

# Kompiuterių Architektūrų Palyginimas: IBM System/370 ir DEC PDP-11

Deimantė Balčiūtė, Bioinformatika 2k., 2025

1. Kokia buvo elementinė kompiuterio/procesoriaus bazė (relēs, vakuuminės lempos, tranzistoriai, magnetinės šerdys, integriniai grandynai (IC) – ar jie hibridiniai, mažo integracijos masto, Didelio integracijos masto (LSI), labai didelio integracijos masto (VLSI), ar tai buvo monokristaliniai šiuolaikiniai mikroprocesoriai)? Kokios buvo fizinės įrangos savybės (svoris, dydis, energijos suvartojimas)?

PDP-11, suskurtas „Digital Equipment Corporation“ 1970 metai (Bell, 1978), buvo 16 bitų minikompiuteris, modeliai buvo gaminami su diskrečia logika, mažo integracijos mąsto ir vidutinio masto integracijos TTL(Transistor – Transistor Logic) integrinius grandynus. Vėlesni modeliai, kaip PDP-11/03, buvo vieni pirmųjų, kurie naudojo didelio masto integraciją, o procesorius buvo įdiegtas keliuose lustuose. Monokristalinių mikroprocesorių nenaudojo. Prosesorius nebuvvo vienas čipas, o daug susijungusių integruotų grandinių per kelias spausdintinių plokščių. (Copilot, 2025). Svarbiausia architektūros dalis buvo UNIBUS magistralė: ji leido CPU bendrauti su pagrindine atmintimi ir periferinėms sistemoms siųsti ir gauti informacija. (Wikipedia, 2025)

PDP-11 modeliai buvo įvairaus dydžio. Modeliai, kurie montuojami į spintą buvo 13,3 cm x 48,3 cm x 64,8 cm svérė apie 24,8 kg, ant stalo statomi modeliai 15,2 cm x 56,5 cm x 72,4 cm, svérė apie 31,5 kg, ant žemės statomi modeliai 62,2 cm x 25,4 cm x 72,4 cm. PDP-11/84 44,5 kg. Dauguma modelių veikė su standartinėmis elektros srovėmis: JAV 120V ir Europa 240V. Didžiausiai PDP-11/84 spintos konfigūracijai reikėjo trifazės srovės. (Datapro Research Corporation, 1988). Elektros suvartojimas buvo įvairus: Mažesni modeliai suvartodavo 300-400W, o didelės sistemos – virš 1 kW. (Copilot, 2025)

IBM System/370, taip pat sukurtas 1970 metais, yra bendros paskirties sistema. Ją sudarė pagrindinė saugykla, CPU, vienas ar daugiau kanalų ir I/O prietaisai (IBM, 1975). Ankstyvieji modeliai nenaudojo mikroprocesorių, vietoj jų naudojo hibridinius integrinius grandynus ir monolitinius integrinius grandynus, kurie buvo mažo masto integracijos ir vidutinio masto integracijos. IBM sukūrė „Solid Logic Technology“ (SLT), kuri buvo hibridinė pakavimo technologija. Vėlesni modeliai naudojo didelio masto integraciją, tačiau procesoriai buvo sudaryti iš daugelio lustų ant plokščių ar kelių lustų modulių, o ne iš vieno mikroprocesoriaus. (Copilot, 2025).

Palyginus su PDP-11 modeliais, IBM System/370 buvo didelė kompiuterių sistema. Vien CPU užimdavo 2 m x 0,76 m x 0,9 m. Tipiskai visas sistema užimdavo nedidelį kambarį. Pagrindinės sistemos svoris siekė 900 – 1100 kg, o periferiniai prietaisai galėjo pridėti dar daugiau kilogramų svorio. Elektros naudojo labai daug – siekė 6-8 kW. Pilna sistema reikalavo trifazės srovės, oro kondicionavimo, tad energijos suvartojimas galėjo pasiekti ir 20 kW. (Copilot, 2025)

2. Kokio tipo architektūrą turėjo abu kompiuteriai? Ar jie buvo akumulatoriniai, registriniai, stekiniai (dėklo architektūros), atmintis-į-atmintį architektūros arba kokios nors kitos architektūros?

Abu, ir PDP-11 ir System/370, naudojo CISC(Sudėtingų Instrukcijų Kompiuterija) architektūrą: registras-į-registrą, registras-į-atmintį, atmintis-į-atmintį architektūrą. (Wikipedia, 2025) (Wikipedia, 2025)

3. Ar tai buvo beadresinės (stekinės), vieno adreso, dviejų adresų, trijų adresų ar keturios adresų mašinos?

PDP-11 dažniausiai naudojo du adresus, naudojo ir stekines, vieno adreso (Bell, 1978). System/370 taip pat naudojo du adresus. (IBM, 1974)

4. Kokie buvo registrai abiejose architektūrose? Ar šios architektūros iš viso turėjo registrus? Ar tai buvo bendrosios paskirties registrai, ar specializuoti registrai? Kiek registrų turėjo kiekviena architektūra? Kokie buvo šių registrų duomenų pločiai? Kokia buvo specifinė registrų paskirtis?

PDP-11 visas atminties nuorodų adresavimas naudoja 8 bendros paskirties registrus. Šie registrai funkcionavo kaip akumulatoriai(saugojo duomenis), rodyklės(saugojo duomenų adresus), indeksų registrai(registro duomenys pridedami prie antro instrukcijos žodžio, kad gauti kito operando adresą). Registrų duomenų plotis galėjo būti 1-3 žodžių, arba 16 – 48 bitų. R6 visada naudojamas kaip steko rodyklė, R7 kaip programos skaitiklis(PC). R0-R5 bendros paskirties registrai (Bell, 1978).

Šie registrai turėjo kelis režimus, kurie turi ir savo atidėtas versijas: registro, autoinkrementinė, autodekrementinė, indeksinė. Programos skaitiklis turi savo režimus: tiesioginis, absoliutus, reliatyvus, reliatyvus atidėtas. (Bell, 1978)

Taip pat yra papildomas 16 bitų registratoras – procesoriaus būsenos žodis. Jame saugomos būsenos vėliavėlės(neigama – N, nulis – Z, perpilda – V, perkėlimas - C), procesoriaus prioriteto lygis, atminties valdymo režimai. (Bell, 1978)

IBM System/370 turėjo 16 bendros paskirties registrų, kontrolės registrų po 32 bitus, 4 slankaus kablelio registrus po 64 bitus ir 64 bitų procesoriaus būsenos žodį. Valdymo registrai buvo naudojami virtualios atminties valdymui. (IBM, 1974)

Palyginimui: PDP-11 iš viso turėjo 16 + 1 registrą, o System/370 36 + 1 registrus.

5. Ar požymių bitai buvo naudojami šiose architektūrose? Kokie požymiai buvo naudojami?

Abi architektūros turėjo PSW(Processor State Word). PDP-11 naudojo standartinius požymius: neigama – N, nulis – Z, perpilda – V, perkėlimas – C. (Bell, 1978). Taip buvo įsiterpimo bitas ir žingsnio vykdymo bitas, kuris leido programą vykdyti po vieną žingsnį. (Wikipedia, 2025)

IBM System/370 naudojo panašaus principo bitus – sąlygos kodus. Juos 2 bitai, kurių reikšmė galėjo būti 0 - nulis, 1 – mažiau už nulį, 2 – daugiau už nulį, 3 – perpilda . Jie taip pat galėjo turėti kitas reikšmes priklausomai nuo konteksto. Visos galimos reikšmės pateikiame lentelėje. (IBM, 1983)

Instruction	Condition Code			
	0	1	2	3
RESET REFERENCE BIT EXTENDED RESUME SUBCHANNEL	R bit zero, C bit zero Function initiated	R bit zero, C bit one Status pending	R bit one, C bit zero Function not applicable	R bit one, C bit one Not operational
SET CLOCK SHIFT AND ROUND DECIMAL SHIFT LEFT (DOUBLE/SINGLE)	Set Zero Zero	Secure < zero < zero	-- > zero > zero	Not operational Overflow Overflow
SHIFT RIGHT (DOUBLE/SINGLE) SIGNAL PROCESSOR START SUBCHANNEL	Zero Order accepted Function initiated	< zero Status stored Status-pending	> zero Busy Busy	-- Not operational Not operational
STORE CHANNEL REPORT WORD STORE CLOCK	CRW stored Set	Zeros stored Not set	-- Error	-- Not operational
STORE SUBCHANNEL SUBTRACT, SUBTRACT HALFWORD SUBTRACT DECIMAL SUBTRACT LOGICAL	SCHIB stored Zero Zero --	-- < zero < zero Not zero, no carry < zero	-- > zero > zero Zero, carry > zero	Not operational Overflow Overflow Not zero, carry --
SUBTRACT NORMALIZED TEST AND SET TEST BLOCK TEST PENDING INTERRUPTION	Zero Left bit zero Usable Interruption code not stored	< zero Left bit one Not usable Interruption code stored	> zero -- -- --	-- -- -- --
TEST PROTECTION	Can fetch, can store	Can fetch, cannot store	Cannot fetch, cannot store	Translation not available
TEST SUBCHANNEL	IRB stored; subchannel status- pending	IRB stored; subchannel not status- pending	--	Not operational
TEST UNDER MASK TRANSLATE AND TEST UPDATE TREE	All zeros All zeros Equal	Mixed Incomplete Not equal or no comparison	-- Complete --	All ones -- Method 2, GR5 nonzero, GR0 negative Overflow
ZERO AND ADD	Zero	< zero	> zero	

Explanation:

```

> zero Result greater than zero
< zero Result less than zero
=< 256 Equal to, or less than, 256
> 256 Greater than 256
High First operand high
Low First operand low
Length Length of first operand
OCB Operand-control bit

```

Lentelė 1 IBM System/370 sąlygos kodai. (IBM, 1983)

6. Koks buvo kiekvienos architektūros duomenų plotis (mašininis žodis)?

PDP – 11 mašininis žodis buvo 16 bitų(Wikipedia, 2024), o IBM žodis siekė 32 bitus, tačiau gebėjo priimti ir baito ilgio, pusę žodžio, dvigubus žodžius. (IBM, 1975) (Wikipedia, 2025)

7. Koks buvo kiekvienos sistemos atminties išdėstymas? Ar adresų buvo erdvė ištisinė, ar ji buvo suskirstytas į segmentus, puslapius, atminties į bankus? Koks buvo (efektyvus) adreso plotis? Koks buvo maksimalus įmanomas atminties kiekis kiekvienoje sistemoje? Koks buvo tipiškas atminties kiekis, su kuria sistema buvo naudojama?

Senesniuose modeliuos PDP-11 atmintis buvo ištisa. Programos naudojo 16 bitų virtualius adresus, kurių apatiniai 56 KB rodė į tuos pačius Unibus adresus. Viršutiniai 8 KB buvo naudojami I/O įrenginiams. Vėlesni modeliai implementavo atminties valdymo įrenginį(MMU), kur 64 KB

atvaizduojami į 256 KB Unibus adresuose, naudojo puslapiavimą. Kiekvienas puslapis buvo 4096 žodžių ilgio. Buvo galimi keli atvaizdavimo variantai: 16, 18 ir 22 bitų atvaizdavimas. Skyrėsi fizinės atminties kiekis. Virtualaus adreso plotis visada 16 bitų, efektyvus priklausė nuo modelio: ankstyvujų – 16 bitų, maksimalus atminties kiekis 64 KB, su 18 bitų Unibus magistrale – 18 bitų su maksimalia 256 KB atmintimi, vėlesni su 22 bitų magistrale – 22 bitų su maksimalia 4 MB atmintimi. (Digital Equipment Corporation, 1979). Tipiškai ankstesniuose modeliuose buvo naudojama 32-64 KB, vėlesniuose - 128-256 KB. (Copilot, 2025)

IBM System/370 turėjo 16 MB virtualios atminties, kuri turėjo segmentinius puslapius. Segmentų dydis – 64 KB ir jų buvo 256. Puslapiai buvo dviejų dydžių priklausomai nuo OS/VS versijos. OS/VS1 viename segmente turėjo 32 puslapius po 2 KB, OS/VS2 segmentas turėjo 16 puslapį po 4 KB (IBM, 1973). Atmintis išdėstyta atminties bankuose, dokumentacijoje vadinam logine atmintimi. Taip pat naudojama dviejų atminties lygių sistema: labai greita buferinė atmintis arba šiaisiai vadinama „cache“ ir lėtesnė pagrindinė atmintis. Buferinė atmintis dažniausiai buvo tiesiogiai asocijuojama su CPU arba I/O procesoriumi. Ankstyvuose modeliuose adresų plotis buvo 24 bitų, vėliau buvo praplėstas iki 31 bitų. (IBM, 1983). Tipiškas atminties naudojimas priklausė nuo darbo paskirties. Mažesnės instaliacijos: 256–512 KB, vidutinės sistemos: 1–4 MB, didžiausios: 4–16 MB.

8. Ar buvo palaikoma virtualioji atmintis ir kaip? Ar virtuali atmintis buvo realizuoti, naudojant puslapiavimą, segmentavimą, abu šiuos mechanizmus?

PDP-11 virtualius adresus galėjo naudoti tik vėlesni modeliai, pirmi turėjo tik realius adresus. Virtualaus adreso pradžioje buvo 3 bitai, kurie nusako kuriame puslapyje yra fizinis adresas. Toliau 13 bitų adresas reliatyvus puslapio pradžiai (Digital Equipment Corporation, 1979).

IBM System/370 nuo pat sukūrimo galėjo naudoti virtualią atmintį. System/370 naudojo atskirą įrenginį adresų transliacijai – DAT (Dinamiškas Adresų Transliacija). Naudojamos dvi lentelės: segmentų ir puslapių lentelės. Transliacija yra kontroluojama dviejų PSW bitų – 5 bitas DAT režimas, 16 bitas adreso erdvės kontrolės bitas. Jei DAT bitas išjungtas, CPU ir visi adresai skaitomi kaip realūs adresai. Jei DAT įjungtas, o kontrolės adresas išjungtas, adresai naudojami kaip pirminiai virtualūs adresai, kai abu bitai įjungti – kaip antriniai virtualūs adresai. (IBM, 1983)

9. Kokia buvo kiekvienos architektūros komandų sistema (ISA)? Kiek mašinos komandų turėjo kiekviena architektūra? Kokios buvo instrukcijų (komandų) klasės? Kokius instrukcijų formatus palaikė kiekviena architektūra? Pateikite 8–16 instrukcijų pavyzdžiai. Kokios komandos buvo panašios abi architektūros? Kurios komandos skyrėsi?

Abi architektūros implementavo CISC. PDP-11 turėjo 7 komandų kategorijas ir su visais plėtiniais virš 100 komandų: (Digital Equipment Corporation, 1983):

- Vieno operando – pirma žodžio dalis – operacijos kodas, antra – operando vieta
- Dviejų operandų – pirma žodžio nusako koks bus atliktas veiksmas, atroji dviejų operandų vieta
- Išsišakojimo – pirma žodžio dalis nusako koks bus atliktas veiksmas, antroji nuo toliau vyks programa.
- Nušokimo ir subrutinos – nusakomas operacijos kodu ir adresu
- Trapų – turi tik operacijos kodą

- Kiti
- Būsenos kodo – skirti išvalyti ir nustatyti būsenos kodus.

Pavyzdžiai:

- **MOV R1, (R2)+**; perkelia reikšmę, kuri yra R2 saugome adrese į R1, padidinti R2
- **CLR R5**; išvalo nurodytą registrą
- **ADD #5, R0**; prie R0 reikšmės prideda 5
- **SUB R2, R1**; R1 - R2, reikšmė įrašoma į R1
- **BIS #0x80, R1**; R1 OR 0x80, bitai įrašomi į R1 registrą
- **TST R0**; testuoja būsenos bitus
- **ASH R0**; aritmetinis poslinkis dešinėn
- **BR .+6**; besalyginis šuolis priklausomas nuo programos skaitliuko.
- **BEQ target**; šuolis, jei PSW Z bitas = 0
- **Ccc**; išvalo būsenos bitus.

IBM System/370 priklausomai nuo modelio ir plėtinių turėjo apie 150 – 200 komandų. Operandai galėjo būti registro reikšmė, tiesioginis skaičius arba galėjo būti patalpintas ir paimamas iš atminties (IBM, 1974). Komandos skirstomos į (IBM, 1983):

- Bendras – naudojama informacijos keitimui registruose, atmintyje arba tiesiogiai paduodami komandą srauto.
- Dešimtainių skaičių komandos – papildomos komandos veiksmams su dešimtainiais skaičiais.
- Slankaus kablelio komandos – naudojamos skaičiavimams ir veiksmams su įvairaus ilgio ir dydžio slankaus kablelio skaičiais.
- Kontrolės komandos – leidžiamos naudoti tik tada, kai CPU yra prižiūrėtojo režime.
- I/O komandos.

Pavyzdžiai:

- **LA R1, 100 ;R1 = 100**
- **LR R2, R1 ;R2 = R1**
- **XR R1, R1 ;R1 = R1 XOR R1**
- **AR R1, R2 ;R1 = R1 + R2**
- **SR R1, R2 ;R1 = R1 – R2**
- **CR R1, R2 ;palygina registrus ir nustato būsenos kodą(CC)**
- **BE target ;palygina registrus, jei CC = lygūs, tada peršoka**
- **B target ;peršoka**
- **LA R1,4(R1) ;i R1 įrašo reikšmę, kuri yra adrese R1 + 4**

Abiejų ISA yra panašūs, bet PDP-11 turi mažiau komandų, daug automatinių veiksmų. IBM turi daug tikslų komandų daugeliui atvejų, daugelį veiksmų reikia nustatyti tiksliai.

~~10. LA R1,10 ; R1 = 10~~

11. Kokius adresavimo būdus palaikė kiekviena architektūra? Kurie režimai buvo panašūs, o kurie skyrėsi?

PDP-11 turėjo aštuonis režimus (Digital Equipment Corporation, 1979):

- **Rn** – registrinis – operandas yra pačiame registre.
- **(Rn)** – registrinis netiesioginis – registras nurodo operando adresą atmintyje.
- **(Rn)+** - automatinis didinimas – registras nurodo operando adresą. Po kreipimosi į operandą, registro reikšmė padidinama.

- $@(Rn)^+$  - netiesioginis automatinis didinimas – registras nurodo adresą, po viso nuskaitymo, registras padidinamas 2.
- $-(Rn)$  – automatinis mažinimas – registras nurodo operando adresą. Po kreipimosi registro reikšmė sumažinama.
- $@-(Rn)$  – netiesioginis automatinis mažinimas – registras nurodo adresą, prieš kreipimąsi registro reikšmė suamžinama 2.
- $X(Rn)$  – indeksinis – operando adresas yra  $X + [Rn$  turinys].
- $@X(Rn)$  – netiesioginis indeksinė – operando adresas yra  $X + [Rn$  turinys]

PC registras (R7) taip pat gali būti naudojamas kaip bendros paskirties registras, tačiau po kiekvieno kreipimosi skaitliukas padidinamas 2. PC turi dar kelis režimus:

- $#n$  – tiesioginis – operando reikšmė yra žodyje iškart po instrukcijos
- $@#A$  – absoliutus – operando adresas yra žodyje iškart po instrukcijos
- $A$  – reliatyvus – reikšmės adresas yra poslinkio, kuris saugomas žodyje iškart po instrukcijos, ir  $PC+2$  suma.
- $@A$  – reliatyvus netiesioginis – reikšmės adreso adresas yra poslinkio, kuris saugomas žodyje iškart po instrukcijos, ir  $PC+2$  suma.

IBM System/370 adresai konstruojami sumuojuant tris skaičius: bazinis adresas, indeksas ir poslinkis. Bazinis adresas yra 24 bitų skaičius laikomas bendros paskirties registre, kurį nurodė pati programa. Jis gali būti naudojamas nurodyti individualios programos ir duomenų erdvę. Indeksas yra 24 bitų skaičius taip pat laikomas bendros paskirties registre, kuris nurodytas pačios programos. Naudojamas tik RX režime. Poslinkis yra 12 bitų skaičius laikomas instrukcijos erdvėje. Gali nurodyti už bazinio adreso nurodytos vietas. (IBM, 1974)

Adresavimo būdai (IBM, 1983):

- RR – abi reikšmės yra imamos iš nurodytų registrų.
- RX – pirmas imamas iš registro R, X – indeksuotas adresas.
- RS – naudojamas trečias registras, kuris nurodo adresą.
- SI – naudojamas adresas ir tiesiogiai nurodyta konstanta.
- S – naudojamas atminties adresas ir netiesioginis operandas.
- SS – naudojami du atminties adresai.

RX, RS, SI, S, SS adresai yra sudaromi pagal anksčiau nusakyta metodą. Bazinis adresas yra saugomas nurodytame registre, o poslinkis nurodo vietą nuo nurodytos vietas. (IBM, 1983)

PDP-11 Rn,  $#n$ ,  $X(Rn)$  yra atitinkamai panašūs, beveik vienodi, į System/370 RR, SI, RX būdus. Netiesioginis adresavimas panašus abiejuose, tačiau minikompiuteryje tai buvo vienas pagrindinių režimų, o System/370 atitinkmuo atminties tipo(S, SS, SI, RX) atvejis. S/370 turėjo automatinius didinimo ir mažinimo režimus, ko S/370 neturėjo.

PDP-11 architektūra pasižymėjo ortogonalumu ir lankstumu – adresavimo būdai, instrukcijos ir registrų buvo nepriklausomi. (Wikipedia, 2025)

## 12. Kokios buvo kiekvienos architektūros I/O galimybės?

PDP-11 visų periferinių įrenginių, CPU ir atminties sąsaja vyko per Unibus magistralę. Tai reiškė, jog įrenginiai galėjo būti tiesiogiai prijungti prie sistemos, dalinti tiesiogine prieigai prie atminties (direct memory access DMA). Unibus taip pat turėjo atskirą atminties dalį būtents sasažai su I/O įrenginiais. Komunikacija gali įvykti dvem signalais: BR – bus request – magistralės užklausa, siunčiamas kai reikia pertraukti CPU darbą ir NPR – non-processor request – procesoriaus nereikalaujanti užklausa, kai reikia tik apsikeisti, pasidalinti duomenimis. (Digital

Equipment Corporation, 1979). Turėjo daug palaikomų įrenginių: diskai, magnetiniai kaupikliai, spausdintuvai ir t.t.(Copilot)

IBM Systems/370 naudojo kanalų sistemą, kuri leido visiškai atriboti CPU nuo periferinių įrenginių. Kanalai buvo atskiri procesoriai, kurie buvo atsakingi už informacijos pernešimą tarp įrenginio ir atminties. Buvo trys tipai: baito multiprekserinis, bloko multiplekserinis, selektorinis. Multiplekseriniai galėjo sujungti duomenis iš kelių įrenginių, o selektorinis iš vieno. S/370 turėjo ir atskiras komandas I/O procesams. (IBM, 1974)

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

PDP-11 pertraukimai buvo automatiniai ir vykdomi per Unibus. Įrenginiai gali pertraukti CPU tik gavę magistralės kontrolę. Įrenginys paprašo pertraukti siūsdamas INTR signalą ir pertraukimo vektorių. Vektorius nukreipia CPU į atminties vietą, į kurią programa anksčiau įkėlė pertraukimo aptarnavimo rutinos (ISR) pradinį adresą ir PSW, kuris nurodo užklausos prioritetą. CPU priima pertraukimo vektorių ir patvirtina SSYN (Slave SYNC), kad vektorius buvo priimtas. Įrenginys perleidžia magistralės kontrolę CPU, išvalydamas INTR, pašalindamas vektorių ir išvalydamas BBSY. CPU patvirtina, išvalydamas SSYN, išsaugo informaciją, kurios reikia grąžti į pertrauktą programą (šiam tikslui naudojamas atmintyje esantis aparatinės įrangos stekas), ir pradeda pertraukimo apdorojimo seką. Kai pertraukimo operacija baigianturėja, CPU pašalina informaciją, kuri buvo saugoma steke, ir pagal išsaugotus PC ir PSW, grąžta į tą vietą, kur buvo pertrauktas. Pertraukimas gali būti iškviečiamas išorinių įrenginių, įvykus programinei klaidai, sutrikus elektros tiekimui. (Digital Equipment Corporation, 1979)

PDP-11 turi prioriteto sistemą, nuo kurio priklauso kuris įrenginys perima magistralės kontrolę. Sistema yra dvimatė – yra vertikalūs prioriteto lygai ir horizontalūs. Yra tik penki vertikalūs prioritetų lygai – NPR, BR7, BR6, BR5 ir BR4. Kai prie to paties lygio prijungiamas keletas įrenginių, atsiranda horizontalus prioritetas. Jei daugiau nei vienas įrenginys pateikia užklausą tame pačiame horizontaliame lygyje, didžiausią prioritetą turi tas įrenginys, kuris elektriškai yra arčiausiai CPU. Aukščiausias lygis BR4. (Digital Equipment Corporation, 1979)

IBM System/370 pertraukimų sistema yra griežtesnė. Ji paremta pertraukimų klasėmis ir PSW apsikeitimu. Yra šešios pagrindinės klasės: mašinos patikra, priežiūros iškvietimas, programos pertraukimas, išorinis pertraukimas, I/O pertraukimas, paleidimo iš naujo. (IBM, 1974)

Pertraukimas susideda iš dabartinio PSW išsaugojimo kaip senojo PSW, informacijos, identifikuojančios pertraukimo priežastį, išsaugojimo ir naujo PSW paėmimo. Apdorojimas tesiama pagal naują PSW. Senasis PSW paprastai yra adresas komandos, kuri būtų vykdoma toliau, jei pertraukimas nebūtų įvykęs, taip leidžiant testi pertrauktos programos vykdymą. Programos ir priežiūros iškvietimo pertraukimų atveju saugoma informacija taip pat apima kodą, kuris identifikuoja paskutinės vykdytos instrukcijos ilgį, taip leidžiant programai reaguoti į pertraukimo priežastį. Kai kurių programos sąlygų atveju, kai įprasta reakcija yra pertraukimą sukėlusios instrukcijos pakartotinis vykdymas, instrukcijos adresas tiesiogiai identifikuoja paskutinę vykdytą instrukciją. Išskyrus paleidimą iš naujo, pertraukimas gali įvykti tik jei CPU yra veikimo būsenoje. Paleisti iš naujo galima ir neveikimo būsenoje. (IBM, 1983)

Procesai panašūs tuo, kad abu naudojo pertraukimus panašioms reikmėms. Esminis skirtumas yra pertraukimo apdorojimo adreso gavimas: PDP naudojo vektorinį metodą, o S/370 naudojo fiksuočių vietų metodą. Abu išsaugodavo PSW ir PS, bet PDP saugojo steke, o S/370 fiksuojoje atminties vietoje.

14. Kokius duomenų tipus palaikė kiekviena architektūra aparatūros lygyje? Ar buvo palaikoma fiksuočio kablelio, slankiojo kablelio aritmetika? Ar sveikieji skaičiai buvo koduojami kaip ženklas-dydis, kaip vieneto papildinys (atvirkštinis kodas), dvejeto papildytu (papildomas kodas)? Kokius kitus „egzotiškus“ duomenų tipus palaikė architektūra (pvz. dešimtainius skaičius, kompleksinius skaičius ir kt.)?

PDP-11 palaikė sveikuosius skaičius, simbolų eilutes, dešimtainių eilutes, slankaus kablelio skaičius: (Digital Equipment Corporation, 1983)

- Sveikieji skaičiai galėjo būti 8 bitų arba 16 bitų žodžio. Koduojami kaip dvejeto papildinys, MSB nurodė ženkla: jei 1 – neigiamas, jei 0 – teigiamas.
  - Ilguosius skaičius saugomas dviem žodžiais. Pirmame – vyresnieji bitai, antrame – jaunesnieji. Instrukcijose užrašoma atvirkščiai dėl suderinamumo.
- Simbolų eilutės skirstomos į tris tipus:
  - Simbolis – 8 bitų baitas, ilgis prilyginamas vienetui. Naudojant instrukcijoms bendros paskirties registre, simbolis užima žemesnę baito pusę, instrukcijų sraute – žemesnę žodžio pusę. Kitos pusės privalo būti lygios 0, kitaip instrukcijos rezultatas bus nenuspėjamas.
  - Simbolų eilutė – atmintyje iš eilės einantys baitai, kuriuose yra simboliai. Naudojami kaip operandai nurodant du iš eilės einančius registrus, kur į Rx įrašomas eilutės ilgis, o į Rx + 1 pirmo simbolio adresas .
  - Simbolų aibė – poaibį sudaro 256 galimi simboliai, kurie gali būti užkoduoti viename baite. Naudojamas kaip operandas nurodant du iš eilės einančius registrus, kurių pirmame vyresnioji pusė yra 0, o jaunesnioji nurodo kaukę, sekantis registras nurodo lentelės adresą. Lentelę sudaro 256 baitai. Kiekvienas baitas nurodo iki 8 ortogonalų simbolų poaibį. Kaukę nurodo kokia šių poaibų kombinacija bus naudojama.
- Dešimtainės eilutės saugomas panašiai kaip ir simbolų eilutė: iš eilės einantys baitai. Skaičiai gali būti su ženklu arba be ženkliai. Deskriptoriuje nurodomi du žodžiai: pirmame ilgis, duomenų tipas, sekanciame eilutės adresas. Buvo skirta versti skaičius į simbolų eilutes naudojant ASCII kodus.
- Nevisi PDP-11 procesorių modeliai galėjo naudoti slankaus kablelio skaičius. Tie modeliai kurie turėjo FP11 palaikė du tipus – viengubo tikslumo *float* – 32 bitų, ir dvigubo *double* – 64 bit. Intervalus:  $2,9 \cdot 10^{-37} - 1,7 \cdot 10^{38}$ . Slankaus kablelio skaičių matematinė forma:  
$$2^k \cdot f \cdot K$$
 – laipsnis, f – trupmena. Laipsnis buvo 8 bitų ir galėjo būti nu -127 iki 127, pirmas bitas laikomas ženklo bitu.

IBM System/370 palaikė šiuos duomenų tipus (IBM, 1974):

- Sveikieji skaičiai. Naudojo dvejeto papildinį kaip ir PDP-11
- Fiksuočio kablelio skaičiai. Naudoti kaip sveikieji skaičiai. Gali būti 16, 32 arba 64 bitų ilgio. Neigiamų skaičių aibė bus visada vienu didesnis nei teigiamų skaičių aibė dėl pirmo ženklo bito, nes nėra neigiamo nulio.
- Dešimtainiai skaičiai susideda iš atmintyje iš eilės einančių baitų. Buvo dviejų formatų „zoned“ ir „packed“. „Zoned“ tipo kiekvienas skaičiaus baitas susidėjo iš dviejų pusbaicių: vyresniajame buvo zonas bitai, išskyrus jauniausią skaičiaus zoną, kuri galėjo būti skaitoma kaip ženklas, jaunesnieji bitai vadinami skaitiniais,

- kurie nurodo skaitmenį. „Packed“ tipo taip pat dalin batus į pusbačius, visi nurodo skaitménį, išskyrus jauniausią pusbaitį, kuris nurodo ženkla.
- Slankaus kablelio skaičiai turėjo fiksuotus formatus, galėjo būti 4, 8 arba 16 bautų ilgio. 4 ir 8 bautų pirmas bitas yra ženklo bitas, toliau einantys 7 bitai nurodė laipsnį, likę bitai nurodė trupmeną: 6 arba 14 šešioliktainius skaitmenis.
  - Simbolių eilutės nebuvo atskiras duomenų tipas, tai tiesiog buvo 8 bitų sekos, kurios būdavo koduojamos EBCDIC koduote. Naudojant kaip operandą visda nurodomas eilutės ilgis. (Copilot, 2025)

15. Kokia buvo kiekvienos sistemos greitaveika? Kokie buvo taktinių generatorių dažniai, vidutinis/mažiausias/didžiausias ciklų skaičius, reikalingas kiekvienai komandai įvykdyti, vidutinė sistemos greitaveika? Kuri sistema buvo našesnė? Koks buvo kainos ir našumo santykis?

16. Ar architektūros naudojo spartinančią atmintį? Jei taip, kokio dydžio?

IBM System/370 naudojo visi modeliai, dydis priklausė nuo modelio. Pavyzdžiu Modelis 165 turėjo 8 KB arba 16 KB spartinančios atminties. (IBM, 1970).

Tik kai kurie PDP-11 modeliai naudojo spartinančią atmintį. 34A turėjo 2 KB, 44 turėjo 8 KB, 60 turėjo 2 KB, 70 turėjo 2 KB.

17. Kokios buvo tipinės kiekvienos architektūros taikymo sritys? Kaip šios architektūros buvo naudojamos? Trumpai apibūdinkite (vienoje pastraipoje) vieną konkretų kiekvienos architektūros panaudojimo pavyzdį.

PDP-11 tipiškai naudojami moksliniuose skaičiavimuose, laboratorijose, universitetuose, realiojo laiko sistemose, telekomunikacijoje, medicinoje, ankstyvuosiuose Unix serveriuose ir daug kitur. Tai buvo tam metui mažas įrenginys, tad turėjo panaudojimo būdų. (Wikipedia, 2025)

MIT Dirbtinio Intelekto Laboratorija naudojo PDP-11/45 su RSX-11M operacine sistema robotikosstudijoms ir MIT Lisp Mašinos maketų kūrimui 1970-ujų viduryje. Įrenginys kontroliavo roboto ranką su momentine programine įranga, kuri apdorojo davinlių informaciją (padėties enkoderiai, jėgos grižtamasis ryšys), apskaičiavo atvirkštinę kinematiką ir perdavė komandų signalus pavaroms. PDP-11 16-bitų struktūra su plačiu adresavimo būdų pasirinkimu ir nutraukimų sistema idealiai tiko tokiems momentinio valdymo darbams, o RSX-11M kelių užduočių vykdymo galimybės leido vienu metu atliliki davinlių nuskaitymo, skaičiavimo ir pavarų kontrolės veiksmus. (Copilot, 2025)

IBM System/370 naudojo didžiosios korporacijos ir vyriausybės institutai: bankų ir finansų duomenų apdorojimas, draudimo kompanijų kliento valdymo sistemos, vyriausybės registrų ir mokesčių sistemos, didelių gamyklų apskaitos ir ERP sistemos, aukštųjų mokyklų administracijos sistemos, aviakompanijų rezervacijų sistemos.

„American Airlines“ SABRE (Specializuota Automatinė Verslo Tyrimų Aplinka) rezervavimo sistema, pradėta naudoti 1960-ujų viduryje ir vėliau perkelta į System/370, apdorodavo dešimtis tūkstančių skrydžių užsakymų per dieną iš šimtus agentų terminalų visoje šalyje. Sistema taikė IMS/DC (Duomenų Komunikacijos) transakcijų valdymą ir IMS/DB hierarchinę duomenų bazę, saugodama skrydžių grafikus, keleivių informaciją ir laisvų vietų galimybę. Kiekvienas rezervavimas buvo realaus laiko sandoris, kuris turėjo per kelias sekundes patikrinti vietų prieinamumą, pakeisti duomenų bazę ir patvirtinti užsakymą, o System/370

galingas I/O kanalų dizainas ir CICS sandorių apdorojimo sistema garantavo, kad sistema galėtų apdoroti šimtus tuo pačiu metu vykstančių rezervacijų be informacijos praradimo ar nesuderinamumo. (Copilot, 2025)

18. Kiek programinės įrangos buvo parašyta kiekvienai aptariamai architektūrai, ar ji (vis dar) prieinama, kur ji buvo naudojama? Kokie buvo prieinami kompiliatoriai ir programavimo įrankiai (derintojai, profiliuotojai, surinkėjai)? Kokios programinės įrangos bibliotekos buvo prieinamos?

PDP-11:

**Kompiliatoriai ir programavimo kalbos** (Copilot, 2025):

- **MACRO-11** - standartinis simbolinis asembleris
- **FORTRAN IV/FORTRAN IV-PLUS** - moksliniams ir inžinerinės skaičiavimams
- **BASIC-PLUS** - išplėsta BASIC versija su matematinėmis funkcijomis
- **C kalba** - vienas iš pirmųjų C kompiliatorių (1972/73 m.) buvo sukurtas būtent PDP-11
- **COBOL** - verslo programoms
- **PL-11** - CERN sukurtą kalba sistemą ir mokslo programavimui

**Programavimo įrankiai:**

- **Asembleriai:** MACRO-11 su makrokomandų palaikymu
- **Derintuvai:** ODT (Octal Debugging Technique) - aparatūrinio lygio derintuvas
- **Diagnostikos įrankiai:** XXDP diagnostikos rinkinys
- **UNIX aplinkoje:** db derintuvas C programoms

**Bibliotekos:**

- **PLOT-11** - braižymo funkcijos
- **STAT-11** - statistikos bibliotekos
- **EG-11** - grafikos terminalų valdymas
- **COGO-11, PCS-11, CSP-11** - specializuotos inžinerinės ir verslo bibliotekos

IBM S/370:

**IBM System/370 Architektūra**

**Kompiliatoriai** (Copilot, 2025):

- **IBM COBOL, PL/I, FORTRAN** - pagrindinės verslo ir mokslo kalbos
- **IBM Assembler** - aparatūros valdymui
- **Dignus C/C++** - modernesnės kalbos
- **Pascal** - švietimo tikslams

**Programavimo įrankiai:**

- **ISPF Editor** - pagrindinis redaktorius ir kūrimo aplinka
- **IBM Debug Tool** - pažengės derintuvas
- **HLASM (High Level Assembler)** - standartinis asembleris
- **SCLM** - kodo versijų valdymas

**Bibliotekos ir API:**

- **Language Environment** - vykdymo aplinka kelioms kalboms
- **CICS, IMS, DB2** - transakcijų ir duomenų bazės bibliotekos

# Bibliografija

Bell, C. G. M. C. J. M. E. J., 1978. *Computer Engineering: A DEC View of Hardware Systems Design*, s.l.: Digital Press.

Copilot, 2025. [Tinkle]

Available at: <https://github.com/copilot/c/f75b5824-7714-41eb-974f-ca10c0f75a01>

Užklausa: "Koks buvo tipiškas atminties kiekis, su kuria PDP-11 buvo naudojama?"  
[Kreiptasi 07 12 2025].

Copilot, 2025. [Tinkle]

Available at: <https://github.com/copilot/c/a9e3f7ad-c8b4-4f39-9507-e2c974f11399>

Užklausa: What was the basic computer/processor architecture of PDP-11 and IBM Systems/370 - were they hybrid, low-scale integration, large-scale integration (LSI), very large-scale integration (VLSI), or were they monocrystalline modern microprocessors)? What were the physical characteristics of the equipment (weight, size, power consumption)?  
[Kreiptasi 05 12 2025].

Copilot, 2025. [Tinkle]

Available at: <https://github.com/copilot/c/7e5819f7-7932-42b4-9228-4965dd7d5dde>

Užklausa: Ar IBM system/370 naudojo simbolių eilutes?  
[Kreiptasi 08 12 2025].

Copilot, 2025. [Tinkle]

Available at: <https://github.com/copilot/c/291c8b91-6dc2-452c-aa50-7d1abaae16ea>

Užklausa: Kiek programinės įrangos buvo parašyta PDP-11 ir IBM system/370 architektūrai, ar ji (vis dar) prieinama, kur ji buvo naudojama? Kokie buvo prieinami kompiiliatoriai ir programavimo įrankiai (derintojai, profiliuotojai, surinkėjai)? Kokios programinės įrangos bibliotekos buvo prieinamos?

[Kreiptasi 08 12 2025].

Datapro Research Corporation, 1988. *DEC PDP-11 Family*, s.l.: DATAPRO RESEARCH CORPORATION.

Digital Equipment Corporation, 1983. *PDP-11 Architecture Handbook*. s.l.:s.n.

Digital Equipment Corporation, 1979. *PDP-11 Processor Handbook*. s.l.:DIGITAL.

IBM, 1970. *A Guide to the IBM System/370 Model 165*. 1st mont. s.l.:s.n.

IBM, 1973. *Introduction to Virtual Storage in System/370*. s.l.:s.n.

IBM, 1974. *IBM System/370 Principles of Operation*. 4th mont. s.l.:s.n.

IBM, 1975. *IBM System/370 System Summery*. 5th mont. s.l.:s.n.

IBM, 1983. *IBM System/370 Extended Architecture Principles of operation*. s.l.:s.n.

Wikipedia, 2025. *IBM System/370*. [Tinkle]

Available at: [https://en.wikipedia.org/wiki/IBM\\_System/370](https://en.wikipedia.org/wiki/IBM_System/370)

[Kreiptasi 05 12 2025].

Wikipedia, 2025. *PDP-11*. [Tinkle]

Available at: <https://en.wikipedia.org/wiki/PDP-11>

[Kreiptasi 05 12 2025].

Wikipedia, 2025. *Unibus*. [Tinkle]

Available at: <https://en.wikipedia.org/w/index.php?title=Unibus&action=history>

[Kreiptasi 05 12 2025].

Lentelė 1 IBM System/370 sąlygos kodai. (IBM, 1983) ..... 3