

DST203 3.c lekcija

Klasiskā skaitļotāja galvenie mezgli

Neatkarīgi no paaudzes katrā datorā var izdalīt šādas galvenās sastāvdaļas:

- -aritmētiski loģisko mezglu;
- -vadības mezglu;
- -atmiņu;
- -ievada -izvada iekārtas.

Aritmētiski loģiskais mezgls ir nesaraucjami saistīts ar vadības mezglu, tāpēc šodien tos apvieno, pievieno vēl dažus mezglus un nosauc par procesoru.

1. Aritmētiski -loģiskais mezgls aparātūras vienkāršošanas nolūkā satur tikai summatoru, kurš izpilda tikai saskaitīšanas darbības, jo atskaitīšana ir tā pati saskaitīšana apgrieztajos vai papildus kodos. Reizināšana, dalīšana un citas algebriskās darbības realizē ar programmu palīdzību kā daudzkārtējas saskaitīšanas darbības, jo arī aritmētikā $3 \times 4 = 4 + 4 + 4$. Obligāta šī mezgla sastāvdaļa ir reģistri, kuri noder īslaicīgai saskaitāmo un rezultātu uzglabāšanai. Starp šiem reģistriem vienadreses sistēmā īpaša vieta ir akumulatoram- speciālam reģistram, kuru izmanto viena saskaitāmā un pēc tam iegūtā rezultāta uzglabāšanai. Katru aritmētisko komandu bez tam pavada virkne rezultāta loģisko novērtējumu, kuri kā atbilde 1 vai 0 tiek ievietoti īpašā karodziņu reģistrā.
2. Vadības mezgls nodrošina daudzo vadības signālu izstrādāšanu, kuri nepieciešami visu datora bloku saskaņotai darbībai laikā. Skaitļotāja darbības izpratnē īpaša nozīme ir:
 - -komandu skaitītājam, kurš norāda pašreiz izpildāmās komandas numuru jeb adresi, no kuras šī komanda tikusi nolasīta;
 - -komandas reģistram, kuru izmanto no atmiņas nolasītā komandas koda uzglabāšanai uz komandas izpildes laiku, t.i., tās atmiņas šūnas saturs, kuras adresi norāda komandu skaitītājs;
 - - karodziņu reģistram, kurš palīdz vadīt turpmāko programmas izpildīšanas virzienu. Visbiežāk lieto nulles karodziņu, kurš iestādās tad, ja darbības rezultāts ir mašīnas nulle, un pārnesuma karodziņu, kurš summēšanas un bīdes procedūrās norāda uz pārnesuma parādīšanos ārpus darba reģistra.
3. Atmiņa paredzēta skaitļu un komandu glabāšanai. Atmiņa sadalīta šūnās, un katra atmiņas šūna glabā vienu baitu. Katrai atmiņas šūnai ir sava adrese, kura kā vadības signāls tiek padota atmiņas adresei ieejās un vada elektronisko komutatoru tā, ka atmiņas datu ieejas tiek pievienotas tikai šai šūnai. Īpašs vadības signāls nosaka, vai dati no šīs šūnas tiks nolasīti vai arī tajā ierakstīti.
4. Datora sakarus ar ārējo vidi realizē ievada un izvada iekārtas. Komandas adrese daļa norāda tās iekārtas numuru jeb adresi, kurai dotajā brīdī jāpiedalās ievada vai izvada operācijā, bet vadības mezgla signāli nosaka ievada un izvada iekārtu darbības laika diagrammas.

Īpašu ievada -izvada iekārtu veido operātoru pults, kura nodrošina cilvēka dialogu ar datoru. Pirmajām datoru paaudzēm tā izpildīta kā speciāla pults, bet, sākot ar trešo paaudzi, par operātoru pulti noder kāda no ievada -izvada iekārtām kopā ar speciālu vadības programmu -operacionālo sistēmu vai monitoru. Pēdējās uzdevums -nodrošināt cilvēka doto komandu nodošanu datora aparātūrai izpildīšanai.

Vienkāršota datora struktūra parādīta 1.3. zīmējumā, bet 3.1 zīmējumā parādīta vienkārša ar maģistrāles principu veidota datora struktūra, kurā parādīti nepieciešamie vadības signāli, kas vajadzīgi šāda datora darbībai.

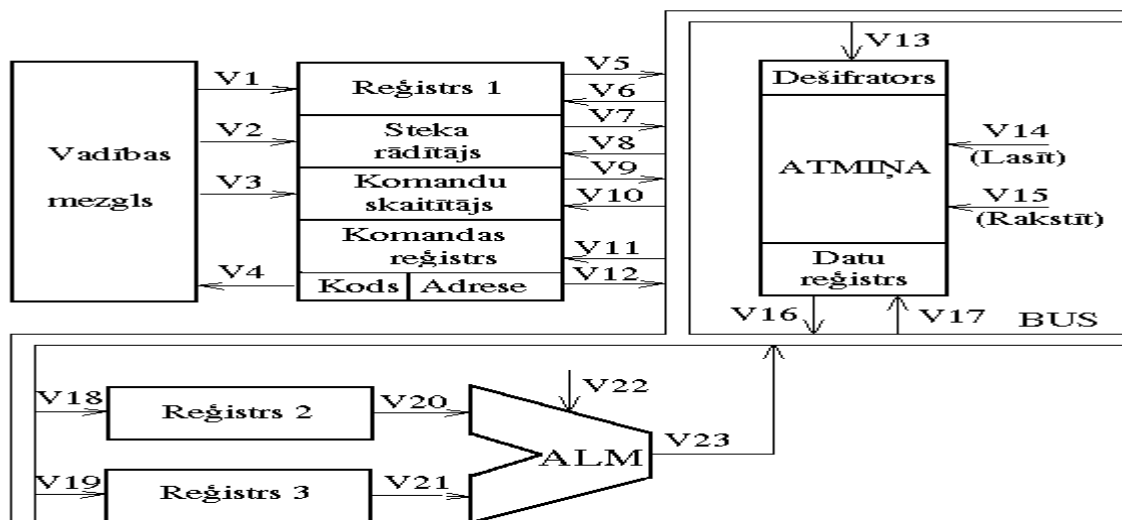
Programmas izpildīšana

Datora darbību uzdevuma izpildīšanas laikā nosaka programmētāja sastādītā darba programma, kura pirms izpildīšanas jāievieto datora atmiņā. Šim nolūkam speciālā vadības bloka reģistrā -komandu skaitītājā- ievieto tās atmiņas šūnas numuru, ar kuru jāsāk ievadīt programmu, un palaiž ievada iekārtu. Pēc katra informācijas vārda nolasīšanas un ievietošanas atmiņā reģistra saturam automātiski pieskaitās vieninieks, nodrošinot nākamā vārda ievietošanu nākamajā atmiņas šūnā.

Lai programmu izpildītu, komandu skaitītājā ievieto tās atmiņas šūnas numuru, ar kuru sākas programma, un palaiž mašīnu darba režīmā. Tagad mašīna sāk izpildīt programmētāja sastādīto programmu, sākot ar tās pirmo komandu.

Katras mašīnas komandas izpildīšana rada laikā sadalītu sadarbību starp attēlā parādītajām mašīnas sastāvdaļām. Līdz ar to katras komandas izpildīšanas laiku var sadalīt sīkākos cits citam sekojošos laika posmos, kuros notiek:

- -komandas nolasīšana no atmiņas un ievietošana komandas reģistrā;
- -komandas izpildīšanai nepieciešamo vadības signālu veidošana;
- -operandu nolasīšana no atmiņas un ievietošana ALM darba reģistros;
- -komandā prasītās darbības izpildīšana;
- -pāreja uz nākamās komandas izpildīšanu.



3.1. zīm. Klasiska vienkārša datora galvenie mezgli un vadības signāli

Katrā komandas izpildīšanas posmā noteiktā secībā nepieciešami vairāki vadības signāli, kuru izstrādāšanu veic vadības bloks. Notiekošo procesu sinhronizēšanai katru šo posmu sadala vēl sīkākās daļās -taktīs, katrā no tām izstrādājot noteiktus vadības signālus. Komandas izpildīšanas posmi un tajos nepieciešamie vadības signāli vienkāršotā klasiskā vienadresu datorā ar maģistrālo struktūru parādīti 3.2 zīmējumā.

1. Komandas izpildīšanas pirmajā posmā komandu skaitītāja saturs tiek padots uz atmiņu kā adrese, kurā glabājas izpildāmā komanda. Atmiņā pēc norādītās adreses elektronisks slēdzis pieslēdz vajadzīgo šūnu atmiņas datu izejām. Vadības bloks no šīs šūnas nolasīto skaitli vada tā, ka tas nonāk vadības bloka komandu reģistrā.
2. Nākošajā laika posmā, veidojot vadības signālus, skaitlis komandu reģistrā tiek sadalīts divās daļās, no kurām katra daļa tiek apstrādāta atsevišķi. Darbības jeb operācijas kods noskaņo vadības un aritmētiski -loģisko mezglus un tālāk nosaka visu vadības signālu secību dotās darbības izpildīšanai, piemēram, saskaitīšanai, datu nosūtīšanai no akumulatora uz atmiņu utt.
3. Komandas adrešu daļa tiek padota uz atmiņu vai ievada -izvada iekārtām kā adrese un nosaka tās atmiņas šūnas vai iekārtas numuru, kura turpmāk piedalīsies dotās darbības izpildīšanā. Vadības signāli vienlaikus nosaka arī to, vai informācija no šīs adreses tiks nolasīta vai arī tajā ierakstīta. Ja komandā prasītās darbības izpildīšanai nepieciešami dati, tie no uzrādītās atmiņas šūnas vai iekārtas tiek nolasīti un nosūtīti uz aritmētiskā mezgla darba reģistru. Vienadresu datora komandās, kas nosaka aritmētiskās darbības, ir pieņemts, ka šajā laika posmā otrs skaitlis jeb operands atrodas speciālā reģistrā -akumulatorā.
4. Darbības izpildīšanas posmā aritmētiski loģiskais mezgls saņem no vadības mezgla nepieciešamos vadības signālus un izpilda darbības kodā prasīto darbību. Tā, piemēram, summators var saskaitīt akumulatora un komandas adrešu daļā norādītās atmiņas šūnas saturu, rezultātu ievietojot atpakaļ

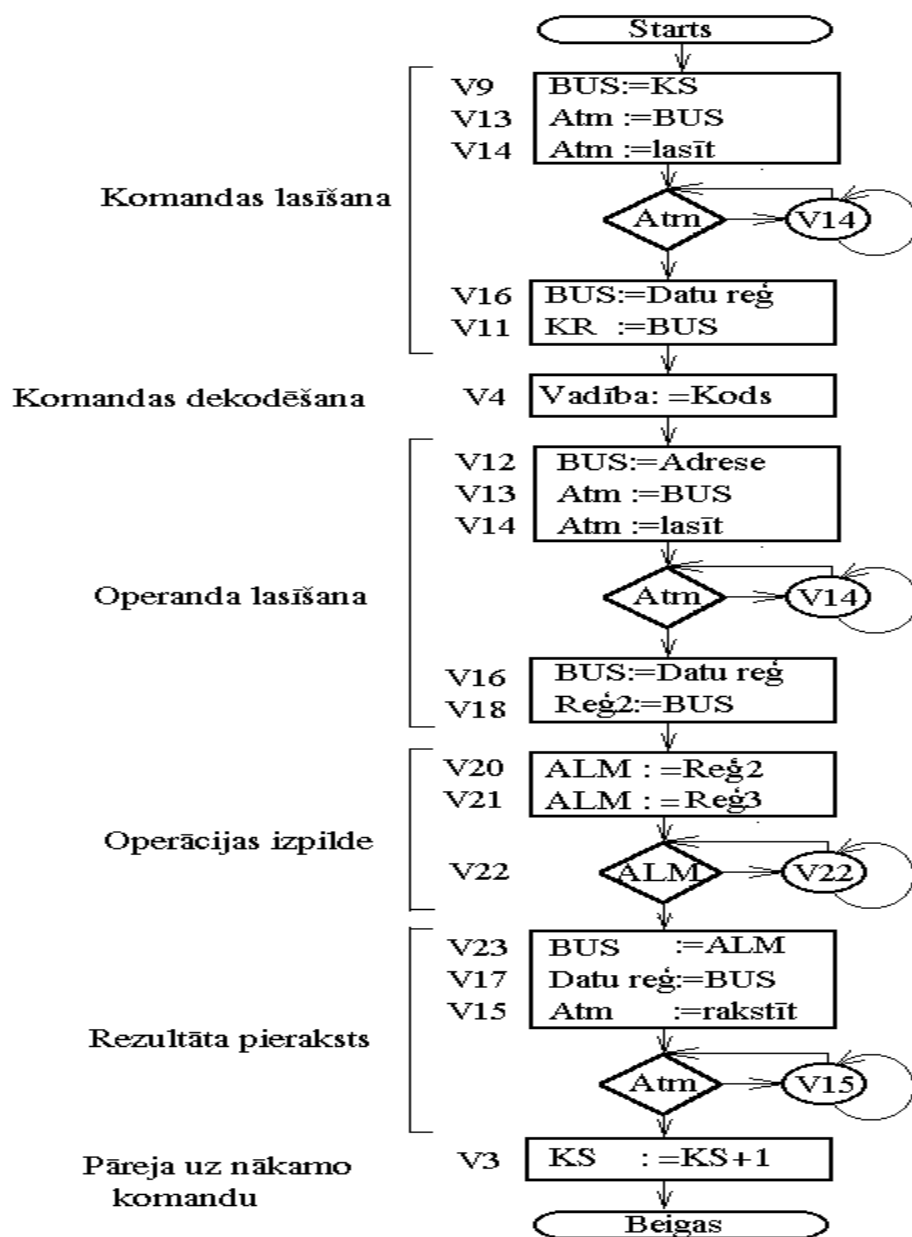
akumulatorā. Pēc aritmētiskās darbības izpildīšanas vienmēr notiek rezultāta loģiska novērtēšana, un šis novērtējums kā 1 vai 0 tiek ierakstīts karodziņu reģistra attiecīgajā bitā. To var izmantot programmas automātiskai vadīšanai.

5. Praktiski vienlaikus ar komandas izpildīšanu komandu skaitītājā pieskaitās vieninieks, un pēc darbības pabeigšanas aritmētiski loģiskajā mezglā var sākties nākamās komandas izpildīšana. Tāpat kā iepriekšējā komanda, tā sākas ar komandu skaitītāja satura nosūtīšanu uz atmiņas adresu ieejām un norādītās atmiņas šūnas satura ievietošanu vadības bloka komandu reģistrā.

Vadības nodošanas komandas

Vadības nodošanas komandas veido īpašu komandu grupu, kuras atkarībā no iegūtā rezultāta ļauj automātiski vadīt skaitļošanas procesu un izpildīt programmu zarošanos. To ir trīs veidi:

- -nenosacītā vadības nodošana;
- -nosacītā vadības nodošana;
- -apakšprogrammu izsaukšana.



3.2. zīm. Vienkāršota komandas izpildes secība

Nenosacītā vadības nodošana prasa izpildīt programmu tālāk citā vietā, neprasot to turpināt no nākamās komandas. Šajā gadījumā programmā jāizdara lēciens, un komandā ir uzdota adrese, no kuras programma jāturpina. Tātad komandas darbības izpildīšanas posmā uz komandu skaitītāju jānosūta komandas adrešu daļa, un programma jāturpina no šīs jaunās adreses.

Nosacīto vadības nodošanu vada iepriekšējā komandā izstrādātais karodziņš un atkarībā no tā vērtības iespējami divi programmas turpināšanas ceļi:

- -programma jāturpina ar nākamo komandu, kuras adresi norāda komandu skaitītājs pēc vieninieka pieskaitīšanas, un vadības nodošanas komandas adrešu daļa netiek izmantota;
- -programmā jāizdara lēciens un uz komandu skaitītāju jānosūta komandas adrešu daļa, un programma jāturpina no šīs jaunās adreses.

Apakšprogrammu izsaukšana ir īpaša darbība, kura pēc apakšprogrammas izpildīšanas prasa turpināt doto programmu ar nākamo komandu. Šajā gadījumā komandu skaitītāja saturu, ti., nākamās komandas

adresi nosūta uz steku uzglabāšanai un atgriešanai komandu skaitītājā pēc apakšprogrammas izpildīšanas, bet komandu skaitītājā tiek ielādēta komandas adrešu daļa, kura norāda jaunās darba programmas sākumu.

Komandu izpildīšana turpinās tikmēr, kamēr vadības mezgls saņem komandu apstāties.