

SYDTRUG NEWS

SYDNEY TRS-80 USERS GROUP NEWSLETTER

Volume. 4 Issue. 8 APRIL 1984

IN THIS ISSUE

We begin this months newsletter with a few words from our Secretary.
A variety of articles supplied by Stefan Keller and Chris Tinney.

The first being :

SLOW DOWN mod. allows you to slow down games etc.

DISK ENABLE mod. Disables the accessing of the disks when Slow Mod is on.

EXTRA KEYS mod. Gives direct key input of ASCII 91-95 and 123-127

Lindsay Douglas once again supplies us with an article.

PORT FD DECODER for LNW BOARD.

Back to Stefan and Chris for a comprehensive article on:
COMPUMUSE MUSIC SOUND SYNTHESISER.

TAPE HINTS by Jim Whittaker.

Finally some Bulletin Board news.

GENERAL.

ALAN BIZY is our new Distribution officer and you should have received your newsletter before the first meeting of each month. (providing I get it to him on time)

Thanks Alan and to all contributors to the newsletter.

How about every member making an effort to provide at least one or two articles a year to the newsletter.

Surely everyone must have some small piece of worthwhile material to submit. (Even gossip of a humorous nature).

Gary Bryce's temporary transfer to Melbourne has been extended and I am not sure when he will be back in the Editors chair.

REMEMBER No news no newsletter.

You can get in touch with me at the Club meetings or at home on 773-4433 if your not sure how to submit the articles.

MEETING DATES.

The dates for the forthcoming meetings in April and May appear below.

All meetings commence at approximately 1:00 p.m. at the rear of Patterson's Florist Chegwyn St. Botany.

APRIL

14th Monthly Meeting

MAY

12th Monthly Meeting.

19th Special Interest Group

The contents of this publication are copyright (c) 1983 by SYDNEY TRS-80 USERS GROUP
All rights reserved. Material appearing in this publication may be reprinted in similar computer club newsletters and nonprofit publications if accompanied by the following notice:

Reprinted from SYDTRUG NEWS

SECRETARY'S SECTION.

This is just a few ramblings from your faithful Secretary.

Firstly let me assure you that the committee members are working for the benefit of all members of the Club and if you have any problems we would like to be the FIRST to hear about them. Any comments, grouches or remarks will be heard by me (at least) and any action we take will be in the best interest of the club.

As a matter of house keeping I intend to circulate a PUBLICATION list of members. Any member who has not already ticked the NFP slot on the application form may restrict the right to publish their name etc. by contacting me either at meetings or at home.

The PUBLICATION list will show the following details: NAME; PHONE; SYSTEM; DRIVES; PRINTER; INTERESTS; RENEWAL DUE DATE. Any members who have changed their details or do not receive the SYDTRUG newsletter, should contact me for an update on 772-2009 or at the meetings.

Club membership presently stands at 147 and one of our aims is to increase this number so I have been sending a personal letter and a copy of SYDTRUG newsletter to those who send enquiries. If you know of any person who may be interested in the activities of the SYDNEY TRS-80 USERS GROUP then again please get in touch with me.

Another reminder is that we have to hold to hold an Annual General Meeting as per the Club Constitution. All positions will be open for nominations. Please be aware that a group is only as good as its management and that management can only progress and improve the group by the full backing, support and cooperation of the members. In the past great managers have passed on BECAUSE of the APATHETIC stance taken by the members.

REMEMBER:: ITS UP TO YOU.!!!!

JIM WHITTAKER. 772-2009
Secretary.

SLOW DOWN MOD / DISK DISABLE (reset enable) MOD / EXTRA KEYS.

by Stefan Keller and Chris Tinney.
Stefan Keller 533 - 1612
Chris Tinney 759 - 5052

After having read the SYDTRUG news letter for a year or so, we decided that it was high time that we submitted something. So here is our effort. The article consists of a few mods that we and a few of our mates have made to our systems. The mods are not major, and can be implemented in an hour or so. Here goes.

INTRODUCTION

"Why the #\$%*& would anyone want to slow down his CPU" we hear you say! Most people want their computer going as fast as their RAM can multiplex - so why slow it down? Well Rally Racer is a pretty fast moving game..... The slow down mod will slow your box down to around a fifth or a sixth of its original clock speed, and it can be varied anywhere between this slowest speed and the full CPU speed.

The disk enable mod needs a little explaining. Remember back to the days when you had a 16K tape system.

What happened when you pressed RESET?

Well this mod (a switch) will let you disable the disk controller so that RESET works as it used to. Its main use however, is with the slow down mod. Trying to write a sector or two with the apparent CPU speed at 400Khz is like rubbing a 14" speaker magnet over the back of your disk box. We know, having once tried to change PDRIVES at half speed. The Disk Controller Disable will prevent all disk operations, while the CPU is being slowed down.

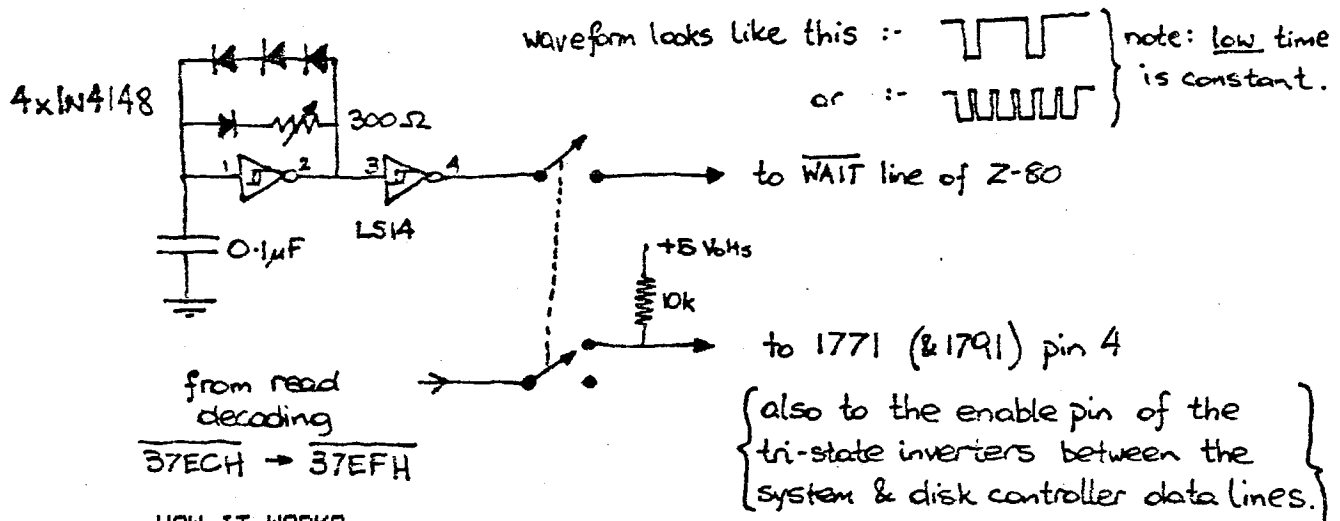
SLOW DOWN

To make the slow down, you will need the following components.

- 1- 74LS14 chip
- 4- 1N4148 diodes
- 1- 0.1uF greencap capacitor
- 1- 300 ohm pot (anything close will do)
- 1- DPST switch

Wire them up as follows :-

Slow down mod circuit.



HOW IT WORKS.

The 1st schmitt inverter is wired up as an oscillator with a variable duty cycle. Its output is HIGH for a constant time, and LOW for a time determined by the setting of the 300ohm pot. The output of this gate goes through another schmitt gate so as to 'clean it up' and to invert it. The resultant waveform is applied to the WAIT line of the Z-80.

When the WAIT line goes LOW, the Z-80 stops doing everything (even refreshing the RAMS, that's why you can't slow down to 1 Hz). In this mod, a high frequency signal is applied to the WAIT line. The CPU will, therefore, stop every 10 to 50 microseconds (depending on the position of the pot), for a couple of microseconds. The result is that the Z-80 doesn't quite get as much done, ie it slows down.

DISK ENABLE MOD.

As mentioned before, it would be wise to disable the disks when the slow down is on (just to be safe). This can be done by first finding pin 4 on the disk controller chip(s) (1771 (& 1791)). This pin is the read enable pin of the controller chips and goes low every time the disk status byte is read (37ECH) or any of the controller's registers are read (37EDH - 37EFH).

Follow this line back to where 37ECH is decoded. This will be the output of a chip (probably a 74LS10 - that's what it is on ours).

If the output of this chip is isolated (via an open switch) and pulled high (via a 10K resistor), then every time you try to read from any of these locations, you will get a 255. This makes the DOS system think there is no FDC, so no disk operations will be performed and the Reset will merely force a jump to 6CCH.

We built the 80 MICRO disk controller (December 1982) which included a 'reset enable' switch, so you could look up the article to find out more about it.

If you also build the slow down, series this switch with the second pole of the Slow Down switch. This means slowing the machine down will automatically turn the FDC off, so protecting your valuable disks.

We'll leave it up to you as to how the mod(s) will be fitted, but it's probably best if you keep the wiring for the reset enable as short as possible. If you have an expansion interface, it's probably best if you mount the mods there. If you don't have disks, then the slow down can be put anywhere.....inside the CPU or outside in a box, it's not really critical. Consult the technical manual for your CPU to find the best place to hook up the slow down to the wait line. On the edge connector is probably the easiest.

EXTRA KEYS .

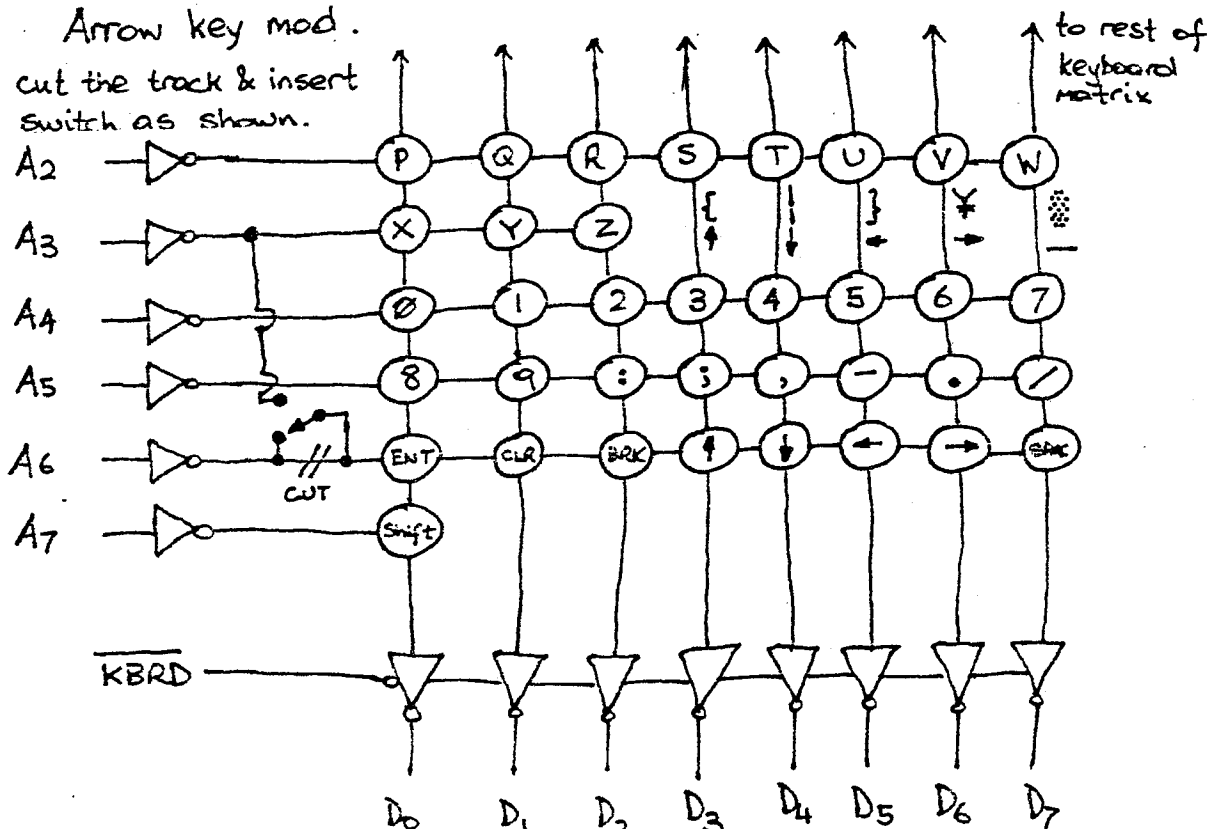
Another easy modification for the Model I, is the installation of a switch enabling the direct input of ASCII codes 91-95 (the arrows) and 123-127 (other rubbish like braces). If you use PASCAL a lot, or dial up your local VAX every now and again, then you will find this mod useful, as they require these ASCII codes often.

Have a look at the keyboard matrix, you will notice that there are only 3 key switches in the XYZ row. The missing keys are those which decode the extra ASCII codes. Since the missing codes are for the arrows on the TRS-80, a switch can be installed so that the Arrow Key row is paralleled (??) with the XYZ row.

The mod is mindlessly simple. Get a SPDT switch and a sharp instrument (for cutting tracks). Cut the A6 track (common to the Enter, Clear, Break, Up-, Down-, Left-, Right-Arrows and Space). Now connect as per the diagram above. (The actual layout will vary with the model of your machine.)

Arrow key mod.

cut the track & insert switch as shown.



Now, when you switch in the mod, the arrow keys will give you arrows, the space bar will give you a cursor character, and shifting any of these will give those other characters. As an ADDED BONUS, those of you with X Y and Z keys that don't perform up to the standard of the other keys on your keyboard will be able to use the Enter Clear and Break keys as quasi X Y and Z keys..... so don't panic when you forget to flick that switch back to it's normal position and you find that Enter doesn't work properly.

One last thing before we finish. Here is a question that has puzzled me for a long time. Why are there so few members of the fairer sex who are interested in Micros? Looking at the membership listing we all received, not one female name could be spied! (Although some come close.) Never have I seen any females at the USER Group meetings!

Now before Senator Ryan hacks me into little bits I should point out that I don't think this is good or bad - but very, very, very puzzling. In the words of a fuzzy haired chocolate salesman ;

"WHY IS IT SO ??"

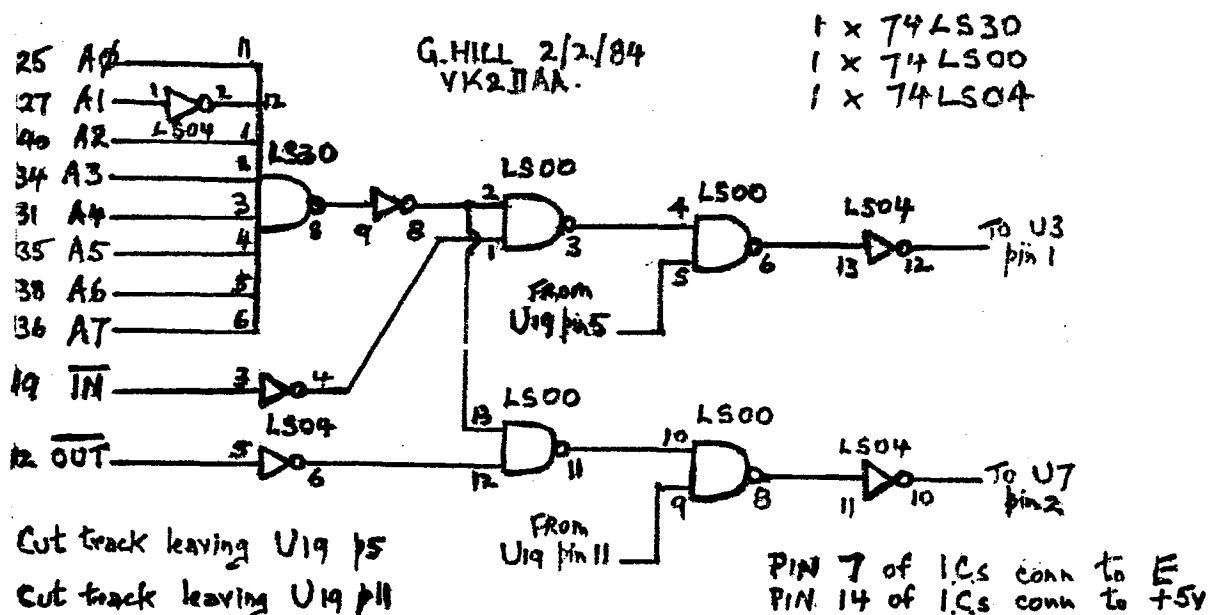
PORT FD DECODER FOR L.N.W. BOARD.

Lindsay Douglas. VK2 ON
043 25-2254.

The Accompanying circuit is self explanatory.

It enables the LPRINT command to function with the SYSTEM-80.

The circuit was built on a piece of Vero-board and placed on top of the L.N.W. board.

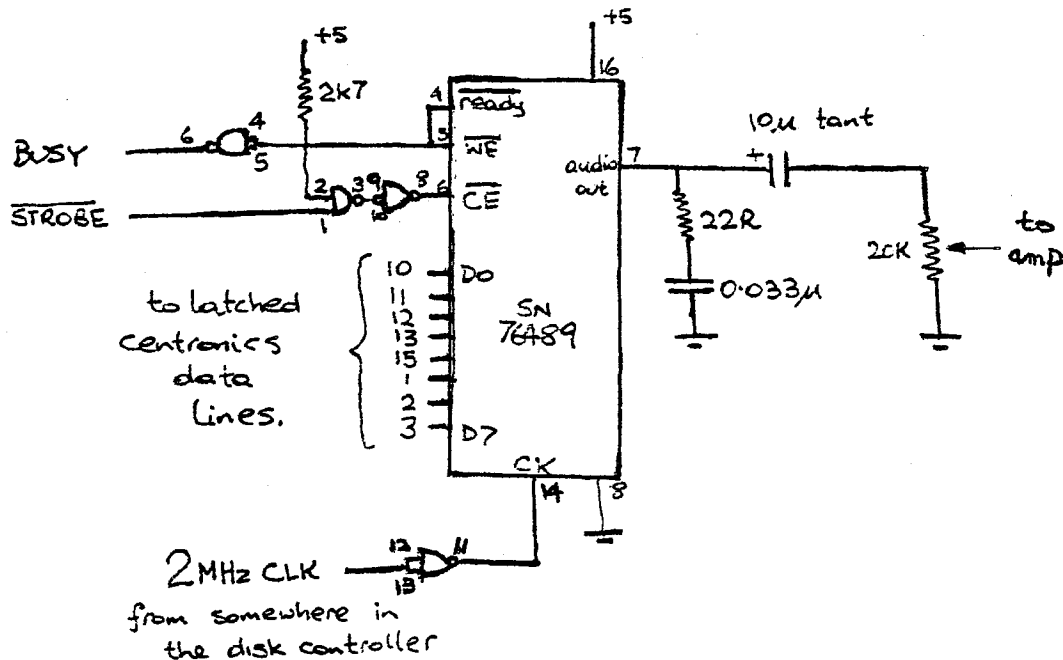


PARALLEL PRINTER DECODING FOR LNW BOARD TO
USE BOTH PORT FDH AS PER SYSTEM 80 AND 37EB4 AS PER TRS80

Some of more hardware oriented of you, who read Electronics Australia, may have noticed the Compumuse sound synthesiser project in the August issue last year. Attracted by its low cost, we set about modifying it to suit the TRS-80's system exactly. When you construct it, we are sure that you will agree that the range of sounds and their quality are exceptional for the price of the SN76489 sound generator and the handful of other components that you will need.

If you take a look at the circuit :-

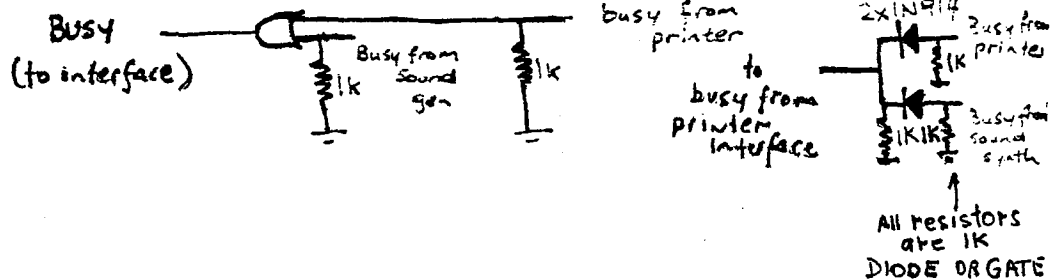
The Centronics Port in the Expansion Interface is ideal for this.



If you don't have a parallel printer (like me) then the lines can be soldered directly to the printer interface : but if you do, you can make up a 34 pin edge connector like the printer's one and just plug in the one you want.

Or wire this up and turn on the printer or the sound generator.

OR wire this up
& turn on
either the printer
or sound gen.



You will notice that it is nothing more than a 74LS00, the SN76489, a couple of capacitors, resistors and a trimmer pot. As an amplifier, you could use your cassette recorder, or (as we have) your family sound system (if it is nearby). (50 watts per channel with the bass and loudness control on full is really great for those deep space X-wing laser cannon explosions!) This is the first change that we made to the EA circuit. We figure that if the wheel has already been invented, then there's no use in trying to do it all again and build another amplifier.

The second main change to the circuit is the clock. As you already have a 2MHz clock sitting in your disk controller, why make another? After all, you would need another 2 chips if you did that! The clock frequency could of course be anything from a few hundred KHz to 4MHz, but as Peter Vernon (the EA author) points out, 4MHz gives you good frequency accuracy but no bass sounds - and 1MHz gives you great bass but shot frequency accuracy. With a 2MHz clock you can produce frequencies between 61.03Hz and 62.5KHz at a very acceptable frequency accuracy (5 full octaves for those of you that want to play music).

The only other change to the EA circuit concerns the BUSY signal going back to the centronics interface. The EA circuit, being a little short of inverters, sends an inverted BUSY back to the printer interface. This is OK if you want to POKE the commands out to the generator, or if you don't have a printer, BUT if you'd like to LPRINT commands, or have your printer connected all the time (saves mucking about with 34 pin connectors) it is best if the BUSY signal remains conventional.

Well, its about time that you got that soldering iron of yours smoking again, so hop to it! I (Stefan) constructed mine on a small piece of wire wrap board, while Chris made his on some veroboard. Either way, use 2 sockets for the chips (please). It means that if you ever want to modify the circuit, or move it onto a new board, it won't cost you another SN76489.

A short length of 12 conductor ribbon cable is good enough to run back to the printer interface, and a short length of shielded cable should be used to run the clock signal back from the disk controller. We're sorry that we can't tell you where exactly to get the clock signal from. It's just that we have home brew interfaces and haven't ever set eyes on the insides of another interface. A quick look at the technical manual for your interface though should be all it takes for you to spot it. The disk controller chips run off 1MHz and the clock will either be 4MHz or 8MHz, so it shouldn't be too hard to tap into the clock divider chain somewhere. (We obtained ours from the output of a 74LS74 wired up as a divide by 2)

Now, those of you with printers will be wanting to know how to rig up both the printer and the sound generator to the one centronics port. Well, the sound synthesiser can be soldered directly to the interface. The only real problem with this is the 2 BUSY signals - how can they be managed? They will have to be logically ORed so that the printer port will signal BUSY when either the printer or the sound generator is busy. You can either piggyback a 74LS32 (quad 2 input OR gate) or wire up a diode OR gate as shown above. Don't forget to include the pull down resistors.

Obviously, the old BUSY line running from the 34 pin edge connector to the buffer chip in the centronics interface will have to be cut. The new ORed BUSY can be connected to the buffer chip, and the BUSY line coming from the printer can be connected to one of the OR gate's inputs.

You are probably thinking (by now) that having the two BUSYs connected together will cause some problems with your lightning fast printer. Think again! With a clock frequency of 2MHz to the SN76489, the BUSY line from it will stay HIGH for around 32 microseconds. So for the synthesiser to interfere with your printer's BUSY signal would require a 32000 characters per second printing rate. We feel that most modern fast printers would find it a little difficult to keep up with this rate (not to mention the ROM in conjunction with your Z-80).

If you don't want to use the printer, then simply turn it off, and your box can spit out commands to the synthesiser as quickly as its CPU clock can cycle (almost). If you'd like to get a more lasting record of the data that you send to the centronics interface, then all you have to do is turn on the printer..... you could even leave the sound synthesiser on if you like that sort of thing. (????!!)

But to start you off in the world of composing, we have entered a few short BASIC music programmes (including the 2 EA demo programmes). These are at this time just sitting in our disks' sectors just waiting to be COPYed (well I neva cood spel eniwai) Also, there is a BASIC file of DATA pertaining to the values that have to be sent to the sound generator to obtain each note of each octave for each of the voices. You may find it useful in manufacturing your very own loony tunes.

You may, however, be wondering how to program the SN76489 by yourself. Well we certainly did and here the EA article is fairly lacking. You can (if you are a sado-masochist) go through the article and make your own summary of their dissertation - or you can read mine.

SN76489 - Programming Commands

Instructions sent to the chip are either 1 byte or 2 bytes long. Frequency commands for the three voices take 2 bytes.

Volume commands and the frequency select for noise take 1 byte.

The chip considers there to be 4 voices. Voices 1-3 and the Noise source (or Voice 4). Commands are either frequency commands or volume commands and are sent to one of the 4 voices using the following format :-

Byte 1

Bit 7 : 1 => this is byte #1

0 => this is byte #2

Bit 6 : 1 => Set Volume

0 => Set Frequency

(Bit 6 tells the chip what type of command this is)

Bits 5,4 : a register (ie voice) address field

0 0 => Access voice 1

0 1 => Access voice 2

1 0 => Access voice 3

1 1 => Access voice 4 (ie noise)

Bits 3-0 : For a volume command Bits 3-0 are a 4 bit volume field 0-15. (0=Min (off), 15=Max)

: For a frequency command on voice 4 (noise)

0 0 - freq = clock / 512

0 1 - freq = clock / 1024

1 0 - freq = clock / 2048

1 1 - freq = freq set by voice 3

: For a frequency command on voices 1-3

Bits 3-0, together with 5-0 of the 2nd


```

        byte form a 10 bit frequency field
        freq = clock / (32*10 Bit field)
        Bits 3-0 = 4 LSBits of this field
Byte 2
    Bit 7 : 0 => Byte #2 (Must be this!)
    Bit 6 : Don't care.
    Bits 5-0 : 6 MSBits of the 10 bit freq field, the
                4 LSBits of which were given in the preceding
                command.
    
```

And that's it. Its very simple really. Simply select a voice no., whether its volume or freq, and give it the appropriate commands. These can be Poked from BASIC, or LD'ed from machine language. So - get playing!

Just a final warning. NEVER set your power amp's volume to 5073 watts per channel (esp. if wearing headphones) and turn your sound synthesiser on. The SN76489 powers up in a random state, and you really wouldn't want that GREAT 3 voiced chord + electronically produced noise pumping through those miniature orifices at the side of your head so soon after power up. It might also be a good idea if you had an AUTO file to set all volumes to 0 as quickly as possible after booting.

TAPE HINTS.

J. Whittaker 772-2009

```
10 FOR X=1 TO 10: PRINT#-1,X : NEXT
```

Type in and run this small program and see how long it takes. Have you ever let that large Data Base program go by just because people have told you that it would take forever to load. Well think again. Below are some tape hints that I have learned by gleaning information from many and varied sources.

1. TAPE LEADER.

The format used to 'print#-1', firstly writes a 255 byte leader (i.e no data) and then a sync byte. This takes approx. 4 seconds and allows the tape motor to reach a stable speed. However the current tape recorders need only about 1 second. The saving is obvious, if you have 100 pieces of data to save you will save 300 seconds. As the routine is only 25 bytes long and you are not using disks, put the code into a free area of the System RAM at 405EH to 4074H where it will not interfere with any other programs.

Just make the first two lines of your program read:

```
10 FOR I = 16478 TO 16502: READ Q: POKE I,Q: NEXT I: POKE 16842,195: POKE
16843,94: POKE 16844,64
```

```
20 DATA 254,35,192,205,254,1,6,64,175,205,100,2,16,251,62,165,205,100,2,62,128
,50,15 6,64 ,201
```

Be aware that if your program reads any other DATA you may have to do a dummy read to increment the data pointers.

Now type in :

```
30 FOR X=1 TO 10: PRINT#-1,X: NEXT
```

and see if there is any time improvement. Note that 'INPUT#-1' is not affected.

2. PACKING the DATA.

When the Com writes DATA to tape, it sends it in lots of 255 bytes. v

If your DATA is only 10 bytes long, example A\$(1)="surname #1" it will write 10 bytes of the surname and 205 bytes of nothing. The same thing will happen if you write more than one name

(example FOR X=1 TO 10: PRINT#-1,A\$(X): NEXT

The write will be 10 bytes of NAME1 and 205 bytes of ZERO, 10 bytes of NAME2 and 205 bytes of ZERO etc.

Obviously what you have to do is to pack the strings before they are written.

You can do this by following the example below:

40 FOR X=1 TO 10 STEP 4: PRINT#-1,A\$(X), A\$(X+1), A\$(X+2), A\$(X+3): NEXT

The STEP you take will depend upon the length of your DATA eg. if A\$ is 60 bytes long you can pack 4*60 =240 bytes into each write. If A\$ is 100 bytes long you can pack 2*100 =200 bytes into each write.

BULLETIN BOARD NEWS.

The Sydney TRS-80 Users Group bulletin board "CLUB 80" is now operational. and has approximately 54 programs listed in its Catalogue.

BULLETIN BOARD LISTING.

Phone No.	Name	Operators.
02 332-2494	CLUB-80	SYDNEY TRS-80 USERS GROUP.

For a list of current Bulletin Board phone numbers ring the above B.B. No. and access a file named AUSTPAMS/TXT.

FOR SALE

Disk Drive Pertec 40 track single sided in original case with power supply. Just been fully serviced.

Factory equipped as "FLIPPY". Price \$250:00 no offers.

Contact Paul Darville. (02)628-3902.