

STOS

Maestro



USER GUIDE

STOS *Maestro*

Jon Wheatman
Robert Shettle
Richard Vanner
Chris Payne
Dave McLachlan
Ian Waugh/Stephen Hill
Rozi Morris
Rob Morris

STOS Maestro programmer
STOS Maestro Hardware/Graphics
Project manager
Marketing manager
Graphics
Manual authors
Voice samples
Musical soundtrack

If you have any difficulty with this product, please write to:

Mandarin Software,
Europa House, Adlington Park,
Adlington, Macclesfield SK10 4NP

No material may be reproduced in whole or in part without written permission. While every care has been taken, the publishers cannot be held legally responsible for any errors or omissions in the manual or the software.

STOS is a registered trademark of Jawx and Mandarin Software.

Contents

Introduction.....	1
Connecting the STOS maestro plus cartridge to an ST.....	1
Making a backup.....	2
What is a sampler?.....	3
Chapter 1: The sound sampler editor.....	5
Loading and Playing a sample.....	5
The main screen.....	6
Spectrum analysers and pointers.....	6
Sample display area.....	7
Play and record area.....	7
The menu bar.....	10
Cut and paste.....	10
Disc options.....	11
FX menu.....	13
Special options.....	14
Creating your own welcome sample.....	15
Chapter 2: Disco and Phaser programs.....	16
STOS Maestro Disco unit.....	16
STOS Phaser program.....	17
Chapter 3: STOS Basic Interface.....	17
Installing the new STOS Maestro extension file.....	18
Maestro accessory.....	18
Accessory tutorial.....	19
Drum machine.....	20
Chapter 4: STOS Maestro commands.....	21
Playback commands.....	21
Sampler control commands.....	29
Chapter 5: Creating your own sampler in STOS Basic.....	30
Initialisation.....	30
Setting the sample speed.....	31
Recording a sample.....	31
Playing a sample.....	32
through the loudspeaker.....	32
Chapter 6: Technical appendix.....	33
Adding sampled sounds to an existing STOS Basic program.....	35
Assembly language interface.....	36
STOS Maestro Plus cartridge specifications.....	38

Introduction

Congratulations for buying STOS Maestro, an exciting new utility which, together with the STOS Compiler, continues Mandarin's commitment to support STOS Basic.

STOS Maestro allows you to incorporate "real" sounds of professional quality – speech, music and special effects – into your STOS Basic programs with the addition of only a few lines of code. Not only will it enhance the style and quality of your games and programs but it's also a real fun package to use.

The package is available in two formats: STOS Maestro and STOS Maestro Plus. The only difference is that the latter package includes the professionally-designed sampler cartridge.

With the STOS Maestro Plus sampler cartridge you can record your own sounds as easily as you would play a record or cassette and the whole world of sound is open to you. To get you started, the STOS Maestro package contains over 50 superb sampled sounds, along with the two discs – one containing the programs, the other containing the samples. You will also find a registration card and details about the STOS Maestro Plus sampler cartridge and how to order it – if you don't already have one.

Connecting the STOS maestro plus sampler cartridge to an ST

If your STOS Maestro set up includes the sampler cartridge you must follow the steps below. They detail the proper use of the cartridge.

- 1 Before inserting or removing the cartridge turn the ST off. Otherwise the cartridge or even the computer could develop a fault.
- 2 Insert the cartridge into the ST's cartridge port (located on the left side of the system), taking care to ensure that the STOS Maestro sticker, is facing up and towards you.
- 3 Now insert the STOS Maestro programs disc into drive A and turn on the ST.

External lead requirements

The cartridge links the ST to the outside world of sound. You must now connect an external sound source to the cartridge.

We have supplied a female phono port to which you must connect an external lead. For best results take the output from the headphones port of your external source (hi-fi, tape recorder, walkman, musical keyboard). This output can be controlled via the volume control. If you directly transfer the sound from, say, a CD by phono to phono then you'll probably find the sound output is too high and won't be controllable. If you don't have such a lead then any electronic shop will be able to supply you with one – if you're unsure, take the cartridge with you and explain your requirements to the shopkeeper.

Synthesiser sounds

5th String violin
Clockwork
Easy synth
Fullrank
Moonsynth
Orchestra
Piano brass
Pitch bend
Spacie sound
Synth Bass
Weird sound
Yeah sound

Musical Instruments

Acoustic guitar
Electric guitar
Gong
Horn
Orchestral stab
Panpipe
Piano
Saxophone
Strings
Tubular bells

Voices

Bonus points
EEEEERRRRR...
Level
One, Two, Three Four, Five
Game over
H-e-l-p!
Get ready

Sound effects

Boing
Bounce
Computer bleep
Electric door
Explosion
Helicopter rotor blades
Laugh
Lazer gun
Machine gun
Metal bar sound
Pistol shot
Siren
Smashing
Spring effect
Swoosh
Zangy

Making a backup

STOS Maestro is not copy-protected which means you can make backups or load it onto a hard drive if you have one. But please don't give copies to other people. STOS Maestro took many months to produce and it is only by continuing sales that we will be able to develop new extension discs and accessory programs.

The two STOS Maestro discs have been specially formatted so as to fit the extra music demos on. This means that side two of each disc cannot be copied using the standard Gem Desktop backup procedure. Don't worry about this because copying the discs as single format will backup the other files – it's only the DEMO.TOS files that can't be re-duplicated. If you have a special copying utility such as PROCOPY then you'll be able to back the discs up fully. The following procedure shows how you should backup the discs single sided.

- 1 Slide the write protect tab of the original disc so that you can see through the hole. This will guard your disc against possible mistakes during copying.
- 2 Place a blank disc into drive A and format it single sided.
- 3 Now put the STOS Maestro program disc into drive A and drag the drive A icon on to the drive B icon.

4 Follow the prompts displayed in the Gem dialogue boxes.

5 Repeat steps 1 to 4 for the STOS Maestro samples disc.

You will now be left with two backup discs, but the DEMO.TOS files won't work

As you read through the manual and experiment with the samples, you may well create some new sounds which you really like. Keep a spare formatted disc to hand for storing your samples but don't worry if you forget – the STOS Maestro editor has a built-in disc formatter.

What is a Sampler?

You probably know that sampled sounds are extremely realistic – they are, in fact “real” sounds – and most ST games make a feature of the life-like music and sound effects they contain. But how does the sound get inside the computer in the first place?

Here's the theory behind sampling – but don't worry, there's not much of it and it won't hurt a bit. If you know how sampling works you'll get much more out of the program.

If you really don't want to know what's going on you can memorise the following simple rule and skip ahead to Chapter 1: The higher the sampling rate the better the sound quality but the more memory it will take.

Simply, sampling is the process of converting analogue sound into digital information, storing it in the computer and then converting it back to analogue sound again so we can hear it. The routines required to do this are known, naturally enough, as analogue-to-digital and digital-to-analogue converters or A-to-D and D-to-A for short. These are terms you will come across whenever samplers are mentioned.

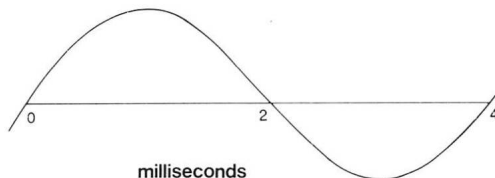
STOS Maestro does all this conversion work for you automatically so you don't have to worry about it at all. But it will help if you understand a little about the mechanics of the conversion process.

If you look at sound waves on an oscilloscope you'll see that they consist of lots of smooth curves and continuously varying lines. Computers, as you probably know, aren't very good with curves. At their lowest level they can only store the digits 0 and 1 and when they draw a curve on the monitor you'll notice that it's not a continuous unbroken line but a series of steps, generally pixel size.

So how can we represent a continuously varying value (sound waves) as numbers? The solution is to take samples of the sound wave at regular intervals and store the reading as numbers.

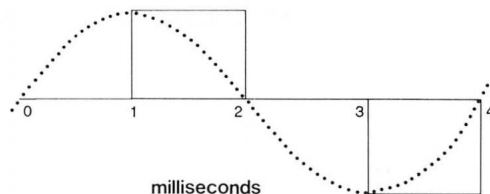
Let's put this little bit of theory into practice and see what happens when we sample a sine wave, the purest sound in the universe. It's a little like a plain, flat whistle or flute sound and does not contain any “other noises” (the technical term for these is harmonics) which make a sound bright and raspy like brass, thin and reedy like an oboe or hollow like a clarinet (sine in Latin means “without”).

If you feed a sine wave into an oscilloscope the display will appear something like this:

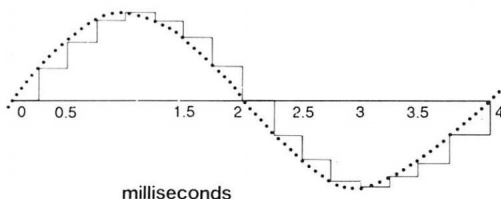


It tells us that the entire sine wave lasted only four milliseconds (four thousandths of a second). Or to put it another way, 250 of these sine waves will occur every second. The actual pitch produced by this will be an out-of-tune B below middle C, but we'll use this value because it makes things simpler if we stick to round numbers.

How many samples do we need to take to convert this to digital form? Let's try taking a sample once every thousandth of a second. We'd get something like this:



It's a nice try but it's not a very good representation of a sine wave, is it? Let's increase the sample rate to four thousandths of a second:



That's a bit more like it, although it's not quite perfect.

To avoid having to talk in terribly small numbers like so-many thousandths of a second, we use the term kiloHertz which is abbreviated to KHz. So we can say our first digital conversion was sampled at 1KHz and the second at 4KHz.

The sampling rate in STOS Maestro can be varied from 5KHz to 32KHz. You can see from the last diagram that we were getting pretty close to a sine wave with a sample rate of 4KHz. At 32KHz our digital representation of the wave will be pretty close to the real thing.

Great! So you could sample everything at 32KHz. Well you could but I've got some good news and bad news for you.

As we've seen, the more samples taken in a given time, the more accurate the digital model will be and consequently the better the sound quality will be on playback. that's the good news.

However the more samples you take in a given time, the more data you will accumulate, the more memory you will require and the more jerkily your STOS sprites will move because so much processor time is being taken up. That's the bad news.

You will find that some samples will sound fine at low sample rates but others will lose some of their brightness and "edge". Here's the bottom line: Sampling is a trade off between fidelity and memory used.

But here's some more good news. In order to convert digital data into sound, samplers use a conversion table and the quality of the sample depends upon the precision and accuracy of this table. Maestro incorporates a super quiet, high precision conversion table and you'll find that many samples can be recorded and played back at quite low sampling rates and still sound good.

And here's even more good news – STOS Maestro employs a technique called oversampling which enables samples to be taken at low sample rates with greater than average accuracy. It also features a unique pack facility which lets you squeeze more sample into less memory. Oversampling and pack are fully explained in Chapter 1.

Okay, that's enough theory, now let's get sampling.

1: The Sound Sampler Editor

Quick Start Guide

By now you'll want to see – and hear – what STOS Maestro can do so we'll run through a few basics to get you up and running.

***WARNING:** If you are going to create samples of your own with the STOS Maestro Plus sampler cartridge, make sure your Atari ST is switched off before plugging the cartridge into the cartridge port.*

Insert your backup program disc into the disc drive and turn on your computer. Double-click on the Drive A icon and a window will open. Double-click on STOSM.PRG

When the loading screen appears, press the 1 key on the ST's keyboard to load the sampler program. Alternatively the sampler can be loaded by double clicking on the MAESTRO.PRG file, which will save you 40k of memory. The same applies to the Disco and Phaser programs described later.

Upon loading the display will change to a screen full of icons and a welcome sample will be played.

Loading and playing a Sample

Remove the STOS Maestro program disc and place the back-up sample disc in your drive. To load a sample you must first select the Disc menu (we'll explain all about the various menus in a moment) so click on the disc icon on the second line from the bottom:

Disc menu: The Disc menu will roll down on the bottom line. Click on the leftmost icon and a Gem-type file selector box will appear. You will notice that the samples have been divided into four types and stored in four folders: Music, Sound, Synth and Voice.

Open a folder by clicking on the folder name. Then load a sample by double-clicking on the sample name or by clicking on it to highlight it and then clicking on the OK box. A message will appear in the middle of the screen telling you which sample file the program is loading then you will see a small blue and white visual representation of the sound to the left of the central display area. To hear the sample, left click on the Play icon:

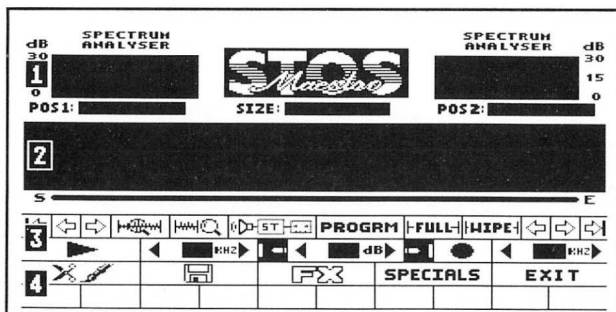
Play: To stop playback press the spacebar. Whenever the program is sampling or playing back you can always halt the operation by pressing the spacebar. You'll notice that when you do this an orange bar appears in the display area marking the point at which you stopped playback. This helps pinpoint small sections of a sound within a larger sample and is useful during editing.

Now that you know how to load and play samples, you can try out some of the other sounds on the disc. Let's now look at all the exciting features STOS Maestro has to offer.

The main screen

STOS Maestro only has one main screen so it's very easy to use. At first glance some of the icons may look strange but they will all be explained in detail. You will soon master STOS Maestro and using it will quickly become second nature.

The STOS Maestro screen is divided into four areas:



Options are selected by moving the arrow around the screen with the mouse and left clicking on an icon. The right button has some special functions which we'll explain when we come to them.

1 Spectrum analysers, position and size displays

Spectrum analysers: These are the two large boxes in the top left and right hand corners of the screen. They give a visual representation of the signal which is entering the Audio-In socket of the STOS Maestro Plus cartridge. They will work whenever you aren't doing anything which requires the program's full attention such as sampling and playing back.

Use the indicators as a guide to the volume level of the signal. They are calibrated in decibels (dB) which is how sound levels are measured. The more decibels, the louder the signal.

When you're recording a sample, get as much signal into the program as you can but don't turn the volume up so high that the bars hit the top of the windows and are truncated as this will distort the sample.

Here's a useful tip: When you're recording, try to avoid trailing the lead which connects the sound source to the cartridge over your computer or alongside your monitor as it may pick up background noise.

Position and size indicators: Underneath the spectrum analysers are the position indicators labelled POS1 and POS2. These show which area of memory you are currently working on and reflect the position of the two white dotted pointers in the sample display area below.

In between the position indicators is the size indicator which tells you how much memory (in bytes) lies between the two pointers. This is very useful: Not only does it save you doing complicated sums but it can tell you exactly how large a sample is so you know how much memory it will take up in your STOS programs.

The total amount of memory will depend upon the size of your ST's ram. On a 1040 ST you will have a little less than 700k to play with. We'll look at this in more detail now.

2 Sample display area

This gives you a visual representation of the sample(s) in memory. Upon loading the program, this area will show all the free memory – and it will be completely empty. The two white dotted lines at either end of the display area are the pointers. The left mouse button controls the left pointer and the right mouse button controls the right pointer.

Using the mouse, position the arrow anywhere in the display area and press the left button. The left pointer will move to the arrow position. Move the arrow to the right and press the right button and the right pointer will do likewise. Note that the position and size indicators change as you move the pointers. Also note that the left pointer cannot be further right than the right pointer, and vice versa.

In addition you can click and drag the pointers around the display area by keeping the left or right button pressed and moving the mouse.

The area between the pointers is the current work area and all recording, playing and editing functions take place between these pointers. If you want to hear just a part of a sample, isolate it with the pointers.

Just below the sample display area is the memory bar indicator. This indicates how much of the available memory is actually in view in the display area. It also shows where it lies in relation to the whole of the memory. This will become clear when we look at the Magnify and De-Magnify facilities. At the moment the bar runs the full width of the display area indicating that the whole of the available memory area is on view.

3 The Play and Record area

This consists of the two rows of icons below the sample display area. Let's look at them one by one.



Moves the left pointer to the extreme left of the sample display area.



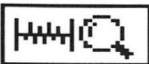
Moves the left pointer to the left of the display by one pixel. If it is already at the extreme left of the display area you won't be able to move it any further.

Although the pointer moves one pixel at a time, the actual number of bytes it steps through will vary according to the amount of memory represented by the display area. On a 1040 ST, if the whole of available memory is shown in the display area a one-pixel movement will step on by about 2250 bytes.



Moves the left pointer to the right by one pixel. Using these step functions you can home in accurately on a particular part of a sample. The right pointer can be controlled with similar icons located at the far right of this line of icons

Magnify: This "blows-up" the current work area (that part of memory between the two pointers) to fill the sample display area. Let's try an experiment. Load the BONUSPNT.SAM file in the VOICE folder.



Move the right pointer a little to the right of the end of the sample (until the size indicator reads between 15000 and 16000) and click on the Play icon to check that the whole sample is between the pointers.

Now click on Magnify. That's more informative, isn't it. Now move the right pointer so that it's just to the left of the last little "blip" in the display area. Click on play. You should now have lost the "s" off "Points". It should be easy to see where "Bonus" ends and "Points" begins. Move the right pointer further left until you only hear "Bonus" when you click on Play.

Now try something a bit more tricky. See if you can move the right pointer so that the "Bonus" become "Bone". Fun, isn't it?

You can magnify an area, define another area and magnify that and so on down to a maximum magnification level of about 320 bytes. At this level, the encapsulated sample will be so short it will be little more than a click.

You cannot magnify more than 15 times. However if you are working on a specific area of a sample, you will probably find you reach the maximum 320 byte level after only four or five magnifications.

Watch the memory bar indicator as you magnify. It shows exactly where the sample sits in relation to the total memory.

De-Magnify: Reverses the magnification process and steps backwards through the various stages of magnification which have been made. Watch the memory bar indicator.



Thru: Clicking here turns the mouse pointer from an arrow into a busy bee. Any sound source fed into the STOS Maestro Plus sampler cartridge will now sound through your monitor as it plays. The quality of the sounds is determined by the record sample rate (second row, extreme right – see later). This allows you to check the quality of a sound before recording it.



Program: Used to program the function keys with samples. Click on the Wipe icon to clear memory and then load the GAMEOVER.SAM sample.

PROGRAM Magnify it and use the right pointer to isolate "Game" – it's not where you'll probably look for it first. Left click on Program and the display area will prompt you to press a function key. Press F1. Right click on Program, press F1 and you will hear "Game". Unlike normal playback mode you can press the function key before the sample has finished playing to create the famous "N-N-N-N-Nineteen" effect – or "G-G-G-G-Game" in this case. Press the spacebar to escape from playback mode. Now isolate "Over" and store it in F2. If you accidentally select Program and don't wish to set a function key then you can quit by pressing any other key.

Program doesn't store the actual samples, rather it keeps a note of the positions of the left and right pointers. We can demonstrate this easily: Click on the Full icon (the very next one on our list) and load GETREADY.SAM. Right click on Program and press F1 and F2.

Full: A useful control which you have already used. It restores the pointers to the start and end of memory.



Wipe: This wipes (empties) the whole of memory, not just the section between the pointers (use the Cut function which you'll find in the Edit Menu). It's non-reversible and you get a warning prompt before it proceeds. It resets the pointers to the start and end of available memory.



Play (and Play Menu): Left clicking here plays the sample between the two pointers.

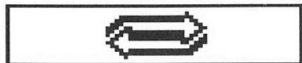


Right clicking will roll down the Play menu on the bottom line. Load GETREADY.SAM (you may have to click on the disc icon to get the Disc Menu back). Magnify the sample and move the right pointer pretty close to the end of the sample. Now for some fun: Right click on Play.

Backwards Loop: Plays the sample in reverse and loops it. Press spacebar to stop.



Forwards Loop: Plays the sample forwards and loops it.



Forwards/Backwards Loop: Plays the sample forwards then backwards, also in a loop.



Backwards: Plays the sample backwards once.



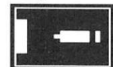
Forwards: This is the default Play button and plays the sample forwards. Notice that as you select each of the icons they appear in the second line in place of the Play icon. This allows you to use any of these types of playback during editing without having to call up

the Play menu eachtime.

Playback Rate: If you recall the sampling theory from the Introduction, you'll remember that the more samples we take in a given time, the better the quality. It's probably occurred to you that you can sample at one rate and play back at another. If you haven't tried this yet, let's try it now. The principle involved is rather like speeding up or slowing down an audio recording using a record deck (playing 45s at 33rpm) or a reel-to-reel recorder. You probably know that if you slow down a recording the pitch becomes lower and speeding it up produces Pinky and Perky voices.

It's the same with sampling. If you increase the playback speed of a sample – the rate at which the digital information is passed through the D-to-A converter – the pitch becomes higher, and vice versa. This is easy to do on a computer. This icon is used to set the playback speed which can be fully varied from 5KHz to 32KHz. Click on the left and right arrows to adjust the rate and check the result by playing the sample. If you double the playback speed, the pitch of the sample will double. In musical terms it increases by an octave. If you right click on this icon it will change the playback rate to that of the record rate.

Playback Plug: This connects the playback routine to the Trigger Level circuitry.

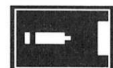


Trigger Level: This sets the Trigger Level in decibels (we mentioned dBs briefly in relation to the spectrum analysers). When plugged into the Playback or Record section of the program it holds up the function until it receives an input signal equal to or higher than the dB level.

When plugged into a section and activated, the screen will change colour while waiting for a signal of the correct level. You can escape from this, as usual, by pressing the spacebar. Check the dB level of incoming signals on the spectrum analysers.

When recording, the Trigger Level can be useful if you want to record a sound with silence in front of it.

Record Plug: This connects the Record routine to the Trigger Level circuitry.



Record: Click here to begin recording. Recording will terminate when the current work area is full. Press the spacebar to terminate earlier.



Note: Unless the Trigger is "plugged in", STOS Maestro will begin sampling as soon as Record is pressed even if there is no sound at the input.

Oversampling

In order to help you make better samples, STOS Maestro uses a process known as Oversampling. It's completely automatic and you don't even have to switch it in and out.

When STOS Maestro is sampling at 32KHz it's running pretty close to flat out. At lower rates, however, it has a little processing power to spare and at sampling rates of 19KHz and less it automatically goes into Oversampling mode.

Instead of sampling at the set rate it takes as many samples as it can during the sampling time and then works out an average figure. So if you set the sample rate to 10KHz and Maestro oversampled at 50KHz, it would take five samples instead of the one and average them out. This will provide a more accurate representation of the sound during that period of time than just one sample.

Record Rate: This determines the rate in KHz at which the recording is sampled. It works in



exactly the same way as the playback rate icon. If you have the STOS Maestro Plus sampler cartridge, try recording the same sample at different rates and see how much mem-

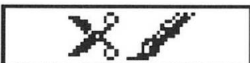
ory each one uses, then compare the quality.

To hear the playback at the same pitch as the recording, make sure the playback rate is the same as the record rate. If you right click here then the record rate will assume the value of the playback rate.

4 The Menu Bar

The third line of icons is used to call up various menus onto the bottom line. You've already used the Play menu and part of the Disc menu.

Cut & Paste Menu:



Rolls the Cut & Paste menu onto the bottom line. It contains a number of tools for manipulating the samples. Load a sample and try out the various editing procedures.

Play cut buffer: Click here to play the contents of the cut buffer. If you haven't Cut anything



yet it will have no effect. Right clicking on this button allows you to edit any sample currently in the cut buffer. All operations in the current work area will only affect the cut buffer data. When you need to return to the

full memory sample, simply right click on this icon again.

Cut: Cuts the area between the two pointers and stores it in the cut buffer. You can't cut more than 50k of sample (check the size in the size indicator window) and if you try, a dialogue window will appear advising you of the fact.



If you haven't done so already, load a sample. Use the right pointer to isolate it, then magnify it. Now click on Cut and the sample will vanish from the display area. Clicking on Play will have no effect but if you click on Play cut buffer you'll hear the sample.

Copy: This is similar to Cut but it copies the area into the cut buffer leaving the original intact.



Again there is a 50k limit on the size of the area you can copy.

Splice: This cuts out the area between the two pointers and stores it in the cut buffer. Then it closes up the gap – useful for swapping samples around memory

SPLICE

Paste: This copies the contents of the cut buffer into memory starting at the position of the left pointer. This overwrites any data which may already be there but the buffer remains intact. If the area between the two pointers is smaller than the contents of the cut buffer, the full contents of the buffer will not be copied.

PASTE

Insert: This is similar to paste but instead of overwriting memory it makes a space for the buffer contents by moving any data to the right of the insertion point (the position of the left pointer) further to the right. It will always insert the full contents of the buffer – any data at the end of memory will be lost. The distance between the left and right pointers dictates how much of the buffer will be inserted.

INSRT

Mix: This operation too, begins at the position of the left pointer. It mixes the contents of the cut buffer with the data which is already there. This can produce some stunning effects. For example, copy a complete sample (try one of the voices) into the cut buffer. Move the left pointer a little to the right and hit Mix. Move it further to the right and Mix again. Make sure the right pointer is sufficiently far right so it does not truncate the Mix. Depending upon the offset, this will produce reverb, delay and variations on the N-N-N-Nineteen effect – or a combination of all three!

MIX

Delete: This simply erases the area between the two pointers.

DEL

Chop: Erases the sample outside the two pointers and moves the sample (between the pointers) to the start of free memory.

CHOP

Flip: This reverses the sample between the two pointers. Playing it will now have the same effect as clicking on the Play backwards icon in the Play menu.

FLIP

Disc menu:



Rolls down the Disc menu on the bottom line which is where all file handling takes place. The icons below are available after you click on this icon.

Load sample: Produces a Gem file selector box which allows you to load a file of sample data at the position of the left pointer, allowing you to store several different samples in memory at the same time. If the memory area between the left and right pointers is smaller than the sample then only this amount will be loaded, so make sure you set them up or use Wipe beforehand.



On first access, the routine will look for files with the extension SAM. We recommend you add this extension to all your sample files – but it's not compulsory.

To search for other extensions, click on the path name so that the text cursor appears after SAM. Backspace three times to delete SAM and type *. don't press Return because a bug in Gem will send you back to the main screen. Instead, click on the top of the file selector box where it says *.SAM. The directory will now show all files and folders regardless of their extension. A quick tip is to click on the path name press Escape and then click on the file selector box top – this defaults to drive A and uses a *.* search specification.

It's interesting to note that the sample files created by most samplers for the Atari ST produce the same kind of data. You will be able to load most samples made with other samplers into Maestro – and vice versa. Even more interesting is the possibility of grabbing samples from commercial games. Files of sample data may have the extension SAM, SPL, SNG, SND or something else. But beware: Trying to access some protected discs may cause the STOS Maestro program to lock up.

You may have discovered that you can load any file into STOS Maestro and it will treat it as sample data. If it's not a sample it will sound terrible – probably like a screech.

Save sample: This does not prompt for a filename but saves the area between the pointers with the same name as the last filename used. It will be the name of the last sample you loaded or saved, otherwise it will use the default SAMPLE.SAM. This is useful as it allows you to work on a sample and save it quickly without having to type in a filename. If the file already exists on the disc, a dialogue box will appear requesting permission to update the file.



A unique STOS Maestro feature means that the playback rate is saved to disc along with the sample data. You've probably noticed the playback rate change when loading various samples. No longer will you have to determine the sample speed by experimentation. However when loading samples from games or created on other samplers you'll have to find the correct playback rate by trial and error.

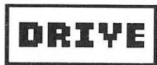
Save sample as: Click here to save the sample between the two pointers. This option allows you to enter a new filename just as you would with any Gem file selector.



Insert sample: This works in a similar way to the Insert function in the Cut & Paste menu. It reads a file of sample data from disc and inserts it at the left pointer position and moves the sample in memory to the right to make room for it. If you want to combine samples use mix in the Cut & Paste menu.



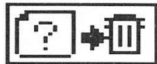
Drive: Prompts for the current drive. When prompted press a letter from A to H. The default depends on the drive from which you booted STOS Maestro.



Disc Info: Gives you information about the current drive: Identifier, total disc space and total free space.



Delete File: When the file selector appears, click on the file you wish to delete and then click on OK or just double click on the file. After selecting a file there is a confirmation prompt so that you can have second thoughts about this action.



Rename: Use this to rename a file. It can be very useful if you've forgotten to give a file an extension or if you've a list of samples such as WOW1, WOW2, WOW3,...WOW16. Click here and the file selector will appear. Select the file you wish to rename by double clicking on it or by clicking on it once to highlight it and then clicking on OK. A new file selector box will appear with the text cursor on the selection heading. Enter a new filename and press Return or click on OK and the file will be renamed. The routine will not allow you rename a file with a name which already exists.



Format: Allows you to format a disc as single or double sided without leaving STOS Maestro. This produces a warning prompt as formatting a disc will erase all data on it. Get past this and you are asked to select Drive A or B and then the number of sides you want to format.



New Folder: Folders are very useful for sorting your samples into types. You will have noticed that the samples on the data disc are stored in four folders. Click here and enter the name of the new folder – eight letters maximum. To escape from this option, press the 'escape' key - a null entry will also terminate the procedure.



FX: Rolls down the effects menu onto the bottom line. This reveals the four icons below.



Echo: An echo is produced by repeating the original signal after a short delay. Although he delay is short you will be able to hear the two (or more) echoes quite distinctly. A characteristic of natural echo is that subsequent echoes decrease in volume. If you click on Echo with the left button, STOS Maestro will automatically add echo to the sample between the pointers. This is irreversible so a dialogue box pops up to warn you. You can repeatedly add more and more echo to a sample for increasingly pronounced effects. As Echo effectively extends the length of a sample you may have to move the right pointer further right to accommodate all the repeats.

Clicking on Echo with the right button adds echo to any sound being fed in from the STOS Maestro Plus sampler cartridge in real time.

Reverb: Reverberation, to give it its full title, is produced by thousands of extremely short echoes following hot on the heels of each other. They are so short, in fact, that you cannot tell when one ends and another begins – they blur together and seem to become an extension of the original sound. Reverb produces a ringing, cavernous effect. Try it with voice samples and use it on dripping sounds. If your game is set underground you will need reverb for atmosphere.



Do a left click to add reverb to the sample between the pointers and right click to add reverb in real time to a sound being played through the STOS Maestro Plus sampler cartridge.

Hall: The type and quality of reverb will vary according to the environment. Compare the reverb qualities of a concert hall with your bathroom, for example. Hall reverb is more pronounced than Maestro's standard reverb – a little slower, perhaps. Left click to apply it to the sample between the pointers, right click to apply it to a sound coming in through the STOS Maestro Plus sampler cartridge.



Room: This is the third type of reverb effect which STOS Maestro has to offer and it's a little shorter than the others. As usual, left click to add echo between the pointers, right click to add it to a sound playing through the STOS Maestro Plus sampler cartridge.



Delay FX

You can have a lot of fun and create lots of fascinating sounds with reverb and echo.

Here are a few suggestions:

- For a well-pronounced effect, give a sample three, four, or more doses of echo or reverb.
- Add both Echo and Reverb to a sample.
- Flip a sample, apply an effect and then Flip it back.
- Move the left pointer slightly before applying subsequent effects.
- Use Mix to create combinations of echo and reverb.
- Split a sample (Insert blank memory) and add an effect only to the middle of it.
- Don't forget to Magnify the sample in order to home in on selected areas.

Amplify: Amplifies the sample between the pointers to make it louder. This can be very useful if you can't record your favourite sound at a high enough volume. You can also apply it to a sample if it has undergone a lot of editing. You will probably have noticed by now that lots of manipulation can reduce the effective volume of the sample. But use Amplify sparingly as it will increase the background noise as well. If you continuously amplify a sample it will soon distort.

AMP

Pack: This is another unique Maestro feature to help you record even better samples. It packs the data between the pointers, effectively halving the size of the sample and saving lots of memory. It also means that your STOS Basic games won't slow down as much when samples are playing.

PACK

In order to hear a packed sample at its original pitch you only need to play it at half its sampled rate. For example, you could record a sound at 32KHz, Pack it twice and play it back at 8KHz. The quality will be better than if you had recorded at 8KHz because the Pack routine takes an average reading of the data (like Oversampling).

Fade In: Creates a Fade In on the sample between the pointers. The wider apart the pointers, the slower the fade. You'll notice that this drastically reduces the size of the sample in the display area.

IN

Fade Out: Creates a Fade Out on the sample between the pointers. The comments relating to Fade In apply here, too.

OUT

Soft: This is the opposite of amplify and reduces the volume of the sample between the pointers, too much softening and you will probably notice an increase in noise. While you may be able to amplify a softened sample back to its original volume without much signal deterioration, softening an amplified sample will bring with it some of the noise created by Amplify.

SOFT

Filt: Passes the sample between the pointers through a low pass filter. This passes the low frequencies and filters out the high ones. Noise tends to reside in the high frequencies so use this to help reduce the noise level on very noisy samples. It also "softens" a sound.

FILT

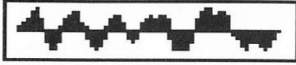
Specials: This rolls down another menu containing more display options.

SPECIALS

About box: Gives you information about STOS Maestro – the version number and credits.



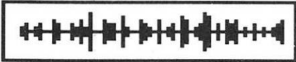
Filled Waveform: This option and the next two determine how samples are displayed in the sample display area. This is the default option and shows the waveform of the sample with the positive (upper) and negative (lower) cycles filled in white and blue.



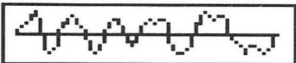
If you have a musical instrument which can produce sine waves (see the Introduction), plug it into the cartridge, sample a sine wave and Magnify it. This option will clearly show the sine wave shape in the display area.

Synthesisers often use a square wave to produce a clarinet sound. If you can analyse either of these you will see the square-ish waveforms (a clarinet will produce waves which are not quite square due to additional harmonics). Brass and string sounds are often constructed from sawtooth waves (sometimes called ramp waves) and make for interesting analysis, too.

Envelope: This displays the volume envelope of the sample – in other words it shows how loud it is, rather like the spectrum analysers. You will see that the positive and negative cycles are identical (as if they had been reflected along the central line) as you can't really have a negative volume.



Waveform Outline: This is similar to the Filled Waveform option but it shows the waveform of the sample as an outline, just like you'd see it on an oscilloscope. It's often easier to see the exact shape of a waveform when displayed in this way.



Oscilloscope: The STOS Oscilloscope lets you see the shape of the sounds which are entering the STOS Maestro Plus sampler cartridge. Most sounds in everyday life, particularly music and the human voice, are very complex indeed and their waveforms are very complex, too. Try it for yourself.



If you can produce sine waves you'll be able to see them quite clearly. Low notes produce "longer" waves while higher notes bunch the waves closer together. You will even be able to spot sine wave characteristics in the spectrum analysers. Exit to the Main Screen by pressing the spacebar.

Exit: Click here to exit back to the Gem Desktop. If you have edited the sample in memory, it will give you the option of saving it before quitting.



Creating your own welcome sample

As you enter STOS Maestro you are welcomed by a voice sample. You can customise your own welcome sample using the following easy procedure:

- Create a sample using Record at around 10KHz
- Save the sample from the disc menu under the filename WELCOME.SAM, making sure it's in the same folder from which you run MAESTRO.PRG

Every time you now load STOS Maestro it will pick up your sample and play it. If this becomes annoying then just remove the WELCOME.SAM file from the folder.

Now you know everything there is to know about STOS Maestro – have fun!

2: Disco, Phaser and DEMOS

STOS Maestro disco unit

The STOS Maestro Disco unit is a real-time digital delay program. You can load it directly from disc by double-clicking on DISCO.PRG or from the STOS Maestro loading screen by selecting the 2 key on your Atari ST. You will of course need the STOS Maestro Plus sampler cartridge to use this program.

Make sure your ST is switched off before plugging the cartridge into the ST's cartridge port. Plug a sound source into the STOS Maestro Plus cartridge. To hear exactly what Disco does, it will help if you have control over the duration of the sound – a microphone or musical keyboard is ideal. If you intend to use music, then compose a piece which is fairly simple. Once you hear what's going on you can try it out on the 1812 Overture! Disco is a digital echo machine but unlike the echo function in STOS Maestro it only plays back a single echo at a time. However the delay time is fully under your control and can be in the range of 0 to 9.99 seconds. Disco plays the original sound as soon as it enters the cartridge and then it plays it again after the delay period. It's a continuous process so the delayed sound will be heard at the same time as the original sound – trailing behind it a little. This can produce some fascinating overlapping sound effects.

Controlling Disco

Disco is controlled with the function keys which are mainly used to alter the delay time:

Function key	Effect
F1	Increased by 10 milliseconds (100th of a second)
F2	Decreased by 10 milliseconds
F3	Increased by 100 millisecond (10th of a second)
F4	Decreased by 100 millisecond
F5	Increased by 1 second
F6	Decreased by 1 second
F7 – F9	Pauses the display when held down.
F10	Press twice to reset ST and default back to Gem Desktop

You can change the delay time as music is playing through the unit. Very short delays will produce a chorus effect which "thickens" the sound or longer delay times sound very effective if you can match the delay setting to the beat of the music. Alternatively, offsetting the delay time – a little faster or slower than the beat – can produce interesting effects, too, as the music and the repeat move out of sync with each other. Try it with any music of a rhythmic nature. It works particularly well with electronic music and many of Jean Michel Jarre's compositions have suffered at the hands of (or been enhanced by?) the Disco program. The central area of the screen produces a graphic display (spectrum analyser) of the sound entering the cartridge port. Plug in some music, turn the lights down, pump up the volume, watch and relax.

The STOS Phaser program

The STOS Phaser is a small utility which we have included on the program disc as an extra. Although it's a simple program it can produce some fascinating effects. Load STOS Phaser directly by double-clicking on the PHASER.TOS file or by loading it via STOSM.PRГ.

Warning In order to use the phaser you require the STOS Maestro sampler cartridge. Make sure your ST is switched off before plugging the hardware into your ST's cartridge port.

Phasing is a very popular effect especially in electronic music, producing a sort of sweeping effect. Play some music and listen for yourself. If you want to know what a phaser actually does to a sound and how it produces its effects you need to know a little about waveforms. A detailed explanation is beyond the scope of this manual and it's not strictly necessary – you can use the program without knowing the theory behind it. However, if you want a visual analogy, think of a zebra walking behind a black garden fence – sometimes you'll see different bits of it appearing and disappearing in a cyclic fashion. A phaser does a similar kind of thing to the harmonics of a sound. Harmonics are the particles of a sound which make it bright, hollow, brassy and so on. As it sweeps through the harmonics, the tone of the sound changes. You will hear the effects of the phaser best with sounds which contain lots of harmonics such as brass and strings. Flutes, on the other hand, don't contain many harmonics so there is less for the phaser to sweep through.

Controlling the phaser

The phaser is very easy to control – like the Disco it only uses the function keys.

F1 Increases the phaser speed by 1

F2 Decreases the phaser speed by 1

The phaser speed determines how quickly the phaser sweeps the signal. At slow speeds the effect can be very tranquil and relaxing. At faster speeds it can be quite hectic.

F3 Increases the phaser depth by 1

F4 Decreases the phaser depth by 1

Phaser depth is the degree or amount of sweep. A low or shallow depth setting can produce a very subtle effect. At greater depths it really rakes through those harmonics.

F5 Select phaser sweep – SWP

F6 Select phaser forward sweep – FWD

F7 Select phaser backward sweep – BWD

These keys determine the direction of the sweep: F5 (up and down), F6 (down only) and F7 (up only).

F8 Turn phaser off

F9 Turn phaser on

F10 Reset computer and default to the Gem Desktop (Must be pressed twice to work).

Try out the Phaser by playing some of your favourite pieces of music through it – using a fairly low depth and slow speed for a subtle effect, then increase both gradually to taste.

The DEMO.TOS files

If one of your Atari floppy disc drives is double sided then you will be able to run these demos. Simply double click on a DEMO.TOS icon to play a tune. The two STOS Maestro discs have a special dual format – single drive owners can still use the discs but won't be able to access the demos. The programs run from side one and then load up data from side two of each disc. Turn up the volume for an example of what can be achieved with STOS Maestro.

3: STOS Maestro Basic Interface

Installing the STOS Basic extension file

Before you can access the sampler from Basic, you first need to install the new commands onto your current STOS language disc. Load up STOS Basic, insert the STOS Maestro program disc and enter the following command:

```
load "basic\install.bas"
```

Once this has loaded, you should place a copy of your STOS language disc into drive A and type:

```
run
```

The new Maestro extension can now be installed onto all your copies of STOS Basic by pressing the E key. Select a STOS folder to update using the resulting file selector. Once inside a STOS folder press Return– the new extension file will now be saved into the STOS folder. Since the installation procedure holds all the extension commands in a STOS memory bank, no disc swapping is required.

If you own a copy of the STOS Compiler, you should now place your **backup** of the compiler disc into drive A and press C. The appropriate compiler extension can now be installed into the COMPILER folders in the same manner as above. You can now add sampled sounds to your Basic programs and compile them in the normal way.

The Maestro accessory

A special memory bank system is used by the STOS Basic extension to hold samples. This makes loading in samples and playing them child's play. To make life as easy as possible we've devised a powerful accessory capable of generating sample banks from pre-recorded sample files.

This combines up to 32 separate samples into a single memory bank, which can then be incorporated directly into your STOS Basic programs. You can enter it from the Maestro disc with a line like:

```
accload "basic\maestro"
```

Here is a breakdown of the various menu options.

STOS

About Bank Creator: This displays a copyright message. Click on the left mouse button to return to the main screen.

QUIT : Exits back to your current program. Warning! don't call this option without saving

your sample bank first!

QUIT & GRAB: Grabs the samples installed into the ST's memory, and places them into bank 5 of your current program. This option is only available if Maestro has been loaded as an accessory.

DISC: The disc menu allows you to load and save samples onto the disc.

LOAD SAMPLE.???: Loads a sample into bank number 5. These samples are those which have been imported directly from the STOS Maestro Plus cartridge (or compatible). The Maestro accessory can be used with the output of most sound samplers currently available for the Atari ST. This allows you to enter samples from many commercial games. Look for files with extensions like .SAM, .SMP or .SND. Feel free to experiment with this option as much as you like. The worst that can happen in practice is that the sample will fail to load.

LOAD SAMPLE.MBK: This enters a memory bank previously created with the Maestro accessory.

LOAD SAMPLE FOLDER: Loads an entire folder full of samples into memory. This option is especially useful when using the disc, as it enables you to install all of a folder's contents into a single memory bank in one smooth operation.

SAVE SAMPLE.MBK: Saves all the samples currently installed into memory. This data can be subsequently loaded for use by the SAMPLAY and SAMMUSIC commands.

SAMPLES

DELETE SAMPLE: Deletes one of the 32 possible samples from the ST's memory. The sample to be erased is indicated using a number from 1 to 32.

RENAME: Changes the name of one of the samples currently installed into memory.

PLAY SAMPLE: Plays one of the samples which has been loaded into the accessory. Before the sample is played, you will be requested for the speed. This should be input as a number from 5 to 22.

Maestro tutorial

We'll now show you how the various features of the Maestro accessory can be actually used. Start off by loading the accessory from the Maestro disc with:

```
dir$="basic"  
accnew:accload "maestro"
```

First enter the accessory from the Help menu by typing <HELP><F1>. Then place the samples disc into drive A and select the LOAD SAMPLE.??? option from the DISC menu. This presents you with a standard STOS file selector. Now load a sample by clicking on the SOUND folder, and choosing the file EXPLSION.SAM. You can play this sample from the Sample menu using a speed setting of around 10.

This process can now be repeated for the samples LAZERGUN and PISTOL. Insert a blank disc into drive A and save these samples with the SAVE.MBK option from the disc menu. Use the filename EFFECTS.MBK for this purpose. Now exit to the STOS Basic editor with Quit and type:

```
load "effects.mbk":rem Load samples into bank 5
```

These samples can be played from STOS Basic using the lines:

click off:sound init:rem Initialise sound chip
samplay 1:rem Play Explosion
samplay 2:rem Lazer
samplay 3:rem Gunshot

We'll now show you how you can load an entire folder full of samples in exactly the same way. Call up Maestro using <HELP><F1>, and select the Load Folder option from the Disc menu. When the file selector appears, access the VOICES folder in the normal way. Now move the mouse pointer over the filename box and enter the text: .SAM

The contents of the Voices folder will now be automatically loaded into the ST's memory. You can then save this data under the filename of VOICES.MBK by placing a blank disc into drive A and calling the SAVE.MBK option.

Finally choose the Quit option to re-enter STOS Basic then type in the following program.

```
load "voices.mbk"  
  
10 click off:sound init:samspeed auto  
20 for i=1 to 11  
30 samplay i  
40 wait key  
50 next i
```

This pages through each of the 11 voices stored in memory bank number 5.

The Drum machine

In order to illustrate the possible applications of this package, we have included a simple Drum machine on the Maestro disc. This can be loaded using the line:

```
load "basic\drums.bas"
```

When you run the program, you are presented with a picture of a drummer and his dog. The action of the various instruments can be controlled through the keyboard numbers from one to six. As you can see, some of these instruments are fairly unusual!

Another feature is the R option which allows you to record long sequences of drum beats into the ST's memory. This performance can be interrupted at any time using the spacebar. Your composition can now be replayed by pressing P. Notice how the program has cleverly preserved the exact timing of your original key-presses.

The drum machine was written entirely in STOS Basic. You can therefore list the program to the screen or the printer in the normal way. Interestingly enough, the actual program is remarkably simple. The samples for the drums are held in bank number 5, and the recording itself is stored in bank 6. Each individual sound corresponds to a single entry in bank 6, with the gaps in your music being represented by a Chr\$(0).

Here are a few ideas for possible extensions.

1. Replace the drum samples in bank 5 with some of your own. Remember that these samples should first be converted into the correct format using the Maestro accessory.
2. Introduce the ability to change the pitch of a note during play. This could be generated by assigning a row of keys to particular notes. You could then simply replace the SAMPLAY instructions with the appropriate SAMMUSIC commands.

3. Extend the number of available instruments. it's possible to add up to 32 different instruments to the drum machine.
4. Add the ability to load and save sequences on to the disc. This would involve saving bank 6 using the BSAVE command from STOS Basic.
5. Allow the user to speed up or slow down drum patterns he has recorder.

4: STOS Maestro commands

STOS Maestro instructions

The STOS Maestro extension includes a number of powerful instructions which allow you to incorporate sampled sounds directly into your own STOS Basic programs. A demonstration of how these commands can be found in the programs PLAYER.BAS and DRUMS.BAS on the STOS Maestro program disc.

The Playback commands

These commands are used to play back a previously generated sound sample within the STOS Basic system. Any programs which use these instructions can be run independently of the STOS Maestro Plus cartridge.

SOUND INIT (*Initialise the ST's sound chip*)

The SOUND INIT command kills any existing sounds and prepares the sound chip for use by the sampler instructions. It is normally called at the start of a program like so:

```
10 click off:rem Switch off keyboard clicks
20 sound init:rem Initialise sound chip
```

The keyboard click will interfere with the sampler system. It is therefore necessary to turn off the keyboard when using the sampler.

SAMPLAY (*Play a sample from the current memory bank*)

SAMPLAY *n*

SAMPLAY plays one of the 32 possible samples stored in the current memory bank (normally bank 5). *N* is the sample number ranging from 1 to 32.

Example:

```
load "effects.mbk"

click off:sound init:samspeed auto
samplay 1: rem Plays an explosion
samplay 2: rem Laser beam
samplay 3: rem Gun
```

This example uses the EFFECTS.MBK file created in the tutorial on the STOS Maestro accessory.

These samples are played back using interrupts and are completely independent of the rest of the STOS system. This means that you can happily add sound effects to an arcade game such as Zoltar, without slowing down any of the action.

Note that only a single sample can be played at any one time. So if you start a new sample before the existing one has completed, the old sample will immediately be replaced.

It's possible to exploit this feature to produce a number of intriguing effects. Take the following example:

```
load "voices.mbk"
```

```
10 click off:sound init:samspeed auto
20 for i=1 to 5
30 samplay 1:wait 5
40 next i
```

This generates a noticeable stutter in the sampled voice. VOICES.MBK is an example file created in the section on the Maestro accessory. It contains the entire contents of the VOICE folder on the samples disc. One minor limitation of the SAMPLAY instruction is that it is unable to work alongside any of the STOS music functions. If your program uses music, this should be suspended with MUSIC FREEZE before a sample is played, and then restarted after the sample has been completed.

SAMSTOP (*Stop the current sample*)

The SAMSTOP command terminates all current sampling operations.

Example:

```
reserve as data 6,12000

bload "A:\sound\explosion.sam",6:rem Load a sample from the samples disc

10 click off:sound init:samspeed auto
20 samraw start(6),start(6)+length(6):rem Plays a sample
30 wait key:samstop
```

SAMSPEED (*Set the sampling rate*)

SAMSPEED *n*

Samspeed sets the rate which will be used for any subsequent sampler commands. *N* can range between 5 and 22KHz, with 10KHz as the default.

Example:

```
load "voices.mbk":rem Created with MAESTRO.ACB

10 click off:sound init
20 for i=5 to 22
30 samspeed i:samplay 1:wait 20
40 next i
```

Incidentally, while the STOS Maestro Plus cartridge is capable of working at speeds of up to 32KHz, you are currently restricted to a maximum of 22KHz in your STOS Basic programs. In order to play 32KH samples from STOS Basic you will need to use MAESTRO.PRG and call the Pack option from the FX menu. The compressed sample can then be played at a speed of 16KHz from STOS Basic. It's not 32KHz but it's the only alternative – STOS is too busy handling sprites and all the other interrupts to play back samples at a rate faster than 22KHz.

SAMSPEED AUTO (*Enter automatic sampling mode*)

SAMSPEED AUTO automatically selects the appropriate sample speed for the current sample. The automatic sampling mode expects the speed of the sample to be encoded into the current memory bank during the recording process. This information is included with all samples produced using the STOS Maestro sampler program. However if you try to use this option in conjunction with samples created from another source, such as the SAMRECORD function, a syntax error will be generated.

Also note that automatic mode is effectively cancelled the next time you change the sampling rate with SAMSPEED.

Example:

```
load "voices.mbk":rem Load some sound effects

10 click off:sound init
20 print "Default speed":samplay 1
30 print "Hit a key to continue":wait key
40 samspeed auto:rem Set appropriate speed automatically
50 print "Recorded speed":samplay 1
60 wait key
```

SAMSPEED MANUAL (*Exit automatic sampling mode*)

SAMSPEED MANUAL exits from automatic playback mode. It immediately returns the sample speed back to the rate which was set by the most recent SAMSPEED *n* command in the program.

Example:

```
load "voices.mbk":rem Loads some sound effects

10 click off:sound init
20 samspeed 20:rem Set very fast speed
30 print "20 KHz":samplay 1
40 print "Hit a key to continue":wait key
50 samspeed auto:rem Set proper speed
60 print "Recorded speed":samplay 1
70 wait key:samspeed manual:rem Set samspeed back to 20KHz
80 print "20 KHz again":samplay 1:wait key
```


SAMMUSIC (*Play sample as a note*)

SAMMUSIC *n*, 'note'

The SAMMUSIC instruction allows you to alter the tone of a sample when it is played. It's specifically intended for samples which represent some sort of musical instrument. The effect is to widen the scale of the sound to encompass the full range of musical notes.

N can be any one of the 32 possible samples stored in the current memory bank.

'note' must be a string containing one of the following notes;

C,C#,D,D#,E,F,F#,G,G#,A,A#,B

The current pitch of the sample is assumed to be middle C.

Example:

load "voices.mbk":rem Load some effects created previously

10 click off:sound init

20 for i=1 to 12

30 read n\$:print "Key ";N\$:sammusic 3,n\$

40 wait key:rem Wait for the voice to be finished

50 next i

60 data "C","C#","D","D#","E","F","F#","G","G#","A","A#","B"

A better example of SAMMUSIC can be found in the file PLAYER.BAS on the STOS Maestro program disc. This lets you load a sample into memory, and then 'play' it from the keyboard. PLAYER assumes that the sample found on the disc is in **raw** form. It's therefore possible to load samples straight off the samples disc. However, don't try to input .MBK files created with the STOS Maestro accessory as you will generate an error message from the program.

SAMLOOP ON (*Repeat a sampled sound*)

After this command has been performed, any sample which is subsequently played using the SAMPLAY, SAMMUSIC and SAMRAW functions will be continually repeated from beginning to end.

reserve as data 6,12000

bload "A:\sound\explosion.sam",6:rem Load a sample from the samples disc

10 click off:sound init:samspeed auto

20 samloop on

30 samraw start(6),start(6)+length(6)

40 print "Press a key to end"

50 wait key:samstop:samloop off:click on

Like all the sampling commands, SAMLOOP is accomplished using an interrupt routine. So once you have started the loop your program can get on with more important work, and forget about the sampler completely.

SAMLOOP OFF *(Exit from a sample loop)*

The SAMLOOP OFF command instructs the sample routine to play any future samples exactly once. This instruction has no effect on the sample which is currently being played. If you wish to halt an existing sample you will need to use the SAMSTOP directive instead.

Example:

```
reserve as data 6,12000
load "A:\sound\explosion.sam",6:rem Load a sample from the samples disc

10 click off:sound init:samspeed 10
20 samloop on:rem Repeat the next sample
30 samraw start(6),start(6)+length(6):rem Play sample
40 print "Press a key to end"
50 wait key:samstop:samloop off:rem Turn off looping
60 samraw start(6),start(6)+length(6):rem Play a single shot
```

SAMDIR FORWARD *(Play a sample from start to finish)*

Normally all samples are played back in the direction they were originally recorded. However you can change this using the SAMDIR BACKWARD command.

SAMDIR BACKWARD *(Play a sample backwards)*

This reverses the action of all future playback commands. Any subsequent samples will now be played backwards until cancelled by using SAMDIR FORWARD.

Example:

```
reserve as data 6,10000
bload "A:\voice\gameover.sam",6:rem Load a sample from the samples disc

10 click off:sound init
20 samdir backward
40 samraw start(6),start(6)+length(6)
50 wait key:samstop
```

The above program plays the phrase "game over" back to front. You can also call the SAMLOOP command with a reversed sample. Add the following lines to the above example:

```
30 samloop on
55 samloop off
```

Although SAMDIR may seem rather pointless at first, it's actually surprisingly useful. You can, for instance, create a convincing impression of an "alien" language, by simply reversing normal speech, and then altering the playback speed. This technique was used to great effect in the radio version of *The Hitch-hiker's Guide to the Galaxy*.

SAMSWEEP ON (*Repeatedly play a sample forwards and backwards*)

The SAMSWEEP command is ideal for the generation of 'swishing' noises like those produced by an aeroplane. The effect is to repeatedly cycle a sample forwards and backwards. Note that, like SAMLOOP, the sweeping action only takes effect from the next sample which is played.

Example:

```
reserve as data 6,784
bload "A:\sound\swosh.sam",6:rem Load a sample from the samples disc

10 click off:sound init
20 samsweep on:rem Start sweeping
30 samraw start(6),start(6)+784
50 wait key:samstop
```

SAMSWEEP OFF (*Stop sweeping effect for the next sample*)

This removes the sweeping effect from any sample which is subsequently played with the SAMPLAY, SAMMUSIC and SAMRAW instructions. Like SAMLOOP OFF it has no effect on the sample which is currently being performed.

SAMRAW (*Plays sample straight from memory*)

SAMRAW start,end

Plays a raw sample stored from addresses *start* to *end*. Raw samples can be placed anywhere in the ST's memory. So it's perfectly possible to load a sample into a string, and then subsequently manipulate it with the normal string functions.

Example:

```
10 SOUND$=string$(chr$(0),12000)
20 bload "\VOICEGAMEOVER.SAM",varptr(SOUND$)
30 samraw varptr(SOUND$),varptr(SOUND$)+len(SOUND$)
40 wait key
50 rem Chop the word "over" from "game over"
60 OVER$=right$(SOUND$,8000)
70 samraw varptr(OVER$),varptr(OVER$)+len(OVER$)
```

Raw samples can also be looped, reversed and swept using the appropriate sampler commands.

Example:

```
10 reserve as work 8,100000:rem Reserve 100k to hold samples
15 click off:sound init:samloop off:samsweep off
20 FS=file select$("**.sam","Load sample to play")
```

```

30 if F$="" then end
40 open in #1,F$:L=lof(#1):close #1:rem Get length of sample
50 blob F$,8:rem Load sample
60 input "Playback speed";S
70 if S<5 or S>22 then print "Invalid Speed":goto 50
80 samspeed S
90 samraw start(8),start(8)+L
100 print "Hit a key to continue":wait key:goto 20

```

SAMBANK (*Select current bank for SAMPLAY and SAMMUSIC commands*)

SAMBANK *n*

N is a number from 1 to 15 specifying a STOS memory bank to be used by the sampler.

As a default, the SAMPLAY and SAMMUSIC operations expect a list of previously created samples to be available from memory bank number 5. Unfortunately this can cause serious problems for anyone wishing to add sampled sound to an existing program.

Supposing you wanted to include sound effects in a program such as Zoltar. In this case, you can't place the samples in bank 5, as this is already allocated to one of the screens. You therefore need to change the current sample bank with SAMBANK. See the section called *Adding samples to your games* for more details of this procedure.

Another application is to allow you to incorporate more than the maximum of 32 samples in a STOS Basic program.

Example:

```

load "effects.mbk",5
load "voices.mbk",6

10 click off:sound init:samspeed auto
20 sambank 6:rem Use bank number six
30 for i=1 to 5
40 samplay i:print "Hit a key to continue":wait key:rem Press a key
50 next i
60 sambank 5:rem Set bank back to 5
70 goto 30:rem Playback 5 sound effects

```

SAMCOPY (*Copy a sample from one place to another*)

SAMCOPY *start,end,dest*

This is just a special version of the standard STOS copy command which allows you to start copying from an **odd** location -that is, from any location whether it is on a word boundary or not. SCREENCOPY only works on word boundaries for reasons of speed. *Start* is the beginning of the address which is to be moved. It can take any value you wish, and does not have to be even.

End is the address of the end of the sample which is to be copied.

Dest is the destination of the new sample.

Example:

```
10 reserve as work 10,30000:rem Reserve 30k
20 input "Place the sample disc in drive A";AS
30 samspeed 12
40 dir$="A:\VOICE\"
50 blood "gameover.sam",10:rem Load a sample
55 rem Make a second copy the sample in bank 10
60 samcopy start(10),start(10)+10000,start(10)+10001
65 rem Play the entire sample
70 samraw start(10),start(10)+length(10)
```

SAMPLACE (*Return the current position in the sample*)

p=SAMPLACE

The SAMPLACE function retrieves the present position in the sample which is currently being played. This number is relative to the start of the sample. The possible values taken by SAMPLACE range from 0 to the total sample size (in bytes).

Imagine you had a sample which was 20k long. If you were to get the position at the halfway point, a value of around 10000 would be returned.

```
10 rem Simple tape counter
20 click off: sound init: curs off: hide
30 input "Sample number";N
40 input "Sample rate";S
50 samspeed S: samplay N
60 XC=xcurs: YC=ycurs: P=samplace/1000: print "Tape counter: ";P; " ";:locate XC,YC: if
inkey$="" then 60
70 goto 30
```

The SAMPLACE function is often used to determine when a sample has finished playing. This information is essential if you wish to combine melodies produced from the MUSIC accessory with sampled sounds.

Example:

```
load "music.mbk":rem Load from the STOS Basic accessory disc
load "effects.mbk":rem Created previously with MAESTRO accessory

10 click off:music 1:tempo 30
20 if rnd(10000)<>3 then 20:rem Produce random sound effects
30 music freeze:sound init:samplay 1
40 if samplace=11842 then samstop:music on:goto 20:rem Sample 10k long
50 goto 40
```

The Sampler control commands

These functions are designed solely for use with the STOS Maestro Plus cartridge (or equivalent). They give you complete control over the various features provided by the sampler and allow you to create customized sampling programs in STOS Basic. This opens the door to a bewildering range of possible applications, from speech synthesizers to simple speech recognition systems!

SAMRECORD (*Records a sample from the sound cartridge*)

SAMRECORD *start,end*

SAMRECORD starts the sampler running, and records a sample using the current speed setting. The new sample is entered directly into the memory locations between *start* and *end*. While the sample is being recorded, the sampled sound is continuously played through your TV or monitor.

Start is the address of the start of a sample. This is usually a memory bank.

End is the address of the end of the sample. When this point is reached, the sampling process is automatically terminated.

Samples created using this function can be saved to the disc and accessed directly from either the Maestro program or the SAMRAW command.

Example:

```
10 rem Recording a sample
20 click off : sound init
30 reserve as work 6,1024*50 : rem Reserve 50k for sample
40 input "Sample speed";S
50 if S<5 or S>22 then print "Invalid speed" : goto 40
60 samspeed S
70 print "Prepare your recorder"
80 samthru : input "Hit return to start sampling";R$
85 print "Recording sample"
90 samrecord start(6),start(6)+1024*50
100 if samplace<49999 then 100 : rem Wait until recording finished
110 print "Playing back sample"
120 samraw start(6),start(6)+length(6) : rem Play sample
130 if samplace<1024*50 then 130 : rem Wait until playback finished
140 input "Play again Y/N";A$: if A$="Y" or A$="y" then 120
150 goto 40
```

As you might expect, the sampling rate can be set using the SAMSPEED *n* instruction, and the sampling can be aborted at any time with SAMSTOP.

It's important to note that this instruction does store the recording speed as part of the sample. So if you try to play back a sample using the SAMSPEED AUTO instruction, an error will be generated.

SAMTHRU (*Play the input from the sampler through the monitor's speaker*)

The SAMTHRU command simply pipes the current sound output through your TV speaker. This allows you to listen to the input without creating an actual sample.

Example:

Place a tape in your recorder and press PLAY. Now type:

```
click off:sound init:samthru
```

This THRU mode can be interrupted at any time using the SAMSTOP command.

SAMPLE (*Read the sample value directly from the cartridge port*)

```
s=SAMPLE
```

S holds a value from -127 to +127 which represents the volume of the sound in a digital form. The STOS Maestro Plus cartridge reads this value thousands of times per second and uses it to create a digital representation of the sampled sound.

Example:

```
10 click off:sound init:l=0:mode 1:cls physic:cls back
20 ink 1:hide:samthru
30 S=sample/2:draw l,100 to l,100-S:inc l
40 if l>=640 then l=0:cls physic:goto 30 else 30
```

5: Creating your own sampler In STOS Basic

Initialisation

This sets up the screen and reserves space for your samples in one of the STOS Basic memory banks.

```
10 rem Sampler
20 mode 0:click off:sound init:rem Initialise sound chip
25 rem Set up screen
30 cls physic:cls back:windopen 1,0,0,40,25
40 title "BASIC SAMPLER":windopen 2,1,5,38,19,0
50 rem reserve maximum available memory
60 MEM=free-50000:if mem>0 then reserve as work 6,MEM else print "Not enough
memory":end
```

Most of the above code is fairly obvious, but pay special attention to line 60. This reserves all the currently available memory for the sample, leaving 50k for the program itself.

Entering the user's commands

The input routine forms the heart of your sampling program and interprets the commands as they are entered by the user. You can make this routine as simple or as complex as you wish. Commands can be entered using either the keyboard, STOS menus or the mouse. Here is a simple example of an input routine.

```
65 rem Keyboard input routine
70 input "Input command ";C$:C$=upper$(C$)
80 if C$="K" then gosub 490: goto 70: rem Set speed
90 if C$="P" then 170: rem Play sample
100 if C$="R" then 250: rem Record sample
110 if C$="T" then 320: rem Thru sample
120 if C$="L" then 450: rem Load sample
130 if C$="S" then 370: rem Save sample
140 if C$="Q" then samstop: default: end
150 print "Command not recognized": goto 70
```

We'll now go through each of these commands in turn and show you how they can be implemented in STOS Basic.

Setting the sample speed

```
490 rem Set speed
500 input "Sample speed";S
510 if S<5 or S>22 then samstop:boom: goto 500
520 samspeed S
530 return
```

The SET SPEED routine uses samspeed to set a sampling rate between 5 and 22 KHz. Note the check at line 510.

Recording a sample

```
240 rem Record sample
250 if S=0 then gosub 490
260 gosub 550: rem position
270 samrecord start(6)+BEGIN,start(6)+FINISH
280 gosub 620: samstop: rem Display counter
290 input "Repeat sample?";A$: if upper$(A$)="Y" then 270
300 input "Reset interval?";A$: if upper$(A$)="Y" then 250
310 goto 70
```

This routine uses the SAMRECORD function to enter a sample into the ST's memory. The gosub at line 250 is just a call to the set speed routine I showed you earlier. This forces the user to choose a speed before recording the sample. The size and position of the recording is set using a special Set Interval routine like so:

```
540 rem Get starting point
```



```

550 input "Starting point";ST: BEGIN=ST*1000
560 if BEGIN<0 or BEGIN>MEM then print "Number must range between 0 and ";
MEM/1000: goto 550
570 input "Ending point ";SP: FINISH=SP*1000
580 if FINISH<=BEGIN then print "End point must be greater than start": goto 570
590 if FINISH>MEM then print "Number must range between 0 and "; MEM/1000:
goto 570
600 return

```

Each unit corresponds to a thousand bytes of the sample. So a start point of 5 and an end point of 50 will record from locations START(6)+5000 to START(6)+50000. Another useful routine is a Tape Counter which continuously displays the current sampling position.

```

610 rem Print current position
620 curs off: hide
630 XC=xcurs: YC=ycurs: P=samplace/1000+ST: print "Tape counter: ";P; " ";
if P<SP then locate XC,YC: goto 630
640 curs on: show
650 return

```

The equation $P = \text{samplace}/1000 + ST$, converts the position in the current sample to the position relative to the entire memory bank.

Playing a sample

```

160 rem Play sample
170 if S=0 then gosub 490
180 gosub 550: rem Get position
190 samraw start(6)+BEGIN,start(6)+FINISH
200 gosub 620: samstop: rem Display counter
210 print: input "Play again?";A$: if A$="Y" or A$="y" then 190
220 input "Reset interval?";A$: if A$="Y" or A$="y" then 170
230 goto 70

```

As with the Record function, I've added a check for whether the sample speed has been set previously. This compensates for the lack of the sample speed in the recorded sound.

Playing a sample through the loudspeaker

This can be performed using the following code:

```

320 rem Thru
330 input "Press play";A$
340 samthru: print "Press the spacebar to stop"
350 if inkey$<>" " then 350
360 samstop: goto 70

```

Finally, we'll add the routines which allow you to save and load samples to the disc.

```
370 rem Save sample to disc
380 F$=file select$("*.*","Save a sample")
390 if F$="" then 70
400 gosub 540
410 if dfree<FINISH-BEGIN then print "Not enough disc space":
print "Try again": goto 390
420 print "Saving ";F$
430 bsave F$,start(6)+BEGIN to start(6)+FINISH: goto 70
```

Note the call to the Set Interval routine at 540, which enables you to save any part of the sample independently.

Loading a sample

```
440 rem Load a sample
450 F$=file select$("*.*","Load a sample")
460 if F$="" then 70
470 print "Loading ";F$
480 bload F$,start(6): goto 70
```

Although the program is undoubtedly crude, it's more than adequate for adding the finishing touches to samples produced from MAESTRO.PRG. You can therefore fine tune your samples without having to leave the STOS Basic environment. Feel free to modify it as much as you like for use with your own programs.

6: Technical appendix

Adding sampled sounds to an existing program

If you've been using STOS Basic seriously for any appreciable length of time, you'll probably have dozens of programs which would sound great with a little sampled sound. This is especially true for those wishing to enter the STOS Gameswriter of the Year competition.

We'll now demonstrate how you can incorporate samples into a large STOS Basic program. In order to keep our example as realistic as possible we'll use the Zoltar game from the STOS Games Disc.

Switch on your ST and load Zoltar into memory with a line like:

```
load "zoltar\zoltar"
```

If you list this program you will be confronted with several pages of densely packed instructions.

The first step is to check which banks are currently being used by the program. You can do this using the LISTBANK command:

```
listbank
```

Reserved memory banks:

1 sprites S:\$0E9900 E:\$0ED300 L:\$003A00
3 music S:\$0ED300 E:\$0ED700 L:\$000400
5 data S:\$0ED700 E:\$0EF200 L:\$001B00
6 data S:\$0EF200 E:\$0EFF00 L:\$000D00

The precise numbers aren't really important and will vary depending on your system configuration. The only crucial fact is that bank number 5 is already used by the program – this clashes with the default memory bank used by the samplay instruction. You will therefore need to change the sample bank somewhere early in the program using SAMBANK. In the case of Zoltar the first unused bank is number seven. You can set this at the start of Zoltar like so:

200 reserve as screen 10:cls 10:sambank 7

You can now incorporate the standard pieces of initialisation code which will be required by your program.

42 click off:sound init:samspeed auto:rem Initialise sound chip

The next phase is to make a list of the various different sound effects which are currently used in your game. If the program is one of your own, this process will be fairly trivial. But in the case of Zoltar you will need to repeatedly scan the program with the STOS Basic SEARCH instruction. The table below shows the lines which require changing.

Breakdown of the sound effects used in Zoltar

Line	Sound commands	Purpose
10465	envel 1,1000 : noise 3,2	Player fires a missile
10485	envel 4,750 : noise 3,15	Ship fires a missile
10500	envel 1,4000 : noise 3,31	Attacker destroyed
10515	envel 1,20000 : noise 3,31	Player destroyed
10810	envel 1,1000 : noise 3,2	Player fires missile
10840	envel 4,750 : noise 3,15	Magedon fires missile
10845	envel 1,20000 : noise 3,31	Player destroyed
10900	voice off : volume 16: envel 1,35000 : noise 31	Magedon destroyed

The precise nature of these effects was discovered by simply calling them directly from the editor. Once you have isolated the sound effects in your program you can then choose some appropriate samples from the Samples Disc. These raw samples must now be combined into a single memory bank using the Maestro accessory. For the Zoltar game a useful selection would comprise of explosion, lazer and pistol. A file containing these samples was created during the tutorial on the Maestro accessory.

You are now in a position to incorporate these samples into your STOS Basic program. First load the samples into a memory bank with a line like:

load "effects.mbk",7:rem Load effects used by Zoltar

Now replace each of the NOISE instructions with the relevant SAMPLAY command. When these have been introduced try running Zoltar - you will quickly discover that the sampled sounds interfere with the music. As a general rule it's safest to avoid the combination of music with sampled sound altogether. Any accesses to the MUSIC instruction should therefore be removed. This can be simplified by checking through the program with the SEARCH commands.

1014 music 1

2005 music off : M=1 : T\$="Please select one option" : gosub 11000

5000 WAVE=1 : LIVES=3 : SCRE=0 : music 1 : gosub 10400 : music off

Alternatively, you can halt the music with MUSIC FREEZE before the sample is played, and then restart it using the MUSIC ON command. So the SAMPLAY instructions could be replaced by:

music freeze:samplay 1:wait 50:music on

Unfortunately this simple-minded approach will only work for games which use interrupt-driven sprites as used by the majority of STOS Basic programs. Zoltar, however, manipulates the sprites solely with the SPRITE command and the game will be halted for about a second after every explosion. Also note that the above technique requires you to resume the music after the sample has completed. Otherwise the music will fail to resume after the MUSIC ON statement.

Here are a few possible extensions to the Zoltar program:

- 1 Add sampled voices for "GETREADY" or "BONUSPOINTS": See lines 1155 and 10405 in Zoltar.
- 2 Give each possible sound a different sample.

Finally I'll summarise the various steps in the modification procedure.

Adding sound to a STOS Basic game

- 1 Check for the existence of bank 5. If it is already in use, set the sample bank to the next free bank using SAMBANK.
- 2 Add the initialisation code to the start of the program. This will typically consist of the following instructions.

click off:sound init:samspeed auto

- 3 List the various sound effects used and create a bank of replacements using the Maestro accessory.
- 4 Add the new effects to the appropriate points in your program.
- 5 If your program is using music you are presented with two possible options.
Either remove the music completely, or halt the music using MUSIC FREEZE before a sample is played and resume with MUSIC ON after it has finished.

Assembly language interface

If you have a little experience in assembly language, you will be delighted to discover that any samples created using the STOS Maestro Plus cartridge can be replayed using just a few lines of assembler.

In order to access these features, you will need to incorporate the code found in the MAESTRO.OBJ file into your program. This can be loaded at the end of your program using the INCBIN directive within your source file.

sampler incbin "maestro.obj"

If your assembler does not include this facility, you will be forced to load the object file directly from the disc in your assembly language program. Full details of the required BIOS calls can be found in any decent ST reference manual.

Since the sampler uses an interrupt routine, it cannot be guaranteed to work with any existing interrupts which access the sound chip.

The calling procedure is simplicity itself. First define a number of important system variables at the start of your program like so:

Equates for accessing the routines and variables

SAMINIT	EQU \$000	; Offset to the initialisation routine
SAMPLAY	EQU \$020	; Offset to the playback routine
SAMSTOP	EQU \$370	; Offset to the samstop routine
AUTO	EQU \$3A2	; Offset to a variable holding AUTO status
TYPE	EQU \$3A8	; Offset to variable holding the sampler mode
		; TYPE(A6)=1 for forward play
		; TYPE(A6)=2 for backwards play
		; TYPE(A6)=3 for forwards loop
		; TYPE(A6)=4 for backwards loop
		; TYPE(A6)=5 for sweep

SPEED EQU \$3AA

; Offset to variable controlling the speed. Permissible values
; range from 5 to 32 KiloHertz. Note this option only works if
; AUTO(A6) is set to ZERO

Now load the address of the sampler routine into a convenient address register.

lea SAMPLER,a6

You must then initialise the sound chip using the SAMINIT function.

jsr SAMINIT(a6) **; Call routine to initialise sound chip**

Finally enter supervisor mode with a call to SUPER and play your sample with:

lea SAMPLE_START_ADDR,a0	; Start Address of sample
Move.l #SAMPLE_LENGTH,a1	; Length of sample in bytes

lea SAMPLER,a6	; Address of MAESTRO.OBJ code
move.w #1,TYPE(a6)	; Forward play
move.w #0,AUTO(a6)	; Set manual mode
move.w #10,SPEED(a6)	; Set sample speed to 10KHz
jsr SAMPLAY(a6)	; Call playback function

You can stop this sample at any time using the instructions:

lea SAMPLER,a6	; Start address of .OBJ file
jsr SAMSTOP(a6)	

Here are a couple of Macro definitions which simplify this procedure:

```

;PLAYSAM Play a sample
; Usage: playsam SAMPLER,START,END,SPEED
;SAMPLER = Start address of MAESTRO.OBJ file
;START = Start address of sample
;END = End address of sample
;SPEED = Speed of sample (in kilohertz)

```

```

SAMPLAY:MACRO LEA\1,a6      ; Get address of OBJ file in a6

```

jsr SAMINIT(a6)	; Init sound chip
move.l #2,a0	; Start of sample in a0.L
move.l #3,a1	; End of sample in a1.L
move.w #4,SPEED(a6)	; Speed of play
clr.w AUTO(a6)	; Not auto speed select
move.w #1,TYPE(a6)	; Play forwards (TYPE=1)
jsr SAMPLAY(a6)	; Call routine

```

ENDM      ; End macro

```

```

;SAMSTOP Stop a sample
; Usage: samstop SAMPLER
;SAMPLER = Start address of MAESTRO.OBJ file

```

```

SAMHALT:MACRO

```

lea\1,a6
jsr SAMSTOP(a6)2

```

ENDM

```

If you use these macros, don't forget to enter supervisor mode first! Other languages which incorporate some form of machine code call command will be able to take advantage of the MAESTRO.OBJ file – simply set up the relevant variables and call the code.

STOS Maestro Plus cartridge specification

Signal to noise ratio	30.48 Dbs
Maximum frequency	98.5 KHz (not guaranteed)
Frequency Selection	Software controlled
Resolution	8 bits
Linearity	+/- 0.5 LSB
Maximum input signal	2.5 V RMS (peak to peak)
Anti-aliasing filter	Yes

Internal successive approximation logic circuitry.

Chip conversion activated by ROM-SEL 3 on cartridge port.

