

Interrupt'ai pagal kilmę yra skirstomi į **vidinius** ir **išorinius**.

Vidiniai interrupt'ai iš pačio procesoriaus (arba jie išreikštai kviečiami programinio kodo arba neišreikštai, iškilus dėmesio reikalaujančiai situacijai). Jie turi **didžiausią** vykdymo prioritetą.

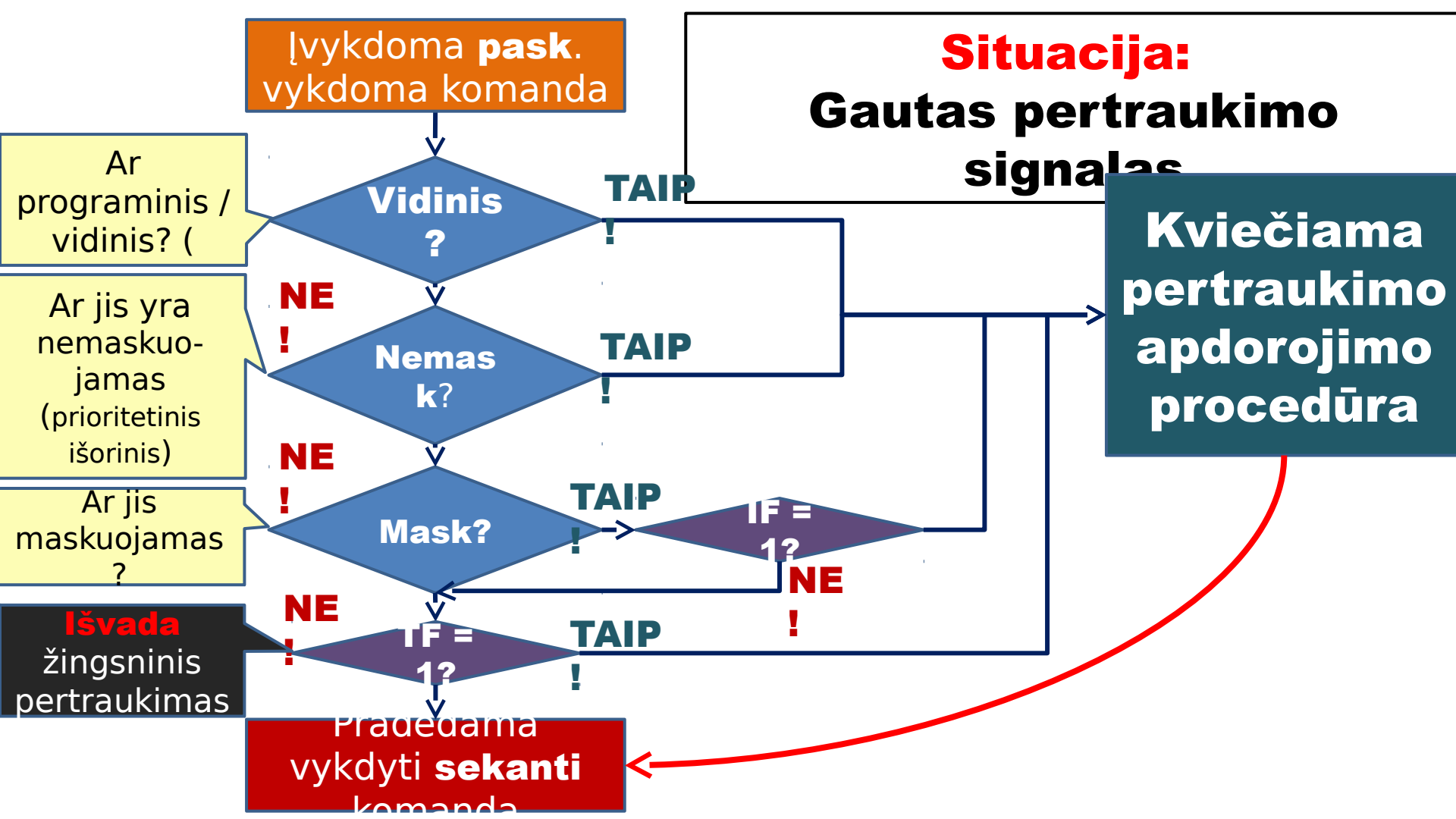
Išoriniai interrupt'ai kyla dėl išorinių priežasčių. Kadangi ne visada logiška yra reaguoti į absoliučiai visus išorinių įrenginių pertraukimus, **išoriniai** interrupt'ai yra papildomai skirstomi į **Maskuojamus** ir **Nemaskuojamus**.

Maskuojami pertraukimai gali būti ignoruojami, nustačius *Interrupt žymę (IF) = 0*.

Interrupt'ai skirstomi tam, kad, jei kiltų keli interrupt'ai vienu metu, būtų galima juos vykdyti tam tikra, logiška ir aprašyta, tvarka.

Vykdymo prioritetai:

1. Dalyba iš nulio
2. Programinis
3. INTO (overflow)
4. Nemask išoriniai
5. Mask išoriniai
6. Žingsninis (debug)



Pertraukimo procedūrų adresavimas

Kadangi į skirtingus pertraukimus reikia reaguoti skirtingai, skirtingų pertraukimų apdorojimo paprogramių (procedūrų) yra labai daug. Tam, kad būtų užtikrintas išplėčiamumas ir suderinamumas ateičiai, pagal susitarimą procesoriai laiko visų pertraukimų adresus pirmajame atminties kilobaite.

Kiekvienam pertraukimui (kurių gali būti 256) yra skiriami 4 baitai, vadinami **pertraukimo vektoriais**, kurie atitinka to konkretaus pertraukimo adresą atmintyje.

Pertraukimo vektoriaus keturi baitai atitinka pertraukimo procedūros **IP_J, IP_V, CS_J, CS_V**

Pertraukimo vektoriaus adresas apskaičiuojamas pagal formulę **$n * 4$** , kur **n** yra pertraukimo numeris.

Pertraukimo procedūrų adresavimo pvz.



Pertraukimo **INT 3h** vektoriaus **absoliutus adresas** yra **0000Ch**.
baitai adresu atitinka procedūros:

0000Ch: **IP_J**, 0000Dh: **IP_V**, 0000Eh: **CS_J**, 0000Fh: **CS_V**

Jei norime rasti pertraukimo procedūros absoliutųjį adresą, tereikia nueiti į pertraukimo vektorių, pasiimti keturis baitus ir iš jų „sulipdyti“ absoliutų adresą.

Jeigu atmintis būtų tokia, tai:

AA=00000: 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25
pertraukimo procedūros INT 3h: IP_J IP_V CS_J CS_V

CS = 2524h IP = 2322h AA = CS*10h + IP
pertraukimo procedūros **AA = 27562**

Sąvokų žodynėlis

- **Pertraukimo vektorius (vektoriaus adresas)** – adresas, nuo kurio keturiuose baituose aprašytas pertraukimo apdorojimo paprogramės adresas (lpj, lpv, Csj, Csv).
- **Pertraukimo absoliutus adresas** – pertraukimo apdorojimo paprogramės absoliutus adresas
- **Pertraukimo numeris** – pertraukimo numeris (skaičius rašomas po INT arba mašininiame kode po komandinio baito CD)
pvz.: CD 05 reiškia, kad bus įvykdytas **INT** 5h

Pertraukimo numeriai, kuriuos reikia mintinai mokėt:

- **INT** 0h = Dalybos iš nulio pertraukimas (CD 00)
- **INT** 1h = Žingsninio režimo pertraukimas (CD 01)
- **INT** 3h = Kontrolinio taško pertraukimas (CD 03) ARBA (CC)
- **INT** 4h = Perpildymo pertraukimas (CD 04) ARBA (CE)

Interrupto kvietimo mechanizmas

Pertraukimo kvietimas (INT n):

1. Pagal interrupto numerį (n) iš vektorių lentelės apskaičiuojamas vektoriaus absoliutus adresas
2. Į steką paeiliui padedamos SF, CS, IP registų reikšmės
3. TF ir IF Status flag'o žymės nunulinamos
4. Perduodamas valdymas

1. Adresas $n*4$:

IP_J, IP_V, CS_J, CS

2. PUSH SF

3. PUSH CS

4. PUSH IP

(SP – 6)

5. TF = 0, IF = 0

Pertraukimo paprogramėje vyksta apdorojimas

Grįžimas iš pertraukimo (IRET):

1. Iš steko paeiliui paima IP, CS, SF (atkreipkite dėmesį, eiliškumas atvirškčias dėl steko realizacijos)
2. Grąžinamas valdymas

1. POP IP

2. POP CS

3. POP SF

(SP + 6)

Pavyzdinė užduotis (1)



Duotos registrų reikšmės:

DS = FE21, SS = 5634, CS = C131, ES = 3EE3, SF = 05FF,
BP = 92A2, BX = C5D6, SI = 45FA, DI = 22F1, SP = FFE4

Duotas atminties fragmentas:

AA=00000: 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25

AA=00010: 26 27 28 29 30 31 32 33 34 35 36 37 38 39 40 41

AA=00020: 42 43 44 45 46 47 48 49 50 51 52 53 54 55 56 57

Užduotis: Apskaičiuoti pertraukimo INT 5h vektoriaus
absoliutųjį adresą.

Sprendimas (1)

- Gerai paskaitom sąlygą ir pažiūrime, ko mūsų prašo, mūsų prašo pertraukimo **VEKTORIAUS (!)** absoliutaus adreso.
- Vektoriaus absoliutus adresas skaičiuojamas $n \cdot 4$, kur n yra pertraukimo numeris.
- Kadangi pertraukimo numeris yra 5h, tai padauginam (naudodami sudėtį).
- Gautą atsakymą **14h** išplečiam iki 5 skaitmenų (kadangi absoliutaus adreso laukas susidaro iš 5 skaitmenų) ir gauname **00014h**, kas ir bus mūsų vektoriaus absoliutus adresas.

$$\begin{array}{r} + \quad 5h \\ + \quad 5h \\ \hline Ah \end{array} \quad \begin{array}{r} + \quad Ah \\ + \quad Ah \\ \hline 14h \end{array}$$

Ats: 00014h

Pavyzdinė užduotis (2)



Duotos registrų reikšmės:

DS = FE21, SS = 5634, CS = C131, ES = 3EE3, SF = 05FF,
BP = 92A2, BX = C5D6, SI = 45FA, DI = 22F1, SP = FFE4

Duotas atminties fragmentas:

AA=00000: 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25

AA=00010: 26 27 28 29 30 31 32 33 34 35 36 37 38 39 40 41

AA=00020: 42 43 44 45 46 47 48 49 50 51 52 53 54 55 56 57

Užduotis: Apskaičiuoti pertraukimo INT 5h apdorojimo
procedūros absoliutųjį adresą.

Sprendimas (2)

- Reikia apskaičiuoti pertraukimo apdorojimo procedūros absoliutųjį adresą. Mes žinome, kad jis yra laikomas pertraukimo vektoriuje, keturiuose baituose pavidalu:

IP_J , IP_V , CS_J , CS_V

- Kadangi INT 5h vektoriaus absoliutus adresas **00014h** (kaip gauti aprašyta praeitame sprendime), mums tereikia atmintyje tuo adresu surasti mus dominančius keturis baitus.

AA=00000: 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25

AA=00010: 26 27 28 29 30 31 32 33 34 35 36 37 38 39 40 41

IP_J IP_V CS_J CS_V

- Susilipdome **CS** ir **IP**: **CS** = 3332h, **IP** = 3130h ir pasirašom absoliutųjį adresą pagal formulę: **AA = CS*10h + IP.** **Ats:**
36450h

Pavyzdinė užduotis (3)



Duotos registrų reikšmės:

DS = FE21, SS = 5634, CS = C131, ES = 3EE3, SF = 05FF,
BP = 92A2, BX = C5D6, SI = 45FA, DI = 22F1, SP = FFE4

Duotas atminties fragmentas:

AA=00000: 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25

AA=00010: 26 27 28 29 30 31 32 33 34 35 36 37 38 39 40 41

AA=00020: 42 43 44 45 46 47 48 49 50 51 52 53 54 55 56 57

Užduotis: Kokia bus registrų SF ir SP reikšmių suma įvykdžius pertraukimo komandą INT 7h?

Sprendimas (3)

- Kadangi mums reikia apskaičiuoti SP ir SF reikšmių sumą įvykdžius pertraukimo komandą, mums reikia prisimint, kaip keičiasi jie keičiasi.
- Kviečiant pertraukimą į steką padedamos (PUSH) trys reikšmės. Tai reiškia, kad SP reikšmė mažės 6 (2 už kiekvieną PUSH).

Duotas **SP** = FFE4, $\text{FFE4} - 6 = \text{FFDE}$

naujas SP = FFDE

- Kadangi IF ir TF reikšmės yra nunulinamos, tai SF'o reikšmė keisis.

Duotas **SF** = 05FF, išsirašom dvejetainė: 0000 0101 1111 1111

XXXX 0DIT SZXA XPXC

Pakeičiam IF ir TF į 0 ir perrašom:

0000 0100 1111 1111

Paverčiam į šešioliktinę:

naujas SF = 04FF

- Sudedam **SF** ir **SP**: $\text{FFDE} + 04\text{FF} = 104\text{DD}$ (sumos atsakymui nėra taikomi lauko apribojimai, t.y. rašom tokį atsakymą, kokį gaunam) **Ats:**
104DDh

Užduotys

1. Atminties baituose su adresais nuo 00000 iki 000FF yra užrašytos reikšmės nuo 0 iki 255. Koks bus komandos INT 55h pertraukimo vektoriaus absoliutus adresas?
2. Apskaičiuokite kontrolinio taško pertraukimo vektoriaus absoliutų adresą.
3. Atminties baitai su adresais nuo 00000 iki 000FF užpildyti baitų reikšmėmis nuo -128 iki 127. Apskaičiuokite INT 39h pertraukimo apdorojimo procedūros absoliutų adresą
4. Atminties baitai su adresais nuo 00000 iki 000FF užpildyti baitų seka nuo -128 iki 127. Apskaičiuokite INT 15h pertraukimo procedūros IP reikšmę šešioliktaine sistema.

Užduočių atsakymai

- 1. 00154**
- 2. 0000C**
- 3. 6DBC4**
- 4. D5D4**