

**O`ZBEKISTON RESPUBLIKASI AXBOROT
TEKNOLOGIYALARI VA KOMMUNIKATSIYALARINI
RIVOJLANTIRISH VAZIRLIGI
MUHAMMAD AL-XORAZMIY NOMIDAGI
TOSHKENT AXBOROT TEKNOLOGIYALARI UNIVERSITETI
NUKUS FILIALI**

KOMPYUTER INJINIRINGI FAKULTETI

Telekommunikatsiya texnologiyalari yo`nalishi 3-kurs talabasi
Ganjayev Safarboyning
Mikropratsessorlar fanidan



Mustaqil ishi

Mavzu: Mikroprotsessor registrlari va xotira segmentlari .

**Bajardi:
Qabul qildi:**

**Ganjayev S.
Serjanov Q.**

Nukus – 2017

Mavzu: Mikroprotsessor registrlari va xotira segmentlari .

Reja:

1. Mikroprotsessor registrlari va xotira segmentlari.
2. Adreslash usullari.
3. Mikroprotsessor buyruqlar tizimi.
4. Xulosa.
5. Foydalanilgan adabiyotlar.

2.5.Mikroprotsessor registrlari va xotira segmentlari

MP tizimidagi xotira maydoni odatda bir qancha asosiy maxsus funksiyalar uchun ajratilgan. Boshlang'ich yuklanish dasturi doimiy xotira qurilmasi yoki flesh-xotirada yozilgan bo'ladi. Protssessor manbaaga ulanishi bilan aynan shu boshlang'ich yuklanish dasturi orqali o'z ishini boshlaydi va RESET tugmasini bosish bilan o'z ishini tugatadi. Ma'lumki, qurilmaning boshlang'ich yuklanishini mikroprotsessor xotiraning nolinchi manzilidan boshlab amalga oshiradi. Kiritish va chiqarish qurilmasini uchta maxsus guruhga bolish mumkin:

- foydalanuvchi qurilmasi interfeysi (foydalanuvchi axborotni kiritishi va axborotni foydalanuvchiga chiqansh);
- axborotni uzoq saqlash uchun kiritish \ chiqarish qurilmasi;
- taymer qurilmasi.

Mikroprotsessor ichki xotiradagi kiritish \ chiqarish qurilmasiga yoki shinali qurilma tizimiga o'zining tizim xotirasiga murojaat qilish kabi imkoniyati mavjud.

Manzillar xotirasini va kiritish \ chiqarish qurilmasini ajratishda odatda ikki asosiy yo'nalish mavjud:

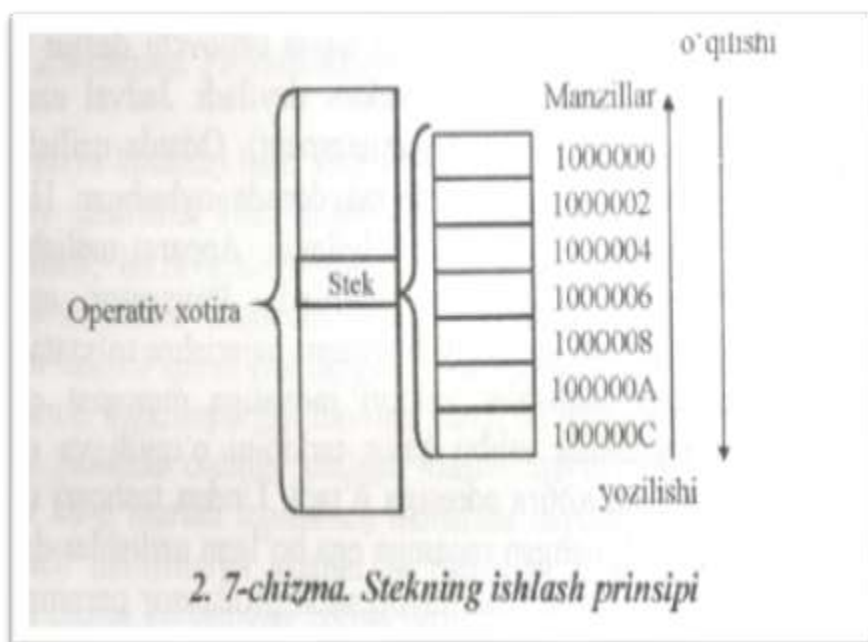
- kiritish \ chiqarish qurilmasi uchun umumiy manzillar joyini belgilash;
- xotira manzillar joyini va kiritish \ chiqarish qurilmasi to'liq ajratish;

Bundan birinchi holat shunisi bilan yaxshiki, protsessor kiritish \ chiqarish qurilmasiga murojaat qilganida xotiraga murojaat qilgan buyruqlar orqali boglanishi mumkin. Lekin manzilli xotira joyi kiritish \ chiqarish qurilmasi xotirasidagi joy bilan to'g'ri kehshi lozim. Masalan, 16 - razryadli shina manzili 64K adres bo'lishi mumkin.Undan 56K adreslar manzillar o'rniga mos keladi, 8K esa katta manzillar - kiritish \ chiqarish qurilryasi osti joyiga tegishli bo'ladi. Ikkinchi usulning farqi shundaki, bunda xotira mikroprotsessor tizimida manzilli joyning barchasini egallaydi. Kiritish \ chiqarish qurilmasi bilan boglanish uchun shina boshqaruvining magistralida maxsus buyruqlar va axborot almashinishning maxsus stroblari orqali bog'laniladi. Personal kompyuterlardabu usul aynan shunday amalga oshiriladi. Ammo bu usul xotira hisobida olinganda kiritish \ chiqarish qurilmasi bilan lmkoniyatlari chegaralangan. Fon Neyman arxitekturasida xotira buyruqlar va ma'lumotlar uchun umumiy bo'ladi. Garvard arxitekturasida esa, buyruqlar xotirasi CSEG(Code Segment) va ma'lumotlar xotirasi DSEG(Data Segment) alohida ajratilgan bo'lib, ularda alohida manzillar joyi va unga kirish holati mavjud.

Barcha zamonaviy MP larda alohida «stek» maydoni ajratilgan bo'lib, ular protsedura parametrlarini uzatishda va adreslarni saqlab ularga qaytishda ishlatiladi. Stek MP maxsus ichki maydonida yoki operativ xotirada joylashishi mumkin. Keltinlgan holatda DSEG adres maydonimng qismida joylashgan.Stek LIFO(Last in - first out) rejimida ma'lumotlarni vaqtinchalik saqlash uchun foydalaniladi. Stekning afzalligi, belgilangan va o'zgartirilmaydigan adreslash usuli hisoblanadi.

Stekdan ma'lumotlarni o'qishda stek ko'rsatkichidagi adreslar orqali o'qiladi va stek ko'rsatkichi inkrementlanadi. Natijada LIFO usulidagi xotira tashkillashtiriladi, ya'ni birinchi keldi oxirgi ketdi (2.7.-chizma).

Masalan, stek ko'rsatkichining jony holati 1000008, unga ikkita so'z yozish kerak. Birinchi adres 1000006(yozishdan oldin stek ko'rsatkichi ikkita kamayadi), ikkinchi adres 1000004. Yozish natijasida stek ko'rsatkichi-1000004 ga teng bo'ladi. Agar stekdan ikkita so'zni o'qish kerak boisa, birinchi bo'lib 1000004 adresdan so'z o'qiladi, keyin stek ko'rsatkichi 1000006 ga teng bo'ladi. Ikkinchi bo'lib 1000006 adresdan so'z o'qiladi, stek ko'rsatkichi 1000008 ga teng bo'ladi. Hammasi o'z holatiga qaytdi. Birinchi yozilgan so'z ikkinchi bo'lib, ikkinchi so'z birinchi bo'ladi.



Ushbu keltirilgan adreslash dasturi bir necha marta joylash- tirishda qulaydir. Masalan asosiy dastur bajarilish jarayonida 1 chi qism dastur chaqiriladi. Agar asosiy dastur holatini saqlash kerak bo'lsa stekda saqlaymiz va qism dastur bajarilgandan so'ng stekdan chaqirib olishimiz mumkin.

Stek tizimidagi ma'lum vaqt ichida ko'p pog'onali uzilishlarni tashkillashtirishda foydalanadi. Akkumulyator vazifasini stek cho'qqisiga uzatuvchi stek arxitekturasini mavjud. Stekli tashkillashtirish qisqa uzunlikka ega bo'lgan adressiz buyruqlardan foydalanish imkoniyatini yaratadi. Adressiz buyruqlar stek cho'qqisi va undan keyingi joylashgan ma'lumotlar ustida amallar bajaradi. Operatsiyalar bajarilish jarayonida joriy operandlar stekdan chaqirilib, natija stek cho'qqisiga uzatiladi. Stekli arxitektura yuqori hisoblash effektivligiga ega. Yuqori pog'onali til FORT mavjud bo'lib, u adressiz buyruqlar asosida yaratilgan. Bunday arxitektura yuqori o'tkazuvchanlikka ega bo'lgan protsessorlarda, xususan RISC protsessorlarida qo'llaniladi. Keyingi maxsus xotira maydoni DSEG- bu uzilish vektorlari jadvalidir.

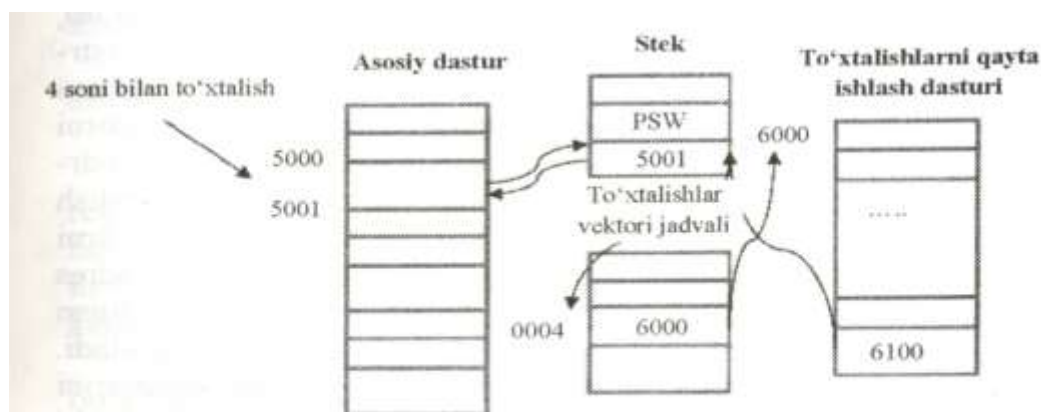
Uzilishlar - bu faqat tashqi qurilmalardan so'rovlarni tashkil qilish emas, balki protsessor ketma-ketlikdagi ishining buzilishidir. Masalan arifmetik operatsiyalarning

noto'g'ri bajarilishi natijasida (nolga bo'lish) uzilishlar bo'lishi, yoki uzilishlar dasturiy bo'lishi masalan qism dasturdan chiqish va qaytishdagi uzilishlar bo'lishi mumkin. Har qanday uzilishlar vektor uzilishlar jadvali asosida qayta ishlanadi. Ushbu jadvalda uzilishlarni qayta ishlovchi dastur boshlang'ich adresi joylashgan bo'lib, vektor deyiladi. Jadval uzunligi katta bo'lishi mumkin (bir necha yuz element). Odatda uzilishlarni tashkil qilish vektori xotiraning bosh maydonida joylashgan. Har bir vektoring adresi uzilish raqamini belgilaydi. Apparat uzilishlarda uzilish raqami qurilmalar orqali beriladi. Protsessor apparat uzilishlarni olgandan so'ng joriy buyruqni bajarishni to'xtatadi va xotira maydonidagi uzilishlar vektori jadvaliga murojaat qiladi. Protsessor o'z navbatida ushbu qator tarkibini o'qiydi va ushbu vektor orqali berilgan xotira adresiga o'tadi. Undan tashqari ushbu adresdan boshlab belgilangan raqamga ega bo'lgan uzilishlar dasturi joylashadi. Ushbu jarayon bajarilayotganda protsessor parametrlari stekda saqlanadi.

Masalan, protsessor (2.8,-chizma) xotira adresi 5000 da joylashgan buyruq va asosiy dasturni bajarayapti. Shu vaqtda vektor adresi 4 ga teng uzilishlar so'rovini oladi.

Protsessor 5000 adresdan buyruq bajarishni tugallaydi. Keyin (5001) buyruqni hisoblagichi jony qiymatini stekda va PSW «registr so'z holati» ni saqlaydi. Undan keyin protsessor 4 adresdan uzilish vektor kodini o'qiydi. Bu kod 6000 ga teng bo'lsin.

Protsessor 6000 xotira adresiga o'tadi va shu adresdan boshlab uzilishlarni qayta ishlashga o'tadi. Dasturi 6100 adresda tugallanadi deb hisoblaymiz. Ushbu adresga kelib protsessor uzilgan dastur joyiga qaytadi. Buning uzilishlar tashkil qilingan vaqtdagi stekdan 5001 adres qiymatini oladi, keyin protsessor 5001 adresdan buyruqni o'qib asosiy dasturni bajarishga o'tadi.



2.8-chizma. To'xtalishlarni qayta ishlashning algoritmi

Avariya holatlari ham xuddi shunday uslubda amalga oshiriladi. Dasturiy uzilishlar ham uzilishlar vektor jadvali asosida xizmat ko'rsatiladi, lekin uzilish raqami buyruqlar tarkibida ko'rsatiladi. Bunday uzilishlar dasturchi uchun xotiraning istalgan joyida uzilishlarni tashkil qilish imkoniyatini yaratadi. Uzilishlarni qayta ishlash dasturi bajarilayotgan davrda yangi so'rovlar bolishi mumkin. Bunday holatda oldingi uzilgan dastur qayta ishlanadi. Bunday holatlar ko'p martali ichma-ich uzilishlar deyiladi. Stek mexanizmi ichma-ich uzilishlarga xizmat ko'rsatishga imkon yaratadi, ya'ni stekdan oxirgi joylashgan kodga birinchi bo'lib xizmat ko'rsatiladi. Murakkab holatlarda uzilishlar vektor jadvalida uzilishlar qayta ishlash dastur bosh adresi emas, balki uzilishlar deskriptori joylashadi. Lekin deskriptor qayta ishlash natijasi uzilishlar qayta ishlash dastur bosh adresi boladi. CSEG va DSEG dan tashqari hamma zamonaviy mikroprotsessorlar maxsus ajratilgan kichik hajmli ma'lumot bo'shlig'iga ega, ular dasturli kirish to'plamiga ega RSEG (Register Segment). CSEG va DSEG ga qaraganda RSEG registrlari markaziy protsessor va MP mng ichida bevosita ALQ ga yaqin joylashgan bo'lib, ma'lumotlarga fizik kirishni ta'minlaydi. Ularda oraliq hisob natijalar saqlangan bo'ladir. RSEG registr joyi DSEG ma'lumot maydonidan ajratib qo'yilishi, qisman kesishishi mumkin va DSEGning adres qismi bo'lishi mumkin.

RSEG ning ichki mantiqiy tashkillashtirilishi turli xil bo'lib, mikroprotsessor klassifikatsiyasida asosiy rolni o'ynaydi. MP registrlari funksional turdosh: ular ma'lumotlar yoki adresli ma'lumotlarni saqlash uchun xizmat qiladilar, boshqalar - markaziy protsessorni boshqarish uchun xizmat qiladilar. Shunga asosan ushbu registr-larni: ma'lumot registrlari, ko'rsatkich va maxsus registrlarga ajratish mumkin. Ma'lumot registrlari arifmetik va mantiqiy operatsiyalarni bajarishda operandlar manbai va ma'lumotlarni qabul qilish, adres registrlari yoki ko'rsatkich registrlari asosiy xotirada joylashgan buyruq va ma'lumot adreslarini hisoblash uchun foydalaniladi. Maxsus registrlar-markaziy protsessor holati va tarkibiy qismlarini boshqarishda foydalaniladi. Umumiy ishlatiladigan registrlar bloki (UIRB)-ma'lumot adreslar va ma'lumotlarni saqlash uchun foydalaniladi. Keltirilgan registrlardan foydalanishga qarab mikroprotsessor arxitekturalari afzalliklarini ajratishimiz mumkin. Ma'lumot registrlari ichida ko'pincha bitta registr akkumulyator A (Accumulator) ajratish mumkin. U arifmetik va mantiqiy operatsiyalarni qayta ishlash bilan bog'liq, ya'ni operatsiyalar bajarilishi natijasida har doim bir operand akkumulyatorida saqlanadi va natija akkumulyatorida bo'ladir. Ushbu arxitektura akkumulyatorli deyiladi. Kam-chiligi esa-kichik tezlik, ya'ni har doim operatsiyabajarishdan oldin operandni kiritish kerak.

Ishchi registrlar RO, RI...v.b. - arifmetik va mantiqiy operatsiyalar natijasi bitta registrda emas turli registrlarda saqlanishi mumkin, bu esa o'z navbatida imkoniyatlarni yaratadi. Dasturlash jarayonida registrlar adreslari ko'rsatiladi. Bunday tipli arxitektura registrli arxitektura deyiladi. Bularga intel firmasiga mansub x86 mikroprotsessorlari misol bo'ladir, Bir qancha real vaqtda ishlaydigan mikroprotsessorlarda ishchi registrlar

to'plami nazarda tutilgan. Ma'lum ishchi registrga murojaat ketma-ketlikda bajariladi. Ushbu qurilmalarga intel firmasiga mansub.

Maxsus registrarga MP ichki qismida joylashgan va MP tarkibiy qismlari, turli ishlemi boshqarish funksiyasini bajaradi. Ko'p uchraydigan registrarga adres registrari(yoki ko'rsatkich)-RS va SP- va PSWdastur holat so'zi kiradi. Adres yoki ko'rsatkich registrar, MP ma'ium buyruqlarida foydalaniladigan operand adreslash usullanda foydalaniladi. Ushbu adreslash MP modeliga bog'liq. Ular albatta ikki ishchi funksiyani bajaradi:

- bajanlayotgan buyruqning xotiradagi adresini aniqlaydi(buyruq hisoblagichi);
- joriy stek adresini aniqlaydi(stek ko'rsatkichi).

Turli protsessorlarda har bir funksiya uchun bir yoki ikki ichki registrar ajratilishi mumkin. Keltirilgan registrar bir biridan tarkibim o'zgartirishi bilan farqlanadi. Tarkibidagi dastumi o'zgar- tirilishi kompyuter ishining buzilishiga yoki xotirani o'zgarib ketishiga olib keladi. RS tarkibi quyidagi tartibda o'zgartiriladi.

Tizim ishining boshlanishida o'zgartirilmaydigan qiymat yozib qo'yiladi. Bu nolga teng boigan boshlang'ich ishga tushirilgan dastur adresi. Xotiradan buyruqlami o'qish natijasida buyruq formatiga qarab dastur hisoblagichi tarkibi osha boradi.

Dastur holatlar so'zi:PSW(Program Status Word) operatsiya bajarilish natijalari alomatlanni saqlaydi intel kompaniyasiga tegishli MP x86 ko'rib chiqamiz (2.9-chizma).

- ZF-(zezo) nomli natijaning bayroq belgisi agar nol natija olinsa, 1 yoziladi, aks holda (ZF) = 0.
- CF-(Carry) siljitish bayrogi operatsiya bajarish natijasida katta baytda siljitsa, yoki qo'shish yoki ayirish operatsiya jarayonida 1 qarzga olinganda o'matiladi
- SF -(Sing) natija belgisining bayrog'i birga teng, agar natija manfiy bo'lsa, ya'ni natijaning katta belgisi bitm dubllashtiradi.
- PF -(Parity) juftlik bayrog'i (PF=1) teng agar natija bitining 2 modulli yig'indisi nolga teng boisa (birlik bitlar soni juft).
- AF -qo'shimcha siljitish bayrog'i (Auxiliary) agar kichik tetradaning katta bitidan (bit D3) katta tetradaning kichik bitiga (bit

D4) siljirilganda o'matiladi operatsiyalarda joylashtirilgan BSD sonlari ustida qo'laniladi

- OF -to'ib ketganlik bayrog'i (Overflow) agar operatsiya natijasi bir yoki ikki baytli diapazondan oshib ketganida o'matiladi va boshqa bir necha hollarda o'matiladi.
- TF(Trap Flag)- ketma-ket qadam rejimi bayrog'i.
- IF(Interrupt Flag)-uzilishlarga ruxsat bayrog'i.
- DF(Direction Flag)- qatorli operatsiyalarda ko'rsatish bayrog'i.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
				OF	DF	IF	TF	SF	ZF		AF		PF		CF

2.9-chizma. x86 prossessori registrining holati

Xotirani segmentli tashkil qilish bilan bir qatorda, x86 mikroprot- sessorlarida qo'shimcha ravishda xotirani sahifali tashkil qilish ham mumkin. Xotirani sahifali tashkil qilish mexanizmi CRO regisrining RG bayrog'ini, o'ratish (tashlash) yo'li bilan dasturiy ulanishi (uzilishi) mumkin.

Butun chiziqli adres fazosi soni 1024 ga etadigan bo'limlarga ajratiladi. Har bir boim o'z navbatida o'lchami 4 k bayt qilib belgi- langan 1024 tagacha sahifani ichiga olishi mumkin, sahifalaming boshlang'ich adreslari fizik adres fazosida qat'iy belgilangan. Sahifalaming chegaralari 4 kilobaytli bloklar cliegaralari bilan mos tushadi.

2.6. Adreslash usullari

MPT uchun dastur yozish -juda murakkab jarayondir. Samarali dasturlash uchun protsessor buyruqlar tizimidan xabardor bo'lish kerak. Kompakt va tezkor dastur va qism dasturlar Assembler tilida yaratiladi. Assembler tili mashina tilming raqamli kodlari, buyruqlar tizimining mashina kodlarini ta'minlaydi. Protsessor buyruqlar tizimidan xabardor boimay yozish mumkin boigan dasturlar juda ko'p, ulardan ko'proq ishlatiladigani Si dasturlash tilidir. Assembler dasturlash tili - personal kompyutyerdan tortib mikrokontrollerlarda dasturiy ta'minotning asosiy zarur qismidagi dasturlar uchun effektiv natija beradi. Buyruq kodlari qanday operandlar bilan yoki qaysi buyruq bajanlishito'g'risidaprotsessorga maiumot beradi. Buyruqlar bir baytdan bir necha baytgacha uzunlikka ega boiadi. Buyruq kodlari protsessorda qayta shifrlanadi va mikrooperatsiyalar to'plamiga aylantinladi.

Bir turdagi buyruqlar kirish operandlarini, boshqalari esa chiqish operandlarini talab qiladi. Kirish operandlari manba-operand, chiqish operandlari qabul qilish-operand deyiladi. Hamma operand kodlari qayerdadir joylashishi kerak. Ular ichki regisrlarda, xotira tizimida joylashadilar. Ulaming joylashuvi buyruq kodlari orqali ko'rsatiladi. Maiumotlami qayerdan olib qayerga joylashning turli usullari mavjud boiib ular adreslash usullari deyiladi [12],

Adreslash usuli deb-operand adresini kodlash yoki buyruq kodlaridagi operatsiya natijasiga aytiladi.



2.10-chizma. Buyruq kodlari

OPK-operatsiya kodi.

OPA1- birinchi operand adresi.

AR-natija adresi.

Zamonaviy mikroprotsessor modellarida keng foydalaniladigan adreslash usullari;

-registrli adreslash;

-to'g'ri adreslash;

- bevosita adreslash;

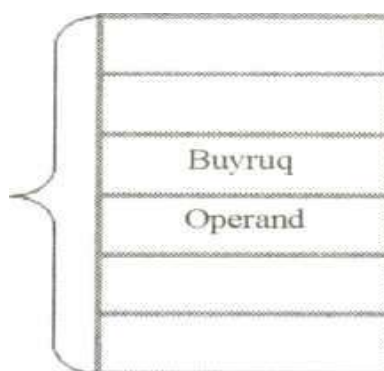
-mavhum registrli adreslash;

-mavhum avtoinkrement/avtodekrementli adreslash; -baza bo'yicha indeksli adreslash;

-segmentli adreslash.

Bevosita adreslash. Bevosita-operand ko'rsatkichining eng oddiy usuli. Komandalar adres maydonida operand adresi emas, balki operand joylashadi. Operand avtomatik ravishda komanda bilan xotiradan chaqinladi va darhol bevosita adreslash uchun tayyor boiadi. Bevosita adreslashda qo'shimcha xotiraga murojaat talab etmaydi. Lekin uning kamchiligi bor. Binnchidan faqat konstantalar bilan ishlash mumkin, ikkinchidan qiymatlar maydoni chegaralangan. Shunga qaramay bu texnologiya ko'p arxitekturalarda butun sonlar ustida ishlash uchun foydalaniladi (2.11-chizma).

Xotira



2.9-chizma. Bevosita adreslash

To'g'ri adreslash. To'g'ri - adres maydonida (ADR) operand adresi joylashadi. Komanda har doim bitta xotira adresiga murojaat eta oladi.

Masalan 1000000 xotira adresini tozalash kerak bolsa, u holda tozalash buyrug'idan keyin joylashadi. (2.12-chizma).

Registrli adreslash. Operand protsessormng ichki registrida joylashadi. Masalan, buyruq quyidagicha boiishi mumkin, nol registrdan birinchi registrga ma'lumot uzatish. Ikkita registrmng raqami buyruq kodlari orqali aniqlanadi. Kod operatsiyalarida registr adreslari shifrovka qilingan. Buyruqda adres maydoni ko'rsatilmaydi (2.13-chizma).

Mavhum registrli adreslash. Protsessor ichki registrida operand emas balki uning adresi joylashadi. Operand fizik adresi mavhum adres DP (Data Pointer)da joylashadi DP (Data Pointer) sifatida UIRB (RON) yoki registr maxsus adres registr joylashadi.

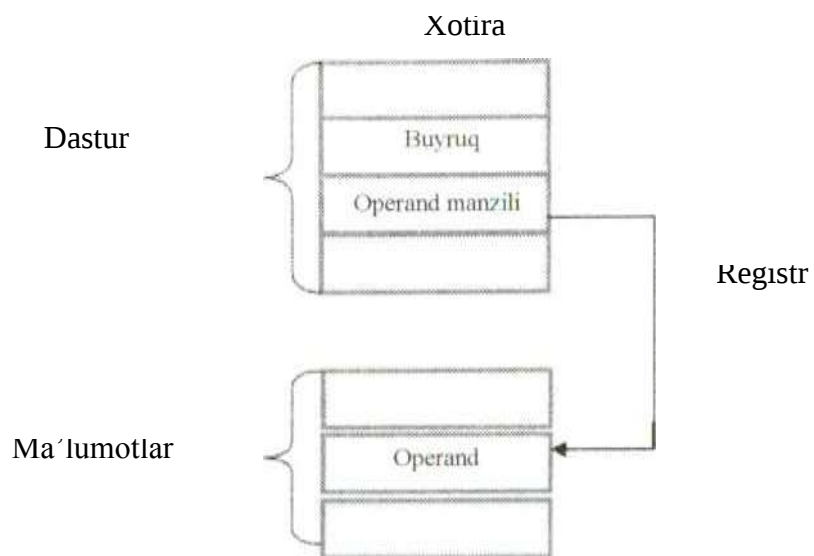
(2.14- chizma).

ADD A, MEM

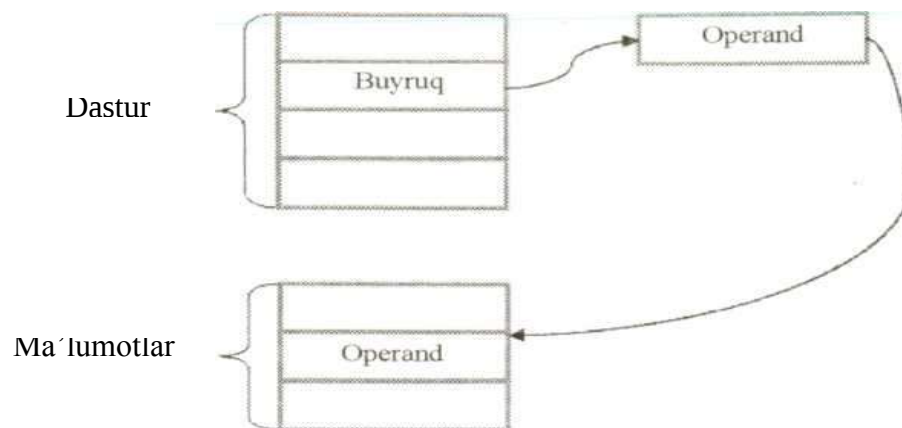
MEM : XD AT A C3F0



2.9-chizma. To'g'ridan to'g'ri adreslash



2.10-chizma. Manzillashni ro'yxatdan o'tkazish



2.14-chizma. Mavhum manzillash

Masalan, Assembler 51 tilida MEM adresi bo'yicha xotiraning ikki baytini tozalash quyidagicha boladi:

MOV DP, #MEM

CLR

MOVX

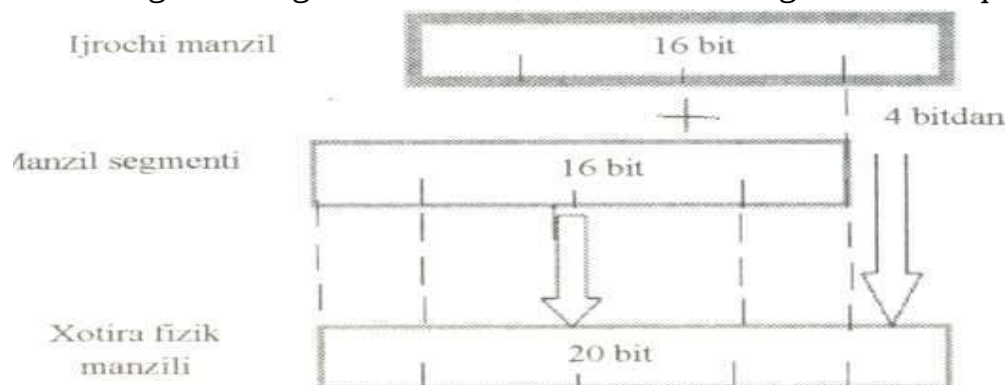
INC

MOVX @DP, a

MEM: XDATA C3F0h

a
@DP, a
DP

Xotirani segmentli adreslash. 8086 nukroprotsessori 20 razryadli adreslar shinasiga ega va u xotiraning 2^{20} yoki taxminan milliondan bir yacheykasiga murojaat qilinishiga imkon beradi. 16 bitli ma'lumotlar shinasiga ma'lumotni baytli yoki so'zli ko'rinishida uzataoladi. Xotira odatda chiziqli bir o'lchamli baytlar massivi ko'rinishida tuzilgan va 2 qo'shni baytlar bir so'zdek ko'rinishi bo'lishi mumkin. Xotiraning butun megabaytli sohasi 64 Kbitli 4 segmentga bo'lingan (2.15,-chizma). MP har bir daqiqada dastur segmentiga (CS), ma'lumotlar segmentiga (DS), qo'shimcha ma'lumotlar segmentiga (ES) va stek segmentiga (SS) murojaat qilish imkoni bor. Bu segmentlarning boshlang'ich adresi CS, SS, DS va ES registrlarida saqlanadi.



2.15-chizma. Manzil segmentidan xotiraning fizik manzilga o'zgarishi

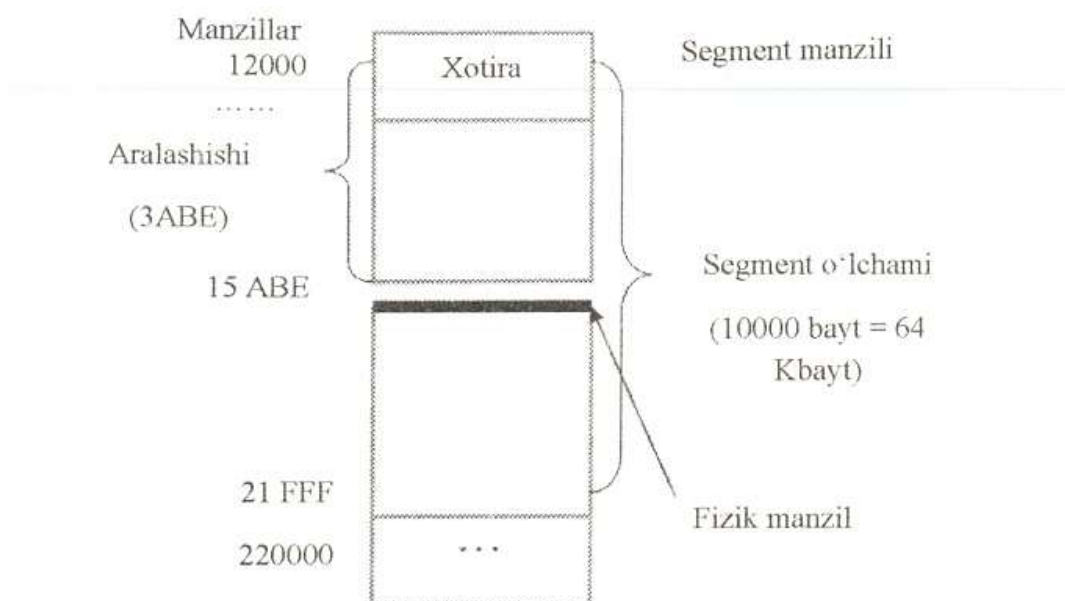
Bu registrlar 16 bitli, adresli soha esa 20 bitli bo'lganligi sababli MP boshlang'ich segmentli adresni 20 bitli summatorida chapga 4 bit suradi (16ga ko'paytirishga teng) va (IP, SP, DI, SI) registrlardan biridagi ma'lumotga qo'shadi. Hosil qilingan son fizik adres deb ataladi. Masalan, xotiradan navbatdagi dastur kodi baytini olishda MP quyidagi formula orqali fizik adresni hosil qiladi:

Fizik adres = (IP) + (CS)*16, bu yerda (IP)-siljitish, bajaruvchi adres (CS)-segmentli adres, (SS)* 16-boshlang'ich segmentli adres deb ataladi. 2.16,-chizmada xotirani tashkil etish keltirilgan.

Protsessor 4 ta guruh buyruqlar to'plamini o'z ichiga oladi:

- ma'lumotlarni uzatish buyruqlari;

- arifmetik buyruqlar;
- mantiqiy buyruqlar;
- o'tish buyruqlari.



2.16-chizma. Segmentdagi fizik manzil (hamma kodlar – o'n oltilikda)

2.7. Mikroprotssessor buyruqlar tizimni

O'tish buyruqlar hech qanday operandalar ustida operatsiyalarni talab qilmaydi. Operandlar manbaa registrlardan (Source) qabul qiluvchi registr'larga (Destination) uzatiladi.

Arifmetik buyruqlar qo'shish, ayirish, Uo'paytirish, boiish, inkrementlash, dekrementlash operatsiyalarini bajaradi. Ushbu buyruqlarga bir yoki ikki kirish operandi zarur bo'ladi.

Mantiqiy buyruqlar mantiqiy VA, YOKI, inversiyalash, turli xil surish operatsiyalarini amalga oshiradilar.

O'tish buyruqlar dastur ketma-ketligi holatini o'zgartiradi. Warning yordamida tarmoqlangan dasturlar va qism dasturlarga o'tish tashkil qilinadi. O'tish buyruqlar shartli va shartsiz bo'ladi. Aynan shu buyruqlar murakkab ma'lumotlarni qayta ishlash operatsiyalarini amalga oshiradilar. Har bir bajarilgan buyruq asosida ma'lumotlar natijasi alomati tekshiriladi (PSW). Turli protsessorlarda komandalar farqlanadi, lekin ularning bajarilish funksiyasi o'xshash. Masalan 8086 protsessorida 133 buyruq bor. Uzatish buyruqlari.

1.MOV DST, SRC; (SRC) ni (DST) ga uzatish. Shu yerda va keyinchalik registr ichidagisi, masalan AL registri (AL) yoki (al) ko'rinishida belgilanadi izohni uzatish esa <-- belgi bilan belgilanadi.

mov al ch; <—(ch)

mov cx, dx ;

mov bn, [mems]; merns simvollik adresli xotira yacheykasi ichidagisi VN registrga uzatish

mov al, [bx]; VX registrida joy lashgan adresli xotira yacheykasini akkumulyatorga uzatish.

mov bx, OFFSET src;joriy segmentda SRC xotira yacheykasi adresining siljishini BX gajoylashtirish.

Buyruqlar jadvali

2.1- jadval

Buyruq ilishidan oldin al, [table+bx]	registr VX	registr AL	Manzil	Kod
	010	XX	0800(table)	c
			8xx	Cx
			810	8
Buyruq ilishidan so'ng al, [table+bx]	registr VX	registr AL	Manzil	Kod
	010	XX	0800(table)	c
			8xx	Cx
			810	8

1.PUSHRP; juftlik registridan ma'lumotni stekning yuqorisiga joylashtirish (masalan, push bx).

2.POPRP; stek yuqorisidan 2 bayt olib uni RP juftlikka joylashtirish (masalan, pop ax).

3.XChG DST, SRC; (DST) va (SRC)lar ichidagilarini joylarini almashtirish. Ikki operand bir vaqtda xotira yacheykasidagi ma'lumot bo'la olmaydi.

4.XLAT SRC; jadval boshidan (AL) soniga teng ma'lumotlar bayti SRC boshlang'ich adresli jadvaldan olib, uni AL ga joylashtirish. SRC adres BX registrida joylashgan boiishi kerak.

5.IN ACCUM, PORT; AL yoki AX baytini yoki PORT adresli portdan so'zni akkumulyatorga joylashtirish. Agar port adresi <= FF bo'lsa, port adresi bevosita ko'rsatilishi mumkin, agar port adresi >FF bo'lsa, port adresi registri ifodasi orqali

bevosita ko'rsatiladi (POH maxsus funksiyasi).

OUT PORT, ACCUM, AL akkumulyatoridan yoki AX baytidan yoki so'zni PORT simvolik adresni TQ ga uzatish.

6.OUTOUT PORT, ACCUM; AL akkumulyatoridan yoki AX baytidan yoki so'zni PORT simvolik adresni TK ga uzatish.

Arifmetik buyruqlar.

1.ADD DST, SRC; SRC va DST qiymatlarini qo'shish.

add al, [membute]; mem_bute-bir bayti xotira yacheykasi

add [memword], dx; memword-ikki baytii xotira yacheykasi

add ch, 10001010b;

2.INC DST; (DST) ni 1 taga oshirish (inkrement (DST)).

inc si; (SI) $\leftarrow$ (SI) + 1

me count; (count) $\leftarrow$ (count) + 1

3.SUB DST, SRC; (SRC) ni (DST)dan ayirish va natijani DST ga joylashtirish.

4.DEC DST; (DST) ni 1 taga kamaytirish; _k

5.CMP DST, SRC; DST va SRC ni ichidagisini solishtirish. Bu buyruq (SRC) dan (DST)m ayirishni bajaradi, lekin farqini DST ga joylashtirmaydi va operatsiya natijasini natijasi bo'yicha bayroqlarga ta'sir ko'rsatadi.

DST > SRC	OF SF ZF CF
DST = SRC	0 / 1 0 0 0
DST < SRC	0 0 1 0
	0 / 1 1 0 1
shart	bayroqlar

ayirish operatsiyasi natijasida qoldiq qiymatini katta ikkilik razryadiga o'tkazganda va shu katta ikkilik razryadidagi bilan mos kelganda boshqa aniqlashda OF 1 qiymatini qabul qiladi, agar natija berilgan oraliq mos ravishdagi sonlar oralig'idan oshib ketsa.

0/1 - operatsiyaning qiymatiga qarab bayroqni 0 yoki 1 ga tengligini bildiradi.

OF va SF, bayroq belgili sonli operatsiyalarda ma'noga ega, CF bayrog'i esa 1 ga qo'yiladi, agar qo'shish yoki

DST > SRC va lkkalasi ham bir baytli sonlar boisa, unda:

DST: 1. (+127) 2. (+127)

SRC: - (+2) - (-2)

(+125) (OF)=0 (+129)? (OF)=1

Ikkinchi misolda natija oraliqdan oshib ketayapti: $128 \leq x \leq +127$ SF belgilangan bayroq "1" ga qo'yiladi, agar operatsiya natijasining katta biti lga teng bo'lsa, ya'ni masofiy natijada. Teskari bo'lganda tushirib ketiladi. ZF nol bayrog'i (!) nolli natijada "1" ga qo'yiladi aksida tushirib qoldiriladi. Uzatish bayrog'ga CF=1 teng, agar qo'shishida katta razryaddan uzatish yoki ayirishda kichik razryaddan qoldiq olish

bo'lsa. Aks holda bayroq tushirib qoldiriladi. Birinchi misol uchun SF=ZF=CF=0, ikkinchisi uchun: SF=1, ZF=CF=0.

Mantiqiy va so'nish buyruqlari.

1.AND DST, SRC; razryad bo'yicha "I" (VA).

mov dh, 10101100b;

and dh, OfOh;

shu ikki buyruqni bajarilishi natijasida DH ning ichidagisi 10100000b ga teng bolib qoladi.

2.OR DST, SRC; razryad bo'yicha mantiqiy element "ILI" (YOKI).

or bx,dx;esli (BX)=5F0Fh,(DX)=7777h bo'lsa operatsiyadan so'ng;(BX)=7F7Fh.

Mantiqiy surish jadvali

2.2-jadval

BX	101 1111 0000 1111= 5F0F
DX	111 0111 0111 0111 = 7777
BX (natija)	111 1111 0111 1111 = 7F7F

3.XOR DST, SRC; razryad bo'yicha mantiqiy "sikl. ILI" xor al,55h;agar (AL)=5ah bo'lsa, operatsiyadan so'ng (AL)=0fh.

4.NOT DST; qabul qiluvchi hamma bitlarni inversiyasi.

5.TEST TEST DST, SRC; AND operatsiyasini bajaradi, ammo faqat bayroqlarga operatsiyalarni o'zgartirmasdan ta'sir ko'rsatadi.

6.ShR DST, CNT; mantiqiy o'nga surishgandan bo'sh qolayotgan bitlar nollar bilan to'ldiriladi, o'ng tomondagi chetki bit SF bayrog'iga chiqarib yubonladi. DST operandi xotira yacheykasi bolishi mumkin.

mov bl,10110010b,(CF) = x

shr bl, 1 ;(BL) = 01011001 ,(CF) = 0

Siljishgacha		1	0	1	1	0	0	1	0		(CF)=X
Siljishdan keyin	0— ->	0	1	0	1	1	0	0	1	— ->	(CF)=0

mov cl,4;

shr bl,cl;(BL) = 00000101, (CF) = 1.

ShL DST, CNT; matiqiy chapga surish.

RLC DST, CNT; qoldiqni o'tkazish orqali chapga siklik surish.

7.RRC DST, CNT; qoldiqni o'tkazish orqali o'ngga siklik surish.

8.ROR DST, CNT; chapga siklik surish.

9.ROL DST, CNT; o'ngga siklik surish.

Boshqarishni uzatish buyruqlari

1.CALL SUB R ; SUBR adresli qism dasturini chaqirish call delay; *

mov....

2.RET CALL ga bevosita qism dastundan keyingi operatorga qaytarish, ya'm yuqorida keltirilgan misoldagi MOV ga .

3.JMP NAME ; NAME simvolik adresli buyruqga shartsiz o'tish jmp short name; name belgisiga o'tish, quyidagidan kam

bolmagan holda:

-128 yoki +127 bayt. jmp near name; name belgisiga o'tish, quyidagidan kam bo'lmagan holda:

65535 bay, bitta segmentda. jmp name;oddiy jmp near name;

4.JA NAME yoki JNBE NAME : shartli o'tish, agar masalan: CMR DST, SRC solishtirish natijasida qabul qiluvchi uzatuvchidan absolyut kattalik bo'yicha katta bolsa, name belgisiga o'tish kerak.

5.JB NAME yoki JNAE NAME: shartli o'tish agar masalan, CMR DST, SRC solishtirish natijasida qabul qiluvchi manbadan absolyut kattalik bo'yicha kichik boisa, unda name belgisiga o'tish kerak (4n va 5n, buyruqlari belgisiz sonli operatsiyalami bajarish natijalari ustida bajariladi).

6.JZ NAME yoki JE NAME: oiish, agar nol inchi bayroqqa ta'sir qiluvchi operatsiya natijasi - nol bolsa ("nol" **bo'yicha** o'tish).

7.JNZ NAME: yoki JNE NAME; "nol emas" bo'yicha olish.

Sikllami boshqarish buyruqlari .

1.LOOP NAME: bu buyruq (SX)ni 1 ga noaniq kamaytiradi va yaqin belgiga oiishni bajaradi, agar (SX) 0 gateng boimasa.

2.LOOPZ NAME yoki LOOPE NAME: bundan tashqari ZF bayrogining tekshirishni ham bajaradi. Shuning uchun sikl (CX)=0 yoki (ZF)=0 yoki u ham bu ham birgalikda bolgandagi shart bilan tugaydi. Shuning bilan bu buyruq birinchi nol emas boigan natijani aniqlash uchun xizmat qiladi.

3.LOOP NZ, LOOP NE - "nol" boyicha sikldan chiqish.

Qatorlar ustida ishlash buyruqlari

1.LODSB: lodsb buyrugi maiumotlar segmentidan ST regisr orqali adreslangan baytm yuklaydi va ST ni lga ko'paytiradi, agar bundan oldin CLD buyrugi kintilgan bo'lsa (DF yo'nalish bayrog'ini tozalash) va St ni lga kamaytiradi, agar STD buyrugi ishlatilgan boisa (yo'nalish bayrogi o'matilishi).

DATA

string DB 'abcdefg'¹ CODE

cld; AL da bu buyruqlami bajarilgandan so'ng mov si,OFFSET [string+2]; ASCII ga 's' kodi yuklandi. lodsb; SI tarkibidagilar'd¹ ni ko'rsatadi.

2. MOVSB: bu buyniq SI registridagi xotira yacheykasi adresidan bir baytni DI registrdagi xotira yacheykasi adresiga oikazadi va (SI) va (DI) ni 1 ga ko'paytiradi. SI ning qiymati DS ma'lumotlar segmentida ham qo'shimcha ES segmentidajoylashishi mumkin. DI ning qiymati faqat ES qo'shimcha segmentdajoylashishi mumkin.

DATA

msg DB 'Hammasi O K.'

LEN = \$ - msg; LEN 8 ga teng CODE

cld

lea si,msg;v SI manba adresi

mov ax,0b800h; Bnqeo xotira segment hajmi

mov es,ax;qo'shimcha segmentga o'tkazish

lea di,es:(0a0h * 3); yuqoridan 4-satr

mov cx,LEN; LEN - chiqish simvollar soni

rp: movsb; ekranning joriy pozitsiyasiga simvolni yuborish

me di;atribut pozitsiyasidan sakrab o'tish

loop rp; tugaguncha davom ettirish (CX)

...; stroka 'Vsyo O.K.' displey yuqorisidan ;4- satrga chiqariladi.

3. REP: buyruqm qaytarish prefiksi. Masalan, oldingi dastuming tugashi quyidagichayozilishi mumkin:

mov cx,LEN; qaramasdan, massivi 'Hammasi O.K.' boiadi. rep movsb; B800 ni xotira maydoniga ko'chiriladi:(AO * 3), ga

;display ekraniga quyidagi yozuv chiqadi:Hammasi OK. Nimaga?

4. CMPSB; (ST) adresni beruvchining qator baytini, (DT) adresli qabul qiluvchining qator baytini sohstirishni bajaradi: ya'm ((ST))- ((DI)) ayirishni bajaradi. CMP DST, SRC buyrug'i bilan yangilish- maslik kerak, qaysiki qabul qiluvchidan beruvchini ayirish bajariladi. CMPSB buyrug'i (CX)ni 1 ganoaniq kamaytiradi va (SI) va (DI) ni 1 ga ko'paytiradi, agar (DF)=0 bo'lsa.

5. REPZ yoki REPE: qaytarish prefiksi. Agar (CX)=0 yoki (ZF)=0 bo'lsa, buyruqm bajarish tamomlanadi.

DATA

src DB 'To be, or not to be' dst DB 'To be,or not to be'

len = \$ - dst; len 19 ga teng CODE

cld; (DF) = 0

push ds; manzillami joylashtirish pop es; ds va es segmentlari mov cx, len; dst qator uzunligini cx gayubonsh lea di, dst; manzilim yuklash (tartibtgajoylashish) qatorlami dst dan DI ga lea si,src

repe cmpsb; baytlab solishtirish

je equal; agarda baytlar mos tushsa, unda belgilarga o'tish not cx; agar yo'q bo'lsa - mos tushmagan baytlarni hisoblash add cx,len; jmp notequal; equal:

notequal:

Ushbu masala oxirida mos kelmaydigan birinchi nomeri bayt (CX)=5 ga teng.

Miroprotsessorni boshqarish buyruqlari.

- CLC: o'tkazish bayrog'ini tushirib yuborish (CF)~0. .
- STC: o'tkazish bayrog'ini o'matish (SF)=1.
- CMC: yo'nalish bayrog'im inverslash.
- CLD: yo'nalish bayrog'ini tozalash(DF)=0, bunday holatda qator (bayt zanjirlari ustida operatsiyalar) kichik adresdan katta adresgacha bajariladi.
- STD: yo'nalish bayrog'ini o'matish (DF)=1, baytlar zanjirlari ustida ishlash katta adresdan kichigigacha bajariladi.
- STI: urinish bayrogini o'matish (IF)=1, tashqi qurilmalardan uzilishni ruxsat berish.
- CLT: uzilish bayrogini tozalash.
- MOP: maxfiy operatsiya.

Uzilish buyruqlari.

1. INT INUM : bu komanda dasturli uzilishni chaqiradi, ya'ni to'rt baytdan saqlanayotgan adresni INUM *4 adresidan boshlab, bu yerda INUM=(0...25) ga teng xotira yacheykasiga o'tish Bu 4 baytli son berilgan uzilishning qism dasturi qayta ishlovchining ko'rsatkichi boiadi va u boshqacha nomi uzilish vektori deb ataladi.

Dastur uzilishlar yordamida vujudga keladigan operatsiyalar AN registridagi kod orqali aniqlanadi, masalan,

mov ah,14d; displeyga simvollarni chiqarish

mov al,31h; kursoni bitta o'ngga surish.

int 10h; ekranga '1' simvolni chiqarish(ASCII kod 31 h).

DATA

privet DB 'Xayrli tong!'

CODE

lea dx,privet;DX registring maxsus funksiyasi mov ah,9;9 - displeyga chiqarish int 21h;salomlashish chiqariladi .

Xulosa.

Xulosa qilib shuni aytish mumkinki Arifmetik amallarni bajarishda teskari, qo'shimcha ikkilik-o'nlik va ikkilik kodining boshqa turlarini tadqiq etish maqsadga muvofiq bo'ladi Ulami qo'llaganda o'zgartirgich elektr signallarini maksimal soddalashtirish va amallarni bajarish vaqtini kamaytirish mumkin. Bunday qulayliklarga erishish natijasida vazifa bajarilishi osonlashadi.

Raqamli tizimlar, umuman olganda. tashqi muhit bilan doim periferiya (tashqi) qurilmalar orqali bog'langan.

Raqamli qurilmalar inson xayotida tobora ko'proq ahamiyatga ega bo'lib bormoqda va deyarli barcha qurilmalar shifratlar asosida loyixalashtirilmoqda. Bu yakun emas ya'ni bunday qurilmalar rivojlanishda davom etadi.

Foydalanilgan adabiyotlar

1. X.YU.Abasxanova,U.B.Amirsaidov. Mikropratsessorlar.Toshkent 2016.
2. W.W.W. Ziya.net
3. W.W.W. Google.uz
4. W.W.W. Kutubxona.uz