

Karakterisztikák alkalmazása és értelmezése

A Technika I. tankönyv megfogalmazása szerint: „A rendszer karakterisztikája olyan összefüggés, amely megmutatja... Igen megmutatja a be- és kimenő jel közötti összefüggést. Mint összefüggés és

mint koordináta-rendszerben ábrázolt mennyiség sugall valamilyen függvénykapcsolatot. A technikai gyakorlatban — a hétköznapi élet számára — azonban nem elsődlegesen a matematikai össze-

függés a fontos, hanem az éppen figyelembe vett két mennyiség (pl. a be- és kimenő jel) alakulásának jellege. Nagyon gyakran előbb ismert a — sok mérés segítségével felrajzolt — jelleggörbe, és annak ismeretében írható fel az összetartozó mennyiségek közötti matematikailag megfogalmazott függvénykapcsolat.

Az 1. program a Feladatgyűjtemény 1.6 feladatához kapcsolódik. Lehetővé teszi a feladatban mért feszültség- és áramerősség-értékek beírását. Elvégzi az ellenállás kiszámítását, és a mért, illetve számított értékeket azonnal táblázatos formában közli. Az adatok közlése maximum 9 adatig, vagy a feszültség mért értékeként 0 beírásáig lehetséges. Az utolsó mérési adat beírása után a program (a feladatlapp előírása szerint) léptékhelyesen megrajzolja a jelleggörbét. (Most már csak az lenne szebb, ha ezt a mérési jegyzőkönyvet ki is nyomtatná a program, de annak sincs sem elvi, sem gyakorlati akadálya, hogy a mérési adatokat a számítógép periferián keresztül gyűjtse. Például technoMIR interfész alkalmazásával.)

A házi vízellátó berendezésekben a mezőgazdaság öntözőhálózatokban a folyadék energiátartalmának növelésére centrifugálszivattyút használnak. A centrifugálszivattyúk jelleggörbéje nem lineáris. Az üzemi jellemzőket (a hatások karakterisztikájának ismeretében) a szivattyú gépkönyvében található térfogatáram függvényében a szállítási magasságot ábrázoló jelleggörbéből lehet megállapítani. A 2. program ezt mutatja be. Az összetar-

tozó értékeket az MA—20 jelű szivattyúra a 60—70 DATA sorban tároltuk. A jelleggörbe alakja sok szerkezeti és üzemi körülménytől függ. A program lehetővé teszi a szivattyú perccenkénti 3000-es fordulatszáma mellett felrajzolt jelleggörbéje mellé más INPUT-ként közölhető fordulatszám-értékhez tartozó jelleggörbe ábrázolását. A program a $V1/V2 = n1/n2$ és $H1/H2 = (n1/n2)^2$ értékek kiszámítását és az összetartozó értékek léptékhelyes ábrázolását teszi lehetővé. Ez a jelleggörbe a lehetőség oldaláról mutatja be a szivattyút. A hálózat igénye szintén ábrázolható. A két jelleggörbe (a lehetőség és igény) találkozási (metszéspontja) kijelöli a két illesztett rendszer közös munkapontját.

A 3. program az előző szivattyú jelleggörbéjét felhasználva a hálózatot jellemzőinek INPUT-ként való közlése után egymásra rajzolja a két (részben számított) jelleggörbét, és így kijelöli a munkapontot (ha az a közölt adatok alapján egyáltalán lehetséges). Hasonlóan az adott teljesítményű erőgép (traktor) vonzóerő jelleggörbéjéből (ez itt hiperbola) leolvasható az adott sebesség mellett kifejezhető vonzóerő.

A 4. program a vonóhorgon kifejezhető maximális teljesítmény- és haladási sebesség értékekből felrajzolja a vonóerő hiperbolát (karakterisztikát). A kifejezhető vonóerő és sebesség (igény) INPUT-adatok alapján az üzemi pontot kijelöli a képernyőn, és lehetőséget ad a kiértékelésre.

```
10 '1.6 feladat FELADATGYUJTEMENY.OPI 1981 Szerk.Ujvari K.
20 CLS:K$="#.##":L$="####":ADATKOZLES.U=0-val bef.
30 FORI=0TO127:SET(I,0):SET(I,6):SET(I,12):SET(I,18):NEXT
40 FORI=17TO127STEP10:FORI=0TO18:SET(I,Y):NEXT:NEXT
50 PRINT@65,"U"(V):PRINT@193,"I"(A):PRINT@321,"R (Ohm)":
60 S=1:D=9:Z=1.
70 PRINT@640," ":INPUT"U=":U(Z):IFU(Z)>=6THEN70
80 IFU(Z)<=0THENZ=Z-1:GOTO140
90 PRINT@644*S+D,USINGK$:U(Z):
100 PRINT@640," ":INPUT"I=":I(Z):IFI(Z)>=100RI(Z)<=0THEN100
110 PRINT@644*(S+2)+D,USINGK$:I(Z):
120 R(Z)=U(Z)/I(Z):PRINT@644*(S+4)+D,USINGL$:R(Z):
130 Z=Z+1:D=0+S:IFZ=10THENZ=9:GOTO140ELSEGOTO70
140 UM=0:UN=10:RM=0:RN=100000
150 FORI=1TOZ
160 IFU(I)>UM THEN UM=U(I) ELSEIFU(I)<UN THEN UN=U(I)
170 IFR(I)>RM THEN RM=R(I) ELSEIFR(I)<RN THEN RN=R(I)
180 NEXT
190 PRINT@640," ":PRINT:PRINT:PRINT
200 FORI=20TO43:SET(9,I):NEXT I:FORI=2TO93:SET(1,43):NEXT
210 PRINT@451,"R":PRINT@832,USINGL$:RN:PRINT@512,USINGL$:RM:
220 PRINT@964,"0 1 2 3 4 5 U":
230 RA=(RM-RN)/15:UA=(UM)/70
240 FORI=1TOZ:SET(9+U(I)/UA,40-(R(I)-RN)/RA):NEXT
250 PRINT@1010,"UJRA 'T'":IFINKEY$<"T"THEN250ELSESERUN
```

1. program

```
10 REM SZIVATTYU JELLEGGORBE
20 CLS:DIMH(20),V(20):N=3000
30 FORI=0TO40:SET(11,I):SET(11+I,40):NEXT
40 PRINT@4,"H":PRINT@858,"Q":
50 PRINT@129,"70":PRINT@849,"20":
60 DATA34,35,36,36,36,36,35,34,33,32
70 DATA31,30,28,26,25,22,15,10,2
80 FORI=0TO18:READH:H(I)=H:NEXT
90 FORI=0TO18:V(I)=2*I:NEXT
100 FORI=0TO18:SET(12+V(I),40-H(I)):NEXT
110 PRINT@210,"n=3000":
120 PRINT@890," ":INPUT"n>n2 n2=":M:
130 IFM>NTHEN120ELSEIFM<100THEN170
140 FORI=0TO18
150 SET(12+V(I)*M/N,40-H(I)*M*M/N/N)
160 NEXT:GOTO120
170 END
```

2. program

```
10 REM SZIVATTYU JGG.MUNKAPONT
20 ONERRORGOTO200
30 CLS:DIMH(20),V(20):N=3000
40 FORI=0TO40:SET(11,I):SET(11+I,40):NEXT
50 PRINT@1,"H m":PRINT@858,"Q 1/2":
60 PRINT@129,"70":PRINT@849,"20":
70 DATA34,35,36,36,36,36,35,34,33,32
80 DATA31,30,28,26,25,22,15,10,2
90 FORI=0TO18:READH:H(I)=H:NEXT
100 FORI=0TO18:V(I)=2*I:NEXT
110 FORI=0TO18:SET(12+V(I),40-H(I)):NEXT
120 PRINT@896,"hp+hq (m)=":INPUTH1
130 FORI=12TO16:SET(1,40-H1/2):NEXT
140 PRINT@896,"CS6 ATMERO (m)=":INPUTD
150 PRINT@896,"CS6 HOSSZA (m)=":INPUTL
160 K=0.03*L/D/20000
170 FORI=0TO42STEP2
180 SET(I+12,40-(H1+K*I*I)/2):NEXT
190 PRINT@890," ":END
200 PRINT@896,CHR$(220):
210 RESUME NEXT
```

3. program

```
10 CLS:REM P=F*v diagram
20 INPUT"TELJESITMENY (W) =" :P
30 INPUT"MAX. SEBESSAG (m/s) =" :V
40 ONERRORGOTO190:CLS:A=40/V:F=7.5/P
50 PRINT@148,"P=":P/1000:"kW":
60 FORI=0TO38:SET(10,I):SET(11+I,38):NEXT
70 PRINT@4,"F":PRINT@858,"v":
80 PRINT@836+A/2,"1":PRINT@855,V:
90 PRINT@INT((38-P*F)/3)*64,P:
100 FORI=0TOVSTEPV/20
110 SET(10+I*A,38-P/I*F):NEXT
120 PRINT@890," ":INPUT"F=":E
130 PRINT@890," ":INPUT"v=":V
140 FORU=1TO10:SET(10+V*A,38-E*F)
150 FORT=1TO30:NEXT:RESET(10+V*A,38-E*F)
160 FORT=1TO30:NEXT:NEXT
170 PRINT@274,"F*v=":E*V/1000:"kW
180 GOTO120
190 RESUMENEXT
```

4. program

Tuning-4

A HT felhasználóinak egyik táborában kapcsolódás után azonnal meghívja a bővítést (mások meg idegenkednek tőle). Meghívni bármikor meghívhatjuk a bővítést, annak különböző szolgáltatásait. Sokszor viszont jó lenne visszatérni az alap BASIC szerényebb szolgáltatásaihoz, főleg akkor ha játék- vagy rajzprogramunk futtatása során a billentyűzetet kell figyelnie az Interpreternek. Az állapotra való visszaállítás brutális módja lehetséges, de ilyenkor a programot elveszítjük.

Az itt közölt kis gépi kódú rutin (5. program) a KILL kulcsszó alkalmazását teszi lehetővé, segítségével „kírhathatjuk” a bővítést anélkül, hogy a gépben lévő program és adatok elvesznének. A SYSTEM/12288 segítségével a bővítést a korábbi gyakorlatnak megfelelően továbbra is bármikor meghívhatjuk. Ez a tuningolás is a gép kikapcsolásáig él a többi funkció és a memória méretének zavarása nélkül.

```
10 ' KILL bovitestbol vissza alap BASIC-be
20 POKE16785,195:POKE16786,76:POKE16787,64
30 FOR I=16460TO16476:READ A:POKE I,A:NEXT
40 DATA 33,214,6,17,4,64,1,30,0,237,176,
205,201,1,195,24,26
50 NEW (c) Felföldi J. - Sz. Bukacs J.
```

5. program

Méretezés a technikaórán

A számítógép — egy jól felállított méretezési algoritmus segítségével — igen gyorsan kiszámítja az ismert adatokból a képletek felhasználásával a szükséges értékeket.

Elég bonyolult számításhoz kell lennie annak — vagy nagyon sokszor kell elvégezni azt —, amit érdemes számítógéppel végezteni. Az olyan méretezés esetében viszont, amelyben nem konkrét is-

mert, hanem több lehetséges adat felhasználásával kell végeredményhez jutni, vagy olyan esetekben, amelyekben a számítás során kapott részeredmények módosítják a kiindulási feltételeket, ott még az órákózi feladatmegoldásokhoz is érdemes elkészíteni, illetve használni a megfelelő programot.

Az 1. program transzformátor méretezéséhez ad segítséget. A feladatgyűjteményben egy max. 20 leágazású, 500 VA teljesítményű transzformátor méretezése a feladat. A méretezéshez a szükséges szekunderfeszültség értékeket kell közölni a programmal. Itt az adatközlést nulla feszültségigény beírásával fejezzük be.

A 10.—12. programsorok az esetleg észszelvi beírt értékeket rendezik növekvő sorrendbe. A szükséges határfokot és tapasztalati értékeket a 20.—23. sorok olvassák ki a táblázatból. A 34.—36. sorok biztosítják a biztonság javára a nagyobb vasmagkeresztmetszet kiválasztását. A transzformátor elkészítéséhez szükséges adatokat pedig a 44.—50. sorok között írja ki a program a képernyőre.

Tán érdemes ezt a programot használni, hiszen a technika óra keretében kívánjuk elkészíteni a transzformátort. A program segítségével igen hamar

hozzájuthatunk a gyártáshoz szükséges adatokhoz.

Egy tartó méretezése során más jellemző feladattal találjuk magunkat szemben. A kiinduláskor még nem ismerjük a szerkezet, a tartó önsúlyát, esetenként az nagyobb lehet, mint a hasznos terhelés.

A 2. program leegyszerűsítve mutatja be a hajlításra igénybevevett, önsúlyával terhelt tartó méretezését.

A 230. soron határoztuk be azt, hogy meddig folytassa a számítás finomítását a program.

Az alapfeladathoz a tartó anyagára jellemző szilárdsági adatot, a terhelést és a támaszközt kell közölni a géppel. A program minden egyes számítás után kiírja a figyelembe vett terhelésértéket és az ebből adódó szükséges átmérőt.

A program egyszerűsége (a feladatleegyszerűsítése ellenére) szemléletformáló lehet. A többszöri, különböző adatokkal történő programfuttatás során kitűnnek azok a tartományok, nagyságrendek, melyek esetében az iteráció nélkülözhetetlen, de fellelmezhető az is, hogy a módszer alkalmazása nem feltétlenül szükséges minden esetben.

Sz. Lukács János

```

1 'TRANSZFORMATOR MERETEZES
2 Zorad:Gyak.feladatok a technika orakon
3 CLEAR100:DIMNR(20),L(20),D(20),U(20)
4 CLS:Z=1:PRINT"TRANSZFORMATOR MERETEZES"
5 INPUT"PRIMER FESZ. Up=":UP
6 PRINT"ADATKOZLES VEGE HA A Usz=0"
7 IFZ>20THENPRINT"TUL SOK LEAGAZAS":END
8 PRINTZ:"SEKUNDER FESZ.=":INPUTU(Z)
9 IFU(Z)<=0,Z=Z-1:GOTO9ELSEZ=Z+1:GOTO6
10 C=0:M=Z:IFM=1THENUM=U(1):GOTO13
11 FORI=1TOZ-1
12 IFU(I)>U(I+1)THENC=1:UT=U(I+1):
    U(I+1)=U(I):U(I)=UT
13 NEXT:IFC=1THENGOTO9 ELSE UM=U(I)
14 INPUT"MAXIMALIS ARAMEROSSEG I=":I
15 IFI<=0THENGOTO13
16 PP=UM*I:PM=PP
17 IFPM>500THENPRINT"500 VA-nal NAGYOBB
    TELJESITMENY ERRE A PR.NEM JO !":END
18 DATA10,0.5,0.75,0.5,50,0.6,0.85,0.3
19 DATA100,0.75,0.92,0.12,500,0.85,1,.1
20 FORZ=1TO4:READP(Z),E(Z),G(Z),C(Z):NEXT
21 IF M=1 AND P(Z)>=PM THEN E = E(Z+1):
    G = G(Z+1) : C = C(Z+1)
22 IF P(Z)<=PM AND P(Z+1)>PM , E=E(Z+1):
    G=G(Z+1):C=C(Z+1)
23 NEXT
24 PP=PP/E:Q=G*SQR(PP) : CLS
25 PRINT"Pmax=":PP;"VA","Q=":Q;"cm2"
26 DATA 42/14,48/16,54/18,60/20,66/22,
    78/26,84/28,92K/30,106K/34
27 DATA 84/42,130/36,92K/42,130/46,
    106L/47,150/40,150/50
28 DATA 150/60,40,45,51,57,63,74,80,85,
    97,120,100,120,131,134
29 DATA 114,143,171,1.86,2.43,3.08,3.8,
    4.6,6.43,7.44,8.55,10.9
30 DATA 11.4,11.62,11.98,13.1,13.4,
    15.2,19,22.8
31 FORZ=1TO17:READNR(Z):NEXT
32 FORZ=1TO17:READD(Z):NEXT
33 FORZ=1TO17:READL(Z):NEXT
34 FORZ=1TO17
35 IFL(Z)<=QTHENJ=Z+1
36 NEXT
37 PRINT"VASMAG TIPUSA: EI ";NR(J)
38 PRINT"TENYLEGES KER.M.=":L(J);"cm2";
    " ";D(J);"DB 0.35 mm-ES LEMEZ"
39 N=45*UP/L(J) : PRINT : NS=(1+C)*N/UP
40 PRINT"PRIMER MENETSZAM=":INT(N)
41 PRINT"MENET/VOLT =":INT(10*NS+0.5)/10
42 KR="UH#=# V LEAGAZAS:####.ME"
43 KR=KR+"NETNEL Ueff=####.V"
44 FORZ=1TOM
45 NN=U(Z)*NS:UN=UP*NN/N
46 PRINTUSINGKR;Z,U(Z),NN,UN
47 NEXT : IP = PM/UP
48 DP=SQR(0.5*IP):DP=INT(10*DP+.5)/10
49 DS=SQR(0.5*I):DS=INT(10*DS+0.5)/10
50 PRINT"Dpr=":DP;"mm","Dssek=":DS;"mm"
51 IF INKEY#="" THEN 51 ELSE RUN

```

1. program

```

10 CLS:PRINT:PRINT"ITERACIO":PRINT:PRINT
20 PRINT"KONCENTRALT ERÖVEL TERHELT"
30 PRINT"KÖRKERESZTMETSZETÜ "
40 PRINT"KETTAMASZU TARTO"
50 PRINT:PRINT"ÖNSÜLYANAK FIGYELEMBEVÉTELE"
60 PRINT:PRINT"A TARTO ANYAGARA MEGENGEDETT"
70 INPUT"FESZULTSEG N/mm²=":SZ
80 CLS:FORI=2TO39:SET(I,11):NEXT
90 PRINT#257,"A":PRINT#275,"0":
100 GOSUB250:INPUT"A TAMASZKÖZ (m)=":L
110 GOSUB250:PRINT#278,"L=":L;"m":
120 K=L/2:S=20:FORI=1TO9:SET(S,I):NEXT
130 FORI=8TO10:SET(10+I,I):SET(30-I,I):NEXT
140 PRINT#65+(S/2),"F=":
150 GOSUB250:INPUT"AZ ERÖ (N)=":F
160 GOSUB250:PRINT#65+(S/2),"F=":F;"N":
170 PRINT#448,"F+J (N) M (Nm) D (mm)"
180 J=0:KQ="#####"#####"###.#"
190 T=F+J:DU=D:M=F*L/4+J*L/8
200 KX=M*1000/SZ:D=(0.0982*KX)^(1/3)
210 PRINTUSINGKR;T,M,D
220 J=D*D*3.14*L*78/4000
230 IFD=DU>.5THEN190 ELSEPRINTTAB(23)"-----"
240 IFINKEY#=""THEN240ELSERUN
250 PRINT#832,"":RETURN

```

2. program

Tuning-5

A programokban gyakran tárolunk DATA sorban értékeket. Ezek lehetnek egy gépi kódú program vagy egy ábra adatai. Ezeket általában READ után a programból egy megadott címtől kezdődő memóriaterületre pakoljuk (POKE) át. A kiolvasás és a kívánt helyre való áttöltést FOR...NEXT ciklus segítségével hajtjuk végre.

Az itt látható segédprogram (3. program) a PUT (alap BASIC-ben nem értel-

mezett utasítás) kulcsszónak olyan értelmezést ad, amely mintegy összevonja a ciklust, a kiolvasást és a POKE utasítást.

A PUT címdec, adat, adat, adat, ... alakú programsor a kulcsszót követő számot memória terület kezdő címnek értelmezi és ide tölti a következő adatot, majd minden további adatot a következő memóriára címre tölti be.

```

1 'TUNING-5 sorozat POKE PUT-segítsegevel
2 POKE16770,195:POKE16771,76:POKE16772,64
3 FORI=16460TO16477:READ A:POKE I,A:NEXT
4 DATA 205,2,43,213,207,44,205,28,43,209,
    18,19,62,44,190,40,242,201,FJ - LJ
5 PUT15360,65,66,67,68,69,70,71

```

3. program

Érzékelés-jelátalakítás-kijelzés

Egy rendszer, berendezés működéséről, állapotáról érzékszerveinkkel közvetlenül információt kaphatunk. Az érzékszervi észlelés mindig személytől függő (szubjektív), az illető felkészültségétől, begyakorlottságától, pillanatnyi egészségi és lelkiállapotától függ. Ha az érzékeléssel együttjár a beavatkozás szükségessége, akkor a személy ügyességétől függő reakcióidő elteltével kerül arra sor. A

számítógépes játékok zöme begyakorlás során az adott játékhoz megkövetelt reakcióidő csökkentését eredményezi.

Az 1. program elősegíti a C-16 számítógép billentyűzetének megismerését, és ennek az érzékelési folyamatnak megfigyelését. Az egy-egy véletlenszerűen kiírt karakternek megfelelő billentyű lenyomásáig eltelt időt a képernyő felső szélén (digitális alakban) kijelzi a prog-

ram. Ha a karakterek megjelenítését a visszajelzés (billentyűkezelés) megvárása nélkül folytatjuk, beláthatjuk, hogy érzékelésünknek, beavatkozásunknak vannak korlátai.

A HT számítógép elvileg 100 000 Hz frekvenciájú hang kibocsátására alkalmas, az átvitel, a hangszóró korlátai miatt jó, ha 15 000 Hz megszólaltatható. A 2. program lehetővé teszi egy adott frekvenciájú hang kibocsátását. Amennyiben a gépkezelő ezt a hangot hallja, úgy egy billentyű lenyomásával kell jeleznie azt. Több tanuló által kipróbált program esetében a személytől függő korlátok kimutathatók.

A 3. programmal egy Yackley-tárcsás kapcsolót kötünk a HT pontjához (ábra), és a kapcsoló szögelfordulását kívánjuk leolvasni a képernyőn. Ez a jelátalakítás diszkrét bemenetről diszkrét kimenetet biztosított. A C-64 botkormány bemenet, ha oda potenciómétert kötünk, akkor a 4. program segítségével az analóg bemenőjelet diszkrét kimeneti, jellel alakítja át.

Az analóg-digitális kijelzésre, a leolvasás pontosságának bemutatására szolgál a C-16-ra írt, a képernyőre órát kirajzoló (és több kevesebb pontossággal időmérésre is használható) 5. program.

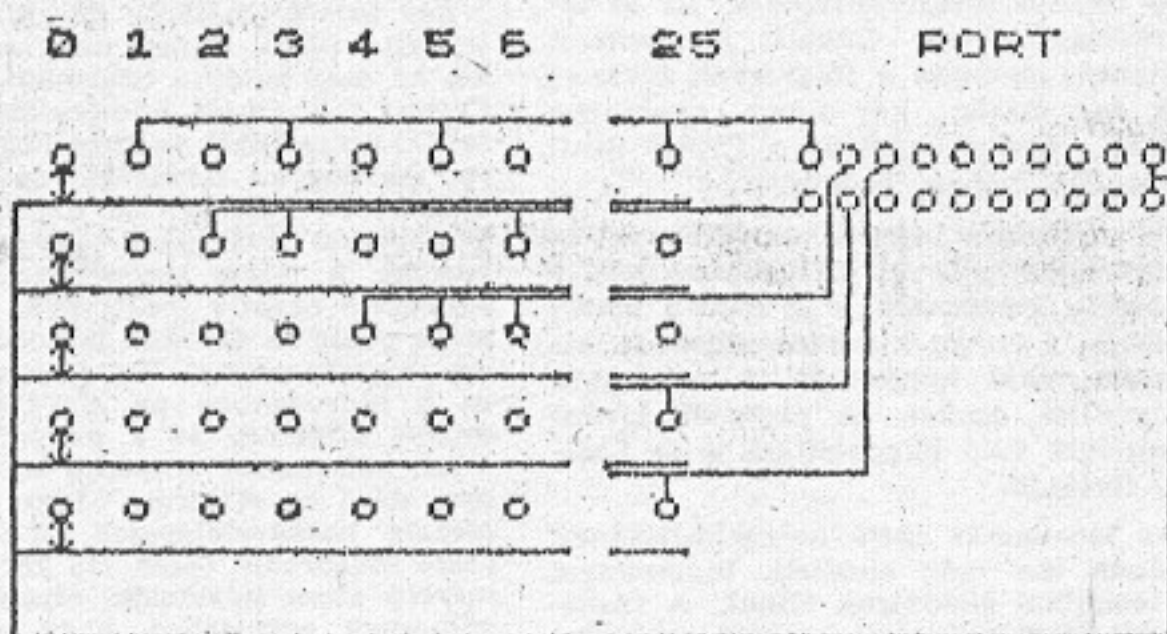
Sz. Lukács János

```
10 REM REAKCIO IDO VIZSGALAT
20 CLR:PRINT"MIT GYAKOROLSZ ?"
30 PRINT" 1 HA ALAP BILLYENTUZE"
40 PRINT" 2 HA SHIFT GRAFIKA" :R=9999
50 PRINT" 3 HA C= GRAFIKA"
60 PRINT"HA ELAKADSZ 'SPACE'-T NYOMD LE"
70 PRINT" 1/2/3 LENYOMASAVAL VALASSZ!"
80 GETKEY V$:IFV$<"1"ORV$>"3"THEN GOTO80
90 IFV$="1"THEN B=33 :C=95
100 IFV$="2"THEN B=186:C=219
110 IFV$="3"THEN B=161:C=184
120 PRINT"II"
130 A=B+INT((C-B)*RND(0)):A$=CHR$(A)
140 PRINT"#####";A$
150 TI$="000000":K=TI
160 GET V$ : IFV$=" " THEN GOTO20
170 IF V$=A$ THEN GOTO 190
180 IFTI$>"000150"THEN END :ELSE GOTO160
190 I=TI-K:F=F+1:IF I<R THEN R=I
200 PRINT"#####";F;"II.FELADAT. IDEJE:";I;
210 PRINT"II " :PRINT"II REKORD:";R;"II "
220 S=S+I:PRINTUSING"II ATLAG:####.#";S/F
230 GOTO130
```

1. program

```
10 ' HALLASKUSZOB HT
20 OUT31,8:OUT30,15:OUT31,7:OUT30,254
30 CLS:K=30:OUT31,0
40 K=K-1:OUT30,K:IFK<4 ,V$="N":GOTO80
50 PRINTUSING"##### Hz " ;109166/K;
60 PRINT"HALLOD I/N ? "
70 V$=INKEY$:IF V$="I",PRINTV$:GOTO40
80 IF V$="N",OUT30,0:END ELSE 70
```

2. program

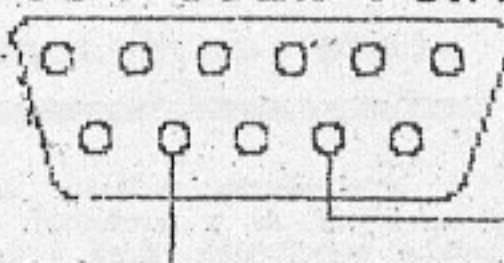


Bekötési vázlat a Yackley-tárcsa szögelfordulásának kijelzéséhez.

```
10 'YACKLEY TARCSEA SZOGELFORDULASA HT
20 CLS:E=1:OUT31,7:OUT30,0:OUT31,14
30 K$="AZ ELFORDULAS MOST: ####.# fok"
40 PRINT"A KOR 26 RESZRE OSZTHATO"
50 N = 255-INP(31) : F = N*360/26
60 IF N=E THEN GOTO 50
70 PRINT USING K$:F : E=N : GOTO 50
```

3. program

C64 USER PORT



300KOHM

```
10 REM ANALOG DIGITAL ATALAKITO C=64
20 PRINTCHR$(147) : A0 = 500
30 C = : REM 'MUSZER ALLANDO'
40 A = PEEK(54297)
50 REM ESETLEG A = A*C
60 IF A <> A0 THEN PRINT A
70 A0 = A : GOTO40
```

4. program

```
10 REM ANALOG DIGITALIS KIJELEZES -C16
20 PRINT"FONTOS IDO ORA/PERC/MASODPERC"
30 INPUT"120330 ALAKBAN :";TI$
40 GRAPHIC 1,1:CIRCLE 1,160,100,90
50 FORI=0TO6.2STEPPI/6
60 X1=160+75*COS(I):X2=160+90*COS(I)
70 Y1=100+75*SIN(I):Y2=100+90*SIN(I)
80 DRAW1,X1,Y1 TO X2,Y2 :NEXTI
90 O$=LEFT$(TI$,2) :P$=MID$(TI$,3,2)
100 M$=RIGHT$(TI$,2) :P=VAL(P$)*PI/30
110 O=VAL(O$)*PI/6-PI/2+P/12:P=P-PI/2
120 IF O>24 THEN O=0
130 IF P<0 THEN GOSUB 160
140 K$=O$+":"+P$+":"+M$:CHAR1,0,0,K$
150 CIRCLE 1,160,100,4 : GOTO 90
160 OX=160+55*COS(O) :OY=100+55*SIN(O)
170 EX=160+55*COS(OE):EY=100+55*SIN(OE)
180 DRAW 0,160,100 TO EX,EY :OE=O
190 DRAW 1,160,100 TO OX,OY
200 PX=160+70*COS(P) :PY=100+70*SIN(P)
210 EX=160+70*COS(PE):EY=100+70*SIN(PE)
220 DRAW 0,160,100 TO EX,EY :PE=P
230 DRAW 1,160,100 TO PX,PY :RETURN
```

5. program

Tuning-6

A DATA sorokban tárolt adatok kiolvasása a HT-gépeken kötött sorrend szerint történik. Amennyiben az adatsor valamelyik részére van szükségünk, akkor akár BASIC-ból is kiolvashatjuk (RESTORE után) a kívánt részt úgy, hogy a DATA sorokban elhelyezünk olyan értékeket, amelyek az egyes példál dallamok, ábrák, kérdések stb. kezdetét jelzik. Ez a mindig újra kezdődő kiolvasatás aránylag hosszú időt vesz igénybe.

Megkönnyíti dolgunkat az egyes gépeken (Spectrum, C-16) meglévő RESTORE sorszámutasítás.

Az itt közölt segédprogram (6. program) a DATA mutatót adott program-sor elejére állítja.

A segédprogram betölthető, lefutatható, és még a NEW sem bántja. Továbbra is él az alap tulajdonságú RESTORE, de a LINE utasítás sorszáma nélkül is ugyanúgy viselkedik. A LINE sorszáma alakú utasítás viszont a programban bárhol alkalmazva biztosítja, hogy READ segítségével a kijelölt DATA sorból kezdve használhassuk fel az adatokat. Használata során vigyázzunk a RE parancs alkalmazásával, mert a LINE utáni sorszámot nem számítja ki és nem írja át.

```
1 'TUNING -6 RESTORE n>> LINE n
2 POKE16803,195:POKE16804,76:POKE16805,64
3 FORI=16440TO16480:READ A:POKE I,A:NEXT
4 DATA 237,75,164,64,40,9,205,79,30,192,
229,205,44,27,225,11,237,67,255,
64,201,Felfoldi,-,Sz.Lukacs,1986
```

6. program

µC-Iskolaszámítógép-programok

A számítógép kapcsolata a külvilággal

```

10 CLEAR500:DIMB$(20),J$(20):Z=1:CLS
20 PRINT"A PROGRAM EGY OLYAN BEJARATI
    AJTOT SZIMULAL, MELYNEL AZ AJTO-
    HOZ ERVE IRANYTOL FUGGOEN AZ
    EGYIK KAPCSOLORA KELL LEPNI,MAJD
    AZ AZONOSITO KODOT BE KELL IRNI."
30 PRINT"HA A 'COMPORT' UGY DONT, AZ
    AJTO KINYILIK,30 MASODPERCEN BE-
    LUL A KOVETKEZO KAPCSOLO LENYO-
    MASAVAL TUL KELL HALADNI A KAPUN"
40 RESTORE:FORI=1TO10:READJ$(I):NEXT
50 PRINT:PRINT:PRINT"TOVABB ?
    NYOMJ LE EGY BILLENTYUT !"
60 IFINKEY$=""THEN60
70 A$=STRING$(3,191):B$=A$+A$:CLS
80 PRINT@32,"BENT VAN:"; : FOR I=1TOZ
90 PRINT@40+I*64,B$(I); " " :NEXT
100 PRINT@50+(I-1)*64," " :
110 PRINT@196," " :PRINT@212," " :
120 FORI=640TO896STEP64
130 PRINT@I,CHR$(220);:NEXT
140 PRINT@320,B$ "A$ "B$A$ "A$ "B$
150 FORI=15TO280STEP64:PRINT@I,CHR$(191);
160 NEXT : OUT31,7:OUT30,63:OUT31,15
170 A=INP(31):IFA<>255THEN170
180 A=INP(31):A=255-A:IFA=0THEN180
190 ON A GOTO 210,250
200 GOTO80
210 PRINT@196,"->":GOSUB310:IF K=2 ,80
220 GOSUB370:IFK=1,GOSUB500:GOTO80
230 GOSUB410:IFA=0ORA=1THEN80
240 B$(Z)=N$:Z=Z+1:GOTO80
250 PRINT@212,"<-":GOSUB310:IF K=2 ,80
260 GOSUB370:IFK=0THENGOSUB500:GOTO80
270 GOSUB410:IFA=0ORA=2THEN80
280 B$(V)="" : FORI=1TOZ
290 IFB$(I)=""THENFORK=I+1TOZ:
    B$(K-1)=B$(K):NEXT:Z=Z-1:I=Z+1
300 NEXT:GOTO80
310 K=0:PRINT@640," " :REM MOZOGHAT ?
320 M=0:INPUT"KOD=";N$
330 FORI=1TO10:IFJ$(I)=N$THENM=1:I=20
340 NEXT:IFM=1THENRETURN
350 K=K+1:IFK=2THENGOTO500
360 PRINT"KOD HIBAS, MEGEGYSZER":GOTO320
370 REM BENT VAN ?
380 K = 0 : FORI=1TOZ
390 IFN$=B$(I)THENK=1:V=I:I=Z
400 NEXT:RETURN
410 REM AJTO
420 FORI=271TO795STEP-64:FORT=1TO100:NEXT
430 PRINT@I," " :NEXT
440 OUT31,7:OUT30,63:OUT31,15
450 FORI=1TO500
460 A=INP(31):A=255-A:IFA=1ORA=2,I=1000
470 NEXT
480 FORI=79TO280STEP64:FORT=1TO100:NEXT
490 PRINT@I,CHR$(191);:NEXT : RETURN
500 REM RIASZTAS
510 OUT31,7:OUT30,248:OUT31,8:OUT30,14
520 OUT 31,0 : FORI=1TO3
530 FORO=200TO80STEP-1:OUT30,0:NEXT
540 FORO=80TO200:OUT30,0:NEXT:NEXT
550 OUT30,0:RETURN
560 DATA SAMU,FIFI,TODOR,a,10,belepesre,
    jogosult,neve,!,!

```

Az ember jártában-keltében, no meg a tv-filmeket nézve olyan bejáratú ajtókat lát, amelyekhez érve a belépésre jogosító igazolványt a leolvasó berendezésbe kell helyezni. A berendezés leolvassa a szükséges információkat, s ha a jogosultsághoz nem fér kétség, mintegy „Szezam táruj” jellegre az ajtó kinyílik.

Ilyen ajtók és berendezések láthatók pl. a Ferihegy 2 repülőtérén, de a metró korábbi peronzáró berendezése is hasonló, igaz, ott a jogosultságot a pénzérme bedobásával szereztük meg. A berendezés, a számítógépportás elkészítéséhez vagy legalább modellezéséhez számítógépünknek a külvilágtól kétféle információt is be kell szereznie.

1. Erkezett-e és ha igen, melyik irányból valaki az ajtóhoz, ha kinyílt az ajtó áthaladt-e rajta?

2. Az ajtóhoz érkezett személy jogosult-e mozogni az ajtón keresztül? (Vigyázat, ugyanaz a személy csak akkor mehet be a legközelebb, ha már közben kijött!)

A személy azonosításához a név beírását az INPUT-ként a billentyűzetről fogja beolvasni a programunk. A mozgás érzékeléséhez fénykaput vagy az ábra szerint, egyszerű bekötésű, „taposásra” működő záróérintkezőket fogunk alkalmazni. Ennek a kapcsolásnak alig van alkatrészigénye és a szerelés sem igényel igényesebb munkát.

A program 70-160. sora a képernyő bal oldalára felrajzolja az ajtót és az érzékelők helyét. Az érzékelőket ténylegesen a porton keresztül figyeli a program. (Akik először csatlakoztat-

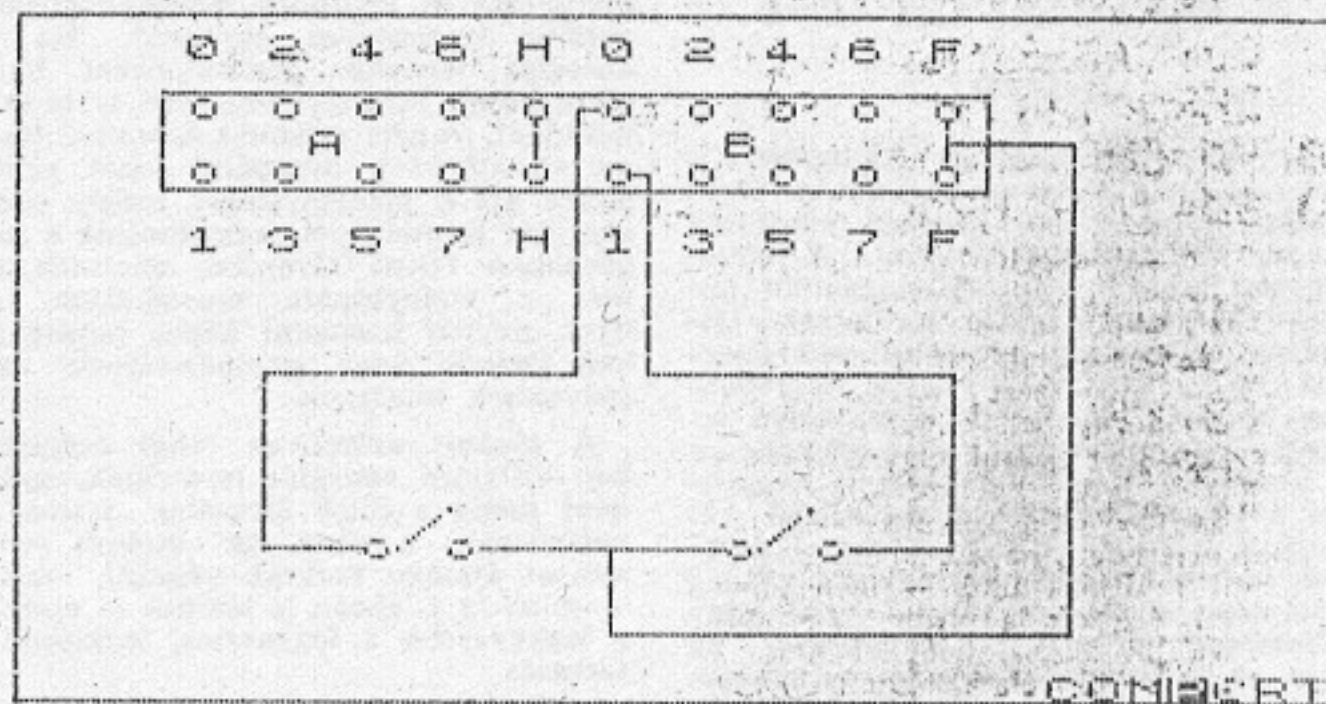
nak a portra, azok szíves figyelmébe ajánljuk, hogy a billentyűzetről nem lehet tönkretenni a gépet, de ha már a porthoz vagy a bővítéshez kellő körültekintés nélkül nyúl valaki, az könnyen kárt okozhat!)

A 170-180. sorok lekérdezik a kapcsolókat és az érkezés irányától (pl. kintől befelé) függően közel azonos felépítésű programszegmenseken folytatódik a program futása. A 310-400. sorok közötti szubrutin INPUT-ként billentyűzetről kéri a kódot (N\$).

A beérkezett kódot több szempont szerint vizsgálja. Összehasonlítja az 560. DATA sorban rögzített névsorral és azzal, hogy a bemenni akaró személy nem használ-e fel olyan kódot, amivel már bent van valaki a „területen”. A bentlétők nevét a képernyő jobb oldalán (a diszpécser-szobában?) leolvashatjuk. Illetéktelen kód alkalmazása esetén az 500-550. sorok riasztást adnak.

A helyes kód és mozgási irány esetén az ajtó kinyílik. (Itt most csak rajzilag, tessék továbbfejleszteni a programot.) Beérkezőnek természetesen csak azt a személyt tekint a program, aki az ajtó mögötti (belső) kapcsolót is működtette. Ha ez a működtetés a 450. soron kijelölt időtartam alatt nem következik be, az ajtó lezár.

A program még néhány „biztonsági intézkedést” tartalmaz. Bizom a diákokban, hogy találnak olyan „lyukat”, amelyet még érdemes figyelembe venni és a programot azon szempontok szerint továbbfejleszteni.



Tuning-7

Ebben a programban ismét a listázás a témánk. A múltkor kiemelt felhasznált rövid kis programmal a listázás sebességét változtathatjuk. Ehhez „saját” gépi kódú programot kell beírni:

- a kitekintésről ugorjunk a saját programra,
- mentsük a regiszterek tartalmát,
- határozzuk meg a listázás sebességét,
- hívjuk meg a ROM késleltető rutinját,
- állítsuk vissza a regiszterek eredeti tartalmát.

A segédprogram a gépi kódú programot a rendszerváltó terület olyan helyére tölti, ahol a gép kikapcsolásáig megmarad. Így a beírt vagy betöltött programok mindegyike a beállított sebességgel listázható. De ha nem tetszik, akkor a program használatát kikapcsolhatjuk POKE 16863,201 segítségével, és vissza-kapcsolhatjuk POKE 16863,195 paranccsal. A program futásának tartamára ezt a segédprogramot ki kell kapcsolni!

Sz. Lukács János

```

10 'LISTAZAS LASSITO LASSU A=255
20 INPUT "LISTAZAS SEBESSEGE";A
30 IF A<1 OR A>255 THEN GOTO 20
40 POKE 16446,A : POKE 16863,195
50 POKE 16864,76 : POKE 16865,64
60 FOR I = 16460 TO 16475
70 READ A : POKE I,A :NEXT I
80 DATA 229,197,213,245,1,0,42,
    62,64,78,205,96,0,195,121,2
90 'NEW : (c) F.J-SZ.L.J

```


Energiaátvitel

Az erőgépek hajtásvázlatának tárgyalásakor is használható a tankönyv II. kötetének 24. oldalához kapcsolódó sebesség-váltó hajtóművet és annak működését bemutató program. A program fő blokkjai:

10—180 felrajzol egy 2X2 fokozatú csúszótömbös hajtóművet.
190—320 bekéri az elemi fogazású hajtómű fogszám- és bemenő fordulatszám értékét (azonos modulú elemi fogazású kerekeket tételez fel a program, az irreális adatok

közlését az 510—540 sorok akadályozzák meg).

330—470 az 1-2-3-4-0 billentyűk segítségével biztosítja a csúszótömb mozgását, és kiszámítja a hajtott tengely fordulatszámát.

480—500 az INPUT adatokat ellenőrző és a rajzolás segítő szubrutinok.

Az ismertetett programok elkészítésével a mozgó ábrák szerkesztéséhez kaphatunk ötleteket, de ha „csak” a tanítást segítő oktatóprogramra van szükségünk, akkor a TII „Erőátvitel” című programját célszerű beszerezni.

```
10 CLEAR700:CLS
20 PRINT33,"SZORZO RENDSZERU SEB.VALTO"
30 VA=CHR(26)+STRING(11,24):
  AA=" "+CHR(171)+CHR(129)+" "+VA+" "
40 FORI=5TO25:SET(I,10):NEXT I
  FORI=3TO17:SET(I,1):NEXT I
  FORI=6TO14:SET(I,1):NEXT I
50 AA=AA+CHR(171)+CHR(129)+CHR(197)+
  CHR(170)+" "+VA+" "
60 SET(12,3):SET(14,3):SET(12,17):
  SET(14,17):SET(18,6):SET(20,6):
  SET(18,14):SET(20,14)
70 AA=AA+CHR(170)+CHR(198)+CHR(170)+
  " "+VA+CHR(140)+CHR(174)+STRING
  (7,191)+CHR(140)+CHR(140)+VA
80 FORI=2TO53:SET(I,25):NEXT I
90 H1=328:H2=402
100 AA=AA+" "+CHR(170)+CHR(198)+CHR
  (170)+" "+VA+" "+CHR(186)+CHR
  (144)+CHR(197)+CHR(170)+" "+CHR
  (26)+STRING(4,24)+" "+CHR(186)+
  CHR(144)+" ":PRINTH1,AA
110 VA=CHR(26)+STRING(7,24)
120 BA=" "+CHR(171)+CHR(129)+" "+VA+
  " "+CHR(171)+CHR(129)+" "+CHR
  (170)+" "+VA
130 BA=BA+CHR(140)+CHR(174)+STRING
  (3,191)+CHR(140)+CHR(140)+VA+" "+
  CHR(186)+CHR(144)+" "+CHR(170)+
  " "+VA+" "+CHR(186)+CHR(144)+
  " ":PRINTH2,BA
140 FORI=30TO56:SET(I,10):NEXT I
150 FORI=0TO20:SET(33,I):NEXT I
160 FORI=3TO17:SET(47,I):NEXT I
170 SET(32,0):SET(34,0):SET(32,20):
  SET(34,20):SET(46,3):SET(48,3):
  SET(46,17):SET(48,17)
180 PRINT128,"nbe":PRINT192,"->":
  PRINT157,"nki":PRINT220,"->":
190 PRINT4,"z1=?":GOSUB500:Z1=Z:
  PRINT7,Z1:PRINT704,"z2=?":
200 GOSUB500
210 Z2=Z:PRINT707,Z2:PRINT72,"z3=?":
220 GOSUB500
230 IFZ1+Z2-Z<17THENGOSUB540:GOTO230
240 Z3=Z:PRINT75,Z3:Z4=Z1+Z2-Z3:PRINT
  777,STR(24):PRINT775,"z4="
250 PRINT720,"z5=?":GOSUB500
260 IFZ1+Z2-Z<17THENGOSUB540:GOTO260
270 Z5=Z:PRINT723,Z5:Z6=Z1+Z2-Z5:
  PRINT20,STR(26):PRINT18,"z6="
280 PRINT786,"z7=?":GOSUB500
290 IFZ1+Z2-Z<17THENGOSUB540:GOTO290
300 Z7=Z:PRINT789,Z7:Z8=Z1+Z2-Z7:
  PRINT91,STR(28):PRINT89,"z8="
310 GOSUB530:PRINT832,"":INPUT"b be
  menő=":NA=N:VAL(NA):IFN(1ORN)>9999
  ORN<>INT(N)THEN310ELSEPRINT256,N:
  GOSUB530
320 PRINT960,"1/2/3/4/0 fokozat":
```

```
330 FA="1":GOTO350
340 FA=INKEYA:IFFA=" "THEN340
350 IFFA<"0"ORFA>"4"THENGOTO340
360 REM FOKOZAT KAPCSOLASA
370 IFFA="0"THENC1=330:C2=H2ELSEIF
  FA="1"THENC1=328:C2=402
380 IFFA="2"THENC1=328:C2=406ELSE
  IFFA="3"THENC1=332:C2=402
390 IFFA="4"THENC1=332:C2=406
400 S=(C1<H1)-(C1>H1):IFS<>0THENGOSUB480
410 S=(C2<H2)-(C2>H2):IFS<>0THENGOSUB490
420 PRINT282,"":IFFA="0"THEN
  PRINT284,"0"):PRINT832,"Üresben
  van "
430 IFFA="1"THENPRINT282,INT(N*Z3/Z4*
  Z5/Z6):PRINT832,"1. hajtás:z3-z4
  -z5-z6"
440 IFFA="2"THENPRINT282,INT(N*Z3/Z4*
  Z7/Z8):PRINT832,"2. hajtás:z3-z4
  -z7-z8"
450 IFFA="3"THENPRINT282,INT(N*Z1/Z2*
  Z5/Z6):PRINT832,"3. hajtás:z1-z2
  -z5-z6"
460 IFFA="4"THENPRINT282,INT(N*Z1/Z2*
  Z7/Z8):PRINT832,"4. hajtás:z1-z2
  -z7-z8"
470 H1=C1:H2=C2:GOTO340
480 FORI=H1TOC1STEPS:PRINTI,AA:
  FORT=0TO10:NEXT NEXT:RETURN
490 FORI=H2TOC2STEPS:PRINTI,BA:
  FORT=0TO10:NEXT NEXT:RETURN
500 REM FOGSZAMOK ?
510 PRINT832,STRING(96," "):PRINT
  832,"":INPUT"fogszám=":ZA=Z:VAL(ZA)
520 IFZ<17ORZ>200ORZ<>INT(Z)THEN510
530 PRINT832,STRING(96," "):RETURN
540 PRINT832,"REALIS ERTEKET KEREK!":
  FORT=0TO500:NEXT:RETURN
```

Tuning-8

A BASIC-programok bizonyos részéről gyakran hívunk meg gépi kódú rutint az USR (n) függvény segítségével. A HT géppel ez a meghívás csak akkor sikerül, ha a gépi kódú segéd-rutin (akár a ROM egy ismert rutinja) címét előzőleg a rendszerváltozó terület 16526/16527 címekre beírjuk. Főleg ha több rövid gépi kódú programrészt alkalmazunk, ez a címtárolás programilag is nehézkes —, de persze hol

jön ez a meghívandó programrész megírásának munkáigényéhez?

Az itt közreadott két kis (a korábbi gyakorlat szerint a NEW után is élő) segédprogram lehetővé teszi az A=USR (kezdőcím dec) alakú alkalmazást.

A zárójelbe decimális alakban beírt memóriacímek kezdődő programrész most már (minden korábbi cím kijelölése nélkül) zökkenőmentesen végrehajtásra kerül.

```
1 'gepi kodu rutin hivas USR(CIM dec)
2 POKE16809,195:POKE16810,76:POKE16811,64
3 FORL=16460TO16472:READ J:POKE L,J:NEXT
4 DATA 229,215,207,40,205,79,30,237,83,
  142,64,225,201: ' Tuning-8/FJ-LJ
```

```
1 'gepi kodu rutin hivas USR(CIM dec)
2 POKE 16526,76:POKE 16527,64:L=16460
3 POKE L,205:POKE L+1,127:POKE L+2,10
4 POKE L+3,233: ' Tuning 8.1 /FJ-LJ
```

Lengyel műszaki-gazdasági napok

Április 14—16. között rendez meg a Lengyel Külkereskedelmi Kamara, az MTESZ és a Kereskedelmi Kamara együttműködésével azt a rendezvény-sorozatot, amelyen 23 szakelőadás hangzik el.

Az előadások az elektronika, az ipari automatika, a járműipar, a vegyipar

gépgyártás területét, valamint a mező-gazdasági és élelmiszeripari ágazatokat érintik. Az előadók a Lengyel Népköztársaság tudományos és műszaki életének neves szakemberei közül kerülnek ki. A rendezvénysorozatot az MTESZ Budapest, V. ker. Kossuth Lajos tér 6—8. sz. alatti székházában tartják.

µC-Iskolaszámítógép-programok

Villamos energia elosztása, számla készítése

A lakáson belül, de országos szinten is a villamosenergia-fogyasztók összes teljesítménye jóval meghaladja a villanyóránban lévő biztosíték, illetve a rendelkezésre álló erőművek terhelhetőségét. Az élet ritmusa, a fogyasztók sokfélesége azonban egyidejűleg nem szokta igényelni a maximumot. Ezt a tervezés során egyidejűségi tényező figyelembevételével vesszük számításba. De mi van, ha mégis a lehetőségeket meghaladó egyidejű terhelés lép fel? Kiold a biztosíték, és azon a hálózatszakaszon egyik fogyasztó sem kapja meg a kívánt energiát.

Az itt közölt 1. program egy olyan vezérlőterem munkáját modellezi, amely a hálózat maximális terhelését biztosítva fontossági sorrendben teszi lehetővé az energia vételezését.

A program kiindulási feltétele: SM = 100 kW maximális teljesítményfelvétel.

A 40–70 sorokon kell fontossági sorrendben beírni a fogyasztók teljesítményigényét. (A jól követhető „játék” kedvéért a beírt teljesítményigények 10–40 között

legyenek, és összegük haladja meg a 150 kW-ot.)

A 160–190 sorok figyelik a kézi bekapcsolásokat az 1–8 billentyűk valamelyikének lenyomásával.

A kikapcsolás SHIFT és 1–8 billentyűkkel történik.

A program a képernyőre felrajzolja (ábra) a fogyasztók kézikapcsolóinak állapotát, kiírja a pillanatnyi teljesítményfelvételt.

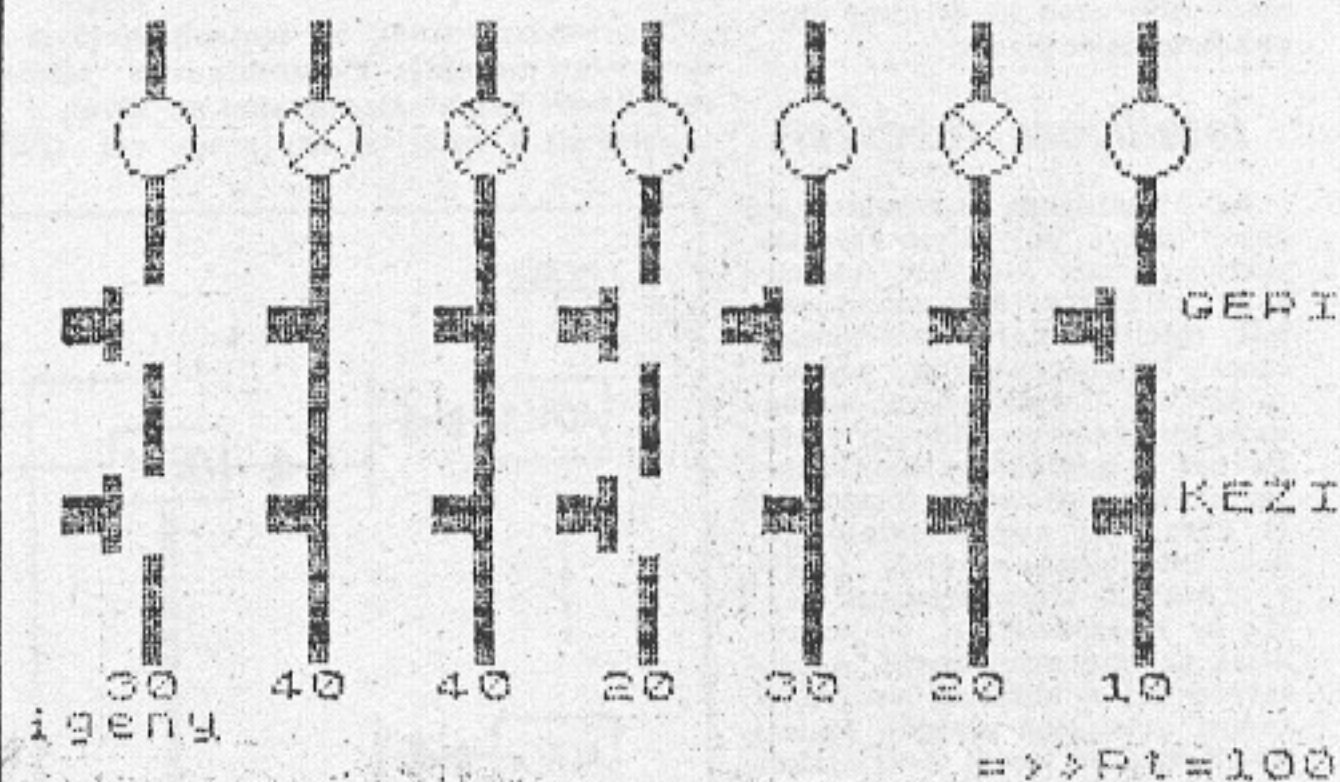
A 200–330 sorok a hozzájuk tartozó szubrutinokkal elvégzik a szükséges vizsgálatokat. Akkor végzik el a „gépi” rákapcsolást, ha az így kialakuló teljesítményigény nem haladja meg a maximumot. A fogyasztók mindenkor hálózatra kapcsolását (itt most csak rajztechnikailag) fontossági sorrendben végzi.

Egy „sorban álló”, alárendeltebb fogyasztónak akkor ad energiát, ha egy nálánál fontosabb fogyasztó kézi kapcsolással kilép a hálózathoz, és az így felszabaduló energia fedezi az igényt.

A program kapcsán a háztartás fogyasztói, azok fontossági sorrendje is vizs-

ENERGIAOR

Pm = 100



```

10 REM ENERGIAOR
20 SM=100:Z=8:B$="" "+CHR$(174):K$=CHR$(174)+" "
30 CLS:PRINT"A FOGYASZTOK FONTOSSAGI SORRENDEN:"
40 FORI=1TOZ
50 PRINTI:INPUT".FOGYASZTO TELJ. IGENYE=";P(I)
60 IFP(I)<=0ORP(I)>99THENS0
70 NEXT : CLS:PRINT08,"ENERGIAOR"
80 PRINT0897,"1 2 3 4 5 6 7 8"
90 FORI=1TO8:PRINT0960+(I-1)*3,P(I);:NEXT
100 FORI=1TO8:FORV=9TO40:SET(3+(I-1)*6,V):NEXT
110 GOSUB540:RESET(3+(I-1)*6,11):NEXT : SU=0
120 PRINT0473,"GEP";:PRINT0665,"KEZI";
130 REM KAPCSOLO FIGYELES
140 PRINT026,"MAX=";:PRINT091,SM;
150 PRINT0218,"PILL=";:PRINT0284,SU;
160 A$=INKEY$
170 IFA$>"0"AND A$<"9"THENI=VAL(A$):GOTO200
180 IFA$>"!"AND A$<"")THENI=ASC(A$)-32:GOTO400
190 GOTO160
200 REM BE
210 IFB(I)=1THEN130 :GOSUB 480 ELSE GOSUB 480
220 IFSU+P(I)<=SMTHENGOSUB500:GOTO130: SIMAN BE
230 REM VAN'E ALARENDELTE BEKAPCSOLVA ?
240 J=0:JP=0:LP=0:II=I
250 FOR N=I+1 TO Z:IFA(N)=1THENJ=1
260 IFA(N)=1ANDP(N)=P(I)THENJP=1:LP=N
270 NEXT:IFJ=0THEN130
280 IFJP=1THENI=LP:GOSUB560:I=II:GOSUB500:GOTO130
290 REM AZ ALARENDELTEK KIKAPCSOLASA FEDEZI E EZT?
300 SL=0 : FOR N= I+1 TO Z : IFA(N)=1THENSL=SL+P(N)
310 NEXT : IFSM-SU+SL<P(I)THEN130
320 REM HA IGEN LEKAPCSOLOM OKET
330 GOSUB340:GOTO130
340 II=I:FORN=I+1TOZ : IF A(N)=1 , I=N :GOSUB 560
350 NEXT:I=II
360 REM VISSZAKAPCSOLHATO VALAMI?
370 FORN=IT0Z
380 IFB(N)=1AND A(N)=0ANDP(N)<=SM-SU,I=N:GOSUB500
390 NEXT:RETURN
400 REM KIKAPCSOL KEZZEL
410 IFB(I)=0THEN130
420 K=0:IFA(I)=0THENK=1:GOSUB 540: ELSE GOSUB 540
430 REM BEKAPCSOLHATO E EGY VARAKOZO ?
440 J=0:FOR N= 1 TO Z:IFB(N)=1AND A(N)=0THENJ=J+1
450 IFJ=1THENK=N
460 NEXT : IFJ=0THEN130
470 I=0:GOSUB340:GOTO130
480 REM KEZI KAPCSOLO RAJZ
490 B(I)=1:PRINT0640+(I-1)*3,B$;:RETURN
500 REM GEP BEKAPCSOL
510 SU=SU+P(I):A(I)=1:PRINT0448+(I-1)*3,B$;
520 PRINT0257+(I-1)*3,"*";:RETURN
530 REMLEKAPCS RAJZA
540 PRINT0640+(I-1)*3,K$;:B(I)=0
550 IFK=1THENRETURN
560 A(I)=0:SU=SU-P(I):PRINT0448+(I-1)*3,K$;
570 PRINT0257+(I-1)*3,"0";:RETURN

```

1. program

gálható. Továbbfejlesztve a programot azt is figyelembe lehetne venni, hogy melyek azok a fogyasztók a háztartásban, amelyek egy ilyen „energiaór” önkényes döntése alapján várakozni nem kapcsolhatók sem be, sem ki.

Amennyiben a program kapcsán a lakásban lévő fogyasztásmérővel és a Díjbeszedő Vállalat által küldött számlával

is sikerül megismerkedni, akkor kézenfekvő a villanyszámla elkészítésére programot készíteni. A 2. program semmi programozási újdonságot nem tartalmaz, de készítésekor az egységár INPUT-kénti beírása során a tarifák közötti különbségek indokai megvilágíthatók. A 40. sor feltételezi, hogy a fogyasztásmérő 9999 kijelzés után átfordul és 0-tól számlál.

```

10 ' HAZI DIJSZAMITO/SZAMLAZO
15 CLS:INPUT"EGYSÉGAR Ft/kWh:";A
20 INPUT" ELOZO LEOLVASAS kWh:";E
25 IF E<0 OR E>10000 ,GOTO20
30 INPUT"UTOLSO LEOLVASAS kWh:";U
35 IF U<0 OR U>10000 ,GOTO30
40 IF U<E , E1 = E-10000 ELSE E1=E
45 F=(U-E1)*A : CLS
50 PRINTUSING"Elozo: #### kWh";E
55 PRINTUSING"Utolso: #### kWh";U
60 PRINTUSING"Fogyasztas:#### kWh";U-E1
65 PRINTUSING"Fizetendo: ####.## Ft";F
70 PRINT0640,"":GOTO20

```

2. program

Tuning-9

A keretprogramok különböző adatokkal tölthetők fel. A szalagon történő adattárolás — összes problémáját elismerve — sok lehetőséget kínál a „PRINT kettős kereszt” / „INPUT kettős kereszt” felhasználásával.

Minden, maximum 255 byte hosszú adatfile előtt a HT 255 0 byte-ot küld ki a magnószalagra. Ez a bevezető byte-sorozat hosszadalmassá teszi az adatok szalagra írását, illetve beolvasását.

A 0 byte-sorozat szerepe a szalagsebesség állandósulásának kivárása. Egyéni, gépi adottságoktól is függ, hogy a magnó

hogyan gyorsul fel, de általában 20...30 byte magnószalagra küldése, kb. 0.2...0.5 másodperc alatt a szalagsebesség állandósul.

A „PRINT kettős kereszt” régrehajtatása — minthogy itt is van kiltekintés a ROM programból — beavatkozhatunk. Az itt közölt program az adatfile bevezető 0 byte-ok számát csökkenti.

Ennek a segédprogramnak az alkalmazása főleg akkor indokolt, ha egy-egy adatfile aránylag rövid, és ezekből többet kell magnószalagon tárolni.

Sz. Lukács János

```

1 ' Tuning-9 PRINT# GYORSABBAN fjlj
2 POKE16842,195:POKE16843,76:POKE16844,64
3 FORL=16460TO16480:READ J:POKE L,J :NEXT
4 DATA254,64,200,51,51,254,35,194,155,32,
   205,254,1,6,30,205,137,2,195,150,32

```

3. program

Elektromos számítások

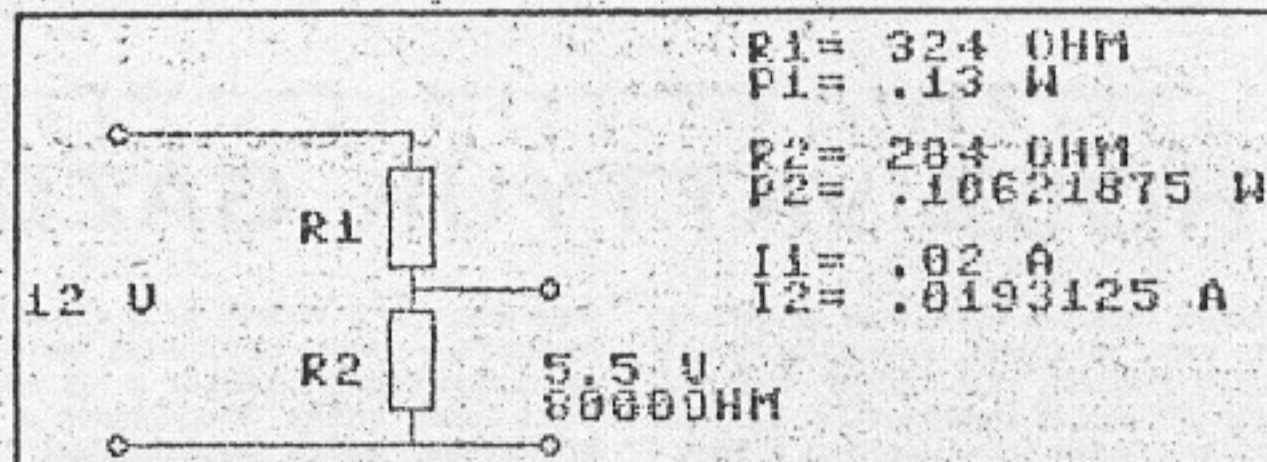
A technika tárgy keretében a feladatlapok egy sor elektromos és elektronikus kapcsolás kivitelezését javasolják. A tényleges manuális tevékenységet a tervezőmunka mindig megelőzi.

Itt néhány ötletet adunk, amelyek a számítógép segítségével alapszintű elektromos számításokat végeznek. A programok mindegyike adja azt a többletet is, amelynek alapján a feladat kapcsolási rajza a képernyőn meg is jelenik.

Az 1. (HT-re írt) program a párhuzamos ellenállások eredőjét számítja ki. A 30. soron definiáljuk az ellenállás

rajzjelét, a 40...110 sorok között. INPUT-ként beírt ellenállásértékeket felhasználva kiszámítja a program az eredő ellenállást és felrajzolja a kapcsolást. Az adatbevitel $R=99$ érték beírásával fejezhető be. Az utolsó összetevő beírása után a 120...140 sorok segítségével a képernyőre írathatjuk az eredő értékét. A 160...190 sorok a zárlat $R=0$ érték kijelzésére szolgálnak.

A 2. program feszültségosztó méretezését teszi lehetővé. C16 grafika adta lehetőségeket kihasználva még a kapcsolási rajz is megjelenhet a képernyőn (ábra).



Az ábrát a 20...100 sorok készítik el. A 110...170 soron körbe a program a méretezéshez szükséges adatokat. Az alapadatokból a 300...310 sorok határozzák meg a szükséges ellenállás értékét. Az ellenállások és teljesítményeik értékét a 320...370 sorok írják ki a képernyőre. A képernyőre rajzolt kapcsolási rajzok feliratozása a felirat megfelelő elhelyezése pl. a 2. program 200. sorához hasonló megoldással lehetséges.

A 3. program az elkészített rajz feliratozásához egy másik, általánosabban használható megoldást mutat be. A gra-

fikus képernyőre szerkesztett rajzba a képernyő bal felső sarkából a kurzormozgató billentyűkkel helyezhetjük a megfelelő helyre a kurzort. A megfelelő helyre érve, az odairni kívánt karakter billentyűjének lenyomásával a képernyő kívánt helyén megjelenik a felirat. A kurzor visszaugrik alapállásba mindaddig, amíg a RETURN billentyű lenyomásával ki nem lépünk ebből a szubrutinból. Az ötlet továbbfejleszthető saját definícióval — például elektromos rajzjeleket tartalmazó — karakterek kiírásának megvalósításával.

```
10 REM PARHUZAMOS ELLENALLASOK EREDOJE
20 CLS:K=-50
30 FORI=1TO6:READA:R=R+CHR$(A):NEXTI
40 PRINT@832,"":INPUT"R=":"E$
50 R=VAL(E$):IF R<0 THEN GOTO 40
60 K=K+128:V=V+6
70 IF R=99 OR R>800 THEN GOTO 120
80 PRINT@K,R$;:PRINT@K+9,R;
90 FORI=4TOV:SET(27,I):SET(40,I):NEXT
100 IF R=0 THEN K=K+128:GOTO 160
110 C=C+1/R:GOTO 40
120 PRINT@832,"":C=1/C
130 C=INT(100*C+.5)/100:PRINT@K+1,""
140 PRINT"R EREDO=","C:"OHM"
150 IF INKEY$="" THEN 150 ELSE RUN
160 FORI=1TO5:PRINT@K,"ZARLAT";
170 FORT=1TO60:NEXT
180 PRINT@K,"* * *";
190 FORT=1TO60:NEXT:NEXT
200 PRINT@832,"":C=0:K=K+64:GOTO 130
210 DATA 140,174,179,179,157,140
```

1. program

```
10 REM FESZULTSEGOSZTO MERETEZESE C-16
20 GRAPHIC 2,1
30 CIRCLE1,30,30,2:CIRCLE1,30,100,2
40 CIRCLE1,130,65,2:CIRCLE1,130,100,2
50 DRAW1,32,30TO98,30TO98,38TO98,38TO98,60
60 DRAW1,93,60TO103,60TO103,38TO98,38
70 DRAW1,98,60TO98,70TO98,70TO98,92TO98,92
80 DRAW1,98,92TO103,92TO103,70TO98,70
90 DRAW1,98,92TO98,100:DRAW1,98,65TO128,65
100 DRAW1,32,100TO128,100
110 CHAR1,0,8,"U1":INPUT"U1=":U1
120 K$=STR$(U1)+" V":CHAR1,0,8,K$
130 CHAR1,15,10,"U2":INPUT"U2=":U2
140 K$=STR$(U2)+" V":CHAR1,15,10,K$
150 IF U1<U2 THEN PRINT"ADAT HIBA":END
160 INPUT"PRIMER ARAMERO=":I1
170 INPUT"SZEKUNDER ELLENALLAS=":R3
180 K$=STR$(R3)+" OHM":CHAR1,15,11,K$
200 CHAR1,9,6,"R1":CHAR1,9,10,"R2"
300 R=U1/I1:R1=INT((U1-U2)/I1)
305 I3=U2/R3:I2=I1-I3
310 R2=INT(U2/I2):P1=(U1-U2)*I1:P2=U2*I2
320 K$="R1="+STR$(R1)+" OHM":CHAR1,22,1,K$
330 K$="P1="+STR$(P1)+" W":CHAR1,22,2,K$
340 K$="R2="+STR$(R2)+" OHM":CHAR1,22,4,K$
350 K$="P2="+STR$(P2)+" W":CHAR1,22,5,K$
360 K$="I1="+STR$(I1)+" A":CHAR1,22,7,K$
370 K$="I2="+STR$(I2)+" A":CHAR1,22,8,K$
```

READY.

2. program

```
10 GRAPHIC1,1
100 REM ABRA "ELIRATOZO SZUBRUTIN C-16"
110 X=0:Y=0:CHAR1,Y,X,""
120 GETKEY$:CHAR1,Y,X,""
130 IFA$=CHR$(13)THEN RETURN
140 IFA$=CHR$(145)ANDX>0 THENX=X-1:GOTO200
150 IFA$=CHR$(17)ANDX<24 THENX=X+1:GOTO200
160 IFA$=CHR$(157)ANDY>0 THENY=Y-1:GOTO200
170 IFA$=CHR$(29)ANDY<39 THENY=Y+1:GOTO200
180 IFA$>"!"ANDY$<"Y" THENCHAR1,Y,X,A$
190 GOTO110
200 CHAR1,Y,X,"":GOTO120
```

READY.

3. program

Tuning—10

A Sinclair gépeken használható a GOTO N, viszont nincs ON N GOTO utasítás. Esetenként jól jön — természetesen az átszámozáskor nehézséget okozhat —, ha a program futása inputként, vagy valamilyen függvénykapcsolat alapján kiszámított változó értékeinek megfelelő soron folytatódik. Az ON N GOTO után

írt sok (mondjuk 100) cím amúgy is igen „csúnya”.

A HT-gépeken alkalmazható LINE GOTO, illetve LINE GOSUB utasítást a 4. program mutatja be. Szakítva a korábbi gépi kódú programtárolási móddal, a kisegítő rutint a program első REM sorában tároljuk. Ezért igen fontos a sorban a helybiztosítás!

Sz. Lukács János

```
10 REM ::::::::::::::::::::::::::::::::::::
KISZAMITOTT GOTO/GOSUB LINE GOTO
20 DATA 43,215,126,254,141,40,22,254,
145,194,151,25,229,205,1,43,
225,195,197,30,195,239,66,52,LJ
30 FORI=17135TO17146:READA:POKEI,A:NEXT
40 FORL=0TO16:POKEI,I,PEEK(7857+L):NEXT
50 FORI=17164TO17171:READA:POKEI,A:NEXT
60 FORI=16803TO16805:READA:POKEI,A:NEXT
100 INPUT A
110 LINE GOTO COS(0)*(A+1)*10:REM PL.
200 PRINTA:"EZ 200":GOTO100
300 PRINTA:"EZ 300":GOTO100
```

4. program

SZÁMÍTÓGÉPES APRÓSÁGOK

A következőkben néhány apró ötletet, hasznos vagy kevésbé hasznos „POKE”-adunk közre, bízva abban, hogy ol-

vasaink találjanak köztük újat, és még újabb ötleteiket később velünk és olvasóinkkal is megosztják.

ZX 81

A POKE 16510,0 parancs az első, már beírt programsornak 0 sorszámot ad. Ez a sor a továbbiakban a szokásos editálással már nem módosítható. Jó szolgálatot tesz ez az első REM sorban tárolt

gépi kódú rutin alkalmazása esetén.

A POKE 16418,0 lehetővé teszi mind a 24 képernyősor használatát, ilyenkor akár kiadhatjuk a PRINT AT 23,27; „OKE” utasítást is.

```
10 REM SPACE KIIRTO SEGEDPROGRAM C=16
12 FOR L=830 TO 955 : READ J
14 POKE L,J : S=S+J : NEXT L
16 IF SC<14027 THEN PRINT "H I B A" :END
18 PRINT "MEGHIVHATO SYS 830" :NEW
20 DATA 166, 43,202,134,167,134,169,166
22 DATA 44,134,168,134,170,160,000,132
24 DATA 2,177,167,240, 31,201, 34,208
26 DATA 8,170,165, 2, 73,255,133, 2
28 DATA 138,166, 2,208, 4,201, 32,240
30 DATA 5,145,169, 32,153, 3, 32,160
32 DATA 3, 76, 79, 3,160, 0,132, 7
34 DATA 162, 5,177,167,145,169,240, 16
36 DATA 132, 7, 32,153, 3, 32,160, 3
38 DATA 202,208,239,132, 2, 76, 79, 3
40 DATA 230, 7,165, 7,201, 3,240, 17
42 DATA 76,128, 3,230,169,208, 2,230
44 DATA 170, 96,230,167,208, 2,230,168
46 DATA 96, 32,153, 3,166,169,164,170
48 DATA 134, 45,132, 46, 32,147,138, 32
50 DATA 24,136, 76,126,134,000
```

READY.

ZX Spectrum

A nagy- és kisbetűs üzemmód programból is megváltoztatható. A POKE 23658,8 nagybetűs, a POKE 23658,0 a kisbetűs üzemmódra való átváltást eredményezi.

A képernyő teleírása után megjelenik a scroll? felirat. Ha beírjuk a POKE 23692,255 utasítást, akkor ez a kiírás csak 255 végrehajtott scroll után jelenik meg. Hogy ez mit jelent, a próbához

futtassuk a következő programot:

```
10 POKE 23692,255
20 FOR I=1 TO 1000
30 PRINT I
40 NEXT I
```

A billentyű ismétlési sebessége a 23562 memóriacímre írt értékétől függ. A javasolható érték 1 és 10 közötti, de kísérletezni lehet.

HT

Egyes programrészek végrehajtásának idejére a számítógép billentyűzetét vagy csak néhány billentyű hatását letilthatjuk. Break letiltása POKE 16396,164 beírásával valósítható meg, csak Shift+Break hatásos a POKE 16396,165 után. Az alapállapot a POKE 16396,201 segítségével állítható vissza.

A billentyűzet letiltása inkább csak rossz tréfa lehet, hiszen ennek végrehajtása után a számítógép „nem áll szóba” velünk.

A programban alkalmazott POKE 16406,82 letiltja a billentyűzetet és valamivel gyorsabb programfutást eredményez. A billentyűzet interpreter általi figyelése a POKE 16406,237 program-sor segítségével állítható vissza.

C 64

Nagybetűs és kisbetűs kiírás átváltása C 64-számítógéppel a PRINT CHR (8), illetve PRINT CHR (9) segítségével lehetséges.

A kiírás módja a POKE 657,128, illetve

POKE 657,0 segítségével is hasonló módon módosítható. Vannak mániákusan programvédelemre törekedő emberek, ezeknek közöljük a következő lehetőségeket.

A programot eldugjuk a Basic értelmező elől (ez jól jöhet programok összemészolásakor is):

POKE 44, PEEK (46)+1
visszaállítható az alapértelmezés és

C 16

A listázás tiltása, minthogy az ezen részt végrehajtó rutin a C 64-hez ké-

pest más helyen van, más értékek beírását kéri.

Itt is közreadunk néhány ötletet:

```
POKE 775,128 139
POKE 774,187 110
POKE 774,126
POKE 775,134
POKE 774,120
POKE 774,128
```

A listázást elrontó beavatkozások a program végrehajtását nem befolyásolják, de jó lehet adott esetben a gép alapállapotba történő visszaállítása.

Mielőtt megváltoztatunk egy rendszer-változó területen lévő értéket, előtte az alapértéket olvassuk ki és jegyezzük fel! A program leállítását megakadályozza a POKE 806,103 beírása, de ugyanerre a címre beírt

POKE 806,104 azt eredményezi, hogy a RUN beírása után READY-val vála-

zol a gép anélkül, hogy a programot lefuttatta volna.

A C 16 aránylag kis szabad memóriával rendelkezik grafikus üzemmód használatára esetén.

A memóriaterülettel való takarékoskodás nagymesterei a ZX 81 alapgép tulajdonosai, a nagyobb memóriával rendelkező gépek tulajdonosai sokszor indokolatlanul pazarolják a tárolóhelyet.

Ha egy programírás során nagyon teleltöttük a memóriát, jól jöhet párszor tíz byte megnyerése is. Ilyenkor célszerű a szóközöket elhagyni a programból. Az itt közölt program SYS 830 meghívásával a program összes felesleges szóközét kiveszi. Igaz, áttekinthetlenebb, de rövidebb ezek után lesz.

Sz. Lukács János

Fényújság

Ez a program HOMELAB II. (AIR-COMP 16) számítógépre készült.

A program a kiírandó szöveget INPUT-tal veszi be, a korábban kért számú sztringbe (20 és 50. sor). A sztringek száma maximum 25 lehet, és sztringenként 255 karaktert tartalmazhat.

Futás közben a program átírja önmagát (50. és 150. sor, és így sorban az A\$, B\$, C\$, ... stb. sztringbe tölti. A 80. sorban egyenként áttölti az aktuális sztring tartalmát a Z\$-jelű munkaváltozóba. A 100. sor a kurzort az első sor utolsó pozíciójába állítja, majd a 110. sor kinyomtat egy-egy karaktert.

A 120. sor meghívja a 10. sorban a

4090H címre fárolt mozgatórutint. A 170. sor alapállapota hozza a változókat. A 180. sor előlről kezdi a kiíratást. A program megváltoztatása vagy pontatlan gépelése esetén a program tönkreteszi önmagát.

Megoldható még a sztringek magnóra mentése s visszatöltése is.

Balázs Gábor

Megjegyzés:

Az itt közölt pár soros program aránylag ritka — de annál lelkesebb tulajdonosok kezében lévő — számítógép-típusra készült. A számítógép memóriájának ismeretében bármely típusra könnyen alkalmazható a programban található szellemes módszer.

```
5 PRINT CHR$(12/;"FENYUJSAG"
10 POKE 16528,33,2,192,17,1,192,1,39,0,237,176,201
20 INPUT "HANY SZTRINGET KER ?";N
30 FOR I=1 TO N
40 PRINT CHR$(12/;"KEREM AZ";I;" SZTRINGET"
50 INPUT A$: POKE 16685,PEEK/16685/+1
60 NEXT I: PRINT CHR$(12/
70 FOR I=1 TO N
80 Z$=A$
90 FOR L=1 TO LEN/Z$/+40
100 POKE 16404,39,192
110 PRINT MID$/Z$,L,1/
120 CALL 16528
130 FOR T=1 TO 20: NEXT T
140 NEXT L
150 POKE 16728,PEEK/16728/+1
160 NEXT I
170 POKE 16685,97: POKE 16728,97
180 GOTO 70
```

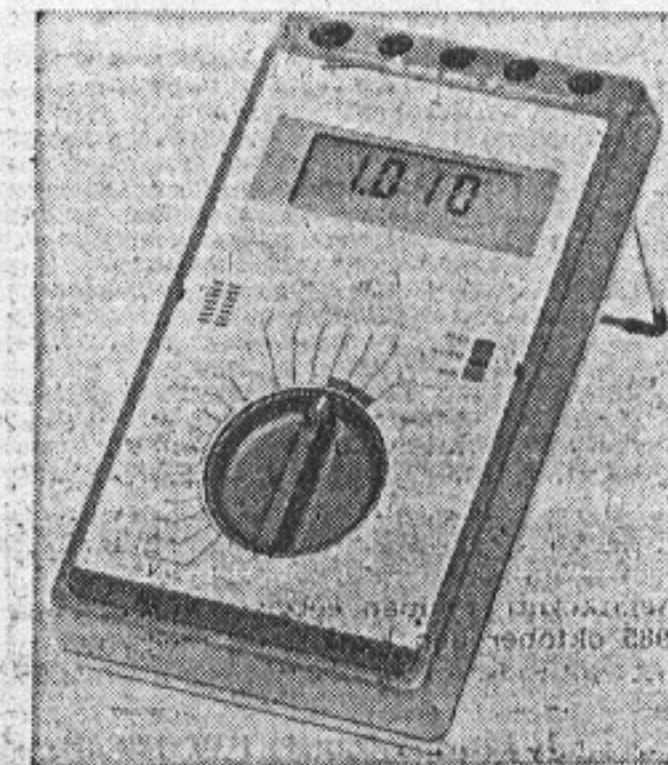
Mezozóos köolajprogram

Megkezdődött Türkméniaiban (Szovjet-Közép-Ázsia) a mezozóos program megvalósítása. Az első kis mélységű fúrás kb. 66 m-es volt.

A mezozóos rétegek szénhidrogén-tárolóképességének felfedezése Dny-Türkméniaiban az úrfelvételek kiértékelésén alapulnak, a szomszédos területek (É-Kaukázus, Irán) geológiai felépítésének hasonlósága alapján. A prognózis szerint a köztársaság jelentős részén várható a mezozoikumából származó köolaj. Ugyanakkor a mezozóos rétegek megközelítése Türkméniaiban igen bonyolult. Az asztal-símaságú puszták hihetetlen mennyiségű kőzetet rejtjenek a felszín alatt. A gyűrt kőzetek több száz, néha több ezer méter mélyen fekszenek. Így itt a mezozoikum mélyebben fekszik, mint a Szovjetunió területén bárhol másutt. Korábban is próbálták már itt megfúrni a mezozóos rétegeket, de sem a fúróberendezés, amit ide telepítettek, sem a rendkívül bonyolult geológiai viszonyok nem tették lehetővé a termelőrétegek elérését. Az új, BU-6500 típusú fúróberendezéssel elérheti a kívánt réteget, hiszen a 6500 m mélységig képes lefúrni.

Sokoldalú kéziműszer

A Siemens-gyár legújabb kéziműszere — a B 1010 típusszámú multiméter —, digitálisan jelzi ki a mért áram, feszültség-, ellenállás-, és hőmérsékletértékeket. Már ez utóbbi is több, mint amit a szokásos kéziműszerek nyújtanak, az azonban még érdekesebb, hogy három fázis csatlakoztatása esetén még a forgómozgó irányának meghatározására is alkalmas. Az ábrán látható a kényelmes leolvasást elősegítő, kibillenthető támasz is.



MŰHELYSAROK

A különböző típusú számítógépek, sőt egy bizonyos típusú gépek különböző generációi az általános és középiskolák programellátásának biztosítása szempontjából igen sok nehézséget vetnek fel. Vannak törekvések az iskolák közötti számítógépcserékre és egy-egy iskola azonos típusú gépparkkal való felszerelésére.

Az ilyen gépcserék esetében a korábban már megírt programokat át kell írni az új géptípusra. Ennek a feladatnak teljesítése sokszor csak mechanikus munkát igényel. Márpedig, ha tényleg csak (vagy zömében csak) mechanikus programírást kell végezni, akkor azt célszerű lenne a számítógépre bízni.

A Kaffka Margit Gimnáziumban működő oktatástechnikai bázison — mint-hogy a fent vázolt probléma égetően jelentkezett — az ábra szerinti gépkapcsolással biztosítják a HT 1080Z számítógép programjainak mágneslemezen történő tárolását. A lemezen tárolt HT-program a Commodore C16 (Plus 4) gép memóriájába tölthető.

kat tartalmazó programok, e programokban a képernyőkiírás és a grafikus képernyőfelbontás különbözősége miatt komoly utólagos átirások szükségesek.

3. nem fordíthatók, ugyanis nem érdemes lefordítani azokat a programokat, amelyek POKE USR kulcsszavakat tartalmaznak, hiszen a memória kiosztása, az esetleg meghívott gépi kódú rutinok címei a két gépben alapvetően eltérnek.

Az egy menetben fordítható programok áttokenizálásához tudni kell, hogy a programsorok tárolása a két gépben azonos. A tokenek — kulcsszavak — memóriában tárolt egy (esetenként több) byte-os rövidítését pl.: HT-gépekhez: Felföldi—Lukács; HT 1080Z iskolaszámítógép fix tárolója, a C16 (Plus 4)-hez Erdős: Commodore ROM-lista Plus 4, C-16, C-116 könyvéből tanulmányoz-

3. Lemezről C=plus 4-be LOAD

„NEV”, 8.

4. Fordítóprogram rátöltésének (AP-PENDE) előkészítése POKE 43, PEEK (45)—3:POKE 44, PEEK (46)

5. A fordítóprogram betöltése LOAD „FORDIT”, 8

6. Fordítóprogram indítása RUN

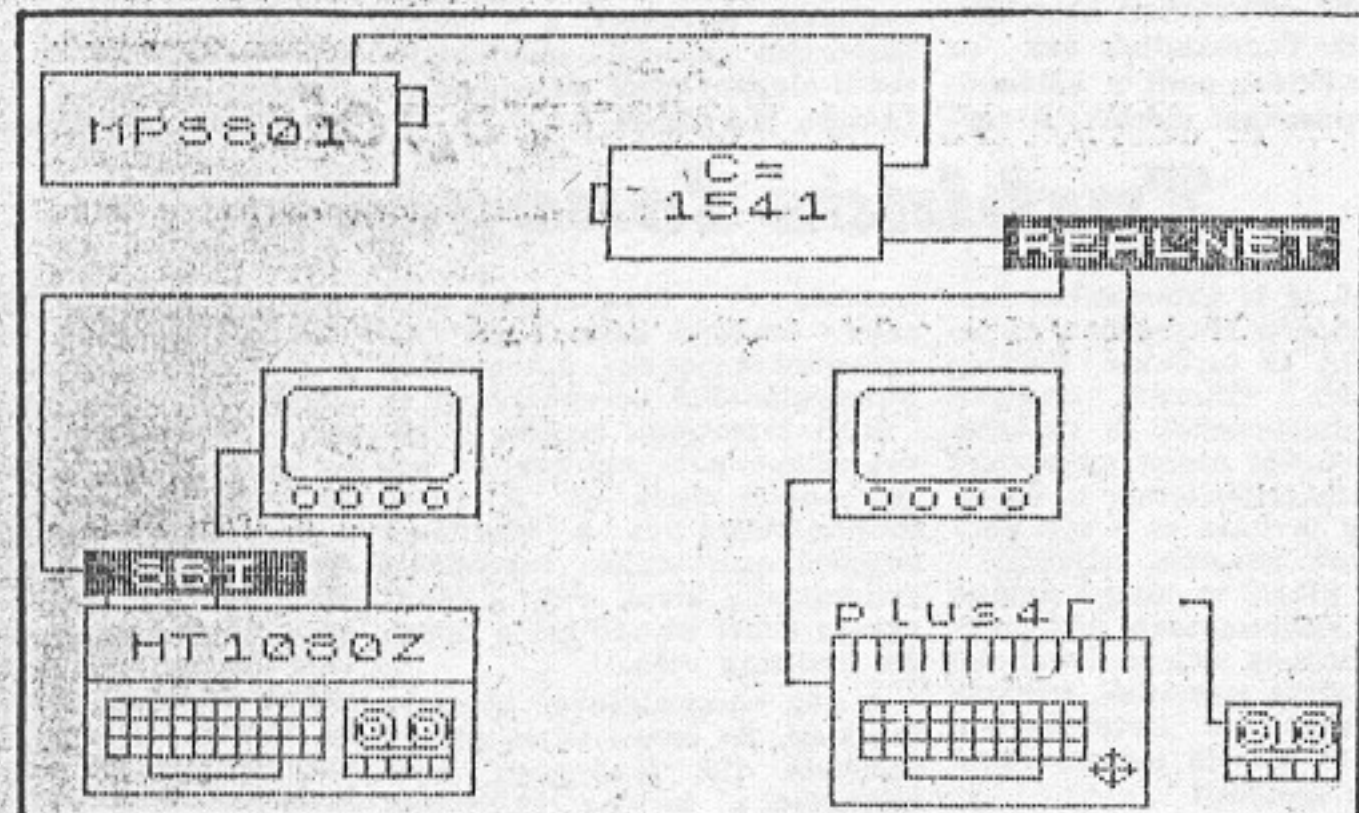
7. Fordítás „kész”, ellenőrzés, utánalgatás

8. Fordító törlése DELETE 60000—60400

9. Az átirított program rögzítése SAVE „NEV”, 8

A tapasztalatok szerint az egy menetben fordítható átlagos igényességű szótárprogram, vagy egyenletmegoldást, fizikai feladatokat gyakoroltató program minden nyolcadik-tizedik byte-ját fordítja át a program.

Sz. L. J.



A program áttöltéséhez a vázlat szerinti hardver kiegészítések zömében szükségesek. A HT 1080Z bővítő csatlakozójára helyezett soros busz interfész biztosítja a Commodore—1541 mágneslemez tárolójának HT általi működését. A REALNET más számítógép összekapcsolási feladatokban is hasznos lehet. Segítségével biztosítható, hogy akár 6 db számítógép használjon 1 db mágneslemez tárolót, illetve nyomtatót.

A Commodore számítógépbe töltött HT-program olvashatatlan. A program futtatható formába való hozatala általában 2-3 menetben végezhető el. Vannak programok, amelyek az első menetű fordítás után már működőképesek, és vannak olyanok, amelyek átfordítása mechanikus átirással nem is lehetséges.

A programok kategóriái e szempontból:

1. egy menetben a mellékelt programmal futtatható alakban kapjuk meg azokat a programokat, melyek PRINT, INPUT, GOTO, GOSUB utasításokat, függvényeket és matematikai műveleteket tartalmaznak,

2. képernyő-kiírással kapcsolatos PRINT, a TAB (vagy grafikus utasítások)

hatjuk. A két gép tokenjeinek összehasonlítása, egybevetése sugallja a megoldást.

Az átirást természetesen csak olyan programok esetében érdemes elvégezni, amelyek kipróbáltak jól működnek. Igen fura dolgok adódhatnak szintaktikai hibás programok átirása esetén. A szintaktikai elvárások esetenként amúgy is eltérnek a két gép esetében. Pl. REM, THEN kulcsszavak rövidítése. Az itt közölt program csak egy az egybe fordít, így nincs szükség a kapcsoló címek átirására, viszont gyakrabban teszi szükségessé a kézi utánigazítást.

A programsorok hosszának változását eredményezi, sőt szerkezetében más lesz az a sor, mely INKEY\$ kulcszót tartalmaz. Az IF INKEY\$="" THEN GOTO sort tartalmazó programok fordítását az egyes „programozók” stílusát figyelembe vevő szubrutin beépítésével végeztethetjük el.

A fordítás menete:

1. HT-program behívása CLOAD
2. HT-program lemeze SAVE „NEV”, 8

```
60000 REM C16-BA LEMEZROL BEOLVASOTT
60010 REM HT PROGRAM FORDITO V1.1
60020 V=PEEK(43)+256*PEEK(44)
60030 POKE 43,1 : POKE 44,16
60040 FOR L = 4101 TO V
60050 K=PEEK(L):J=33:REM ! HA NEM FORDIT
60060 IF K=34 THEN GOTO60250:"?"
60070 IF K=39 THEN J=143:GOTO60220:"=REM
60080 IF K=0 THEN L=L+4:GOTO60230
60090 IF K<130 THEN GOTO60230
60100 IF (K>134 AND K<137) THEN J=K-5
60110 IF (K>136 AND K<149) THEN J=K-4
60120 IF (K>201 AND K<229) THEN J=K-35
60130 IF (K>242 AND K<251) THEN J=K-48
60140 IF (K>186 AND K<190) THEN J=K-25
60150 IF K=132 THEN J=232: REM CLS
60160 IF K=149 THEN J=213: REM ELSE
60170 IF K=161 THEN J=145: REM ON
60180 IF K=178 THEN J=153: REM PRINT
60190 IF K=184 THEN J=156: REM CLEAR
60200 IF K=191 THEN J=251: REM USING
60210 IF K=201 THEN GOTO60290: INKEY$
60220 POKE L,J :PRINT L/J
60230 NEXT:PRINT"MELESO MENET KESZ !"
60240 END
60250 REM MASOKAKOROM KOZT UGRIK
60260 DO:L=L+1:K=PEEK(L)
60270 IF K=34 OR K=0 THEN EXIT
60280 LOOP:GOTO 60080
60290 F=L:X=1:GOSUB 60390
60300 IFPEEK(L)=213:GOTO60310:ELSE 60320
60310 J=33:L=F:GOTO60220:INKEY$=NEM MEGY
60320 REM X$=INKEY$
60330 L=F:X=-1
60340 FOR I=1 TO 3:GOSUB 60390:NEXT
60350 Z=PEEK(L):POKE L,161:POKE L+1,Z
60360 POKE L+2,36:POKE L+3,32
60370 FOR I=L+4 TO F:POKE I,58:NEXT
60380 L=I-1:GOTO 60230
60390 DO:L=L+X:IF PEEK(L)>32 THEN EXIT
60400 LOOP:RETURN:REM SPACET ATLEP
```

Szénből folyékony üzemanyag

Az angol Országos Szénhivatal (National Coal Board=NCB) 1987-ben egy kísérleti telepet helyez üzembe, amelyben naponta 2,5 t szenet dolgoznak fel és cseppfolyósítanak. A tervezett telepet a kilencvenes években kereskedelmi kapacitásúvá fejlesztik és 1 t szénből 225 l petróleum és 360 liter más üzemanyag kivonását teszi lehetővé. Ezek a számok az NCB kutatólaboratóriumában végzett kísérleteken alapulnak, ahol az LSE-névezett eljárást kifejlesztették. Bár az utóbbi években a különböző országokban számos eltérő szénecseppfolyósítási módszert dolgoztak ki, az NCB-eljárással állítólag kétszer annyi folyadék termelhető ki 1 t szénből, mint egyéb ismert rendszerekkel.

Az — egy nagy szénbánya közelében — építendő kísérleti telepet felhasználják annak bizonyítására, hogy az eljárás működjön megvalósítható körülmények között. Az eljárás során az olajjal kevert szénből az eljárás gazdaságosságát feloldott szén 50%-át.

meg tudják határozni, a termékek belsőegésű motorok üzemanyagává alakíthatók át.

A teljesen integrált telep főbb egységei a következők: porszénnek az eljárásból származó oldóolajjal való iszaposításra szolgáló berendezés, széniszap szuszpenzióérlelő tartály, az ásványi anyagokkal elegyített szén elkülönítő berendezés, a hasznosítható anyagokat hidrogénező és hidrokraakoló berendezés, valamint termékeket koksizáló részleg elektrodakosz gyártására.

A gyártási folyamat a következő műveletekből áll: a) szénőrlés, b) az őrlt szén optimális nedves tartalmúvá szárítása, c) visszakeringezett olajjal szénszap szuszpenzió kialakítása, ahol 400...450 °C közötti hőmérsékleten extrahálják, e) az át nem alakított szénüledékeket szűrővel elkülönítik, amikor a feloldott szénből annyi oldatkivonatot kapnak, amely tömeg szerint elérheti a

A kivont oldatot desztillálással két frakcióra bontják: a könnyebb frakciót enyhén hidrogénbeve — oldóképeségének helyreállítására — visszakeringetik az iszaposító egységbe. A nehezebb frakciót kb. 200 bar nyomással és 450 °C-ig terjedő hőmérsékleteken hidrogénezik és krakkolják. A hidrokraakolt anyagokat frakcionálva állítják elő. Az oldószert és szurokületéket visszakeringetik, amelyekből a koksizolóban elektrodakosz gyártható.

A másodlagos termékekből hagyományos átalakító eljárásokkal könnyen vonható ki benzín és egyéb motorüzemanyag, pl. dízelolaj.

A teleppel kapcsolatos munkálatokat, miután az átalakítás gyakorlati lehetőségét meghatározták, 1984-ben kezdték meg az NCB mérnökei. Együttműködve a Babcock Woodal-Duckham vállalat szakembereivel. Részletes műszaki specifikációt és létesítési programot, valamint létesítmény-építési költségbecslést készítettek.

A program szerint a kísérleti vizsgálat 3 esztendőig tart, ez esetleg 2 esztendővel meghosszabbodik. Az üzembeállítás utáni első esztendőben kiértékelik az integrált rendszer összes műveletét. A második évben 2000 óráig fogják üzemeltetni a telepet azzal a céllal, hogy megállapítsák a megnevezett bitumenes szén speciális műveleti követelményeit.

A harmadik esztendőben más típusú szeneket vagy ligniteket szándékoznak cseppfolyósítani, és ha szükséges, akkor a negyedik és ötödik évben is folytatják a vizsgálatokat.

A kísérleti telep üzemeltetésével párhuzamosan az NCB kutatói a szükség szerint módosítják az eljárást úgy, hogy a telepen olyan párlatot állítsanak elő, amelyek a gépkocsimotorokban jelenleg használatos benzín és dízelolaj üzemanyagokká alakíthatók át, és azzal egyenértékűek. A finomítási eljárás módosításával lehetséges olyan benzín gyártása, amely kielégíti a várható jövőbeli környezetvédelmi előírásokat.

Nyomoz a számítógép

Ha valahol a vízben mérgező anyagok tűntek fel, eddig többnyire nehéz volt megállapítani azt, hogy honnan származnak azok. Fráankfurt am Mainban újabban a Siemens által kifejlesztett számítógép nyomoz a környezetszennyezők után. A számítógép memóriájába 123 vegyi anyag jellemzőit tárolják. A memóriában elraktározták, milyen vállalat és milyen berendezés milyen típusú szennyezőanyagot állít elő és üríthet ki a szennyvízcsatorna-hálózatba.

A vízben megjelenő szennyezés esetén az adatbank segítségével másodpercek alatt megállapítható, hogy az azonosított szennyezőanyaggal melyik cég hozható kapcsolatba. Így módon azután a környezetszennyezőnek nagyon könnyű a nyomára jutni. Ráadásul ez a megoldás az eddignél gyorsabb segítő beavatkozást is lehetővé tesz.

µC·Iskolaszámítógép-programok

A MESTERSÉGES INTELLIGENCIA ELEMEI

A számítógépeket megjelenésükkel egy időben egy sor, korábban csak élőlényre, elsősorban emberi szellemi tulajdonságra, értékre vonatkozó jelzővel ruházták fel a köznyelvben. Így találkozhatunk az okos, sokat tudó, gondolkodó, sőt intelligens jelzővel.

Lapunk hasábjain is több cikket olvashattunk a mesterséges intelligenciáról. (Meggjegyezni kívánjuk, hogy a különböző tudományterületek képviselői eltérően értelmezik az intelligenciát. E vitába nem kívánunk bekapcsolódni, hanem követjük a számítógépekkel kapcsolatban e témában kialakult álláspontot.)

Az emberi — született és tanult — intelligenciát és a gondolkodást agyunk, máig nem minden részletében tisztázott működési mechanizmusával, mintegy 10 milliárd agysejtjével biztosítja.

A számítógépekkel — az iskolákban található méretű személyi számítógépekkel — legfeljebb megközelíthetjük modellezni a gondolkodás, a tanulás, a következtetés mechanizmusának néhány részletét. A mesterséges intelligenciával foglalkozó munkák három fő területet neveznek meg, amelyek leginkább (leglátványosabban) modellezik az embe-

ri gondolkodást, tanulást, véleményalkotást. Ezek: a szakértői rendszerek, az öntanuló és a felismerő programok.

A következő számokban közlésre kerülő programok a lehető legkisebb terjedelműek, és igyekeznek látványosan bemutatni e fejezetek egy-egy részterületét. Egy-két külön jelölt program kivételével bármelyik számítógépen futnak ezek.

Előrebocsátjuk, hogy célunk az érdeklődés felkeltése hasonló problémák felismerésére, megoldására, tudván tudva azt, hogy egy „igazi” alak- vagy beszéd-felismerő program memóriagénye sokszorosan meghaladja a C-16, Plus 4, vagy HT kapacitását.

Minthogy modellezésről van szó, a közérthetőség kedvéért BASIC-nyelvű programokat adunk, ami lassabb futást eredményez, de bepillantást enged a számítógép, a program „gondolkodásának”, a probléma mesterséges intelligencia szintjén történő megoldásának menetébe.

A SZAKÉRTŐI RENDSZEREK az adatbázisukban tárolt ismeretanyag alapján segítenek a döntéshozatalban, vagy közlik a „szakvéleményt”. Egy közönséges szótár nem tekinthető intelligens segéd-

eszköznek. A szótárprogramok egyértelmű megfeleltetés alapján rendelnek az egyik halmaz egy eleméhez egy másik halmazban található párt.

Amíg egy egyjegyű szám beírása után azt betűkkel kiírva kapjuk vissza a géptől, addig csak szótárral van dolgunk. Ha azonban többjegyű szám szöveges kiírását kérjük, akkor a programba a helyiérték, a tizedespon, az előjel és bizonyos helyesírási szabályok értelmezését is be kell építeni. Amennyiben a számjegy számnévve átírását az eddigiek figyelembevételével elvégzi a program, úgy beszélhetünk kezdetleges intelligens programról, mesterséges intelligenciáról. Az 1. program számnévként kiírja az inputként beírt számot. A program fő részei:

- 10 — 70 adatfeltöltés a tömbökbe,
- 80 — 90 a szám beírása, ellenőrzése,
- 100 — 130 nulla és negatív előjel vizsgálata,
- 140 egész rész átírása,
- 170 törtrész átírása,
- 190 a számnév kiírása,
- 200 — 210 helyiérték alapján szeletel,
- 220 — 260 tizedespon és tizedes rész után a jobb oldali nullák elhagyása,
- 270 — 330 helyiérték alapján számnévvé alakít.

Hasonlóan átírást végez a 2. program, amely a hagyományos családi, rokonkapcsolatok megfigyeltetését biztosítja. Bemenő adatként szülő-, gyermek-, testvér-, házastárskapcsolat láncolatát kéri, és bizonyos határig megoldásként adja az „eredő” rokon fokot. Pl.:

házastársam szüleinek gyermeke (INPUT-ként HSG irandó) = sógor, házastársam gyerekének házastársa (INPUT-ként GHG irandó) = vő.

A program fő részei:

- 20 — 30 adattár feltöltése,
- 40 — 80 a vizsgált (keresett) rokon kapcsolat beírása,
- 90 — 110 triviális esetek vizsgálata,
- 120 — 200 összetett rokon kapcsolatok triviálissá alakítása,
- 210 — 240 következtetés kiírása.

A program mesterséges intelligenciája természetesen nem tudhatja — vagy csak igen összetett interaktív adatbevitel után ismerheti — a rokonság, a család nagyságát. Mindig feltételezi a keresett személy meglétét. Így ha a családban két gyermek van (honnan tudja a program?), akkor a testvérem testvére = testvér. A program — egy esetet kivéve — a kérdező személyre visszavezető meghatározást nem ad. Itt azonnal felismerhető a modell és a valóságos, a természetes intelligencia közti különbség. Belátható annak a sok „mellékes információnak” döntő jelentősége, amelyek birtokában gondolkodunk, ítéleteket alkotunk, döntéseket hozunk.

A szakértői rendszerek másik csoportjánál, amelyek rávezetési úton, irányított kérdésekre adott válasz alapján segítenek a döntésben, azok kérdéstárában ott kell lennie azoknak a kérdéseknek, amelyek ezeket a mellékes körülményeket is figyelembe tudják venni.

De arról a következőkben.

Sz. Lukács János

10 REM MESTERSEGES INTELL. ALAPJAI SZAMATIRO

```
20 DIMB$(9,4),V$(4):REM HA HT CLEAR 500
30 FOR I=1 TO 9:READ B$(I,1):FOR J=2TO4
40 B$(I,J)=B$(I,1):NEXT J:NEXT I
50 FOR I=1 TO 9:READ B$(I,3):NEXT I
60 FOR I=1 TO 4:READ V$(I):NEXT I
70 B$(2,4)="KETTO":B$(1,1)="" :B$(1,2)=""
80 INPUT"10000>X>-5000 X=":X$:X=VAL(X$)
90 Y=X:IF X>=10000 OR X<=-5000 GOTO 80
100 A$="" : IF X<>INT(X) THEN GOSUB 220
110 IF Y<0 THEN A$="MINUSZ ":X=ABS(X)
120 IF X<1 AND X>0 THEN A$=A$+"NULLA "
130 IF X=0 THEN A$="NULLA"
140 GOSUB 200:L=0:IF Y=INT(Y) GOTO 190
150 IF Z=2 THEN A$=LEFT$(A$,LEN(A$)-2)
160 A$=A$+" EGESZ ":X=VAL(X$):L=LEN(X$)
170 GOSUB 200:K$=LEFT$(A$,LEN(A$)-2)
180 IF RIGHT$(X$,1)="2" THEN A$=K$
190 PRINTA$:" ":V$(L):PRINT:GOTO80
200 J=1000:FOR I=1 TO 4:Z=INT(X/J)
210 GOSUB 270:J=J/10:NEXT:RETURN
220 FOR I=1 TO LEN(X$)
230 IF MID$(X$,I,1)=". " THEN K=I+1
240 NEXT I
250 IF RIGHT$(X$,1)="0" THEN
X$=LEFT$(X$,LEN(X$)-1):GOTO250
260 X$=MID$(X$,K,4):RETURN
270 X=X-Z*J:IF Z=0 THEN RETURN
280 IF J=1000 THEN A$=A$+B$(Z,1)+"EZER":
IF ABS(Y)>2000 THEN A$=A$+"-"
290 IF J=100 THEN A$=A$+B$(Z,2)+"SZAZ"
300 IF J=10 AND (Z>2 OR INT(X)>0) THEN
A$=A$+B$(Z,3)
310 K$=B$(Z,3):IF J=10 AND Z<3 AND INT(X)
=0 THEN H=LEN(K$)-2:A$=A$+LEFT$(K$,H)
320 IF J=1 THEN A$=A$+B$(Z,4)
330 RETURN
340 DATA EGY,KET,HAROM,NEGY,OT,HAT,HET,
NYOLC,KILENC,TIZEN,HUSZON,
HARMINC,NEGYVEN,OTVEN,HATVAN,
HETVEN,NYOLCVAN,KILENCVEN,TIZED,
SZAZAD,EZRED,TIZEZRED
```

1. program

10 REM MESTERSEGES INTELL. ALAPJAI ROKONSAG

```
20 CLEAR500:DIMR$(25),F$(25),N$(25):CLS
30 FORI=1TO16:READR$(I),F$(I),N$(I):NEXT
40 INPUT"AZ ILLETO NO VAGY FERFI ":N$
50 IF N$<>"N" AND N$<>"F" THEN GOTO 40
60 INPUT "AZ ILLETO ROKON S/G/T/H: ":R$
70 IF R$="" THEN 60 ELSE R=LEN(R$)
80 IF R<1 OR R>5 THEN GOTO 60
90 FOR PROBA=1TO5:J=0:FOR I=1 TO 16
100 IF R$=R$(I) THEN J=I:I=20
110 NEXT:IF J<>0 THEN GOTO 230
120 IF PR=1 THEN PRINT"EZ BONYOLULT: ";
130 PRINTR$
140 R=LEN(R$):FORI=1TOR-1:V$=MID$(R$,I,2)
150 IF V$="GT" OR V$="HG",X$="G":GOTO 260
160 IF V$="TT" OR V$="SG",X$="T":GOTO 260
170 IF V$="TS" OR V$="SH",X$="S":GOTO 260
180 IF V$="HH",X$="":GOTO 260
190 IF V$="GS",X$="H":GOTO 260
200 NEXT:NEXT PROBA
210 IF R$="" ,PRINT"EZ EN VAGYOK":GOTO 40
220 PRINT:PRINT"EZ NEM MEGY":GOTO 40
230 FORPROBA=1TO1:NEXT
240 IFN$="N",PRINTN$(J)ELSEPRINTF$(J)
250 GOTO40
260 R$=LEFT$(R$,I-1)+X$+RIGHT$(R$,R-I-1)
270 GOTO130
280 DATA SS,NAGYAPA,NAGYANYA,ST,NAGYBACSI,
NAGYNENI,SSS,DEDAPA,DEDANYA,STG,
UNOKATESTVER,UNOKATESTVER,GG,UNOKA,UNOKA,
GGG,DEDUNOKA,DEDUNOKA,GH,VO,MENY,GHS,
NASZURAM,NASZASSZONYOM,HT,SOGOR,SOGORNO
290 DATA HS,APOS,ANYOS,TH,SOGOR,SOGORNO,
TG,UNOKAOCS,UNOKAHUG,S,APA,ANYA,T,FIVER,
NOVER,H,FERJ,FELESEG,G,FIAM,LANYOM
```

2. program

µC-Iskolaszámítógép-programok

A MESTERSÉGES INTELLIGENCIA ALAPJAI

A szakértői rendszerek igen széles körű adatbázisra épülnek. Készültek és a középiskolai oktatás számára is hozzáférhetőek már olyan programok, amelyek növények, gombák meghatározását segítik elő. Vannak programok, amelyek egyes betegségek diagnózisát, mások műszaki berendezések hibáinak meghatározását adják. Az ilyen programok megírásához az adott szakterület kiváló ismerője és a számítástechnikus tudása, intelligenciája szükséges.

Itt most kétféle meghatározó-rendszert mutatunk be. Mindkettő leegyszerűsített problémakörrel modellezi a mesterséges intelligencia területét.

Az 1. program azt az utat követi, amellyel irányított kérdésekre adott válaszok alapján juthatunk el a végkövetkeztetéshez. Az ilyen programok esetében döntő fontosságú a kérdések sorozatának logikus egymásra építése és azok

kiértékelő ágakat is kell tartalmazniuk a válaszok helyességére, következetességére vonatkozóan.

A programban:

- H\$ a programba felvett gyermekbetegségek kezdőbetűi,

- L\$ az adott tünet alapján lehetséges betegségek kezdőbetűi,

- B\$ a tünetek alapján lehetséges betegségek kezdőbetűi,

- A rendelkezésre álló adatok alapján úgy is meghatározhatunk, azonosíthatunk egyedeket, hogy megvizsgáljuk az átlagos számszerűsített tulajdonságokkal, méretekkel rendelkező etalonról való eltérést.

A meghatározáshoz a minta, a vizsgált egyed és az etalon méreteit koordináta-rendszerben ábrázolhatjuk. Ha csak két jellemző tulajdonság alapján azonosítunk, akkor az ábra szerint Pithagorasz tételét használhatjuk fel az eltérés, az egyes halmazoktól való távolság

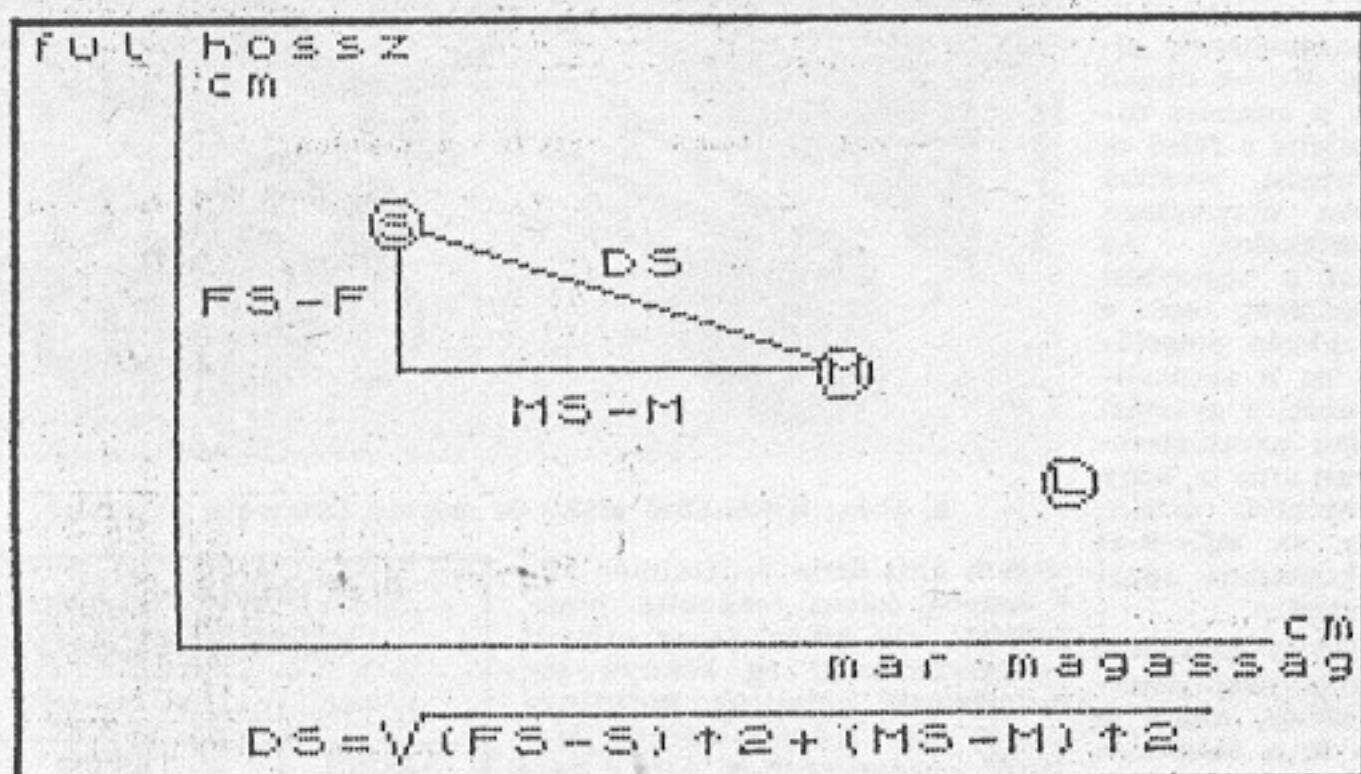
```
100 REM MESTERSEGES INTELLIGENCIA ALAPJAI / DOKI
110 CLS :H$="BKMS":FOR I=0 TO 5:READ B$(I):NEXT
120 PRINT@320,"GYEREKBETEGSEGEK DIAGNOZISA"
130 B$="" :PRINT @960,"KIUTASE VAN I/N?", :L$="BKRS"
140 GOSUB 360 :F=0 :IF A$="N" THEN GOTO 200
150 PRINT"LAPOS PIROS FOLTOK I/N?", :L$="KR":GOSUB360
160 PRINT"KIEMELKEDO, HOLYAGOS I/N?", :L$="B":GOSUB360
170 PRINT"APRO VOROS PONTOK I/N?", :L$="S":GOSUB360
180 IFF<>2,PRINT"PONTOS VALASZT KEREK":PRINT:GOTO 130
190 GOTO 210
200 PRINT"FULTO, ARC DUZZADT I/N?", :L$="M":GOSUB360
210 PRINT"MIRIGYEK DUZZADTAK I/N?", :L$="RM":GOSUB360
220 PRINT"NYELESNEL FAJDALOM I/N?", :L$="SM":GOSUB360
230 PRINT"KOHOOG I/N?", :L$="K":GOSUB360
240 PRINT"LAZA MAGAS I/N?", :L$="KS":GOSUB360
250 PRINT"ROSSZULLET, HANYAS I/N?", :L$="S":GOSUB360
260 PRINT:PRINT B$:PRINT
265 IF B$="" THEN PRINT "TAN NEM IS BETEG":GOTO 340
270 FOR I=1 TO LEN(B$):T$=MID$(B$,I,1)
280 FOR T=1 TO 5:IF MID$(H$,T,1)=T$ THEN T(T)=T(T)+1
290 NEXT T:NEXT I:V=0
300 FOR I=1 TO 5:IF T(I)>V THEN V=T(I):B=I:NEXTELSENEXT
310 F=0:FOR I=1 TO 5:IF T(I)=V THEN F=F+1:NEXTELSENEXT
320 IF F<>1,PRINT"ILYEN TUNETEK MELLETT BIZONYTALAN"
330 PRINT"VALOSZINULEG":CHR$(23)B$(B):PRINT
340 IF INKEY$="" THEN RUN ELSE GOTO 340
350 END
360 A$=INKEY$:IF A$<>"I" AND A$<>"N" THEN GOTO 360
370 IF A$="N" THEN F=F+1:L$=""
380 PRINT A$:B$=B$+L$:PRINT L$:RETURN
390 DATA MEGHATAROZHATATLAN
400 DATA BARANYHIMLO,KANYARO,MUMPSZ,RUBEOLA,SKARLAT
```

1. program

10 REM MESTERSEGES INTELL. ALAPJAI OSZVER

```
20 CLS:PRINT"LO VAGY SZAMAR?":PRINT
30 PRINT:INPUT "MAR MAGASSAG :CM=":M
40 IF M<80 THEN PRINT"EZ TAN PONILO":
  "VAGY KIFEJLETLEN EGYED":GOTO 30
50 IF M>190 THEN PRINT"IRREALIS":GOTO 30
60 INPUT "FULENEK HOSSZA CM=":F
70 IF F<5 OR F>30 THEN PRINT
  "IRREALIS ERTEK":GOTO 60
80 DS = SQR((M-100)^2+(F-20)^2)
90 DL = SQR((M-155)^2+(F-12)^2)
100 IF DS<18 THEN PRINT "SZAMAR":GOTO 30
110 IF DL<25 THEN PRINT "LO":GOTO 30
120 PRINT"VALOSZINULEG":X=DL-DS
130 IF ABS(X)<15,PRINT "OSZVER":GOTO 30
140 IF X<0 THEN PRINT "LO"
150 IF X>0 THEN PRINT "SZAMAR"
160 GOTO30
```

2. program



szabatos megfogalmazása. A döntéshozatalt, a végső definíció megfogalmazását döntően befolyásolhatja a programot kezelő szakmai tudása, szubjektív megítélése, hiszen pl. egy színárnyalat fontos lehet egy növényfajta meghatározásakor, de a színárnyalat megítélését a program kezelőjére bízjuk!

Az 1. program a fertőző gyermekbetegségek meghatározását végzi el. Meg kell jegyezni, hogy a program szemléltetésre szolgál, ennyi és ilyen mélységű (felületességű) kérdés alapján senki se kezdje el a gyógykezelést!

Egyetlen orvos sem ad telefonon keresztül diagnózist, még akkor sem, ha egy sor „mellékes” információval rendelkezik a páciens környezetéről, korábbi egészségi állapotáról. Egy korrekt — megbízható — szakértői rendszer kérdéssorainak természetesen olyan kérdéseket is kell tartalmaznia, amelyek az említett körülményekre is kitérnek, sőt ellenőrző,

meghatározásához. De ha rendelkezésünkre áll további értékelhető tulajdonság, akkor térben kell meghatározni a két pont közötti távolságot.

A 2. program igen leegyszerűsítve a lovak és szamarak küllemi jellemzőit használja fel egy hazánkban található „lőszerrő állat” azonosítására.

Kiindulásként ismert, hogy az „átlagos”

	számár	összver	ló
marmagassága	85...105		145...170 cm
fülhossza	20...25		10...15 cm
testtömege	250 400...500	500...700 kg	

Az említett probléma megoldására az itt bemutatott 2. program nevelésnek tűnhet, de a hangok azonosítására szolgáló formáns frekvenciákat vizsgáló program hasonló elveken épül fel. Igaz, az ilyen programok már nem BASIC-nyelven íródnak, és az adatokat sem billentyűzetről közlik a programmal.

Sz. Lukács János

140 ÉVES A PEST-SZOLNOKI VASÚTVONAL

Luk nem gondoskodott, sőt ötnak a korporsót is a város csináltatta, a többiek korporsójának árát a munkások magok tették össze. Ezen betegek közül sokan később koldulgáltak a városban, sokan pedig télen által is vagy egyes lakosoknál, vagy a kertek között istállóknál találtak menedéket. En részemről a Vármegyének is tettem e részben jelentést, de már ekkor a betegség szűnni kezdett, és hihetőleg a tett intézkedés következtében később a betegek Irsáról Pestre szállítottak. A vasútra vonatkozólag más körülményekről tudomásom nincs azt azonban hallottam, hogy a munkások... jelenleg rossz fizetésről panaszkodnak, a munka béréből a vállalkozók ahol lehet, elhúznak.”

Szítányi Ullmann Mór idegeit annyira megviselték a hónapokon át tartó kihallgatások és szembesítések, hogy 1846. május 17-én, a vasúttársaság közgyűlésén igazgatói tagságáról lemondott. A vizsgálat végül lezárult, de a sok tisztázhatatlan ellentmondás miatt senkit nem vontak felelősségre.

A vasútvonal építése közben az elkészült részeket anyagszállító vonatok köz-

lekedtek, amelyekkel több baleset is történt. Az egyik eset 1847. augusztusban történt. A Nemzeti Újság írta: „A pest-szolnoki vaspályavonalon múlt napokban ismét szerencsétlenség történt. Egy mozdony, mely Vecsésre vegyes bizottmányt szállított, visszatérte alkalmával egy láncot elvesztett. Rögtön rá egy vonat üres kocsikkal érkezett, melyen a vasúti igazgatóság legszigorúbb parancsa ellenére több ember foglalta helyét. A lánc, amellyel a sineken feküdt, valódi alaborként megakasztani igyekezők a mozdonyt, de ez hevesen keresztül robotott rajta, s a véletlen lökés következtében az emberek mind lehulltak a kocsikról, s az árkolatba gurultak. Közülük három egyén meghalt, többen pedig tetemesen megsérültek.”

A vasútvonal átadása

A teljes vonal megnyitását megelőzte 1847. augusztus 5-én a pest-kőbánya vasútrész átadása. Az ünnepélyes megnyitó vendégei a kőbányai állomástól félórát gyalogútra levő sörkertben mulattak, majd a délutáni hatalmas felhőszakadás után, lábbelit és ruhákat összeszórva tértek vissza az állomásra, illetve Pestre.

Szeptember 1-jéig hetenként három napon volt személyszállítás Pestről Kőbányára 15, 12 és 10 krajcárért az I., II. és III. osztályon.

1847. augusztus végén megtörtént a pályaszakaszhivatalos műszaki átadása. A Bécsből érkezett szakértők és a vasúttársaság igazgatósága jelenlétében megállapították, hogy a pálya a nyilvános forgalomnak átadható.

A vasútvonal megnyitására 1847. szeptember 1-jén ünnepélyes külsőségek között került sor. Ünnepi beszédet gróf Zichy Ferenc vasúti igazgató és István főherceg, a megnyitó díszvendége mondott. A 600 főnyi vendégsereg zeneszó, ágyú-, lövések és a többeszes tömeg éljenzése kíséretében 1/4 9 körül indult el Szolnokra. Minden közbeeső állomáson — Kőbánya (alsó), Vecsés, Üllő, Mór, Pils, Albert-Irsa, Bercel, Cegléd, Abony — rövid időre megálltak, hogy fogadják az előjárók, illetve a lelkesek köszöntését. A virágokkal feldíszített állomásépületek körül az egész környék apraja-nagyja tolongott, hogy az ünnepi menetet megtekintse. Szolnokra fél 12-kor futott be a vonat, amelyet az újságok szerint az „István” és „Haza” nevű mozdonyok vontattak, előttük vonalkémlelőként a „Nádor”

haladt. A vendégek díszében vettek részt, amelyre a vasúttársaság 14 ezer pft-ot költött. A korabeli újságok néhány napos késéssel, terjedelmesen számoltak be az eseményről.

Az új vasútvonal gyorsan népszerű lett, ezt bizonyítja a Nemzeti Újság 1847. szeptember 10-i számában megjelent hír is: „Mióta a pest-szolnoki vaspályavonal megnyitott, annyi termékek mátha és olly sok utazó szállítatik rajta, hogy legközelebből is a szállítandó tárgyak halmaza miatt 7 óra helyett csak 8 órakor indulhatott csak meg a vonat.”

Szeptembertől a menetrend a következő volt: Pestről Szolnokra reggel 7 óra, Szolnokról Pestre du. fél 3. Pestről Kőbányára hétfőn és csütörtökön du. 4 óra, Kőbányáról Pestre este fél 8 óra. A menetidő Pesttől Szolnokra 2 és fél óra volt.

A vonal egyvágányúra épült, a második vágány Ceglédig 22 év múlva készült el. A talpfák trapéz alakúak, talprészük 12, magasságuk 6, felső lapjuk 5, sínillesztésnél 8 1/2 hüvelyk, hosszuk 7 1/2 láb volt. A sínzálak hossza 5,69 m, tömege méterenként 26,6 kg volt. Egy sínzál alatt 6 db talpa volt elhelyezve, az ágyazat bányakavics volt.

Németh László

ALAKFELISMERÉS

Az előzőekben látott programok — a rendelkezésükre álló információk alapján — megneveztek valamit. Ha az információt vizuális úton nyeri (nyerheti) a számítógép, és az így szerzett ismeretek birtokában közli az eredményt, akkor általában alakfelismerő programról beszélhetünk.

Az ábrán a hétszegmenses kijelző leolvasására szolgáló berendezés elvi elrendezését láthatjuk. A használható modellhez az intenzív fénykielbocsátáshoz és az érzékelő fotodiódák árnyékolhatóságához a számkijelző helyett fénycsövek-ből összeállított eszköz javasolható.

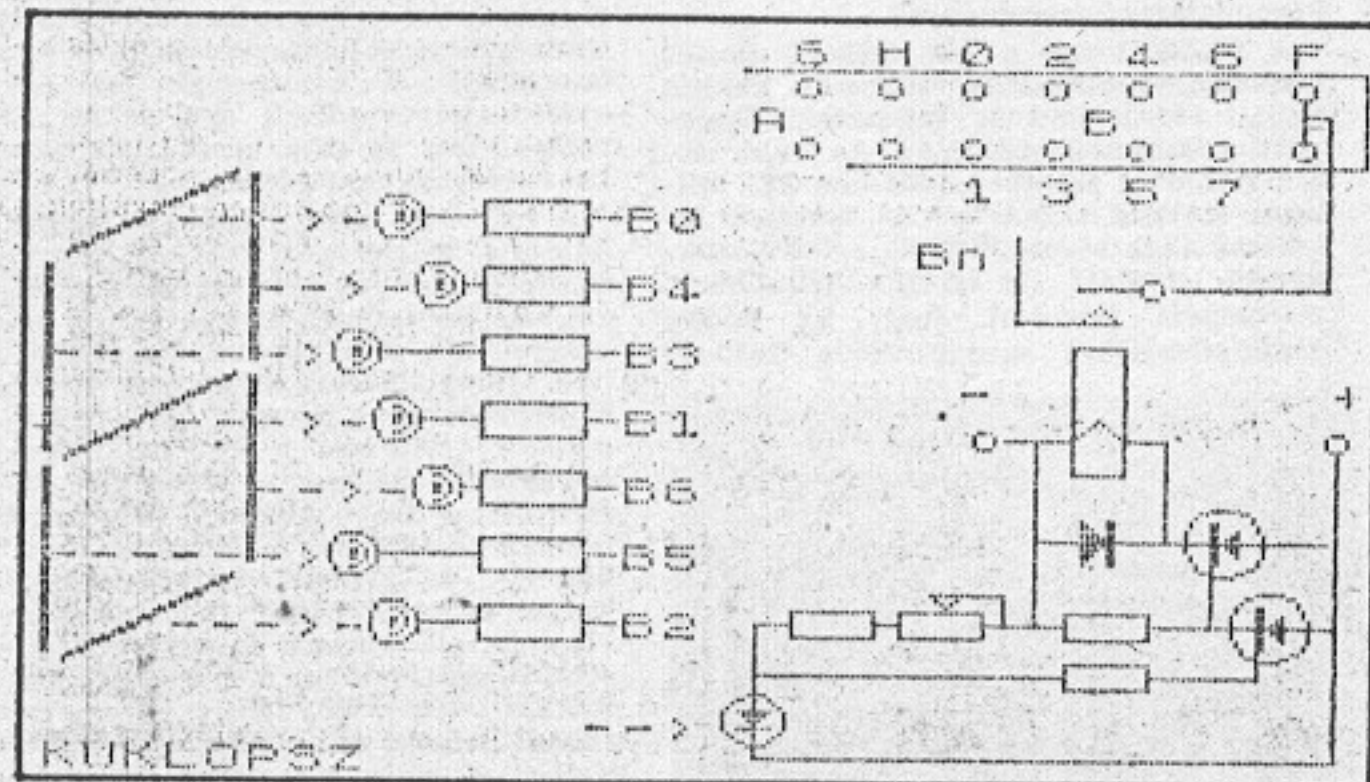
Ebben a berendezésben nem a program, nem az információ kapun keresztül

A második programban az alakfelismeréshez a betűket a képernyő kijelölt 5×7-es képpontjára kell felrajzolni.

A 7.—12. programsorok között mozgathatjuk a „?”-et. Az I vagy N betűvel kérhetünk az adott rászterpontra satírozást. Törölést a felfelé és balra nyíllal végezhethetünk.

A program 13.—16. sorai soronként letapogatják az alakzatot, és a betöltött pontok alapján kettes számrendszerbeli számbázisra fordítva minden sort egy számjeggyel reprezentálnak. (Ha szükséges, a betűt a mező bal szélére helyezi át, mert csak a bal szélre rendezett betűt képes felismerni, 27. sor.)

A 17.—21. programsorokon soronként le-



Elvi elrendezés hétszegmenses kijelző leolvasására

beolvasása a nehéz, hanem a megfelelő, leolvasni kívánt számkijelző összeállítása és működtetése. Az 1. program a HT-1080Z számítógép kapuját kérdezi le a 70-es soron. A B kaput beolvasásra a 20. soron állítottuk. A kapuról érkező érték (az egyes szegmensek aktív állapotát érzékelő diódák a kettes számrendszer hatványainak megfelelő értéket biztosítanak) alapján a 30.—60. sorok által meghatározott választék alapján írja ki a program a felismert jelet a képernyőre.

Ez a választék elég szegényes. A betűk, karakterek (a számítógép teljes jelkészlete) megkülönböztethető kiírására azokat különböző rászterkiosztású mezőkre építik fel.

tapogatja a betűt és a T\$ tömb megfelelő eleméből kiválaszthatja a lehetséges B\$-t.

Ez a program tehát a memóriájában levő 26 különböző A betű bármelyikét A betűnek azonosítja, de más variációban már nem ismeri fel azt.

A 30.—32. sorok a vízszintes letapogatás után még egy, a betűkre jellemző helyen függőleges szeletelést és vizsgálatot végeznek, de ez a rész is inkább a nem kívánt alakzatok kiszűrését, mintsem a korábbi szakaszban fel nem ismert betűk azonosítását végzi. Azaz ennek a programnak megfelelő mesterséges intelligenciából is hiányzik az emberre jellemző absztrakciós képessége.

Sz. Lukács János

```
10 REM MESTERSEGES INTELL. ALAPJAI
    KUKLOPSZ
20 DIM B$(85):CLS:OUT31,7:OUT30,1:OUT31,15
30 FOR I=1TO85:READ B$(I):NEXT I:PRINT""
40 DATA F,L,C,E,,,,,2,,,,,
50 DATA ,,,5,1,7,,,,,3,,4,9,,9,,
60 DATA ,,,G,6,,,,,J,,,,,H,A,U,0,,8
70 B=255-INP(31)
80 IF B=BE OR B>127 OR B<42 THEN GOTO 70
90 PRINT B$(B-42):BE=B:GOTO 70
```

1. program

```
1 REM MESTERSEGES INTELL. ALAPJAI / ABC
2 CLS:PRINT @330,"KIS TURELMET":CLEAR 1000
3 DIM T$(7,31):DEFINT A-Z:FOR F=0 TO 6
4 FORV=31TO1STEP-1:READT$(F,V):NEXTV,F
5 CLS:FORF=0TO6:FORV=0TO4:PRINT@F*64+V,".":
6 NEXT V,F:FOR F=0TO6:FOR V=0TO4
7 P=F*64+V:PRINT @P,"?":
8 A$=INKEY$:IF A$="" THEN GOTO7
9 V=V+(A$=CHR$(8) AND V>0):
10 F=F+(A$=CHR$(91) AND F>0)
11 IF A$<>"I" AND A$<>"N",PRINT@P,".":GOTO 7
```

```
10 IF A$="N" THEN PRINT@P,".":B(F,V)=0
11 IF A$="I" THEN PRINT@P,CHR$(191):B(F,V)=1
12 NEXT:NEXT:L$="ABCDEFGH IJ"
13 FORF=0TO6:E(F)=0:FORV=0TO4:
14 E(F)=E(F)+B(F,V)*2[V:F(V)=0:NEXT V
15 IF E(F)=0 THEN GOTO 29 ELSE NEXT F
16 FOR V=0TO4:FOR F=0TO6:F(V)=F(V)+B(F,V):
17 NEXT F:NEXT V
18 S=0:FOR F=0TO6:S=S+B(F,0):NEXT:
19 IF S=0 THEN GOSUB 27:GOTO 13
20 FOR F=0TO6:S$="":FOR X=1TO10:
21 B$=MID$(T$(F,E(F)),X,1)
22 IF MID$(L$,X,1)=B$,S$=S$+B$ELSE S$=S$+" "
23 NEXT X:L$=S$:PRINT@F*64+8,L$:NEXT F:S=0
24 FOR I=1 TO 10:V$=MID$(L$,I,1):
25 IF V$<>" " THEN S=S+1:GOTO 30
26 NEXT:IFS<1,L$="?.?.?.?.?.?"
27 PRINT@F*64+8,L$:PRINT@640,"";
28 IF S<1 THEN PRINT"EZ NINCS A MEMORIABAN"
29 IF S>1 THEN PRINT"PROGRAM PONTOSITANDO"
30 PRINT@768,"T LENYOMASARA UJ PROBA."
31 A$=INKEY$:IF A$<>"T",GOTO26 ELSE GOTO5
32 FOR F=0TO6:FOR V=1TO5:B(F,V-1)=B(F,V):
33 P=64*F+V-1:IF B(F,V-1)=1 THEN PRINT @P,
34 CHR$(191):ELSE PRINT @P,".":
35 NEXTV,F:RETURN:REM BALRA LEPTETETT
36 PRINT @640,"NINCS KITOLTVE":GOTO 25
37 W=I-1:IF ((V$="B" OR V$="E") AND F(1)<>3) OR
38 (V$="A" OR V$="F") AND F(1)<>2 THEN GOSUB 34
39 IF V$="A" AND (F(0)<>F(LOG(E(6)-1)/LOG(2)))
40 OR (V$="I" AND ((E(0)=7 AND F(1)<>7) OR (E(0)=
41 31 AND F(2)<>7) OR (E(0)=1 AND (F(0)<>7 OR F(1)
42 <>0))) THEN GOSUB 34
43 IF V$="H" AND (F(0)<>7 OR F(1)<>1 OR LOG(E(0)
44 -1)/LOG(2)<>7) THEN GOSUB 34
45 GOTO21
46 L$=LEFT$(L$,W)+" "+RIGHT$(L$,9-W):S=S-1
47 RETURN
```

```
36 DATA ABC.EFG.IJ,..C...G...J.....J...
    ..J.....H.....J
    ABCDEFG..J.A.C...G.IJ...C...J...
    ..H.....J,ABCEFG.IJ.A.C...G..J,
    .....H...A.C...J,BCDEFG...A.C...
    .....I...:1.SOR
37 DATA .....CDEF.....ABCEFGH...
    ..J.....,ABC,ABCEFGH.....J,
    ..ABCD...H.....IJ.....I,
    ..C.EFG.I...:2.SOR
38 DATA A...EF.H...EF.....B.D,
    ABCD...H.....J,AB..EF.H...EF,
    ..B,AB.D...H.....J,AB..EF.H...E
    F,AB...H.....IJ.....EF...EF..I
    ..C.EFG.I...:3.SOR
39 DATA A...EF.H..B..EF.....G.....G
    .....D,AB.D...H.....J,AB..EF.H
    ..B..EF.....G...AB.D...H.....J,A
    B..EF.H..B..EF.A...H.....IJ..B..E
    F.....EF..I...C.EFG.I...:4.SOR
40 DATA A...EF.H...EF.....G.....G
    .....B.D,ABCD...H.....J,A...EF.H
    ..EF.....G...B,AB.D...H.....J
    ,A...EF.H...EF,AB...H.....IJ...
    ..EF.....EF..I...C.EF..I...:5.SOR
41 DATA .....BC.E.ABCDE.GH.J.....
    ..BC,ABCD...GH.J...ABCD...H.J.....I
    .....EF..I...C.EF..I...:6.SOR
42 DATA .BC.E.G.IJ...C.....A.....H
    ..BCDE.G..J...C...G..J...C...A.....H,
    ..B.DE...IJ...C...G..J,A...H...C..B.DE,
    ..C..F...J.....F..I...:7.SOR
```

2. program