсванията** (1 във **F** и **I**).
- ❸ **Записва в ПБ** прочетеното от адрес:

ВИД	АДРЕС	ПРОГРАМНИ	
		МИ	АДРЕС
НУ (reset)	FFFE, F	SWI	FFFA, B
НМП (NMI)	FFFC, D	SWI2	FFF4, 5
ОМП (IRQ)	FFF8, 9	SWI3	FFF2, 3
БП (FIRQ)	FFF6, 7		

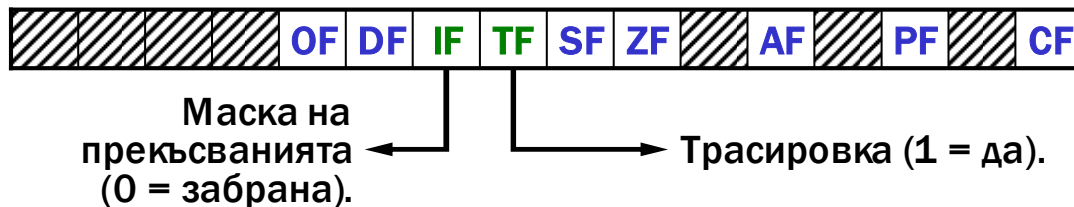
КА - 09

16/20

ПРИМЕР 3: I80x86

Този ЦП има 14 16-битови регистри: акумулатори **AX, BX, CX** и **DX**, базов **BP** (и **BX**), стеков **SP**, индексни **SI** и **DI**, сегментни **CS, DS, SS** и **ES**, указател на инструкцията (**IP**) и **PU** – флагове (**FLAGS**).

Блокът за преобразуване на адресите до 20 бита използва сегментните регистри. **Регистърът на условията (FLAGS) има вида:**



КА - 09

17/20

I80x86 (продължение)

Макар и странно, но при **I80x86** няма **ПБ**. **Неговата роля** се поема от **двойката CS:IP**.

Особена роля при прекъсване играят **първите 1024 байта от ОП**, наречени **вектор на прекъсванията**. Те съхраняват **256 двойки CS:IP**, които определят **адрес за обслужване на прекъсването**.

При прекъсване I80x86:

- ❶ Запазва в стека (**SS:SP**) **FLAGS, CS** и **IP**.
- ❷ Нулира **IF** и **TF** (= **забрана**).
- ❸ Изисква **8-битова идентификация** на **ПУ**, която **определя елемента с новите CS:IP**.

КА - 09

18/20

СПЕЦИАЛНИ ВЕКТОРИ

И80x86 има **МИ INT n** с непосредствен операнд, който определя номер на вектор. Специална **еднобайтова МИ INT3** осигурява възможност **за контролни точки** в програмата. **Възврат от прекъсване** осигурява **МИ IRET**.

Някои **специални вектори** са следните:

ВЕКТОР	АДРЕС	ПРЕКЪСВАНЕ
0	00000	Делене на 0
1	00004	Трасировка
2	00008	Немаскируемо прекъсване (NMI)
3	0000C	Контролна точка (INT3)
4	00010	Препълване при МИ INTO

КА-09

19/20

**БЛАГОДАРЯ ВИ
ЗА ВНИМАНИЕТО!**

**БЪДЕТЕ С МЕН И В
СЛЕДВАЩАТА ЛЕКЦИЯ,
КОЯТО ЩЕ НИ ОТВЕДЕ
В НЕВЕРОЯТНИЯ СВЯТ НА
ЕЗИКА АСЕМБЛЕР**