## **Motorola 6800 ir Intel 4040 architektūrų palyginimas**

Parengė Ieva Lileikytė, 1 grupė, 2 pogrupis

1. **Kokia elementinė kompiuterio/procesoriaus bazė? Kokios fizinės įrangos savybės (svoris, dydis, energijos suvartojimas)?**

Motorola 6800 ir Intel 4040 yra mikroprocesoriai. Abu išleisti 1974 metais.

Motorola 6800 mikroprocesorius pagamintas naudojant “DIP 40-pin” (“Dual In-Line Package”) 40 kontaktų korpusą. Tokio korpuso dydis yra maždaug: 14mm 52 mm 5mm. Turi 4.100 tranzistorių.

Svoris: apie 23 gramus. Energijos suvartojimas: apie 1.3W.

Intel 4040 mikroprocesorius pagamintas naudojant “DIP 24-pin” korpusą. Tokio korpuso matmenys yra maždaug: 18mm 31mm 5mm. Turi 3.000 tranzistorių.

Svoris: apie 15 gramų. Energijos suvartojimas: mažesnis nei 0.5W.

1. **Kokio tipo architektūra turėjo abu kompiuteriai?**

Ir Motorola 6800 procesorius ir Intel 4040 remiasi registrine architektūra.

“The architecture of the 4004 was a fairly straightforward register-based architecture” (Intel4004.com, 2024).

Intel 4040 yra patobulinta versija Intel 4004.

1. **Ar tai buvo beadresinės (stekinės), vieno adreso, dviejų adresų, trijų adresų ar keturių adresų mašinos?**

Motorola 6800 yra dviejų adresų mašina, o Intel 4040 yra vieno adreso mašina.

1. **Kokie registrai šiose architektūrose?**

**Registrai Motorola 6800:**

Akumuliatorius A (ACCA): 8 bitų registras, naudojamas aritmetinėms ir loginėms operacijoms.

Akumuliatorius B (ACCB): 8 bitų registras, naudojamas aritmetinėms ir loginėms operacijoms.

Indekso registras (IX): 16 bitų registras, dažniausiai naudojamas laikiniems duomenims saugoti arba kaip indeksas, kai naudojamas indeksuotas adresavimas.

Programos skaitiklis (PC): 16 bitų registras atsakingas už dabartinio vykdomo kodo adresavimą.

Steko rodyklė (SP): 16 bitų registras. Naudojamas funkcijų iškvietimams ir grįžimams stebėti.

Vėliavų registras (Condition Code Register arba CCR):

* Tarpinių bitų pernešimai (H): nustatoma, jei skaičiuojant rezultatą buvo perkelta iš 3 į 4 bitą.
* Pertraukimai (I): nustatoma, jei IRQ pertraukimas išjungtas.
* Neigiamas (N): nustatomas, jei nustatytas reikšmingiausias rezultato bitas.
* Nulis (Z): nustatomas, jei rezultatas lygus nuliui.
* Perpildymas (V): nustatomas, jei paskutinio rezultato skaičiavimo metu buvo perpildymas.
* Carry (C): nustatomas, jei paskutinio rezultato skaičiavimo metu buvo perkėlimas iš 7 bito.

Iš viso 6 pagrindiniai registrai.

**Registrai Intel 4040:**

Programos skaitiklis (PC): 12 bitų.

Steko registrai: “There are seven stack level registers, which is sufficient to implement 7-level deep subroutine calls. Every subroutine call stores return address in one of the stack registers. Contents of the accumulator and the index registers is not preserved. When the program is interrupted, the contents of the send register (SRC) is also preserved in the stack.” (CPU-World, 2024)

Akumuliatorius (ACC): 4 bitų, aritmetinėms operacijoms, loginėms operacijoms, taip pat naudojamas keistis duomenimis su RAM ir I/O.

Indekso registrai: “The microprocessor has 24 4-bit registers, organized as 16 registers in bank 0 and 8 registers in bank 1. Lower 8 registers can be selected from bank 0 or 1 using SB0 and SB1 instructions. The registers can work in pairs as 8 8-bit registers.” (CPU-World, 2024)

Iš viso 33 registrai.

1. **Ar požymių bitai buvo naudojami šiose architektūrose? Kokie požymiai buvo naudojami?**

Abi architektūros naudojo požymių bitus.

**Motorola 6800**

Vėliavų registras (Condition Code Register arba CCR):

* Tarpinių bitų pernešimai (H): nustatoma, jei skaičiuojant rezultatą buvo perkelta iš 3 į 4 bitą.
* Pertraukimai (I): nustatoma, jei IRQ pertraukimas išjungtas.
* Neigiamas (N): nustatomas, jei nustatytas reikšmingiausias rezultato bitas.
* Nulis (Z): nustatomas, jei rezultatas lygus nuliui.
* Perpildymas (V): nustatomas, jei paskutinio rezultato skaičiavimo metu buvo perpildymas.
* Carry (C): nustatomas, jei paskutinio rezultato skaičiavimo metu buvo perkėlimas iš 7 bito.

**Intel 4040**

Intel 4040 neturi specialaus vėliavų registro kaip Motorola 6800, tačiau naudoja požymių bitus. Naudoja “Carry flag” (C).

1. **Koks buvo kiekvienos architektūros duomenų plotis?**

Motorola 6800 duomenų plotis 8 bitai.

Intel 4040 duomenų plotis 4 bitai.

1. **Koks buvo kiekvienos sistemos atminties išdėstymas?**

**Motorola 6800**

“Program, data and stack memories occupy the same memory space. The total addressable memory size is 64 KB.

**Program memory** - program can be located anywhere in memory. Jump and subroutine call instructions can be used to jump anywhere in memory. Conditional and unconditional branches are limited to memory addresses positioned no farther than -125 - +129 bytes from the branch instruction.

**Data memory** - data can be anywhere in memory space.

**Stack memory** - stack can be placed anywhere in memory space.

**Reserved memory locations**:

* FFF8h - FFF9h: Pointer to IRQ interrupt-processing routine.
* FFFAh - FFFBh: Pointer to software interrupt-processing routine.
* FFFCh - FFFDh: Pointer to NMI interrupt-processing routine.
* FFFEh - FFFFh: Pointer to RESET handling code.
* Some memory addresses may be reserved for memory mapped I/O as the processor doesn't have hardware I/O capability.” (CPU-World, 2024)

**Intel 4040**

**“Program memory** size is 8 KB organized as two 4 KB memory banks. At any point of time the processor works with one 4 KB bank, when necessary the processor can switch to different bank using new DB0 and DB1 instructions. Like in Intel 4004, all conditional instructions in the 4040 work within currently selected ROM (256 bytes). Unconditional jump and jump to subroutine instructions can be used to jump to any address within currently selected bank.

**Data memory** size is 640 bytes. RAM access is done in the same way as access to I/O ports. First, a SRC instruction is used to tell the processor what memory address to access, and successive WRM or RDM writes accumulator data to memory or reads data into accumulator. Data memory is separate from program memory.

**Stack** is 7-level deep. Stack is separate from program memory and data memory.” (CPU-World, 2024)

**Ar adresų erdvė ištisinė ar ji buvo suskirstyta?**

Motorola 6800 atmintis yra ištisinė, nes “Program, data and stack memories occupy the same memory space.” (CPU-World, 2024)

Intel 4040 adresų erdvė yra segmentuota, nes “Program memory size is 8 KB organized as two 4 KB memory banks.” (CPU-World, 2024).

**Koks buvo (efektyvus) adreso plotis?**

Motorola 6800 efektyvus adreso plotis buvo 16 bitų, o intel 4040 12 bitų.

**Koks buvo maksimalus įmanomas atminties kiekis kiekvienoje sistemoje?**

Motorola 6800: “The total addressable memory size is 64 KB.” (CPU-World, 2024).

Intel 4040: **“**Program memory size is 8 KB organized as two 4 KB memory banks. <…> Data memory size is 640 bytes.” (CPU-World, 2024).

1. **Ar buvo palaikoma virtualioji atmintis?**

Nei Motorola 6800 nei Intel 4040 nepalaiko virtualios atminties. Abi turėjo fizinę atmintį.

1. **Kokia buvo kiekvienos sistemos architektūros komandų sistema (ISA)?**

Motorola 6800 architektūros komandų sistema buvo CISC, taip pat ir Intel 4040.

**Kiek mašinos komandų turėjo kiekviena sistema?**

Motorola 6800 turėjo 72 instrukcijas, o Intel 4040 - 60.

**Kokios buvo instrukcijų (komandų) klasės?**

Motorola 6800:

* Duomenų perkėlimo instrukcijos.
* Aritmetinės: sudėtis, atmintis, neigimas, inkrementuojančios, dekrementuojančios, lyginimo operacijos.
* Loginės: AND, OR, XOR, komplementavimas, poslinkio/rotacijos instrukcijos.
* Kontrolės instrukcijos: sąlyginiai ir nesąlyginiai šuoliai, “subrout” iškvietimai ir grįžimai.
* Kitos: sąlygų vėliavėlių išvalymas/nustatymas, bitų testai, programinės įrangos nutraukimas ir kt.

Intel 4040:

* Duomenų perkėlimo instrukcijos.
* Aritmetinės: pridėti, atimti, didinti, mažinti.
* Loginės: pasukimo, AND ir OR.
* Valdymo perdavimo: (ribojamas iki esamo ROM), besąlyginis, iškviečiama paprogramė ir grįžtama iš paprogramės.
* Įvesties/išvesties instrukcijos.
* Su pertraukimu susijusios instrukcijos – sustabdyti, įjungti, išjungti ir grįžti iš paprogramės.
* Kita – pernešimo vėliavėlės operacijos, dešimtainis koregavimas, banko pasirinkimas ROM ir indeksų registrams ir kt.

**Kokius instrukcijų formatus palaikė kiekviena architektūra? Pateikti 8-16 pavyzdžius.**

Motorola 6800 instrukcijos (ne visos):

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

Intel 4040 instrukcijos (ne visos):

A document with text and numbers

Description automatically generated

A screenshot of a document

Description automatically generated

Motorola 6800 ir Intel 4040 panašios komandos: ADD, SUB.

Motorola 6800 turi CMP komandą, o Intel 4040 ne ir pan.

1. **Kokius adresavimo būdus palaikė kiekviena architektūra?**

Motorola 6800 adresavimo būdai:

* Numanomas – duomenų reikšmė/duomenų adresas netiesiogiai susiejamas su instrukcija.
* Akumuliatorius – instrukcija reiškia, kad duomenys yra vienas iš akumuliatoriaus registrų.
* “Immediate” (nedelsiantis) – instrukcijoje pateikti 8 bitų arba 16 bitų duomenys.
* Tiesioginis – instrukcijoje pateiktas vieno baito operandas nurodo atminties adresą nuliniame puslapyje (0000h – 00FFh), kuriame yra duomenys.
* Išplėstinis – instrukcijoje pateiktas dviejų baitų operandas nurodo atminties adresą, kuriame yra duomenys.
* Santykinis – vienas baito poslinkis pridedamas prie kitos komandos adreso (programos skaitiklio registro turinys + 2). Poslinkis yra ženklas su ženklu nuo -127 iki +127.
* Indeksuotas – prie IX registro turinio pridedamas vieno baito operandas, gauta 16 bitų reikšmė yra rodyklė į atmintį, kurioje yra duomenys.

Intel 4040 adresavimo būdai:

* Registro (4 bitai).
* Tiesioginė atmintis.
* Registro netiesiogiai. Registras netiesiogiai gali pasiekti duomenis tik dabartinėje ROM.
* Nedelsiantis (“immediate”) (4 ir 8 bitų duomenys).

Adresavimo būdai panašūs dėl nedelsiančio adresavimo (“immediate”). Taip pat dėl tiesioginio adresavimo.

Skiriasi dėl akumuliatoriaus, išplėstinio, adresavimo, santykinio, indeksuoto būdo, kurį turi Motorola 6800, bet neturi Intel 4040.

O Intel 4040 turi registro netiesioginį būda, kurio neturi Motorola 6800.

1. **Kokios buvo kiekvienos architektūros I/O galimybės?**

Motorola 6800 neturi I/O “portų”, o Intel 4040: 16 4 bitų įvesties ir 16 4 bitų išvesties.

1. **Ar buvo palaikomi pertraukimai? Kuo pertraukimų mechanizmai buvo panašūs, kuo jie skyrėsi abiejose architektūrose?**

Motorola 6800 pertraukimai:

IRQ – maskuojamas pertraukimas. Įvykus pertraukimui programų skaitiklis, indeksų registras, akumuliatoriai ir sąlygų kodų registrai yra išsaugomi steke, kiti pertraukimai išjungiami ir procesorius peršoka į atminties vietos adresą, kurios adresas yra saugomas atmintyje FFF8h - FFF9h. Norėdami grįžti iš pertraukimo, apdorojimo rutina turėtų naudoti RTI nurodymus. Šį pertraukimą galima įjungti / išjungti naudojant CLI/SEI instrukcijas.

NMI – neužmaskuojamas pertraukimas. Įvykus pertraukimui programų skaitiklis, indeksų registras, akumuliatoriai ir sąlygų kodų registrai yra išsaugomi steke, kiti pertraukimai išjungiami ir procesorius peršoka į atminties vietos adresą, kurios atmintyje saugomas FFFCh - FFFDh. Norėdami grįžti iš pertraukimo, apdorojimo rutina turėtų naudoti RTI nurodymus. Šio pertraukimo negalima išjungti.

SWI – programinės įrangos pertraukimas. Šį pertraukimą galima iškviesti tik iš programos. Įvykus pertraukimui procesorius išsaugo programų skaitiklį, indeksų registrą, kaupiklius ir sąlygų kodų registrus, išjungia tolesnius pertraukimus ir peršoka į atminties vietos adresą, kurio adresas yra saugomas atmintyje FFFAh - FFFBh. Norėdami grįžti iš pertraukos, apdorojimo rutina turėtų naudoti RTI instrukcijas. Šio pertraukimo negalima išjungti.

Intel 4040 pertraukimai:

Procesorius turi vieną maskuojamą pertraukimą. Pertraukimą galima išjungti arba įjungti naudojant DIN ir EIN instrukcijas. Kai įvyksta pertraukimas, programos skaitiklio ir siuntimo registro (SRC) turinys išsaugomas. Norėdami grįžti iš pertraukos, pertraukimo apdorojimo kodas turi vykdyti BBS instrukcijas.

Motorola 6800 turi 3 pertraukimus, o Intel 4040 tik vieną.

Abi architektūros naudoja maskuotą pertraukimą, kai jis ivyksta procesorius išsaugo duomenis svarbius. Abiejose yra vykdomas specialus nurodymas, kad procesorius galėtų grįžti prie darbo, tačiau Motorola 6800 naudoja RTI nurodymus, o Intel 4040 naudoja BBS nurodymus.

Motorola 6800 išsaugo registrus: programos skaitiklį, indeksų registrą, akumuliatorių, sąlygų kodų registą, o Intel 4040 išsaugo tik programos skaitiklį ir siuntimo registrą (SRC).

1. **Kokius duomenų tipus palaikė kiekviena architektūra aparatūros lygyje?**

Motorola 6800 galėjo dirbti su 8 bitų sveikais skaičiais, o Intel 4040 su 4 bitų sveikais skaičiais.

**Ar buvo palaikoma fiksuoto kablelio, slankiojo kablelio aritmetika?**

Motorola 6800 nepalaikė tiesioginės fiksuoto kablelio aritmetikos. Nepalaikė ir slankiojo kablelio aritmetikos, tačiau slankiojo kablelio operacijos galėjo būti atliekamos programiškai, naudojant atitinkamus algoritmus ir operacijas su sveikaisiais skaičiais.

Intel 4040 taip pat nepalaikė fiksuoto kablelio aritmetikos. Kaip ir 6800, Intel 4040 nebuvo specialiai sukurta slankiojo kablelio aritmetikai, ir tokie skaičiavimai būtų turėję būti įgyvendinti programiškai.

**Ar sveikieji skaičiai buvo koduojami kaip ženklas-dydis, kaip vieneto papildinys (atvirkštinis kodas), dvejeto papildomas kodas? Kokius kitus „egzotiškus“ duomenų tipus palaikė?**

Motorola 6800 alaikė sveikųjų skaičių operacijas, tiek su ženklu, tiek be ženklo. Tokių skaičių kodavimas: ženklas-dydis (sign-magnitude) kodavimas arba, dažniausiai, dvejeto papildinys (two's complement) su 8 bitų pločiu, todėl nebuvo tiesiogiai naudojami tokie metodai kaip vieneto papildinys.

Intel 4040 palaikė sveikųjų skaičių operacijas, ir dažniausiai buvo naudojamas dvejeto papildinys (two's complement) kaip kodavimo metodas, kaip ir Motorola 6800.

Taigi: nei Motorola 6800, nei Intel 4040 nepalaikė tiesiogiai slankiojo kablelio ar fiksuoto kablelio aritmetikos. Abi architektūros naudojo paprastus sveikuosius skaičius ir dažniausiai dvejeto papildinį.

Egzotiškų duomenų tipų palaikymas, kaip dešimtainiai ar kompleksiniai skaičiai, buvo įgyvendintas tik per programinius metodus (pvz., naudojant BCD arba kuriant atitinkamas struktūras ir algoritmus). Tiesiogiai nepalaikė tokių tipų.

1. kius duomenų tipus palaikė kiekviena architek

tūra aparatūros

1. lygyj
2. Kokius duomenų tipus palaikė kiekviena architektūra aparatūros
3. Ar buvo palaikomi pertraukimai? Kuo pertraukimų mechanizmai buvo
4. panašūs, kuo jie skyrėsi abiejose architektūrose
5. Ar buvo palaikomi pertraukimai? Kuo pertraukimų mechanizmai buvo
6. panašūs, kuo jie skyrėsi abiejose archite
7. Koks buvo kiekvienos architektūros duomenų plotis (mašininis žo