volume four/ June 1st, 1985 number 20

bi-monthly periodical of the ***Exidy Sorcerer Gebruikers Groep***

a translation in English of the original Dutch version

ESGG

The L O G I C A L partner to a Sorcerer

Subscriptions : per annum f. 27,50 (Europe)
f. 32,50 (other countries)

Subscription administration :
see page 2 (first column)

Secretariat ESGG Foundation :

Send your copy to : redaktie ESGG
p/a postbus 510
1000 AM AMSTERDAM Holland

Coordinator foreign relations: C. Boone
Stationsplein 26
B-9100 LOKEREN Belgium

****

CONTENTS OF THIS ISSUE

****

EDITORIAL STAFF.

chief editor : Welmoed J. Jonker.
ass. chief editor : Theo Huijgen.
hardware editor : Rob Borkent.
software-editor : Kees van Duijvenbode.
general editor : Don Siahaya.

SUBSCRIPTION.

You become a subcriber when transferring the fee due to postcheckaccount number 5368539 or bankaccount number 337107793 of ESGG at Lopik, The Netherlands. Transfer by postal money order also is allowed. Payments other than the afore mentioned will be refused! Refer to 'subscription periodical'. Subscriptions run from June 1st of the current volume.

SUBSCRIPTION-ADMINISTRATION.

Send your change of address and complaints about the delivery to:

Secretariat ESGG Foundation
Administratie ESGG periodiek
Prins Hendrikstraat 3d
3071 LG ROTTERDAM Netherlands

ADVERTISEMENTS.

macro's: *exclusively for business.*
acquisition: H. Herstel
Mauritsingel 29
3135 JM VLAARDINGEN.

micro's: *Only open to private persons.*

Size : -a line of text consists of 66 characters or spaces.
-there is a maximum of six lines for each advertisement.
Cost : the cost per two lines of text is Dfl. 3,00.
Entering: send your advertisement to the editor before the first day of each odd month. Mention also the total of 66 character lines and your account nr.
Payment : Remit the amount due to postal check account number 5368539 of ESGG at Lopik, referring to 'micro's'.

If your remittance is not received before the first day of an odd month insertion in that number is no longer possible!

COPYRIGHT ESGG.

SOFTWARE COLLECTOR.

If you like to offer yourself developed so-called public domain software to your fellow members, send it to:

Cassette library:
Wim Warning
Vogelweide 83
3815 HE AMERSFOORT

diskette library:
Hermine Bakker
Falklanddreef 18
3563 AC UTRECHT

The Netherlands

ESGG-SERVICE.

Prices are including postal rates for second class mail all over the world. The language of explanations in the software is mainly Dutch! Because of transport risks no mail-order possible for diskettes! Advance payment and collect at Sorcerer Days possible!

Ordering : Only by transfer to postaccount number 5368539 or bankaccount 337107793 of ESGG at LOPIK, referring to: ESGG-service.

in your order state name and quantity of the desired article.

you do not receive an acknowledgement of your order.

if the article is not in stock or no longer available THEN you will be informed so!

Ordering diskettes: see restriction above.
Catalog obtainable from CP/M-users group Netherlands.

Available formats are 77 tracks hard- and softsectored, 40 and 30 tracks softsectored. The last-named two formats are 2 and 3 disks respectively. We always send the mentioned quantity of diskettes to you (possibly only formatted).

Non ESGG members and non subscribers pay Dfl. 10,= extra per vol.

Guarantee: Electronic articles from ESGG are subject to guarantee for proper operation. ESGG is not liable for damage caused by incorrect installation by others than the official technicians at Sorcerer days.

Below is a list of articles available at this very moment:

name article (prices a piece!)	Sorcererday-collect	by mail
1. Collect-cassettes with various programs (nrs. 1 to 20)	Dfl. 7,50	Dfl. 15,00
2. Collect-disks *) with various programs, per volume: 77 HS/SS	Dfl.25,00	
40 SS	Dfl.30,00	
30 SS	Dfl.40,00	
*) see: ordering)		
3. ESGG diskettes 1 - 12	as in pt. 2	
4. Eprom Basic EXTension (version 8) with description of installation and manual		Dfl. 35,00
5. Manual Bext		Dfl. 5,00
6. Invers video print (assembled)		Dfl. 20,00
7. Lightpen ESGG		Dfl. 50,00
8. List cassette software		Dfl. 6,00

INPUT.

a column to ask questions and also to give your opinion or comment.

If you have a problem, describe it as clear as possible and send it in a post-paid envelope to the editor. Our team then will try to find a solution. We claim the right of publication for question and solution in our magazine.

Editorial.

When the board of ESGG decided in 1982 to start and issue a periodical, we could not have guessed its high flight.
Now we are at the start of the fourth volume and it is not peculiar to stop and look at the past. In 1980 the Exidy Sorcerer Users Group had been founded. At that time there were some 200 people present, giving you an idea about the crowd we got in in that small room in hall 'Turfschip' at Breda.

With the drive, familiar to hobbyists a start had been made developing the group. First there were not much ways to communicate from the board to the members. The only way was by means of the HCC (Hobby Computer Club) Newsletter. This magazine however was meant for all of the user groups, so the make-dedicated users groups did not profit too much. We did not like that situation and also intended to inform the members better and more about their system. This lead to the decision to issue a club magazine. The proof of its being needed became clear after häving held an inquiry, in which the member's majority wished to have their own magazine even when they had to pay for it!

Upon issueing the first number at the March 1982 Sorcerer Day the first edition had been sold almost completely and over 400 members did use the offer of subscribing to it. From that day on the number of subscribers did increase. In order to be able to copy from other club magazines, a decision had been taken to translate the periodical. It goes without saying that a lot of foreigners used the possibility of subscribing to the English version of our magazine. The number of English speaking subscribers however is a small number compared to the number of subscribers to the Dutchperiodical. It is however a sign that there is a need for Sorcerer dedicated information!

Last year we have again had an inquiry to get information about the system composition within the users group. The outcome is to help and point the way for both board and editor in future. We will bear in mind the demands and requests of the members as well as the fact that we are on our own now. Not peticularly because the Exidy no longer will be manufactured for the market for the manufacturer never paid attention to us or had us helped out. Being on our own mainly is in the software field. The members are developing software of good quality in a sizable speed which is a positive fact that will contribute to a continueing use. We however feel the suction from new systems having more memory and possibilities. Some among you must join progress for reasons of decay in knowledge, others are forced to join as their school or company is trading old for new. This is a hard to stop development.

Lately one can note that a new line of users is coming in: The time of program development seems to be over, the new hobbyist often only is a user. We regret this development as we actually give information but this information is often lost as one does not feel for research. The 'old force' however can not be stopped, thus increasing the number of options of our Sorcerer.

With this introduction I hope to stimulate the new members and subscribers in entering the research field exploring all hidden aspects of the Sorcerer. By actually experimenting with your Sorcerer it will become obvious that you do not own some computer! The unique qualities (and habits) of your system make that you now, but also in future, have a computer that seems to be somewhat older but being extremely usefull!
Foundation and Users Group will assist in exploring the new possibilities. You, at your end can help by offering your knowledge to us. Bringing together your and our knowledge will help to get the new users on the right

track.

I wish to you a new volume with numerous and important information. The future seems to be bright enough!

Welmoed Jonker.

****

I N F O I N F O I N F O I

* Again a volume closed and a new one entered. With your aid we will try to create the best of it. This of course does mean that you can not do with passively standing aside and merely *read* the periodical. We also need *your* input!
  This input does not have to be a learned piece of writing! a question for a solution of a problem also can provide an answer important to one or more readers. In such a case it is necessary that your question is provided with sufficiant information in order to have a clear image of the problem!

  What information is need in such cases? At least a system description and an explanation of the circumstances when the problem occurred. If you are able to send examples, this is appreciated! All that can contribute to the solution is needed.

* At the last Sorcerer Day a number of brand-new diskdrives had been sold for almost half the actual price, thus increasing the number of CP/M users again. Using new equipment usually gives some problems, before one is somewhat familiar with the way of use. If you have problems, do not hesitate to bring it out in the open! You almost certainly are not the only one, so the solution fitted for you may also help others!

* The editors team is getting complete now. Upon the departure of Aad van Duijvenbode, who's job became to busy, we now have Rob Borkent who is going to fill the gap. Rob had a technical training at the school for engineers (HTS) and regularly is busy developing electronic devices.
  We hope Rob will feel at home in our team.
  We are sure that your questions in the hardware field will be treated with skill by him.

* S O R C E R E R D A Y N E W S
  It is always good to know in time when the next Sorcerer Day is going to be held!
  As usual this day will be held in hall KUNSTMIN at Boelekade 69 at Gouda, from 10-16 hrs.
  Now this is when: 7 **September 1985**!
  Keep this day available in your diary? Be seeing you at Gouda!

* With the announcement of the Sorcerer Day the request for assistence in the organization also is out. If you like to spend some hours or the entire day for you club, do call on our secretary, mr. Charles Netteler at 010-330493.

* The next issue of the periodical will have more information on the activities of 7 September coming.
  One of these activities is surely the newcomers corner! It became clear at the previous Sorcerer Days that there is a need for starter's information.

* It will be a good thing to know for users of Pascal that there also has been founded a Pascal users group. The board of that group intends to

give support in any level in the use of Pascal. If you like to support their task you may address to the secretary of the Pascal Users Group, mr. Jan Leijerweerd, Laan van het Kinholt 41 at Emmen.

* Our English subscriber William H. Askew likes to get in touch with other Sorcerer users. Readers that like to do so are invited to write to William at the address 40 Holly Lane, Cliftonville, MARGATE, Kent CT9 3ND, England.

* If you did purchase cassette 20 at the latest Sorcerer Day you also have received a manual for some of the programs. We regret the omission of the name of the creator of the programs Loupe and DoAll. By this we thank mr. J.J.A. van Asten of Leiden for his contribution.

* We have call attention to the following: In the program Loupe there is a disassembler routine that has been taken from the program Expan by the Van Montfort Brothers. We have not investigated whether permission had been given, as has been mentioned in the copyright clause this is the responsability of the sender of the program.
Having been called to the presence of the disassembler as well as to the missing approval, we have been talking to the Brothers and we are glad to have gained there co-operation. They did not like to cause trouble for us refusing admission. We thank them for their kindness.
THe situation makes that we anew have to point to the fact that in case you like to contribute to the library, you need to provide an admission of the original author in case the program consists of parts of other routines or programsparts derived from other programs. In this way you protect yourself against the consequenses of having violated the copyright.
This will prevent a lot of inconvience for you and us!

* We also like to point to a different way of purchasing software diskettes than the mail route. At CP/M Days the ESGG always is present with the entire library. You then can -almost without delay- provide yourself with all of the software released up to that moment in all known Sorcerer formats (except for 8"!).
The next coming CP/M Day is to be at Saturday 26 October 1985. This is a unique occasion to update their library or learn about their system! May we welcome you there too??!!

* Finally we have our award. This time we granted the award to the sender of the article on the adaptations to the Graphics Package by Howars Arrington,

mr. B. Geraads
Oeverbergstraat 51
5995 AJ Kessel.

Thank you for your neath explanation and congratulations!

****

A Z I N E S F R O M O T H E R M A

* **Databus** nr. 3: This number entirely has been dedicated to voice recogniction and digital speech generation. To admirers a 'must'.
Databus nr. 4: An article on the attempts of Epson to achieve standardization of control codes for printers... Besides that and informational on networks.

* **PCM** nr. 3 : A (not deep digging) article on tax deduction of your hobby. Information on Viewdata systems. As a bonus a flexi disk (some kind of ancient demo record like thing) holding two Basicode pro-

grams.
PCM nr. 4 : This number holds the item of CAD/CAM, spread out in several articles. They also have been testing Wordstar 2000. Additional info is given in 'news' and 'book reviews'.

I N F O P R O D U C T I N F O P R

* Someone is said to have been saying whatever comes from far has to be good! If this applies to the next is to your judging.

THe ION WEATHER company from Morristown N.J., U.S.A., has sent a price list to us of hardware goodies for the Sorcerer. They offer 48K Sorcerers (or look-alikes) unused, diskdrives with power supply, controller and cabling, bare S100 boards, diskcontroller boards (bare), manuals and so on. Those being interested bear in mind the charges for transportation and Customs dues, they may apply to the editor for the address.

* The Computer Contact Foundation likes to give notice by this means on their work. The Foundation intends to function as a registration intermediary on non-commercial basis for those willing to swap, buy or sell a.o. second-hand computers.
To be able to achieve that intention one is invited to have one's equipment or program for sale or wanted, registered FREE. One may phone or write to the Foundation.
Registration by means of the computer allows an easy search for the right partner, pointed to by the Foundation. For this contact they only charge Dfl. 10,= for administration expenses.
The address of the Foundation is P.O. Box 410, 2501 CK 's GRAVENHAGE.
The Foundation can be reached by telephone 070-242732.

****

I N P U T I N P U T I N P U

* *Mr. G.M. Ottenhoff from Neeg lets us know:*
I have recently purchased a printer (Brother HR5) and so I started looking over the ESGG periodicals for useful printer routines. The routine in nr. 8, 'Printing or LPrinting' gave trouble in combination with MBASIC. Lines are cut off at 64 characters, wich is extremely unfriendly in checking listings.
By poking the original memory contents into the printer routine this problem seems to be solved:

```
Printer on : POKE 16618,147:POKE 16619,233
Printer off: POKE 16618,171:POKE 16619,184
```

In MBASIC INPUT$(X) is a neath way to choose from a menue or a selection without having to hit (CR). To protect this way of INPUTting I developed next subroutine that:

- determines the X argument in the statement
- accepts Capitals as well as undercast
- rejects any other than the intended input and returns to INPUT.

This routine seems to the way around but finally saves space and gives maximum security.

```
100 REM Choice routine
110 K$=INPUT$(INSTR(HK$,"/")-1): REM determination of X
120 REM From undercast to capitals
130 FOR K=1 TO LEN(K$)
```

```
140 KA=ASC(MID$(K$,K,1))
150 IF KA<97 OR KA>122 THEN 170: REM only characters
160 MID$(K$,K)=CHR$(KA-32)
170 NEXT K
180 REM Evaluate
190 IF INST(KH$,K$)>0 THEN RETURN ELSE GOTO 110
200 REM Presentation
210 PRINT:PRINT"Your choice : "¡KH$="j/n":PRINT KH$
220 GOSUB 100
```

* *Mr. Bonninga from Groningen has been successful in tranferring programs to disk. As he writes:*

Your method for transferring programs from disk systems with a certain format to those with a different one, goes fluently from 77 to 30 track but not the other way around.
In one of the magazines there is a modification for updating an Exidy from A3 to C4 level. What is the gain?
The program Double Basic from volume 5 does not run. After having typed RUN "DB3INST" I only return to CP/M level with A>.

Mr. Bonninga, trouble seem to be around, ain't it? When the transfer method goes from 77 to 30 tracks, it should also work the other way around. The 77 track system writes well and the 30 track system reads well, so that part of the cassette interface seems to be in order. There is a possibility that the reading procedure of the 77 track system is not doing well. Try to record a program in 300 Baud with your 30 track system and read it with the other. If this is successful, then your 77 track system needs adjusting. This of course is an assumption as you did not point out what went wrong specifically (describing the error conditions and reports may be very helpful in determining the cause!).
It is very wise to follow the rule of "What is working all right, leave that unchanged!". That rule also applies to improvements meant to lift your Sorcerer from A3 to C4 level. If you intend to install the memory expansion of Gerard Evers, then your Sorcerer needs to be at C4 level.
The C4 level also is required in cases that timing problems cause malfunctioning of your diskdrives. With some types the problems have disappeared after having made the changes for C4.
To the problem of Double Basic I can be brief: Did you put the progam together with Exbasic or Mbasic on the same disk? An advise from the experienced, applicable to more users is: First read the 'DOC' files with the information on the program. This often prevents trouble!

* *Mr. Every from Plymouth (England) regrets that we can not mail diskettes to users abroad. He likes to try and use this means to turn the opinion in this matter in favor of the foreigners.*

If you can not send disks abroad for reasons of transportation risks, one could consider transfer to cassette for mailing purposes. Cassettes have proved to be reliable media in mailing transport systems.
The WP Pack has little attention in your magazine. I regret this too as this program is a very good one, especially when being used with disks. It also is helpful in creating MBasic programs.
Finally a question: Can you provide me with the diagram of the connections of the UART?

Mr. Every, that we have not many articles on the WP probably lie in the fact that one obviously does not pay much attention to this program. There are numerous adaptations to this program available on

cassette as well as on disk. If you feel to writing an article on the WP, we keep ourselves available for publishing.
The information on the UART is going to you by mail. Your idea of using cassetts as an intermediary for disk programs has been taken in consideration before. Mind that all those activities are the work of many a volunteer. Transferring takes a lot of (their limited) free time, already claimed so much by the club!

****

INTERVIEW.

by Welmoed Jonker and Don Siahaya.

Every now and then we like to pay a visit to 'prominents' from the users world of the Exidy Sorcerer. To keep it in line, we do not leave the country as the expenses to cover, can not be met. If one decides to star interviewing, one can not go around the Van Montfort Brothers.
They are Sorcerer Users from the start and also have given birth to many a change in soft- and hardware.
It therefore goes without saying that they are on top of the list.

One realises the distance to Heerlen, the home of the Van Montforts, driving for several hours before the skyline of the city appears from the haze. As a marker the building of the Civil Pension Fund has been mentioned, towering highly over the surrounding.
The house, in which the 'workshop' of Harry and Tonie is situated, lies in a quiet street behind the high CPF building.

We are lead to the top floor where the plans are made and ideas are put into practice tests. The room seems to be alike the place where many a hobbyist is spending his time, differing only in the number of computers that take up space. Although the Sorcerer is present majority we have been able to discover a number of other makes.

Having been provided with coffee, the conversation almost naturally turns to the past. It is for some years now that we have the improved 1.3 monitor version , also available with bootstrap loader for an easy startup of disk systems. Readers of the late magazine ESC issued by the European Sorcerer Club of England may recall the review by Andrew Marland. It is a balm to the national feeling that our Van Montfort monitor turned out to be one of the better improved version. Tonie knows that 500 copies of this monitor have found their way to the user.

In the past the bothers were three. Nowadays Harry and Tonie have to split jobs into two as the third brother withdrew from the 'company'. That this had consequences is illustrated by a lot of work not being done now. Especially the creation of the necessary programs to support the hardware developments got in the scrape. Tonie thinks aloud: "It would be a great help when readers having the knowledge, could provide us with these programs. A support like that would make more users get the most out of their equipment."

When going over the advertisement of the Brothers, the many supporting hardware is eye-catching: RAM pack (with or without battery back-up), Switch pack and disk-designed articles. From the information in the description speaks care and consideration for ease in use with the Sorcerer. It goes without saying that many a commercial program can be started with the intermediary help of the Van Montfort software.

Tonie arrives, upon the question whether all that has been thought of, or that has been developed also is going to market, with a piece of hardware being an IC and some flatcable attached. "I say", he explains, "this IC to be a perfect recording tube for the computer. With the right sofware of course, one has the means to put a picture into the memory." Whether this is marketed or not, we ask. His reply is one like "Oh, if there is a need, there is a possibility. We only do not have the time to develop the necessary software for it. That's why we do not sell it yet."

A thing that is ready for the market is the hard disk for the Sorcerer. Tonie turns around and gives an impressive demonstration of the speed with which the hard disk reads its information. The demo type has a capacity of 18 Mbyte. Larger also is possible. Replying to the "when" question Tony tells us that this hard disk and card can be ordered for from now on.
There is a large number of disk drives standing on beside another, so we ask about the number of drives actually connected. This number turns out to be larger than can be handled by CP/M 2.2. The bothers made a series of adaptations thus creating capability for over 3 drives. With CP/M+ (3.0) it is to be done too. This version is not available for the Sorcerer now.

We like to know whether there is still a market for them, the Sorcerer line being stopped (supply only to schools and so on)? Tonie feels no problems to occur in the near future. The Sorcerer may be a somewhat older machine but being capabla of coming on in the 8 bitter field. In several years from now, when Sorcerer users are going over to other makes, then they have to reconsider their target. That they have overthought the possibilities can be found in the presence of previously mentioned computers.

What, now that most thing needing improvement have been done, are the possibilites for 'face-lifting' the Sorcerer? For an example we point to larger memory e.g. 3x64K. Tonie fingers to a Sorcerer saying "That one there already has a memory of 192K. The only thing is that programs only can use up to 3x48, or 3x56K as the stack has to be moved into each bank." He explains why such an extension of memory is not being able to work rightaway. Especially when one is using disks, a memory of over 64K creates problems as CP/M does not know bankswitching in its present form. The monitor also is not ready for it. Actually installing a memory expansion is not a job too difficult. With the new 64K Ram memory chips it can be done in a jiffy! The actual hard job is the creation of a new monitor that also has to be compatible with the previous versions.

This requires a very thorough knowledge of the control systems of the Sorcerer and of CP/M to fullfill such a task. We do not think of the hours that are involved in this operation. According to the brothers bakswitching is possible with CP/M 3.0. They do not recall this version being available to the Sorcerer.

Increasing speed of operation from 2 to 3 MHz can be done. Whether this is a useful job? They consider higher speed useful when actual high frequencies (over 4 MHz) can be reached, handling thing slightly faster then. The lay-out of the main board prevents this. Thinking of higher speed is actually making a new design for the board. Is there a point in it, wonders Tonie.

It goes without saying that when visiting the 'connaisseurs' one drops his problems too. These fill the gaps making time goes by fast. It is getting night when we make arrangements to travel westward again. This is the moment Tonie shows some of the small 'gadgets' inserted into new software. One is a new software selectable character set, enabling 64 and 80 characters per line. It also is possible to add two extra lines to a screen page. Handy for those using a clock and liking time to be resident on screen, or for an automatically status line.

Finally we turn to the future (beyond the Sorcerer) by asking a buyer's advise when needing a new computer. The bothers consider this an awkward question, they are no salesmen.
Starting from the possibilities they point hesitatingly to the Tulip. Many software from the Exidy runs on that system. Being hardware and software developers that machine is less interesting for them as it is hard to get the necessary manuals. Without them there is little that can be done.

Humans not being computers but needing food, we say good bye and drive home. We agree on the fact that there are sufficient ways for the Sorcerer. Together we can be able to exploit them to the extend.

Those among you that have the ability to fill the gap in the software short of the Van Montfort brothers have an interesting foreland. Please write or phone to them if you like to help developing programs.

****

DID YOU BY NOW TRANSFER THE SUBSCRIB-TION FEE FOR THE PERIODICAL??

TRICKS with TRACKS.

Although Aad van Duijvenbode is no longer part of the editor's team we do have some 'left-overs' from that period. As a follow-up on previous remarks about the way of use of disksystems, he has an explanation about the division of disks. We like to point to the fact that especially newcomers should not start implemention his advise. You may run the risk of loosing valuable information or destroy entire disks. The information given by Aad can be checked with the help of programs like Disk Utility by Ward Christensen, as far as this is in the software field!

As more and more EXIDY-users obtain floppy-drives to store their programs, it might be usefull to give them a little background information about the way the tracks are spread over such a disk.
As one already supposes, there are different standards! Each disk is divided in imaginary concentric circles, the so-called tracks. The width of a track depends on the drive. The narrower they are, the higher the demands on quality of the head and the mechanical properties of the drive. This makes that drive more expensive, but the more tracks there are, the more information you can write on the disk.

So the number of tracks on a disk depends of that track-density 'TRACKS per INCH' (or TPI). The most commonly used values are 48 and 96 TPI. At the first value the distance between two tracks is 1/48 inch, about 0,5 mm. The coated space on a disk is about 30 mm wide. But there is an envelope around it, limiting the user-space at about 25 mm. This means that there is a maximum space for (25:.5) 50 of those tracks per side of the disk. But due to tolerances in dimensions, play, wear and tear, etc. the manufacturer limits the usable area to 40 tracks for a 48 TPI standard. This means that there are 40 tracks CERTIFIED (if the electronics are smart enough of course), but you can go a little bit further before the head of the drive meets a stop. Usually you can get 43 tracks on a 40-track drive.

The second standard is 96 TPI. So this means about 100 tracks per side. For the same reason as with the other standard, there are 'normally' 80 tracks. These are the so-called '80-trackers'. But as you'll say, I have a '77-tracker', what's that? Well, this is diverted from another standard. EXIDY namely used drives with a track-density of 100 TPI (why, oh why??).

So on the same space there are more tracks possible. But what does that same Exidy, it only uses 77 tracks, where certainly 80 of them are possible. In ESGG nr 12 you can read how to add tracks, and as each track can contain 5Kbyte, it is worth while.
The most annoying of the EXIDY-standard (100 TPI) is that the track to track distance (1/100 inch) is not the same as the commonly used standard of 96 TPI (1/96 inch). So disks of other systems can never be read completely, and other systems cannot read EXIDY-software (could this be the reason why ...??). For most EXIDY-users it is annoying, but they'll copy allright, you can depend on that; if there is no other way then via cassette!

But it is of course not necessary to use such a 100 TPI drive, you can as well use a 96 TPI drive with your EXIDY, the choise is larger, and they are cheaper too. Those drives are usually able to take 'double steps', making them act as 48 TPI drives. Thus you can also read disks made with 40 track- drives; and this standard is also used by EXIDY (thank you, thank you). Exchanging or selling of disks is impossible then, those disks are only usuable by yourself; but as told before, if you work with 40 tracks (with 48 TPI) you are compatible again.
You can also use different types at the same time with your controllerboard, eg. a 40-tracker (48 TPI) as A-drive and a 77 tracker (100 TPI) as B-drive. Easy when you want to covert from one format to the other.

Until now we only talked about the soft-sectored drives, and even with those EXIDY has a non-standard with the use of VISTA drives. But these are very rare over here, so we will not talk about them; if we did we had besides the 40, 48, 77, 80, 96 and 100 also 10 and 16 sectors, frightening us when liking to buy drives!
But no matter what kind of drive you use, it is the same as with the EXIDY SORCERER itself: once you have it, you can miss it no more!!

****

A CLOSE VIEW OF -TURBO PASCAL-.

This is the second article about Turbo Pascal by Fred Knottenbelt. In this article he will highlight variables, parameters, inline code, external subroutines and recursion. The Pascal is a CPIM-Z80 version.

Variables.

Each variable belongs to a memory location. The variable value will be stored in that memory location and if necessary in one or more consecutive memory locations. In the last case the variable points to the start address of the value.A variable of the type byte, char and boolean needs only 1 memory location, while an integer needs 2 and a real needs 6 bytes. We call these types 'simple'. The length of these type of variables is fixed. A variable of the type string or array has no fixed length. Therefore the length must be given during declaration of the variable. Also with this type the variable points to the first element of the stucture. We call it 'stuctured types'.
In Pascal all variables have to be declared before use, during declaration all variables will be assigned to a (start) address and the memory space they need will be reserved. We can check this very easily by compiling the program 'nop' and observing to the consumed data space:

```
program nop;
type t = byte
var a, b, c, d: t;
begin
end.
```

In spite of no variables have got any value yet, memory space has been reserved. For other types you can test for yourself how much room has been reserved. In Turbo Pascal however 4-bytes will always be reserved automatically for the main program and also 4-bytes for each procedure, that means compilation of a program without variables or procedures (begin and end. only) will assign 4-data bytes.
Due to an error in Turbo Pascal the displayed databytes address is one too high so if you want to investigate the memory map you have to subtract one. Besides, the number of data bytes is one more than is indicated. You can correct these errors by changing the contents of the memory locations 285BHex (2BH) and 2868H (13H) to 0.
So if you mention a variable in a program in fact you point to an address that belongs to that variable.With the assignment statement a:=3, you only change the addressed contents to 3.

Parameters in procedures.

In Pascal when you call a procedure (or function),you are able to pass parameters to it. If you want you can change certain parameters in the procedure and other parameters will keep their old values. This is depending on the heading of the procedure and can be done as follows:

```
procedure test(p: t; var q: t);
```

The variable t must be declared in the mainprogram itself. Assuming the type of t = byte. p and q are formal parameters. p is a value-parameter and q a var-parameter. Both are declared within the procedure by simply mentioning them in the procedure heading.
The procedure is called from the main program by mentioning the name and in the right sequence the parameters of the same type as p and q are,for instance:

```
test(a, b)
```

Both a and b are actual parameters; a has to be of the same type as p and b the same as q. By calling a procedure you may fill in a constant at the position of a value-parameter:

```
test(3, b)
```

During compilation of the procedure the value-parameter p will be assigned to an address in which the value of 'a' will be copied upon calling the procedure. By this it is possible to change the value within the procedure while the value of a is not changed, a is only copied into p. If 'a' and p are arrays, all array values will be copied into the new array p.This not only takes a lot of time but also takes extra memory space!
By calling the procedure, the local variable p in the procedure has become the value of a.All in the mainprogram you may use another variable p,this one is then unknown within the procedure only the local p will be used because during declaration both p's have different addresses.

During compilation the var-parameter q will be assigned to an address as well, when calling the procedure not the value of b but the address of b will be copied in q's address. Therefore q needs 2 bytes for storing the address, while p only needs 1 byte because a was copied into it and a is also 1 byte.If t was of the type array[1...100] of byte, than q needs still 2 bytes but p needs 100 bytes.

If within a procedure a value is assigned to q, the value will not be copied in q's address but in the address where q's address points to, in this case b's address. However if we need the value of q within the procedure (in consequence of the statement q := q+p;) the value will be collec-

ted via the address in q, that means b's value. This is the way of indirect addressing and will be recognized by assembler programmers. In general this way of addressing takes some more time than direct addressing, but no extra memory space will be used. Via a var-parameter the procedures outside world can be affected.

In relation to this it is sensible to declare arrays as var-parameters even if only data has to be read and no changes have to be made. Not only room will be saved but it takes also less time, because we do not have to copy data. We can see it with large arrays, especially if the procedure is called frequently, normally during each call all data would have been copied.
If b is an array, its address will be placed into q's address (and in the next one) and q's address contents is the first array element.When we need the tenth element of the array (if it was accorded type byte) 9 has to be added to the address that is placed into q, to access.

Inline code.

In the Turbo Pascal manual a small part is written about using inline code (including an example), inline code means that we may put processor instructions directly in a pascal program. But we have to compile the mnemonics to machine code ourselves! Unfortunately in the manual it is not explaned how data tranfer takes place from parameters into Z80 registers.

You can put machine code in Turbo Pascal as follows:

```
inline($21/$1234/                (LD       HL,1234H)
       $7E/                      (LD       A,(HL)   )
       $C6/$20/                  (ADD      A,20H    )
       $77)/                     (LD       (HL),A   )
```

All bytes have to be separated by a slash. If you use two bytes together (such as $1234), they will be placed in memory in reverse order as usual for Z80 addresses. In this example no Pascal parameters has been used. However we can do this as follows:

```
inline($21/a/                    (LD       HL,a  {put       a's       address
                                    {in HL)
       $7E/                      (LD       A,(HL){put     a's     value     in
                                    {the accumulator)
       $C6/$20/                  (ADD      A,20H )
       $77/                      (LD       (HL),A)
```

This small machine code program add 20Hex to the contents of parameter a. You are free to use all Z80 registers as long as the stackpointer does not change. A parameter will always be assigned to a 2 bytes address.You have to make sure that the right part will be placed in the registers if the type is other than byte viz.integer or real (for detailed information refer to the part dataformats in the manual).

We have to be careful if we want to use inline code within a procedure: changing the value of p we can do in the same way as above because p is a value-parameter. However changing the value of q (actually the value of a) means that we have to load HL with the contents of q's address, thus indirect! That means the first machine code statement will be LD HL,(q), the code for this is $2A/q/ as an inline statement.
Unfortunately, only this exceptional case of inline code is mentioned in Turbo Pascals manual as an example.

Those who are familiar with assembler language can check these inline codes by disassembling Pascal program parts. Because, after compilation it

is indicated where the codes and data areas are located. So if you write a simple Pascal program especially with inline codes, you can see after disassembling what has happened with the inline codes.
You can start with the program containing 'begin end.' only, (without digging too deep, because you will be sent from one subroutine to another). Next you can add a write statement or an inline code. You can also create an 'empty' procedure with only a value parameter which you can change afterwards in a var- parameter. It is very interesting to see the difference in calling a procedure with a value- and a var-parameter. If a call is made from the procedure itself the code will be different!
In this manner you can experience how Pascal statements will be carried out in Turbo Pascal.

External procedures and functions.

There is another way to communicate with machine language routines.You can declare a procedure or function 'external' with mentioning its absolute address. Be sure of the external procedure is available during running the program. It has to be placed in a free memory area, not directly after your program because this area will be used by the program and so it will be overwritten. Thus don't link it with your program.
You can, before compiling (to disk) reduce the highest RAM address and place the external procedure above or change the lowest address and put it into that free space. Keep in mind, Turbo Pascal uses the area below 100H. The very best area is in ROM.
Both value- and var-parameters can be sent to these routines. Transfering of these parameters takes place via the 'stack', as usual for all procedures. First of all get the return address from stack for example POP IY.
Then you have to collect the variables (for addresses and two byte values the sequence is the reverse).The format of all variables has been described in the manual. Finally, if all parameters have been collected, put the return address back onto the stack with PUSH IY. The external procedure can be terminated by the RET instruction (you may also use JP (IY) instead of PUSH IY and RET).

Recursive procedures and functions.

Supposing that you have a penny slot machine, you can indicate the cash-drawers contents in two ways from the moment it was emptied the last time. You can say, after the k-th throw-in the contents is k*valuecoin (only one type coin will be accepted) or you can say the contents is 1*valuecoin more than the previous throw-in (k-1). In the fist case the contents is immediately known in the second it is not. Nevertheless we can calculate the contents in the second case: then you have to know the contents the k-1st time and that was 1*valuecoin more than the k-2nd time etc. If we go back till the k-kth time then we have an empty cash-drawer. Now running the process in the right direction and adding every time 1 to valuecoin you are able to calculate the contents.
This cumbersome calculation we call recursion, the result is expressed in the preceding result. Mathematically noted:

y(k) = y(k-1) + x(k)

After the k-th throw-in the contents of the cash-drawer is y(k) and x(k) is the value of that throw-in. We call this a differential equation.

x(k) is constant in our example, but if there is more than one coin value in the penny slot machine, x(k) has at any time a different value. The contents is then not k*valuecoin, but x(k) + x(k-1) + x(k-2) + ...+ x(1) (that means the sum of all throw-ins).
Based on the above expression we can see the next is true, if k has been changed by k-1:

y(k-1) = y(k-2) + x(k-1)

Substitution of this expression in the first one yields:

y(k) = y(k-2) + x(k) + x(k-1)

The next step is then:

y(k) = y(k-3) + x(k-3) + x(k-1) + x(k-2).

At last we will get:

y(k) = y(0) + x(k) + x(k-1) + x(k-2) + ... + x(1)

We know we started with an empty cash-drawer so y(0) = 0. That means that the differential equation has been solved now.

The differential equation is more complex, in a lot of other cases solving the problem can not be done directly. In those cases we program the expression as it is written and we have to complete the progression back to the starting point. But that means if we write a function or a procedure to calculate y(k), it must be able to call itself, because in the right part we have the unknown expression y(k-1). However if the function or procedure can calculate y(k) then it can also calculate y(k-1), we only have to lower the parameter by 1 for each next call.
At a certain time we have to terminate these continuous calls. In our example it will happen if the parameter k is equal to zero. Because at that moment we know the value of y (viz. 0).

Imagine, calling the procedure from the mainprogram we have to return directly after the procedure call. This is controlled by putting the return address onto the stack so that we know were we are coming from. In addition to this all parameters will be transported by the same stack and within the function or procedure the parameter values then will be collected from the stack and written to the internal parameter addresses.
If a function or procedure calls itself, the same process will happen and the stack will longer because of all the return addresses.

It interesting to know that the procedure can not calculate any value until the first or starting value is known. The sum of y(k-1) and x(k) can not be calculated if y(k-1) is unknown (in our example we where also not able to do it). However at that moment the procedure will call itself to calculate the value of y(k-1).
Again it can not produce the value of y(k-2) etc. If at last the value of y(0) has been found the first return takes place and in reverse order we calculate the sum of y(1), x(1) and y(0), after that the sum of y(2), x(2) and y(1) etc., until finally the value y(k) can been calculated with x(k) and y(k-1).

There is one additional problem: If the procedure has been called from itself, as mentioned before the parameters we need will be transported via the stack. But entering the procedure all parameters will be put into their addresses, which have been assigned during compilation! That means all previous values will be overwritten and obviously this may not happen. Therefore in Turbo Pascal on entering the procedure all existing parameters will be saved on a so called 'recursion stack'. Also this stack will grow, even faster, because on the first stack only return addresses were saved. Returning from the procedure all previous values will be collected from the recursion stack. The recursion process can only run this way.
With the CP/M-Z80 Turbo Pascal version, we have to indicate if we need to have the recursive capability. By default, recursion is switched off and the recursion stack will not be used, that means present parameter values

are not saved on entering a procedure or function. This is not necessary if we do not use recursion and obviously the program runs faster. We can switch on recursion with the swich: {$A-}.

Now we will give a procedure which can calculate the contents of the cash-drawer in a recursive way. This is an example only because it is more efficient for this case to calculate it directly!

```
{$A-}
program sum;

var k           :byte;
    cash-drawer:integer;

procedure contents(p: byte; var q: integer);
{p = number of the throw-in, q = contents of the cash-drawer}
begin
   if p >= 1 then
   begin
      contents(p-1,q); writeln('q is now ',q);
      q := q + 1 {value of coin is pound}
   end
   else q :=0
end;

{Main program}
begin
   write('number of throw-in: '); readln(k);
   contents(k, cash-drawer);
   writeln('The contents has a value of ',cash-drawer, 'pound')
end.
```

Note: printed value of q! You can see the reverse order very clearly. The first print action takes place if we return from the procedure and that only happens when q = 0! You can determine the influence of q on the main-program's variable cash-drawer directly by changing the write statement in the procedure into writeln(cash-drawer).

****

MAIDENHEAD LOCATOR SYSTEM.

by Welmoed Jonker (PE1CHO).

Radio-amateurs are users of, as the law says 'radio-electrical transmission devices'. Do not confuse with the users of the 27 Mc! Although both are using the ether, the difference is in the way they do, but also their purpose of doing it. The use of the 27 Mc is free to everyone, but the radio-amateurs have to pass a proficiency test before they are allowed to use any transmitting equipment. In the Netherlands there are four possible stages, A, B, C and D-amateurs. New-comers usually start with the D-licence.

As a licenced radio-amateur you are allowed to do experiments with your apparatus, and to make contact with fellow-amateurs all over the world. You have to make a list of the the contacts you made, and you are allowed to use special equipment like a Telex (RTTY=Radio TeleType).
As the computer became a commonly used thing, it is logical that there were numerous experiments for using it to send data via the ether.
The computer is also often used to keep the log-book (of the contacts you made) up to date. A 'locator-system' is used to locate those contacts. The locator consists out of a combination of five characters (e.g. CM55A

stands for a part of Amsterdam). The existing locator-system has a shortcoming, in that it is repeating itself several times. This means that the same QTH-locator exists a few times in the world. That was no problem 10 years ago, but as technics are developing rapidly nowadays, making the transmitting devices used by the amateurs become better and bridging-over longer distances, it became a handicap.

In order to overcome this abuse, an English amateur developed an entirely new system which does not repeat itself. The QTH-locator of this new system is called 'Maidenhead locator' and consists out of a code which is a combination of six characters. When an amateur contacts via satelite or moon-bounce (reflection via the moon-surface) a fellow-amateur somewhere in the world, he is able to locate him via this code, with a very little tolerance (2.5 min.). The Maidenhead locater for the amateur in Amsterdam becomes JO22KI.

There are programs available to calculate someones QTH-locator from his map-coordinates, and the other way around. This does not yet exist for the Maidenhead locator, but I am sure the numerous ham-amateurs with a computer will develop such a program for the new system too.

As there are many ham-amateurs in our Exidy-usergroup, we would like to receive the utilities, programs of a disk- or tape- oriented log-book and whatsoever things you have developed regarding this hobby. It gives you an opportunity to tell about your solutions or give others a means to help you with special difficulities.
I, myself, and I assume many a radio amateur the world over are eagerly waiting for your contribution.

****

CP/M MODIFICATIONS (5).

Dany Rosseel has some remarks to the remark by Fred Knottenbelt about the supposed slowness of our southern neighbours:

Although I tried very hard to type GO 0 within 15 seconds after hitting double reset, I could not get my Sorcerer to reproduce the warm boot error mentioned in CP/M modifications nr. 2.
Nevertheless I owned a program that each time at the end when a warm start (JP 0) was called, got stuck totally! I then tried out the change mentioned in ESGG 16, hoping this would be the solution.
No difference upon having made the changes: the program hung!

After having experimented it turned out that the program booted well when I changed LD A,0D0H into LD A,0D8H (immediate interrupt). In this regard the suggestion is, in a changed form, of importance to me.

Would you please, before publishing, give this information to messrs Knottenbelt and Ringenaldus.
And not forgetting: A feather on the hat of Fred for the splendid lay-out and complete contents of the CBIOS listing I received from him. This document is a must to those going deeper into CP/M.

The editor regrets not having received a reply from Fred Knottenbelt on this adaptation. Perhaps there is going to be one following this publication, with the confirmation of this text. Of course it also is possible that one of the readers is sending in his reply after having made some checks. We will be awaiting.

****

RECEIVED ANNOUNCEMENT.

On the occasion of the bi-annual exhibition taking place at Saturday 9 and Sunday 10 November 1985 in hall

Centrum Zonnedauw
170 Kapelsesteenweg
Kalmthout Belgium

there will be held at Saturday 9 November 1985 from 10 a.m. an

O T H E L L O game

also known as REVERSI that will be a contest between computers as well as man and computers.

Programmers are invited to demonstrate the value of their program, written according to the regulations of the game of Othello.

Valuable prices will be granted. The outcome of the contest will be published.

Entries are to made by transferring BFr. 180,= or DFl. 10,= per entered program or 'brain' to account number 068-2010846-63 in the name of Elektronika Club Heide, mentioning 'Othello contest'.

Last entry date is 15 October 1985. Post Office stamp is proof of entry date. The following systems will be available to the contestants: Apple II DAI, Exidy Sorcerer, TRS-80 models 1 and 3, BBC, Spectrum, Commodore 64, TI and Zenith 100.
All other makes can be entered in the game, but are not on hand there.

The complete regulations for this game are available upon request, considering an addressed envelope and sufficient reply coupons are sent in. For further information apply to the secretary of ECH.

E.C.H. vzw. ELEKTRONIKA CLUB HEIDE (of which an important part has the computer for hobby) dedicates this exposition to "Applications of electronics in the automobile".

As been stated all systems can be entered provided they are self-supporting and using 220 Volts AC (50 Hz). No other power supplies on hand. In the entry both machine and man can be found. Of course the time in which the game may take place is limited. The maximum time is preset to 1 hour, this is half an hour per player. Time may be changed due to the number of entered players.

The contest filters the best player and the best program. Of course in the end there will be only one general winner!
Those who feel to have a powerful program or who are playing the game without mechanical help, please keep this day free.
All others have time to train or write the winning program. Mind the rules of the game!

The game is played on an 8 by 8 field board. This being the same as with chess. The numbering of the fields is identical. A field is detemined by a character (A to H) and a figure (1 to 8) in this way that the field bottom left is named A1 and the upper right field is H8. The pieces have one white side and one black side.

Upon start there are 4 pieces on the table as shown hereafter:

```
  . a b c d e f g h .
8 . . . . . . . . . 8
7 . . . . . . . . . 7
6 . . . . . . . . . 6
5 . . . W Z . . . . 5
4 . . . Z W . . . . 4
3 . . . . . . . . . 3
2 . . . . . . . . . 2
1 . . . . . . . . . 1
  . a b c d e f g h .
```

W=White top of piece up; Z= black top of piece up.

In turns the contestants place a piece with their colour up on the board, however this can not be done random. There need to be at least one piece of the opponent been turned. Pieces turned are opposing pieces between one's own piece and the piece place on the board. There is not to be an empty field or one or more opposing pieces between this and the piece on the board. Hits may be done in all directions, vertically, horizontally, as well as diagonally.
There may be situations in which a player can not put a piece on the board in this situation he looses his turn and the other player may take over.

The game ends with a fully filled board or when neither player has been able to put a piece on the board. Winner will be the player who has the most of pieces on the board. In contrary to the game of chess the game is opened by the player given 'black'.
We await your entry.

****

TIP FROM ABROAD.

Hans de Jong, a subscriber from Alexandrahills (Australia) made a pleasant discovery. He was transferring software from another computer, working with an Epson FX-80 printer, to his Exidy, with which he uses an Epson MX82/FT. In that specific program there was a command PRINT CHR$(27);"M"; causing the FX80 to set to Elite character printing (12 char/inch).
Although he could not find this command in the Epson manual, the printer did understand this command and carried it out. It is good to know!

****

LAST MINUTE INFO LAST MINUTE INFO

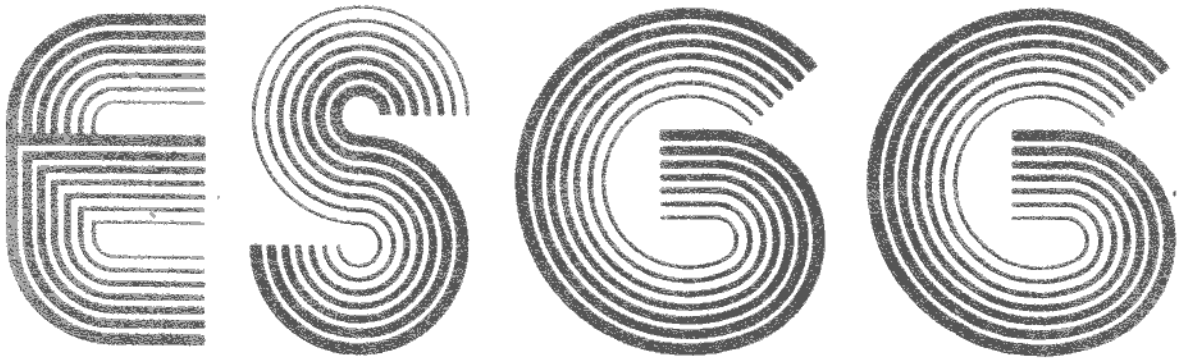
1 9 8 6 S O R C E R E R D A Y S !

S A T U R D A Y 1 5 M A R C H

S A T U R D A Y 6 S E P T E M B E R

RESERVING THE HALL AT GOUDA TO BE AVAILABLE!

****



The L O G I C A L partner to a Sorcerer

For whom is the ESGG?
For anyone being interested in the use and the possibilities of the Exidy Sorcerer.

Why the ESGG?
Because the ESGG trys to give as much knowledge to the possibilities of the Exidy Sorcerer and especially to the possibilities in the use of the Exidy Sorcerer, outside as well as inside the Hobby Computer Club of the Netherlands.

What does the ESGG?
Software distribution: We only supply software that is free of COPYRIGHT (so-called Public Domain Software) on collect cassette and on diskette.

For Exidy Standard Basic we supply a Basic EXTension in EPROM, that expands the possibilities of the Basic Pack very much.

Hardware development: Non commercial designs, that is designed by members and put at the disposal of the other members, are judged by ESGG and -if of importance to others- produced (e.g. video inverter).

Sorcerer days: Twice a year (usually in March and September) ESGG organizes these meeting places of many a Sorcerer user. These days meanwhile have become well-known.

Publications: Our bi-monthly issued ESGG-periodical, full of things worth knowing about the Sorcerer and alike matters. For only DFl. 27,50 (in Europe) or DFl. 32,50 for non European countries your can insure yourself of the most recent information concerning your Sorcerer! (see page 2). In the running subscription year we supply all issued numbers!

Subscribe?... Just transfer the fee for the subscription by bank to account number 337107793 or by postal money order to the treasurer of ESGG, Herman de Manpark 41, 3411 ZN LOPIK, the Netherlands, referring to "subscription to ESGG periodical".

You like to know more about the ESGG? Apply to the sectretary, mr. Charles Netteler, Pr. Hendrikstraat 3d, 3071 LG Rotterdam.