

# IBM System/370 vs. PDP-11

## Architektūrų palyginimas

Viktorija Ramonaitė, Bioinformatika, 2 k., 2024

IBM System/370	PDP-11	Pastebėjimai
<b>Kokia buvo elementinė kompiuterio/procesoriaus bazė, kokios buvo fizinės įrangos savybės?</b>		
Elementinė procesoriaus bazė yra monolitiniai integriniai grandynai (IC) su mažo integracijos masto ir vidutinio integracijos masto (LSI) technologijomis. Šie monolitiniai integriniai grandynai yra monolitinės sistemos technologijos (MST) dalis (Wikipedia, IBM System/370, 2024). Ši sistema užimdavo daug vietos – jai reikėdavo viso kambario (Wikipedia, IBM System/370, 2024).	Elementinė procesoriaus bazė yra integriniai grandynai (IC), kurie yra didelio integracinio masto (LSI) (Wikipedia, PDP-11, 2024). Šios sistemos dydis varijuoja, tačiau buvo kurti modeliai nuo to meto vadinamų mikrokompiuterių iki didesnių sistemų, kurios galėjo užimti daug vietos (PDP-11 Architecture Handbook, 1983, 13-22 p.)	Abiejų kompiuterių elementinė bazė pagrinde yra integriniai grandynai (IC), tačiau jie tuo pačiu ir skiriasi. IBM System/370 turi mažo ir vidutinio integracijos masto grandyno, o PDP-11 didelio masto. Taip pat jų fizinės savybės, kaip dydžiai skyrėsi. IBM System/370 buvo dideli ir užimdavo daug vietos, o PDP-11 galėjo būti palyginus mažo dydžio.
<b>Kokio tipo architektūrą turėjo abu kompiuteriai?</b>		
Šio kompiuterio architektūra buvo grįsta operavimu registrais ir atmintimi. Galėjo būti registras į registrą, registras į atmintį arba atmintis į atmintį (Wikipedia, IBM System/370, 2024).	Šio kompiuterio architektūra buvo grįsta operavimu registrais ir atmintimi. Galėjo būti registras į registrą, registras į atmintį arba atmintis į atmintį (Wikipedia, PDP-11, 2024).	Abu kompiuteriai turi tokią pačią architektūrą, kuri operuoja su registrais ir atmintimi.
<b>Ar tai buvo beadresinės (stekinės), vieno adreso, dviejų adresų, trijų adresų ar keturios adresų mašinos?</b>		
Tai yra dviejų adresų mašina, kontrolės registras 1 saugo pirmąjį adresą, o CR7 saugo antrąjį (Wikipedia, IBM System/370, 2024)	Tai dažniausiai yra dviejų adresų mašina (Wikipedia, PDP-11, 2024), bet yra lanksti, nes gali turėti ir stekinį adresą (PDP-11 Architecture Handbook, 1983, 53 p.)	Nors abiejų kompiuterių pagrindas yra dviejų adresų valdymas, vis dėl to PDP-11 yra lankstesnis nei IBM System/370. PDP-11 gali turėti stekinių adresų, ko neturi IBM System/370.
<b>Kokie buvo registrai abiejose architektūrose?</b>		
Ši architektūra turi 16 bendrosios paskirties registrus (32 bitų), 16 kontrolės registrų (32 bitų), 4 slankaus kabelio registrus (64 bitų) (Wikipedia, IBM System/370, 2024).	Ši architektūra turi 8 bendrosios paskirties registrus (16 bitų). Šie registrai gali būti naudojami laikinai duomenų saugyklai, adresavimui ar kaip akumuliatorius skaičiavimuose.	Abiejose architektūrose matomi registrų skirtumai tiek skaičiumi, tiek duomenų pločiu. IBM System/370 turėjo 3 tipų registrus ir gan didelės duomenų apimtys

Bendrosios paskirties registrai gali būti naudojami kaip bazės adreso registrai ir indeksų registrai atliekant adreso aritmetiką, taip pat kaip akumuliatoriai atliekant bendrąją aritmetiką ir logines operacijas. Šie registrai naudojami arba nurodyti, kad operacija gali būti vykdoma, arba pateikti specialią informaciją, kurios reikia tam tikrai funkcijai. Slankaus kablelio registrai skirti operacijoms su slankaus kablelio skaičiais. (IBM System/370 Principles of Operation, eight edition, 1981, 2-3, 2-4 p.)	Tačiau kai kurie registrai turi ir specifines funkcijas (PDP-11 Architecture Handbook, 1983, 28 p.). Vienas registras (dažniausiai R5) yra naudojamas kaip programos skaitiklis (PC). Kitas registras (dažniausiai R6) yra naudojama kaip steko rodyklė (SP) (Wikipedia, PDP-11, 2024). Vėlesniuose modeliuose dar buvo pridėti 6 slankaus kablelio registrai (64 bitų) (PDP-11 Architecture Handbook, 1983, 7 p.)	(nuo 32 bitų iki 64 bitų). Iš viso turi 36 registrus. Tuo tarpu PDP-11 turi tik 8 registrus, visi jie bendrosios paskirties. Taip pat ir duomenų apimtis mažesnė – 16 bitų. Bet vėliau buvo pridėti dar papildomi slankaus kablelio registrai. Apibendrinant galima manyti, jog IBM System/370 buvo labiau orientuotas į didesnių duomenų apdorojimą, sudėtingesnių skaičiavimų atlikimą negu PDP-11, kuris efektyviau dirbo su mažesniais duomenimis.
<b>Ar požymių bitai buvo naudojami šiose architektūrose? Kokie požymiai buvo naudojami?</b>		
Požymių bitai buvo naudojami, jie yra kanalų komandos žodyje (Channel Command Word, CCW). Naudojami duomenų susiejimo (CD), komandų susiejimo (CC), netinkamo ilgio indikacijos (SLI), perėjimo (SKIP), programos valdomo pertraukimo (PCI), netiesioginio adreso (IDA) požymiai (IBM System/370 Principles of Operation, eight edition, 1981, 12-34 p.)	Požymių bitai buvo naudojami slankaus kablelio statuso registre. Yra C (Carry flag), V (Overflow flag), Z (Zero flag), N (Negative flag) požymiai (PDP-11 Architecture Handbook, 1983, 129 p.). Pertraukimo procese naudojamas Done/Ready požymis (PDP-11 Architecture Handbook, 1983, 216 p.). Atminties valdymo registras 0 (MMR0) turi savo požymius (PDP-11 Architecture Handbook, 1983, 259 p.)	Abi architektūros naudoja požymių bitus. IBM System/370 požymiai labiau orientuoti į sistemų kontrolę. PDP-11 turi detalesnius požymius, skirtus aritmetikos ir loginėms operacijoms.
<b>Koks buvo kiekvienos architektūros duomenų plotis (mašininis žodis)?</b>		
Šios architektūros duomenų plotis yra 32 bitai (Wikipedia, IBM System/370, 2024)	Šios architektūros duomenų plotis yra 16 bitų (Wikipedia, PDP-11, 2024)	IBM System/370 architektūros mašininis žodis yra dvigubai didesnis nei PDP-11 architektūros.
<b>Koks buvo kiekvienos sistemos atminties išdėstymas?</b>		
Architektūros atmintis (adresų erdvė) buvo laikoma ištisine. Adreso plotis atmintyje buvo 24 bitų iki 16 MB atminties kiekio (IBM System/370 Principles of Operation, eight edition, 1981, 3-2 p.). Vėlesniuose modeliuose	Nors architektūros adresų erdvė atrodo ištisinė, iš tikrųjų adresų erdvė buvo išdėstyta puslapiuose. Adreso plotis skyrėsi skirtinguose modeliuose. Senesniuose modeliuose sukurti 16 bitų adresai iki 64 KB vietos, 18 bitų	IBM System/370 buvo pritaikyta dirbti su didesniais duomenimis. Palyginus su PDP-11 maksimalaus adresų pločiu, kuris siekė 4 MB, IBM System/370 maksimalus adresų plotis buvo net 4 kartus didesnis (16 MB).

adreso plotis buvo praplėstas iki 31 bitų (Wikipedia, IBM System/370, 2024)	adresai iki 256 KB ir vėlesniuose 22 bitų adresai iki 4 MB (PDP-11 Architecture Handbook, 1983, 223-226 p.)	
<b>Ar buvo palaikoma virtualioji atmintis ir kaip?</b>		
IBM System/370 naudojo virtualią atmintį (sistemą, kurioje pagrindinė atmintis atrodo didesnė nei faktiškai įdiegta). Nurodyti vietas virtualioje atmintyje yra naudojami virtualūs adresai. Virtualūs adresai dinaminio adresų vertimo mechanizmu yra verčiami į realius adresus, o po to į absoliučius, pasinaudojant segmentų ir puslapių lentelėmis. Vertimas vyksta adresų blokais, kurie vadinami puslapiais. (IBM System/370 Principles of Operation, eight edition, 1981, 3-10, 3-11 p.).	PDP-11 palaikė virtualią atmintį. Pasinaudojant atminties valdymo blokais (MMUs), atmintis dinamiškai laikoma puslapiuose (PDP-11 Architecture Handbook, 1983, 236 p.). Pasinaudojant aktyviais puslapių registrais (APRs) virtualūs puslapiai laikomi fizinėje atmintyje (PDP-11 Architecture Handbook, 1983, 225 p.).	Abejos architektūros palaiko virtualią atmintį ir ji realizuota naudojant puslapiavimą.
<b>Kokia buvo kiekvienos architektūros komandų sistema (ISA)? Pateikite 8–16 instrukcijų pavyzdžiai</b>		
Šioje architektūroje komandų sistema (ISA) buvo CISC (Wikipedia, IBM System/370, 2024). Iš viso skaičiuojama apie 149 mašinos komandas (IBM System/370 Principles of Operation, eight edition, 1981, 7-5, 7-6, 8-4, 9-5, 9-6 10-3 p.). Instrukcijų klasės: bendros instrukcijos, dešimtainės instrukcijos, slankaus kablelio instrukcijos, kontrolės instrukcijos (IBM System/370 Principles of Operation, eight edition, 1981, 7-1, 8-1, 9-1, 10-1 p.). Instrukcijų formatai buvo RR, RRE, RX, RS, SI, S, SSE, SS, kur yra dveji SS formatai (IBM System/370 Principles of Operation, eight edition, 1981, 5-3 p.). Komandų pavyzdžiai: AR (ADD), NR (AND), BAL (BRANCH AND LINK), BCTR (BRANCH ON COUNT), C	Šioje architektūroje komandų sistema (ISA) buvo CISC (Wikipedia, PDP-11 architecture, 2024). Iš viso skaičiuojama apie 68 mašinos komandas (PDP-11 Architecture Handbook, 1983, 73-79 p.). Instrukcijų klasės: Vieno operando, dvigubo operando, šakojimosi (branch), peršokimo (jump), spąstų (trap), Miscellaneous, Condition Code (PDP-11 Architecture Handbook, 1983, 73 p.). Palaikomi instrukcijų formatai buvo vieno operando, dviejų operandų, branch, JSR, RTS, Condition Code (PDP-11 Architecture Handbook, 1983, 73-79 p.). Komandų pavyzdžiai: ADD, SUB, MUL, XOR, BR (branch), BMI (Branch if minus), BHI (branch if higher), TRAP, WAIT (wait for interrupt), RESET (reset unibus)	Abi architektūros yra tos pačios komandų sistemos CISC. IBM System/370 turi daugiau komandų nei PDP-11. Klasės taip pat skiriasi, kur PDP-11 mažesnės, labiau specifinės. Tai dar karta rodo, kad IBM System/370 buvo labiau skirta dirbti su didesniais ir įvairesniais duomenimis. Abejose architektūrose yra panašios loginės komandos tik skiriasi jų pavadinimai (Pvz.: AR ir ADD). Vis dėl to skyrėsi ne tik komandų pavadinimas, bet ir jų funkcionalumas. Pavyzdžiui, PDP-11 branch komandos yra specifinės pagal skaičių (šokti esant neigiamam ar nuliui), o IBM System/370 branch komandos plačios, neapsiriboja tik peršokimu pagal skaičių bet gali ir

(COMPARE), LR (LOAD), AP (ADD DECIMAL), DP (DIVIDE DECIMAL), LD (LOAD (long)), MVCP (MOVE TO PRIMARY) (IBM System/370 Principles of Operation, eight edition, 1981, 7-5, 7-6, 8-4, 9-5, 9-6 10-3 p.).	(PDP-11 Architecture Handbook, 1983, 73-79 p.).	susieti (link).
<b>Kokius adresavimo būdus palaikė kiekviena architektūra?</b>		
Sistema palaikė realiojo adresavimo (real mode), pagrindinio virtualiojo adresavimo (primary virtual mode), antrinio virtualiojo adresavimo (secondary virtual mode), Branch adresavimo, įvesties/išvesties įrenginių adresavimo būdus (IBM System/370 Principles of Operation, eight edition, 1981, 3-5, 3-11, 12-8 p.)	Sistema palaikė pagrindinius adresavimo būdus (registrų, automatinio inkrementavimo, automatinio dekrementavimo, indeksų), netiesioginius adresavimo būdus (su atidėtais adresavimais), PC registro adresavimo būdus (tiesioginio, absoliutaus, santykinio (relative)) (PDP-11 Architecture Handbook, 1983, 66-68 p.)	Architektūrų adresavimo būdai pagrinde skiriasi. Gal kažkiek panašumu tarpusavy turi tik IBM System/370 realiojo adresavimo ir PDP-11 pagrindinio adresavimo būdai.
<b>Kokios buvo kiekvienos architektūros I/O galimybės?</b>		
Ši sistema naudojo I/O kanalus, kurie tvarkė I/O operacija atlaisvindami tai nuo procesoriaus. Valdymui naudojamos I/O instrukcijos. Taip pat ir I/O adresavimas. Naudoti kanalų komandų žodžiai (CCW), kurie buvo dekoduojami kanalų ir I/O įrenginių. Taip pat buvo galimos ir pertrauktys. (IBM System/370 Principles of Operation, eight edition, 1981, 12-3, 12-6, 12-8, 12-44, 12-50 p.)	Ši sistema naudojo atminties žemėlapių I/O. I/O įrenginiai yra traktuojami kaip atminties lokacijos. Taip pat naudojami I/O registrai, instrukcijos. (PDP-11 Architecture Handbook, 1983, 223-245 p.)	Abi architektūros turėjo tam tikras I/O galimybes. PDP-11 I/O buvo kiek paprastesnis, pritaikytas mažiems kompiuteriams. IBM System/370 I/O palyginus sudėtingesnis.
<b>Ar buvo palaikomi pertraukimai?</b>		
Architektūra palaiko pertraukimus. Yra 6 pertraukimų klasės: išoriniai, I/O, mašinos patikros, programinius, restartavimo, supervizoriaus iškvietimo (IBM System/370 Principles of Operation, eight edition, 1981, 6-5 p.).	Architektūra palaiko pertraukimus. Pagrindie pertraukimai vyksta programoje, kai procesorius tikrina Done/Ready bitą (PDP-11 Architecture Handbook, 1983, 215-216 p.)	Abi architektūros turi pertraukimus, tik IBM System/370 architektūroje jų yra daugiau ir įvairiau, dėl ko sistema labiau galėjo „gaudyti“ klaidas ir būti efektyvesnė.
<b>Kokius duomenų tipus palaikė kiekviena architektūra aparatūros lygyje?</b>		
Architektūra palaikė fiksuoto kablelio aritmetiką – tam buvo skirtos ir instrukcijos. Taip pat	Architektūra palaikė sveikojo skaičiaus, slankaus kablelio ir string duomenų tipus. String	Abi architektūros turi slankaus kablelio duomenų tipą. Tačiau jos turi ir kitų

buvo palaikoma ir slankaus kablelio aritmetiką – tam buvo ne tik instrukcijos, bet ir atskiras formatas. Taip pat pritaikytas ir dešimtainis formatas. Sveikieji skaičiai buvo koduojami binariškai (IBM System/370 Principles of Operation, eight edition, 1981, 2-2, 2-3, A-2 p.).	duomenų tipas dar skirstomas į character ir dešimtainio duomenų tipus. Sveikieji skaičiai yra koduojami kaip dvejetainio papildyti (PDP-11 Architecture Handbook, 1983, 27-28, 35 p.)	tipų. IBM System/370 duomenų tipai pritaikyti dirbti su dideliais duomenimis. PDP-11 duomenų tipai labiau pritaikyti paprastoms operacijoms.
<b>Kokia buvo kiekvienos sistemos greitis?</b>		
155 modelio pagrindinio mašinos ciklo laikas yra 115 nanosekundžių. Pagrindinės atminties ciklo laikas yra 2,07 mikrosekundės. Tačiau didelės spartos buferinė atmintis veikia taip, kad efektyvus sistemos atminties ciklo laikas yra nuo vieno trečdalo iki vieno ketvirčio tikrojo pagrindinės atminties ciklo laiko (IBM System/370 Model 155 Functional Characteristics, 1971, 5 p.)	Procesorius galėjo veikti iki 3 MB/s. (PDP-11 Architecture Handbook, 1983, 23 p.)	Nepakankamai informacijos dokumentacijoje, kad būtų galima palyginti.
<b>Ar architektūros naudojo spartinančią atmintį? Jei taip, kokio dydžio?</b>		
Ši architektūra turėjo spartinančią atmintį (cache), dokumentacijoje dar vadinama didelio greičio buferio saugykla. Dydis nėra nurodytas. (IBM System/370 Principles of Operation, eight edition, 1981, 1-2 p.).	Ši architektūra turėjo spartinančią atmintį (cache), 8 KB (PDP-11 Architecture Handbook, 1983, 21 p.)	Abi architektūros turi spartinančią atmintį, galima teigti, jog abi buvo pritaikytos dirbti didesniais greičiais.
<b>Kokios buvo tipinės kiekvienos architektūros taikymo sritys?</b>		
Ši architektūra buvo kurta siekiant patenkinti 1970-ųjų duomenų apdorojimo poreikius. Bankai, oro linijos ir mažmeniniai ypač vertino šią architektūrą nuotoliniam skaičiavimui, kai valdymo informacijos tinklai laukuose naudojami centralizuotu kompiuteriu. Aeroquip Corporation, kurios kosminių laivų detalės buvo naudojamos „Gemini“ ir „Apollo“ misijų manevrinams varikliams, naudojo	PDP-11/20 sukūrė proveržio komercinio minikompiuterio archetipą, pasižymėjusį išskirtiniu inžineriniu sprendimu, orientuotu į interaktyvų naudojimą. Sistemos prieinama konstrukcija, lankstus augimas ir tvirtas atsparumas leido ją naudoti įvairiose pramonės šakose: gamyba, komunalinės paslaugos, kariniai tikslai, ligoninės, mokyklos ir laboratorijos telekomunikacijos ir kt. Kai kuriuose JAV	Abi šios architektūros turėjo plačius pritaikymus ir darė didelę įtaką tolesnei technologijų raidai.

System/370 kasdieniams daugiau nei 71 000 dalių užsakymų stebėjimui ir valdymui. Sistema leido vadovybei greitai gauti informaciją apie klientų užsakymus ir gamybos efektyvumą (IBM, The IBM System/370, 2024).	kariniuose jūrų laivyno laivuose, kaip teigiama, vis dar naudojamos PDP-11/34 sistemos, kurios valdo svarbias radarų sekimo ir ginklų sąsajas, per jautrias, kad būtų galima atnaujinti į tinklinius sprendimus. (History tools, The PDP-11 Minicomputer: An Enduring Legacy of Digital Innovation, 2024)	
<b>Kiek programinės įrangos buvo parašyta kiekvienai aptariamai architektūrai, ar ji (vis dar) prieinama, kur ji buvo naudojama?</b>		
GNU Compiler Collection (GCC) buvo pritaikyta šiais architektūrai (Wikipedia, IBM System/370, 2024). Taip pat Hercules emuliatorius (The Hercules System/370, ESA/390, and z/Architecture Emulator, 2024). IBM COBOL buvo programavimo kalba, kuri veikė šioje architektūroje. Ji taip pat turėjo ir savo kompailerius (Wikipedia, IBM COBOL, 2024)	Asembleris MACRO-11 buvo parašytas šiai architektūrai (Computer History Wiki, MACRO-11, 2024). Iki šiol yra sukurta C kalbos kompaileris būtent šiai architektūrai (GitHub, Pavel Krivanek, legacy-cc, 2018)	Abi architektūros turėjo įvairių programinių įrangų, tik šiandien jie yra sunkiais prieinami ir randami.
<b>Ar yra emuliatorių abiem architektūroms?</b>		
Yra Hercules System/370, ESA/390, and z/Architecture Emulator ( <a href="http://www.hercules-390.org/">http://www.hercules-390.org/</a> )	Yra SIMH emuliatorius, kuris palaiko keletą architektūrų, įskaitant ir PDP-11 ( <a href="https://simh.trailing-edge.com/">https://simh.trailing-edge.com/</a> )	

### Šaltiniai:

Wikipedia, IBM System/370, 2024:

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

IBM System/370 Principles of Operation, eight edition, 1981:

[https://bitsavers.trailing-edge.com/pdf/ibm/370/princOps/GA22-7000-7\\_IBM\\_System\\_370\\_Principles\\_of\\_Operation\\_8th\\_ed\\_198103.pdf](https://bitsavers.trailing-edge.com/pdf/ibm/370/princOps/GA22-7000-7_IBM_System_370_Principles_of_Operation_8th_ed_198103.pdf)

Wikipedia, PDP-11, 2024:

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

Wikipedia, PDP-11 architecture, 2024:

[https://en.wikipedia.org/wiki/PDP-11\\_architecture](https://en.wikipedia.org/wiki/PDP-11_architecture)

PDP-11 Architecture Handbook, 1983:

[http://www.bitsavers.org/pdf/dec/pdp11/handbooks/EB-23657-18\\_PDP-11\\_Architecture\\_Handbook\\_1983.pdf](http://www.bitsavers.org/pdf/dec/pdp11/handbooks/EB-23657-18_PDP-11_Architecture_Handbook_1983.pdf)

IBM System/370 Model 155 Functional Characteristics, 1971:

[http://www.bitsavers.org/pdf/ibm/370/funcChar/GA22-6942-1\\_370-155\\_funcChar\\_Jan71.pdf](http://www.bitsavers.org/pdf/ibm/370/funcChar/GA22-6942-1_370-155_funcChar_Jan71.pdf)

IBM, The IBM System/370, 2024:

<https://www.ibm.com/history/system-370>

The Hercules System/370, ESA/390, and z/Architecture Emulator, 2024:

<https://sdl-hercules-390.github.io/html/>

Wikipedia, IBM COBOL, 2024:

[https://en.wikipedia.org/wiki/IBM\\_COBOL](https://en.wikipedia.org/wiki/IBM_COBOL)

History tools, The PDP-11 Minicomputer: An Enduring Legacy of Digital Innovation, 2024:

<https://www.historytools.org/products/dec-pdp-11-computer>

Computer History Wiki, MACRO-11, 2024:

<https://gunkies.org/wiki/MACRO-11>

GitHub, Pavel Krivanek, legacy-cc, 2018:

<https://github.com/pavel-krivanek/legacy-cc>