

Klicova slova jazyka BASIC-G pro CONSUL 2717:

```

ABS -absolutni hodnota promenne; PRINT ABS(-2)...=2
ADR -adresa umisteni promenne v pameti;
    PRINT ADR(X);ADR(DM)...=9255 9249
AND -logicky soucin binarni; 14 AND 7=6;
APEEK -obsah dvou po sobe jdoucich bytu v pameti;
    PRINT APEEK('8001')...=-32768
ASC -dekadicka hodnota 1.znaku retezce;
    XM='Abc':PRINT ASC(XM)...=85
AT -urcuje radek(0-24) a sloupec(0-47) v PRINT AT;
    PRINT AT 4,10;TM;AT 8,10;UM;AT 8,20;VM
ATN -arcustangens cisla; ATN(1)=.785398;
AUTO -automaticke cislovani radku pri psani programu;
    AUTO200,3 - od radku 200 s krokem 3: 203,206,...
AXES -vykresleni os v pruseciku souradnic;
    AXES0,0:AXES255,242 - vykresli ramecek;

BEEP -zvukovy signal (pipnuti);
BIT -hodnota bitu B(0-7) promenne X(0-255);
    X=5:BIT X,0=1:BIT X,1=0:BIT X,2=1:...
BMOVE -presun pera na sloupec S(0-47) a radek R(0-242);
    BMOVE S,R:...BMOVE0,0 - pozice vlevo nahore;
BPLOT -vykresleni obrazce zadaného retezcem bytu;
    TM=CHR$(A)+CHR$(B)+CHR$(C)+CHR$(D)+...
    BPLOT TM,2 -vykresli A+B,pod nimi C+D, atd.

CLEAR -vynulovani vseh promennych v programu;
CODE -volani podprogramu ve strojovem kodu zadanem
    jako retezec sestnactkovych bytu,koncici C9(=RET);
    PM='3E41C93234C1CD0085C9':CODE PM -vypise A;
CONTROL -nastaveni vstup/vystupniho kanalu K(0-7),jeho
    registru R(0-x) uvedenymi daty: CONTROL4,3;132,5
    nastavi: kanal 4, port B -vystup, bit PC2=1
CONT -pokracovani programu zastaveného povelom STOP;
COS -hodnota cosinu uhlu v radianech/stupnich;
    RAD: COS(3.14)...=-1:DEG: COS(180)...=-1;

DATA -seznam dat pro cteni povelom READ;
    DATA5,3.14159,Konec:READA,B,CM...A=5,CM='Konec'
DEF FNC -definice uzivatelske funkce;
    DEF FNC A(X)=(X-2):Y=FNC A(2)...=4
DEG -uhly pro goniometrické funkce jsou ve stupnich;
DIM -vymezení číselného/retezcevého pole v pameti;
    DIM A(2 3)-dvourozmerne pole o 3*4 prvcich;
    DIM AM(9)-jednorozmerne pole o 10 prvcich;
DISP -zobrazení výrazu/textu v dialogovém radku;
    DISP 2*5.55: DISP KM: DISP TAB(10);A/B
DLOAD -ctení dat z magnetofonu do dimenzovaného pole;
    DIMAM(0):DLOAD 01;AM(0) -soubor 01 do AM(0);
DSAVE -zaznam souboru dat na magnetofon z daného pole;
    DSAVE 02;CM(2)'Vysledky'...Nazev souboru;

END -ukončení programu;
ENTER -vstup dat z kanalu a registru KR do seznamu S;
    ENTER401;X - vstup dat do X z reg.01 4.kanalu;
EXP -mocnina zakladu prirozenych logaritmu;
    E=2.71828....EXP(2)=E-2=7.38906

FILL -vyplnění oblasti o sirce(0-255) a vysce(0-242)
    binarnim profilem M(0-83): FILL S,V;M
    FILL10,20;5 -profil 0101 široky 10 a vysoky 20;
FNC -vyvolání definované uživatelské funkce; Y=FNC A(X)
FRE -velikost volné pameti pro program nebo retezcevé
    promenne: FRE(X)<=14830, FRE(XM)<=3841;
FOR_TO -[STEP_1]-zahlaví program.cyklu; provede opakované
    povelý mezi FOR a NEXT,zvysuje promennou o STEP
    FOR I=0TO9:PRINT I:NEXT I-vypise hodnoty promenne;
    není-li uveden STEP, zvysuje se promenna o +1;

```

GCLEAR-vymazani stinitka obrazovky;

GOSUB -volani podprogramu na radku cislo/vyraz;
GOSUB 120 nebo: V=5*X:GOSUB V

GOTO -nepodminený skok na radek cislo/vyraz;
GOTO 240 nebo: R=10:GOTO R

HEX# -prevod desitkového vyrazu na šestnactkove cislo;
HEX#(256)...=100; HEX#(-1)...=FFFF

CHECK -overeni kontrolního součtu nahraného programu
prikazem SAVE: CHECK 02...02=cislo nahravky

CHR# -prevod desitkového vyrazu (0-255) na znak ASCII;
CHR#(85)...=A, neznáme znaky se zobrazí jako '.'

IF_THEN -povel podmínený splněním vyrazu za IF; není-li
vyraz splněn, pokračuje program na dalším radku;
IF Q=0 OR Q>7 THEN PRINT Q:GOTO350

INK -ridi mod.vypisu v PRINT: INK(1)-polojas,
INK(2)-blikani, INK(4)-inverse, INK(5)-inv.+polojas

INKEY -nacteni hodnoty (0-11) klavesy klice F0-F11;

INP -prirazení bytu od portu (0-255); INP(7);

INPUT -cte znaky z klavesnice do seznamu promenných;
INPUT A,Z#,...-kontroluje syntax u cis.promenne;

INT -zaokrouhli dolu desetinné cislo; INT(-2.8)=-3;

LABEL -rozkreslení vyrazu v násobcích sirky S(1-51)
a vysky V(1-34): LABELS,V;T# LABEL2,4;9;X; 'E'

LEFT# -omezení řetězce na určený počet znaku zleva;
R#='abcde':LEFT#(R#,2)='ab'

LEN -určení délky (počtu znaku) řetězcové promenne;
X#='qwert':LEN(X#)=5

(LET) -nepovinný prikaz prirazení hodnoty promenne;

LIST -vypis programu na obrazovku od začátku nebo
od určeného radku: LIST 130

LIST# -vypis programu na tiskárnu připojenou na kanál
K(0-7) a jeho registr R(0-X): LIST#404;130

LLIST -vypis programového radku do dialogového radku
k editaci; LLIST -od začátku; LLIST 120 -od radku;

LOAD -nacteni programu uvedeného čísla z magnetofonu;
LOAD 05 - jiné nahravky necte,zobrazí jejich názvy;

LOG -přirozený logaritmus kladného vyrazu;
LOG(1+1.71829)=1

MID# -omezení řetězce od určene pozice na určený
počet znaku. X#='12345':MID#(X#,3,2)='34'

MOVE -presun grafického pera do bodu X,Y určeného
meritkem: SCALE Xmin,Xmax,Ymin,Ymax

NEXT -ukončení programového cyklu,zacínajícího FOR_TO_;

NEW -zrusení programu i jeho promenných v pameti;

NOT -logická negace vyrazu (dvojkový doplněk);
A=5:NOT A=-0:NOT (A-1)=-A

NULL -vlození (0-72) prázdných znaku po každém
radku v pameti programu: NULL(8)

ON ERR-potlaci nejbližší vypis chyby a program
pokračuje uvedeným prikazem: ON ERR GOTO20;

ON_GOSUB_-programový prepínac podle hodnoty promenne;
pro N<=5:ON N GOSUB10,20,30,40,50

ON_GOTO_-programový prepínac skoku podle promenne;
seznam radku skoku musí mít N hodnot;
je-li N mimo rozsah,provede se další radek;

OR -logicky binární součet vyrazu;
7 OR 14=15: 2 OR 7=7

OUT -vystup promenne B(0- ra port I(0-255);
OUT I,B: OUT 4,132:

OUTPUT-vystup dat ze seznamu S na kanál K(0-7)
a registr R(0-X): OUTPUT 404;9,A,T#

PEN -ridi mod vykreslovani grafiky; PEN 1-polojas
 2-blikani, 4-inverse, 8-negovani, 16-mazani
 PEEK -zjistí hodnotu bytu na adrese pameti;
 B=PEEK(A): PEEK('8000)= 'Ç3
 PAUSE -casova prodleva v programu v nasobcich 0.1sec;
 PAUSE 20...2 sec: PAUSE0...25.5 sec
 PLOT -vykresli spojnice do bodu daneho souradnicemi;
 z predchazejici pozice pera, nebo pouze bod (P=1);
 PLOT X,Y,P; PLOT 20,20;50,5;90,10;70,20,1
 POKE -zapise na adresu pameti byte (0-255);
 POKE '2E,80 - zmeni vypis na 80 znaku/radek;
 PRINT -vypis vyrazu ze seznamu na obrazovku;
 PRINT 1;2;3 -vypise cisla vedle sebe s mezerou;
 PRINT 1,2,3 -vypise cisla s tabelaci:1,15,29

 RAD -uhly dalsich goniom. funkci jsou v radianech;
 READ -cteni polozek z radku DATA do seznamu v READ;
 DATA 1,a:READ X,Tu...X=1,Tu='a'
 REM -komentar v programu; povelý za REM jsou neucinne;
 RESTORE-nastavi ukazatel na radek DATA urceny pro READ
 RETURN-povel navratu z podprogramu volaneho GOSUB;
 RIGHTu-omezení retezce na urceny počet znaku zprava;
 Au="abcde":RIGHTu(Au,2)="de"
 ROM -presun programu z pameti typu ROM do pameti RAM
 na adresu '7000 a jeho spusteni: ROM 0
 RND -generator pseudonahodneho cisla z intervalu(0-1);
 RUN -spusti program od zacatku nebo daneho radku: RUN90

 SAVE -zapis programu daneho cisla (<99) na mg.kazetu;
 SAVE 99,NAZEV...nazev muze mit 0-8 znaku;
 SCALE -nastavi meritko souradnic pro graficke prikazy;
 SCALE0,255,0,242 nebo: SCALE -1,1,-10,10
 SGN -hodnota znamenska vyrazu v zavorce;
 SGN(-5)=-1: SGN(0/2)=0: SGN(3*4)=1;
 SIN -hodnota funkce sinus uhlu v radianech/stupnich;
 DEG:SIN(90)=1:RAD:SIN(3.14159/2)=1
 SQR -druha odmocnina z vyrazu; SQR(500+29)=23
 SPC -vlozeni uvedeneho poctu mezer do textu;
 PRINT 1;SPC(5);2;SPC(5);3;...
 STATUS-nacte z kanalu(0-7) hodnotu registru(0-X)
 S=STATUS 4,X...X=(0-3) registry portu 8255;
 STEP -cast zahlaví programoveho cyklu FOR_IO_(STEP_1
 STOP -zastavení provedeneho programu; vypise
 ** Stop at line 95 ** ...Zastavení na radku 95;
 STRu -premeni cislo na retezec znaku;
 X=-1.234:STRu(X)='-1.234'

 TAB -odskok na zadanou pozici v radku textu;
 PRINT TAB(8);a: DISP TAB(10);Tu
 TAN -hodnota tangens uhlu v radianech/stupnich;
 RAD:TAN(3.14159/4)=1:DEG:TAN(45)=1
 THEN -soucast podminkoveho povelu IF_THEN_
 TO -soucast zahlaví programoveho cyklu FOR_IO_










 USR -volani uzivatelskeho podprogramu ve stroj.kodu;
 X=USR('7000)...v promenne je ulozen strdac;

 VAL -premeni retezec cisel na cislo; opak stru;
 X=8:VAL(STRU(Y))=8
 WAIT -zastavení programu az do splnení podmínky ctení
 hodnoty z portu s bity danými maskou M(0-255)
 a neekvivalenci se zadanou hodnotou K(0-255):
 WAIT P,M;K...WAIT 4,15;11
 ? -zkraceny prikaz vypisu zpravy do dialog.radku;
 program pokracuje po stlaceni lib.klavesy;
 ¢ -symbol retezecove promenne (dolar).

Vyznam ridicich a editacnich klaves:

C-D -zobrazeni posledni zpravy pocitace;
 CLR -vymazani celeho dialogoveho radku;
 DEL -vymazani znaku na miste kursoru a posunuti zbyleho textu o znak vlevo;
 END -posun kursoru za posledni znak textu;
 EOL -odeslani textu ke zpracovani pocitacem (provedeni prikazu nebo zarazeni radku);
 F0-F11-programove klice;po predchozim stlaceni WRK se do nich uchova text dialogoveho radku;
 INS -na miste kursoru vytvori mezeru pro vlozeni dalsiho znaku;
 RCL -vyvolani posledniho textu odeslaneho EOL;
 SIOP -prerusi provadeni programu po ukonceni vseh povelu programoveho radku; pokracovat lze povelom CONT (a EOL);
 WRK -umoznuje zapis do programovych klicu F0-F11;
 ← → -posun kursoru o jeden znak vlevo/vpravo;
 ↶ ↷ -navrati kursor na zacatek dialogoveho radku;
 ← → -posun dialogoveho radku o 1 znak vuci kursoru;
 ↑ -premykac (shift) umoznuje psani hornich znaku klaves a malych pismen; po dobu stlaceni pozastavuje spusteny program;
 ↑+C-D -prechod do rezimu terminalu;
 ↑+CLR -vymaze text od kursoru vpravo;
 ↑+DEL -odstartuje program od adresy 0000; byl-li tam jiz interpret BASIC-G, tak jej spusti;
 ↑+END -vypne/zapne zvukovou signalizaci (pipani);
 ↑+F0-F11-programove klice 'horni' pro uchovani textu;
 ↑+PRINT-vypnuti/zapnuti opisu dialogoveho radku do pracovni casti stinitka;
 ↑+RCL -prepnuti do rezimu MONITOR (navrat .+DEL);
 ↑+RSI -reset systemu; nouzova funkce pro pripad havarie programu; spusti program od adr.0000;
 ↑+WRK -vyvola znak nasledujici v tabulce ASCII za znakem, na kterem stoji kursor;
 ↑+↵ -vymaze obrazovku;
 ↑+↵ -prepne zobrazeni na inversni (cerne na bilem);
 ↑+↵ -prepne dialogovy radek na polojas (blikani).

Rozkresleni znaku na hornim radku:

Adr:	Bity	Obsah adresy:
HEX	012345	HEX DEK
'C000		= '1F = 31
'C040		= '01 = 1
'C080		= '01 = 1
'C0C0		= '07 = 7
'C100		= '01 = 1
'C140		= '01 = 1
'C180		= '1F = 31
'C1C0		= '00 = 0
'C200		= '00 = 0

Chybova hlášení BASIC-G:

Err.Allog -chybné nebo dvakrát dimenzované pole;
Can't cont -nesprávně použit příkaz CONT;
Data exha -chybějící řádek DATA pro příkaz READ;
Div by zero -pokus o dělení nulou;
Field lost -překročen počet vstupních dat v INPUT;
File bound -překročen maximální počet záznamů;
File error -chyba při nahrávání (přenosu) dat;
File small -data z mgp neodpovídají dimenzovanému poli;
Fnc.param -nesprávný parametr funkce;
Input err. -chyba při vstupu dat (znak místo číslice);
No for stm -chybně zadána program.smyčka FOR_TO_NEXT;
No str.spc -preplněná paměť pro řetězcové proměnné;
Numb.nonex -neexistující program.řádek v GOTO;GOSUB;
Only in pg -příkaz lze použít jen v programovém modu;
Overflow -překročen rozsah výpočtu (výsledek >10³⁸);
Pg too big -není již volná paměť pro program;
Return err -použit RETURN bez volání GOSUB;
Stop at ln -příkaz STOP na řádku ln zastavil program;
Str algrt. -výraz v řetězci je složitý; rozdělit;
Str.long -delka řetězce přesahuje 255 znaků;
Subscr.rng -index proměnné mimo rozsah určený DIM;
Syntax err -chybně zadán příkaz (překlep);
Type conv -chybný převod čísla na řetězec a opačně.

Příkazy pro práci v režimu monitor:

DUMP -vypis obsahu paměti na obrazovku; DUMP 8427
8427 4F 73 20 72 65 61 64 79 0s ready
(lze pozastavit pomocí SHIFT a zastavit STOP);
JOB -přenos programu z adresy AROM v modulu ROM do RAM
na adresu APOC v delce LONG: JOB AROMLONGAPOC
JOB 000025000000...zavádí interpret BASIC-G z ROM;
JUMP -odstartování programu na zadane adrese;
Např.: JUMP 0000...odstartuje interpret BASIC-G;
MEM -vypis 16 bytů z paměti od zadane adresy do dialog.
řádku v šestnáctkovém tvaru: MEM C030
monitor změní výpis na SUB C030 pro modifikaci dat,
MGSV -zápis programu z paměti na magnetofon pod uvedeným
číslem(<99), od adresy po adresu, s názvem (0-8 znaků)
MGSV 15,0000-2500,BASIC G
MGEND kontrola nahraneho programu na magnetofonu; je-li
správný kontrolní součet bytů záznamu, vypíše: OK
MGLD -nactení programu uvedeneho čísla z magnetofonu do
paměti počítací: MGLD 15 (jina čísla nenahrává);
SUB -SUBSTITUTE-změna obsahu paměti na uvedene adrese;
↑+RST-přechod do režimu BASIC-G přes RESTART;
↑+DEL-odstartuje program v paměti od adresy 0000; např.
pokud byl již v paměti BASIC-G, spustí jej.

Hlášení monitoru:

Error in adres -chyba v adrese (4-místná; šestnáctková);
Error in data -chyba v datech SUB (šestnáctková);
Executive -oznamuje spuštění programu ve stroj.kódu;
File error -čtený soubor je chybný nebo nebyl
nahráván v režimu MONITOR;
Memory overflow-preplněná paměť kliců F0-F11 ('7000-'7F00);
No command -použit příkaz neznámý monitoru;
No key -stisknutý klíč nemá zadan obsah;
Mg stop ! -ukončeno nahrávání programu.

Znaky v kodu ASCII (ISO-7):

00 NUL	20	40 @	60 `
01 SOH	21 !	41 A	61 a
02 STX	22 "	42 B	62 b
03 ETX	23 #	43 C	63 c
04 EOT	24 \$	44 D	64 d
05 ENQ	25 %	45 E	65 e
06 ACK	26 &	46 F	66 f
07 BEL	27 '	47 G	67 g
08 BS	28 (48 H	68 h
09 HT	29)	49 I	69 i
0A LF	2A *	4A J	6A j
0B VT	2B +	4B K	6B k
0C FF	2C ,	4C L	6C l
0D CR	2D -	4D M	6D m
0E SO	2E .	4E N	6E n
0F SI	2F /	4F O	6F o
10 DLE	30 0	50 P	70 p
11 DC1(X-ON)	31 1	51 Q	71 q
12 DC2	32 2	52 R	72 r
13 DC3	33 3	53 S	73 s
14 DC4	34 4	54 T	74 t
15 NAK	35 5	55 U	75 u
16 SYN	36 6	56 V	76 v
17 ETB	37 7	57 W	77 w
18 CAN	38 8	58 X	78 x
19 EM	39 9	59 Y	79 y
1A SUB	3A :	5A Z	7A z
1B ESC	3B ;	5B [7B {
1C FS	3C <	5C \	7C
1D GS	3D =	5D]	7D }
1E RS	3E >	5E ^	7E ~
1F US	3F ?	5F _	7F x

Vyznam nekterych ridicich znaku:

NUL -prazdny znak;
 SOH (Start Of Header)-zahlati dat;
 STX (Start of Text)-zacatek textu;
 ETX (End of Text)-konec textu;
 ENQ (ENQuiry)-pozadvek spojeni;
 ACK (ACKnowledge)-potvrzeni zpravy;
 BEL -zvukovy signal;
 LF (Line Feed)-novy radek;
 CR (Carriage Return)-navrat voziku;
 NAK (No ACKnowledge)-nepotvrzeni;

Prevodni tabulka cisel desitkove/dvojkove/sestnactkove:

0 = 00000000 = 00	0 = 00000000 = 00
1 = 00000001 = 01	16 = 00010000 = 10
2 = 00000010 = 02	32 = 00100000 = 20
3 = 00000011 = 03	48 = 00110000 = 30
4 = 00000100 = 04	64 = 01000000 = 40
5 = 00000101 = 05	80 = 01010000 = 50
6 = 00000110 = 06	96 = 01100000 = 60
7 = 00000111 = 07	112 = 01110000 = 70
8 = 00001000 = 08	128 = 10000000 = 80
9 = 00001001 = 09	144 = 10010000 = 90
10 = 00001010 = 0A	160 = 10100000 = A0
11 = 00001011 = 0B	176 = 10110000 = B0
12 = 00001100 = 0C	192 = 11000000 = C0
13 = 00001101 = 0D	208 = 11010000 = D0
14 = 00001110 = 0E	224 = 11100000 = E0
15 = 00001111 = 0F	240 = 11110000 = F0