

BRNO UNIVERSITY OF TECHNOLOGY

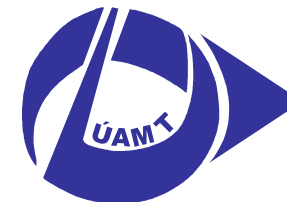
FACULTY OF ELECTRICAL ENGINEERING AND COMMUNICATION

DEPARTMENT OF CONTROL AND INSTRUMENTATION

Kolejní 4, 616 00 Brno

tel.: +420 5 4114 1113 fax: +420 5 4114 1123

E-mail: kucera@feec.vutbr.cz, <http://taceo.eu>



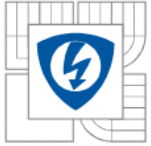
Operační Systémy Reálného Času

X.

Pavel Kučera

CAK, E112

kucera@feec.vutbr.cz

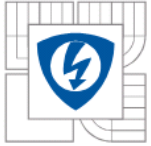


RTOS - X

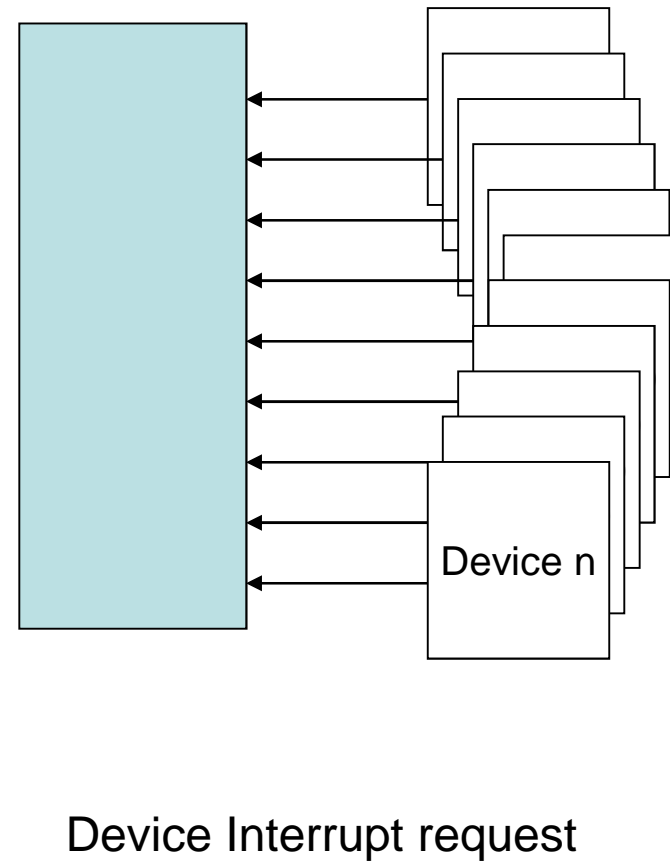
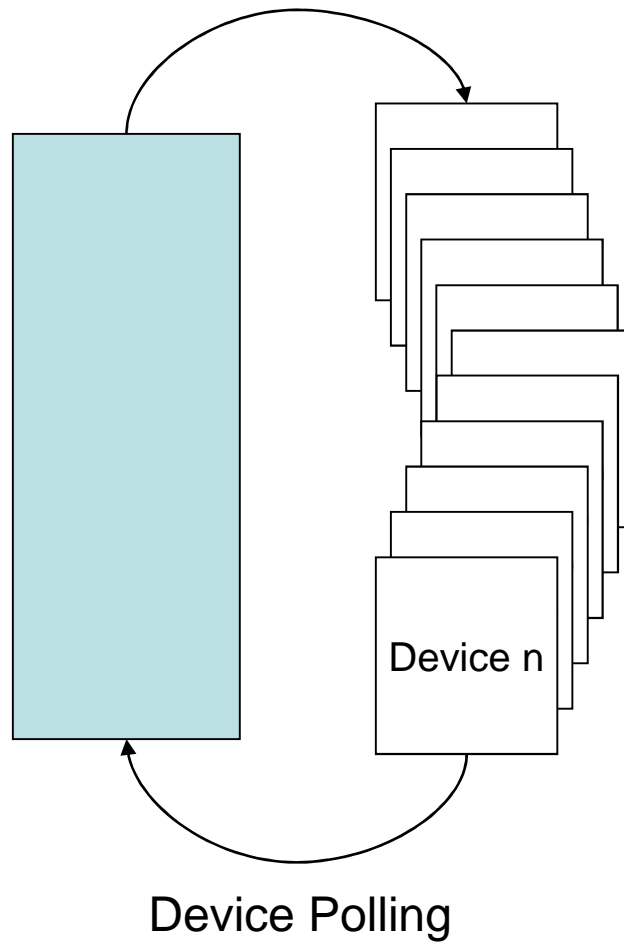


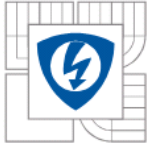
Obsah

1. Systém přerušení v PC
2. Systém přerušení v RTX



System přerušení v PC



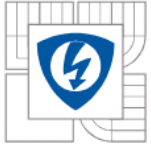


System přerušení v PC

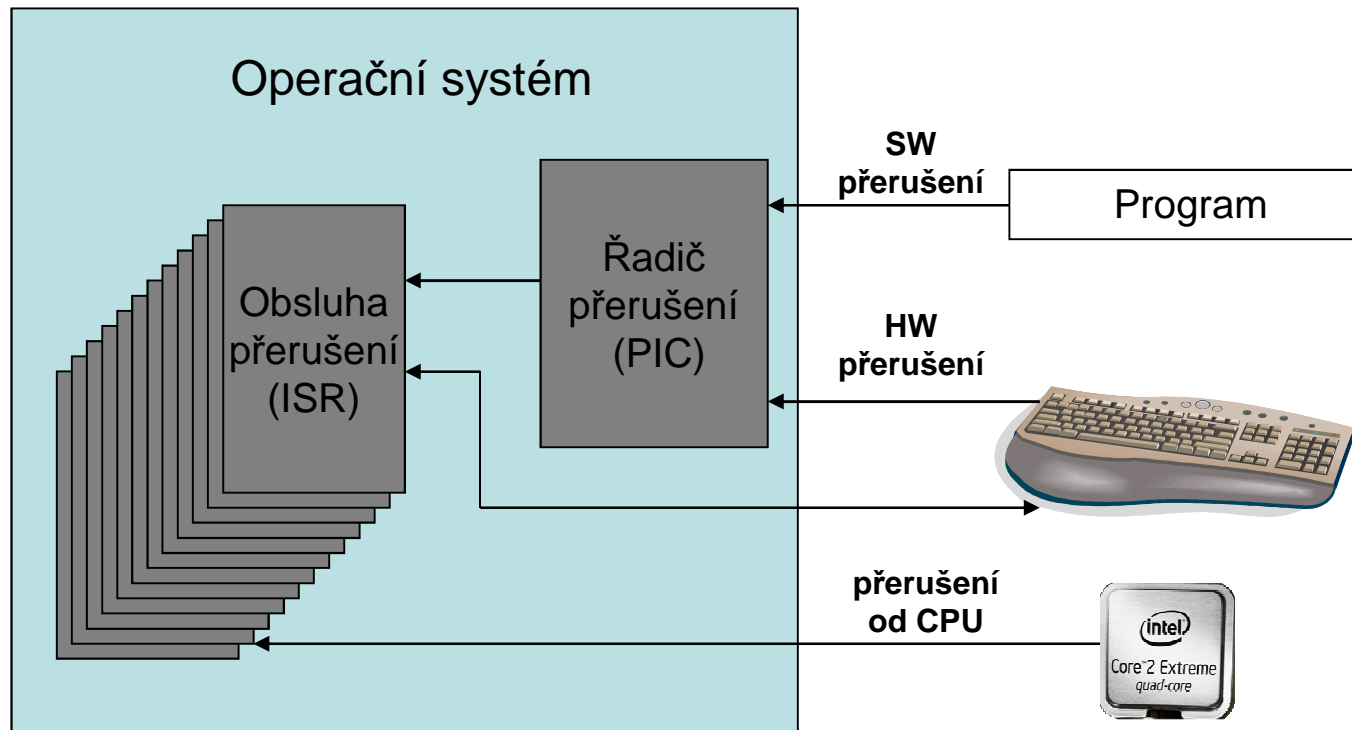
Device Polling technika, při níž CPU periodicky zjišťuje, zda-li dané zařízení vyžaduje obsluhu. To zpomaluje celý systém a na úrovni OS lze použít jen ve speciálních případech.

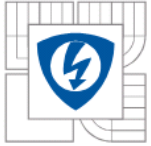
Device Interrupt request technika, při níž každé zařízení, které vyžaduje obsluhu posílá do CPU požadavek na přerušeni.

Uživatelské programy pak během svého chodu často kombinují obě techniky dohromady. Existují aplikace, kde obsluha pomocí pollingu zrychluje odezvu ISR.



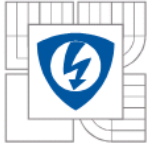
System přerušení v PC





System přerušení v PC

Hardware interrupt	přerušení, které je generováno konkrétním HW prvkem.
Software interrupt	přerušení, které je generováno instrukcí.
Maskable interrupt	přerušení, jehož požadavek lze ignorovat pomocí techniky masky přerušení.
NMI	(Non-Maskable Interrupt) – přerušení, které nemůže být ignorováno technikou maskování přerušení.
ISR	(Interrupt Service Routine) rutina obsluhy přerušení. Rutina je vykonána, když nastane příslušné povolené přerušení.
PIC	(Programmable Interrupt Controller) programovatelný řadič přerušení. Obvod, který je zodpovědný za generování signálu INT na procesoru, když nastane HW přerušení.
IDT	Interrupt Descriptor Table tabulka vektorů přerušení. Obsahuje seznam adres (instrukcí skoků) začátku obslužných rutin příslušných přerušení.



System přerušení v PC

ADS# Address Status - valid cycle and address

READY# Bus cycle complete

NA# Next Address - active even if the end of the current cycle is not being acknowledged on **READY#**

BS16# Bus Size 16 (D0-D15)

HOLD Other device requires bus mastership

HLDA Bus Hold Acknowledge

PEREQ Coprocessor Request

BUSY# Coprocessor Busy

ERROR# Coprocessor Error

INTR Maskable Interrupt Request (D0-D7 interrupt vector)

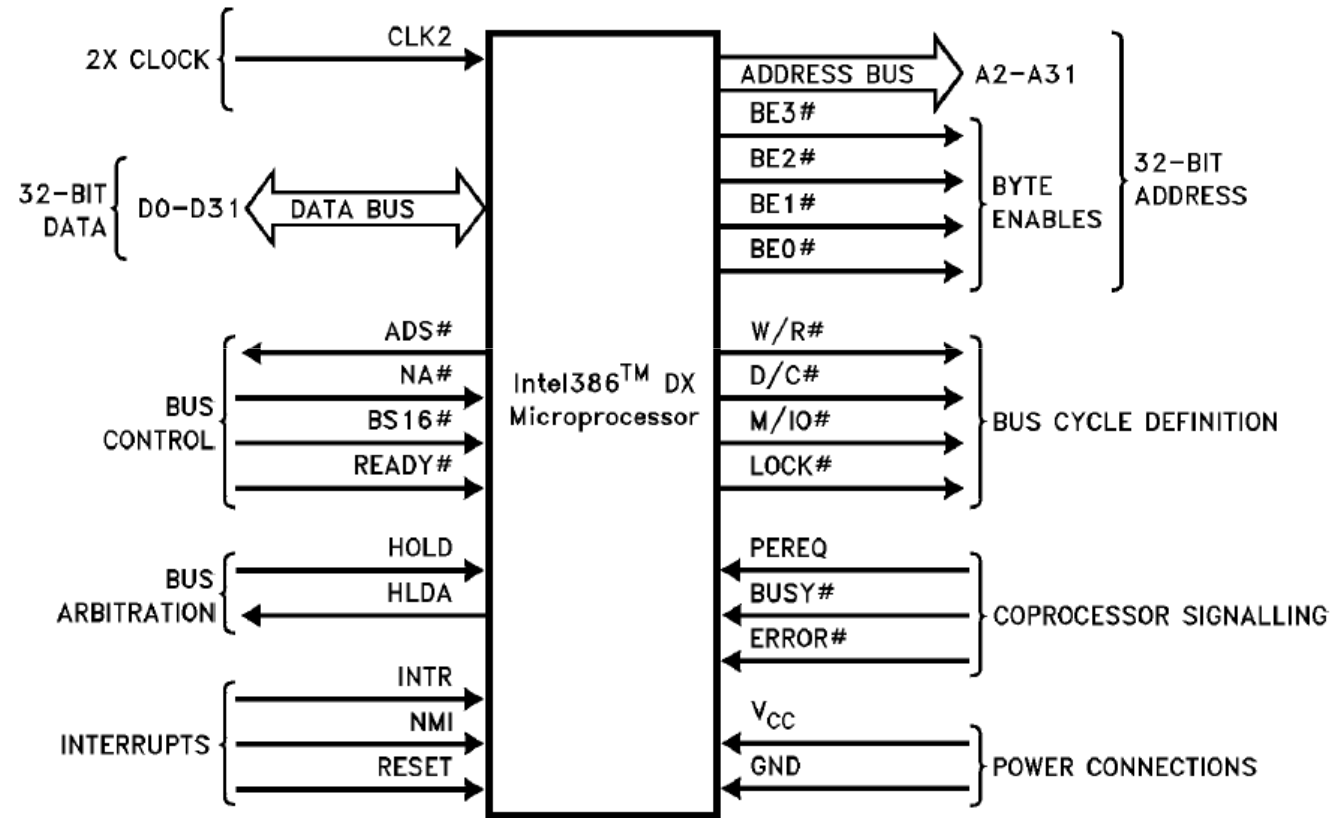
NMI Non-maskable Interrupt (slot 2 interrupt table)

BE0#-BE3#, A2-A31 Address Bus
4GB for memory, 64 KB for I/O

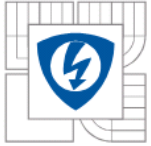
W/R# Write/Read cycle

D/C# Data/Control

M/IO# Memory/IO access



Zdroj: Intel 80386 reference manual

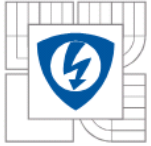


System přerušení v PC

NMI

Pokud je aktivní signál NMI na lince procesoru, pak CPU provede:

1. Procesor dokončí právě prováděný cyklus na sběrnici (instrukci).
2. Uloží obsahy registrů do paměti.
3. V tabulce vektorů přerušení vyzvedne na indexu 2 adresu obslužné rutiny.
4. Zavolá obslužnou rutinu (ISR).
5. Po skončení ISR obnoví obsahy registrů a vrátí se zpět do místa, kde byl přerušen.



System přerušení v PC

NMI

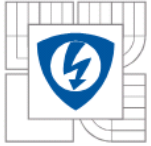
V IBM PC AT/XT je signál NMI použit zejména při chybě parity v paměti a ztrátě napájení. Signál je však maskován 7. bitem I/O portu na adrese 70h.

Při zapnutí počítače je signál NMI maskován kvůli POST fázi startu počítače.

Povolení NMI: Zápis na I/O port 70h 7. bit v logické nule

Zakázání NMI: Zápis na I/O port 70h 7. bit v logické jedničce

Na konci POST fáze systém povolí NMI !



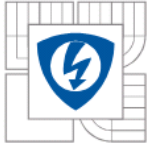
System přerušení v PC

RESET

Pokud je aktivní signál RESET na lince procesoru, pak CPU provede:

1. Procesor ukončí jakýkoliv výpočet a aktivitu na sběrnici.
2. Registr CS (Code Segment) a IP (Instruction pointer) nastaví na hodnotu 0xFFFFFFFF0.
3. Vymaže obsah všech ostatních registrů a příznakový registr.
4. Vymaskuje signál INTR.
5. Procesor vstoupí do módu reálných adres.
6. Vykoná instrukci na adrese 0xFFFFFFFF0.

Přerušení je namaskovatelné.



System přerušení v PC

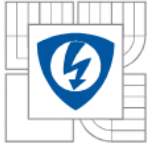
INTR

Pokud je aktivní signál INTR na lince procesoru, pak CPU provede:

1. Procesor dokončí právě prováděný cyklus na sběrnici (instrukci).
2. Otestuje příznak IF (Interrupt Enable Flag) a pokud je nulový, tak pokračuje další instrukcí – přerušení je maskováno.
3. Pokud je příznak IF roven jedné, pak procesor potvrdí akceptování požadavku o přerušení provedením dvou cyklů sběrnice INTA. INTA **není** signálem procesoru, lze ho odvodit z cyklu sběrnice kdy:

$$D/C\# = 0, W/R\# = 0, M/IO\# = 0$$

4. Při druhém cyklu INTA přečte procesor z datové sběrnice 8-bitové slovo a použije ho k získání adresy příslušné ISR.
5. Uloží aktuální obsahy registrů do paměti.
6. Zavolá obslužnou rutinu (ISR).
7. Po skončení ISR obnoví obsahy registrů a vrátí se zpět do místa, kde byl přerušen.

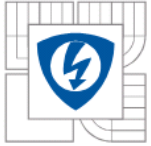


System přerušení v PC

INTn

Softwarové přerušení, které je generováno okamžitě po dokončení instrukce INT n. Poté provede následující aktivitu:

1. Procesor vymaže příznaky IF a TF.
2. Procesor uloží aktuální stav registrů do zásobníku.
3. Vyzvedne v tabulce vektorů přerušení adresu příslušné ISR.
4. Zavolá ISR.
5. Po skončení ISR obnoví obsahy registrů a vrátí se zpět do místa, kde byl přerušen.



System přerušení v PC

IDT

Tabulka vektorů přerušení.

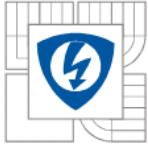
Obsahuje až 256 záznamů o umístění ISR v paměti.

Každý záznam se skládá ze 4 bajtů ve formátu:

OFFSET:SEGMENT

Procesor získá adresu přerušení v tabulce vynásobením kódu přerušení čtyřmi.

Tabulka leží v paměti v režimu reálných adres od adresy 0000:000h. V protected módu dle registru IDTR.



RTOS - X



IDT

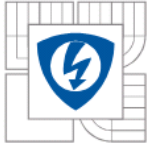
Interrupt	Address	Type	Function
00h	0000:0000h	Processor	Divide By Zero
01h	0000:0004h	Processor	Single Step
02h	0000:0008h	Processor	Nonmaskable Interrupt (NMI)
03h	0000:000Ch	Processor	Breakpoint Instruction
04h	0000:0010h	Processor	Overflow Instruction
05h	0000:0014h	BIOS/Software	Print Screen
05h	0000:0014h	Hardware	Bounds Exception (80286, 80386)
06h	0000:0018h	Hardware	Invalid Op Code (80286, 80386)
07h	0000:001Ch	Hardware	Math Coprocessor Not Present
08h	0000:0020h	Hardware	Double Exception Error (80286, 80386) (AT Only)
08h	0000:0020h	Hardware	System Timer - IRQ 0
09h	0000:0024h	Hardware	Keyboard - IRQ 1
09h	0000:0024h	Hardware	Math Coprocessor Segment Overrun (80286, 80386) (AT Only)
0Ah	0000:0028h	Hardware	IRQ 2 - Cascade from Second programmable Interrupt Controller
0Ah		Hardware	Invalid Task Segment State (80286, 80286) (AT Only)
0Ah		Hardware	IRQ 2 - General Adapter Use (PC Only)
0Bh	0000:002Ch	Hardware	IRQ 3 - Serial Communications (COM 2)
0Bh		Hardware	Segment Not Present (80286, 80386)
0Ch	0000:0030h	Hardware	IRQ 4 - Serial Communications (COM 1)
0Ch		Hardware	Stack Segment Overflow (80286, 80386)
0Dh	0000:0034h	Hardware	Parallel Printer (LPT 2) (AT Only)
0Dh		Hardware	IRQ 5 - Fixed Disk (XT Only)
0Dh		Software	General Protection Fault (80286, 80386)
0Eh	0000:0038h	Software	IRQ 6- Diskette Drive Controller
0Eh		Software	Page Fault (80386 Only)
0Fh	0000:003Ch	Software	IRQ 7 - Parallel printer (LPT 1)
10h	0000:0040h	Software	Video
10h		Software	Numeric Coprocessor Fault (80286, 80386)
11h	0000:0044h	Software	Equipment List
12h	0000:0048h	Software	Memory Size
13h	0000:004Ch	Software	Fixed Disk/ Diskette
14h	0000:0050h	Software	Serial Communication
15h	0000:0054h	Software	System Services
16h	0000:0058h	Software	Keyboard
17h	0000:005Ch	Software	Parallel Printer
18h	0000:0060h	Software	Load ROM Basic (PC Only)
18h		Software	Process Boot Failure (XT, AT)
19h	0000:0064h	Software	Boot Strap Loader
1Ah	0000:0068h	Software	Time of Day



System přerušení v PC

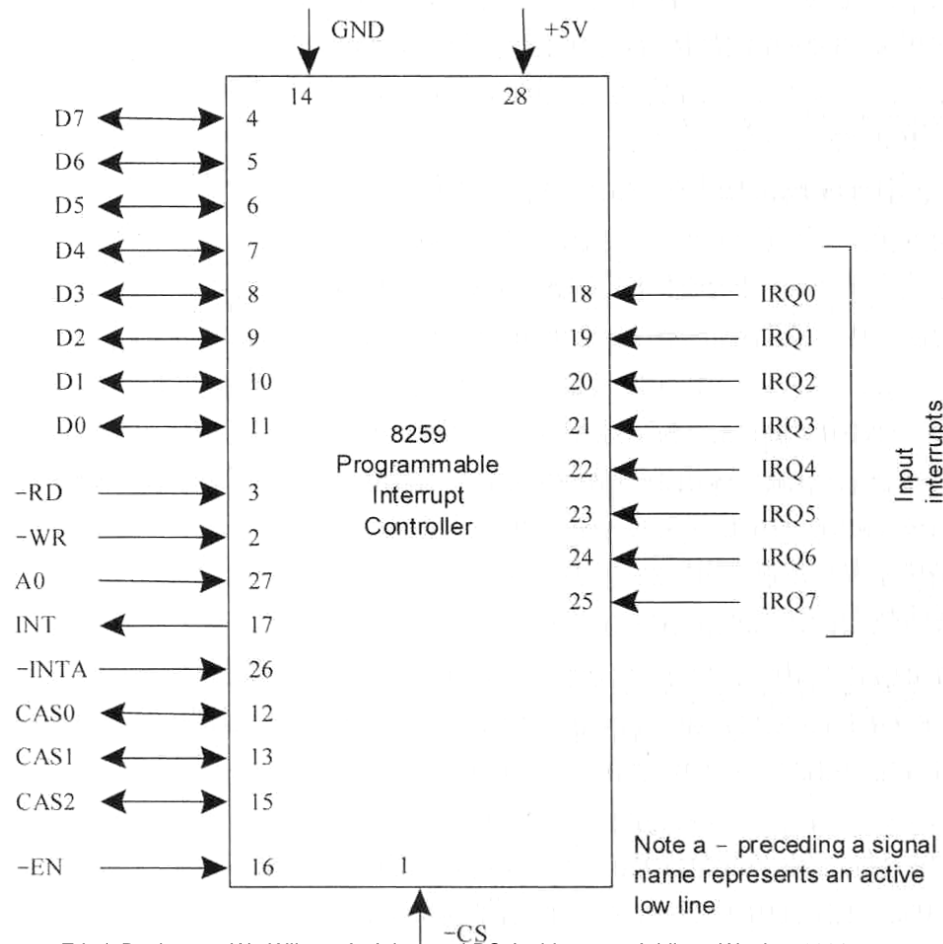
Tabulka IRQ

IRQ	Interrupt	Function
0	08h	System timer IC
1	09h	Keyboard controller IC
2	0Ah	Second IRQ controller IC
3	0Bh	Serial port 2 (COM 2: 2F8h-2FFh and COM 4: 2E8h-2EFh) or hardware modem use
4	0Ch	Serial port 1 (COM 1: 3F8h-3FFh and COM 3: 3E8h-3EFh) or serial port mouse use
5	0Dh	Parallel port 2 (LPT 2: 378h or 278h) or general adapter use (typically used for sound cards)
6	0Eh	Floppy disk controller
7	0Fh	Parallel port 1 (LPT 1: 3BCh [mono] or 378h [color])
8	70h	Real time clock or RTC
9	71h	Unused (redirected to IRQ 2)
10	72h	USB on systems so equipped (can be disabled) or general adapter use
11	73h	Windows sound system on systems so equipped (can be disabled) or general adapter use
12	74h	Motherboard mouse port on systems so equipped or general adapter use
13	75h	Math coprocessor
14	76h	Primary AT/IDE hard disk controller
15	77h	Secondary AT/IDE hard disk controller on systems so equipped (can be disabled) or general adapter use

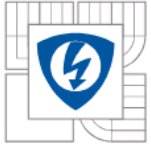


System přerušení v PC

Řadič přerušení 8259

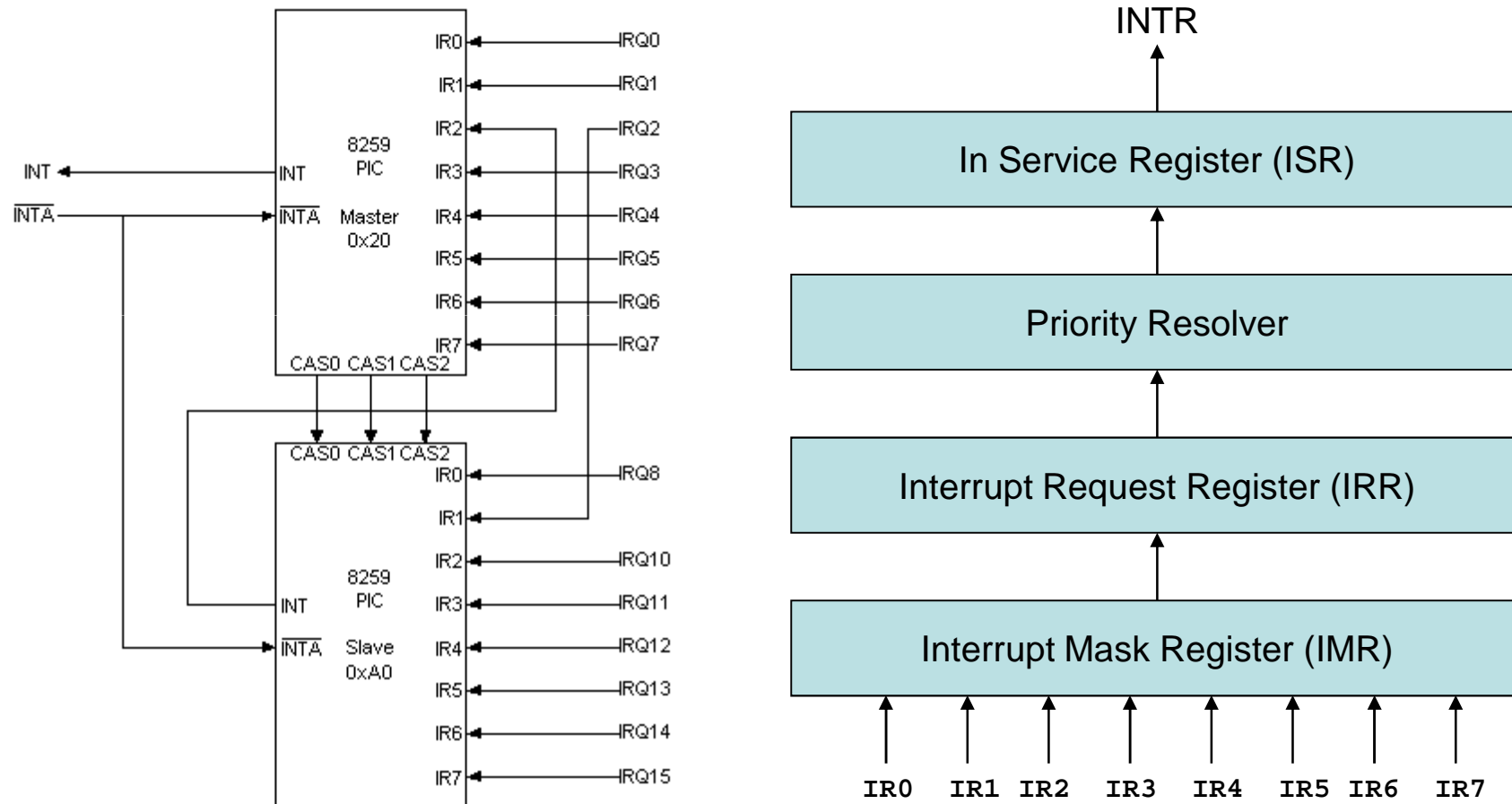


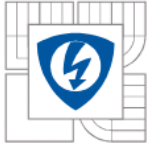
D0-D7	Bi-directional, tristated, buffered data lines. Connected to data bus directly or through buffers
RD-bar	Active low read control
WR-bar	Active low write control
A0	Address input line, used to select control register
CS-bar	Active low chip select
CAS0-2	Bi-directional, 3 bit cascade lines. In master mode, PIC places slave ID no. on these lines. In slave mode, the PIC reads slave ID no. from master on these lines. It may be regarded as slave-select.
SP-bar / EN-bar	Slave program / enable. In non-buffered mode, it is SP-bar input, used to distinguish master/slave PIC. In buffered mode, it is output line used to enable buffers
INT	Interrupt line, connected to INTR of microprocessor
INTA-bar	Interrupt ack, received active low from microprocessor
IRQ0-7	Asynchronous IRQ input lines, generated by peripherals.



System přerušení v PC

Řadič přerušení 8259





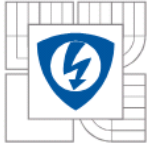
System přerušení v PC

Řadič přerušení 8259

Registry přístupné programově:

- ICP** interrupt control port, řídicí registr určující chování obvodu. Je přístupný na I/O portu 0x20.
- ICM** interrupt mask register, maskuje příslušné přerušení. Je přístupný na I/O portu 0x21.

<http://heim.ifi.uio.no/~stanisls/helppc/8259.html>

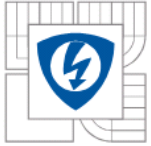


Přerušení v RTOS

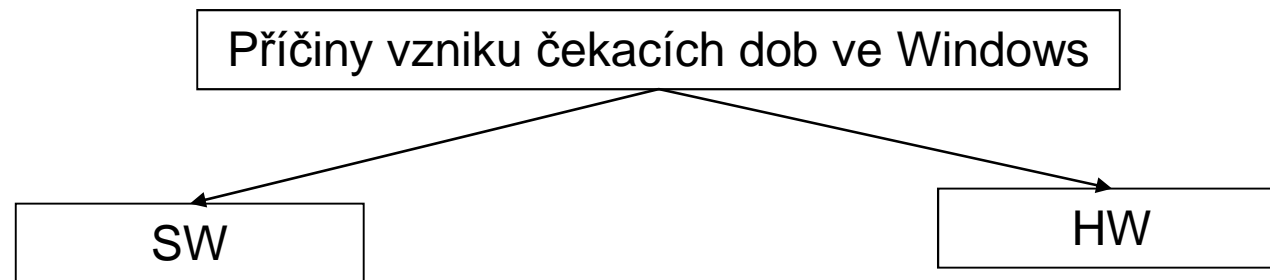
V RTOS je problém čekací doby při přerušení (Interrupt latency) velmi důležitý. Přerušování mohou na dané platformě vznikat v náhodných časech a v libovolném množství. Pokud by čekací doba pro obsluhu přerušování vzrostla nad určitou hodnotu, nebyl by RTOS schopen plnit svoje deadlines.

RTOS obvykle přenechávají problematiku řízení přerušování na uživatelských aplikacích nebo na ovladačích, které se linkují přímo k RTOS.

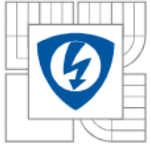
V podstatě neexistuje univerzální způsob jak ve velkém RTOS spolehlivě ošetřit problematiku přerušování, aby bylo zaručeno real-time chování a přitom všechny úlohy mohly přerušování sdílet/využívat.



Přerušení v RTOS



- STI/CLI instrukce volané z jádra OS nebo speciálních ovladačů blokují přerušení – zpoždění řádově v jednotkách μs .
 - IRQ je maskováno PIC (APIC) nebo jádrem OS, HAL a ovladači – zpoždění 100 ms.
 - Samotná obsluha přerušení zvětšuje čekací dobu ostatních přerušení.
- Převzetí sběrnice jiným HW (videokarta, DAQ) nebo DMA přenosy.
 - Vkládání wait cyklů při přenosu dat.
 - Vyplňování a vyprazdňování cache pamětí procesoru včetně znovu vyplnění TLB (translation lookaside buffer).
 - Power management – uspání CPU, snížení výkonu.
 - ...

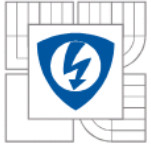


Přerušení v RTX

RT HAL rozšíření přeprogramovává PIC (APIC) tak, aby přerušení RTX vždy dokázalo přerušit Windows a RTX dokáže maskovat jakékoliv přerušení od Windows.

Windows a jeho ovladače nemohou zakázat přerušení RTSS.

Na úrovni přerušení CPU – signál NMI, RTX přebírá úlohu obsluhy tak, že používá spin locks nebo IRQ synchronizaci.



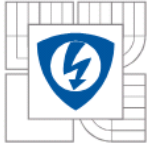
Přerušení v RTX

RTX umožňuje pomocí funkce **RtAttachInterruptVector** přiřadit konkrétnímu přerušení ISR (či spíše **IST** – Interrupt Service Thread).

Když přijde požadavek na přerušení, tak RTX provede tyto kroky:

1. Zdroj přerušení je vymaskován.
2. IST vlákno je spuštěno (resumed).
3. Pokud je priorita IST větší než aktuálně vykonávaného vlákna, tak se IST zaveden do CPU a poběží. **V opačném případě čeká v plánovači až na něho dojde řada!!!!**
4. Poté co obsluha skončí (handler obsluhy je vrácen) je zdroj přerušení demaskován a IST uspáno.

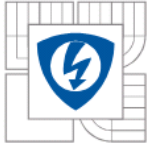
K demaskování zdroje přerušení a tím jeho povolení dojde vždy až po vykonání dané IST (obsluhy přerušení)!



Přerušení v RTX

RTSS umožňuje 12 úrovní priorit pro přerušení. Tato přerušení jsou mapována do standardního rozsahu priorit v RTSS takto:

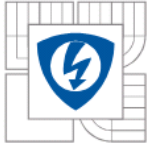
<i>RTSS Priority</i>	<i>Interrupt Priority</i>	<i>Remark</i>
0 - 117	0	Lowest Interrupt Priority
118	1	
119	2	
120	3	
121	4	
122	5	
123	6	
124	7	
125	8	
126	9	
127	10	Highest Interrupt Priority



Přerušení v RTX

Poznámky k přerušovacímu systému v RTX:

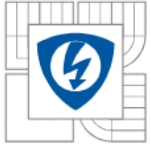
1. Protože obsluha přerušení je vlákno, může být obsluha v okamžiku jejího zavolání zastíněna vláknem s vyšší prioritou.
2. Mechanismus time-slicing může ovlivnit chování IST.
3. Mechanismus přerušení může být zablokován pro skupinu přerušení, která jsou přidružena jednomu procesu (jeho vláknům) tak, že se zvýší priorita právě běžícího vlákna. Tato úroveň musí být větší než priorita jakéhokoliv vlákna přiděleného funkcí **RtAttachInterruptVector** k obsluze přerušení.
4. Vlákna s vysokou prioritou **nikdy** nemaskují přerušení od časovačů; časovače jsou tedy vlákna běžící vždy s nejvyšší efektivní prioritou. K zamaskování přerušení od časovače je možné použít funkci **RtDisableInterrupts**.



Přerušení v RTX

Funkce **RtAttachInterruptVector** přiřadí danému vektoru přerušení obslužné vlákno – IST pro určitou sběrnici.

IRQ přerušení v architektuře IBM PC umožňuje připojení zařízení buďto přes sběrnici ISA nebo PCI. Zařízení jiného typu (SCSI, IDE, USB ...) jsou k CPU vždy připojena přes některé z těchto rozhraní.



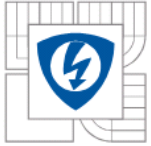
Přerušení v RTX

Funkce **RtAttachInterruptVectorEx** umožňuje mechanismus sdílení přerušení.

Vlákna v RTSS **mohou** sdílet vektory přerušení s jinými vlákny v RTSS (tak jako to mohou vlákna ve Win32).

Vlákna v RTSS a vlákna ve Win32 **nemohou** sdílet stejná přerušení! Je nutné se ujistit, jaké přerušení využívají Windows samotné a jaké ovladače, aby nedošlo ke konfliktu.

Mechanismus sdílení přerušení **zvětšuje** Interrupt latency, protože při vyvolání přerušení se nejdříve zavolá obsluha s vyšší prioritou, která posléze rozhodne které IST na stejném vektoru bude zavoláno. Proto je nutné zařízení, u nichž se požaduje real-time odezva, připojovat na samostatné vektory přerušení.



RTOS - X



Přerušení v RTX

http://taceo.eu/lectures/mrts/data/RTX_Interrupts_en.pdf