

K A P I T O L A 1

Složení soupravy počítače, úvod

- vybalení součástí
- složení soupravy počítače
- spuštění počítače
- testování
- kontrolní seznam první pomoci

Nezapínejte

Tento uživatelský manuál je zaměřený na pomoc při využití všech výhod mikropočítače SAM Coupé. Pročtete tuto kapitulu před samotným použitím. Tento manuál je vytvořený na pomoc pro začátečníky, ale i pro uživatele, kteří chtějí využít odborné informace.

Váš SAM
Coupé

Tento moderní počítač poskytuje rozsáhlou paměť, skvělou grafiku a zvuk. Je Vaším prostředkem pro pobavení a především vzdělání a zábavu. Stejnou důležitost ale sehrává pro Miles Gordon Technology, jako pro Vás což má zapříčinu, že si přejeme vědět o každém uživateli a seznamovat ho s nejnovějšími novinkami a výhodami týkajícími se Sama.

Oslovte nás.

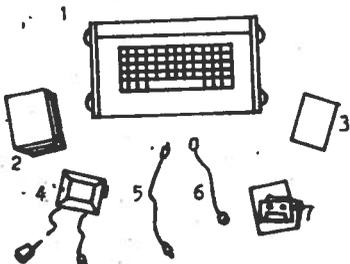
Vyplňte Vaši Registration/Guarante tiskopis a my vykonáme zbytek. Požádáte-li o speciální informace nebo detaily, máme připraven Technický manuál pro pokročilé. Napište nám na adresu/níže/ a nebo zatelefonujte požadavky, nápady, nebo úspěchy. / Po-Pá od 9-19,00 hodin /

MILES GORDON TECHNOLOGY
LAKESIDE, PHOENIX WAY
SWANSEA ENTERPRISE PARK
SWANSEA SA75EH, UK

SAM CUSTOMER CARE LINE: tel. 0792 791100

Co jste dostali to vidíte na obrázku!

Po otevření zkontrolujte zda je všechno v pořádku a identifikujte následovně části.



- 1 - SAM Coupé počítač
- 2 - Uživatelský manuál
- 3 - Guarantee / Registration karta
- 4 - Jednotka na přívod el.energie se zásuvkou na spojení s televizorem,kabel s 6ti jehlovou DIN zástrčkou, které přivádí el.proud do SAMa a další kabel s pojistkou v zástrčce, který bude odebírat energii ze sítě.
- 5 - Spojovací kabel s 3,5 jack zástrčkou na obou koncích, na přenos a přijímání údajů z počítače přes kazetový přehrávač.
- 6 - Propojovací kabel na televizor a / 4 /
- 7 - Naši kazetu dat s příručkou ukázkových programů, včetně " FLASH "

Zapojení
kabelů ke
zdroji

Nezasouvejte do zdroje dříve než vypínač nastavíte na OFF v zadní části počítače. Identifikujte kabel z jednotky / krabičky / zdroje a zapojte do zástrčky na zadní straně SAMa. Je tam pouze jedna 6ti kolíková zástrčka, je vpravo.

Zapojení na
barevný TV

Určete kabel, zapojte do TV a potom do krabičky na zdroj pro počítač. Nyní můžete zastrčit do elektrické sítě zapnout TV přístroj a stlačit tlačítko na ON, které je v pozici IN.

Nastavení
Vyladění

Počítač vysílá signály které se snímají na 36.kanálu běžných TV. U speciálních monitorů a stereo SCART zásuvek se o zapojení dozvíte ve 2. a 9.kapitole. Tato kapitola je aplikovatelná pro běžné UHF televizory. Vyladíte přístroj na 36.kanál až se objeví testovací vzor s následující zprávou

Vítejte
obraz

MILES GORDON TECHNOLOGY plc
c 1989 SAM Coupé 256K

Text se objeví vždy po zapnutí počítače a nebo opětovném nastavování po zhroucení. Číslo na konci oznamuje obsah paměti. Byla-li přispůsobena s extra pamětí, bude to 512K

Testovací
testování

Na přezkoušení spolehlivosti se provede barva a zvuk.

- v manuálu se používá na označení na klávesnici:

BORDER 1

Můžete napsat velkými i malými písmeny, bez rozdílu.

Lokalizujte širokou klávesu / RETURN / a stlačte. Okraj obrazovky by se měl měnit do modra se zprávou

š ok, š : 1 na levém spodním okraji. Nastavte pokud je příliš tichý zvuk a přistupte na červený okraj napsáním

BORDER 2

Následuje RETURN

Přesvědčte se zda zvuk je nastavený na malou hodnotu a vyžkejte na klávesnici

BEEP 1,1

Následuje RETURN. Z reproduktoru by se mělo ozvat pípnutí. Gratulujeme! Jestliže BORDER mění barvy a reproduktor pípá už jste nastartovali programování Vašeho nového počítače. Nyní se jen začtete do ostatních kapitol a najděte, co je důležité pro práci s počítačem.

Kontrolní
seznam 1.
pomocí

Napomůžeme Vám identifikovat možné chyby při zapojení Vašeho SAMa

Problém:

Obrazovka je prázdná

TV ukazuje okrajově bílé tečky, nebo šumí

Obraz je dobrý, ale chybí zvuk

Objevují se cizí geometrické obrazy, nebo znaky na okraji obrazovky, nebo obrazovka černá

Pokud se vyskytne jiná porucha zavolejte naši linku.

Pomoc:

skontrolujte ON - OFF. Vypínač na ON pozici. Zkontr. TV zde je ON. Zkontr. zástrčky, pojistky na zástrčce a ostatních výstupech

Zkontrolujte ON na počítači, naladěni na 36. kanál, zapojení kabelu TV a počítače

Zkontrolujte ovládnání zvuku na TV. Zkontrolujte TV kanál na jemné doladění

stlačte kterékoliv tlačítko, stlačte BREAK tlačítko, vzadu vlevo na počítači stlačte RESET tlačítko, nalevo konektorové zásuvky v zadní části počítače

KAPITOLA 2

Co je co ?

- co způsobují zásuvky, porty a tlačítka
- výlet po klávesnici
- péče o Váš počítač a jeho software

Zásuvky,
porty a
tlačítka

Jestli že si představíte Váš počítač jako inteligentní matku - přítelkyni, potom všechny druhy pomocných modulů a komunikačních vynálezů spojených s ní, dávají počítači ještě vyšší využitelnost.

Přední část
počítače

SAM má zabudovanou rozsáhlou paměť a může používat informace uložené na magnetických páskách a discích. S paměti a disky se budeme zaobírat v kapitole 10. MGT vyrábí speciální DISK DRIVE, které jsou vsazené přímo do vnitřku počítače a přístup je přes úzké štěrbinu napravo k DISK DRIVE 2 a nalevo k DISK DRIVE 1.

Zadní část
počítače

Pro spojení s vnějším světem slouží speciální zásuvky, porty a tlačítka. Podle zobrazení:

4 - BREAK BUTTON

Po stlačení se zastaví program. Programátoři ho používají jako příkaz ESCAPE, BREAK nebo CRASH.

2 - MIDI OUT

Hudební přístrojový digitální meziplošný výstup pro vysílání hudebních informací. Standartní 7-jehlicový DIN konektor může být použitý do ostatních počítačů v NETWORK / síti /

3 - MIDI IN

Podobná zkratka ale pro vstup, může přijímat pulzy, nebo informace z bubnů, syntetizátorů a ostatních nástrojů jak se objasní v kapitole 7. Konektor 7-jehlicový DIN se může v NETWORK-u použít na odposlouchávání z jiných počítačů.

4 - JOYSTICK PORT

Zásuvka pro standartní 9-jehlicový "Atari" typ zastrčky. Používá se na kontrolu akcí při hrách. Druhý může být připojený za použití speciálního MGT duálního joystick adaptéru.

5 - MOUSE INTERFACE

Zásuvka pro 9-jehlicový DIN "myš" konektor. Používá se na pohyb po obrazovce. Lze ji zakoupit u MGT

6 - RESET tlačítko

Po stlačení nastane vymazání programu. Kvůli upřesnění přečtete souhrn.

7 - EXPANSION CONNECTOR

Tato mezíplocha je standardní 64 jehlicový Eurokonektor. Rozšířením počítačového systému se zabývá kapitola 9.

8 - CASSETTE JACK

Standardní 3,5 mm mono jacková zásuvka slouží na připojení kazetového přehrávače. Když se přijímají informace "IN" má být zapojený do EAR zásuvky v rekordéru. Když vysílá informace "OUT" má být zapojený do MIC zásuvky. Pokud to rekordér umožňuje, tak je možné zapojení EAR a MIC současně.

9 - Světelné pero a stereo zvuk

Standardní 5ti jehlicový DIN konektor se používá jak na vstup pro světelné pero a světelnou zbraň, tak i pro stereofonní vstup, pokud se nasadí na AUX vstup zvukového systému

10 - Tlačítko ON - OFF

Tlačítko přerušuje přívod el. energie pokud je v pozici OFF. El. energii propouští do počítače v pozici ON.

11 Zásuvka pro SCART

Speciální 21 jehlicový přispůsobený k použití všech video a stereo výstupů. Pročti kapitolu 9, rozšíření systému

12 - Vstup energie

Dvou účelový 6ti jehlicový konektor DIN, který pojme 15 voltů z krabičky přívodu el. energie. Nese i video signály a mono zvuk zpět do zásuvky umístěné v zásuvce. Pročti kapitolu 1.

Předtím než začnete číst dále, ubezpečte se, že přesně poznáte funkce zásuvek a tlačítek na zadní části počítače.

Viz str. 11 originál manuálu.

Vrch počítače

Položte si ruce na počítač. Šikmý povrch jako u závodních automobilů mu zavalil jméno Coupé. Dizajnerská koncepce MGT počítala s praktičností, ale i s estetickou stránkou počítače. Při porovnání se staršími druhy jistě pocítíte větší pohodlí při použití SAMA. Celý soubor klíčů slouží ke komunikaci s počítačem, ale jsou o mnoho lepší než na obyčejném psacím stroji, protože můžete do jejich adres vložit různé příkazy.

Výlet po klávesnici

Předpokládáme, že jste začátečník a nepoznáte funkce jednotlivých klíčů. Nyň se seznámíme s tlačítky zobrazenými na straně 12 originál manuálu a ohraničenými silnou čarou. V celém manuálu se zobrazuje tlačítko, aby vykonalo funkci s daným symbolem v hranatých závorkách, např. A.

Když musí být stlačeno současně dvě tlačítka tak to zobrazujeme v příručce [SHIFT] [A]

Znaky a mezery

Znaky orámované silně na obrázku můžete najít na běžném psacím stroji. Horní řada zahrnuje čísla a znaky. Tři střední řady reprezentují písmena abecedy a některé znaky a spodní řada je zabraná mezerníkem.

[SHIFT]

Tlačítko se nachází na obou stranách poslední řady písmen. Při současném stlačení s písmenem abecedy způsobí zobrazení velkého písmene. Způsobuje také změnu pokud se používá současně s číslicemi a nebo znaky.

[CAPS]

Pokud je třeba psát velkými písmenami celý text stlačte tlačítko [CAPS]. Zpětné převedení do původního režimu vyžaduje opětovné stlačení.

[DELETE]

Protože nemůžete používat žiletku a nebo zmizík na obrazovce televizoru bez toho, že by jste ji poškodili, tak jsme připravili toto tlačítko. Nachází se napravo nahore. Odstraní s ním všechny náhodné chyby při psaní. Stlačením zmizí poslední písmeno. Když tlačítko podržíte, rychle vymazává písmena zprava do leva. Při použití [SHIFT] [DELETE] vymazává opačným směrem. Ostatní tlačítka mají specializované funkce, ale neobávejte se o ně,

Kapitola 2

a zatím si pouze zapamatujte jejich místo na klávesnici.

[RETURN]

Toto úhelníkové tlačítko se nachází napravo ve střední části klávesnice.

[ESC]

Nachází se celkem vlevo nahoře. Obě jsou podrobně vysvětlené v následující kapitole.

Skupina tlačítek vpravo je rozebrána v kapitole 4. Obsahuje 10 klíčů funkcí očíslovaných [F0] až [F9], tlačítko desetinné tečky a čtyři tlačítka na ovládní kurzoru. Také se seznámíte s [EDIT] tlačítkem vpravo dole vedle mezerníku.

Psaní textů a grafika se nachází v kapitole 6, kde jsou vysvětlené zbyvajcí tlačítka. [TAB] [INV] [EDIT] [SYMBOL] [CNTRL]

Péče o Vašeho
SAMA Coupé

Poškodit počítač programem, nebo stlačením chybného tlačítka není možné. Nezkoušejte násilím zasouvat předměty do zásuvek a portů a nebo neodborně manipulovat ve vnitřku. Vám by to též nebylo milé a podobně i počítači. Počítač vyžaduje běžné pracovní prostředí. Udržujte ho čistý, oddálte od mrazu, slunečních paprsků, koupele, vlhkosti a blesku.

Nechte proudit zespolu a ze stran čerstvý vzduch při používání a jeho teplota nepřesáhne teplotu krve. Malé množství tepla vyprodukované při chodu je normální. Používání chybného zdroje je nebezpečné a bouchání do něho, nebo nalévání nápojů nejsou nejlepší myšlenky. Utírejte měkkou hadrou a zabraňte prachu, brusným časícím a roztokům jako je alkohol a benzín ve styku s povrchem.

Programy jsou uskladňované na discích a kazetách, k mŕže dojít k poškození magnetem, nebo radiací. Oddě všechny disky a kazety od reproduktorů, nenechávej na televizorech a nebo pod telefony. Pokud se pohyb a převážíte software dejte pozor na podzemní mag ve vlcích a na x-paprsky na letištích při kont

**Ochraňuj a
zachovej**

Zabraňte působení přímých slunečních paprsků a vyhřívání, sucha a vysoké vlhkosti na data počítače. Za zadním sklem automobilu se nachází nebezpečná zóna. Nestavějte nápoje v blízkosti softwaru, nesnášejí se, a nevyřvejte nic na disky a nebo kazety. Pokud chcete zachovat důležitou část softwaru, tak vylomte na kazetě zezadu kolíky na nahrávání a nebo uložte do speciální tabulky na 3,5 palcovém disku po nahrání dat, aby se zabránilo náhodnému vymazání. Vždy si pořídte kopii důležitého softwaru jak je objasněné v další kapitole.

Každý by měl vědět, že obrazovka má svoje negativa. Zůstat zdravý můžete při dodržení následujících rad. Protože Vám může být zatěžko odejít od SAMA, sedte co nejdále od obrazovky a provádějte krátké přestávky po hodinách. Měkkou látkou se dotkněte obrazovky, aby se e elektrostatické napětí vybil, kůže Vám poděkuje za to že se nedotýkáte obrazovky a nepřikládáte prsty na tvář.

**Péče o TV
přístroj**

Obrazovka přijímače se po dlouhém zobrazení stejného obrázku může zčernat. Proto po čase asi 22 minut stlačte jakékoliv tlačítko a obraz naskočí zpět. Zmizení není chybou, ale slouží jako ochrana obrazovky.

- K A P I T O L A 3

- Co dokáže Software a jak
- Psaní programů
- SAVE
- VERIFY
- LOAD
- MERGE

ROM	Počítač je vybavený několika druhy paměti. ROM /čti jen paměť / je zpracovaná ve výrobním podniku a umožňuje pouze čtení, zachovává si obsah i po vypnutí počítače.
RAM	RAM představuje buňky vnější přístupné paměti, které čekají na uskladnění nových myšlenek a údajů. Obsah z paměti mizí po vypnutí počítače. Takže nám chybí metoda na učení nových myšlenek rychle a precizně. Pomůžou nám programy které jsou vloženy a vytáhneme je příkazem LOAD. Spuštění počítače a testování obrazovky příkazy BORDER a BEEP můžeme provádět po každém spuštění počítače, trvá to pouze pár sekund. Jestliže si program, který jste tvořili celý den chcete uschovat pomůžte Vám příkaz SAVE. Úspěšně s ním uschováte programy na kazetách, nebo discích pro pozdější použití.
Komerčně dostupný software	Pokud chcete používat SAMa k počítačovým hrám, máte štěstí. Vyberete si z tisíců možností. Hry, užitečné výukové programy a pro speciální využití, které vytvořili jednotlivci a nebo firmy můžete koupit pro vlastní použití ve formě nahraných kazet a disků. Jsou označeny jako software. Mimo software speciálně vybaveného pro Váš počítač můžete použít i software počítače SPECTRUM. SAM automaticky přezkouší za použití speciálního programu tento software. Tento program je přiložený na demonstrační kazetě "FLASH". Pokud je všechno v pořádku přetvoří ho do tvaru, ve kterém bude běžet pro Vaše potřeby. Aby jsme byli čestní, musíme nadšeným hráčům poradit, že mohou ukončit čtení manuálu. Ale upozorňujeme, že jim bude chybět představitelství a radost z vlastního softwaru. Potřebujete sledovat pouze instrukce LOAD-u, který je připojený ke hrám. Pokud váháte, prohleďte si instrukce v dalším odseku této kapitoly.
Domácí software	Pokud začnete realizovat vlastní představitelství při tvorbě programů neobejdete se bez následujících příkazů. SAVE - nahrávání do vnější paměti a LOAD - spětné nahrávání do počítače, na monitor.

SAVE

Příkaz, který se používá na přípravu počítače k nahrávání instrukcí na pásku nebo na disk.

VERIFY

Používá se na ověření, že cokoliv právě nahrané je bez chyb a že se to shoduje s obsahem paměti počítače

LOAD

Připravuje počítač na přijetí nových informací, které jsou někde již uloženy. / Na disku, nebo na pásce /

MERGE

Umožňuje nahrání jako LOAD, ale informace vložené do počítače se nevymažou, ale přidají

Uchování
informací

Zapojení
přehrávače

Váš první
program

Číslování
řádků

Pokud jste šťastný majitelé MG T Coupe disk drive, proveďte postup: Vložte ho do Drive 1 přes úzkou štěrbinu na přední straně počítače. Potřebné informace jsou vytištěné v manuálu pro disk drive a v kapitole 10. Následovně informace jsou pro uživatele kazetových přehrávačů. Zapojení je vysvětlené v kapitole 2. Ven se vysílají informace přes 3,5 mm jack zásuvku do zásuvky MIC na přehrávači. Nastavení hlasitosti je třeba vyzkoušet pokusem. Pokud máte na přehrávači ukazovatel, tak správně by hodnota neměla přesáhnout 10 dB. Kazety si vždy zkontrolujte, zda mají vzadu kolíky na nahrávání. Použijte nové kazety.

Nyní jste připravený na otestování systému jednoduchým programem. Program je slovo, které se používá pro soubor příkazů určujících co má SAM provést. Nezapomeňte jak používat [SHIFT] a kde se nachází [DELETE] tlačítko pro případ opravy.

Počítače nemají velkou představitost pokud je nepovzbudíte a potom již vykonají všechny Vaše příkazy od začátku do konce.

Probíhá celkem jednoduše a výhodně. Když se rozhodnete pro změny a potřebujete najít příslušný řádek výpisu příkazů. Každý řádek má vlastní číslo na začátku a běžné číslo stoupá po desítkách. Umožňuje se tím vkládání řádků mezi původní řádky při opravách.

[RETURN]

Aby se každý řádek uložil do paměti musíte za ukončením řádku ho odeslat příkazem RETURN. Neobjeví se na monitoru.

```

10 BORDER 3 [RETURN]
20 PRINT "IMAVEAMNESIA" [RETURN]
30 BEEP 3,0 [RETURN]
40 BORDER 1 [RETURN]
50 PRINT "WHAT is my name?" [RETURN]

```

Při psaní řádků se objevují příkazy na spodní části obrazovky. Po stlačení **[RETURN]** se objeví v horní části, spolu se znakem >, která ukazuje řádek, kterým se právě zabýváte. Tento znak se nazývá indikátor, více se o něm dozvíme později.

Spuštění programu

Program spouštíme příkazem **RUN [RETURN]**. Potom se ozve pípnutí, změní se barva obrazovky a na konci se objeví OK. zpráva, aby se určil řádek, kde se program zastavil. Program se může předvádět stále po stlačení **RUN [RETURN]**

SAVE

Pokud si chcete připravit šálek čaje, tak před vypnutím počítače si nahrajte náš program "amnesia". Potom ho můžete znovu nahrát zpět pomocí příkazu **LOAD**.

Použijte: **SAVE "amnesia" [RETURN]**

Údaj se objeví na obrazovce, připravený počítač čeká na nahrání programu na kazetu. Kabel je propojený z IN do MIC zásuvky na přehrávači. Potom stlačte libovolnou klávesu. Proveďte se příkaz. Střed monitoru se vyprázdní po dobu čekání na nahrávání, okraj mění barvu. Úzke barevné proužky oznamují, že program je nahráný. Zakrátko se objeví údaj OK a můžete zastavit nahrávání.

VERIFY

Na zajištění dvojnásobné spolehlivosti záznamu na pásku a v paměti počítače přetočte zpět pásku. Vyměňte zapojení kabelu do zásuvky EAR a vypište:

VERIFY "amnesia" [RETURN]

Pusíte zpět nahráný program. Je-li vše v pořádku, na obrazovce se objeví Program amnesia s potvrzením OK.

LOAD

Zkontrolujte zda je kazeta přetočená a připravená zaslat informace ven cestou EAR zásuvky a napište:

```
LOAD "amnesia" [RETURN]
```

Objeví se to samé jako při použití příkazu VERIFY, jak jste zřejmě očekávali dopředu. Program však zůstane bezpečně uložený na pásece, připravený kdykoliv k použití anebo k úpravě.

Pokud se nedaří na poprvé nahrát program, musíte zkontrolovat zapojení kabelů a přetočit pásku zpět a nastavit hlasitost, až je SAM šťastný.

Údaje o
chybách

Při používání příkazů nahrávání LOAD a SAVE se každá chyba objeví na display údajem "Loading error" Pokud by jste chtěli nahrát program na zařízení, které neexistuje, např. disk drive, jiný počítač, objeví se údaj "Invalid device" /chybné zavezení /

Názvy
programů

Jestliže nspecifikujete název programu, pište:

```
LOAD " "
```

Počítač nahraje první program, na který přijde. Výhoda určování názvů spočívá v jednodušším vyhledávání v řetězci programů zapsaných na pásece kazety. Nyní můžete napsat další program a nahrát ho SAVE na kazetu kde se vynechá mezeru po programu "amnesia".

Nejprve запиšte:

```
50 PRINT "please tell me your name" RETURN
60 INPUT n$ [RETURN]
70 BORDER 6 [RETURN]
80 BEEP 3,12 [RETURN]
90 PRINT "I remember now! Hello"; n$ [RETURN]
RUN
```

INPUT

Jestliže napíšete RUN po těchto příkazech, poprvé se budete moci popovídat s Vaším SAMem. Způsobilo to použití příkazu INPUT, který čeká na napsání údaje na obrazovku, který máte odeslat následovně do programu. Napište své jméno a uvidíte jak SAM odpoví. Nemusíte zasílat vlastní jméno, závisí na Vás na co se cítíte.

Název programu může tvořit až 10-ti znakový název. Označme napsaný program Vaším jménem a nahrajte ho:

SAVE "moje jméno" [RETURN]

Přetočte pásku na začátek a zkuste nahrát **LOAD "moje jméno"**. SAM rozezná, že první program zní "amnesia", ale bude ho vynechávat a hledat dále udaný název programu, až do jeho najití.

MERGE

Vypněte počítač a přetočte kazetu na začátek, znovu zapněte počítač. Opět proveďte **LOAD "amnesia"** a stlačte **[RETURN]**. Nyní vyzkoušíme něco chytré, a sice spojit dva programy. Vypíšte **MERGE "moje jméno" [RETURN]** a zašlete do paměti počítače přesně stejnou cestou jako při **LOAD** proceduře. Vyzkoušejte výpis.

Uvidíte, že 50 řádek "amnesia" vymění řádek 50 "moje jméno" a program je nyní 9 řádků dlouhý. Spusťte ho, a radujte se z prvního vlastního programu. Je možná jednoduchý, ale neznamená zlý začátek.

Závěrem

Právě jste udělali první kroky na půdě počítačového jazyka "BASIC". Následující kapitoly jsou věnované seznámení s tímto jazykem. Dříve než se pustíte brázditi neznámou půdu, zapamatujte si 4 následovné rady. SAVE "jméno" pojmenuje program a přikáže počítači, aby zaslal kopii z paměti, která má být nahraná LOAD " " nahrává první program identifikovaný počítačem přičemž se vymezává starý obsah paměti LOAD " jméno " nahrává programspecifikovaná jménem. Obchází jiné programy nahrané na pásce. MERGE " jméno " přihraje do paměti počítače nový program, přičemž starý nevymaže, pokud nejsou totožné programové řádky. VERIFY " jméno " kontroluje stejnost právě nahraného programu s obsahem paměti. Musí mít stejný název.

[ESC]

Na závěr. Pokud se dostanete do problémů při chodu programu můžete problémům uniknout stlačením tlačítka **[ESC]**. Zastaví chod počítače a vstoupí do paměti. Po **ESC** jednoduše stlačte **[RETURN]** a zobrazí se výpis programu.

KAPITOLA 4 S A M B A S I C

* BASIC pro začátečníky

- klíčové slova
- REM - poznámky, spuštění, zastavení
- PRINT - tisk
- rozhodování IF a THEN

* Běžný BASIC

- řetězce, proměnné a matice
- ELSE, LOOP, FOR-NEXT, STEP
- podprogramy
- výpočty
- funkce
- zdokonalený BASIC

* Pokročilý SAM BASIC

- náhodné čísla
- DATA, READ, RESTORE, DIM
- DO, WHILE, UNTIL
- AND, OR, NOT
- DEF, FN
- řetězcové proměnné
- procedury, REF a LABEL
- systémové proměnné

Tato kapitola pojednává o práci SAM Coupé a jeho jazyku. Vylepšená verze je dílem Dr. Andy Wright a pro uživatele jednodušších BASICů je příjemným překvapením. Zkušení programátoři si můžou vybrat z potřebných informací. Hráči můžou ukončit čtení, protože kapitola je věnovaná pro experimentátory nových myšlenek. Členěná je do třech sekcí.

BASIC pro začátečníky

Klíčové
slova

Příkazy jsou v celém manuálu / originál / vytištěné silně velkými písmeny. Jednotlivé příkazy je možné uložit do paměti a není třeba je vypisovat, o tom se však pojednává později.

SAM automaticky vypisuje velké písmena na display i když tisknete malé. Musíte však dát pozor na spojování písmen, extra znaky musí být oddělené mezerou. Např.

```
□ 10 PRINT x [RETURN]
```

se zobrazí PRINT x, ale jestliže zapomenete mezeru za PRINT

```
□ 10 PRINTx [RETURN]
```

počítač to bude považovat za název speciální procedury. Toto se však stává pouze jen s písmeny ale ne s čísly

```
□ 20 print1 [RETURN]
```

a automaticky se objeví PRINT 1

Zadávání
příkazů

Stačí napsat příkaz a odeslat ho příkazem RETURN. Tento však už dále nebudeme uvádět, musíte ho používat z paměti.

Pokud zadáváte příkazy bez chyb, zobrazují se na horní části displeje, ale každá chyba syntaxe se objeví s otázkovníkem. Tím žádá o opravu. Spodní dva řádky slouží na zvláštní zprávy pro odstraňování poruch. Zobrazují se možné chyby na zjištění a odstranění chyb v programu. Nazývají se "error codes" - adresy chyb a vysvětlené jsou všechny v kapitole 12.

Můžete velmi výhodně využívat tuto pomoc, protože každý řádek je možno způsobit na délku podle přání.

může dosahovat 127 individuálních údajů, a každý údaj příkaz /např. PRINT/ následovaný něčím jiným. Program může obsahovat 65279 řádků, takže údaje o chyběch mohou Vám zachránit hodiny vyhledávání.

EDIT a kursor

EDIT Vám významně pomůže při psaní vlastních programů. Po stlačení EDIT se objeví znak >-kursor, před řádkem který má být editován. Pokud chcete editovat konkrétní řádek, napište číslo a stlačte EDIT. Kursorem pohybuje 4 tlačítka v pravé dolní části klávesnice.

SAM používá dva druhy kursorů - inverzní * při normálním stavu klávesnice a při stlačení CAPS velké znaky objeví se inverzní +

DELETE

Jestliže chcete změnit program, smažte staré příkazy před napsáním nových pomocí příkazu DELETE. Zapamatujte si, že stlačení se vymaže znaky nalevo. Současným stlačení [DELETE] a [SHIFT] znaky napravo.

DELETE a TO m

V tomto tvaru můžeme vymazávat celé bloky z programu.

Od řádku l se vymaže až po řádek m. Pokud m není zadáno tak včetně posledního řádku programu. Jeden řádek se vymaže, pokud hodnoty m, a jsou stejné.

Pokud hodnoty m a n existují tak se smaží všechny znaky mezi nimi.

Další způsob vymazávání se uskutečňuje napsáním čísla a odesláním prázdného řádku RETURNem.

REM

Do programu můžete zahrnout krátké poznámky, které vám napomůžou rozeznat jednotlivé sekce vašeho programu. Poznámky jsou velmi výhodná a praktická pomůcka.

□10 REM amnesia

můžete si ji kdykoliv vyvolat, anebo připsat k ní text.

Interpunkce v programu	Pokud potřebujete použít více příkazů v jednom řádku musíte je oddělit, aby se zamezilo nejednoznačnosti.
Dvojtečky	Používají se na oddělení jednotlivých příkazů. □ 10 PAPER 6: PEN 2: PRINT "red on yellow"
Čárky	Na oddělení znaků stejného významu se používají čárky. □ 10 READ x,y,z 20 PRINT x,y,z 30 DATA 1,2,3,4,5 Před zahájením psaní počítačové hry a opisováním řádků z tohoto manuálu si musíte ozřejmit několik koncepcí v BASICu.
Spuštění, zastavení	Existuje několik cest jak zvládnout postup ve vašem programování, ale základ je vědět jak spustit program.
RUN	Spuštění programu se provede po RUN a [RETURN]. Pokud chceme spustit program od určitého řádku napíšeme RUN n [RETURN] n je číslo řádku
STOP	Je povel k zastavení programu. Zastavuje program na každém předdefinovaném bodě.
PAUSE	Umožňuje v určité části programu získat potřebný čas. Stisknutím PAUSE, nebo PAUSE 0 se zastaví program na vždy, pokud chcete pokračovat musíte stlačit libovolnou klávesu a program se opět rozběhne. Kratší zastavení se určuje číslem, což znamená zastavení na rámeček. Jedna sekunda má 50 rámečků. Tímto si můžeme spočítat potřebný čas na zmravení programu, například: □ 100 PAUSE 100 zrazí obraz na jednu pětinu sekundy
Čekání na vstup	INPUT - uvědomuje počítač, aby čekal na vložení údaje. Použijeme STOP, INPUT, GOTO na napsání hry na konci této sekce.

Skákání

GO TO - je klíčové slovo v BASICu. Můžou se psát odděleně i spolu. Tímto příkazem se skáče na daný řádek programu.

```

10 REM greetings
20 PRINT "happy birthday to you"
30 PAUSE 50
40 GO TO 20

```

Jestliže spustíte program, tak se Vám objeví každou sekundu blahopřání k narozeninám. Před zaplněním obrazovky, ale zastavte program s ESC

Pokračování

CONTINUE

Program se může zastavit při chybě, anebo při úmyslném přerušení. Opětovně ho rozeběhneme příkazem CONTINUE. Spusťte nyní daný program, až se obrazovka zaplní a počítač se zeptá jestli chcete "scroll" scroll ? - se objeví, když počítač zjistí, že obrazovka je zaplněná znaky a diví se, že chcete pokračovat ve výpisu. Není to příkaz což dokazují malá písmena. Stlaďte nyní [RETURN] na svinutí obrazu.

Smazání
obrazu
CLS

Pokud potřebujeme při zastaveném programu vymazat obrazovku a uvolnit ji pro další činnost použijte příkaz CLS. Tento příkaz smaže vše co je na obrazovce.

NEW

Velmi silný příkaz, proto ho používejte velmi opatrně. NEW vymaže celý program se všemi proměnnými a jediné co zanechá je prázdná obrazovka. Spůsobí to samé jako stlačení tlačítka RESET v zadní části počítače. Spůsobí zanesení celé vaší práce do zapomění.

Tisk

PRINT - specifikace se používají na zobrazení znaků. Existují jednoduché pravidla interpunkce a první říkají o použití úvozovek. Pokud se v paměti počítače nachází údaje, zbavte se jich použitím NEW a [RETURN]. Potom pečlivě vypíšte:

```

10 PRINT "G'day: how are you, SAM?"

```

Stlaďte [RETURN] a RUN. Všechny znaky mezi úvozovkami se

interpunkce
tisku

zobrazí na obrazovce.

Při této specifikaci sehraává interpunkce důležitou úlohu. Čárky usměrňují tisk od levého okraje, anebo od následující police při TAB. To závisí od toho co se nachází blíže. Vyskoušejte spustit:

```
□ REM columns
  10 PRINT 1,2,3,4,5,6,7,8,9,10
```

Středníky usměrňují tisk znaků od levého okraje za sebou.

```
10 PRINT 1;2;3;4;5;
```

Apostrofy způsobují tisk na levém okraji řádku.

```
10 PRINT 1'2'3'4'5'
```

Vyzkoušejte vícero kombinací tisku s interpunkčními znaménky.

Více údajů o PRINTu se nachází v kapitole 6. Příkazy výstupu na tiskárnu jsou rozebrány v kapitole 9. Takže tato kapitola přinesla hodně teorie a téměř žádnou zábavu. Ale můžete využít první nápady a uskutečnit vaše první programátorské pokusy.

rozhodování

Pokud by počítač pouze vykonával soubor instrukcí, řádek za řádkem, programy by byly velmi omezené a úmorné. Zážrak se začíná projevovat když se počítač zamyslí a začíná rozhodovat.

IF, THEN

Když chceme poskytnout možnost počítači na rozhodování, potřebujeme mu dát podmínku IF/když/ pravda=true, chyba=false THEN /potom/. Používají se následovně symboly: =, < > /nerovný/, >, <, >=, <=

Použijte NEW na vymazání starých programů z paměti počítače. Vypište další program

```
□ 10 REM secret numbers
  20 PRINT "Think of a number between 1 and
    100 and type it in secretly"
  30 INPUT "a"
  40 INPUT " Find the secret number", b
  50 IF b > a THEN PRINT "WRONG...try loWer"
```

```

60 IF b < a THEN PRINT "WRONG,go higher."
70 IF b = a THEN PRINT "RIGHT,well done!":
STOP
80 GO TO 40

```

Znaky <a> se zobrazí po stlačení [SYMB] [a] a [SYMB] [w]. Nechte přítele vymyslet a napsat tajné číslo, které Vám počítač pomůže objevit. Pozorujte jak GO TO, PAUSE, IF a THEN pracují, jak se zobrazuje příkaz PRINT, jak STOP zastaví hru když číslo uhodne.

Běžný
BASIC

Následující sekce představuje páteř programovacího jazyka. Najdete tu všechny návody a myšlenky potřebné na psaní programů. Musíte se být vždy jistý, že ovládáte dané pojmy a nepředbínáte přes text bez utvrzení vědomostí. Nejdřív je potřebné ozřejmit si pojmy - řetězce, proměnné, matice.

Řetězec

Je to jednoduché spojení určitého počtu znaků/písmena, čísla, mezery nebo značky/. Délka může být libovolná do 65 520 znaků. V programu se deklaruje každý řetězec jménem a hned za ním se přiřazuje speciální symbol. My používáme symbol "dollar".

```

□ LET sam $ = " společník je doma "

```

Podřetězce

Vidíte, že sam S pozůstává z písmen a mezer. Pokud by jste určili délku řetězce číslem, tak by se zobrazila jistá část řetězce, nazývá se podřetězec. Pokud rozumíte pokračujte dále.

STRINGS \$
/n, a \$ /

Řetězec opakujeme podle zadaného čísla v závorce, slouží na to tento příkaz. Spusťte tento malý žert:

```

□1C PRINT STRINGS / 8, " ha "/

```

Nyní jste připravený na omnoho důležitější myšlenky.

Používání proměnných

Proměnné předstávají hodnoty, které reprezentují jisté znaky. Při práci s počítačem slouží velmi účinně na výkonnost počítače. Zapisují se:

```
□ 10 LET x=2
  20 PRINT "THE ANSWER IS "; x+x
```

Což se zobrazí jako odpověď je 4. Změňte hodnotu x a zkoušejte dále.

Matice/pole

Často budeme používat celou sadu podobných proměnných. Pole je skupina takových proměnných, které jsou odlišené od sebe navzájem číslem v závorce za jménem pole. Toto číslo znamená rozměr a prostor, který se rezervuje v paměti počítače.

Jméno pole může obsahovat 10 znaků, nesmí obsahovat mezery. Číslo v závorkách může být v rozmezí 1 - 65535

DIM

Každá matice/pole/ musí být určena. /dimenzovaná/. Toto provádíme příkazem DIM

```
□ DIM a /100/
```

Nyní vykročíme napříč přes mnoho vyšší obtížnost.

LET

Už jsme se zmínili o jednom z nejdůležitějších příkazů systému jazyka BASIC. Příkaz LET je srdcem všech BASICů. Přiřazuje hodnoty proměnným.

```
□ 10 LET x=1
```

SAM vám umožní použít jich v jednom řádku více, je však třeba oddělit je čárkou. Například:

```
□ 10 LET x=1, flag=2, alpha="yummy"
```

Numerické proměnné

Proměnná, která reprezentuje číslo se nazývá numerická proměnná. Jméno musí začínat písmenem, následovat může písmeno nebo mezera, či číslice o celkové délce 32 znaků. Mezery se nepočítají za znaky. Přípustné jsou: x, SAM je doma, nebo R2D2. Ale 2R2D není přípustné, jako i klíčová slova PRINT, FOR-NEXT a pod.

Řetězce a pole

Proměnné se běžně používají v řetězcích a polích. Jméno třeba však musíme omezit na 10 znaků a musí následovat znak \$. Např: A\$, R2D2\$ a GUITAR\$

CLEAR

Pro vynulování proměnných se používá příkaz CLEAR. Po napsání se nulují všechny proměnné v programu a uvolňuje se paměť, kterou obsazovali. Pokud jste zadali proměnnou, vynulujte ji a připravte se na následující důležitý text. Pokud je potřebné prostudujte poslední část odseku "BASIC pro začátečníky", který pojednává o příkazech IF a THEN. Začne se nám té více zamotávat.

ELSE

Příkaz ELSE musí přijít na začátek nové specifikace řádku. Používá se ve spojitosti s dvojicí příkazů IF-THEN, kde následuje za nimi. Proveďte se v případě, že podmínka není splněná, v případě, že je splněná k provedení ELSE nedojde. Příklad:

```

10 REM lhař
20 LET x=1
30 IF x=1 THEN PRINT "TRUE":ELSE PRINT "FALSE"

```

Po spuštění se potvrdí pravdivý výrok "TRUE". Po změně hodnoty x ve 20 řádku spusťte znovu. Příkaz s IF může dosahovat řádku, dokonce je možné vypustit THEN. Po spuštění programu skuste změnit hodnotu x:

```

10 REM moody
20 LET x=1
30 IF x=1
40 PRINT "HAHAHA"
50 PRINT "snort"
60 ELSE PRINT "BOOHOO"
70 PRINT "SNIFF"
80 END IF

```

END IF

Pokud x=1 vypíše se na obrazovce "HAHAHA" a "Snort", ale pokud x <> 1 objeví se "BOOHOO" a "SNIFF". Řádky můžou obsahovat víc než jednu specifikaci, ale IF cyklus musí končit s END IF

ELSE IF

Podmínka má speciální význam. Skuste si:

```

□ 5 REM vague numbers
  10 PRINT "type in a number between
    1 and 9"
  20 INPUT x: CLS
  30 IF x=1
  40 PRINT "one"
  50 ELSE IF x < 5
  60 PRINT "less than 5"
  70 PRINT "honestly"
  80 ELSE IF x=8
  90 PRINT "eight"
 100 ELSE
 110 PRINT "something else"
 120 END IF
 130 STOP

```

Je-li $x=1$ provede se řádek 40 a následuje řádek 130
 IF $x=2$, nebo 4 provede se řádek 60 a 70 následuje 130
 IF $x=8$ provede se řádek 90 a následuje řádek 130
 IF x =jiné hodnotě provede se řádek 110 a následuje řádek 130

Když počítač pozná, že podmínka je pravdivá, řádek a specifikace v něm se provedou. Když následuje další ELSE nebo ELSE IF automaticky pokračuje program na END IF. Přípustné je spojovat krátké a dlouhé IF-y uvnitř dlouhého IF-u, ale není přípustné mezi krátké IF zapojit dlouhé IF a sloučené s ELSE.

Chcete vypsat dny v roce? Psaní od 1 do 365 by však Vás rychle zničilo. Existující 4 příkazy Vám práci podstatně ulehčí a počítač se sám pustí do práce. Jejich názvy jsou: FOR, NEXT, TO a STEP

FOR - NEXT

Využívají se na zkrácení opakovaného programování. Spusťte:

```

□ 10 REM calendar
  20 FOR day=1 TO 365: PRINT day: NEXT day

```

TO

V příkazu FOR-NEXT způsobuje zvyšování hodnoty.

STEP

Specifikováním kroku STEP se určuje hodnota o kterou se zvyšuje základní hodnota. Bez udání STEP se zvyšuje o jednotku.

```
□ 20 FOR day=1 TO 365 STEP 7:PRINT day:NEXT day
```

Lehké, co? STEP může být i desetinné číslo, ale i záporné

```
□ FOR n=1 TO 100 STEP .142
```

```
□ FOR x=500 TO 250 STEP -25
```

FOR-NEXT proměnné zastupují běžné číselné proměnné, takže se nepotřebujete učit nic mimořádného. Pokud se nepokojíte pouze s výpisy, tak v dalším programování jsou vám nápomocné následavné příkazy z poslední sekce této kapitoly. DO, LOOP, WHILE, UNTIL, EXIT IF.

Podprogramy
podprocedury

Soubor instrukcí zavedených zasebou se nazývá procedura. Často potřebujete v programu provést podobné úkony. Aby jste to nemuseli únavně programovat více krát v jednom programu, stačí vytvořit samostatný podprogram. Tento můžete volat kdykoliv a kolikrát budete potřebovat.

GOSUB

Pokud jste pochopili princip příkazu skoku GO TO, není co vysvětlovat:

```
□ GOSUB 500
```

Instruuje počítač, aby opustil současnou pozici a skočil rovněma nohama na řádek 500, kde provede požadovanou operaci.

RETURN

Je vhodné poznačit si každý podprogram názvem pomocí REM specifikace. Nepovinné označení vám umožní lehké vynalézání při případných opravách. Příkaz RETURN povinně vrátí program/proceduru/ zpět do hlavního programu.

```
□ 500 REM pretend subroutine
  510 ....all the lines that form this subroutine
  ⋮
  600 RETURN
```

Jsou velmi užitečné, jak se dozvíte v kapitole 8.

Výpočty

Pokud se Vám zdá, že se celá teorie zabývá složitými programy, tak BASIC umožňuje i běžné jednoduché operace. SAM-a můžete využít jako jednoduchou kalkulečku. Stačí napsat PRINT matematický výraz a stlačit RETURN. Není třeba použít RUN

□ PRINT 2+2

Počítač splní následovné výrazy:

+ sčítání

- odčítání

* násobení

/ dělení

↑ umocňování : např PRINT 3↑5 je to samé jako

3*3*3*3*3

Výrazy

Při práci s výrazy si musíte uvědomit jednotlivé priority. Násobení a dělení jsou provedené jako první ve směru zleva doprava. Po nich se provádí sčítání a odčítání v tom samém směru. Tento striktní systém můžete obejít použitím závorek. Vnitřek závorek se považuje jako jedno číslo a zpracovává se jako první.

PRINT 10 * 10 - (25 * 4)

Matematické funkce

SAM ovládá matematické a trigonometrické funkce, přičemž používá následovné zkratky. Nebudeme Vás učit matematiku ani trigonometrii a proto pokud nemáte zájem o tyto obory můžete přeskočit následující odsek.

EXP EXP x = e ↑ x

SQR počítá 2. odmocninu z kladného čísla n. Můžeme říct, že hledá číslo, které po vynásobení samým sebou dává opět číslo n.

LN inverzní funkce k EXP, jako základ používá e / přirozený logaritmus / Pro vypočítání logaritmu s jiným základem stačí vydělit přirozený logaritmus přirozeným logaritmem požadovaného základu

INT Argumenty obsahující číslíce po obou stranách desetinné tečky stěžují někdy zbytečně výpočet. Pokud dostačuje tvar celého čísla, tak touto

funkcí se argument vždy zaokrouhlí směrem dolů k nejbližšímu celému číslu, které může být kladné a nebo záporné.

SGM Funkce nám určí matematický argument
 0 je-li argument 0
 +1 je-li argument kladné číslo
 -1 je-li argument záporné číslo

ABS Mění argument na kladné číslo. Učává jeho absolutní hodnotu..

Trigonometrické funkce

Zřejmě očekáváte, že SAM si poradí s trigonometrickými funkcemi, které jsou běžné na kalkulačkách. A on to samozřejmě ovládá.

PI představuje Ludolfovo číslo, které začíná 3,14159653 a jde do nekonečna. Používá se na výpočet kruhu, kružnice, obsahu těles. Prostudujte si obrázek na straně 46 originál manuálu.

SAM používá namísto stupňů radiány. Pro změnu radiánů na stupně vydělíte PI a vynásobíte 180. Změna stupňů na radiány se děje vydělením 180 a vynásobením PI.

Funkce SIN, COS, TAN, ASN, ACS, ATN

Tyto matematické a trigonometrické klíčové slova pracují stejně v programu s několika dalšími, se kterými jsme se setkali. Společně se nazývají funkce. Společně mají to, že pracují s čísly a směřují k výsledným hodnotám. Funkcím se přiřazuje argument. Můžou být směřované s jinými druhy operací v jednom výrazu, ale vždy se provedou jako první před těmito operacemi, pokud nejsou umístěné v závorkách.

FREE

Pomocí tohoto příkazu můžete zjistit kolik máte právě volné paměti.

PRINT FREE

a potom vypíšete jeden, nebo dva řádky programu. Znovu vyzkoušejte a zjistíte kolik z paměti se obsadilo. Jak se počítá paměť a jak se zaplňuje se dozvíte v kapitole 10.

LEN	<p>Tento příkaz nám řekne jaká je délka řetězce. Vyzkoušejte:</p> <p style="text-align: center;"><input type="checkbox"/> PRINT LEN "SAM"</p> <p>I když jste možná ještě stále začátečníci v programování Vašeho SAMa, vaše netrpělivost jistě stoupá. Zkuste nahrát jeden delší program na demonstrační kazetu 'FLASH'.</p>
LIST	<p>SAM Vám vypíše program, který právě máte v paměti. Probíhá od začátku do konce. Užitečnější bývá vypsat si program od určitého řádku, a nebo jeho jistou část. Např.:</p> <p style="text-align: center;"><input type="checkbox"/> LIST 500 <input type="checkbox"/> LIST 500 TO 1000</p> <p>Jestliže udané číslo řádku neexistuje, přijme se nejbližší první řádek. Podobně se to týká i posledního požadovaného řádku. Zkuste si to.</p>
Jak vylepšit Váš BASIC	<p>Jestliže potřebujete zvýraznit jisté části textu, může Vám pomoci [INV]. Část textu se tak zobrazí v inverzní barvě, kterou je možno do původního tvaru usměrnit současným stlačením [SYMB] [INV]</p>
Automatické číslování řádků	<p>Prvním pravidlem BASICu je povinné číslování řádků a Váš systém by se mohl stát velmi chaotický a neuspořádaný. SAM Coupé Vám chce pomoci svým automatickým číslováním. Přes AUTO určuje každému řádku číslo postupně o 10 větš. Automatik funguje pro čísla od 10 do 6'439. Pokud chcete přezkoušet automatické číslování, tak ho vymažte pomocí DELETE a napište vlastní požadované číslo. Dále bude pokračovat AUTO.</p> <p>Některé variace: <input type="checkbox"/> AUTO dává čísla po 10 <input type="checkbox"/> AUTO 150 dává čísla po 10, od řádku 150 <input type="checkbox"/> AUTO 150,5 dává čísla po 5, od řádku 150</p> <p>Přímé příkazy, jako STOP a vymazání čísla řádku vypnou AUTO</p>

RENUM

Přečíslování řádků se začíná od prvního řádku číslem 10.

RENUM n TO m
umožňuje přečíslovat určitou část od řádku n až po řádek m včetně. Jestliže jsou čísla fiktivní, tak se spouští automaticky od začátku po konec

RENUM LINE L

umožní přečíslovat řádek označený L

RENUM STEP S

S určuje krok přečíslování části programu. LINE a STEP mohou být použité a nebo vypuštěné podle přání. Běžnější jsou ve tvaru:

□ RENUM n TO m LINE L STEP S

Do přečíslování se započítávají příkazy GO TO, GOSUB, RESTORE, RUN, LINE, ON atd., ale GO TO VAL "100" /výpomocné/ jsou mimo. Obrazovka si udržuje přechodné údaje po dobu přečíslování a čistí se po ukončení operace.

Vylepšený
listing

LIST FORMAT nebo LLIST FORMAT

Umožňuje výpis všech příkazů, do samostatných řádků. Je to velmi přehledná forma. Prvních 6 sloupců se rezervuje na číselování řádků a mezery. Program může mít 65279 řádků.

□ LIST FORMAT 1 nebo 2

Provede automaticky upravený výpis s 1 nebo 2 mezerami, za každým klíčovým slovem, přemísťuje klíčová slova z původního tvaru. Tyto slova jsou: IF, ELSE, ENDIF, ON, DEF PROC, END PROC, FOR, NEXT, DO, LOOP, EXIT IF, LOOP IF. Pokud se divíte co tyto slova znamenají, tak se s nimi seznámíte na konci této kapitoly.

□ LIST FORMAT 0

Navrátí výpis do původního neupraveného tvaru. Vyzkoušejte proto výpis na jednom z Vašich programů a pozorujte co se stane. .

POKROKOVĚ SAM BASIC

Náhodná čísla	Představte si, že se SAMab budete každé ráno ptát, "Jak se máš?" a odpověď by byla stále stejná, "Velmi dobře, děkuji." Jaké předpisové přátelství by vzniklo! Nejjednodušším způsobem jak zavést prvek překvapení do programu je vložení náhodných čísel do jisté jako-by nádoby. Náhodně se vždy jedno vytáhne a potom vrátí zpět. Takže může pokaždé se stejnou šancí se při náhodném výběru objevit.
RANDOMIZE	Klíčové slovo, které spouští výběr Používá čísla z řady, která uplyne po dobu zapnutí počítače.
RANDOMIZE n	Určí náhodné číslo vybrané Vaším vlastním výběrem přes n v rozpětí 1 až 65535
RND	Funkce vybere číslo z "nádobky" z pevně určeného řadu 65535 čísel, které se důkladně promíchali. Výsledkem je číslo s pohyblivou desetinnou čárkou. Můžete použít INT funkci k odtrhnutí číslic za desetinnou čárkou. Existuje však jednodušší způsob: <p style="text-align: center;">□ RND /x/</p> Vrátí se celé číslo mezi 0 a x. Nyní se můžete vrátit ke hře s tajnými čísly na straně 36 originál manuálu. Nechte hrát počítač sám. Zbavte se řádku 20 a změňte 30 řádek za <p style="text-align: center;">□ 30 LET ε = RND /100/</p> Náhodných čísel lze tak možno využít ke hře kostek, na rozdávání karet, neustále se měnící galaxii, nebo vyhledávání ze sta možných odpovědí na otázku " Jak se máš dnes?"
DATA	Co když nepotřebujete náhodná čísla, ale potřebujete zadat počítači konkrétní řadu informací? Tyto informace, nebo jinak také DATA můžete zadat kdekoliv v programu. Použijí se podobně jako při zadání přes INPUT, tedy přes klávesnici. Každá datová specifikace má za sebou seznam výrazů oddělených čárkami. Čtení hodnot se provede příkazem READ. Vyskoušejte cyklus:

```

10 READ w,x,y,z
20 PRINT w,x,y,z
30 DATA 1,22,33,4444
40 STOP

```

READ

Zadali jste počítači úkol aby přečetl hodnoty do jednotlivých proměnných x,y,z,w. Napoprvé přečte první výraz, podruhé druhý atd, až přečte všechny hodnoty. Zkuste nyní změnit pořadí proměnných v příkazu PRINT a sledujte co se stane. Na závěr odtrhněte poslední číslo ve specifikaci DATA a sledujte jak počítač marně hledá.

RESTORE

Další využití výrazů v seznamu dat je možné po instruování počítače provést obnovení RESTORE a nebo smazání proměnných. Opravte řádky 20 a 30 našeho posledního cyklu a potom připojte následovné:

```

5 READ p,q
6 PRINT p,q
7 RESTORE 5

```

Spusťte jednou a sledujte výsledek. Smažte řádek 7 a pozorujte jaký měl účinek. Proměnné jsou automaticky obnovované když program u BASICu je nahráván / LOAD / nebo spouštěn / RUN /.

READ LINE

Řetězcové proměnné mohou být též načtené, např.:

```

READ a$,b$,c$
DATA "one""fine""day"

```

READ LINE umožňuje vynechání úvozovek ve specifikaci DATA. Takže když:

```
DATA sam,"dog",123
```

byl částí Vašeho programu, READ LINE a\$ nejprve přiřadí a\$="sam", potom "dog" a nakonec "123" sériovým READem. Každé slovo bez úvozovek použité ve specifikaci DATA musí odpovídat numerické proměnné. Takže sam souhlasí, ale 1x je chybné.

TRUNC\$ a\$

Další důležitou funkcí na výběr DAT z řetězcových polí je schopnost zkracovat je. Mezery se vymazávají. Takže když:

```
DIM a$ /10,10 /:LET a$ /1/="sam":PRINT a$ /1/;
"my"
```

vytiskne "sam my" a po použití PRINT TRUNC\$ a\$ /1/ vytiskne "sammy"

DIM

Název pole může mít 10 znaků, ale nesmí obsahovat mezeru. Dimenze v závorkách dosahuje od 1 do 65535 a celková velikost pole závisí od přístupné paměti. Tento příklad zabere až 200K

```
□ DIM name$ / 10 000,20 /
  DIM coords / 100,2 /
```

LENGTH

Rozev pole můžeme zpětně určit příkazem LENGTH, při jedno a nebo dvojrozměrném poli a proměnlivé adresy. Například po určení a\$ jako / 4,6 /

```
□ LENGTH /2,a$/ vydá 6
```

Na najítí adresy prvního bytu čísla použijte

```
□ LENGTH /0,n/
```

Což je užitečné pro specifikaci CALL, určí se místo proměnné a může být modifikované. Závorky jsou povinné za názvem numerického pole.

```
□ LENGTH ( 1,n )
```

DO-LOOP

Mimo příkazů jako RESTORE a FOR-NEXT můžete vytvářet určité sekce v programu dokonce více užitečnější. LOOP oznamuje, že jistá část programu se opakuje. Příkaz se vždy označuje instrukcí DO, v přesně takovém významu jako FOR-NEXT

```
□10 DO
  20 PRINT "PLAY IT AGAIN SAM";
  30 LOOP
```

Procedura bude běžet navždy. Užitečné je blíže vymezit DO a LOOP příkaz pokud chcete něco opakovat jistý počet krát.

WHILE a
UNTIL

Provádění WHILE probíhá v části programu DO a LOOP, pokud tato částečná podmínka zůstává pravdivá. Je však možné použít i hodnotu v podmínce nepravdivé a tato se zastaví ihned po získání pravdivého ohodnocení, použitím DO UNTIL. Podobně LOOP WHILE a LOOP UNTIL umožňuje podmíněčné opakování na konci DO LOOP. Podmínky se používají na obou-začátku i konci LOOPu pokud chcete. A konečně, IF se může použít dvěma způsoby. EXIT IF / podmínka / se používá na opuštění procedury uprostřed, ne na LOOPu nebo DO. Pokud specifikovaná podmínka po EXIT IF je pravdivá program skočí na specifikaci následující po LOOP. V opačném případě procedura pokračuje dále.

EXIT IF

LOOP IF

LOOP IF podmínka odbíhá v případě pravdivosti podmínky z prostředka procedury zpět na DO. Není-li pravdivá, probíhá procedura až do konce.

AND, OR, NOT

Následuje skupina tří funkcí, kterých názvy se obsahově shodují, s anglickými pojmy AND, OR, NOT. Používají se na kombinaci dvou a nebo více logických operací.
IF /je-li/ něco pravda AND /a/ něco jiné je též pravda THEN /potom/ vztah možno objasnit následovně:

```
□10 IF x = 1 AND sky$="blue" THEN GOSUB 500
```

Vztah bude pravdivý vždy pokud alespoň jedna podmínka bude pravdivá.

```
□10 IF x > 1 OR sky$="blue" THEN PRINT x
```

NOT způsobuje pravdivé věci chybnými a pravdivý vztah nepravdivým, pokud se používá samostatně nebo v kombinaci s AND, nebo OR. Zkuste zadat x, y a s vhodné hodnoty a spusíte program, pokud nepochopíte jak to pracuje:

```
□10 IF NOT x <> y OR NOT a$="sam" THEN NEW:GOTO 10
```

Tento logický výraz můžeme uzavřít závorkami jako numerický příkaz. Nejvyšší prioritu dosahuje NOT, následuje AND a nejnižší má OR. Ve skutečnosti při změnách vztahů se můžeme obejít i bez NOT.

ON

Částečnou specifikací můžete oddělit od zbytku řádku použitím klíčového slova ON

```
□ ON x:PRINT"one":PRINT"two":PRINT"sam":GOSUB 100
```

bude tisknout "one" když x=1, "two" když x=2, "sam" když x=3 a GOSUB 100 když x=4. Pokud x=0 nebo x je větší než číslo specifikace následující za ON, program pracuje s dalším řádkem.

Klíče funkcí

nacházejí se vpravo na počítači. Jsou předdefinované:

```
[F0] LIST [RETURN]
[F1] RENUM [RETURN]
[F2] PRINT
[F3] MODE
[F4] RUN [RETURN]
[F5] CONTINUE [RETURN]
[F6] CLS* [RETURN]
[F7] LOAD "" [RETURN]
[F8] LOAD "" code [RETURN]
[F9] BLOT [RETURN]
```

Obraťte se na kapitolu 6 pokud chcete předefinovat tyto kódy.

Definice funkcí

Názvy funkcí mohou mít jakoukoliv délku, musí začínat písmenem a pokračovat písmenami, nebo číslíci.

Např: □DEF FN charset = DPEEK SVAR 566+256

Názvy řetězcových funkcí musí končit se S

```
□DEF FN Left$(a$,n) = a$( TO n)
```

Funkce nemusí mít závorky pokud neobsahují argumenty, ale každý název proměnné použitý v závorkách bude patřit FN, a proto může mít délku pouze jednoho písmene.

Ukážeme Vám několik užitečných funkcí, které jsou podobné. Jsou vytvořené na pomoc při zpracovávání řetězců.

Změny řetězců

STR\$

mění čísla a číselné výrazy do řetězců. Dává ten samý výsledek jako kdyby číslo se zobrazovalo na obrazovce příkazem PRINT

```
PRINT STR$(30 * 5)
```

VAL

Převádí řetězce zpět na čísla, a řetězec který se používá jako jeho argument může dosahovat jakéhokoliv číselného výrazu. Podobná funkce je přístupná pod názvem VAL\$, používá řetězec jako argument a dává řetězec jako výsledek.

```
PRINT VAL "69 * 2 + 3"
```

CODE

může se použít na zadání prvního písmene v řetězci a když je řetězec prázdný - udává 0

```
PRINT CODE a$
PRINT CODE "x"
```

CHRS

Udává jedno písmenný řetězec pokud je aplikovaný v číselném kodu tohoto písmena. Pročtete kapitolu 6, kódy písmen.

```
PRINT CHR$(65)
```

Vyhledávání řetězců

INSTR (n, a\$, b\$) vyhledá b\$ uvnitř a\$ od pozice n. Jeho pozice se vrací je-li nalezen, nebo výsledek 0 je přiřazený je-li hledání neúspěšné. Nezapomíte-li n, hledat se začne na pozici 1

Procedury

Požádáme Vás, aby jste objevili jisté procedury, které zlehčí programování. Moduly můžete vytvářet na provádění speciálních požadavků bez nepříznivého vlivu na hlavní program. Nejprve je však musíme definovat.

Definování procedur

```
DEF PROC jméno
```

Název musí začínat písmenem, následovat řetězcem, písmenem, číslem nebo \$. Mezery jsou zakázané. Malé a velké písmena se nerozlišují. Za názvem procedury může následovat soubor, což musí být názvy proměnných, nebo můžete použít klíčové slovo DATA. Automaticky

	<p>budou LOCAL pro proceduru. Použijete-li DATA namísto souboru parametrů, nejsou žádné označení použité. Ale soubor aktuálních parametrů, který se objeví po zadání procedury může být zjednaný použitím READ-u a funkce ITEM, což je objasněné níže.</p> <p>Umístění procedury v programu neovlivňuje jeho průběh.</p>
LOCAL	<p>Specifikuje seznam proměnných, včetně polí, které chcete aby byly přiřazeny některé proceduře kdekoliv v programu.</p>
ITEM	<p>je funkce, která pokrývá informaci ohledem dalšího ITEM-u, který se má načíst READ-em ve specifikaci DATA. Hodnota 0 se přiřadí když už nejsou další hodnoty, hodnota 1 když další ITEM je typu řetězce, a hodnota 2 když se jedná o číselný zyp.</p>
Ukončení procedury	<p>END PROC - se provede, podle očekávání, při udání konce jmenované procedury. Tím se umožní počítači aby nedošlo k přelétávání po příkazech mezi DEF PROC a END PROC když není daná procedura volaná.</p> <p>END PROC smaže všechny lokální proměnné a přiřadí jim normální hodnoty, pokud existují.</p>
DEFAULT	<p>Někdy se vyskytují okolnosti, obvykle při práci s procedurami, že budete potřebovat vytvořit proměnnou, pokud ještě neexistuje. K tomu Vám poslouží příkaz DEFAULT:</p> <p style="padding-left: 40px;">□ DEFAULT x=20</p> <p>přiřadí x=20 když x ještě neexistuje, jinak si x ponechá svoji hodnotu.</p>
Odložené adresy	<p>POP - odloží GOSUB/PROCEDURE/RETURN adresu. POP x stanoví číslo řádku proměnné x na odloženou adresu RETURNu.</p>

Ověřování
informací

Ve výpise DEF PROC proměnné včetně polí jsou spouštěné přes REF a zjišťují se jejich parametry zda jsou vedené v REF. Znamená to, že proměnná se přejmenuje na jméno definované v DEF PROC a zamezí se jakýmkoliv změnám mimo proceduru. Jinak proměnným zůstanou hodnoty určené ve výpise volání procedury. Je možné vypustit nějakou, nebo všechny hodnoty a přiřadit je z vnitřku procedury, použitím DEFAULT. S proměnnými nerefekovanými v REF, určí se pouze hodnota a proměnná je místa - LOCAL

LABEL

Proměnné je možné tvořit použitím LABELu na začátku řádků určených platným jménem proměnné. Jestliže část programu běží, například přes GO TO, nově vytvořenými proměnnými jsou přiřazené jména po každém LABELu v programu. Jejich hodnota je daná na řádku s takovýmto LABELem/označením/

```

□ 500 LABEL tea:PRINT"two sugars please"
    600 GO TO tea

```

Proměnné jsou obnovené když se stlačí CLEAR, nebo se EDITuje.

Systémové
proměnné

Jistá část bytů v počítači je odložena bokem pro speciální účely a nazývají se systémové proměnné. Můžete se po nich poohlédnout pomocí PEEKu a některé z nich můžou být pozměněné /POKE/ bez zásahu do systému, ale nejsou to ty samé jako proměnné používané v BASICu a počítač nezná jejich jména. Stovky systémových proměnných jsou upřesněné v technickém manuálu, nejužitečnější jsou vypsány v Appendixu originál manuálu.

SVAR

používá se na zadání adresy systémové proměnné, když následuje číslo. Můžete to využít např. při změnách kurzoru z "*" na "x" následovným způsobem

```
POKE SVAR 4,"x"
```

Zavěrem, pokud se těšíte na bližší informace o PEEKU a o POKE, můžete vstoupit do přezkoušení paměti v kapitole 10.

K A P I T O L A 5

BARVY A JAS

- použití PALETTE
- PEN a PAPER
- čtyři video MODES
- BRIGHT a FLASH
- INVERSE a OVER

V minulosti každý programátor toužící po široké paletě barev v jeho programech, byl nucený investovat do relativně drahých počítačů. SAM Coupé všechno změnil! Jsou v něm zabudované 4 video MODE, přičemž každý poskytuje na výběr 128 barev.

PALETTE

Představte si sebe jako malíře vybaveného vykrojenou deskou pro umělce, na které mícháte barvy z různých tub. Umělecký se nazývá deska paletou a SAM Vám poskytuje právě takovou paletu se všemi barvami, které můžete použít. Manufaktúra na výrobu barev vyrábí 128 barev v tubách a vaše PALETTE obsahuje 16 základních barev v označených miskách. Po každém zapnutí počítače a použití příkazy PALETTE máte k dispozici sadu 16 barev, označených 0 až 15. Viz str. 64 originál manuálu - přehled barev.

Dvě tuby černé barvy jsou proto že počítač je vyžaduje. Nyní si představte, že hledáte extra barvy pro vaše mistrovské dílo a chcete vidět celý rozsah barev na počítači. Zapněte SAMa a spusťte program:

```

10 FOR c=C TO 127 STEP 8
20 FOR p=0 TO 7
30 PALETTE p+8, p+c
40 PRINT p+c; TAB 3; PAPER p+8;
   STRINGS /29," " /
50 NEXT p
60 PRINT
70 PRINT "Press any key"
80 PAUSE
90 CLS
100 NEXT c
110 PALETTE

```

Ubezpečte se, že ve 40 řádce jste zapsali mezeru mezi úvozovkami a spusťte RUN.

Co vidíte na obrazovce nyní je škála barevných tub 8x tak bohatší. Na začátku jsou uvedena jejich kódová čísla. Vlastníte-li pouze černobílý televizor tak přes všechnu snahu nemůžeme vymyslet 128 názvů

BORDER

pro různé šedé odstíny. Pro kontrolu si můžete obdivovatelé gloriózních barevných displayů porovnat názvy se škálou barev.

Barevná škála viz. originál manuál, strana 66

Nezapomeňte na možnost měnit barvu na okraji obrazovky. Použijte k tomu příkaz BORDER, kterým můžete měnit barvy od 0 do 15. Např.:

BORDER 1

změní okraj obrazovky na tmavě modro v normální PALETTE.

PEN a PAPER

Umělec potřebuje na malování podklad a štětce. Vám k tomu poslouží PERO a PAPIER. Každé písmeno, nebo grafický obraz který chcete zobrazit na obrazovce může mít jiné pozadí, nebo barvu papíru / PAPER/ a vy si můžete vybrat i barevné pero. Můžete používat příkaz INK namísto PEN, pokud ho máte ve větší oblibě. Ve výpise se vždy zobrazí jako PEN. Zkuste změnit pozadí obrazovky příkazem:

PAPER 6:PRINT bumble bee"

kde se objeví černá barva na žluté. Nyní změňte barvu pera na červené:

PRINT PEN 2;"well rea"

malý experiment s kombinací barev. Pokud zvýhodňujete přechodný stav můžete použít příkaz OVER. Prostudujte Glossary!

Módy obraz-

Počítač je schopný změnit tvar video displee, aby se přispůsobil vašim požadavkům. Má zabudované 4 módy. Každý používá různý rozsah paměti a poskytuje jiné služby. Nový mod se přivolá:

MODE n

kde n je v rozsahu 1-4, přičemž je číslo vysvětlené dále.

Kapitola 5

Speertum
kompaktibil.
MODE

MODE 1 - poskytuje Vám 24 řádků, každý obsahuje 32 buněk^{*} pozůstávajících ze vzoru 8 x 8 bodů a mezer, jako 'A', které je zobrazené na straně 70 originál manuálu. Výběr barev pro pero a papír v každé buňce, je ze 16 barev vybraných ze 128 členné škály barev. Každá obrazovka bude pozůstávat z 768 písmenkových buněk. Tento mód používá počítač SPECTRUM a jeho uživatelé se cítí jako doma. Použijí MODE 1 na využití knihovny her. SAM ještě přezkouší jejich software pomocí překladače, který je připojený na kazetu 'FLASH'.

Alternativní
buňky

MODE 2 - poskytuje 192 řádků s 32 buňkami. Dává 6144 písmenkových buněk z 8 x 1 bodů. Výběr barev zůstává jako v MODE 1.

Uživatelský
mód

MODE 3 - umožňuje až 85 sloupců písmen, co je obzvláště užitečné na aplikaci zpracování slovních textů. Používají se pouze 4 barvy. Nevyužívá vůbec buňky, ale nechá Vás použít jednotlivé body nebo pixels. Dává Vám 192 řádků, na každém 512 pixels, dechromady 98304 bodů na hraní. Když poprvé spustíte MODE 3, první pozice PALETTE je definovaná - černá, modrá, červená a bílá. Když přepnete do jiného MÓDU, tak se poslední Vámi zvolené barvy uloží do nádobek a po opětovném uvedení do tohoto MÓDU se zobrazí. Samozřejmě, že když potřebujete změnu už vybraných barev, máte po ruce výběr ze škály 128.

Vysoký
výsledek

MODE 4 - je nejoblíbenější a je vybraný vždy když SAMA zapnete. Ve 192 řádcích stojí 256 pixels. Používá 16 barev ze škály 128 barev na 49252 bodů obrazovky. Umožňuje použití všech 128 barev současně, což Vám objasníme později. Protože všechno na obrazovce se může neustále měnit s PALETTE, tak se dají dosáhnout nádherné efekty. Demonstracní programy na kasetě FLASH to dokazují. Experimentujte s několika řádky v každém módu. Nyní vyzkoušejte dvě speciální úpravy v MODE 1 a MODE 2. PEN 16, nebo PAPER 16 znamená transparent a PEN 17, nebo PAPER 17 znamená kontrast.

Seuhrn jednotlivých módů

MODE 1 - 32 buněk x 24 řádků = 768 písmenkových buněk, každá se samostatným výběrem pera a papíru, 256 x 192 pixels, výběr 16 barev ze 128 barev

MODE 2 - 32 buněk x 192 řádků = 5444 buněk, každá se samostatným výběrem pera a papíru, 256 x 192 pixels, výběr 16 barev ze 128 barev

MODE 3 - 512 pixels x 192 řádků, každý pixel má samostatný výběr barvy, výběr 4 barev ze 128

MODE 4 - 256 pixels x 192 řádků, každý pixel má samostatný výběr barvy, výběr 16 barev obrazovky ze 128 barev

Vlastnosti

Tyto vlastnosti nejsou určeny pro pero a papír. Můžete vytvořit sérii speciálních efektů použitím dalších dvou vlastností nazvaných BRIGHT a FLASH

BRIGHT
(jas)

Když chcete vytvořit brilantnější barvu, tak příkaz BRIGHT 1 to provede, přičemž BRIGHT 0 vrátí obraz zpět do normálu. Existuje speciální úprava BRIGHT 8, nebo BRIGHT 16, což znamená 'transparent' v jiných slovech. Jas obrazovky se nemění když se udají jakákoliv písmena.

FLASH
(blesk)

Pracuje pouze v MODE 1 a v MODE 2. Tato částečná vlastnost způsobuje rytmické zjevování barvy. FLASH 1 zapíná a FLASH 0 vypíná. FLASH 8, nebo FLASH 16 zadává transparentní stav, který zůstává na každé pozici písmena. Každé písmeno na obrazovce se zobrazí jako obraz z barevných bodů vybraných PENem na pozadí vybraných PAPERem. Když použijete SHIFT a A tak se Vám zobrazí velké písmeno A jako je na obrázku originál manuálu strana 70 vlevo.

INV klíč

Podívejte se na pravý obrázek. Použitím INV erzu před SHIFT A se obrázek zjeví v inverzní podobě.

INVERSE

Existuje také tento příkaz, který má vliv na zobrazení písmen v barvě PAPER na pozadí barvy PEN. Toto se zadává příkazem INVERSE 1 a vypíná INVERSE 0.

OVER

Specifikace sloužící ke kontrole vzoru grafických bodů na obrazovce a způsobuje opětovné vytisknutí bedů, řádků písmen. OVER 1 spouští tento účinek a OVER 0 končí proces. V MODE 1 a 2 písmeno je opakovaně tisknuté přes sebe v rychlé sekvenci písmeno-bílá-písmeno. V MODE 3 a 4 sekvence probíhá následovně: písmeno-barva-písmeno. Běžně se užívá černá barva. Jiné efekty umožňují příkazy DRAW, CIRCLE, PLOT a PUT, které jsou vysvětlené v kapitole 6.

POINT x,y

Slouží k nalezení polohy jakéhokoliv bodu na obrazovce. Vysvětluje se v GLOSSÁRI.

Závěrem uvádíme souhrn způsobů použití PALETTE:

PALETTE mění všechny změny, barev, které můžete vybrat a měnit.

PALETTE pozice, barva - udává pozici mísky na barvy na vaší PALETTE každé požadované barvě. Existuje 16 rozdílných pozic (0-15) a 128 rozdílných barev (0 - 127)

PALETTE 14, 43: BORDER 14

napustí růžovou barvu do mísky s číslem 14 a zobrazí na obrazovce.

PALETTE pozice, barva a, barva b - umožňuje výběr dvou barev, které se mohou zobrazit na stejném místě palety a mění se třikrát za sekundu. Tímto můžete vytvořit speciální blyskavé efekty.

PALETTE 7, 8, 36

PALETTE pozice, barva, LINE y - příkaz, který udává přechodný stav pozice palety a pozice barvy, když y-ová osa grafického souřadnicového systému dosáhne určený řádek y. Nové zadání je možné ještě před dosažením spodního okraje obrazovky. Takto se dá zobrazit více jak 16 barev na obrazovce.

PALETTE pozice, barva a, barva y, LINE y - příkaz, který umožňuje vybrat přechodnou alternativu barvy na blikání (FLASH) na řádku se souřadnicemi y s maximem 127 změn na obrazovce.

PALETTE pozice LINE y - ruší všechny změny, které by způsobilo PALETTE pozice p na řádku y.

KAPITOLA 6

Texty a grafika

- o grafických příkazech PLOT, DRAW, CIRCLE, FILL
- FATPIX, a grafické škály
- SCREENS / obrazovky /
- WINDOW, GRAB a PUT
- animace
- INKEY S a GET
- písmena
- CSIZE, TAB, AT
- BLOCK grafické metody
- uživatelem definované grafické metody

Tato kapitola obeznamuje s použitím výhod Vašeho počítače k práci s textem a grafickými obrázky. Staré psací stroje píšou předurčené písmena ale modernější stroje umožňují měnit mezi různými tvary písmen. Ale Váš SAM Vám umožňuje přiřadit různé symboly a použít je na vytištění komplexních obrázků. Poskytuje celou hromadu příkazů na vytvoření obrysů, vzorů a uměleckých obrazů. Na kazetu FLASH jsme zařadili několik jednoduchých, ale vysoce efektivních grafických souborů. Bude vítané když použijete ESC tlačítko, nebo BREAK na přezkoušení jejich výpisu.

Grafické příkazy

PLOT x,y - specifikace na určení počátečního bodu na obrazovce, používá se vždy na začátku grafického zobrazování a určuje 1 pixel. První číslo v příkaze určuje souřadnici x a druhé číslo souřadnici y. Takže:

□ PLOT 128,86

zaměří bod v blízkosti středu obrazovky v MODE 1, 2 a nebo 4 a nebo nalevo od středu v MODE 3. Můžete si vybrat barvu před použitím PLOT-u. Tak zkuste zadat počítači červenou v MODE 4:

□ 10 PLOT PEN 10; INT / RND x 255 /, INT / RND x 173 /;
GO TO 10

DRAW x,y

Body jsou velmi drobné, ale samotné pixely jsou lehké spojitelné. Provádí se příkazem DRAW. Hodnota x je číslo pixlu, který se má pohybovat napravo a hodnota y, která se má pohybovat směrem nahoru. Pro posun opačným směrem stačí, když zadáte záporné hodnoty x a y.

DRAW x,y,z - pracuje podobně s přidáním osy z, přímka se stává křivkou.

DRAW TO x,y - příkazuje spojit nynější bod s bodem o souřadnicích x,y. Můžete použít PAPER, PEN, OVER atd. po DRAW TO / a po DRAW, PLOT a circle /.

Následující příklad spojuje náhodné řádky, které nevybíhají z obrazovky:

```

□ 10 FOR h=1 TO 100
  20 DRAW TO RND x 255,RND x 173
  30 NEXT h

```

DRAW TO x,y,z - příkaz podobný jako předcházející, při spojování osy z vytváří křivku. Zkuste:

```

□ 10 PLOT 90,86
  20 DRAW TO 166,86,2

```

Zanechte úsměv na obrazovce a vytvořte okolo něho čáru.

CIRCLE x,y,poloměr - nakreslí kruh na obrazovce, pokud je zadán střed /souřadnice x,y / a poloměr kruhu. Nyní zakroužkujte úsměv s:

```

□ CIRCLE 128,100,50

```

Jestli že širší kruh zachází za horní, nebo spodní okraj obrazovky, objeví se na opačném konci obrazovky.

FILL

FILL barva,x,y - slouží na vybarvení ohraničeného prostoru jednou barvou. Barvu musíte specifikovat v jejím kódovém čísle. Plocha musí být zaměřena jakýmkoliv bodem, i hraničním. Ponecháme směřící se kruh na obrazovce:

```

FILL PEN 3,128,86:FILL PEN 6,1,1

```

vybarví prostor okolo zadaného bodu. Speciálních efektů můžeme dosáhnout pomocí vyplnění se vzorky uloženými v řetězcích:

```

FILL USING a$,x,y

```

kde se použije na vyplnění vzor specifikovaný v a\$ řetězci. Nejběžnější způsob na uložení vzorku je pomocí GRAB-u z obrazovky, což je objasněné později.

FATPIX - v grafickém MODE 3, pixely jsou tenké a mají pouze poloviční šířku než v ostatních modech, protože je jich dvakrát tolik. Použitím FATPIX 1 příkazu se dá zdvojnásobit šířka pixelů v MODE 3. Vyrovná se tloušťka vertikálních a horizontálních řádků a kruhy jsou skutečně zaoblené do kruhu. Na zavedení do původního

System
grafického
zobrazování
souřadnic
os

stavu se používá příkaz FATPIX 0.

Nyní si vyzkoušíme manipulaci se souřadnicemi.

Škála a počáteční body používané příkazy PLOT

DRAW, DRAW TO, CIRCLE, GET a FILL - mohou se měnit použitím speciálních proměnných.

XOS - kontroluje x-ovou souřadnici

YOS - kontroluje y-ovou souřadnici

XRG - kontroluje rozsah x-ové osy

YRG - kontroluje rozsah y-ové osy

Hodnota souřadnice se rovná 0, pokud se nepoužije v příkazu LET:

□ LET XOS=128, YOS=69

příkaz se přesouvá po částečný bod všech souřadnic pro grafické zobrazení do středu obrazovky. Podobně při výměně XRG a YRG mění se škála ve které pracují PLOT a DRAW. V kombinaci s příkazem WINDOW můžete vytvořit dokonce speciální efekty.

Normální pozice 0,0 souřadnic x,y se nachází v levé části spodního okraje, právě nad sekci spodní části bodů a vrchol má souřadnice 255,173 a nachází se v pravém horním rohu. V MODE 3 se užívají tenké pixely a x osa se rozšiřuje, vrchol napravo se souřadnicemi 511,173. SAM umožňuje takéž využít obouh řádků určených normálně na editování, na použití PLOT:

□ LET XOS =18 - když písmena jsou 9 pixelů vysoké

□ LET XOS =16 - když písmena jsou 8 pixelů vysoké

Toto určí koordináty 0,0 jako nejnižší okraj a 0,191 vrchol vlevo.

BLITZ

BLITZ ať se používá na provedení řetězce grafických příkazů velmi rychle. I přes to že pracuje ve všech módech, svláší efektní je v MODE 4. BLITZ zodpovídá speciální grafickým škálovacím proměnným XOS a YOS, které umožňují blikání grafického tvaru kdekoliv na obrazovce. Velikost obrysu se může měnit pomocí speciálních grafických proměnných XRG a YRG.

Použití
obrazovek

Po zapnutí SAMa existuje jen jedna přístupná obrazovka zvaná SCREEN 1. Ale vy máte přístup až k celkovým 16 obrazovkám v každém případě.

SCREEN číslo - vybírá částečnou obrazovku na použití Basicovských příkazů. Číslo musí odpovídat od 1 do 16 a musí být nejprve otevřena předpisem:

□ OPEN SCREEN 2

Tímto se zprostředkuje druhá obrazovka a můžete použít příkazy PRINT, PLOT a DRAW. Zatím co první obrazovka zůstává nedotknutelná až do dalšího přivolání.

OPEN SCREEN číslo, mode - rezervuje paměť pro obrazovku, je-li specifikovaná číslem a číslem modu. Číslo obrazovky spadá do rozsahu 2 až 16, MODE 1 až 4 si můžete vybrat. Každá obrazovka obdrží vlastní MODE, PALETTE detaily, PRINT pozici atd.

OPEN SCREEN n,m,0 - provede to samé jako OPEN SCREEN n,m

OPEN SCREEN n,m,1 - otevře běžnou obrazovku, takže když v paměti počítače je vloženo několik programů, tak mohou všechny využít obrazovku pokud jsou spuštěné.

CLOSE SCREEN číslo - uzavírá očíslovanou paměť dané obrazovky. Když uzavřete neexistující paměť, tak se nic neděje.

DISPLAY číslo - mění obrazovku na zobrazení, ale ne obrazovku používající BASIC. Jedna obrazovka se může neviditelně budovat, pokud druhá právě zobrazuje.

DISPLAY, nebo DISPLAY 0 - ukazují hlavní obrazovku.

DISPLAY 1 až DISPLAY 16 - ukazují požadovanou obrazovku.

CLS - používá se na vymazání obrazovky a smazává klíč displeje z obrazovky. CLS, nebo CLS 0 čistí celou obrazovku, CLS 1 čistí pouze plochu zpracovanou přes WINDOW.

Existuje však omnoho důraznější příkaz na vymazávání.

CLS ≠ neprovede jenom smazání obrazovky, ale provede

Přetáčení a
rolování

1 příkazy OVER 0:INVERSE 0:FLASH 0:PAPER 0:PEN 7:
BORDER 0:PALETTE souběžně.

Z času na čas obrazovka neposkytuje dostatek prostoru na zobrazení všech textů a tabulek. Když se toto stane, tak se zobrazí scroll?, a očekává se odpověď. Následuje svinutí směrem nahoru. Můžete si však upravit obrazovku podle potřeby, pohybem celé obrazovky ve vertikálním či horizontálním směru, nebo části obrazovky příkazem WINDOW.

SCROLL_CLEAR - ruší scroll? když je obrazovka zaplněná a automaticky zapíná SCROLL

SCROLL_RESTORE - působí opačně

ROLL směr, číslo pixelů - pohybuje obraz na obrazovce a otáčí dokola. Pohyb je kontrolovatelný směrem:

1 - vlevo, 2 - nahoru, 3 - vpravo, 4 - dolů, následuje číslo pixelů, které se mají pohybovat / 1, 2, 3, 4, 6, 8, 10/

ROLL směr, pixelů, x, y, šířka, délka - roluje obrazovka určená x, y souřadnicemi, následuje číslo, šířka pixelů a délka pixelů. X je zaokrouhlená na celou souřadnici a šířka zaokrouhlená na párny počet pixelů.

Pracujte touto technikou pouze v MODE 3 a MODE 4

SCROLL směr, číslo pixelů - pracuje tím samým způsobem jako ROLL mimo efektu obtáčení. Prázdné pozadí nahrazuje zmaznuté data.

SCROLL směr, pixelů, x, y, šířka, délka - pracuje jako ROLL bez efektu obtáčení

WINDOWS

Okno představuje část obrazovky v celé obrazovce jako na některých video rekordérech. Tato plocha může být námi manipulovatelná nezávisle od ostatní obrazovky.

WINDOW vlevo, vpravo, nahore, dole - určuje textové okno. Hodnoty jsou určované: levým sloupcem / 0, nebo více / pravým sloupcem / obyčejně 31, ale může být i 84 v MODE 3 s pixelovým písmem na šířku horním řádkem, spodním řádkem

Pokud zadáte přemrštěné hodnoty, objeví se Invalid WINDOW

GRAB a¹ x,y,šířka,délka - zabírá plochu obrazovky pro řetězec a, který může být vložen / PUT / někde jinde. Aby mohlo dojít k této rezervaci plochy pomocí GRAB-u tak se definuje x,y jako vrchol vlevo nahoře, následuje jeho šířka a jeho délka v pixelech. GRAB a PUT pracují pouze v MODE 3 a 4. Souřadnice x pro tlusté pixely se zaokrouhluje směrem nahoru k celému číslu. Pro MODE 3 se tenké pixely v souřadnicových hodnotách zaokrouhlují na čtyřnásobky.

PUT x,y,b¹ - Zabírá rezervované místo na obrazovce. Hodnota x se zaokrouhluje na celou. PUT se ozve následnými příkazy:

INVERSE jako v INVERSE 1: PUT x,y,b¹ - a nebo jako v PUT INVERSE 1;x,y,ANY ¹

OVER 0 vytiskne všechno co je už natištěné

OVER 1 udá XOR

OVER 2 udá OR

OVER 3 udá AND

ale PEN a PAPER se nezmění, takže jste závislí na barvě udané v příkaze GRAB

maskování

PUT x₁y₁b₁,x₂y₂b₂ - umožňuje nám použití / musí být stejně dlouhý jako první řetězec / druhého řetězce na grafické maskování, určuje které pixely bude používat první řetězec. Umožňuje umístit celkový obrys na pozadí, jako by byl odsekнутý bez hranic.

Maskování neovlivňuje OVER, ale INVERSE pracuje výborně.

Existuje další způsob na vytvoření masky:

- 1 - vytvořte původní obrys na obrazovce
- 2 - DRAW čtverec dokola, o vrstvu větší než plocha, která půjde do GRABu
- 3 - FILL čtverec s PEN 0 na vytvoření hranic nulami
- 4 - GRAB celou plochu
- 5 - FILL hraniční plochu s PEN 15
- 6 - PUT zadanou plochu přes sebe s OVER 1
- 7 - znovu uložte do GRABu
- 8 - PUT zpět použitím OVER 0, INVERSE 1

Pohyb a animace

9 - naposledy uložte do GRABu, aby jste dostali využitelnou masku

Pohyby jsou ve skutečnosti rychlé sekvence stojících obrazů, které se navzájem pouze jemně odlišují. Počítačová animace pracuje podobně na styl oklamání zraku. Nejjednodušší způsob animace představuje měnění písmen. Uvedeme příklad:

```
10 PRINT "/":PAUSE 3:CLS:
   PRINT "-":PAUSE 3:CLS:
   PRINT "\":PAUSE 3:CLS:
   GO TO 10
```

Na zobrazení pohybu potřebujeme, aby se naše grafické výtvary přesouvali po obrazovce. Na toto musíme obrázek vytisknout na požadovaném místě, potom ho vymazat a na to znovu vytisknout o kousek dále. Takto pokračuje proces. Tak též potřebujeme kontrolovat jejich rychlost a směr pohybu.

Následující program zobrazí pohyblivou housenku:

```
10 PEN 15: PAPER 0:BORDER 0:CLS
20 LET tail=3:DIM aa/tail/:
   DIM bb/tail/
30 LET x=1:LET a=9: LET b=15
40 PRINT AT a,b; PEN 12;" O "
50 LET c=CODE/INKEY$:
   IF c < 193 OR c > 201 THEN GO TO 50
60 IF c=197 THEN CLS:LET x=1
70 LET aa /x/ = a: LET bb/x/=b
80 LET a=a+ / c > 192 AND c < 196 AND
   a < 18/ - /c > 198 AND c < 202 AND a > 1/
90 LET b=b- //c=193 OR c=196 OR
   c=199/AND b > 1/+ //c=195 OR
   c=198 OR c=201/AND b < 30)
100 IF x < tail THEN LET x=x+1:
   GO TO 40
110 IF aa /x/<> a OR bb/x/<> b THEN
   GO TO 130
120 GO TO 40
130 LET aa=aa /1/: LET bb=bb/1/: LET z=1
```

```

140 IF z= tail THEN PRINT AT aa,bb;
    " " : GO TO 40
150 LET aa /z/ = aa /z+1/:
    LET bb /z/= bb/ z+1/: LET z=z+1:
    GO TO 140

```

Před spuštěním zkuste porozumět jak řádek 20 určuje délku housenky a jak řádek 40 určuje její barvu a obrys. Řádek 30 určuje souřadnice proměnným. Řádky 50 a 60 jsou připravené na rozeznání určených klíčových slov. Řádky 80 a 90 zadávají 8mi cestný směr kontroly když určité [F] klíče jsou stlačeny a jejich kódy rozpoznány. [F4] směřuje, [F7] severozápad, [F8] sever a tak dále ve směru hodinových ručiček.

Řádek 60 zadává pro [F5] zvláštní funkci.

Vyzkoušejte pochodování housenky stáčením rozličných klíčů. Zkuste změnit její délku změnou proměnné tail v řádku 20. Čím je delší, tím je unavenější, takže nezačázejte za LET tail=10. Čím je kratší, tím je rychlejší, závisí od PRINTu. Zkuste přiřadit tail=1. Měňte barvu a obrys v řádku 40. Nyní se podívejte na řádek 140 a pozorujte jak mezera mezi úvozovkami v příkazu PRINT způsobuje vymazání posledního članku housenky, při jejím pohybu. Na závěr změňte řádek 140 tak, že namísto mezery se zadá stop, a sledujte jak malý červík zanechává za sebou sliskou stopu.

INKEY\$

Všimněte si v řádku 50 minulého příkladu jak funkce se používá na zjištění stlačeného tlačítka, když nějaké bylo stlačeno.

GET

Způsobí přijetí příkazu bez použití RETURN. Je podobný jako INKEY\$, ale GET čeká na stlačení tlačítka před pokračováním programu. Při použití řetězcové proměnné slouží GET jako psací stroj:

```
10 DO:GET a$:PRINT a$;:LOOP
```

tiskne tlačítka jak se jich dotýkáte. Můžete psát velkými i malými pomocí SHIFT. Když se GET používá s číselnou proměnnou jako GET x, potom proměnná se

RECORD

revná 1 pokud 1 stlačíte, až po 9 pokud stlačíte 9 A nebo a nabývá hodnotu 10, B nebo b 11 atd. GET je velmi užitečný v programech 'menu-driven'

RECORD TO a₅ - umožňuje nahrát grafické příkazy nebo řetězce. Může být tak mnohem rychleji provedené, než když je spuštěno přes BLITZ. Když je spuštěn RECORD nahrají se následovně příkazy:

CIRCLE, CLS, DRAW, DRAW TO, OVER, PAUSE, PEN, PLOT, ale ne PLOT PEN, nebo jiné přechodné výrazy.

RECORD STOP - vypíná nahrávání grafiky. Oba příkazy RECORD TO a RECORD STOP pracují ve všech MODE.

Úprava textů

Druhá část této kapitoly ukazuje jak můžete nejlépe využít SAMovy možnosti na práci s textem.

Předurčení klíčů:

Každý klíč byl předurčený na grafický, nebo textový symbol. V Appendixu originál manuálu je nakreslené, co se stane po použití klíče spolu s následovnými alternativami:

NORMAL - jednoduše stlačení vlastní klávesy.

[SHIFT] - značí to samé jako **[CAPS SHIFT]** tlačítko na psacím stroji, nebo slovním procesoru na tištění velkých písmen

[SYMBOL] - vyúsťuje do série různých písmen, grafických bloků, nebo kódů

[CTRL] - představuje kontrolní stav, provádí některé specializované kódy přímo z klávesnice, jako i alternativní grafické bloky

Zkušený programátor najde v Appendixu speciální číselné kódy umístěné pro každý klíč na identifikaci pozice.

Příkaz **KEY** pozice, x - je vysvětlený v souhrnu.

Kapitola 6

Seuber
znaků

Máte přístup k 256 možným znakům, každý jeden má kód od 0 do 255. Všechny kódy pod hodnotu 32 se používají na speciální účely, jako kontrolu inverze na videu. 32 až 168 byly předdefinované SAMovi. Když je chcete prohlédnout, tak napište:

```
□ 10 FOR a=32 TO 168:PRINT CHR$ a;:NEXT a
```

Můžete vyzkoušet znovu, pokud víte jak zapnout a vypnout speciální BLOCKS grafiky. Když chcete objevit jednotlivé kódy znaků, vyzkoušejte:

```
□ 20 FOR a=32 TO 168:PRINT CHR$ a;" ";a:NEXT a
```

Další technikou vhodnou pro zkušených programátory je vhodné definovat krátké kódy, které se mohou automaticky uplatnit po zadání z klávesnice. Příkaz DEF KEYCODE se nachází v souhrnu. Pokud chcete SAMA použít na napsání románu, nákupního seznamu, nebo složeni katalogu využitím jako světelného psacího stroje jsou Vám k dispozici vícero výhodné možnosti na zobrazení textu.

Rozehněte se pro mód, který nejvíc vyhovuje pro částečný dokument, který chcete psát. Zkoušku písma můžete provést:

```
□ CSIZE šířka, výška
```

Velikost
písma

Čímž nastavíte počítač na změnu velikosti písmen sadaných rozměrů v pixlech. V módu 3 písmena dosahuje do šířky 6, nebo 8 pixelů a výšku můžete zvolit v každém módu od tenkých 6 pixelů až po široké 32 pixelů. Používá se na nadpisové řádky, zobrazení pro začátečníky, nebo speciální účely. Každá výška písmena nad 16 představuje dvojnásobnou výšku písma.

TABulování

Na případné seřazení textu do jednotlivých paragrafů, nebo pravidelných sloupců, slov, nebo čísel se používá tabulátor. Ten se používá jako na běžných psacích strojích, nebo starých procesorech. Nastaví se sloupec pozicí od levého okraje použitím specifikace PRINT

```
□ PRINT TAB 2;"sam"
```

vyobrazí sam po dvou mezerách v MODE 1, zatím co

□ PRINT TAB82;"!"

je možné použít v MODE 3 s písmeny širokými 6 pixly, protože TAB 84 je sloupec na pravé straně. Nezapomeňte, že použitím čárky ve výpisu se zapříčiní to, že specifikace PRINT která následuje, automaticky tabuluje 16 sloupců. Tento efekt závisí od použitého modu.

AT

řídí počítači, kde má tisknout kdekoliv na obrazovce, když je potřebné zadat číslo řádku a číslo sloupce. Takže v MODE 4:

□ PRINT AT 10,16;"!"

vytiskne vykřičník v blízkosti středu obrazovky, na 10. řádku a v 16-tém sloupci.

Pokud jste nespokojený s výběrem znaků v paměti počítače, tak je několik způsobů na jejich změnu podle Vašeho přání. Můžete si je vytvořit i Vy sám. Tyto speciální pozice jsou uloženy v sadě pro uživatelské definování. Nejdříve si prohlédněte skupinu 16 znaků mezaiky, pod názvem bloková grafika.

Tyto se vyvolávají stlačením tlačítek 1 až 8 najednou se symbolem [SYMB] a nebo [CNTRL] tlačítka, jak je zobrazeno. Bloková grafika je vytvořena ze vtordů z pixlů tím jistým způsobem, jako v kapitole 5. Podle požadavku je možné zapnout a vypnout.

BLOCKS

BLOCKS 0 vypíná předdefinovanou blokovou grafiku umístěnou od CHR\$ 128 po 143. Uvolňuje prostor na Vaše opětovné definování jako uživatelem definovaná grafika - UDG zkráceně. Když stlačíte BLOCKS 0 a prohlédnete si pomocí PRINTu jejich CHR\$, ujistíte, že už jsou předznačeny jako cizí znaky.

BLOCKS 1 - spustí původní blokovou grafiku, ale vaše redefinované UDG jsou bezpečně uloženy na pozdější použití. Stačí použít BLOCKS 0.

Karta
blokové
grafiky

	[SYMB] [1]		[CNTRL] [1]
	[SYMB] [2]		[CNTRL] [2]
	[SYMB] [3]		[CNTRL] [3]
	[SYMB] [4]		[CNTRL] [4]
	[SYMB] [5]		[CNTRL] [5]
	[SYMB] [6]		[CNTRL] [6]
	[SYMB] [7]		[CNTRL] [7]
	[SYMB] [8]		[CNTRL] [8]

UDG

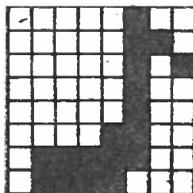
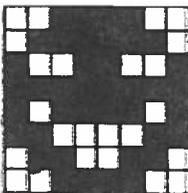
CHR\$ 144 až 168 jsou též císí znaky UDG,
CHR\$ 169 až 255 jsou přístupné pro zručné programá-
tory.

Máte možnost najít každou adresu přes UDG:

□ UDG "A"

udá adresu vzoru písmena A v hlavní sadě písmen, a
když byl sadaný vrchní řad písmen, tak UDG CHR\$ 170
atd. zpracuje.

Na uložení vlastního znaku budete muset vyřezat
počítač jak prezentovat jednotlivé buňky v síti pixelů
které tvoří písmena. Pokud rozumíte postupu binární
(dvojkové) soustavy, tak jste doma. Pokud nerezumíte, tak
nebudete mít s tím problémy taktéž, protože se s nimi
pracuje velmi jednoduše. Vyplněný čtverec nastavený
barvou PEN má hodnotu 1, prázdné pole má barvu pe-
sadi PAPER a hodnotu 0. Zobrazené jsou následující
obrázky.



Použili jsme jednoducho síť 8x8 pixelů v MODE 1. Ale buňka písmena se bude měnit v závislosti od MODE a CSIZE .Jak víte váš UDG má ukázat,použití na překrytí existující písmeno a vložte nový vzor v BIN-árním kodu.Obrázek směřící se tváře jak jsme použili na obrázku se zobrazí jako BIN,příčemž čísla znamenají PEN a PAPER:

```
BIN 00111100
BIN 01111110
BIN 10011001
BIN 11111111
BIN 10111101
BIN 11000011
BIN 01100110
BIN 00111100
```

Vyzkoušejte zda dokážete přepsat notu do tohoto tvaru. Tento číselný vzor je možné vložit do paměti a správná adresa bude UDG"hledaný znak" pro orvní bude /skupina 8 znak/ následuje UDG "hledaný znak"+1 pro následující skupinu,až poUDG "hledaný znak"+7 v našem příkladu. Podívejte se na kapitolu 10,PEEK a POKE
Nesapomeňte,že můžete měnit PEN a PAPER pro každé UDG. Může dojít při záměnách 0 a 1 ke smíšení dvou barev, což se dá využít.

S potěšením přikládáme celý rozsah výběru proněkolic rozdílných typů 'Fontu'/jako i UDG soubor, který můžete běžně používat/ na FLASH 'kasetu.Umožní vám použít celou řadu vymožeností ve vašich programech.V příručce ke 'FLASH 'kasetě se dozvíte o všech schopnostech umění počítače SAM Coupé.

K A P I T O L A 7

TVORBA HUDBY

- BEEP
- nastavení toniny
- frekvence a zkratky
- hlasitost a stereo
- sampler SAM
- zázraky MIDI

SAM Coupé je exelentní hudební syntetizátor. Když použijete zásuvku pro UHF TV, tak zvuky se reprodukují přez váš reproduktor v televizoru./mono/Stereo signál se ozve použitím cesty SCART na zadní straně počítače nebo můžete využívat napojený hi-fi systém na světelné pero / stereo zásuvku/ a použít vlastní sluchátka.

BEEP d,p

Počítač dostává instrukce vyloudit zvuk zadáním příkazu BEEP,následuje číslo /d/ délka v sekundách a číslo /p/ výška hudební tóniny.Když máte spojení na hudbu tak napište následovný proces s náznakem Beethovena:

BEEP 0.5,0:BEEP 0.5,0:BEEP 0.5,0:BEEP 1.5,-4

Délka a výška prvních třech not byla 0.5 /půl sekundy/ a 0,co představuje střední C'na klavíru.Vyšší tóny se určují číslem vyšším než je p=0 po 71 /tóniny, které ocení váš pes/ a hluboké tóny se určují zápornou hodnotou do -60 /což může zapříčinit malé zemětřesení při dostatečném zesílení/

BEEP může být krátký podle vašeho přání,ale maximální délka je 16 sekund.Vyzkoušejte si:

FOR p=60 TO 71:BEEP .3,p:NEXT p

Obrázek na straně 95 originál manuálu ukazuje všechny základní tóny, které zodpovídají bílím a černým klávesám na velkém klavíru spolu s jejich hudebním označením. Více složkové soubory pro speciální efekty určitých typů orientální hudby se dají vytvořit kratšími intervaly:

BEEP 1,0.5

co zahraje 1/4 noty nad středním C

frekvence

Počítač má 8 oktávovou vstupní frekvenci v rozsahu od 31 Hz po 7.81 kHz.Pro speciální tóniny se využívá frekvence v cyklu za sekundu.Frekvence se dá vypočítat pomocí vzorce,kde p=note /výšce/

PRINT 440 x 2 P-9 /12

Skratky

Nezá potřebov přepisovat vaše hudební vtvory vždy, když potřebujete maladit váš počítač najiný hudební

nástroj, nebo změnit klávesu. Trvalé změny klávesy a časování se dosahují zadáním proměnných pro délku a výšku před psaním hudby. Nyní vyzkoušejte následovný příkaz:

```
10 LET d=1
20 LET p=0
30 BEEP d,p+4:BEEP d,p+4:BEEP d,p+5:BEEP d,p+7:
   BEEP d,p+7:BEEP d,p+5:BEEP d,p+4:BEEP d,p+2:
   BEEP d,p:BEEP d,p:BEEP d,p+2:BEEP d,p+4:
   BEEP d,p+4:BEEP d,p+2:BEEP d,p+2
```

Jestliže p má hodnotu 0, dílko Beethovena se zahraje v klíči hlavní C stupnice. Ale přiřazením p=7, například se přesune celé do stupnice G. Zlou zvěstí je to, že Óda na radost zní spíše jako pohřební pochod, je příliš pomalá. Dobrou zvěstí je to, že tempo se dá lehko změnit přiřazením LET d=0.5

Můžete použít též d na zkrácení psaní speciálních hodnot not, nezávisle na tom jaké tempo vyberete. Takže když d představuje celou notu zabírající 4 doby, potom:

d=	4 doby	d/4=	jedna doba
d/2=	2 doby	d/8=	půl doby atd.

Pauzy nebo mezery ve vašich kompozicích vytvoříte příkazem PAUSE podobně jako při grafice.

Jsou zabudované samostatné generátory v počtu 6. Každý frekvenční generátor může vyloudit 256 tónů v každé oktávě, takže různé hudební nástroje a speciální efekty se dají napodobnit. Na demonstrační kazetě je výborný přehled audio možností počítače. V technickém manuálu jsou udané všechny detaily na adresování a kontrolu hudebního čipu, jako i psaní datových bytů na MIDI porty.

**Zázraky
MIDI**

MIDI je zkratka hudebního přístrojového digitálního interfejsu. Může kontrolovat celý orchestr přes přísun informací, přes zásuvku pro MIDI v zadní části počítače. Protože MIDI představuje světový standart, tisíce výrobců vyrábí různé modely, které mohou být pak spojené do jednoho orchestru využitím počítače. SAM zpracovává tyto vstupní údaje na mezinárodně standartní hodnotě 31,25K baud / 31,250 bitů za sec/ přes vstup MIDI-IN a vysílá přes výstup MIDI-OUT.

Počítač neslouží pouze k tvorbě hudby, ale díky MIDI ho můžete využít na opakování, synchronizaci, učení, vyučování a kontrolu hudebních zázraků klávesových nástrojů, syntetizátorů, samplerů, procesorů speciálních efektů, audio-video studií, hudebních hodin atd. A samozřejmě, že SAM je schopný napodobit každé takové zařízení.

Následující kapitola vás seznámí s použitím hudebního čipu na vytvoření efektů a použití hlasitosti a sterea.

K A P I T O L A 8

ZVUKOVÉ EFEKTY

- zvukové efekty v programech
- ZAP, POW, ZOOM, BOOM
- stereo a multi kanál

Sam zvučně zní speciálními zvuky, které můžou program vhodně zpřehlednit podle vašeho výběru. Můžete vytvořit atmosféru, přidat realnost, změkčit, šokovat a přinést komické pozadí.

Zvukové efekty často využívají smíšené frekvence. Jejich manipulace není lehká srozumitelná. Pro zjednodušení jsme jich několik připravili tak, že jich můžete použít kdykoliv v programech jednoduše zavoláním jejich jména jako by to byly jednoduše běžné příkazy.

Před sadou zvuky:

ZAP - zní přesně jako zvuk útočného laserového paprsku.

POW - vyrábí napodobeninu nárazu

ZOOM - používá se na napodobení rychlého pohybu od stíhačky až po skateboard

BOOM - zvuk při dopadu bomb, poschodových domů a poknutí hltačů

Pokročilí programátoři můžou použít příkaz SOUND na zaslání speciálních kódů do zvukového čipu. Uvádá cestu datových bitů d do jednotlivých zvukových registrů čipu r. Může se sadovat až 127 párů těchto čísel. Např:

SOUND r,d;r,d atd

Plné znění najdete v technickém manuálu, ale pro začátečníky v BASICu je vítaný malý podvod, okopírování následujícího programu.

Nejprve navedeme scénu. Všude je tma. Stará parní lokomotiva se pohybuje ze západu směrem dolů. Najednou turistický létající talíř se přehnal nad hlavou na cestě na snídani k ALFA CENTAURI. Ale posádka zapoměla vysvědnout jednoho mimozemšťana z těch mladších. On se snaží zachytit na radar ale chybuje. Šťastnou náhodou je poblísku veřejný telefon. Malíčký vetřelec volá domů. První kukačka na jaře se objevila.

```

10 REM effects
20 GOSUB 3000
30 REM steam locomotive
40 LET L=7:GOSUB 2000
50 LET p=20:GOSUB 1000
100 REM fling saucer
110 LET L=3:GOSUB 2000
120 LET p=20:GOSUB 1000
200 REM sonic scooter
210 SOUND 17,4:SOUND 16,64

```

```
220 LET p=5:GOSUB 1000:PAUSE 200
300 REM telephone
310 GOSUB 3000
320 LET L=7:GOSUB 2000
330 FOR n=0 TO 3
340 SOUND 2,255:PAUSE 18
350 SOUND 2,0:PAUSE 8
360 SOUND 2,255:PAUSE 18
370 SOUND 2,0:PAUSE 80
380 NEXT n
400 REM cuckoo
410 BEEP 0.2,19:BEEP 0.4,15
420 STOP

1000 REM stereo Left to Right
1010 LET a=0:LET b=0
1020 LET L=1:LET R=0:GOSUB 1500
1030 LET L=0:LET R=16:GOSUB 1500
1040 LET L=-1:LET R=0:GOSUB 1500
1050 LET L=0:LET R=-16:GOSUB 1500
1060 RETURN

1500 REM output subroutine
1510 SOUND 2,/a+b/:PAUSE p
1520 FOR n=1 TO 15
1530 LET a=a+L:LET b=b+R
1540 SOUND 2,/a+b/
1550 PAUSE p:NEXT n
1560 RETURN

2000 REM output sound data
2010 DATA 9,64,16,17,0,24,138,28,1,21,4,20,0
2020 DATA 9,255,17,1,20,4
2030 DATA 24,142,20,4,10,111,17,6,9,128,16,32,28,1
2040 FOR n=1 TO L
2050 READ a,d:SOUND a,d:NEXT n
2060 RETURN

3000 REM clear the sound
3010 FOR n=0 TO 31:SOUND n,0:NEXT n
3020 RETURN
```

Experti mezi vámi zjistí, že stereo a hlasitost jsou jedno a to samé. Hodnota 0 znamená ticho a až

16 samaná maximální hodnotu. Změnou úrovně hlasitosti přes dva kanály se tvoří stereo efekt. Poohlédněte se, zda umíte určit místo instrukce SOUND, která kontroluje jednotlivé efekty a změňte jejich kódy.

KAPITOLA 8

ROZŠÍŘENÍ SYSTÉMU

- Eurokonektor
- použití tiskárny
- LPRINT a LLIST
- DUMP
- zásuvka pro SCART
- joysticky
- myši
- světelné pera

	<p>Jednou z hlavních filosofických črt při sestavování počítače bylo umožnění dalšího rozvoje, rozšíření a lepší použití vašeho systému. Nové výrobky, které se objevují na trzích bude možné napojit na váš počítač. Takže SAM nepropadne, ale očekává novinky budoucnosti. Poslední dvě kapitoly rozebírali jak pracuje hudební a zvukový generátor. Tato kapitola objasňuje možnosti dalšího napojení přístrojů na počítač. Další potom pojednávají o přídavné paměti a komunikaci s ostatními počítači a pracovních sítích.</p>
<p>Rozšiřující konektor</p>	<p>V zadní části počítače se nachází hlavní 64 pinový Eurokonektor. Umožňuje přenos energie do jiných přístrojů a zpracovává data, barvy a zvukové signály. Funkce každého pinu je osvětlena na obrázku přiloženém v příloze originál manuálu. Je to zásuvka, která přijímá odkazy jako PRINTERS, SCANNERS a VIDEO DIGITISERS</p>
<p>Zásuvka SCART</p>	<p>Tato 21 pinová zásuvka dokáže zpracovat super kvalitní video signál jako i celkový stereo audio výstup ze SAMa, co způsobuje lepší, audiovizuální dojem jako z UHF zásuvky. Se SCART zásuvkou se červený, modrý a zelený komponent video displeje zpracovává samostatně a jsou synchronizované. Zvukové signály jsou oddělené na levý a pravý kanál. Částečně se to využije při použití monitoru, video digitizátora a profesionálních audiovizuálních nahrávacích zařízení. Funkce každého pinu se nachází v příloze.</p>
<p>Joystick</p>	<p>Představuje nejjednodušší a nejskladnější přídavné zařízení pro váš počítačový systém. Umožňuje kontrolu pohybu a akce v hrách velmi rychlým způsobem. Pohyb se ovládá krátkou pákou, koulí, které mají tu samou funkci jako kurzor s pomocnou schopností kontroly akce nebo tlačítka pal /obeň/. I když tento port je běžně 9 pinový Atari typ, dva hráči můžou ovládat vlastní Joystick. Je totiž vytvořena svléštní linka a umožní to speciální MGT zásuvka/zástrčka adaptér pro duální kontrolu.</p>

MOUSE

Toto malé zvíře se napojuje v zadní části do zásuvky pro myš. Kontroluje se jím kurzor na obrazovce. Víceré možnosti poskytují tlačítka, nebo koule zabudované v ní. Na čtení souřadnic se používá funkce XMOUSE, zjišťuje x-ovou souřadnici a YMOUSE zjišťuje y-novou souřadnici. Nastavení tlačítka od 1 do 3 se uskutečňuje funkcí BUTTON n. Pro upřesnění se podívejte do sumáře.

Světelné pero

Tímto zařízením se přímo komunikuje s obrazovkou. Paprskem světla se přímo zadají x a y souřadnice. Obzvláště výhodně se používá na práci s uměleckými grafickými balíky /obsahem/. Čtení souřadnic se provede použitím funkce XPEN a YPEN.

Použití tiskárny

Na použití SAMe jako slovního procesoru, nebo jednoduše tisknutí programů a obrázků z obrazovky na papír jako tvrdou kopii budete si muset koupit přístupnou tiskárnu. Nejběžnější druh / a běžně i nejlacinější / se nazývají DOT MATRIX PRINTER. Pomocí jemné sítě jehliček klepe na pásku a vykreslí se body požadovaného vzoru. Tyto vzory jsou podobné síti pixelů na obrazovce. DAISY WHEEL PRINTERS používají šebrovaný válec s destičkou obsahující dvojici písmen. Při potřebě změny písma však musíte válec vyměnit. LASER PRINTERS používají na zobrazení laserový paprsek. Mimo operačních informací, které budou dodané s tiskárnou, se musíte seznámit s následujícími specifikacemi.

LPRINT a LLIST

Obě instrukce LPRINT a LLIST pracují stejně jako PRINT a LIST. Oznamují požadavek tisku na tiskárnu namísto na obrazovku. Můžete použít TAB stejně jako se píše v kapitole 6 spolu s AT, který pracuje podobně.

DUMP

Tisk obrázků z obrazovky na papír provádí příkaz DUMP.
Předpokladem je, že vaše více barevné umělecké dílo, které si chcete vytisknout na papír bude vytištěné na monochromatické tiskárně. Může se použít v každém

MODE a kopie se zobrazí následovně!

V MODE 1 a 2 celé sady pixelů na černo

V MODE 3 a 4 jiná barva než PAPER na černo

DUMP nepracuje s listingem, který jste vymazali CLEARem.

Když změníte názor již po zahájení tisku, tak přerušit program můžete pomocí BREAKu.

DUPM umožňuje vytvořit kopii všeho možného na obrazovce. Pokud nesadíte jinak, tak se vytisknou písmena běžných rozměrů. Když chcete dostat celý text obrazovky do tiskárny, potom použijte příkaz DUMP CHR\$. Zkušební programátor může poznamenat, že LPRINT se provádí kanálem 3, který je normálně otevřený ke kanálu „p“ /souběžná tiskárna/

Flexibilní systém na změnu písma se nachází s kódy většími jak 128 v sekvencích, které umožňují změnu sady písmen v době tisku / PRINTu/

Na straně 108 originál manuálu je vyobrazení možnosti připojení různých přístrojů k SAMovy.

K A P I T O L A 10

PAMĚŤ

- operační paměť
- FREE, RAMTOP a MEMO
- OPEN a CLOSE
- strojový kód
- PEEK a POKE
- USER, CALL, CLEAR
- IN a OUT
- používání disk drive

	<p>Čím víc programů budete psát, tím víc budete zjišťovat, že limitujícím faktorem při vaší tvořivé práci bude paměť počítače. Takže musíte co nejužitečněji využít širokou, ale nemilosrdnou paměť.</p> <p>Váš SAMCoupé má 512x1024 bytů DRAM, přičemž 256K jsou vždy k dispozici. Zdvoujnásobit paměť na 512K se dá zapojením dvou mimořádných paměťových čipů, které jsou přístupné u MGT. Jednoduché instrukce pro zapojení jsou přiložené.</p>
Bity a Byty	<p>Byte je jednotka na uložení v paměti, která může uskladnit číslo od 0 po 255 a každý byte má adresu - číslo od 0 po 528K. Každý byte je vytvořený z 8 menších jednotek paměti bitů. Zkušební uživatelé se mohou zapojit do diskuze o strojovém kódu v této kapitole.</p>
Hltače paměti	<p>Komplexní programy BASICu potřebují překvapivě malý obsah paměti, ale celoplošné barevné obrazy grafiky zabírají velkou část této látky. Nejšíře popsany řádek programu může obsahovat až 127 jednotlivých specifikací, což zabere až 16127 bytů. Pole mohou zaplnit každou část přístupné paměti. Řetězec může mít 65520 znaků. Po rozpoznání příkazu se zabere prostor jako při jednom znaku, přitom funkce jsou ukládané jako dva znaky. Ostatní typy informací zaberou různé rozsahy paměti.</p>
MIDI paměť	<p>Pro příklad, každá MIDI datová položka je tvořená 8 bitmi / 1 byte / plus 1 bit pro start a stop. Zvukové informace při přenosu kanálem označeným číslem zabírají 3 byte plus jeden bit pro start a stop.</p>
Obrazová paměť	<p>32K paměti je věnovaná obrazovce. Když jsou zapojené další obrazovky pomocí OPEN SCREEN, tak každá ještě zabere dalších 32K</p>
Funkce paměti	<p><u>FREE</u> - vás neustále obeznamuje s volnou kapacitou paměti pro program v BASICa a pro proměnné. Po nastypování:</p> <p style="padding-left: 40px;">□ PRINT FREE</p>

RAM TOP startuje na 81919 a oznámí vám nejvyšší možnou adresu přípustnou pro BASIC. Zadává se použitím CLEAR příkazu:

```
□ CLEAR 70 000:PRINT RAMTOP
```

MEM\$ /n TO m/

určuje sekci paměti pro řetězce. Vyhledat ho můžete pomocí INSTR a uložit /POKE/ kdekoliv chcete. Délka paměťového řezu musí být kratší než 65536, n a m včetně, přičemž n se může rovnat 0. MEM\$ velmi efektivně pomáhá přesouvat velké plochy paměti velmi rychle použitím POKE. Další hodnotnou schopností je možnost rychlého vyhledávání v paměti pomocí INSTR určité plochy. Proces je velmi rychlý při hledání po celé paměti. Na najetí umístění částečného řetězce použijte:

```
10 LET adr=1
```

```
20 LET adr=INSTR/adr, MEM$/0 TO 65534/, a$/
```

```
30 IF adr <> 0 THEN PRINT adr:LET adr=adr+1:GOTO 20
```

Hledat a přemísťovat řetězce můžete s použitím POKE. Podrobné detaily o paměti najdete v technickém manuálu.

strany extra
paměti

OPEN_n - rezervuje n extra paměťových stran 16K. Každá je použitelná pro BASIC, kde je možné. To umožňuje CLEARu udát vyšší RAMTOP.

CLOSE_n - uvolňuje n 16K stran z použití BASICovými programy a proměnným, ale pouze jen když RAMTOP umožňuje. Příčinou je BASIC, který udržuje všechny stránky, včetně té, kde je uložený RAMTOP. Uvolněné stránky se můžou využít pro onrazovku DOS a jiné programy.

strojový kód

Dlouhé části paměti se zachrání/uvolní/ a zvýší se rychlost když klíčové slova BASICu a procedury se provedou mimo, když použijete přímo počítačové kódy. Strojový kód je sada instrukcí, které mikroprocesor Z80B používá. Můžete s nimi psát přímo programy v jeho povelovém jazyku. Takovýto program musí být zakódovaný v sekvenci bytů používajících assembler. Existuje však i mnoho výborných knížek když se chcete sám naučit o použití strojového kódu.

CODE	<p>Informace se uchovávají a nahrávají v bytech použitím klíčového slova CODE, bez nějakého dalšího definování co je účelem této informace.</p> <p>Například:</p> <pre>SAVE "jméno"CODE LOAD "jméno"CODE ,nebo [F8]</pre>
DPEEK addr	<p>Strojový kóděři budou mít užitek z následujících informací. PEEK addr odhaluje obsah strojové paměti na adrese v rozsahu od 0 po 528K. Vrací výsledek od 0 po 255</p>
DPEEK addr	<p>umožňuje dvojnásobný PEEK. Má stejnou hodnotu jako PEEK addr+256 x PEEK/addr+1/</p>
POKE addr,x	<p>používá se na přímé uložení čísla na adresu paměti v rozsahu od 0 po 524 287.32 čísel se dá najednou uložit do postupných adres, jako např.:</p> <pre>POKE addr,255,129,129,129,129,129,129,255</pre> <p>Podobně můžete uložit i řetězce pomocí POKE:</p> <pre>POKE addr,a\$ nebo např POKE addr,"testing"</pre>
	<p><u>DPOKE addr,value</u> - umožňuje násobný POKE. Hodnota musí být z rozmezí 0 po 65535.</p>
	<p><u>USR n</u> - spustí strojový kód s udaným číslem jako startovací adresou. Hodnoty od 0 po 52487 můžete použít.</p>
	<p><u>USR₂ addr</u> - udává řetězcové rozhodnutí o programech ve strojovém kódu na označenou adresu. Strojový kód může končit s BC=délka, DE=start, A=strana začátku /když DE >=32768/.</p>
	<p><u>CALL n</u> - se může použít na shromáždění částečných adres s výpisem číselných a řetězcových parametrů, když požadujete například:</p>
	<pre>CALL,address,x,y,SAMS /objasňuje technický manuál/</pre>
	<p><u>CLEAR</u> - uvolňuje prostor v paměti počítače obsazený proměnnými. Vymaže je pryč.</p>

	<p><u>CLEAR_n</u> - pracuje podobně jako CLEAR, ale mění RAMTOP systémovou proměnnou na pozici specifikovanou číslem n.n musí být včetně 4 stran určených BASICem, pokud je zapojený a musí být <=81919 mimo extra stran otevřených OPENem.</p>
IN a OUT	<p>Počítač operuje s 65536 IN/OUT výstupních portů, ve skratce I/O porty. Používají se jako na vnitřní, tak i na komunikaci s vnějšími zařízeními. Každý port se dá kontrolovat funkcí IN a specifikací OUT.</p> <p style="text-align: center;">IN address</p> <p>čte byte z adresového portu a dává ho jako výsledek</p> <p style="text-align: center;">OUT address, value</p> <p>píše danou hodnotu /mezi 0 a 255/ do adresového portu /mezi 0 a 65535/</p>
DISK drive	<p>Výběr jednoho, nebo dvouch ultra tenkých diskových jednotek, které se můžou uložit zepředu do počítače. Jsou přenosné a každé může uložit 1Mbyte paměti na 3.5" floppy disk, poskytující 780K na jednom disku po založení. Disk umožňuje velmi rychle nahrání a natahování programů na disk a z disku. Velmi výhodný je způsob uložení souboru údajů. Když ani tato masivní kapacita nepostačuje, tak musíte se napojit na gigantickou síť dat, o čemž se píše v následující kapitole.</p>
Použití disku	<p>Využít plné výhody extra rychlosti a paměti přístupné s MGT přenosnými disky je možné za pomoci několika dalších příkazů. Nejprve diskový operační systém ve skratce DOS, se musí nahrát do paměti.</p>
BOOT	<p>Nahrává DOS z disku a umožňuje použít speciální DOS operace:</p>
FORMAT	připravuje nový disk na příjem dat
DIR	zkouší DIR řízení souboru na disk
MOVE	přesouvá soubory dokola
ERASE	vymazává nepotřebné soubory
DEVICE	uvádí do činnosti příkazy SAVE, LOAD, MERGE a VERIFY

při práci s diskem, nebo sítí více počítačů. Prostudujte Sumář kvůli detailům, ostatní se nachází v manuálu přiloženému k disku.

K A P I T O L A 11

KOMUNIKACE

- síť
- výuka
- toky a kanály
- pirátství je krádež

Síť

Jednou z nejvzrušivějších črt Vašeho SAMa je jeho schopnost komunikovat s jedním, nebo více podobnými počítači. MIDI IN a MIDI OUT spojky v zadní části počítače jsou připravené na zapojení do takovéto sítě.

Výuka

Na nejjednodušší úrovni není požadované žádné dokonale zařízení poslední doby ani software, protože všechno se nachází v počítači. Kasetový systém pracuje velmi dobře a můžete hrát více uživatelské hry na dvou, nebo více různých místech. Jedinou podmínkou je vaše představitivost! Důkladnější síť se tvoří diskami a tiskárnami a existují metody jak se napojit na celosvětovou síť.

Jako samostatný počítač, SAM Coupé byl zkonstruovaný, aby exeloval jako výukový počítač. Od základních škol až po univerzity jeho barvy, zvuk a přídatné zařízení poskytují kompletní rozsah výběru. Začátečníci a invadní uživatelé mohou využít výhody MGT myši na jednoduchou kontrolu, jako i MGT světelné pero na pomoc při čtení. CSIZE příkaz vytvoří stálou širokou škálu znaků pro zřekové postihnuté. Až 16 přístrojů se dá napojit do sítě ve školní třídě. Spojují se samozřejmě všechny výhody při ohraničení zdrojů disků a tiskáren. Data se nahrávají do každého počítače a výhody celého systému se monitorují na počítači učitele. Dají se i nahrát z něho. Kanál ~~20~~ slouží normálně jako vysílací stanice, spolu s kanály ##1 až ##5 používanými pro stanice osobního operátora, které jsou umístěné na počítačích studentů.

Existuje široký rozsah výukového softveru pro SAMa. MGT má velkou radost ze spolupráce s řidiči, učiteli a samozřejmě mladými uživateli, kteří mají zájem o poslední informace o výukových programech a pod.

Následující sekce je částečně technická a můžete ji přeskočit.

V síti je kanál #0 vysílací stanice, s kanály #1 a #15 používanými pro kanály ostatními uživateli. Proudů 0 a jsou normální OPEN kanálu k.

```
□ PRINT# 0;"testing":PAUSE
```

vytiskne na spodním okraji obrazovky, když PRINT #2 tiskne nahoře obrazovky jako obyčejně /používá kanál S/ PRINT #3 působí jako LPRINT, používá kanál 'p' a můžete si vybrat i způsob použití:

```
□ OPEN #6;"p":PRINT #6;"hello"
```

způsobí vyslání výstupu do tiskárny přes tok #6.

INPUT #2 - je též přípustný a může se použít na získání přístupu k vrchní části obrazovky jako v následujícím příkladě pro tisknutí hodnot blízko "hodnoty"

```
INPUT 2;"value:";v
```

Na rozdíl od dolní obrazovky, řádek vlevo není vymazán po stlačení RETURN. Příkaz RECORD pracuje přes tok 16. Ten tlačí do řetězcové proměnné specifikované RECORDem a bude tak pokračovat až po RECORD STOP.

Piráctství

Není nic moudřejšího nad krádež. Když kopírujete program pro zadní kolečka, je to praktické, ale kradete. Okrádáte a zraňujete autory. Piráctství zdražuje ceny softwaru a odstraňuje lidi z obchodu. Když bude méně a méně původního softwaru produkovaného, tak to postihne každého z Vás.

Při použití speciálních programů, značička BREAK, se dá použít na přenos softwaru z kazety na disk pro vaše vlastní potřeby. Prosíme aby jste to nezneužili pro veřejné potřeby. MGT chce dát jasně najevo, že dáme tvůrcům softwaru všechny informace na ochranu jejich her, které běží na SAM Coupé. My očekáváme pomoc od odpovědných rodičů na zabránění piráctví našich výrobků.

K A P I T O L A 12

KÓDY CHYB

- zjišťování chyb
- kdy chyb a oznámení, rady
- použití ON ERROR

Když uděláte chybu při programování, nebo když žádáte aby SAM provedl nemožné, tak vám SAM pomůže při zjišťování a ozřejmí i daný problém. Takto se stane, že na spodní části obrazovky, se automaticky objeví, co se děje chybně a kde se chyba nachází.

Když je programová chyba náhodná, pomocné rady jsou podávány v následujícím pořadí:

jedno z kódových čísel a údaj, který přesně ukáže druh chyby, následuje číslo v řádku, kde je chyba ukrytá a číslo specifikace v tomto řádku, která zapříčiňuje chybu.

5 NEXT without FOR, 700:3

Říká, že typ chyby je 5-co znamená, že jste použili NEXT ale zapoměli příkaz FOR a tento problém je na řádku 700 ve specifikaci 3

Kód o kterém sníváte je 0, co znamená, že je všechno OK. Ale chyb se dočkají i nejzkušenější programátoři.

Poskytujeme Vám výpis kódů chyb a příslušných instrukcí k nim.

Kódy chyb

Chybové oznámení

- | | |
|---|---|
| 0 | OK - bez problémů, úspěšné sestavení, všechno v pořádku |
| 1 | Mimo paměti - není dostatek prostoru v paměti počítače pro vaše přání |
| 2 | Jméno proměnné není možné najít - počítač neumí najít nějakou proměnnou, protože buď není doposud nahráná, nebo určena, nebo zadána, nebo jste neurčili její rozměr |
| 3 | Data jsou všechny načtená - chcete načítat ale už jste všechny data prošli |
| 4 | Index je chybný - buď číslo indexu je chybné, nebo index je mimo rozsahu pole |
| 5 | NEXT bez FOR i když je správná proměnná s tím samým jménem, tak kontrolní proměnná nebyla založená specifikací FOR |
| 6 | FOR bez NEXT-i když je FOR smyčka čekající na spuštění neexistuje specifikace NEXT, která jí usměrňuje |
| 7 | FN bez DEF FN - chybí definování funkce uživatelem |

- 8 Return bez GOSUB-chybí procedura do které „y RETURN zavítal
- 9 Chybějící LOOP- zapoměli jste, že znak DO potřebuje LOOP
- 10 LOOP bez DO - když LOOP čeká na zpětný skok a neexistuje DO, kde by začal
- 11 nejsou data POP- chcete použít POP, ale namáte data na skladě
- 12 chybí DEF PROC - chcete ukončit proceduru s END PROC ale neexistuje. Tatu chyba může vzniknout i při psaní jako "PRUNT"
- 13 END PROC chybí-zapoměli jste označit proceduru
- 14 BREAK do programu-stlačili jste BREAK nebo ESC mezi dvěma specifikacemi. Objeví se číslo specifikace před BREAKEM. CONTINUE spustí program na následující specifikaci
- 15 BREAK-CONTINUE se opakuje-BREAK byl stlačený když se uskutečňovala periferní operace, takže po spuštění CONTINUE se poslední specifikace opakuje
- 16 STOP specifikace - když chcete pokračovat, tak s CONTINUE na následující specifikaci
- 17 STOP v INPUTU - když chcete pokračovat, tak opět na té poslední specifikaci INPUT, kde program zastavil
- 18 Chybné / neplatné/ číslo souboru - když chcete nahrát soubor a zapoměli jste mu přiřadit název, nebo je delší než 10 znaků
- 19 Nahrávací chyba-soubor který chcete nahrát je nalezen ale cosi je chybné a odmítá se nahrát, nebo selhává při VERIFY. Zkontrolujte kabely, úroveň zvuku, kazetovou pásku a nečistou hlavu přehrávače kazet
- 20 Chybné zařazení - pokoušíte se SAVE, nebo LOAD data, ale používáte chybnou věc na vsut/výstup/ jako disk drive namísto kazetového přehrávače, nebo jste ho nezapojili
- 21 Chybné číslo proudu - Když používáte číslo proudu /toku/, které je nevhodné. Proudů 0 až 16 jsou cestami do různých kanálů.
- 22 Konec souboru-byl dosažen konec souboru, obvyčejně na disku

- 23 Chybná barva - Zkusili jste specifikovat nepřípustnou barvu.Hranice jsou od 0 do 127
- 24 Chybná paleta barev - Paletový stolek má hranice od 0 do 15
- 25 Přiliš mnoho změn palet - maximální počet změn PALETTE je 127 na obrazovce
- 26 Chyba parametru - buď jste použili chybné číslo argumentu,nebo špatný typ argumentu,jako např.číslo místo řetězce
- 27 Chybný argument - použití nevhodného argumentu pro funkci
- 28 Přetečené číslo - přesahuje SAMovu říši
- 29 Nerozumím - počítač je z vás hotový
- 30 Celé číslo mimo rámec -požadované celé číslo přesahuje vhodné rozměry
- 31 Specifikace neexistuje - počítač se neumí rozhodnout nebo splnit instrukci bez potřebné specifikace,např. pro GOSUB jste vymazali specifikace a potom jste RETURN - ovali
- 32 Vypnutá obrazovka - požadavky na grafické zobrazení nemůžou být zobrazené
- 33 Není prostor pro řádek - v paměti není dostatek prostoru na vámi požadovaný řádek,nebo není možné přečíslování řádků.
- 34 Chybný mód obrazovky - existují 4 módy pro obrazovku 1 až 4
- 35 Chybný kd BLITZ - použili jste BLITZ s řetězcem, který nedosahuje vnímatelné grafické příkazy
- 36 Přiliš velký rozsah zabraného prostoru-nemůžete změnit prostor který chcete zaujmout / GRAB / v paměti
- 37 Chybný blok PUT - řetězec který chcete dát na obrazovku nezodpovídá skutečnému bloku dat na obrazovce.
- 38 PUT maska chybná-druhý řetězec nemá délku prvního řetězce

- 39 Chybí END IF-když používáte dlouhý IF přes vícere řádky a rozhodli jste se vynechat THEN, na závěr specifikace jste zapoměli přiřadit END IF
- 40 Chybné jméno proměnné-jméno číselné proměnné se začíná písmenem, nebo je delší než 32 znaků, jméno řetězcové proměnné a jméno proměnné pole řetězcového typu nekončí s \$, nebo je delší než 10 znaků
- 41 paměť pro BASIC je plná - pročti kapitolu 10, která se zaujímá rozdělením paměti.
- 42 Řetězec je příliš dlouhý - přesahuje 65520 znaků, nebo v případě STRING\$ 512 znaků
- 43 Chybné číslo obrazovky-při zapnutí počítače má obrazovka číslo 1, vy zkoušíte vstoupit do obrazovky pomocí OPEN mimo přípustný rozsah 2 až 16. Podobně jak použijete příkazy SCREEN a DISPLAY bez příkazu OPEN.
- 44 Obrazovka je už otevřená-opětovně se pokoušíte otevřít obrazovku, která je už otevřená
- 45 Kanál je už otevřený- chcete otevřít obsazený kanál
- 46 Chybný kanál-pokoušíte se otevřít nevhodný kanál.
- 47 Kanál není otevřený-používáte číslo uzavřeného kanálu
- 48 Chybná adresa CLEAR - číslo pro příkaz CLEAR mimo rozsah BASICu
- 49 Chybná nota-příliš vysoká, nebo příliš nízká nota, rozsah je udán od -60 do 71
- 50 Příliš dlouhá nota-musí být kratší než 16 sekund a proto příkaz nesmí být delší než jako BEEP 16
- 51 Chyba FPC-chyba ve strojovém zakódování výpočtu s pohyblivou desetinnou čárkou.
- 52 Mnoho definicí-může vzniknout použitím DEF KEYCODE nebo DEF FN
- 53 není DOS-není DOS systém na BOOT z disku nebo na disk
- 54 Chybný WINDOW-hranice vašeho WINDOW nejsou přípustné v běžném MODE a CSIZE.

Můžete si naplánovat program na chyby a počítač při nalezení chyby bude postupovat rutinně podle něho. Výpis chyb se spouští příkazem ON ERROR a vypíná se ON ERROR STOP. Když je spuštěný tento proces, tak jsou použité následovně zvláštní proměnné:

ERROR - číslo kódu chyb

LINO - číslo řádku kde je umístěná chyba

STAT - specifikace, kde se adresa nachází

Zkušební uživatelé si můžou vytvořit vlastní program, který se má používat v programu:

```
10 ON ERROR GO TO 100
```

co znamená, že při objevení chyby přeskakuje na řádek 100, kde provede požadované instrukce, nebo GOSUB, kde skáče na podprogram.

ON ERROR se automaticky vypíná v době vaší snahy opravit chybu. Pokud se uvnitř nachází další chyba bude zpracovaná běžně. Podprogramy a procedury se vrátí na specifikaci, která následuje za špatným příkazem pokud končí tyto cykly RETURNem, nebo END PROC.

Nyní se aktivizuje ON ERROR sám a hledá další příležitost. S CONTINUE se můžete vrátit zpět do specifikace, která zapříčinila chybu, bez opětovného spuštění systému ON ERROR. Když však pokračujete zatím co chyba je přerušena 'BREAK' tak se vrátíte až na následující specifikaci. Na udržení pořádku v paměti BASICu použijte POP na zbavení se RETURNu, nebo END PROC adresy, abyste mohli rozhodnout pokračovat namísto zpětného navrácení.

G L O S S A R Y

**Vysvětlení jednotlivých
příkatů**

	<p>Tato sekce zřívatelelské příručky je přívodcem po všech slověch a výrazech, které budete potřebovat při programování na SAMovi. Obsahuje příkazy, funkce, klíčové slova a definice technických termínů a žargon. Nejjednodušší cestou na pochopení každého oíseku je vyzkoušet si techniku a pozorovat, co se přihodí. Počítač nemůžete poškodit experimentem. Klíčové slova, které se vyskytují v programech BASICu jsou tištěné velkými silnými písmeny. Např.: AND</p> <p>Definice a žargon jsou zobrazené velkými a malými písmeny.: Např. Array</p> <p>Aktuální klíče nacházející se na klávesnici počítače jsou zobrazené velkými tisknutými písmeny v hranatých závorkách, jako je RETURN.</p>
ABS	stav pro absolutní hodnotu a je funkcí pro změnu argumentu na kladné číslo.
ACS	stav pro arccos
Address	číselné umístění jednotky dat v paměti počítače.
AND	používá se na kombinování podmínek podobně jako v angličtině: IF a\$="SAM" AND x > 0 THEN PRINT x
ARRAY	pole je souborem proměnných, které jsou navzájem oddělené číslem v závorkách po názvu pole. Před použitím pole se v paměti počítače musí vymezit prostor použitím dimenze, příkaz zní DIM
ASCII	zkratka pro Americký standardní kód pro informační výměru, je nejjastěji užívaný systém kódování znaků pro přenos dat mezi počítačem a různými periferiemi jako třeba tiskárnou.
ASN	zkratka pro arcsin, je to funkce pro zpětné nalezení čísla, které bylo sinováno.
AT	říká počítači, kam má psát na obrazovku, když vyhodnocuje nejdříve číslo řádku, pak číslo sloupce
ATN	zkratka pro arctg, je to funkce pro zpětné nalezení čísla, které bylo tangetováno

Glossary

Attributy	týkají se charakterů grafického bloku na kterékoliv dané pozici obsahující jejich barvu, jasnost a v módech 1 nebo 2 jejich flash /blikání/ stav
AUTO	automaticky počítá vaše programové řádky. Jestliže bylo zadáno Věmi, počítač bude brát základní řádku podle toho, kde je programový kurzor a přidá číslo 10 na každou rovnou řádku. Nebuďte pracovat, jestliže základní číslo řádky je menší než 10 nebo více než 61439. Můžete přezkočit blok čísel řádků, pokud použijete AUTO s číslem řádku a počtem řádků. AUTO začíná číslovat na řádku 10 s krokem 10 AUTO 150 začíná číslovat na řádku 150 s krokem 10 AUTO 150,5 začíná číslovat na řádku 150 s krokem 5
BASIC	Zkratka pro začátečnický více přístupný symbolický kód, je jediný nejvíce používaný jazyk ve světě. Pokročilá verze BASICu je zabudována v paměti Vašeho SAMa.
BEEP délka, výška	instrukce způsobí, že počítač produkuje zvuk
BIN	není to funkce ani příkaz, ale spíše změna spočívající v jiné interpretaci čísla. Za BIN následuje sekvence max 16 bitů / 1 a 0 / jejichž různá sekvence reprezentuje číslo.
BIN \$	automaticky dá dvojkový ekvivalent čísla jako řetězec osmi znaků, jestliže číslo je menší než 256, jako řetězec 16 znaků, jestliže číslo je mezi 256 a 65535. Další instrukce jsou podobné, n BAND m a n BOR m, které dají binární AND, OR
BIT	nejmenší částice dat, která může být reprezentována v počítačové paměti 1 nebo 0
BLOCKS	zapíná, nebo vypíná blok grafických znaků. BLOCKS 0 vypíná blok grafiky a dovoluje předdefinovaný soubor znaků CHR\$ 128 . 143 použít jako uživatelem definovanou grafiku / UDG /. BLOCKS 1 vypíná blok UDG

Glossary

BLITZ aš	je používán ke vyhledávání řetězcového, nebo grafického příkazu velice zřídka. Pracuje ve všech módech, ale výjimečně efektivní je v módu 4. BLITZ komunikuje se speciálními proměnnými XOS a YOS, které povolují grafice blikat kdekoliv na obrazovce a číslo může být zvýšeno změnou spec. grafických proměnných XRG a YRG
BOOT	Naplní DOS z disku do počítačové paměti
BORDER	mění barvu okraje obrazovky na jinou barvu od 0 do 15 např. BORDER 6 mění barvu okraje na žlutou
BREAK	když je stlačeno toto tlačítko na zádi počítače přeruší program. Programátoři mohou používat programové techniky ke změně funkce tohoto přerušení
BRIGHT	je používáno ke zjasnění dané barvy. BRIGHT 1 aktivizuje instrukci, BRIGHT 0 nastavuje jas barvy na normál. Existuje speciální nastavení BRIGHT 16, které způsobí "zprůhlednění": jinými slovy barva na obrazovce se nezmění, jestliže je tam vytisknut znak
Bug	slangový výraz, který hlásí chybu v programu, která vyvolá problémy po spuštění pomocí RUN
Buffer	je oblast paměti umístěná mimo běžného uložení dat
BUTTON n	ukazuje stav tlačítka MOUSE při n=číslu na tlačítku od 1 do 3. Je-li tlačítko stisknuto, objeví se "1", není-li, objeví se "0". Zvolíte-li n=0 pak se objeví "1" bude-li jakékoliv, nebo všechna tlačítka stisknuta
Byte	je jednotka paměti vytvořena z 8 bitů a je dost velká aby pojala jeden znak, nebo celé číslo ≤ 255
CALL n	povolává program strojového kódu na adresu seznamem číselných, nebo řetězcových parametrů / jsou-li žádány. Čtete prosím pozorně návod k obaluze.

Glossary

CAPS LOCK	zvedá příkazy po obrazovce do hořejší pozice dokud není opětým stisknutím tato činnost zrušena.
Charakter set	/soubor znaků/ má 256 možných znaků, každý s kódem mezi 0 a 255. Předdefinovaný soubor znalá může být vytisknut pomocí: 10 FOR a=32 TO 168:PRINT CHR\$ a;:NEXT a
CHR\$	je použit jako číslo a udává jediný znakový řetězec, jehož kód je toto číslo.
CIRCLE x,y,r	kreslí kružnici na obrazovku, následuje-li za příkazem určení středu kružnice na obrazovce /x,y-souřadnice/ a poloměr kružnice/souřadnice r/. Přesahuje-li kružnice přes vrchní, nebo spodní okraj obrazovky, bude cyklicky přetočen na protějščí okraj obrazovky.
CLEAR	zbavuje všech proměnných a uvolňuje prostor v paměti počítače, který byl obsazen
CLEAR n	pracuje jako CLEAR, ale přemísťuje soustavu proměnných RAMTOP do bodu určeného číslem
CLOSE	je používán k zastavení proudu, který je normálně OPEN-propuštěn do určených kanálů
CLOSE n	uvolňuje n 16K stránek. Každá je normálně využita jazykem BASIC a proměnnými, ovšem pouze dovolí-li to RAMTOP. Po nastolení CLOSE mohou být volné stránky použity na zobrazování, DOS nebo další jiné programy.
CLOSE SCREEN n	uvolňuje paměť obrazovky, jestliže tato již existuje tak, že její paměť může být znovu použita, např.: CLOSE SCREEN 2
CLS	znamená "stírání obrazovky" a utírá řadu zobrazení z obrazovky. CLS nebo CLS 0 čistí celou obrazovku zatím co CLS 1 čistí pouze běžnou část - WINDOW - a CLS # nečistí pouze obrazovku, ale provádí i OVER 0:INVERSE 0:FLASH 0:PAPER 0:PEN 7:BORDER 0:PALETTE

CODE	je používán jako řetězec a udává kód prvního znaku v řetězci. Je-li řetězec prázdný, udává 0.
Command	je slovo, nebo krátká fráze jako OPEN SCREEN, DRAW a GO SUB, používané v programu k instrukci počítače provést určitý úkol.
Condition	program musí rozhodovat je-li správný nebo obsahuje-li chybu před jejím nalezením. Příkaz může být použit po EXIT IF, WHILE nebo UNTIL. Stav vždy udává numerický výsledek 1 pro správnost, 0 pro chybu. /vyzkoušejte příkazem PRINT 2=1/
CONTINUE	příkazuje programu pokračovat po té, co byl zastaven zprávou o chybě
CONTROL Codes	jsou zvláštní znaky, které se neobjevují na obrazovce, ale jsou v činnosti když počítač tiskne nebo kreslí. Jakmile jsou jistě kódy vyvolány z klávesnice bude počítač vyhledávat aby se přesvědčil, je-li na ně dekomprimována definice. Viz DEF KEYCODE
COS	následuje-li za ním úhel v radiánech jde o funkci, které počítá cosinus.
CPS	znamená znaky za sekundu a označuje teoreticky nejvyšší rychlost, kterou může tiskárna tisknout na papír
Crash	je slangový výraz pro počítačový program svěřený svou vlastní inteligencí a to, co uvede nelze vyjmout
CWIDTH, height	vás nechá změnit tvar znaků znáčknutím, nebo rostažením. Např. v režimu 3
CWIDTH 6,9	budou všechny vaše znaky nabírat 6 obrazových prvků na šířku a 9 obrazových prvků na výšku. Šířka může být 6 nebo 8 a výška může být od 6 do 32

Glossary	
CTRL	se používá ke změně klávesnice do stavu CONTROL
Cursors	jsou indikátory pozice, obvykle zobrazovány na obrazovce ve formě inverzního symbolu. Řádkový kurzor EDIT je zobrazen inverzním symbolem při normálním stavu klávesnice a inverzním symbolem + pracuje-li CAPS uzávěr. Kurzor běžného řádku v programu je zobrazen symbolem ">"
Cursor keys	jsou čtyři směrové šípové klápy používané k pohybu programového kurzoru po obrazovce, jakož i ovládání pohybu během počítačových her.
DATA	DATA údaje mohou obsahovat klíčová hesla jako "goto" bez toho aby byly změněny v známky. Jiná než tato hesla, čili hesla bez úvozovek, musí být legální numerickou proměnnou. SAM 1 lze přijmout ale 1x ne.
Debug	je slangový výraz pro vytažení chyb programu, nebo "bugs", a jejich oprava
DEFAULT	je hlavně používán v procesech a vytváří proměnnou pouze tehdy, jestliže tato neexistuje
DEF FN	je zkratka pro použitelnou definovanou funkci. Její název může být jakkoliv dlouhý a musí začínat písmenem. Může pokračovat písmeny, čísly nebo podtržením. Řetězcové funkce musí končit znakem \$. Umístění údajů DEF FN nemá vliv na rychlost provedení programu.
DEF KEYCODE	je používán k vytvoření zhuštěných kódů, které budou automaticky roztaženy vyvoláte-li je klávesnicí. Např: <pre>DEF KEYCODE 195:PRINT 123:PRINT "sam":ENTER</pre> přičiří "PRINT 123" atd kódu 195, kód přiřazený klíči F3. Takže kdykoliv stisknete F3 bude výsledek stisknutí 123 sam v horní části obrazovky. Přitete-li výsledný sloupec k řádku DEF KEYCODE, bude potlačeno automatické vrácení RETURN.

Glossary

	<p>takže: DEF KEYCODE 195:PRINT "sam": má výsledek tisku "sam" v ediční oblasti. Namísto v řádku můžete tuto technologii použít v řetězci takto:</p> <p>DEF KEYCODE 195,"TESTING:"</p> <p>Definovaný kód může být vymazán opomenutím, nebo čímkoli, co bylo mezi úvozovkami jako je tomu v obou následujících příkladech:</p> <p>DEF KEYCODE 195: DEF KEYCODE 195," "</p>
DEF PROC	<p>je údaj, který předchází definici PROCesu jenž má název a soubor proměnných, které jsou použity pro vložení dat do procesu.</p>
DELETE	<p>při stisknutí máše směrem doleva. Při současném stisknutí SHIFT a DELETE máše směrem doprava.</p>
DELETE n TO m	<p>máše část řádku programu mezi dvojicí vybraných čísel. Jsou-li čísla řádků totožná, vymáše se jediný řádek. Vynecháte-li n, je použito číslo řádku 1. Vynecháte-li m, je použito číslo posledního řádku v programu.</p>
DEVICE	<p>umožňuje použít příkazy SAVE, LOAD, MERGE a VERIFY pohonem disku, nebo sítí počítače. DEVICE lze použít následujícími způsoby:</p> <p>DEVICE d - nařizuje počítači použít pohon disku 1 DEVICE d+ - dělá to stejné DEVICE D1 - dělá to samé DEVICE N5 - používá síťovou stanici 5 DEVICE T' - používá pásku běžící standartní rychlostí pro nahrávání dat DEVICE T35 = používá běžící pásku, které data nahrává zrychleně. 35 je nejvyšší rychlost vhodná pro ověření, sde je páska schopna nahrát data, 112 je standartní relativně nízká rychlost. Bohužel vysoké rychlosti jsou méně spolehlivé než nízké. Nemí třeba pamatovat si, v jakých rychlostech jste</p>

Glossary

	svá data, protože příkaz LOAD automaticky snímá celou škálu rychlostí.
DIM	vytváří prostor v paměti počítače pro řadu/pole/. Název pole může být dlouhý až 10 znaků včetně mezer. Čísla která tvoří DIMenze / rozměry/, následují v závorkách a mohou být mezi 1 a 65535. Celková velikost pole je limitována pouze dosažitelnou pamětí.
DIR	pracuje pouze s disky
Disk	je magnetické zařízení pro rychlé uložení, skladování a zavádění počítačových dat. SAM Coupé může pracovat s 3.5 palcovými "floppy" disky / pružné disky /
Disk drive	Počítač je opatřen dvěma zdífkami pro obvykle vyráběnou zvláštní paměť pohonů disku. Každý pohon může obsahovat navíc 256K DRAM
DISPLAY n	mění mřížku, které je zobrazena, ale ne mřížku užívanou BASICem. Jedna může být vystavěna tak, že je neviditelná a jiná může být běžně zobrazena.
DISPLAY	
DISPLAY 0	běžná mřížka
DISPLAY 1-16	mřížka s požadovaným číslem
DIV	je ekvivalentní rozdělení jednoho celého čísla jiným, takže: PRINT 24 / DIV 24
DO	pracuje jako znak do které se může stejný údaj smyčky vrátit. Takto: 10 DO 20 PRINT "PLAY IT AGAIN SAM"; 30 LOOP bude obíhat v cyklech neomezeně, dokud nebude určeno příkazem DO. Můžete použít příkaz DO WHILE, jehož výsledkem bude v části programu mezi DO a LOOP provedení WHILE -určená podmínka je správná. Alternativně můžete použít DO UNTIL, jenž je opačným stavem, intru-

Glossary

	<p>ujícím program provést tuto část mezi DO a LOOP při chybné podmínce,ale tak,aby se zastavil dokud se nestane podmínka správnou.</p>
DOS	<p>je zkratka pro diskový operační systém</p>
DPEEK adres	<p>dvojitě PEEK,jenž je ekvivalentem PEEK adresa +256 = PEEK(adresa +1).Viz DPOKE</p>
DPOKE adres	<p>provádí dvojitě POKE jenž je ekvivalentem POKE adresa,číslo-INT (číslo/256) = 256 POKE adresa +1,INT(číslo/256)</p> <p>Jinými slovy,nejméně významný byte čísla je uložen do adresy,nejvýznamější byte je uložen do následující vyšší adresy.Hodnota musí být mezi 0 a 65535</p>
DRAM	<p>je zkratka pro dynamickou paměť s přímým přístupem.</p>
DRAW x,y	<p>je příkaz pro nakreslení relativní linky k této pozici,ale jdoucí x obrazových prvků vpravo a y obrazových prvků nahoru.Jestliže má x,nebo y zápornou hodnotu,pak může být pohyb směřován vlevo nebo dolů.</p>
DRAW x,y,z	<p>kreslí přibližnou linku z běžné pozice při souřadnicích x,y za současného tvoření křivky procházející úhlem z</p>
DUMP	<p>se používá s určitými typy tiskáren k zastavení nepovedených složitých zobrazení jkmile se začnou tisknout černobíle.DUMP se může použít v jakémkoli obrazovém MODE a bude končit odstíněním natištění kopie.Všimněte si,že v MODE 3 bude výsledek dvakrát tak široký než v jiných MODE,protože na obrazovce pracuje horizontálně 512 obrazových prvků namísto 256 a tak největší výpis na obrazovku v tomto MODE nebude vhodný pro malé tiskárny!</p>
DUMP CHR\$	<p>provádí výpis textu na běžnou obrazovku</p>
EDIT	<p>po stisknutí se objeví řádek programu obsahující kurzor v editující oblasti.Napíšete-li číslo řádku a pak zmáčknete klávesu EDIT,objeví se žádaný řádek v edi-</p>

	<p>tační oblasti. Kurzor se pohybuje vlevo, vpravo, nahoru nebo dolů v editovaném /opravovaném, upravoavném/ řádku pomocí kurzorových kláves nebo posouvá nahoru nebo dolů listování.</p>
ELSE	<p>je částí struktury IF _ THEN a musí přijít na začátek nového údaje. Jestliže údaj následující za IF je chybný, program normálně přeskočí na další řádek. Jestliže dvojice IF_THEN, má přiřazen příkaz ELSE později na řádku, bude program pokračovat údaji následujícími za ELSE. Na druhé straně, jestliže podmínka následující za IF je správná, bude řádek proveden pouze k výrazu ELSE.</p>
ELSE - IF	<p>má zvláštní význam v tom, že jakmile je jakákoli podmínka správná /pravdivá/, budou řádky nebo výrazy ní provedeny a jakmile se nějaký příkaz ELSE IF nebo ELSE setká, skočí program přímo do pozice bez odstředně za END IF.</p>
END IF	<p>musí být použit na konci podmínky IF</p>
END PROC	<p>se používá k označení konce procesu vždy definovaného příkazem DEF PROC.</p>
ERASE	<p>pracuje pouze s disky</p>
ESCAPE key	<p>po stisknutí dovoluje přerušit vyjít do programu</p>
EXIT IF	<p>je částí soustavy DO-LOOP a používá se k přenesení soustavy odkudkoli v prostředí, tak aby nebyla blízko u DO nebo LOOP. Je-li určená podmínka za EXIT IF správná, přeskočí provádění programu na údaj za LOOP, jinak se nic nestane.</p>
EXP	<p>je matematická funkce k počítání exponenciálních čísel a může být definována takto:</p> $\text{EXP } x = e^{\uparrow} x$

F keys	je skupina funkčních klapek pojmenovaných F0 až F9 .. Každá z těchto kláves může být předvolena k zajištění specifické funkce jak je ukázáno v kapitole 4 a údaj jednotlivých kláves může být během programování předefinován.
FATPIX	v grafickém MODE 3 jsou obrazové prvky úzké, použítá jinými MODE je pouze polovina rozsahu. Použitím příkazu FATPIX 1 můžete zdvojnásobit šířku obrazových prvků v MODE 3 vytvořením vertikální linky stejné síly jako je horizontální a XRG bude rozděleno. Pro návrat do normálního stavu věci použijte FATPIX 0 a XRG se zdvojnásobí.
File	jsou data vybavená v počítači, která mohou být uložena a pracovat jako jejich magnetický ekvivalent na papírový dokument, plně vybavená a ve své vlastní jmenné vazbě.
FILL	je grafický příkaz pro vyplňování tvaru barvou. Viz kapitola 6.
FLASH	pracuje pouze v MODE 1, 2 a je symbolem dané barvy provádějící FLASH rytmicky. Příkazem FLASH 1 zapínáte, FLASH 0 vypínáte. FLASH 8 nebo 16 nastavuje "transparentní" stav, který je přesunut vlevo při jakékoli pozici symbolu.
Font	popisuje styl a výskyt písmen, čísel a symbolů jak se objeví při psaní na obrazovku nebo na papír. Popisy znaků mohou být změněny jak je popsáno v kapitole 6.
FOR	je příkaz používaný vždy ve spojení s NEXT pro omezení opakovaného programování. Např.: namísto vložení sta řádků pro sto podobných výpočtů můžete zkrátit postup napsáním: FOR times = 1 TO 100: NEXT times
FORMAT	pracuje pouze s disky
FREE	je funkce, která vám přesně ukáže kolik paměti je volné pro programování a proměnné v jazyku BASIC

Glossary

FUNCTIONS	pracují s číselnými hodnotami tak aby jim daly jinou hodnotu a jsou použity napsáním v názvu funkce následované argumentem.Názvy funkcí mohou mít v SAM BASIC jakoukoli délku.
GET	je způsob čtení klávesnice bez použití ENTER.Podobně u INKEY\$ kromě toho,že GET čeká na stisknutí klávesnice před pokračováním.Použije-li se řetězcová proměnná,bude GET pracovat jako písař takto: 10 DO:GET a\$:PRINT a\$:LOOP napíše názvy které jste mu zadali a může se pohybovat mezi vyšší a nižší polohou jak je to běžné.Je-li GET použito s číselnou proměnnou jako GET x,pak bude proměnná rovna 1 při stisknutí 1 až 9 při stisknutí 9.A nebo a má hodnotu 10,B nebo b má hodnotu 11 atd. Get je velice běžný v programech s menu řízením.Viz ON
GO SUB n	instruuje program jít na jakékoli číslo řádku do 65279
GO TO n	instruuje program jít na jakékoli číslo řádku do 65279.GOTO se může použít s LABEL např: GOTO blazes Viz LABEL
GRAB a\$,X,Y, w,L	skladuje zobrazovací prostor pro řetězec, který tam může být někde umístěn.Aby byl zobrazený prostor ohraničen,je definován "x" a "y" jako horním levým rohem,následován "w" pro jeho šířku a L pro délku v obraz.prvcích.Souřadnice x pro široké obraz.prvky se zaokrouhluje dolů na stejnou hodnotu a souřadnice y se zaokrouhluje nahoru na stejné číslo.Pro široké obraz.prvky v MODE 3 jsou hodnoty "x" a "y" zaokrouhleny na násobek 4.GRAB a PUT pracují pouze v MODE 3 a 4
Graphic coordinates	jsou normálně 173 nahoru až - 18 na horním okraji obrazovky.Dva řádky 9-znaků vysokých prvků se používají pro editovací oblast s bodem 0,0 nad nimi na levé straně obrazovky.Výběrem MODE 1 se mění výška znaků.

Glossary

	<p>na 8 obsahových prvků a vertikální y-osa běží od 175 do 16. Horizontální x-osa běží od 0 do 255, kromě MODE 3 kde jsou široké obsahové prvky z 0 do 511. Souřadnicový systém se změní jakmile přímo změníte CSIZE.</p>
HEX\$	<p>mění desítkovou soustavu na hexadecimální, založenou na číslicích 0 až 9 a písmenech A až F, např:</p> <p>HEX\$ 10 = "A", HEX\$ 32768 = "8000", HEX\$ 100000 = "0186A0"</p> <p>HEX čísla mohou být představována "Q", např:</p> <p>Q 0A = 10, Q 8000=32768</p>
Icon	<p>je malá grafická podoba předmětu, pojmu, techniky, nebo hlášení jako B symbolu použitého v návodu aby představoval "psaní následujících dat"</p>
IF	<p>je podmínka, která je kvalifikována THEN. "Dlouhého" IF, které pracuje v několika řádcích, je dosaženo vynecháním THEN a ohraničením IF pomocí END IF</p>
IN	<p>je funkce v jazyku BASIC, která pracuje chvíli jako PEEK. Je-li za ní adresa, čte IN vstupní/výstupní bod adresy a jeho výsledkem je zjištění bytů v tomto bodě. OUT pracuje podobným způsobem jako POKE.</p>
INK	<p>čtete oddíl nazvaný PEN</p>
INKEY\$	<p>je funkce, která čte klávesnici aby znala druh stisknuté klávesy.</p>
INPUT	<p>co je třeba napsat v průběhu INPUTu je ukázáno na dvou spodních řádcích obrazovky, ale mohou být použity jiné cesty, např:</p> <p>INPUT # 2; a\$ or INPUT #2; "number"; x</p> <p>povoluje využít horní část obrazovky</p>

Glossary

INSTR(n,a\$,b\$)	je funkce která vyhledává jeden řetězec a\$ do cílového řetězce b\$, začínajícího z bodu n a vrací se po jeho nalezení do své polohy. Jestliže nenajde cílový řetězec, pak odpoví nulou. INSTR(A\$,B\$) začíná v bodě 1 a symbol \$ můžete použít jako znaky do kterých můžete dosadit cokoli
INT	je zkratka pro část celých čísel a tato funkce převádí zlomky do celých čísel. Zlomek zaokrouhluje vždy dolů.
INV	je klávesa používaná ke zvýšení jasu obrazových znaků odzíranych z negativní obrazovky. Stisknutím [SYMB] [INV] uvedete obrazovku do jejího normálního stavu.
INVERSE	mění způsob znaků a na obrazovce se objeví grafika. INVERSE 1 převádí všechny nové znaky a grafiku do jejich negativní podoby a zaměňuje jejich barvu PAPER za barvu PEN. INVERSE 0 převádí vše zpět do normálního stavu.
I/O ports	je zkratka pro vstupní-výstupní body a tyto jsou použity procesorem pro komunikaci s klávesnicí, tiskárnami, jinými počítači a pod.
ITEM	je funkce, která dává informace o následující položce, jež má být načítána v DATA. ITEM má následující hodnoty a významy: 0 - v běžných DATA údajích není více ITEM 1 - následující ITEM je řetězec 2 - následující ITEM je číslo Je-li použito postupně může ITEM ukázat původ prvního ITEM v DATA listu, které přešlo do procesu.
K	je zkratka pro "kilo", které je nepřesně použito k označení 1,024 bytů paměti počítače. Tak např: 128K značí 131,072 bytů

Glossary

KEY posn,x	přikazuje KEY která je v určené pozici v klíčové aby měnila znak , který je stisknut. Polohy KEY musí být mezi 0 a 279 a x musí být mezi 0 a 255. Kódy 192 až 254 mohou být dány expanzí definic (viz DEF KEYCODE). Mapa klíčů je v dodatku.
KEYIN a\$	zapisuje řetězec tak (jakmile jste si ho zapsali) aby dovolil programům se napsat. Např: KEYIN "100 DATA " + a\$ #, "+b\$ bude přičítat nový řádek k programu po jeho spuštění.
Keyword	klíčové heslo je slovo, které může mít jednoduchý význam v angličtině, ale je rozpoznáno počítačem tak, aby mělo zvláštní význam a jednalo tak aby mohl být příkaz velmi přesně vykonán. Jakmile počítač rozpozná heslo, je to převedeno listováním v programu do velkých písmen.
LABEL	může být použito pouze na začátku řádku, ale může předcházet zabarvení kontrolních kódů nebo prostorů. Jakmile je jakákoli část programu spáštěna klávesou RUN, GOTO, GOSUB aj. proměnná tvořené jménem daným na jakémkoli LABEL v programu jsou jejich hodnoty nastaveny na číslo řádku, kde se nachází LABEL. Název pro LABEL musí být názvem legální číselné proměnné. Tyto proměnné se mažou při použití CLEAR nebo je-li program editován.
Language	je soubor příbuzných příkazů, které dovolují počítači provádět obvyklé úkony. Nejběžnějším jazykem je BASIC a zdokonalená verze je vestavěna do paměti SAM Coupé k co možná nejvyššímu programování.
LEN	je funkce k výpočtu délky řetězce
LENGTH(a, array name)	se používá k určení velikosti pole. n=1 odpovídá prvnímu rozměru. n=2 druhému rozměru.

Glossary

	<p>LENGTH se může také použít tak, aby našel umístěný řetězec nebo pole v paměti za použití např:</p> <p>LENGTH (0, A\$)</p> <p>které udává adresu prvního bytu v řetězci nebo oblast dat v poli. To je obvykle při vyvolávání umístění, když chcete měnit data. Pro číselné pole musíte za názvem použít závorky.</p> <p>LENGTH (1, x())</p>
LET	<p>se používá k přiřazení hodnot proměnným, takto:</p> <p>LET x=1</p> <p>SAM Coupé dovoluje dlouhé řetězce a názvy polí a několik příkazů LET může být napsáno najednou.</p>
Light pen	<p>Při zapojení do správné zdíčky na zadní straně počítače může být toto zařízení použito tak, aby přímo pracovalo s programem počítače dotekem na obrazovku. x a y souřadnice by měli být určeny použitím funkcí XPEN a YPEN</p>
LINE numbers	<p>všechny řádky vložené do programu počítače musí začínat svými čísly aby mohly být uskladněny a aktivovány v případě spuštění programu. Obvykle se řádky číslují po desíti pro případ, že by jste chtěli přidat zvláštní řádky mezi již zadané. Můžete seřadit přeházené číslování řádků vhodným přečíslovacím zařízením RENUM nebo automaticky a čísla řádků mohou být do 65279</p>
LIST, LLIST LIST n LIST n TO m	<p>listuje číslo určeného řádku</p> <p>listuje určenou část řádků. Je-li vynechán řádek č. 1, bere se v úvahu řádek první po 0. Je-li vynechán řádek č. 2 bere se v úvahu následující řádek.</p>
LIST FORMAT LLIST FORMAT	<p>vám dovolí udělat si "pěkné" listování s každým údajem, který je dán vašim řádkem a tyto řádky vysadí dále od kraje aby usnadnil čtení programu. Prvních 6 sloupců na obrazovce je rezervováno pro čísla řádku</p>

	<p>a mezery a program může mít až do 65279 řádků</p> <p>LIST FORMAT 0 udává normální listování</p> <p>LIST FORMAT 1 nebo 2 udává automaticky vysazené listování s jednou, nebo 2 mezerami jakmile je v programu poznáno určité heslo</p> <p>DEF PROC, DO a FOR vysadí všechny následující údaje přez jednu nebo 2 mezery dokud nejsou zrušeny, příkazem END PROC, LOOP nebo NEXT</p> <p>IF, ON ERROR a ON vysadí všechny údaje ve zbytku řádku o 1 nebo 2 mezery</p> <p>ELSE a EXIT IF zruší vysazení o 2 nebo 1 mezeru pouze pro běžné údaje.</p> <p>END IF ruší vysazování k IF</p>
LN	počítá přirozené logaritmy. Logaritmická funkce je inverzní exponenciální funkci
LOAD	<p>se používá k přenosu informací z kazety nebo disku do počítače. Při použití kazetového magnetofonu:</p> <p>LOAD " " zavádí první program nalezený počítačem za současného mazání staré paměti.</p> <p>LOAD "název" ukládá program určený názvem a pomíjí všechny ostatní na disku nebo pásce.</p> <p>LOAD "název" CODE zavádí byty od první adresy ve které byly uloženy přehrávají byty, které tam byly již zaznamenány</p> <p>LOAD "název" CODE start zavádí byty začínající na "startu", prvořadě startovací adrese, na které byly byty původně uloženy.</p> <p>LOAD "název" LINE číslo-zavádí program a jde automaticky na řádek zadaného číslem</p> <p>LOAD "název" DATA a(nebo a\$() mazou jakékoli určené pole, pak vyhledávají nové pole pro zavedení</p> <p>Budete-li si přát zavědět z jednoho z vnitřních pohonů disků, musíte nejdříve počítači sdělit druh pohonu za požitím příkazu:</p>

Glossary

	DEVICE d nebo DEVICE d1 pro disk drajv 1 DEVICE d2 nebo DEVICE D2 pro disk drajv 2
LOCAL	určuje seznam proměnných tak, aby byly umístěny do procesu. Jestliže chcete mohou zahrnovat pole.
LOOP	se používá ve spojení s DO, s LOOP WHILE a LOOP UNTIL pracuje přesně stejným způsobem jako DO WHILE a DO UNTIL.
LOOP IF	přinutí program vytáhnout z prostředka DO smyčku a opět pokračovat v údajích za LOOP, ovšem při pravdivé podmínce. Není-li podmínka pravdivá pak program pokračuje beze změny.
LPRINT	přikazuje počítači směřovat svoje výstupní informace namísto na obrazovku do tiskárny. Viz. kapitola 6
Machine code	je soubor instrukcí, které používá Z80B mikroprocesorový čip počítače SAM Coupé. Můžete psát programy přímo do něj v jazyce symbolických adres. Takové programy musí být kódovány do řady bytů používajících assembler, ale chcete-li je sami kódovat je mnoho knih, které vás to naučí.
Memory	SAM Coupé je vybaven 32K ROM (permanentní paměť) a 256K DRAM (dynamická paměť s přímým přístupem). Počítač má dvě vnitřní zdičky pro dodatečných 256 DRAM která je instalována uvnitř přístroje. Viz. kap. 10
MEM\$(n TO m)	přiděluje část paměti řetězci. Musí obsahovat obě n i m. n může být 0 a délka dílu paměti musí být méně než 65536. Velká část paměti může být změněna přemístěním řetězce do nové polohy. Použitím INSTR ji můžete nalézt.
Menus	pro počítač je tvořený jako menu v restauraci. Nabízí přehled přání na obrazovce, ze kterého můžete vybírat

Glossary

MERGE	<p>MERGE "název" přidává informace z pojmenovaného programu k informacím v paměti počítače, současně přepisuje jakékoli programové řádky nebo proměnné, které nejsou v souladu s novým programem.</p> <p>MERGE "název" CODE potlačuje samospuštění programů BASIC, které jsou již zavedeny. MERGE nemůže být použito jakýmkoli poli, které mohla být uložena.</p>
MGT plc	<p>je zkratka pro Miles Gordon Technology, snad nejlepší výrobce počítačů, kteří kdy působili v podniku ve Swansea Enterprise Park.</p>
MIDI	<p>je mezinárodní standart, podle kterého spolu hovoří hudební nástroje. Znamená číslíkový interface hudebních nástrojů. SAM Coupé dovede kontrolovat a komunikovat se všemi druhy hudebních zařízení přes jejich MIDI-vstupy a výstupy.</p>
MOD	<p>je matematická funkce, jejímž výsledkem je zatížení zbytkem:</p> <p>PRINT 110 MOD 60 bude končit v 50</p>
MODE n	<p>čistí obrazovku a volí číslo zobrazovacího módu</p>
Modem	<p>je zkratka pro modulátor demodulátor, zařízení pro přenos dat mezi počítačem a telefonními linkami</p>
Mouse	<p>je malé zařízení zapojené kabelem do správné zdířky na zadní straně počítače. Když se dostane přes mrtvý prostor může ovládat zobrazovací kurzor a při cvakání jejími klapkami volí, různá přání. Souřadnice myši x a y by měli být určeny použitím funkcí XMOUSE a YMOUSE.</p>
MOVE	<p>pracuje pouze s disky</p>
Nesting	<p>je proces umísťování jednoho nebo více programových celků uvnitř sebe</p>

Glossary

Network	Do 16 SAM Coupé se může napojit na síť, kanálem 0 jako rozhlasová stanice a kanály 1-15 jako stanice osobního telefonisty
NEW	zbavuje se všech starých programů a starých proměnných které jsou v paměti počítače. Použijete-li příkaz NEW, zapomene počítač vše, co jste ho naučili od jeho zapnutí.
NEXT	je zvláštní počítačový příkaz používaný ve smyčce. Vždy se používá ve spojení s příkazem FOR a mohli byste se podívat do sekce FOR abyste věděli jak pracuje
NOT	je obdobou AND, OR podmínek v údaji IF-THEN, kromě toho, kdy je plně nezrušitelný. Je-li poměr NOT správný, pak je poměrem chybným a naopak NOT nemůže být považován za funkci s nízkou prioritou, která nemusí mít závorky, ledaže by obsahovala AND nebo OR nebo obě.
ON	Vám umožní vykonat GOSUB nebo GOTO na jednotlivých číslech řádků v seznamu řady čísel, odpovídající hodnotě číselného výrazu nacházejícího se bezprostředně za ON
ON ERROR	zachycuje chyby ve vašem programování aby se mohl program vyrovnat bez zastavení a ohlašování chyby. To je známé jako vychytávání chyb a je detailně popsáno v kapitole 12
OPEN	je používáno k otevření proudu, který je normálně uzavřen
OPEN n	rezervuje navíc n 16ti kilových stránek paměti pro použití v BASICu. Nastavit vyšší RAMTOP umožňuje CLEAR.
OPEN SCREEN n,m	rezervuje paměť pro zobrazení, které je určeno počtem zobrazení a zobrazovacím módem. Počet zobrazení musí být mezi 2 a 16, a módy mohou být vybrány 1 až 4. Při zapnutí počítače je pouze jediné zobrazení (nazývané 1)

Glossary

	<p>OPEN SCREEN n,m,0 - pracuje stejně jako OPEN SCREEN n,m OPEN SCREEN n,m,1 otvírá běžné zobrazení tak, že jakmile se v paměti počítače nachází několik programů, mohou všechny po svém spuštění použít obrazovky.</p>
OR	<p>je podmínka která se může použít jako část podmínky po IF, WHILE, UNTIL atd. tak, že údaj IF bude pracovat když jeden vztah nebo druhý bude správný. Samozřejmě, že bude stále pracovat i když budou správné oba vztahy.</p>
Origin	<p>je termín používaný pro grafické zobrazovací souřadnice 0,0</p>
OUT p,x	<p>je výraz jako POKE a píše dané hodnoty do určeného místa s danou adresou. p musí být mezi 0 a 65535, x musí být mezi 0 a 255</p>
OVER	<p>je výraz pro kontrolu modelu grafických značek na obrazovce a ovlivňuje druh přetištění bodů, řádků a znaků. OVER 1 tuto schopnost zapíná, OVER 0 vypíná. V módech 1 a 2 jsou znaky přes sebe opakovaně přetištěny v pořadí znak-mezera-znak. V módech 3 a 4 je pořadí znak - barva - znak - obvyklou barvou je černá. Další efekty můžete vytvořit pomocí DRAW, CIRCLE, PLOT a PUT, kde OVER 2 udává výběr každého obsahového prvku obrazovky; OVER 3 udává přidávání za použití OVER 1 za PLOT, DRAW nebo CIRCLE; a pak opakování procesu; končit bude v původním zobrazení, které bylo znovu uloženo.</p>
PALETTE	<p>zastavuje všechny změny barev, které lze použít a nastavuje původní barvy opět na jejich počáteční hodnoty.</p>
PALETTE poloha, barva	<p>nastavuje polohu palety na zvolenou barvu. Paleta má 16 různých poloh (0 - 15) a 128 různých použitelných barev (0 - 127). Viz kapitola 5 obsahující všechny další příkazy PALETTE</p>

Glossary

PAPER n	následuje-li za ním zvolené číslo kódu, pak určuje barvu pozadí zobrazení na kterém jsou natisknuty jakékoli jednotlivé znaky. Také se může použít napsáním: PRINT PAPER n;" cokoli následuje"
Parameter	vztahuje se na celek obsažený v daném procesu, ale může se změnit v různé argumenty.
PAUSE n	umožňuje části programu trvat po přesně vymezený úsek času. Vaše obrazovka zobrazuje 50 pulsů za sekundu. Proto používání čísel počítaných v pulsnících k určení velice přesného načasování "zmrazení obrázku". Např: PAUSE 125 zmrazí vaši obrazovku po dobu 2.5 sekundy nebo dokud je stisknuta klávesa.
PEEK	je funkce, používaná k odhalení obsahu paměti počítače na adrese v rozsahu 0 až 528K
PEN	se používá k výběru barvy vašeho textu a grafiky na obrazovce. Hodnoty 0 - 15 jsou totožné se stejnými hodnotami příkazu PALETTE. Pokud budete chtít, můžete dát přednost použití příkazu INK namísto PEN, který bude dělat stejné operace
PI	je řecké písmeno používané k označení poměru kružnice a jejího obvodu. Ve skutečnosti je PI nekonečným periodickým zlomkem, ale můžete zkusit splést SAMe napsáním PRINT PI
pixel	znamená "obrazový prvek" a vztahuje se na jeden adresný bod zobrazení vytvořeného počítačem
PLOT x,y	je údaj používaný k nastavení počáteční pozice kreslení obrázků a grafů na vaší obrazovku. Umístěním prvního obrázkového prvku kamkoli na obrazovce začnete kreslení. To je dáno napsáním PLOT, následované souřadnicí x, která říká jak daleko má být od levé strany vaší obrazovky a y souřadnice udává vzdálenost od

Glossary

	spodního okraje obrazovky.
POINT(x,y)	udává barvu obrazového prvku v souřadnicích x,y.V MODE 1 a 2 to může být 0 nebo 1(ukazující)zde je obrazový prvek PAPER nebo PEN,v MODE 3 a 4 to může být od 0 do 15 a indikuje barvu prvku.
POKE	je nesprávný údaj, který se chová podle pravidel jazyka BASIC a ukládá číslo přímo do místa v paměti, kterou vyvoláme adresou.Pro paměť, kterou je vybaven SAM Coupé mohou být adresy od 0 do 528K.Vkládání v rozsahu 0 až 16384 není normálně účelné, protože v této části je umístěna ROM.
POP	odstraňuje číslo řádku ze skupiny GOSUB,PROC,DO.Je-li na POP proměnná, pak se vyřazené číslo řádku stane proměnnou, tedy můžete přeskočit rutiny GOSUB, dělat cykly a procesy bez záměny s neužitými čísly řádků. POP x určuje vyřazenému návrat čísla řádku v proměnné jako x
PRINT	je příkaz určující počítači vytisknout určené znaky, grafiku nebo výsledky na obrazovku.Viz kap.4 určená všem podrobnostem práce tohoto druhu
Printer	je přístroj, který dovede vytvořit obraz textu nebo grafiky na papír, známý jako trvalý záznam za současného připojení k příslušné zdířce na SAM Coupé.
Procedures	postupy, usnadňují programování.Použitím údajů DEF PROC, END PROC, DEFAULT, LOCAL a REF, mohou být vytvořeny programové moduly, které dělají určité úkony bez změny hlavního programu.Viz kap.4.
Program	je jednoduše soubor příkazů použitých k řízení počítače.
PUT x,y,a\$	umísťuje skladovací prostor na obrazovku, která již byla vymezena.Souřadnice x se zaokrouhluje nahoru na stejnou hodnotu

Glossary

PUT x,y,a\$,m\$	přikazuje využít druhého řetězce, který má stejnou délku jako první tak, aby pracoval jako grafická maska, která určuje jaké obrazové prvky budou použity z prvního řetězce. To umožňuje umístit složitý tvar na pozadí a jestliže se tak stalo, vystřihnout bez ohraničení. OVER není efektivní v tomto módu "masky", ale INVERSE v něm bude stále pracovat.
Radian	je úhel ležící pod arc, jehož délka je rovna poloměru kružnice
RAM	znamená paměť s přímým přístupem. Data uložená v RAM se ztratí při stisknutí RESET nebo při vypnutí počítače.
RAMTOP	je funkce, která Vám ukáže nejvyšší adresu v počítačové paměti s přímým přístupem, která se dá použít jazykem BASIC
RANDOMIZE	používá "mateřské" číslo k výpočtu "náhodných" čísel. RANDOMIZE nebo RANDOMIZE 0 nastavuje toto na číslo rámečků, generovaných od zapnutí počítače. RANDOMIZE n nastavuje tuto operaci na specifikované číslo.
READ	je výraz za kterým následuje seznam názvů proměnných oddělených čárkami. Pracuje jako INPUT, ale ušetří Vám starost s psaním hodnot proměnných jejich vyhledáváním v údajích DATA. Můžete také použít READ LINE pro řetězcové proměnné, které dovoluují pominout citaci z údajů DATA
RECORD STOP	vypíná grafický záznam. Tento příkaz pracuje ve všech módech
RECORD TO a\$	umožňuje nahrát grafické příkazy jako STRING, tak ty které mohou být provedeny mnohem rychleji při použití příkazu BLITZ. Při zapnutí RECORD se budou nahrávat následující příkazy: CIRCLE, CLS, DRAW, DRAW TO, OVER, PAUSE?PEN?PLOT, ale ne PLOT PEN

Glossary	
REM	<p>je jednoduše připomínkovým údajem v programu, zasazený v něm proto, aby se použil nikoli počítačem. Za příkazem může následovat cokoli a počítač to nebude brát v úvahu. Můžete jej použít jako název souboru informací takto:</p> <p style="padding-left: 40px;">10 REM pravidelný postup příjmu daní</p>
RENUM	<p>automaticky přečíslovává celý program v krocích po 10, začíná řádkem číslo 10.</p>
RENUM n TO m	<p>automaticky přečíslovává část programu počínaje řádkem n a končí řádkem m. Je-li vynecháno n, bere se v úvahu první řádek, je-li vynecháno m končí přečíslování u řádku posledního</p>
RENUM STEP s	<p>vytváří čísla nových řádků v krocích po s, je-li dost místa. STEP, LINE a "kráječ" programu nemusí být použity, ale musí být za sebou v tomto pořadí</p> <p style="padding-left: 40px;">RENUM n TO m LINE l STEP s</p>
<p>tlačítko RESET</p>	<p>při jeho stisknutí na zadní straně přístroje se provede obnovení základní jednotky CPU paměti, řízení floppy disku, zápisu stránek paměti a zápisu MIDI-IN, MIDI-OUT a BORDERu. Neobnovuje barvu vyhledávací tabulky a zapísování zvukového čipu nebo přerušovače řádků LINE INT.</p>
RESTORE n	<p>opakovaně ukládá ukazatel DATA do prvního údaje DATA v řádku n nebo v řádku s nejbližším číslem k n. Bude-li vynecháno číslo, číslo bude RESTORE 0, tj. nastaví na první údaj DATA pokud bude nějaký existovat. K RESTORE dojde vždy při zavěšení BASIC programu</p>
RETURN	<p>je klapka používaná k odevzdání řádku příkazu do programu. Technicky RETURN omezuje podprogram RETURN adresa udává řádek a údaj, do kterého se přeskočí po vykonání GOSUB</p>
RGB	<p>je zkratka pro soustavu barevných bodů, které vytvářejí jakoukoli barvu na vaší obrazovce, kombinací červené/red/ zelené/green/ a modré /blue/</p>

Glossary

ROLL d,p	pohybuje obrázkem po vaší obrazovce a otáčí jím dokola. Pohyb je řízen příkazem d / 1=vlevo, 2=nahoru 3=vpravo, 4=dolu/, za kterým následuje počet pohybujících se prvků p.p může být 1,2,4,6 atd. ale ne 3,5, 7 atd.
ROLL d,p,x,y, w,L	pohybuje pouze tou částí obrazovky, která je určena souřadnicemi x,y označujícími levý horní roh. Za nimi následuje w-šířka obraz.prvků a L-jejich délka. x je udáno v sudých číslech, také w je udáváno v sudých číslech. Příkaz ROLL pracuje pouze v MODE 3 a 4
ROM	znamená permanentní paměť a je nevyjmutelnou částí mozku počítače, která byla udělena v továrně
RND	generuje náhodná čísla počínaje bodem v pevném pořadí 65536 čísel. Viz kap.4
RUN	přikazuje programu se spustit. Ke spuštění programu z jakékoli určeného řádku se využije RUN n.
SAVE "název"	pojmenuje program a přikazuje programu poslat kopii tohoto programu ze své paměti při vytváření jeho trvalého záznamu
SAVE"název" LINE číslo	je hezčí způsob, při kterém se program dostane na určený řádek po svém opětném zavedení do počítače.
SAVE"název" CODE adr,délka	nahrává informace bez jakýkoliv vztahů, na které jsou tyto informace zvyklé. První číslo /počáteční adresa/ musí být adresou první jednotky paměti, známou jako byte. Druhé číslo /délka/ je počtem uložených bytů. Adresa je číslo mezi 0 a 528K, délka může být od 1 bytu do 512K.
	<u>SAVE"název"CODE poč.adresa,délka,adresa provedení,</u> uloží automatické spuštění souboru kódů
	<u>SAVE"název"SCREENS</u> ukládá zobrazení s jeho Palettou a módovou informací
	<u>SAVE"název"DATA a\$()</u> ukládá pole nebo řetězce určené jeho vlastním názvem

Glossary

	<p>Úvozovky se mohou zvolit/nejsou povinné/pro řetězce nebo řetězcová pole</p> <p><u>SAVE řetězec LINE číslo</u> řídí počítači aby nahrál program tak, že při svém zavedení skočí na dané číslo řádku a sám se spustí.</p> <p><u>SCREEN dump</u> je metoda reprodukce zobrazení z obrazovky do tiskárny</p>
SCREEN n	<p>volí zobrazení na něž lze použít příkazy BASIC.N musí být mezi 1 a 16 obrazovka musí být nejprve otevřena. Takže: OPEN SCREEN 2;:SCREEN 2</p> <p>uvádí v činnost příkazy jako PRINT,LIST,PLOT a DRAW a druhé zobrazení nechává zobrazení 1 celé dokud se toto zobrazení opět neznázorní.Viz také DISPLAY</p>
SCREEN\$(r,c)	<p>je funkce která pracuje v kterémkoli móde a identifikuje uživatelem definovanou grafiku v rozsahu CHR\$(32 až 168,jakož i normální znaky.Nerozpozná blokovou grafiku a nedělá rozdíl v tom,je-li určeno BLOCKS 0 nebo 1.</p>
SCROLL d,p	<p>pracuje přesně stejně jako ROLL d,p kromě toho,že nebalí výsledek.Pro vyplnění pozadí znovu dosazuje vypočítavaná DATA.Příkazy SCROLL pracují jen v mode 3 a 4</p>
SCROLL d,p,x,y,w,L	<p>pracuje stejným způsobem jako ROLL d,p,x,y,w,L,ale nebalí výsledek</p>
SCROLL CLEAR	<p>zbavuje "SCROLL" schopnosti působit na zobrazení,takže když je zobrazení celé,bude přetočeno bez jakékoli správy nebo bude třeba stisknout klávesu.</p>
SCROLL RESTORE SGN	<p>zapíná "Scroll?" ihned zpět</p> <p>je znaménková funkce, které se někdy říká "signum". Takže když budete chtít znát výsledek argumentu,SGN udá +1 při kladném argumentu,0 pro argument=0, a -1 při záporném argumentu</p>

Glossary

SHIFT .	při stisknutí současně s jinými klávesami se SHIFT používá k psaní velkých písmen a alternativních znaků.
SIN	je trigonometrické funkce počítající sin
Slicers	jsou určené řetězce nebo podřetězce. Slicers může být číselným výrazem, zvoleným číselným výrazem do jiného zvoleného výrazu, nebo může být prázdný
Software	je seznam příkazů užívaných k řízení počítače, které jsou uloženy a pak zavedeny z kazety nebo z disku
SOUND r,d	posílá data byt d registru zvukového čipu. Může se použít až 127 dvojic čísel, jako: SOUND r,d;r,d;r,d: atd
Space	dlouhá SPACE čára na klapce klávesnice pracuje jako jakýkoli konvenční písař nebo textový procesor
SQR	přikazuje počítači počítat druhou odmocninu čísla n. Výsledkem je číslo, které když umocníte dostanete n.
Stack	/zásobník/ nařizuje počtu datových prvků uložit se tak aby je bylo možno použít k účelu, určenému programem počítače, tak jako zásobník GOSUB nebo kalkulátrový zásobník
Statement	obsahuje individuální příkazy v programovém řádku. Několika násobné údaje v jednom řádku jsou odděleny čárkami.
STEP n	je používán v cyklu FOR-NEXT k určení řídicí proměnné, která je za normálních okolností vždy max. 1, takto: FOR řídicí proměnná=první hodnota TO poslední hodnota STEP n STEP nemusí být kladné nebo celé číslo, např: FOR x=100 TO 1 STEP -1.5
STOP	zastavuje program a zobrazuje STOP kód na obrazovce. Je-li program opět odstartován pomocí CONTINUE pokračuje následným údajem.

Glossary	
Strings	/řetězec/ teoreticky mohou být 65520 znaků dlouhé a jejich legální názvy mohou být max.10 znaků dlouhé.
STRING\$(n;a\$)	opakuje řetězec tolikrát kolikrát zvolíte
STR\$	převádí čísla do řetězců: Funkce končí symbolem \$, který ukazuje, že výsledkem je řetězec: Viz VAL
Subroutine	je blok číslovaných údajů, které provádějí určené ukony a mohou být mnohokrát vyvolány údajem GOSUB
SVAR n	udává adresu systému proměnné n
SYMB	při stisknutí umožňuje přístup k alternativním znakům a operacím na klávesnici
Syntax error	v BASICu je obdobou chybování, v anglickém jazyku při porušení pravidel. Počítač Vám nepromine když uděláte chybu při použití interpunkčních znamének nebo chybne uvedete klíčové heslo! Jisté chyby jako GOSUB a RANDOMISE, budou automaticky opraveny. Nejsnazší metodou psaní programu je napsat malými písmeny a poznamenat si klíčové hesla psané velkými písmeny tak, jak jsou umístěna ve svých řádcích.
System variables	/soustava proměnných / řídí všechny druhy operací a každá má svůj vlastní počet SVAR. Nejvíce používané proměnné jsou vypsány na konci návodu. V technickém návodu SAM Coup= jich je více jak 100.
TAB	kvalifikuje /určuje/ PRINT. Vysazuje text a nastavuje sloupce textu nebo číslovek do pravidelných sloupců.
TAN	je trigonometrická funkce počítající tangent
TO	se používá k nastavení hranic cyklu FOR-NEXT
TRUNC\$ a\$	udává a\$ u prázdnými prostory na konci. To je užitečné při vyjímání data z řetězcových polí, takže normálně: DIM a\$(10,10) bude končit v "sam" "natisknutý, zatímco můžete užít: PRINT TRUNC\$(1) který končí v "sam"

Glossary

UDG " "	udává adresu jakéhokoli uživatelem definovaného grafického znaku od CHR\$ 32 do 168. Rozpětí CHR\$ 169 až 255 je vhodné pro pokročilé uživatele .
UNTIL	je podmínka používaná s DO a LOOP. Zaměřte se prosím na tyto příkazy
USR n	Je-li číselný argument startovací adresou USR spouští strojový kód a uznává hodnoty od 0 do 524287
VAL	je silnou funkcí, která mění řetězce zpět na čísla. Řetězec může být jakýmkoli číselným výrazem. VAL odstraňuje z řetězce uvozovky, určuje hodnotu argumentů a nakonec určí, hodnotu všeho, co zůstalo jako číslo.
VAL\$	je podobná jako VAL, s argumentem, ale udává výsledek, který je opět řetězec
variable	jsou 4 typy proměnných, kterým jsou dány názvy aby určily nějaké hodnoty v programech. Numeric variable - jejich názvy musí začínat písmenem a následující čísla, písmena, podtrhávání a mezery do délky 32 znaků. Mezery se nepočítají jako znaky String variables/řetězcové proměnné/ a array variables /proměnné pole/ jsou obdobné, ale nemohou být delší než 10 znaků a musí za nimi následovat \$
VDU	je zkratka pro obrazovkový terminál. Hlavně poskytuje zajištění obrazovky pro zobrazování počítačem generovaných symbolů.
VERIFY "název"	kontroluje zda informace na pásku nebo disku odpovídají instrukcím se stejným názvem v paměti počítače.
VERIFY"název" CODE	kontroluje byty na pásku nebo disku počínaje adresou, ze které byl uložen první byt.
VERIFY"název" CODE počátek	kontroluje byty na pásku nebo disku vůči těm v paměti počínaje adresou danou číslem

Glossary	
VERIFY "název" CODE začátek délka	kontroluje byty na pásce nebo disku vůči bytům v paměti počínaje adresou danou číslem. Je-li rejstřík, na pásku nebo disku delší než je uvedená délka, bude dána zpráva o chybě při zavádění
VERIFY "název" DATA	porovnává pojmenování pole s pásku nebo disku s polem uloženým v paměti počítače
WHILE	je podmínka používaná s příkazy DO a LOOP. Viz jejich popis
WINDOW levý, pravý, horní, dolní, okraj	nastavuje textové okénko. Hranice jsou určeny zadáním který levý sloupec použít /0 a více/, za ním následuje pravý /max. 84 ve 3 modu se znaky širokými 6 obrazových prvků/, následuje určení horního okraje /0 a více/ a čtvrtým číslem je určen spodní okraj okénka /WINDOW/. Jestliže zadané hranice nemohou být v běžném modu nebo neodpovídají velikosti znaku, tak se objeví oznámení "Invalid WINDOW"
x-axis, y-axis	/osy/ jsou jmenovitě určené čáry používané v trigonometrii, omezují kružnici od strany ke straně a od vrchu ke spodu
x-coordinate y-coordinate	souřadnice určují bod na obrazovce. X je vzdálenost od levé strany obrazovky a Y je vzdálenost od spodního okraje obrazovky.
XMOUSE, YMOUSE	viz MOUSE
XOS, XRG, YOS, YRG	jsou speciální funkce, které řídí soustavu grafické změny měřítka. Termíny měřítka a originál. Viz. kap. 6
XPEN, YPEN	viz PEN
	Následující znaky na straně 167 originál manuálu mají své určené významy při jejich použití v SAM Coupé. # může se použít k uvedení proudového nebo kanálového čísla, nebo jako znamení univerzálnosti. Někdy se vztahuje na pseudopočet, nebo číslo.

S A M C o u p é

Appendix

- hlavní údaje
- hlavní spoje
- pinové spoje
- systémové proměnné
- předefinované klíče
- klavesnice
- kontrolní kódy

Appendix

MILES GORDON TECHNOLOGY , SAM Coupé

Pohon - Z 80B mikroprocesor s frekvencí 6 MHz

Kontrola - VLSI 10 000 ASIC chip

ROM - 32K včetně SAM BASICu, BIOS, samostatný disk

RAM - 256K vstoupně až do 512K / 256Kx4 100nS DRAM

ZVUK - Philips SAA 1099 Syntetizátor; 6 kanálů
8 oktáv, stereo s obalovou a amplitudovou kontrolou, plus vlnový svuk

GRAFIKA - Motorola 1377P video chip. ASIC slouží jako grafický procesor, ve všech modech umožňuje 128 barev zobrazit

MODE 1: 32x24 znakových buněk na obrazovku
každá buňka s dvoubarevnou schopností, 16 barev vybiratelných ze 128, Spectrum je kompatibilní

MODE 2: jako MODE 1, ale 32x192 buněk, každá buňka s dvou barevnou schopností, 16 barev vybiratelných ze 128

MODE 3: až do 85 sloupců textu, 512x192 pixelů
každý s vybiratelnou barvou, 4 barvy vybiratelné pro řádek ze 128

MODE 4: 256x192 pixelů na obrazovku, každý s vybiratelnou barvou, 16 barev vybiratelných v řádku ze všech 128

INTERFACE- UHF TV kanál 36, přes hnací jednotku, vše přes SCART. Atari standartní joystick/pro dva s rozdvojeným kabelem/Myš, Coupé standart Světelná pero, světelná sbraň, coupé standart. Domácí kasetový rekordér. MIDI IN, MIDI OUT, MIDI přes software zapínat. Síť přes rozšifující port. 64 pinový rozšifující port pro další periferie.

DISK DRIVE- 1 nebo 2 vybiratelné a vnitřně umístěné 3.5" ultra tenký Citizen drive, 1Mb neformátovaná paměť, 780K formátovaná

DC pohon - Přikon 4.75 až 5.25 voltů

Spotřeba - 11.2 Wattů

Háras - do 3 G při práci jinak 60 G

Appendix

Vibrace - Při práci 5 až 500Hz/0.5G, když nepracuje
5 až 500 Hz/2G
Prostředí - Okolní teplota od 5 do 45 st.Celsia při
práci, mimo -20 až 50 st.Celsia
Vlhkost - Max.relativní vlhkosti při 29.4 st.Celsia,
nulová kondenzace
Spolehlivost-MTBF:10000 POH.MTTR:30 Mins.Životnost
komponentů > 5 roků
Váha - 2.26 kg

Na straně 171 dále originál manuálu jsou schema zapo-
jení výstupů počítače.

**Předdefinované
klíče**

Na straně 178 až 179 jsou znázorněné předdefinované
klávesy při stlačení jednotlivých kláves.

Čísla v závorkách jsou kódy znaků. Znaky s kódy od 128
nahoru mají vícero interpretací, v závislosti od pou-
žití. BLOCKS 0, nebo BLOCKS 1, nebo když jsou v úvozov-
kách nebo ne. Např. SYMB L dává klíčové slovo LET
když je mimo úvozovek, nebo když je v úvozovkách.

SYMB RETURN spojuje klíče F0 až F9 aby deli čí-
sla.

CNTRL P dává PAPERu kontrolní kód a CNTRL I dává
PENu kontrolní kód:

Stiskněte 0 až 7 a vyberte číslo barvy.

CNTRL B dává BRIGHTu kontrolní kód: stlačte 0 až 1
aby se zintenzívnil PEN a PAPER v rozsahu od 0 do 7
a nebo od 8 do 15. Běžně se některé znaky zjeví na
několika místech klávesnice. Slouží to uživatelům,
kteří mají různé klávesnice.

Expanzní konektor

Standartní 64 - kolíková Eurokonektorová zástrčka s řadami A-C. Všimněte si, že řada A je dole a řada C nahoře na konektoru.

Je více než 600 bytů systémových proměnných, které jsou vhodné pro zkušené programátory. Celý seznam je obsažen v technickém návodu SAM Coupé přístupný od MGT. Následující seznam je většinou běžných systémových proměnných, které se mohou nahrazovat, např.:

POKE SVAR 8,10 nebo
POKE SVAR 1, "LU"

Toto se může přezkoušet např. takto:

PRINT PEEK SVAR 8

NÁZEV	SVAR ČÍSLO	
LNCUR	0	znak kurzoru na běžném programovém řádku/normálně ">" /
LCCUR	1	kurzorový znak při vypnutém CAPS LOCKU
UCCUR	2	znak kurzoru při zapnutém CAPS LOCKU
BIN1DIG	3	znak používaný BIN\$em AS'1'/běžně "1"/
BIN0DIG	4	znak používaný BIN\$ AS"0"/běžně "0"/
INSTHASH	5	znak používaný INSTREM jako univerzální znak / " # "/
SLDEV	6	běžné písmeno jako značka /obvykle "T" pro páskový tape systém/
SLNUM	7	běžná záznamová rychlost pásku nebo opomenutí čísla drajvu když je pohon disku v činnosti
SPEEDINK	8	rychlost probleskování barev/viz PALETTE/
PRRHS	14	pravý textový sloupec pro tiskárnu/žádaná max.délka řádku je větší než 1/
AFTERCR	15	znakový kód odeslán tiskárně po CHR\$ 13 jestliže kanál "P" je v činnosti./běžně 10 aby automaticky podával řádky/. POKE SVAR 15,0 aby zastavil jakýkoli odeslaný znak.

DMPYL	16	/2byty/ adresa v levém horním rohu obrazovky pro grafické výpisy
DMPLEN	18	délka grafického výpisu
DMPWID	19	šířka grafického výpisu/8 širokoprvkových celků, běžně 32/
DMPWM	20	násobitel šířky graf.výpisu/1=normál, 2 nebo 3 znamená dvojitá nebo trojitá šířka/
DMPHM	21	násobitel výšky graf.výpisu/1=normál, ne 1=dvojitá výška/
GCM1	22	/9bytů/počáteční zpráva odeslaná tiskárnám před výpisem
GCM2	31	/8 bytů/zpráva odeslaná před každou vypsanou řadou
GCM3	39	/7 bytů/konečná zpráva odeslaná tiskárnám po výpisu
TABVAR	47	"0" kdyby čára tiskárny tabulkovala po 16-ti sloupcích, ne "0", kdyby bylo použito tabulkování po 8-mi sloupcích.
SOFE	50	vypíná zapnutí/vypnutí výpisu na obrazovku, běžně 0-zapnuto, obrazovka začne černat, jestliže nepoužijete klávesnice po dobu 22 minut. POKE SVAR 50, 1 odstraňuje tuto vlastnost
TPROMPTS	51	BIT 0=1 k potlačení výtisku rejstříku jmen během zavádění; BIT 1=1 k potlačení vlivu během ukládání na pásek
BGFLG	52	Blokuje grafický příznak. 0 jestliže blokuje +, ne 0 jestliže blokuje 0
FL6ORS	53	"0" jestliže Mód 3 používá šestiprvkovou šíři znaku, jinak ne "0"
CSIZEH	54	výška znaku
CSIZEW	55	šířka znaku
UWRHS	56	pravý sloupec hořejšího okénka
UWLHS	57	levý sloupec hořejšího okénka

UWTOP	58	vrchní řada hořejšího okénka
UWBOT	59	spodní řada hořejšího okénka
LWRHS	60	pravý sloupec spodnějšího okénka
LWLHS	61	levý sloupec spodnějšího okénka
LWTOP	62	vrchní řada spodnějšího okénka
LWBOT	63	spodní řada spodnějšího okénka
MODE	64	zobrazovací mód / minus 1 /
YCOORD	65	y - souřadnice běžné grafiky, na vrcholu 0
XCOORD	66	/2byty/ x - souřadnice běžné grafiky
LASTH	513	poslední klávesa stisknuta
KBHEAD	520	klávesa na vrcholu vyrovnávací paměti klávesnice
REPDEL	521	prodlení před automat.opakováním klávesami
REPSPD	522	rychlost, kterou klávesy opakují
CHARS	566	/2byty/adresují 256 bytů pod sadu hlavních znaků
ERRSOUND	568	délka zvuku chyby
CLICK	569	délka cvaknutí klávesnice
KPFLG	615	funkční klávesy při sudém čísle, číselný doplněk je-li liché
KLFLAG	618	8 při zapnutém CAPS LOCK, jinak 0
FRAMES	632	/3byty/ TV páleníaky od zapnutí počítače
UDG	635	/2byty/ adresa znakového modelu pro CHR\$ 144
HUDG	637	/2byty/ adresa znakového modelu pro CHR\$ 169

