

THEMA

ΤΟΥ ΓΙΑΝΝΗ ΚΟΝΤΟΥΛΗ

ΠΡΟΓΡΑΜΜΑ
ΜΟΝΙΜΟ
ΓΙΑ ΤΗΝ
AMSTRAD

Το πρόγραμμα Μονίμο με το οποίο θα υποφέρετε να εξεταστεί με αξιομέτρηση τη μηνήμη του AMSTRAD 486/664/6128 είναι κάτι απαραίτητο για οποιαδήποτε ασχολία, βεβαίως και στοχευμένα με τον κάθε μηχανή. Με αυτό το πρόγραμμα λοιπόν, έχετε τη δυνατότητα να «εξερευνήσετε» όχι μόνο τη RAM αλλά και τη ROM του υπολογιστή σας. Θα υποφέρετε να δείτε πώς αποθηκεύεται το πρόγραμμα στη RAM (αριθμός μημής 1-40). ▶

καθώς και να εξεταστέ το χώρο εργο-
σίας (workspace) του υπολογιστή όπου
βρίσκονται αρκετές χρήσιμες πληροφο-
ρίες, όπως οι HEADER για τις εντολές
CAT, SAVE, LOAD (σελίδα 184), τα περι-
εχόμενα των προγραμματοζυγίων
πληκτρών (σελ. 180) και οι υπορουτι-
νές του λειτουργικού συστήματος (187-
189 σελ.). Στη ROM, θα δείτε τα
μηνύματα λαθών (204-206 σελ.), τις
εξαιρέσεις λέξεων και τα μηνύματα του
υπολογιστή (σελίδες μηνύματ. ROM 6, 7,
39, 40, 227-230).

Η παρουσίαση γίνεται σε 256 σελίδες
για τη RAM (0-255) και σε 128 σελίδες για
τη ROM (0-63 και 192-255). Κάθε σελίδα
αναφέρει στο πάνω μέρος αν δείχνει
την ROM ή την RAM, μετά δίνεται ο αυ-
ξαν. αριθμός της σελίδας και οι δευτεν-
οεισ της μνήμης τις οποίες παρουσιάζει
σε δεκαδικό σύστημα αριθμώσεως. Από
κάτω, σε ένα πινάκo 16x16 δίνονται σε
δεκαεξαδική μορφή τα περιεχόμενα της
μνήμης και δεξιά σε μια στήλη η αν-
τιστοίχη ASCII απεικόνισή τους. Για την
απικόνισή αυτή δε χρησιμοποιούνται
μόνο οι standard ASCII χαρακτήρες (32-
128), αλλά όλο το set χαρακτήρων του
AMSTRAD.

ΔΥΝΑΤΟΤΗΤΕΣ

Οι δυνατότητες που σας παρέχει το
πρόγραμμα είναι οι εξής:

- <SPACE> (ιδιόστημα):

Πάντα,ς το πλήκτρο αυτό θα εμφανί-
ζεται η αμέσως επόμενη σελίδα.

- <A> ΑΛΛΑΞΕ

Για να αλλάξετε σελίδα θα χρησιμο-
ποιείτε το πλήκτρο «A».

- <M> ΜΕΤΑΤΡΟΠΗ

Σ' αυτή τη λειτουργία, θα πρέπει να
εισάγετε από ποιο αριθμητικό σύστημα θα
γίνεται η μετατροπή και σε ποίο, και μετά
θα εισάγετε τους αριθμούς τον ένα μετά
τον άλλο, κανόνια. Για να επιστρέψετε
στο κυρίως πρόγραμμα θα πρέπει να
εισάγετε το σύμβολο "\n" (δεξιά κάτω
στα πληκτρολόγια). Η μετατροπή μπορεί
να γίνει από δεκαδικό έως τριανταεξαδικό
σύστημα.

- <BEAN>

Το βέλη κινών ταυτόχρονα τους δυο
δρομείς, τον ένα πάνω στον ένα και τον
16-δικη απεικόνισή και τον άλλο στην
ASCII. Αυτοί βοηθούν για να μπορούμε να

<θεθινόμε> σε κάποια διεύθυνση, ιδίως
όταν κάνουμε POKE με την αμέσως
επόμενη λειτουργία.

- <P> - POKE

Πάντα,ς το "P" μας ζητείται ένας
αριθμός (0-255), ο οποίος τοποθετεί-
ται στη μνήμη (RAM), στη θέση που
δείχνει ο δρομείας. Θα ήταν προτιμότερο
να έχουμε αποθήκευσι πρώτα το πρό-
γραμμα σε ένα δίσκο ή σε μια κάρτα και
μετά να αρχίσατε το POKE, τα οποία
είναι αρκετά επικίνδυνα.

- <P> ROM/RAM

Πάντα,ς το "R" μπορούμε να δούμε
την ROM, αν είμαστε στην RAM και
αντίστροφα. Η ενημέρωσή, για το αν
είμαστε στην ROM ή στην RAM γίνεται
στο κάτω δεξιά παράθυρο.

- <H> HEX/DEC

Στο ίδιο παράθυρο φαίνεται επίσης αν ο
αυξαν. αριθμός των θέσεων της μνήμης,
που δίνεται στην αριστερή όλη της
σθάνης, θα είναι σε δεκαδική ή δεκαεξα-
δική μορφή. Η εναλλαγή στα δυο
αριθμητικά συστήματα γίνεται με το
πλήκτρο "H".

- <G> GREEK/LATIN

Με το πλήκτρο "G", αντίστροφα,
μπορούμε να αποφασίσουμε αν θα
εχωρίμε απεικόνισή σε ελληνικούς ή
λατινικούς χαρακτήρες. Θα ήταν προτιμότερο
εξετάζοντας την περιοχή του
προγράμματος (στη RAM) να έχουμε τους
ελληνικούς χαρακτήρες, ενώ για την
ROM να χρησιμοποιήσε τους λατινικούς.

- <T> - ΤΕΛΟΣ

Αυτή είναι και η τελευταία λειτουργία.
Πάντα,ς το "T", το πρόγραμμα σταμα-
τά. Μπορείτε να έχετε είτε τους
ελληνικούς χαρακτήρες, αφήνοντας το
πρόγραμμα όπως είναι, ή τους λατινικούς,
προσθέτοντας την εντολή SYMBOL
AFTER & 62 πριν από το END στη γραμμή
170.

Μια ακόμη δυνατότητα που έχουμε είναι,
όταν κάνετε BREAK, (σταματήστε) το
πρόγραμμα. Μετά το BREAK, μπορείτε,
αφού κάνετε κάποιο ελεγχo, να γράψετε
"CONT" και το πρόγραμμα θα συνεχί-
σει ακριβώς από εκεί που άρχισε.

ΓΕΝΙΚΑ ΧΑΡΑΚΤΗΡΙΣΤΙΚΑ

Χαρακτηριστικό του προγράμματος είναι
το ότι δίνεται σε ανεξαρτητες κατά το
δυνατόν υπορουτινές, τις οποίες μπο-

ρείτε να χρησιμοποιήσατε και για άλλα
προγράμματα. Για το λόγο αυτό και για την
ευκολότερη κατανόησή του προγράμ-
ματος έχουν δοθεί αρκετά σχόλια μέσα
αυτό.

Οι ανεξαρτητες αυτές υπορουτινές
είναι:

α) Ορισμός Ελληνικών χαρακτήρων

Από γραμμή 1670 έως 1970.
Σαν DATA δίνονται ο αριθμός του
χαρακτήρα και το ακριβ. νούμερο που
αποστέλνει τη μορφή του, σε δεκαε-
ξαδικό. Μπορείτε να προσθέσετε τα
μικρά ελληνικά ή και τονωμένα.

β) Παρουσίαση Οδηγών

Από γραμμή 180 έως 400.
Ο ασπέριασος (!) στα DATA σημαίνει
αλλαγή γραμμής, και το νούμερο αμέ-
σως μετά, δείχνει τη θέση πάνω στην
γραμμή στην οποία θα συνεχίσει το
τυπωμα (ενο είδος TAB).

γ) Υπορουτίνα INPUT

Από γραμμή 1800 έως 1860.
Επιλέξι αν πατήσε κανείς BREAK κατά
τη διάρκεια της εκτέλεσεως της
εντολής INPUT, το πρόγραμμα θα
σταματήσει σ' έκθεο το σημείο και
δεν θα υποκυθεί η εντολή ON
BREAK GOSUB. Προτιμήθηκε η υπο-
ρουτίνα αυτή που μπορεί να χρησι-
μοποιεί και σ' έκθεο. Το αποτέλεσμα
δίνεται στο string a\$ και επίσης χρη-
σιμοποιείται σαν μεταβλητή η b\$. Η
μεταβλητή SX καθορίζει αν μετά το
INPUT θα συνεχίσουμε στην ίδια
γραμμή ή στην επόμενη.

γ) Μετατροπή αριθμών

Από γραμμή 1240 έως 1460.
Η υπορουτίνα αυτή θα πρέπει να
υπονοείται και από την εντολή
ON ERROR GOTO 1440, καθώς και
της εντολής 600 - 610 από την εισα-
γωγή μεταβλητών. Μπορείτε να γρά-
ψετε ξεχωριστά αυτήν την πραγματι-
κή χρήση υπορουτίνας για να την
έχετε για πρόχειρη χρήση.

δ) Διάβασο της ROM

Από γραμμή 410-450 (basic listing)
και γραμμές 10-110 (M/C listing).
Χρησιμοποιώντας τις υπορουτινές
\$B906 του λειτουργικού συστήματος
που ενεργούν στην απεικόνισή και
την κατάρτη ROM αντίστροφα, καθώς
και τη routine \$B90C που επισωρ-
ρίζει τη ROM στην προηγούμενη κα-
τάστασή τους, κάθε φορά που καλού-

ΘΕΜΑ

με τη διεύθυνση MEM (CALL MEM), η **ακέραια** μεταβλητή B παίρνει την τιμή της διεύθυνσης (στην ROM) που δείχνει η **ακέραια** μεταβλητή A. Δηλαδή $B = PEEK(A)$, μόνο που γίνεται διαβασμό της ROM και όχι της RAM. Αν θέλετε να χρησιμοποιήσετε τη ρουτίνα αυτή μην ξεχάσετε να ορίσετε πρώτα σαν ακέραια μεταβλητές την A και την B: DEFINT A,B. Το CALL αυτό θα μας δώσει σωστά αποτελέσματα μόνο αν εκτελείται μέσα από το πρόγραμμα.

Περισσότερες λεπτομέρειες θα καταλάβετε εξετάζοντας το Assembly listing, το οποίο συνοδεύεται και από αρκετά σχήμα.

7) Αποθήκευση και επαναφορά ελληνικών χαρακτήρων

Αφού κάνουμε Symbol alter και ορίσουμε τους ελληνικούς χαρακτήρες, καλούμε την υπορουτίνα στις γραμμές 130-310 της γλώσσας μηχανής η οποία αντιγράφει το σχήμα των χαρακτήρων, από την αρχική τους θέση σε ένα χώρο από τη διεύθυνση &B36 και πάνω (σχήμα 1).

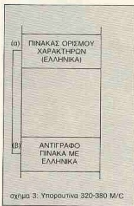


σχήμα 1: Υπορουτίνα 130-310 M/C
Για να ξαναπαίρουμε τους λατινικούς χαρακτήρες κάνουμε Symbol alter και το standard σχήματα των χαρακτήρων αντιγράφονται στη θέση (α) (σχήμα 2), από την ROM



σχήμα 2: Η εντολή Symbol Alter.

Καλώντας στη συνέχεια την υπορουτίνα του κώδικα μηχανής 320-380, οι ελληνικοί χαρακτήρες ξαναεπιγράφονται στον πίνακα ορισμού χαρακτήρων (character definition table), έχουμε δηλαδή ένα τους ελληνικούς χαρακτήρες (σχήμα 3).

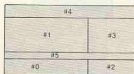


σχήμα 3: Υπορουτίνα 320-380 M/C

Η μέθοδος αυτή για τη διατήρηση των ελληνικών χαρακτήρων είναι απλή, έχει όμως το μειονέκτημα ότι απαιτεί παραγωγή μνήμη για να κρατήσει το αντίγραφο του πίνακα με το ελληνικό.

Στην αρχή του προγράμματος καλείται η υπορουτίνα &B302, η οποία επαναφέρει τα χρώματα, όπως ακριβώς είναι όταν ανοίγουμε τον υπολογιστή (RESET INKS). Μια άλλη ρουτίνα που χρησιμοποιήθηκε είναι και η CALL &B306. Αυτή στοιβάει τη λειτουργία του προγράμματος και περιμένει μέχρις ότου πατηθεί κάποιο πλήκτρο. Ο κωδικός ASCII του πλήκτρου που πατηθήκε βρίσκεται στον Accumulator. Για όσους αναρωτούνται πως κατασκευάστηκε ο δρομέας, υπενθυμίζουμε ότι αν έχουμε κάνει PRINT CHR\$(23)+CHR\$(1), δηλαδή βρισκόμαστε στο XOR GRAPHICS MODE, σχεδιάζοντας ένα σχήμα δυο φορές στην ίδια θέση, το σχήμα εμφανίζεται, και ο.τι υπήρχε πριν απ' αυτό επανέρχεται, σαν να μην είχαμε ζωγραφίσει, παρά τίποτα, στο συγκεκριμένο σημείο.

Τα εξ. παραθύρα που χρησιμοποιούνται από το πρόγραμμα, φρονόται στο σχήμα 4. Πιστεύω λοιπόν πως θα βρείτε αρκετά χρώματα το «neo» MONITOR του υπολογιστή σας και θα μάθετε αρκετά μυστικά της μηχανής του.



σχ. 4

BEMA

```

220 READ A$:FOR I=0 TO 60:NEXT:VRND: SOUND 3,400,20,4:CALL $BDSO:RETURN
230 DATA *.14,EJETACh,MXHMh,KAl,METATrOpH,ArIoMvE,gIA,TOX,AMSTRAD,t.0
240 DATA *.12,TO,POgrAKKA,AYTO,uA,eAc,BDHuEEl,NA,EJETAeEIE,TH,MXHMh
250 DATA *.27,TOX,AMSTRAD, *KAM,KAl,ROM
260 DATA *.7,H.pArOYrIAch,TvM,pErIBXOMEVvN,THh,MXHMh,uA,gIENI,eE,MOCrH,eHl
IdvN
270 DATA *.8,(ApO O,Eva,255),KAcS,eElIdA,EXE1,250,BYTES,IA OpOIA,dINOMTAI,e
B
280 DATA *.14,dSKAFJAdIKH,MOCrH,KAl,eE,ABCI1,ApEIKONIeH,eYgIXONvS
290 DATA *.26,RAM,"",0-255,eElIdA,*.26,ROM,"",0-63,KAl,192-255,eElIdA
300 DATA *.11,eAc,dIETA1,EplEhE,H,dYNAIOTRTA,METATrOpH,ArIoMvE,ApO,EVA
310 DATA *.10,ArIoMUTIKO,eYTHKA,eE,EVA,AlIO,(GyADIKO,Eva,36-dIXO,eYTHKA)
320 DATA *.23,OI,dIAeEeINhE,LEITOrgIhE,EINAl,"",t.0
330 DATA *.12,"ISPACE1",-,pOXv:RdE,eIhE,EpOMENh,eElIdA
340 DATA *.12,"[AI]",-,AlIAJE,eElIdA
350 DATA *.12,"[MI]",-,METATrOpE,ArIoMvE,*.12,"[BB1H]",-,KINhEh,dROMEa
360 DATA *.12,"[P]",-,POKE,(AlAgH,THh,dIEYvYhEh,pOY,dEIXvE1,O,dROMEa
)
370 DATA *.12,"[R]",-,AlIAgH,ApO,ROM,eE,RAM,KAl,ANTIETrOIA
380 DATA *.12,"[H]",-,AlIAgH,ApEIKONIeH,TvM,dIEYvYhEhVvE,ApO,HEX,eE,DEC
390 DATA *.12,"[OI]",-,AlIAgH,ApO,EIhNIKA,eE,IATINIKa,KAl,ANTIETrOIA
400 DATA *.12,"[T]",-,TRIOw,*.0,t.30,PATH=5,EVA,plHEITrO,SED
410 REM --EIAgYgH gIvAcE MNXANhE gIA FHEK ApO ROM --
420 MNX=36000:RESTORE 450:MEMORY MEM:1:FOR I=MEM TO MEM+19:GOSUB 710
430 FHEK:A=FA MOD 256:POKE MNX+1,A:POKE MNX+2,(6A-A)/256
440 B=EB MOD 256:POKE MNX+12,B:POKE MNX+13,(6B-B)/256
450 DATA 2A,80,00,CD,00,89,FE,CD,06,59,7E,32,06,00,FI,CE,0C,89,C9
460 REM --gIA XARAKTHrE EIHNIKA/LATIN--
470 MEM1=36050:RESTORE 460:FOR I=MEM1 TO MEM1+31:GOSUB 510:NEXT
480 DATA CD,AE,BB,FE,AF,21,00,01,4F,06,00,ED,42,29,29,29,5C,E1,E1,ED,45,FA,
8C,2E,FC,8C,11,36,9D,ED,80,C9
490 MEM2=36100:RESTORE 500:FOR I=MEM2 TO MEM2+13:GOSUB 510:NEXT:CALL MEM1
500 RETURN:DATA ED,4E,FA,8C,ED,5E,FC,8C,21,36,8D,ED,80,C9
510 READ B$:POKE I,VAL(MID$(B$,I))
520 REM --EIAgYgH pArAMTrvE --
530 DEFINT A,B,V:DIM VALUE(IG,IS):ORDER=1:CS44+CHR$(24)+HEX=1:MS= ""
540 USERS=CHR$(240)+CHR$(241)+CHR$(243)+CHR$(242)+$FA MRHDT" FOR I=0 TO 15
550 MS=MS+HEX$(I)+" " :NEXT:MS="$ MHHMH "$+MS+SPACES(16)+"ACC11"+SPACE
1(7)
560 REM -- gIA dROME --
570 PRINT CHR$(23)+CHR$(1):VER=32:HOR=24:HORLINE=9:VERLINE=3:X=SORLINE#6-5
580 Y=409:VERLINE#15
590 REM --gIA METATrOpH--
600 RESTORE 610:READ A18,A28,A38,A48:C348=CHR$(34)
610 DATA METATrOpH ArIoMvE, ** ApO 2 Eva 36 MONO **, "ApO BAcN", "eE BAcN: "
620 REM --- pArAyTrA, TrvKATA. ---
630 INK 0,26,INK 1,0:MODE 2:BORDER 10:BI=3:H2=20:V1=66:PEM 0:PAPER 1
640 WINDOW 1,V1,H2+1,26:WINDOW#1,1,V1,BI+1,H2-1:CLS:CLS#1:PEM#1,1:PAPER#1,0
650 WINDOW#2,V1+1,80,H2+1,35:PEM#2,0:PAPER#2,1:CLS#2:WINDOW#3,V1+1,80,BI+1,
H2-1
660 CLS#3:WINDOW#4,1,80,1,H1:CLS#4:WINDOW#5,1,80,H2,H2:PEM#5,0:PAPER#5,1:RE
TURN
670 REM ----- ROM/RAM,eElIdA -----
680 PRINT CHR$(7):B=1:CLS#H:PRINT#H,"uHIEh NA EJETAeIEh IHh:"
690 PRINT#H,TAB(26),"15 ROM":PRINT#H,TAB(26),"2) RAM"
700 PRINT#E:PRINT#H,TAB(26),"dIAIEIEh":GOSUB 1610:--- INPUT ---
710 IF AS="1" THEN ROW=1 ELSE IF AS="2" THEN ROW=0 ELSE 660
720 PRINT#K:PRINT#H,"pOIA eElIdA 17":GOSUB 1610:--- INPUT ---
730 PAGE=VAL(O"4A"):IF PAGE>255 THEN ROW#1:PRINT CHR$(7):COTO 720
740 GOSUB 1990:IF LB=1 THEN 720 ELSE CLS#H:RETURN
750 REM ----- MHHMh -----
760 LOCATE#4,1,3:PRINT#4,C244+M#3+C244:PRINT#5,M#4:SP#5=SPACES(7)
770 LOCATE#4,19,1
780 PRINT#4,R#4" eElIdA ApO"+SP#5+"Eva"+SP#5+"(4EKAdIKO)"
790 WHILE TELDSC<3:M=PAGE+256:GOSUB 1780:GOSUB 1670:OdHgIh#6<

```

BEMA

```

800 FOR I=M TO 65535 STEP 256
810 IF GEEK=0 THEN MEMORY HMN:SYMBOL AFTER 662:MEMORY M=1
820 GOSUB 870:--- QUONH ---
830 CALL MDG:GOSUB 960:--- XrMThs ---
840 IF PG THEN 850 ELSE NEXT:PAGE=0
850 WEND:RETURN
860 REM ----- QUONH -----
870 CLS#1:CLS#3:GOSUB 1780:---R(O/A)M---
880 LOCATE#4,16,1:PRINT#4,RS:LOCATE#4,29,1:PRINT#4,USING"###"PAGE
890 LOCATE#4,37,1:PRINT#4,USING"####"(1:LOCATE#4,47,1
900 PRINT#4,US:NG"####":I+255:1=0:FOR J=1 TO I+255 STEP 16:PRINT#3,TAB(4)
910 IF HEX THEN PRINT#1," "A"+HEX$(J,4)+SPACES(3): ELSE PRINT#1," "USING"##
### "J:
920 FOR K=0 TO 15:A=INT(J+K):IF ROM THEN CALL NEW ELSE B=PEEK(A)
930 VALUE(L,K)=B:PRINT#1,HEX$(A,2)I" "GOSUB 1740:---PRINT CHR--
940 NEXT K:PRINT#1:PRINT#3:L=L+1:NEXT I:PAGE=PAGE+1:RETURN
950 REM ----- XrMThs -----
960 TELCO=0:CK=0:CV=0:PG=0:WHILE TELOS=>0:GOSUB 1630:---dGOMEAs---
970 B=UPPER$(INKEY$):IF B=""THEN 970 ELSE GOSUB 1690:---dGOMEAs---
980 ON INSTR(USERS,B) GOSUB 1000,1020,1050,1030,1100,1190,1220,1260,1480,1
990,1600,1610:WEND:RETURN
1000 REM ----- pANv -----
1000 CY=CY-1 AND CY>1:RETURN
1010 REM ----- KATv -----
1020 CV=CV+1:IF CV>15 THEN CV=15
1030 RETURN
1040 REM ----- dEJIA -----
1050 CX=CX+1:IF CX>15 THEN CX=15
1060 RETURN
1070 REM ----- ArIeTBra -----
1080 CX=CX-1 AND CX>1:RETURN
1090 REM ----- POKe -----
1100 IF ROM=1 THEN PRINT CHR$(7):RETURN
1110 CLM=B=0:PRINT:PRINT"POKE":I+CV+16+CX:"":GOSUB 1810:---INPUT--
1120 A=VAL(A$):IF A<0 OR A>255 THEN 1110
1130 VALUE(CY,CX)=A:POKE I+CV+16+CX,A:LOCATE#1,10+CX#3,CY+1
1140 PRINT#1,HEX$(A,2):PRINT#3,CHR$(5)
1150 GOSUB 1740:---PRINT CHR--
1160 GOSUB 1670:---dGngIEa=0--
1170 RETURN
1180 REM ----- AllAjE -----
1190 CLS:PRINT:PRINT"dSIjE THN dSIIDA":N=0:GOSUB 1810:PAGE=VAL("0"+A$):T
MPST
1200 PG=1
1210 REM ----- EjoMENN -----
1220 H=0:GOSUB 1690:IF LP=1 OR PAGE>255 THEN CALL 8800:GOTO 1190
1230 TELOS=-1:RETURN
1240 REM ----- METATrOpH ArIuMvS ----
1250 MA=0:BOHDEE 15:CLS:CLS#2:H=0:PRINT A1$:PRINT A3$:
1260 GOSUB 1810:---INPUT--
1270 A=VAL("0"+A$):IF A<2 OR A>36 THEN PRINT A2$:GOTO 1260
1280 IF MA=0 THEN B1=A:MA=1:PRINT A4$:GOTO 1260 ELSE B2=A:CLS:PRINT#2,A1$
1290 PRINT#2,A3$,B1:PRINT#2,A4$,B2:PRINT#2,"" TELo METATrOpHs"
1300 PRINT"dveE ArIuMvS":WHILE 1
1310 SX=1:GOSUB 1810:---INPUT--
1320 IF AS=""THEN BORDER 10:CLS#2:GOSUB 1670:RETURN
1330 T=0:FS=0:AS=UPPER(AS):FOR C=1 TO LEN(AS):B$=MID$(AS,C,1):E=ASC(B$)
1340 IF E<48 OR E>90 OR E=95 AND E<65 THEN PRINT" TO "C34$:B$:C34$:" dEM B1
NAI A1ArIuMHTIKO":GOTO 1310
1350 IF E<65 THEN B=E-48 ELSE B=E-95
1360 IF E=B1 THEN PRINT" TO ArIuMHTIKO "C34$:B$:C34$:" BINA1 pOly MEGAlO":
GOTO 1310
1370 IF T(999999999-E)/B1 THEN FS=FS+1 ELSE FS=0
1380 IF FS>3 THEN ERROR 5
1390 T=T+B1-E:NEXT C:B1$=""
1400 IF E=10 THEN E=E+55 ELSE B=E+48
1410 B1$=CHR$(E)+B1$:WEND:PRINT":B1$:IF B1$=""THEN PRINT"0":

```

BEMA

```

1420 IF B8<>0 THEN PRINT "6Tr0g5Y1EWKNO:" ELSE PRINT
1430 WEND
1440 IF BXL=1360 OR BRL=1300 OR BRL=1410 THEN PRINT"TO SOYKH+O BINA: POLY M
EgAIO":RESUME 1310
1450 IF BBR=5 AND BRL=1880 THEN RESUME 90
1460 PRINT"1AU0e":ERR:"THN grAMMH":BRL:END
1470 BEM ----- ROM/RAK -----
1480 ROM=ROM+1 AND ROM(1):GOSUB 1780'-R(C/A)M-
1490 LOCATE#2,8,1:PRINT#2,C24#+" "+R#+" "+C24#:RETURN
1500 BEM ----- HEX/DIC -----
1510 HEX=HEX+1 AND HEX<1
1520 LOCATE#2,8,3:PRINT#2,C24#," "":
1530 IF HEX=0 THEN PRINT#2,"DEC": ELSE PRINT#2,"HEX":
1540 PRINT#2," "":C24#,:RETURN
1550 BEM ----- ELIHNKA/LATIN -----
1560 ORREK=ORREK+1 AND ORREK<1
1570 LOCATE#2,8,5:PRINT#2,C24#
1580 IF ORREK=1 THEN PRINT#2,"ELIHNKA": ELSE PRINT#2,"LATINKA":
1590 PRINT#2,C24#,:RETURN
1600 BEM ----- TBICs -----
1610 TELCS--3:PG--1:RETURN
1620 NAM dCUMAs
1630 MOVE X+CX*HOR,Y-CY*VER/2:DRAWV 0,-VER:DRAWV HOR,0:DRAWV 0,VER:DRAWV -H
OW,0
1640 LOCATE#3,4,CX,CY+1:B=VALUE(CY,CX):PRINT#3,C24#,:GOSUB 1740'-PRINT CHE-
1650 RETURN
1660 BEM ----- OJgJBe STO #0 -----

```

```

1670 CLG:PRINT
1680 PRINT TAB(4):"1SPACH1 - PpOMNH"+SPACES(10)+1A1 - AllAJE =E11dA"
1690 PRINT TAB(4):" 1K1 - METATrOpH"+SPACES(8)+1P1 - POKB"
1700 PRINT TAB(4):" 1R1 - ROM/RAK"+SPACES(10)+1H1 - HEX/DEC"
1710 PRINT TAB(4):" 1G1 - ELIHN/LATIN"+SPACES(6)+1T1 - TBICs"
1720 GOSUB 1490:GOSUB 1520:GOSUB 1570:RETURN
1730 BEM ----- PRINT CHR -----
1740 IF B=0 THEN PRINT#3," "":RETURN
1750 IF B<32 THEN PRINT#3,CHR$(B)
1760 PRINT#3,CHR$(B):RETURN
1770 BEM ----- R(C/A)M -----
1780 IF ROM=0 THEN R#="RAK" ELSE R#="ROM"
1790 RETURN
1800 BEM ----- IPUY -----
1810 A#="" WHILE 1:R#="" WHILE B#<CHR$(127):PRINT#H,B#,CHR$(143):CHR$(8)
1820 A#=#A#B#B1:="" WHILE B#="" B#=#KBY#WEND:IF ASC(B#)=13 THEN 1950
1830 WEND:IF A#="" THEN 1840 ELSE PRINT#H," "":CHR$(8):CHR$(8):A#=#RPT$(A#,L
EN(A#)-1)
1840 WEND
1850 PRINT#H," "":IF EX<1 THEN PRINT#H
1860 EX=0:RETURN
1870 BEM --- E11HN/NO! 1ARAKH:Ba ---
1880 DIM A(8):RESTORE 1920:SYMBOL AFTER #32:RME=HIMEM
1890 WHILE I1J=0:READ A#:#OR 1+1 TO 17:STEP 2:IF A#="" THEN ERASE A:RETR
E
1900 A(1)=VAL("A"+MID$(A#,1,2)):J=J+1:FRY
1910 SYMBOL A(0),A(1),A(2),A(3),A(4),A(5),A(6),A(7),A(8):WEND
1920 DATA 677800000000000000,6418300000000000
1930 DATA 7930000007E000000,8C18300000000000
1940 DATA 6AHEB2000380000FE00,70FB0C0C0C0C0C00
1950 DATA 727C000007C00000000,70FB0C0C0C0C0C00
1960 DATA 66107CD0D0D07C1000,82000000007C101000
1970 DATA 707C0C0C0C0C0000000,1
1980 BEM ----- KHSYMA 1AU0e -----
1990 IF ROM=1 AND PAGE#83 AND PAGE#192 THEN PRINT#H,"E1A1 @THN ROM: (0-63 K
A: 192-255 sBLIGE#)":LP=1 ELSE LP=0
2000 RETURN

```