

MONS este furnizat intr-o forma relocatabila. Se incarca pur si simplu la adresa de la care se doreste executarea, si se lanseaza in executie. Daca se doreste relansarea lui (din BASIC) atunci se executa de la o adresa cu 29 (zecimal) mai mare decit adresa originala. MONS are 5K lungime odata ce a fost relocat dar consuma aproape 6K pentru incarcare din cauza 'tabloului de relocatari' a adreselor care urmeaza dupa codul principal. MONS contine propria sa stiva si e un program de sine statator. La intrarea in MONS adresele sint afisate prin 'PANOUL FRONTAL' in hexagesimal.

Aceste adrese se pot converti in zecimal cu comanda SS+3. Cu toate acestea adresele trebuie introduse intodeauna in hexagesimal. Comenzile se introduc de la claviatura in urma cursorului '>' in forma de litere mari sau mici. Unele comenzi a caror efect poate fi dezastros daca sint utilizate gresit necesita apasarea lui SS impreuna cu litera de comanda. Comenzile se executa imediat si nu este nevoie de incheierea lor cu ENTER.

Unele comenzi necesita introducerea de numere hexagesimale.

Daca terminatorul e semnul '-' atunci se introduce forma negativa in complementul lui 2. Daca se introduc mai mult de 4 caractere atunci doar ultimele 4 se iau in considerare. In orice moment se poate intoarce la interpretorul BASIC prin apasarea lui CS+1.

### COMENZI

SS+3 comuta baza de numeratie a adreselor afisate (hexa->zecimal). Aceasta comanda afecteaza toate adresele afisate de MONS inclusiv cele generate in cursul dezasamblarii dar nu schimba afisarea continutului memoriei care se face intodeauna in hexa.

SS+4 (sau \$) afiseaza o pagina dezansamblata incepind de la adresa continuta in 'MEMORY POINTER'. La a 2-a apasare se face reintoarcerea la 'PANOUL FRONTAL' si orice alta tasta continua cu urmatoarea pagina de dezansamblat.

ENTER incrementeaza 'MEMORY POINTER' cu 1 astfel incit cei 24 de bytes afisati sint concentrati in jurul unei adrese mai mari cu 1 decit cea precedenta.

CS+7 decrementeaza 'MEMORY POINTER' cu 1

CS+5 decrementeaza 'MEMORY POINTER' cu 8

CS+8 incrementeaza 'MEMORY POINTER' cu 8

'>' modifica 'MEMORY POINTER' astfel incit sa contina adresa curenta a stivei (indicata de SP). Aceasta comanda e utila cind se doreste o privire de ansamblu asupra adreselor de intoarcere a rutinelor chematoare.

'G' cauta in memorie un anumit sir (GET). Se afiseaza ':' si se introduce primul byte care se cauta urmat de ENTER si tot asa in continuare pina se defineste intreg sirul.

'H' converteste un numar zecimal in forma sa echivalenta hexazecimala.

'I' (copie inteligenta) este folosit pentru copierea unui bloc de memorie de la o locatie la alta. Este inteligenta in sensul ca blocul de memorie poate fi copiat in locatii unde ar fi 'overlap' locatiile precedente. 'I' raspunde intrebind 'FIRST:' care sint adresele (inclusiv) de inceput si sfirsit ale blocului care trebuie copiat si apoi 'TO:' care reprezinta adresa la care blocul trebuie copiat.

Daca adresa de start este mai mare decit adresa de sfirsit comanda nu se executa. Numerele se introduc in forma hexazecimala.

'J' executa codul de la adresa specificata. Comanda raspunde cu ':' asteptind un numar in hexagesimal care odata introdus duce la resetarea stivei interne, ecranul este sters si executia transferata la adresa specificata. Daca se doreste reintoarcerea la 'PANOUL FRONTAL' dupa executarea codului, se alege un punct de intrerupere cu comanda 'W' in locul la care se doreste intoarcerea la display.

!!! 'J' corupe registrele Z80 inainte de a executa codul. Daca se doreste executarea codului cu anumite valori ale registrelor se executa 'SS+K' de mai jos.

'SS+K' continua executia de la adresa curenta a lui PC. Aceasta comanda s-ar utiliza cel mai frecvent impreuna cu 'W'.

'L' tabeleaza sau listeaza un bloc de memorie incepind de la adresa curenta continuta in MP.

Comanda sterge ecranul si afiseaza reprezentarea hexazecimala si echivalentul ASCII a 80 de nytes de memorie incepind de la valoarea curenta a MP. Adresele se afiseaza in hexagesimal sau in zecimal in functie de starea lui 'SS+3'.

Display-ul este format din 20 de rinduri a 4 bytes/rind si codurile ASCII la sfirsitul rindului.

Pentru scopurile acestei comenzi adresele sub 127 sint decrementate cu 128 si orice valoare intre 0 si 31 este afisata ca ' '.

La sfirsitul paginii se poate face reintoarcerea la 'PANOUL FRONTAL' cu 'CS+S' sau continua cu orice alta tasta.

'M' seteaza MP pe adresa specificata. Raspunde cu ':' si asteapta un numar in hexagesimal. MP este actualizat cu adresa introdusa si display-ul 'PANOULUI FRONTAL' se modifica corespunzator.

'N' initiaza editarea urmatorului sir specificat de comanda 'G'.

Comanda 'G' permite definirea unui sir si cauta prima sa aparitie. Comanda 'N' permite cautarea urmatoarelor aparitii. Cautarea incepe de la MP si actualizeaza display-ul la aparitia sirului cautat.

'O' merge la destinatia unei deplasari relative. Comanda ia byte-ul adresat curent de MP si-l trateaza ca o deplasare relativa actualizind display-ul corespunzator.

De retinut ca deplasările relative mai mici de 07F (127) sint tratate ca negative de microprocesorul Z80, fapt de care comanda 'O' tine cont (vezi comanda 'U').

'P' umple memoria dintre limitele specificate cu un octet specificat.

Intreaba 'FIRST:', 'LAST:' si 'WITH:'. Se introduc numerele in forma hexazecimala : adresa de inceput si de sfirsit a blocului de memorie care trebuie umplut si byte-ul cu care se doreste umplerea blocului de memorie.

'Q' cauta setul de registre. La intrarea in 'PANOUL FRONTAL' setul de registre afisat este cel standard (AF,HL,DE,BC). Folosirea lui Q va afisa setul alternativ de registre (AF',HL',DE',BC') care se poate distinge de cel standard prin semnul '<'> dupa numele registrului.

Daca 'Q' este folosit cind se afiseaza registrul alternativ se trece la cel standard.

'SS+T' stabileste un punct de intrerupere dupa instructiunea curenta si continua executia.

'T' dezassembleaza o portiune de cod, optional la imprimanta. Prima data intreaba 'FIRST' si 'LAST:' adresele codului pentru care se doreste dezasamblarea, in hexagesimal. Daca adresa de inceput este mai mare decit adresa de sfirsit, comanda nu se executa. Dupa introducerea acestor adrese se intreaba 'Printer?'. Se raspunde cu Y (capital) daca se doreste ca dezasamblarea sa fie directionata la codul imprimantei sau cu orice altceva daca iesirea se face pe ecran. Urmeaza intrebarea 'Text?' pentru a introduce in hexagesimal adresa de start a fisierului text pe care-l produce dezasamblarea. Daca nu se doreste generarea unui fisier text se apasa ENTER la aceasta intrebare. Fisierul se produce intr-o forma in care poate fi utilizata de GENS3. Daca se doreste utilizarea textului cu GENS3 trebuie generat la sau mutat la prima adresa data de comanda editorului 'X' deoarece aceasta este adresa de start a 'fisierului text' asteptat de GENS3. Trebuie specificata si adresa de sfirsit a textului. Aceasta se face luind adresa 'End of text' data de dezasamblor si punind-o in locatia TEXTEND a lui GENS3 (vezi manualul GENS3). Apoi se introduce GENS3 cu startul cald pentru a pastra textul. Daca la orice stadiu cind se genereaza fisierul text acesta s-ar scrie peste MONS dezasamblarea este abandonata - se apasa orice tasta pentru revenirea la 'PANOUL FRONTAL'.

Daca se specifica o adresa pentru fisierul text se intreaba in continuare 'workspace:' - adresa spatiului de lucru - care ar trebui sa fie inceputul locului gol al memoriei care

este folosit pentru o tabela de simboluri primitive pentru orice label generat in procesul dezansamblari. Cantitatea de memorie necesara este de 2 bytes pentru orice label generat. Daca se apasa ENTER adresa subinteleasa este # 6000 in hexagesimal.

Apoi se intreaba in mod repetat 'FIRST:' si 'LAST:' adresele (inclusive) pentru blocurile de date care la dezansamblare vor fi interpretate ca DEFB.

Daca valoarea byte-ului de date este intre 32 si 127 (# 26 si # 77) inclusiv, atunci se da respectivului octet interpretare in caractere ASCII. Cind s-au terminat domeniile de date specificate sau daca nu se specifica nici un domeniu se apasa ENTER pentru ambele intrebari.

Comanda 'T' foloseste domeniului de la sfirsitul lui MONS pentru a inmagazina adresele domeniilor de date astfel incit se pot alege atitea domenii de date cit exista memorie disponibila; fiecare domeniu de date necesita 4 bytes pentru inmagazinare. De remarcat caaceasta comanda distruge punctele de intrerupere (comanda 'W').

In acest moment ecranul va fi sters. Daca s-a cerut crearea fisierului text va urma o scurta intirziere (dependenta de marimea sectiunii de memorie care trebuie dezansamblata) cauzata de construirea tabelului de simboluri. Aceasta odata facuta, listingul dezansamblat va aparea pe ecran sau la imprimanta - listingul poate fi intrerupt cu tastele ENTER sau SPACE respectiv CS+5 pentru intoarcerea la 'PANOUL FRONTAL' sau orice alta tasta (mai, putin CS+1) pentru a continua dezansamblarea. Daca un 'opcode' invalid este intilnit, este dezansamblat ca un NOP si urmat de un 'a' (asterix) dupa 'opcode'-ul din listing.

La sfirsitul dezansamblarii display-ul va astepta si daca s-a cerut producerea unui fisier text se afiseaza mesajul 'End of text XXXXX' este adresa in hexagesimal sau zecimal care trebuie POKE-ate ('L' mai intii) in GENS3 la locatia TEXTEND pentru ca ansamblorul sa poata 'culege' adresa fisierului text la un start cald cind s-a incheiat se apasa tasta pentru intoarcere la 'PANOUL FRONTAL' sau CS+1 care (re)aduce in BASIC.

Label-urile sint generate (cind sint relevante) in forma L XXXXX unde XXXXX este adresa absoluta in hexagesimal a label-ului, dar doar daca respectiva adresa se afla in limitele dezansamblarii. Daca adresa se afla in afara acestui domeniu, nu se genereaza nici un label si se da pur si simplu adresa zecimala sau hexagesimala. Daca o anume adresa s-a referit la o instructiune in dezansamblare atunci label-ul sau va aparea in cimpul label-ului (inainte de mnemonic) doar daca listingul este directionat la un fisier text.

'U' folosit impreuna cu comanda 'O'.

Dupa cum s-a precizat comanda 'O' actualizeaza display-ul corespunzator unei deplasari relative (de regula pentru a arata efectul unei instructiuni de tip JR sau JR NZ). 'U' este folosit pentru a actualiza display-ul cu valorile dinaintea executarii comenzii 'O' (a ultimei comenzi 'O').

'V' folosit impreuna cu comanda 'X'

'V' este smilara comenzii 'U' cu deosebirea ca actualizeaza display-ul acolo unde era inainte de executia ultimei comenzi 'X'.

'W' determina un punct de intrerupere la adresa MP.

Un 'punct de intrerupere' -din punctul de vedere al lui MONS- este o instructiune de tip CALL a unei subrutine care afiseaza 'PANOUL FRONTAL' permitind programatorului sa opreasca executia programului si sa inspecteze registrele, flag-urile si orice adrese relevante. Cei 3 bytes ai rutinei de CALL sint inlocuiti cu cei originali deindata dupa executia opririi.

MONS foloseste spatiul de la sfirsit pentru aceasta actiune si deci se pot alege atitea puncte de intrerupere cit exista memorie. Fiecare punct de intrerupere necesita 5 bytes pentru inmagazinare.

!!! Atentie la locul unde se alege un punct de intrerupere pentru a nu patrunde in corpul unei instructiuni pe mai multi octeti.

'X' folosita pentru a actualiza MP cu destinatia unei instructiuni de tip CALL sau JP absolut.

'X' ia o adresa (pe 16 biti) specificata de la MP si MP+1 si apoi actualizeaza display-ul astfel incit sa fie centrat in jurul acestei adrese. De retinut ca primul byte este cel mai putin semnificativ (vezi comanda 'V').

'Y' introduce ASCII de la MP  
'Y' da o noua linie in care se pot introduce caractere ASCII direct de la claviatura. Acestea se introduc in forma hexazrcimala echivalenta si se introduc ni memorie incepind de la valoarea curenta a lui MP. Sirul de caractere trebuie terminat cu CS+5 si DELETE (CS+0) pentru a sterge caracterele din sir. Cind s-a terminat introducerea caracterelor si CS+5 display-ul se actualizeaza astfel incit MP este pozitionat imediat dupa sfirsitul sirului in memorie.

'SS+Z' executie pas cu pas  
Inainte de folosirea acestei comenzi atit PC cit si MP trebuie stabilite la adresa instructiunii care se doreste executata.

Comanda, executa instructiunea curenta si actualizeaza 'PANOUL FRONTAL' pentru a reflecta schimbarile determinate de executarea instructiunii.

Se poate merge pas cu pas si in RAM si in ROM dar doar daca intreruperile sint dezactivate.

<'> (SS+P) Aceasta comanda este exact ca si 'L' (LIST) cu deosebirea ca iesirea se face la canalul imprimantei in loc de ecran.

De retinut ca la sfirsitul paginii se apasa CS+5 pentru intoarcerea la 'PANOUL FRONTAL' sau orice alta tasta pentru urmatoarea pagina (mai putin cs+1).

#### MODIFICAREA MEMORIEI

Continutul adresei date de MP poate fi modificat prin introducerea numarului hexagesimal urmat de un terminator. Daca terminatorul nu este ceva valid comanda nu se executa.

#### MODIFICAREA REGISTRELOR

Daca un numar hexagesimal este introdus ca raspuns la intrebarea '>' si este terminat cu '.', atunci numarul specificat va fi introdus in registrul Z80 adresat curent de sageata '->'. La intrebarea MONS '->' indicind PC introducind '.' ca terminator al numarului hexagesimal, el va modifica PC. Daca se foloseste '.' de unul singur (nu ca terminator) pointerul se va rotii circular de la PC la AF. Nu este posibila adresarea (deci nici schimbarea) SP (STACK POINTER) sau a registrului IR. '.' poate fi folosit si pentru modificarea setului alternativ de registre daca acesta este afisat. Se foloseste comanda 'Q' pentru a comuta setul de registre.

#### PANOUL FRONTAL

Primele 9 linii ale display-ului contin registrele Z80, numele lor, valoarea lor prezenta si continutul a 7 locatii de memorie incepind de la adresa specificata de registru. Registrul cu FLAG-urile este decodificat pentru a arata FLAG-urile setate. Pointer-ul de registru '->' arata registrul adresat curent. Cei 24 bytes afisatii in partea de jos, sint organizati ca adrese urmate de continutul lor, centrate in jurul valorii marcate >...<.