

Závodná pobočka ČSVTS pri STS k. p. Trstená

# Návod na obsluhu

modulu ROM PASCAL pre počítače PMD 85

TRSTENÁ 1989



## Návod na použitie Pascalu PMD V1.02

Pascal PMD je prekladač podmnožiny programovacieho jazyka Pascal. Je vhodný na výuku programovania na stredných školách a v záujmových útvaroch.

Prekladač Pascal PMD V1.02 je určený pre PMD-85-2, PMD-85-2A a v režime kompatibility s PMD-85-2 ho možno použiť aj na PMD-85-3. (Pre PMD-85-1 je určená verzia Pascal PMD V1.01).

### 1 Spustenie prekladača

Pascal PMD sa spúšťa:

— na PMD-85-2 a PMD-85-2A automaticky po zapnutí počítača so založeným modulom, alebo hocikedy stlačením  $\wedge$  RST.

— na PMD-85-3 prechodom do režimu kompatibility s PMD-85-2, teda príkazom JUMP FFF0.

Prekladač sa prihlási verziou 1.02 pre PMD-85-2 a 1.02A pre PMD-85-2A a PMD-85-3. Písmeno A za verziou oznamuje, že je k dispozícii pamäť na adresách 9000 — BFFF. Pri nábehu prekladača sa najprv zisťuje, či bol obsah ROM modulu správne prečítaný (pomocou kontrolného súčtu). Ak nebol, prekladač nenabehne — vypíše sa správa „Os ready“, akoby modul nebol založený.

Ďalšia činnosť prekladača je zistenie, či je v oblasti pamäte (rozdelenie pamäte je na konci tohoto návodu), kde sa bežne ukladá zdrojový tvar programu „dobrý“ obsah. Ak áno, tak sa tento nezruší. Teda ak stlačíme počas práce s Pascalom RST (či už náhodou alebo naschvál), program, s ktorým sme pracovali sa nestratí. (V niektorých výnimočných situáciách sa pri znovunaštartovaní Pascalu program môže stratiť — to však znamená, že už bol predtým narušený). Stratí sa len jeho meno — to musíme napísať znovu pomocou príkazu preMenuj.

### 2 Hlavné menu

Po spustení prekladača sa objaví hlavné menu. Obsahuje všetky základné voľby, ktorými ovládame prácu systému. Voľbu z menu zvolíme stlačením písmena, ktoré je v jej názve veľké.

Edituj	Vyvolanie editora.
Preloz	Preklad programu.
Rob	Spustenie programu (ak bol od posledného prekladu vyvolaný editor, tak sa najprv program preloží).
Koniec	Ukončenie práce, návrat do monitora PMD. Ak nespravíme nič, čím by sme zmazali prekladač z pamäti, môžeme sa z monitora vrátiť do Pascalu stlačením $\wedge$ a DEL alebo príkazom JUMP 0000 (pre PMD-85-1 platí len druhá možnosť).
Novy	Zmazanie programu, vypýta si nové meno.
Zapis	Uloženie programu na magnetofón pod menom, ktoré je uvedené v menu (teda ak program nemá meno, nezabudnite ho pomenovať príkazom preMenuj, inak sa vám nahrá pod prázdny menom). Program sa nahráva po blokoch.
Citaj	Prečítanie programu z magnetofónu a pripojí ho za program v pamäti. Teda ak nechceme, aby bol program pripojený za existujúci, použijeme pred čítaním voľbu Novy, ktorá zruší program v pamäti. Program sa číta po blokoch. V spodnom riadku je názov práve čítaného bloku. Ak počítač niektorý blok prečíta zle (prejaví sa to tým, že pri zobrazovaní názvov ďalších blokov pípa a vypisuje ich mená do bežnej obrazovky), stačí vrátiť pásku na zlý blok a pustiť znova magnetofón (prítom nezastaviť čítanie). Takto je niekedy možné prečítať aj dosť zlú nahrávku. Ak vznikne pri čítaní chyba, počítač vypíše „Zle!“. Čítanie možno kedykoľvek zastaviť klávesom STOP (aj vtedy sa vždy vypíše „Zle!“).
Tlac	Tlač textu programu na tlačiareň.
preMenuj	Premenovanie programu v pamäti.
Indexy	Zapnutie a vypnutie kontroly indexov. Ak je za ním +, bude sa pri každom indexovaní poľa kontrolovať, či je index v rozsahu podľa deklarácie poľa. Ak je —, kontroly sa nevykonávajú — program pobeží rýchlejšie, ale v prípade nesprávneho indexovania môže narušiť systém.



### 3 Editor

Do editora sa vchádza z hlavného menu stlačením písmena E. Editor sa ukončí súčasným stlačením STOP a EOL.

Editor je obrazovkový: ak stlačíte kláves s písmenom (alebo iným tlačiteľným znakom), napíše sa do riadku na pozíciu kurzora. Dĺžka riadkov môže byť maximálne 80 znakov, ale odporúča sa používať riadky do 48 znakov. Ak píšeme za 48. pozíciu v riadku, riadok sa začne „rolovať“.

Každý riadok, z ktorého odídeme kurzorom, sa spracuje a vypíše sa späť na obrazovku v upravenom tvare: všetky kľúčové slová sú zobrazené malými písmenami a všetky identifikátory veľkými písmenami (PMD-85-1 nemá malé písmená — všetko sa vypisuje veľkými písmenami). Medzery sú v riadku rozmiestnené tak, aby správne oddeľovali jednotlivé prvky jazyka. Ak vznikne v riadku chyba (napríklad chýba apostrof, či zlý zápis reálnej konštanty), nie je možné z riadku odísť.

#### 3.1 Funkcie klávesov v editore

znakom  $\wedge$  označujeme kláves preraďovač (shift). Klávesy so „šípkami s čiarkou“ nazývame šípka hore a šípka dole. Kláves so šikmou šípkou nazývame HOME.

šípka doľava	posun o znak doľava
šípka doprava	posun o znak doprava
šípka hore	posun o znak hore (na predošlý riadok)
šípka dolu	posun o znak dole (na ďalší riadok)
EOL	prechod na začiatok nasledujúceho riadku
HOME	
(šikmá šípka)	posunie kurzor na začiatok riadku
END	posunie kurzor na prvú medzeru za riadkom
CLR	zmazanie obsahu riadku (celý riadok sa zapíše medzerami)
$\wedge$ CLR	zmaže znaky na riadku vpravo od kurzora
INS	vloženie medzery na pozíciu kurzora (zvyšok riadku sa posunie doprava)
DEL	zmazanie znaku na mieste kurzora (zvyšok riadku sa posunie doľava)
RCL	vyvolanie riadku v takom stave, aký bol, keď sme doňho prišli šípkou (teda „vrátenie“ riadku v takom stave, aký bol pred jeho menením)
$\wedge$ HOME	nastaví kurzor na prvý riadok programu
$\wedge$ END	nastaví kurzor na posledný riadok programu
$\wedge$ šípka dole	vyroluje program o 20 riadkov dole, zobrazí teda „ďalšiu stránku“ programu
$\wedge$ šípka hore	vyroluje program o 20 riadkov hore, zobrazí teda „predošlú stránku“ programu
$\wedge$ INS	pred riadok s kurzorom vsunie prázdny riadok
$\wedge$ DEL	zruší riadok s kurzorom
$\wedge$ RCL	pred riadkom s kurzorom vloží posledne zrušený riadok pomocou $\wedge$ DEL
C—D	privolá do najspodnejšieho riadku poslednú správu (väčšinou správu o chybe)
WRK	definovanie kľúča — keď po ňom stlačíte K0 až K11 alebo $\wedge$ K0 až $\wedge$ K11, zapamätá sa do kľúča časť riadku vľavo od kurzora
$\wedge$ WRK	znak, na ktorom je kurzor, nahradí znakom s kódom o jedna vyšším

#### 4 Chybovník

Chyby pri preklade aj pri behu sa vyhlasujú tak, že sa na miesto chyby nastaví kurzor v editore. Chyby pri editovaní sa hlásia keď chceme riadok opustiť kurzorom, kurzor zostane na svojom mieste (neukazuje presné miesto chyby).

Syntaktické chyby (hlásia sa pri preklade)

- 1: Na začiatku programu musí byť slovo PROGRAM
- 2: Očakával sa identifikátor
- 3: Zlý začiatok deklarácie
- 4: Chýba rovnítko
- 5: Chýba bodkočiarka
- 6: Chýba dvojbodka
- 7: Chýba pravá zátvorka
- 8: Zlý začiatok popisu typu
- 9: Chýba ľavá hranatá zátvorka



- 10: Chýbajú dve bodky
- 11: Chýba pravá hranatá zátvorka
- 12: Chýba OF
- 13: Chýba znak priradenia — nepochopený príkaz
- 14: Chýba END
- 15: Chýba DO
- 16: Chýba UNTIL
- 17: Chýba TO alebo DOWNT
- 18: Chýba pravá zátvorka
- 19: Chýba bodka za posledným END alebo je END navyše
- 20: Zlý výraz
- 21: Zlý začiatok príkazu
- 22: Zlá konštanta
- 23: Chýba THEN
- 24: Očakávalo sa celé číslo

Sémantické chyby (hlásia sa pri preklade)

- 31: Identifikátor nie je identifikátorom konštanty
- 32: Unárne plus alebo mínus pre typ rôzny od INTEGER alebo REAL
- 33: Zlý typ indexov v deklarácii poľa
- 34: Pole nemôže mať viac než 65536 bajtov
- 35: Identifikátor nie je identifikátorom typu
- 36: Dva formáty sú povolené len pre výrazy typu REAL
- 37: Nevhodný typ hranice v deklarácii intervalu
- 38: Nekompatibilné typy hraníc v deklarácii intervalu
- 39: V intervale nemôže byť dolná hranica väčšia než horná
- 40: Nedeklarovaný identifikátor
- 41: Identifikátor bol už v tomto bloku použitý, nemôže byť predefinovaný
- 42: Identifikátor bol už v tomto bloku deklarovaný
- 43: Výraz v CASE musí byť skalárneho typu
- 44: Nekompatibilný typ v CASE
- 45:
- 46: ABSOLUTE je použité inde než v hlavnom programe
- 47: Viacnásobný FORWARD
- 48: V tele FORWARD procedúry alebo funkcie už nie je možné opakovať parametre a typ
- 49: Chýbajúce deklarácie procedúr alebo funkcií, ktoré boli zadeklarované ako FORWARD (hlási sa na BEGIN)
- 50: Nekompatibilné alebo nesprávne typy v celočíselnej alebo reálnej binárnej operácii
- 51: Nekompatibilné typy v logickej binárnej operácii
- 52: Hranica nie je kompatibilná s premennou cyklu FOR
- 53: Zlý typ v NOT
- 54: Zlý identifikátor na ľavej strane priradenia
- 55: Indexovať je možné len polia
- 56: Nekompatibilný typ indexu
- 57: Takéto typy možno porovnať len na = alebo <
- 58: Priradenie hodnoty funkcie mimo jej tela
- 59: Identifikátor nie je identifikátorom premennej alebo funkcie
- 60: Nekompatibilné typy pri priradení
- 61:
- 62: Očakával sa výraz typu BOOLEAN
- 63: Objekty tohoto typu nie je možné použiť v READ a WRITE
- 64: Formát v príkaze WRITE musí byť typu INTEGER
- 65: Nekompatibilné typy v relačnom vzťahu
- 66: Identifikátor nie je procedúra
- 67: Príveľa parametrov
- 68: Nekompatibilný typ parametra
- 69: Za var parameter nemožno poslať výraz
- 70: Málo parametrov
- 71: Identifikátor nie je funkcia
- 72: Identifikátor nie je identifikátorom premennej
- 73: Zlý identifikátor vo FOR (musí to byť lokálna premenná)
- 74: EXIT nie je v žiadnom cykle
- 75: Interná chyba — pokus o zlé adresovanie

Chyby pri editovaní, pri behu a prekročení obmedzení pri preklade

- 77: Koniec súboru (EOF)



- 78: Parameter SQRT je záporný
- 79: Parameter LN je nekladný
- 80: Parameter EXP je väčší než 88.0
- 81: Neznáme zariadenie v SETOUTPUT
- 82: Zlé meno zariadenia v SETOUTPUT
- 83: Prívelká dĺžka v BLOT (obrázok by bol mimo obrazovky)
- 84: Prívelká šírka v BLOT (obrázok by bol mimo obrazovky)
- 85: Zlé súradnice v BLOT (mimo interval 0..47 reap. 0..255)
- 86: Pri výstupe na GR: bol presiahnutý kraj obrazovky
- 87: Pretečenie pri reálnej operácii, pri TRUNC alebo ROUND
- 88: Index mimo interval
- 89: Delenie nulou
- 90: Dlhý riadok (pri editovaní). Touto chybou editor označí situáciu, keď je po úprave riadok dlhší než 80 znakov, hoci pôvodne bol napísaný na menej znakov (môže sa to stať napríklad ak je v riadku viac reálnych konštánt v semilogaritmickom tvare). Z textu už existuje len 79 znakov — zvyšok je stratený. Na koniec riadku umiestni editor znak '!' (výkričník). Ten spôsobí, že nie je možné z riadku odísť bez jeho úpravy. (Pozn.: ak píšeme riadky do 48 znakov, táto chyba sa prakticky nevyskytne).
- 91: Chýba apotrof (pri editovaní)
- 92: Zlý zápis celého alebo reálneho čísla (pri editovaní aj pri behu). Túto chybu vyhlási aj vtedy, ak zadávame prívelké alebo prímalé číslo (okrem celých čísiel pri editovaní — tie sa nekontrolujú), alebo keď má mantisa reálneho čísla viac než 38 cifier (nie je účelné písať viac než 7 cifier mantisy — číslo nebude presnejšie, ba práve naopak, môžu sa prejaviť zaokrúhľovacie chyby).
- 93: Málo pamäti (pri editovaní, preklade aj pri behu)
- 94: Príliš zložitý výraz (pri preklade)
- 95: Viac než 4 úrovně vnorenia procedúr a funkcií (pri preklade)
- 96: Viac než 255 identifikátorov (pri editovaní)
- 97: Pretečenie tabuliek kompilátora (pri preklade)
- 98: Bol dosiahnutý koniec textu pred logickým koncom programu (varovanie pri preklade)
- 99: Pretečenie zásobníka analyzátoru — prívelá vnorenia procedúr, príkazov a výrazov (pri preklade)
- 00: V riadku je znak, ktorý nemá v Pascale význam (pri editovaní)

## 5 Jazyk Pascal PMD

Pascal PMD je podmnožinou štandardného pascalu (pri jej výbere sme vychádzali z podmnožiny Pascal-S, ktorá je použitá v učebnici N. Wirtha Systematické programovanie, slovenský preklad vyšiel v Alfe v r. 1981) a je rozšírený o niektoré štandardné procedúry a o prostriedky pre programovanie na nižšej úrovni. V tomto odseku popíšeme veľmi stručne zmeny v jazyku oproti štandardnému pascalu (pozri napr. knihu L. Molnár: Programovanie v jazyku Pascal, Alfa a SNTL 1987) a podrobnejšie popíšeme všetky funkcie a procedúry, ktoré sú v Pascale štandardné, ako aj implementáciu vstupných a výstupných príkazov a procedúr. Predpokladáme, že čitateľ je oboznámený so štandardným pascalom.

### 5.1 Obmedzenia oproti štandardnému Pascalu

Zo štandardného pascalu nie sú implementované tieto vlastnosti:

Deklarácia LABEL

Typy údajov súbor, záznam, množina a smerník

Zhustené typy údajov

Parametre typu procedúra, funkcia a konformné pole

Príkazy WITH a GOTO

Štandardné procedúry a funkcie, ktoré narábajú s objektmi takých typov, ktoré nie sú v Pascale PMD implementované (napríklad PACK, NEW, RESET, REWRITE, GET, PUT)

Slová READ, READLN, WRITE a WRITELN považuje prekladač za rezervované na rozdiel od štandardného Pascalu, kde sú to identifikátory štandardných funkcií (v Pascale PMD potom hovoríme o príkaze READ miesto-procedúry READ).

### 5.2 Rozšírenia jazyka

Väčšina rozšírení je motivovaná lepším prístupom k programovaniu na „nižšej úrov-



ni“, teda na narábanie s pamäťou, s jednotlivými bitmi premenných a podobne. Začiatok-  
níci môžu nasledujúce odseky vynechať a vrátiť sa k nim neskôr.

#### Zápis číselných konštánt v šestnástkovej sústave

Číselné konštanty je možné zapisovať aj v šestnástkovej sústave, ak sa pred číslom  
uvedie znak \$ (dolár). Napríklad zápis \$FF je ekvivalentný zápisu 255.

#### Zápis reťazcových konštánt

Okrem bežného zápisu reťazcových konštánt (napr. 'ABCD') je možné použiť aj zápis  
pomocou kódov jednotlivých znakov, kde pred každým z nich je uvedený znak #  
Napríklad zápis #11 #12 #13 predstavuje konstantu typu ARRAY[1..3] OF  
CHAR, ktorej prvý prvok má hodnotu CHR(11), druhý CHR(12) a tretí CHR(13). Oba  
spôsoby zápisu je možné miešať napríklad takto 'AHOJ' #13'JANO'. V zápise nesmú  
byť medzery mimo apostrofov. V kódach je možné použiť aj šestnástkovú sústavu:  
'AHOJ' # \$OD'JANO'. Tento zápis konštánt využijeme ako parameter procedúry  
BPL0T.

#### Určenie adresy premennej

Premenným hlavného programu možno predpísať adresu ich umiestnenia pomocou  
slova ABSOLUTE, za ktorým nasleduje celočíselná konštanta (s výhodou využijeme  
šestnástkovú sústavu). ABSOLUTE zapisujeme za typ premennej. Takto je možné pri-  
stupovať k premenným monitora, alebo zapisovať údaje do špeciálnych oblastí pamä-  
te (teda to, čo sa v basicu dosahuje pomocou PEEK a POKE).

Príklady:

```
VAR FARBA:CHAR ABSOLUTE $C03A;
```

```
OBR:ARRAY[0..63,0..255] OF CHAR ABSOLUTE $C000;
```

#### Operátor XOR

Týmto operátorom je označená operácia „exclusive or“, v matematike nazývaná non-  
ekvivalencia. Operátor možno aplikovať na operandy typu BOOLEAN alebo INTEGER.

#### Logické operácie s číslami

Logické operátory AND, OR, XOR a NOT možno použiť aj s operandmi typu INTEGER.  
V tomto prípade majú význam operácií s bitmi čísiel a výsledok je typu INTEGER.

#### Príkazy BREAK, HALT a EXIT

V pascali PMD nie je implementovaný príkaz skoku. V najnutnejších prípadoch ho  
možno nahradiť „miernejšími“ (ale hlavne jasnejšími) príkazmi, ktoré ukončia vyko-  
návanie presne určenej časti programu. Príkaz BREAK ukončí vykonávanie najvnútor-  
nejšieho cyklu, v ktorom sa nachádza a pokračuje na prvom príkaze za týmto cyk-  
lom. Príkaz EXIT ukončí vykonávanie procedúry alebo funkcie, v ktorej sa nachádza  
(akási obdoba basicovského RETURN). Príkaz HALT ukončí vykonávanie celého prog-  
ramu.

#### Časť OTHERWISE v príkaze CASE

Pre pohodlnejšie programovanie príkazov CASE je možné použiť poslednú možnosť  
OTHERWISE. Príkaz za OTHERWISE sa vykoná vtedy, ak výraz za CASE nenadobúda  
ani jednu hodnotu z predtým vymenovaných prípadov:

```
CASE I OF  
  1:J:=A+B;  
  2:J:=A*B;  
  OTHERWISE J:=0  
END;
```

### 5.3 Zmeny, ktoré vyplývajú z koncepcie editora

Editor analyzuje každý jeden riadok osobitne a prevádza ho do vnútorného tvaru. Z  
toho vyplýva niekoľko obmedzení na zápis programu. Na väčšinu z nich narazíme len vo  
veľmi špeciálnych prípadoch.

- Reťazcové konštanty a komentáre nemôžu prechádzať z jedného riadku na druhý.
- Celočíselné konštanty rozoznáva editor bez znamienka, teda v rozsahu od 0 do 65535, ak  
napíšeme väčšie celé číslo, editor ho „usekne do tohoto intervalu. (Používať čísla nad  
32767 je vhodné len v špeciálnych prípadoch pri programovaní na nižšej úrovni.)
- Reálnu konstantu v semilogaritmickom tvare editor vždy zobrazuje s mantisou na 6  
miest s desatinou bodkou za prvou cifrou (teda ak napíšeme 51.2E3, editor zobrazí  
5.12000E+04).
- Reťazcové a znakové konštanty, ktoré boli zapísané pomocou znaku # zobrazi editor



podľa toho, či ide o tlačiteľné znaky alebo nie (teda ak napíšeme `## 65 ## 1 ## 66`, editor zobrazí 'A' `## 1'B'`).

- Ak bola v zápise refazcovej konštanty použitá šesťnástková sústava čo len v jednom znaku, editor zobrazí všetky netlačiteľné znaky refazca v šesťnástkovej sústave (teda ak napíšeme `## 128 ## 65 ## $FF`, editor zobrazí `## $80'A' ## $FF`).

#### 5.4 Implementácia štandardných typov

Typ INTEGER je implementovaný v 16 bitoch. Má rozsah od `—32768` po `32767`. Štandardná konštanta MAXINT má hodnotu `32767`. Operácie s typom INTEGER nehlásia chybu pri pretečení.

Typ REAL je implementovaný v 32 bitoch s 24-bitovou mantisou a 8-bitovým exponentom. Rozsah absolútnych hodnôt čísiel je približne od `1E-38` po `1E37`. Vnútoraná presnosť je približne sedem platných cifier, pri výstupe sa však zobrazuje najviac šesť platných cifier. Operácie a funkcie s typom REAL hlásia chybu pri pretečení. Pri podtečení je výsledok nula.

Typ CHAR je implementovaný v 1 bajte kód je podľa tabuľky ASCII (ISO-7, resp. KOI-7NO). Rozsah kódov znakov je od 0 do 255.

Typ BOOLEAN je implementovaný v 1 bajte, ktorý nadobúda hodnoty 0 a 1 (čo zodpovedá konštantám FALSE a TRUE).

Vymenované typy sú implementované v 1 bajte.

#### 5.5 Obmedzenia veľkosti

Maximálne vnorenie procedúr a funkcií: 4

Maximálny počet identifikátorov v programe: 255

#### 5.6 Vstup a výstup

Pretože Pascal PMD nepočíta s pripojením disketovej jednotky a práca s kazetovou páskou nezodpovedá predstave súborov v štandardnom Pascale, nie je implementovaný typ FILE.

##### 5.6.1 Vstup

Príkazom READ je možné čítať z klávesnice tak, že sa vstupné písmená zobrazujú na obrazovke tam, kde je nastavený kurzor. Počas zápisu vstupu je zobrazovaný kurzor a je možné používať klávesy šípka vľavo alebo DEL na mazanie zlých znakov. Pri zápise nie je možné prekročiť pravý okraj obrazovky (teda nie je možné jedným príkazom READ prečítať viac než 47 znakov). Vstup je treba ukončiť stlačením EOL.

READLN a EOLN funguje analogicky ako v štandardnom Pascale: ak použijeme READ, hoci platí EOLN, prečítame znak medzera, pri ďalšom použití READ sa bude žiadať nový riadok z klávesnice. Situáciu EOF je možné dosiahnuť stlačením klávesu so šípkou vľavo s čiarkou (šípka hore) — zobrazí sa znak malý štvorček (nedefinovaný znak) a potom EOL.

Príkazom READ môžeme čítať premenné celočíselné, reálne, znakové a refazcové (typu `ARRAY[1..N] OF CHAR`, kde N je ľubovoľné).

Na čítanie bez zobrazovania znakov slúži funkcia READKEY typu CHAR. Čaká na stlačenie klávesu. Logická funkcia KEYPRESSED informuje o tom, či je práve nejaký kláves stlačený. Ak je, tak okrem oznámenia výsledku TRUE uloží stlačený kláves do vyrovnávacej pamäti, odkiaľ si ju môžeme vyzdvihnúť volaním READKEY — v tomto prípade už READKEY nebude čakať na stlačenie klávesu.

##### 5.6.2 Výstup

Na výstup slúžia príkazy WRITE a WRITELN. Možno nimi vypisovať celé a reálne čísla, znaky, refazce a logické hodnoty (objekty typu boolean).

Bežne je nastavený výpis na obrazovku — znaky sa vypisujú od pozície kurzora, príkaz WRITELN spôsobí prechod na nový riadok. Pozíciu kurzora je možné zmeniť volaním procedúry GOTOXY.

Výstup je možné presmerovať volaním procedúry SETOUTPUT na tlačiareň alebo do grafickej obrazovky.



### 5.6.2.1 Formátovanie výstupu

Za výrazom v príkaze WRITE alebo WRITELN môže nasledovať dvojbodka a za ňou celočíselný výraz. Tento výraz nazývame formát. Napríklad v príkaze WRITELN(X:8) je uvedený formát 8. Pri výpise reálnych čísel môžeme použiť aj dva formáty, napríklad WRITE(Y:15:6).

Prvý formát udáva počet znakov, ktoré má vo výpise zabrať príslušná hodnota. Druhý formát u reálnych čísel znamená počet miest za desatinnou bodkou. Ak je 0, tak sa číslo vypisuje v semilogaritmickom tvare. V každom prípade sa z reálneho čísla vypíše najviac 6 platných číslic, ostatné sú nuly. Ak je druhý formát väčší než 38, priradí sa mu hodnota 38.

Ak je vypisovaná hodnota kratšia než prvý formát, doplní sa medzerami zľava. Ak sa číslo (celé alebo reálne) nedá vypísať na daný počet znakov, hodnota formátu sa ignoruje a číslo sa vypíše na najmenší možný počet znakov. Ak je refazec kratší než prvý formát, usekne sa.

Ak v príkaze WRITE alebo WRITELN nie je uvedený formát, doplní sa štandardná hodnota podľa typu výrazu:

Typ výrazu	Št. hodnota 1. formátu	Št. hodnota 2. formátu
znak	1	
refazec	dĺžka refazca	
logická hodnota	6	
celé číslo	6	
reálne číslo	12	0

### 5.7 Zoznam štandardných funkcií a procedúr

Funkcie a procedúry, ktoré nie sú v štandardnom Pascale, sú označené hviezdíčkou.

V zozname používame pri popise typov parametrov a výsledkov tieto zvláštne typy:

SKALAR ľubovoľný skalárny typ (INTEGER, CHAR, BOOLEAN alebo vymenovaný typ)

RETAZEC ľubovoľný typ ARRAY [1..N] OF CHAR

FUNCTION CHR(K:INTEGER):CHAR;

prevod kód → znak

FUNCTION ORD (S:SKALAR):INTEGER;

prevod skalár → poradové číslo, špeciálne pre znak je to kód znaku

FUNCTION SUCC(S:SKALAR):SKALAR;

nasledovník

FUNCTION PRED(S:SKALAR):SKALAR;

predchodca

FUNCTION RAND(N:INTEGER):INTEGER;

náhodné celé číslo od 0 po N-1

FUNCTION ABS(I:INTEGER):INTEGER;

absolútna hodnota

FUNCTION SQR(I:INTEGER):INTEGER;

druhá mocnina

FUNCTION ODD(I:INTEGER):BOOLEAN;

výsledok je TRUE, ak je I nepárne

FUNCTION EOLN:BOOLEAN;

výsledok je TRUE, ak je na vstupe detekovaný koniec riadku

FUNCTION EOF:BOOLEAN;

výsledok je TRUE, ak je na vstupe znak „koniec súboru“. Znak koniec súboru napíšeme stlačením klávesu so šípkou vľavo s čiarkou (šípka hore)

FUNCTION KEYPRESSED:BOOLEAN;

Výsledok je TRUE, ak je v čase volania stlačený nejaký kláves na klávesnici. Výsledok je TRUE aj vtedy, ak bol stlačený kláves počas niektorého predchádzajúceho volania funkcie KEYPRESSED, ale medzitým nebola volaná funkcia READKEY

FUNCTION READKEY:CHAR;

Výsledkom je kláves, ktorý je stlačený počas volania funkcie READKEY, alebo počas niektorého predošlého vyvolania funkcie KEYPRESSED. Ak nie je žiadny kláves stlačený, funkcia čaká na jeho stlačenie. Stlačený kláves sa nezobrazí na obrazovke.

FUNCTION ADR(VAR X:LUBOVOLNY TYP):INTEGER;

Výsledkom funkcie je adresa parametra

FUNCTION INP(PORT:INTEGER):INTEGER;

Výsledkom funkcie je bajt prečítaný zo zadaného portu:



FUNCTION TRUNC(X:REAL):INTEGER;  
 Prevod čísla z REAL do INTEGER s useknutím desatinnej časti. Teda TRUNC(−7.8) = −7, TRUNC(5.4) = 5.

FUNCTION ROUND(X:REAL):INTEGER;  
 Prevod čísla z REAL do INTEGER so zaokrúhlením. Pre nezáporné čísla platí ROUND(X) = TRUNC(X + 0.5), pre záporné čísla platí ROUND(X) = TRUNC(X − 0.5).

FUNCTION RAND:REAL;  
 Pri volaní funkcie RAND bez parametra je výsledkom náhodné číslo z intervalu (0, 1).

FUNCTION ABS(X:REAL):REAL;  
 Absolútna hodnota.

FUNCTION SQR(X:REAL):REAL;  
 Druhá mocnina čísla.

FUNCTION SQRT(X:REAL):REAL;  
 Druhá odmocnina čísla.

FUNCTION SIN(X:REAL):REAL;  
 Sínus uhla v radiánoch.

FUNCTION COS(X:REAL):REAL;  
 Kosínus uhla v radiánoch.

FUNCTION ARCTAN(X:REAL):REAL;  
 Arkustangens — výsledok je v radiánoch.

FUNCTION EXP(X:REAL):REAL;  
 Funkcia „e na X“.

FUNCTION LN(X:REAL):REAL;  
 Prirodzený logaritmus.

PROCEDURE PAGE;  
 Vypíše kód „posuň na novú stranu“. Je ekvivalentné WRITE(CHR(12)). Na CON: sa dosiahne zmazanie obrazovky, na tlačiarňi posunie papier na novú stranu.

PROCEDURE SETOUTPUT(ZAR:RETAZEC);  
 Prepnutie výstupu na iné zariadenie. O zariadeniach je ďalej.

PROCEDURE PEN(I:INTEGER)\*  
 Nastavenie farby písmen, pozadia a spôsobu kreslenia (rovnaké ako v BASICU).

PROCEDURE GOTOXY(X,Y:INTEGER);  
 Nastavenie kurzora zariadenia CON: na X-tý stĺpec a Y-tý riadok.

PROCEDURE Iplot(X,Y:INTEGER);  
 Nakreslí bodku na súradnici X,Y. Súradnice sú v intervale 0 až 255 a presunie grafický kurzor na súradnicu X,Y.

PROCEDURE Imove(X,Y:INTEGER);  
 Presunie grafický kurzor na súradnicu X,Y.

PROCEDURE IDRAW(X,Y:INTEGER);  
 Nakreslí čiaru z predošlej pozície grafického kurzoru do bodu X,Y a presunie aj grafický kurzor.

PROCEDURE Bplot(X,Y:INTEGER; OBR:RETAZEC);  
 Vykreslí obrázok priamo do pamäte obrazovky — je to obdoba príkazu Bplot v basicu. X je od 0 do 47, Y od 0 do 255 (na rozdiel od Iplot sa počíta zhora). Prvý bajt parametra OBR musí obsahovať počet bajtov obrázku, druhý bajt určuje, koľko bajtov sa kreslí vedľa seba (teda šírku obrázku v šesticich bodov), zvyšné bajty sa vykreslia na obrazovku operáciou XOR. Bajty sa kreslia zhora nadol a zľava doprava. Napríklad Bplot(0,0, ## 8 ## 2'????????') nakreslí plný obdĺžnik šírky 12 bodov (2X6) a výšky 4 body v ľavom hornom rohu obrazovky.

PROCEDURE OUT(PORT,BYTE:INTEGER);  
 Vyšle hodnotu BYTE (v intervale 0..255) na port PORT.

PROCEDURE USR(ADRESA:INTEGER; VAR A,HL:INTEGER);  
 Vykoná sa podprogram v strojovom kóde (väčšinou z monitora PMD) so zadanou adresou. Pred volaním sa do registrov A a HL umiestnia hodnoty parametrov A a HL. Po návrate sa do parametra A uloží obsah registra A, do premennej HL obsah páru HL. (Z parametra A sa využíva len nižší bajt.)

## 5.8 Zariadenia

Uvedené mená zariadení používame v procedúre presmerovania výstupu SETOUTPUT (napríklad SETOUTPUT('LST:') presmeruje výstup na tlačiareň).

CON: Konzola. Znaky zobrazujú na obrazovku, znak CHR(13) spôsobí prechod na nový riadok. Znak CHR(12) spôsobí vymazanie obrazovky. Znak CHR(8) spôsobí posun o jeden znak doľava. Neodporúčame ho však používať, lebo pri použití v prvom stĺpci môže zničiť obsah zápisníkovej pamäte.



- LST: Tlačiareň — normálny výstup. Počítač zisťuje typ prípojnej šnúry tlačiarne a podľa neho posiela znaky nenegované (ako PRINT  $\neq$  43 v basicu PMD-85-2) alebo negované (ako PRINT  $\neq$  45). Ak je potrebné znaky negovať (pre rozhranie IRPR), musí byť pin č. 11 (PC5) uzemnený (teda spojený s pinom č. 1). Ak pin nie je uzemnený, znaky sa nenegujú.
- LSN: Tlačiareň — negovaný výstup. Zodpovedá výstupu pomocou PRINT  $\neq$  45 v basicu PMD-85-2. Treba použiť v prípade, že tlačiareň potrebuje výstupné znaky negovať a nie je možné vyrobiť vhodný kábel.
- GR: Grafická obrazovka. Znak sa píše od pozície grafického ukazovateľa (podobne ako príkaz LABEL v basicu, nedá sa však riadiť veľkosť písma). Znak s kódom 13 (CR) a 10 (LF) nemajú žiaden účinok.

## 6 Rozdelenie pamäte

Pre PMD-85-1 a PMD-85-2 je pamäť rozdelená takto:

adresy	obsadenie
0000—0003	skok na začiatok — teplý štart
0008—003F	adresy pre inštrukcie RST (počas prekladu sa používajú RST 1 až RST 7, počas behu len RST 1 až RST 6. RST 7 je počas behu voľné)
0040—004F	tabuľka display (používa sa pri behu programu)
0050— > <—2DAF	miesto pre globálne premenné, za nimi je uložený preložený program vnútorný tvar textu programu, pred ním (smerom od vyšších adries) zásobník počas prekladu a počas behu (počas behu sú v ňom parametre, lokálne premenné a návratové adresy procedúr a funkcií)
2DB0—7DAF	systém Pascal PMD
7DB0—7F00	voľné pre definovanie klúčov (počas editovania sú klúče použiteľné), ak nepoužívame klúče, je táto oblasť voľná.
7F00—8000	používa monitor PMD
8000—8FFF	monitor PMD (ROM)
zápisník	tabuľka identifikátorov a tabuľka symbolov kompilátora

Pre PMD-85-2A a PMD-85-3 je pamäť rozdelená takto:

adresy	obsadenie
0000—0003	skok na začiatok — teplý štart
0008—003F	adresy pre inštrukcie RST (počas prekladu sa používajú RST 1 až RST 7, počas behu len RST 1 až RST 6. RST 7 je počas behu voľné)
0040—004F	tabuľka display (používa sa pri behu programu)
0050—2DAF	miesto pre globálne premenné, za nimi preložený program
2DB0—7DAF	systém Pascal PMD
7DB0—7F00	voľné pre definovanie klúčov (počas editovania sú klúče použiteľné), ak nepoužívame klúče, je táto oblasť voľná
7F00—8000	používa monitor PMD
8000—8FFF	monitor PMD (ROM)
9100—BFFF	vnútorný tvar textu programu, pred ním (smerom od vyšších adries) zásobník počas prekladu a počas behu (počas behu sú v ňom parametre, lokálne premenné a návratové adresy procedúr a funkcií)
zápisník	tabuľka identifikátorov a tabuľka symbolov kompilátora

Na záver:

Pripomienky a postrehy ako aj správy o závažnejších nedostatkoch napíšte výrobcovi na adresu:

**STROJOVÁ A TRAKTOROVÁ STANICA**  
**kombinárny podnik**  
**028 12 TRSTENÁ**

*Vhodná doplnková literatúra:*

1. Molnár — Programovanie v jazyku PASCAL  
Vydal SNTL v r. 1987  
(MŠ SSR a ČSR — schválené ako celoštátna vysokoškolská príručka)
2. Jinoch, Müller, Vogel — Programovanie v jazyku PASCAL  
Vydal SNTL v r. 1987
3. Pecinovský — Seriál PASCAL  
v časopise Elektronika 1988



Poznámky:



Účelová publikácia  
bez jazykovej úpravy

Predajné len pre socialistické organizácie.



Vyrába: STS k. p. Trstená  
028 12 Trstená

Tel.: (0847) 926 112

Telex: 75 472

Tlač: Tlačiarne SNP š. p. Martin, prev. Dolný Kubín



## Dodatok k príručke Pascalu PMS V1.02

Na niektorých počítačoch Pascal PMD vôbec nenabehne - nikdy nesedí kontrolný súčet (a vypisuje sa len Os ready).

Väčšinou je príčinou slabý zdroj +5V. Modul Pascal PMD bol odskúšaný so zdrojmi DAN a PMD 10. S týmito zdrojmi funguje bezchybne. Zdroje EA 1601 sú väčšinou vzhovujúce, niekedy však treba nastaviť prúdovú ochranu na vyššiu hodnotu.

S provizornými zdrojmi (napríklad prerobenými z transformátorov Piko a pod.) modul Pascal PMD nemožno používať!

V ďalšom popíšeme niekoľko drobných chýb, ktoré sa našli v systéme Pascal PMD V1.02 po jeho odovzdaní do výroby (vo verzii 1.01 sa už nevyskytujú - tá bola dokončená neskôr). Zároveň navrhujeme spôsob, ako tieto chyby obísť prípadne opraviť (opravovať ich má zmysel len vtedy, ak nutne potrebujeme použiť vlastnosť, ktorá by bez odstránenia chyby nefungovala správne).

1. procedúra BPLOT(X,Y,Z) pre X=0 a Y deliteľné štyrmi vyhlási vždy chybu 84 (v dôsledku toho nie je použiteľný ani ukázkový príklad v príručke).

Túto chybu môžeme obísť tak, že pri používaní BPLOT nebudeme používať prvý (teda nultý) stĺpec obrazovky. Program teda navrhujeme tak, akoby mala obrazovka len 47 stĺpcov.

Chybu opravíme, ak na adresu 33A8H uložíme 05H a na adresu 33B5H uložíme 00H. To môžeme spraviť tak, že po nabehnutí Pascalu vyjdeme z neho do monitora PMD (voľbou Koniec v hlavnom menu) a tam napíšeme príkazy SUB 33A8 05 a potom príkaz SUB 33B5 00. Do Pascalu sa vrátíme stlačením ↑ a DEL (alebo JUMP 0000). Nezabudnime, že túto operáciu musíme spraviť po každom prečítaní Pascalu z modulu ROM (teda po zapnutí a po každom stlačení ↑RST).

Ak nechceme chybu opravovať zakaždým ručne, môžeme to urobiť v programe takto

- do deklarácií hlavného programu pridáme:

```
VAR BP1:CHAR ABSOLUTE $33A8; BP2:CHAR ABSOLUTE $33B5;
```

- na začiatok hlavného programu (alebo do procedúry, ktorá vykonáva inicializáciu) pridáme:

```
BP1:=#5; BP2:=#0;
```

2. Pri zápise na GR: nemôžeme písať na úplne prvú pozíciu hore (teda Y-ová súradnica 248) a na posledné dve pozície vpravo (X-ová súradnica 249 a 250) - vznikne chyba, hoci by sa na tieto pozície ešte písmeno vošlo (je vysoké 8 bodov a široké 6 bodov, bod, ktorý máme nastavený pomocou IMOVE je ľavým dolným bodom písmena).

Chyba väčšinou nepôsobí žiadne ťažkosti, obídeme ju (alebo si ju vôbec nevšimneme) ak na tieto pozície nepíšeme text.

Chybu môžeme opraviť tak, že na adresu 3526H napíšeme hodnotu FBH a na adresu 355AH hodnotu 07H. Pri oprave môžeme použiť metódy analogické ako pri chybe č. 1.

3. Zariadenie GR: nevie zmazať obrazovku. Teda ak sme použili SETOUTPUT('GR:'), nefunguje procedúra PAGE.

Táto chyba sa nedá odstrániť. Obrazovka sa dá zmazať procedúrou PAGE len ak je výstup nastavený na CON:



STS n. p. TRSTENÁ	KONTROLNÝ LÍSTOK
Počet kusov 1	Súčiastka, číslo výkresu: V402 1592
Dátum: 28. marca 1989 OTK	Vyhotovil: Halušová

STS k.p.Trstená 028 12  
Tel:(0847) 926 112  
Telex: 754 72

#### ZÁRUČNÉ PODMIENKY

Výrobok bol pred odoslaním riadne od-skúšaný a splňuje platné ČSN a TP.V prípade riadne vyplneného záručného listu poskytuje výrobca na výrobok záruku po dobu 6 mesiacov odo dňa predaja zákazníkovi, v ostatných prípadoch najviac 12 mesiacov od dátumu výroby. Záručná doba sa predlžuje o dobu, počas ktorej bol výrobok v záručnej oprave. Záruka sa nevšta huje na závady vzniklé neodbornou manipu-láciou, poškodením pri preprave a skla-dovaní ako aj neodvratnou udalosťou (ži-velnou pohromou a pod.). Právo na záruč-nú opravu zaniká pri nedodržaní inštruk-cí v návode na obsluhu, pri neodbornom zásahu do prístroja, pri porušení plomby a pri strate záručného alebo kontrolného listu. Záručné aj mimozáručné opravy pre-vádza výhradne výrobca STS Trstená.

PRI REKLAMÁCII VŽDY PREDLOŽTE  
ZÁRUČNÝ AJ KONTROLNÝ LÍST

**STS**  
TRSTENÁ

**..TREL**

## ZÁRUČNÝ LÍST

MODUL ROM PMD 85

OBSAH: PASCAL V1 02

VÝROBNÉ ČÍSLO: 1592

Záznamy výrobcu

OTK  
Kontroloval

28. III. 1989  
Dátum výroby

Záznamy predajne

Razítko podpis

Dátum predaja

Záznamy o opravách:

Doba opravy	
Opravil	
Číslo zákazky	
Dátum prijmu	