

bi-monthly periodical of the Exidy Sorcerer Gebruikers Groep

a translation in English of the original Dutch version



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

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

(more information: page 2)

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

ESGG-information and service	page 2
Info	3
From other magazines	4
Input	4, 21
Technical modifications	5
Internal memory expansion	5
Exidy Robotica (2)	6
Chiptip or gadget	11
Transforming WP files	12
Contents ESGG-disk volumes	13
Sorcerer and ZX Spectrum	15
Software exchange on cassette	16
CP/M modifications (3)	22

EDITORIAL STAFF

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

SUBSCRIPTION.

You become a subscriber when remitting the fee due to postal check account number 5368539 of ESGG at Lopik, The Netherlands, referring 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:
Administratie ESGG periodiek
Prins Hendrikstraat 3d
3071 LG ROTTERDAM Netherlands

ADVERTISEMENTS.

macro's: exclusively for business:
acquisition: M. Sanders
Richard Wagnerlaan 25
2253 CB VOORSCHOTEN.

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.

Copying of articles, diagrams or parts of them from this issue is only permitted to members or subscribers for non-commercial purposes, on the condition that one refers to ESGG, and the number of the issue.

Copying by third parties (non-subscribers) is only permitted after having acquired written permission from the ESGG editor. The editor supposes sent-in articles to be from the sender unless explicitly mentioned otherwise.

SOFTWARE COLLECTOR.

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

Cassette library: diskette library:
Wim Warning Hermine Bakker
Vogelweide 83 Falklanddreef 18
3815 HE AMERSFOORT 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 means of postal check to account number 5368539 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:

Table with 2 columns: name article (prices a piece!) and Sorcererday-collect by mail. Lists items 1-8 including Collect-cassettes, Collect-disks, ESGG diskettes, Eprom Basic EXTension, Manual Bext, Invers video print, Manual Bext, Lightpen ESGG, and List cassette software.

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.

INFO.

- * If you have looked over page 2 attentively, you may have noticed the blank line below the item editor has been filled in. To assist the editors team Theo Huijgen applied. Theo is going through all of the routines as a training period. When finishing this he is going to take care of a specific field that has to be described later. We really hope Theo is going to feel at home with us.
- * I do not have to tell you right here what we have on the menu this time. Whatever is in the folder, that is what you perhaps like to know! An article by mr. Quaedvlieg on adaptations to Tiny Pascal, articles on the ZETU assembler by Stan Podger from Canada, a production by Antoine van der Ven on ZCPR2, a switch-box to drive more than one printer (by one of the editors), whatever could be done with the tracks on your diskette, by Aad van Duijvenbode and much more! Of course this does not mean that you do not have to send your articles, it merely means that your entry can be edited by us in a normal way and also translation can be arranged at ease!
- * Although it is hard to believe! The number of members is still increasing, however not in the way it has been before! Even the number of subscribers is still going up. One hardly believes that there are even now Sorcerer users that have been muddling for years without trying to find out whether there could be a thing like a users group. Do you also know these users? Point out the benefits of a free membership (considering them being a HCC member) or the cheap subscription.
- * Making appointments and fullfilling them is very difficult! We did promise to enter an overall list of the cassette library in one of the cassette to be released. Upon the completion of this listing we did encounter a major problem: The file of the library grew so large that it almost took all of the cassette, leaving hardly any space for other programs. With this data you can roughly estimate the number of K's this file would take in your memory. In short, you probably would not succeed in loading both program and file in one piece (a little prayer for over 64K of memory seems useful!) and have a chance to run it too! So: It is going to be a hard-copy listing now. The price? This has to be calculated first, but we will let you know more soon.
- * It seems that the Sorcerer users are going over to the use of disk drives little by little. In what way we have noticed this? The increasing sales of diskettes and the decreasing sales of cassettes! However we are not going to end the sale of cassette-based software! In transportation (through mail) as well as in reliability this means of storage is still good for many years of service. It is said that this can not be matched by the diskette!
- * The ESGG clock is no longer in stock and will not be produced any more. This is a.o. because the clock had a rather cumbersome procedure when turning off one's equipment. Besides one of our members also sells a clock, for which we simply clear the field.
- * The period in what you can send in your ESGG INQUIRY 1984 form is almost at an end! Did you yet send in your contribution to the maintenance and information supply towards your fellow members and readers? Not yet????!! Do not postpone any longer now, but get your form from number 16 and start filling-in immediately! We merely help each other with this information!
- * Our modest little award for the great and illustrious 'brain twisters' this time has been granted to our

Rob de Beer
Slotplein 43
2902 HR Capelle a/d IJssel,

for the special way he has given a new dimension to the possibilities of the print-statement.

If you have brewed something special too, well, send it to the editor and maybe this little award will be your share!

FROM OTHER MAGAZINES.

- * PCM nr. 9: An overall view of standard and use of protocols in communications. A review of the language of LISP. The start of a series on 'better handling Basic'.
- PCM nr. 10: Another software-guide, reviews of Symphony and Framework (integrated software), Computers in school.
- * Databus 10: A new database tested: Microrim R:base 4000, Are you really better off with single sided diskettes used double sided for read and write? Some thinking on pipeline registers for signal handling.
- * Micro/Info nr. 7: A number for two months with CME software databank; the 'better' IBM (AT). Sense and non-sense of programming. Search and sort: the way a database works. News from the Efficiency Fair.

INPUT.

- * From Antoine van der Ven from Groningen we received two reflections on previous publications in the ESGG-periodical:
In regard with the remark by the editor at page 18 of ESGG number 16, the following:

The power supply print of the Sorcerer also is used as heatsink. This is causing trouble when having used the computer for several years. At first the electrolytic capacitors are getting hot, meaning that in the long run they are bound to loose their capacity and disrupt the proper operation of the computer. This happens quicker when the capacitors are placed directly on the board instead of being installed with space. Secondly, the 'meat-pins' are warmed-up and expand, and shrink when the computer is turned off, expand while operating the computer and so on. In the long run this is where contact problems arise from.

Both problems I do know by experience: I had to replace an elco when black bars blurred the screen and thereafter I had problems with the power supply connector of the transformer: one of the springs needed for keeping contact lost its elasticity and thus causing bad contact. This resulted in at first a 'flickering' screen and then a totally 'dead' computer. Soldering the wire directly to the print I considered too drastically. I therefore have soldered an old model golden wirewrap pin in the print and installed next a matching plug (Hint: these pins can be retrieved from dump computers; they are -not only literally but figuratively speaking too- worth gold: ideal for solving all kinds of difficult problems in which a soldering iron plays a part. Did you not yet encounter these problems? So:

1. You are no computer freak, or:
2. Your computer is too new, or:
3. You will have these problems very soon.

Regarding the 'CP/M modifications (2)' of ESGG 16 I have the next problem: At page 14 an error in the console status table has been observed and the suggestion is to swap some data. The way it is put down could be explained in more than one way, leaving me with the question: What has to be swapped with what? I have understood that the author of the article does not like to be annoyed by boring ESGG members (I understand his wish) but I consider my question of general importance. Therefore: Can the editor disturb his peace for getting this problem solved? I sincerely hope so!

Antoine, your hint concerning the power supply will be an appreciated one for the do-it-yourselfers among the ESGG-ers. Any way it is always worth it to take the trouble and pass this kind of information on to your fellow members. Regarding the last remarks: These will be forwarded to Fred Knottenbelt. His reply will be published as soon as possible in the periodical. You will be notified..

TECHNICAL MODIFICATIONS.

By Aad van Duijvenbode.

The following technical modification we are to discuss is probably only interesting to a small group of Exidy users, namely those that use a S100 expansion unit and having a MICROPOLIS disk controller board in it. From those users we address the group that are going to upgrade to 48K of memory.

The problem with this controller-card is, that the boot address is not working with a 48K machine. To solve this, the following changes have to be made:

1. At the S100 card:
 - a. make a connection between pin 21 of the S100 bus (the edge connector for the card) to IC 1A-pin 11.
 - b. connect pin 9 of IC 1A to pin 46 of J3, the connector of the flatcable towards the EXIDY.
2. At the controller card:
 - a. remove all connections (jumpers) at spot D4.
 - b. connect IC D8, pin 4 to IC D8, pin 8.
 - c. connect IC D8, pin 9 to pin 21 of connector J1, the connection towards the drive.
 - d. At the upper side of the board (the components-side), the track between pin 2 and pin 4 of IC D8 is to be cut.
 - e. The BOOT-address now is BC00..

GOOD LUCK.

INTERNAL MEMORY EXPANSION.

As you know the Sorcerer has, because of the use of a Rompack, a wide range of possibilities. One of them is the expansion of the available memory with 8K of RAM. As there are a number of -a.o. financial- objections to this, Gerard Evers has been looking for an internal alternative. He describes his solution hereafter briefly.

The maximum free usable memory space is 48K with the Sorcerer and therefore ends at BFFF. C000 to DFFF (8K) has been reserved for ROM PAC use (Basic, wordprocessor and so on), even when these programs are not used. This part of memory is useless so. Still this 8K can be used and is even a necessity when using the Pascal programs released. In what way one can add the area from C000 to DFFF to the free memory space?

There are two ways: replace the ROMPAC by a RAMPAC. Easy but expensive. So I chose a different manner: use dynamic RAM. A rather obviously idea, as the other part of memory also has been filled with DRAM. There is, however no reserved room on the board to do so. Now there are two possibilities: Take off all memory ICs and replace them by one row of 64K ICs. Rather expensive. With the old-fashioned 16K ICs it can be done cheaper: "piggy-back", that is solder on top of the present IC and in such a manner that the matching pins are mating well. The address and data lines are joint. Two control lines are not involved: RAS' (pin 4) and WE' (pin 3). The are directed separately. RAS' is used to select these ICs and WE' to establish write protect. This can be necessary to protect programs loaded in this area against overwriting. An example of it is ROMBASIC loaded from disk.

A disadvantage of this method is the lot of work needed: cutting tracks, making new connections, soldering ICs piggy-back, installing switches and therefore drilling holes in the case. An on top of it: not making mistakes! Especially the soldering of the ICs is for those, not used to do so, a difficult task but is possible to be done.

The advantage is a cheap memory expansion, the ICs are only Dfl. 7,60 through HCC and one only needs 8 of them. Besides this memory is reliable; it does not differ from the other part of memory. The addressing of the extra ICs does not create problems as has been found while testing. The temperature in the bottom IC increases a little however hardly noticeable. The ROMPAC also can stay in its place: with a switch one chooses between ROMPAC and 2 times 8K of RAM, without loosing the contents of RAM, provided you were not writing or running this program when switching.

To perform this modification successfully it is a must your Sorcerer being at C4-level. All of the newer Sorcerer have that update. In what way one can perform the modification to C4 has been described in ESGG number 10, 11 and 13.

The cost for the parts needed is about Dfl. 85,00 including the 8 memory ICs and the two switches. The manual with the diagrams is rather extended and therefore no good subject for publication. If there is enough interest in the English language area, we could consider translation. If you are interested, send a post card to the editor, with the remark "memory expansion G. Evers". We will keep you informed.

EXIDY ROBOTICA (2).

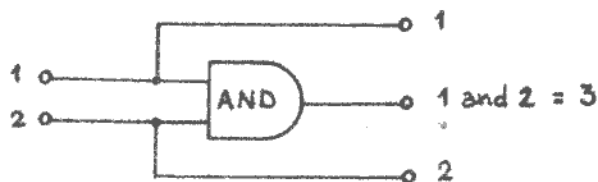
With the second part of the article of Wim van Grieken we conclude these small series. We express our gratitude to Wim for taking the trouble to provide us with the written part of his lecture on 'robotica'.

The use of the decoder/demultiplexer.

As we have been able to ascertain there are eight exits (8 OUT-pins), but in order to drive the six motors in two directions, we need 12 signals! In what way we can derive 12 signals out of the available 8? This can be done by using logical devices. For those among you, not familiar with this term, I will try to explain.

In electronics there are two kinds of techniques, analogues and digital. Analogous technics can be found in temperature meters, revolution-counters etc. This kind of technic is characterized by an analogous relation between the item measured and its result.

Digital technic has no analogous relation as the result adapts itself in steps to the item measured. With analogous technic one always is able to reproduce the result in steps. With digital technics this is always represented + or - one digit. Input as well as output only know two states: On or off. As this is a requirement for clear, logical handling or decisions, this technic also is being called logic technic. This explains the name 'logic elements'.



(diagram 5)

In diagram 5 there are derived by means of the logic element AND three signals out of two. We do not have to create this piece of hardware ourselves. Commerce provides us with an IC, e.g. the HEF4514 (Philips); this is called a decoder/multiplexer.

Out of four input signals this chip creates 16 output-signals ($2^4=16$).

The operation theory (read as: connection) shall not be explained here as the information that comes with the chip is sufficient.

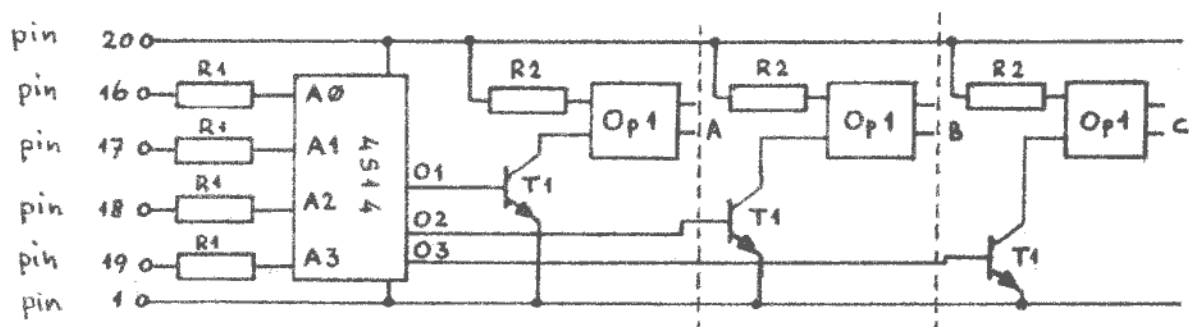
For an example, one line out of the truth-table:

A0	A1	A2	A3	00	01	02	----	05	-----	014	015
H	L	H	L	L	L	L	----	H	-----	L	L

So: a tension of 5 Volt to the inputs A0 and A2 and 0 Volt at the inputs A1 and A3 results in a 5 Volt signal only at exit 05.

The allowed current is not very high, only about 1 mA.

Our diagram now becomes thus:



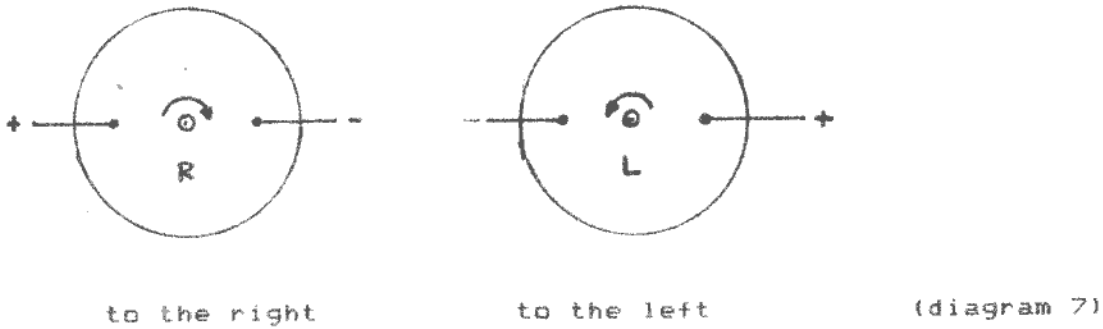
(diagram 6)

Remark: For the parts, ref to diagram 1.

At the exits of the opto-couplers, named A, B, C etc., we now can link connections like e.g. the LED's of diagram 1.

Motor wiring diagram.

In the robot arm DC motors are being used. The direction of the revolution depends on the polarity of the DC power supply.

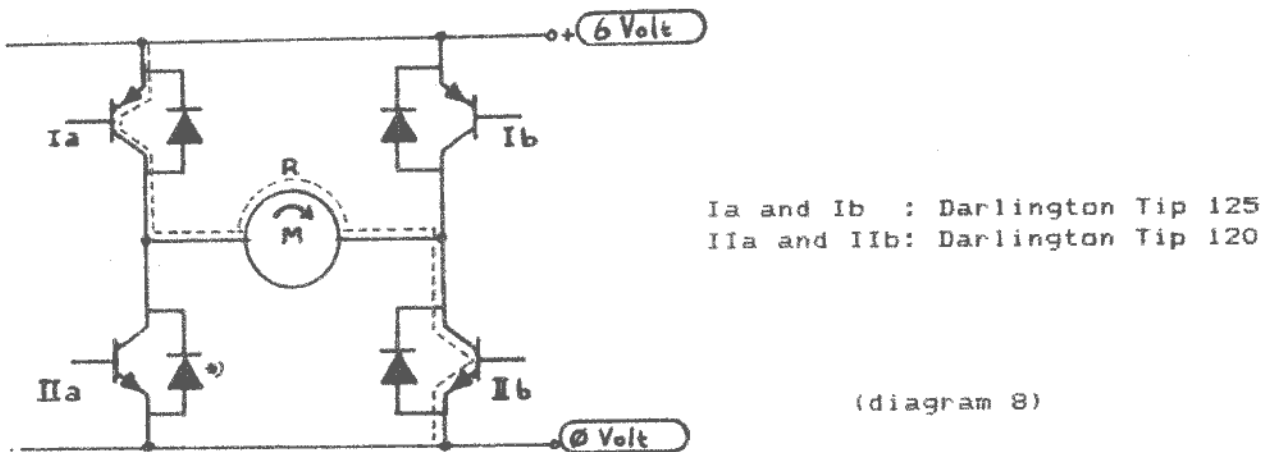


This change of polarity can be obtained by;

- a. relais
- b. transistors

For this project we have chosen for using of (power) transistors. If a motor is blocked it consumes a current of over 1 Ampere. In normal use the consumption of current is approx. 250 mA.

We now use the next diagram



Remark: The 'extinguisher'-diodes^{*)} are inserted in the Darlington's.

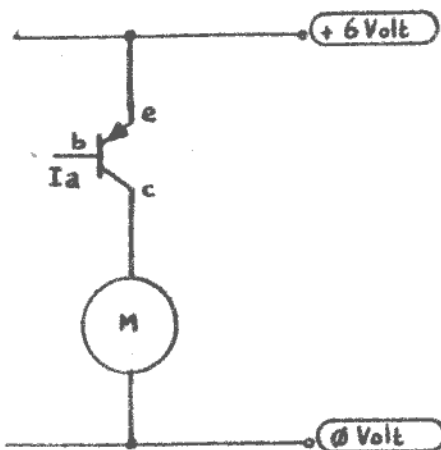
We have the following situation now:

Ia	Ib	IIa	I Ib	dir. of revol.
conducting	non-conduct.	non-conduct.	conducting	e.g. righthand
non-conduct.	conducting	conducting	non-conduct.	lefthand

In what way we make the transistors conducting or not?
 (Remark: this of course has been inserted in diagram 1)

De PNP-transistor.

Remark: To remember the type of transistor, read the PNP as 'point to plate'.



The PNP-transistor conducts when the voltage at the base (=b) becomes less than 5 Volt. The base current I_b gives a larger emitter/collector current I_c (I goes from e to c).

Then $I_c = B * I_b$ is valid.

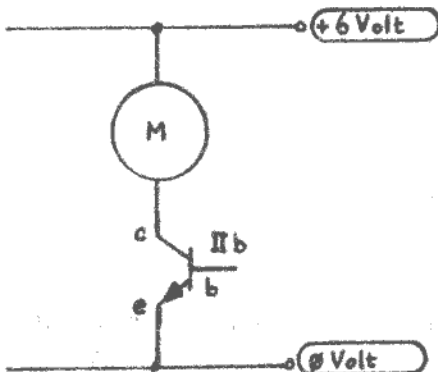
B is the amplifying factor of the transistor.

At 6 volt at the base the transistor is not conducting. So $V_b = 0$ Volt results in a running motor and at $V_b = 6$ Volt it is not.

(diagram 9)

The NPN-transistor.

Having mentioned the previous transistor being a 'point to plate' type, this type is being called 'Non point to plate'.



This transistor conducts when the voltage at the base exceeds 1 Volt. E.g. 6 Volt.

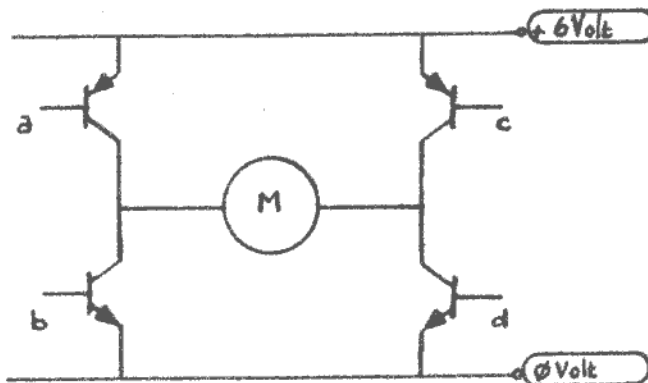
$V_b = 6$ Volt: motor runs.

$V_b = 0$ volt: motor stops.

(diagram 10)

Now by applying the correct base tensions to the transistors of diagram 8, we are able to achieve the required effect.

The diagram is repeated hereafter for your convenience regarding the review of the movements.



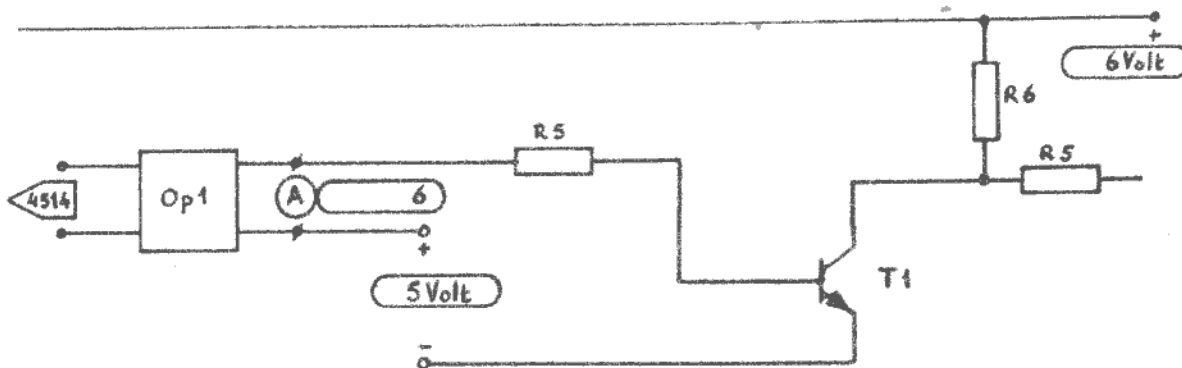
(diagram 11)

The movement-summary:

a	b	c	d	motor turns	
Vb	Vb	Vb	Vb	yes	no
0	0	0	0		X
0	0	6	6	X	
6	6	0	0	X	
6	6	6	6		X

<-- { opposite
 <-- { directions.

How admitting these voltages to a up to d?
 This is done in the following way:



(diagram 12)

- Parts: T1 = BC548 R5 = 560 Ω
 Op1 = T11 112 R6 = 10 KΩ

The theory of operation:

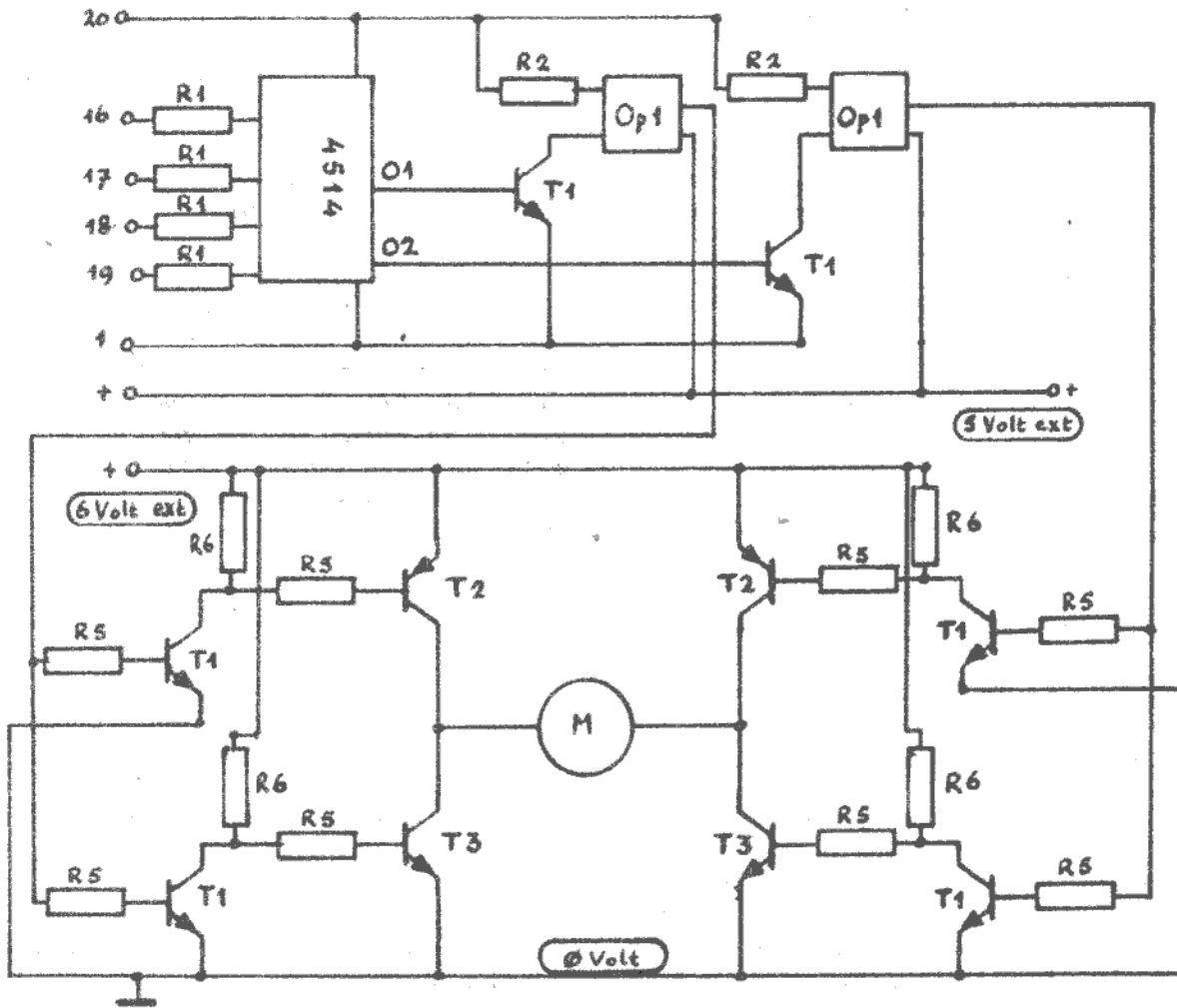
When a 5 Volt signal comes from the 4514 of diagram 6, the opto-coupler becomes conducting, and a 5 Volt tension is applied to the base of transistor T1. This one now also becomes conducting and pulls the 6 Volt tension being at point D to ground level. VD now is 0 Volt. Of course the opposite, no signal from the 4514, let the 6 Volt at point D unchanged.

In other words, we can turn point D into a 0 or 6 Volt level as we please! And this is it. This point D is e.g. point a in diagram 11. From the previous you can derive that for each motor-control we need four times the wiring diagram from diagram 12.

In diagram 13 there is the complete diagram for one motor. For the six motors you need to produce six of these. When doing it with only a single 4514 chip, you only can run one motor. The simultaneously use of motors is only possible with equivalent use of ICs of the 4514 type. So with the use of two chips 4514 you can drive only two motors and so on.

- Parts: C1 = 4514 chip R1 = 3.3 KΩ
 T1 = BC548 R2 = 150 Ω
 T2 = Tip 125 R5 = 560 Ω
 T3 = Tip 120 R6 = 10 KΩ
 Op1 = T11 112 M = 6 Volt motor

Remark: At T2 and T3 the collector-side to the motor!



(diagram 13)

Now about the most important item: the money matter. I have calculated the cost being approx. Dfl. 30.00 for each motor control. It now is obvious that the number of motor controls that you decide to be using, determines the amount of money to be spent on this interface.

I wish you good luck with your adaptations.

CHIPTIP OR GADGET.

Doing nothing and relax is not Hermine Bakker's habit. Instead of taking a break she leaves the household for whatever it is and crawls into the interior of her Sorcerer. I wonder what she found this time!

There I was again, in front of the computer, with an EXBASIC program that I had to type in once again to make it work with STANDARD BASIC. Then I recollected the ESGG-2 Volume with the CP/M-Basicode set: could it be of use for me?

To start with, list the EXBASIC program, divide the lines which are too long; edit anew a CALL into a USR-call; check if there is a ELSE; etc... Now ASCII-save the program:

```
SAVE "PROGX",A
```

Put the cassette in the recorder and let BCWRITEX do its work:

```
SYSTEM
BCWRITEX PROGX.BAS
```

My equipment is running smoothly. Standard Pack is still in its slot. Yes, all I needed now was at ESGG-disk 2; well then

```
EASYBC2
```

Wham!! at once a double-reset; I now could do without the basic-routines. Next to is the classic way

```
POKE 260,0:POKE 261,254:A=USR(A)
```

and look there: the program in Standard Basic! I suppose you can follow me correctly having rewound the cassette and then reading common Basicode.

Now a simple-save of:

```
PRINT PEEK(440)
```

This gave for a result a number of pages. Start CP/M and then save these pages with PROGX.COM. All together this had been quite less work than having to type-in the entire listing. And let us not start discussing on the conversion programs for EXBASIC; it is said they are far from ideal and are mainly created for converting users from cassette to floppy-drive and exactly in that same direction. And I had to go the other way. In such a case one leads the way oneself. The reward comes when being successful. Whenever you may face this problem: remember this solution. One never can tell!

M A Y W E R E M I N D Y O U
T O S E N D I N Y O U R
E S G G - I N Q U I R Y 1 9 8 4 F O R

TRANSFORMING WP FILES.

Mr. Stan Podger from Scarborough (Canada) has sent to us an article about transforming regular wordprocessor files into files for proportional printing. His views and findings are noted below.

One of the facilities of the Sorcerer that I enjoy the most is the Precision Print Driver for the Exidy wordprocessor. When I want to impress someone, like the boss, there is nothing equal to the proportional-spaced, justified copy from a daisy-wheel printer that this option can produce.

If I know that the end copy will be in proportional spacing, there is no problem. However, there have been several times when I started to write something in fixed spacing and later decided that it should have been in proportional spacing. Since the regular files start at 080FH and the proportional-spaced files start at 120FH, it was necessary to move the file up. That can be tricky if your regular WP file goes beyond 120FH. If you try to use the monitor MOVE command to do the job, you can pollute your file. This is because the MOVE command uses the Z80 LDIR command, and it moves bytes starting with the lowest byte. That is, you could be moving your first byte at 080FH onto copy at 120FH that has not yet been moved.

The solution to the problem is to use the LDDR command which moves bytes starting with the highest address in the file. With that command, you are always moving bytes into unused or no longer used space. The only difficulty then is to make sure that you have enough memory to accommodate the new file.

Having faced this problem a few times, I wrote the following short program to do the job. All it does is set the registers HL and DE to the top of the old and new files, set BC to the number of bytes to move, and then use LDDR to move the file. For users of the ASP Stringy Floppy, the GO address should be set to 0001H because that device cannot GO to an address of 0000H. Everyone else can use 0000H or 0001H.

Those familiar with assembly language should note that the following program was written with Zetu, not the Development Pac. The only difference, in this case, is that the hexadecimal numbers are indicated by a # sign at the front instead of an H at the end.

For the novices, just ENTER the numbers in the second column into the addresses in the first column, and store the program. Then, with the Wordprocessor Pac and a fixed-spaced wordprocessor file in memory, LOG in the translating program and it will do the job. Since you would not have the Precision Print Driver in memory at this point, there is not much point in trying to store your file with the wordprocessor "w" command. It will look for a fixed-spaced file starting at 080FH. Instead, you should go to the monitor ("x" command) and DUMP the addresses between 0746H and 074BH and SAVE from the address in 0746H-0747H to the address in 074AH-074BH. The GO address (SE X=...) should be DFFAH. Then you would LOAD in the Precision Print Driver followed by your new file, and you should have a proportional-spaced file ready for editing or printing.

The Listing

```

0000          00000 ;This routine moves regular wordprocessor files
0000          00001 ;          up to become proportional spacing files.
0000          00002 ;
0000          00003 REG>PP ORG #0000
0000          00004 ;
0000          00005 ;The stringy floppy can't LOAD and GO to #0000
0000          00006 ;          so we put a space at the beginning.
0000          00007 ;
0000 00          00008          NOP          ;for stringy floppys
0001  AF          00009          XOR  A          ;clear carry
0002  2A 4A 07    00010          LD   HL, (#074A) ;get old file top
0005  E5          00011          PUSH HL        ;store it
0006  01 0F 08    00012          LD   BC, #080F    ;old file bottom
0009  ED 42        00013          SBC  HL, BC      ;get file length - 1
000B  E5          00014          PUSH HL        ;length - 1 to BC
000C  C1          00015          POP  BC
000D  21 0F 12    00016          LD   HL, #120F    ;prop. file bottom
0010  22 46 07    00017          LD   (#0746), HL ;store new pointer
0013  ED 4A        00018          ADC  HL, BC      ;prop. file top
0015  E5          00019          PUSH HL        ;          to DE
0016  D1          00020          POP  DE
0017  22 4A 07    00021          LD   (#074A), HL ;          and to pointer
001A  E1          00022          POP  HL        ;old file top to HL
001B  03          00023          INC  BC          ;fix file length
001C  ED B8        00024          LDDR         ;move file
001E  C9          00025          RET          ;finished
    
```

CONTENTS ESGG-DISK VOLUMES.

by your disk-jockey Hermine Bakker.

To fulfill many a request I will sum up of whatever is on the ESGG-volumes right here. At present there are 4 volumes. Merely showing the DIRectories is somewhat like showing a menu card with all kinds of difficult names for things that are supposed to be delicatessen. On the other hand the printing of DOC-files require at least one full number of our ESGG-periodical. So I will keep it to a resume.

VOLUME 1:

- On this disk you find Tiny Pascal by Dr. E. de Vries, somewhat adapted to 56 K disk systems. Abundantly provided with explanation and elucidation with a large number of demonstration programs.

VOLUME 2:

- This volume consists of a highly improved disk version of Tiny Pascal from disk 1 with a number of new demonstration programs.
- Also a program set Basicode with CP/M, this version is only to be used with MBASIC.
- The programset CASCAT through which you can CATalogue your cassette programs automatically to disk.
- The tested program ESGG database as it is used by board and editor.
- A splendid reconstruction of CBIOS of CP/M 1.42/3.
- Finally EASYBC2, a program to run Basicode-2 with Standard Basic, more user friendly than the Basicode handling programs being released up to now, and
- A program to test empty formatted disks.
- CPMIO with ready-to-use routines of the CP/M I/O functions and for conclusion
- DATAC, the improved version of a previous DATACOM program to make Sorcerers of different disk formats communicate.

VOLUME 3:

- In this volume we find the first implementation of ZCPR2 for the Exidy. In one line mentioned but a splendid piece of programming!
- Two different versions of DATACOM,
- Three programs using the Exidy real time clock
- Three disk utilities (EXDISORT -directory sort- EXMENU -the directory shown in a menu- and GO -to lead you to a choosen address-).

VOLUME 4:

- An improvement on ZCPR2 for 48K (see volume 3)
- A program to print labels for your diskettes
- An improvement on 1.42/1 systems with various memory size.
- DMPDSK40, a Dutch Tiny DU enabling you to look into and alter the disk organization.
- A number of programs to be used together with the real time clock, a.o. one that ticks and produces the Big Ben chime.
- The Basicode-2 routines (CP/M version) but now for use with EXBASIC! And an improvement on the one for MBASIC.
- Two LLIST programs: printer listing of ASCII saved BAS.-files and alike ASCII format files. Yes, with page header, page number, margin and end of page fold jump.
- A program to test empty formatted disks.
- An improvement on Computata 2.2 (scroll error correction).
- A quicker version of CP/M based Basicode-2 CONVersion program.
- A program able to discover differences between two supposed identically files.
- A COP (filecopy) for single disk systems that can not use PIP.
- For conclusion a remarkable number of programs handling traffic between Standard Basic and the wordprocessor v.v. and also two programs to convert .WPF files into correct .ASM files and for converting an Editor made ASCII file into a .WPF file.

Resumes run the risk of showing a somewhat pover image of the entire contents but I can assure that the goal has been to obtain a true image. Each disk, even bought on trust 'blind folded' can be understood through the

many extended .DOC files and easily used. Those that like to know the kind of machinelanguage hidden in it almost always are helped by the .ASM files for study or improvement.

A lot of programs is combined into a package.

If you feel my summing up is too compressed, do realize that I am bound to use a limited space in your magazine. Wherever needed you may address the board or me for necessary information and elucidation.

We continue to go on this way and try to fill a volume 5: Are you helping to have it filled?? Programs are always welcome.

SORCERER AND ZX SPECTRUM.

A hint from mr. Theo Bartlett from Silverton (RSA) for those using more than one make of computer.

A point of interest to Sorcerer users is that the user defined graphics on the ZX Spectrum and the Sorcerer are the same. Both computers use an 8x8 matrix for a character.

Here is a quick way to make user defined graphics on the Sorcerer, which I learned from the Spectrum. Perhaps you have published it before, but not in any ESGG which I have, so maybe other new members will like it.

1. Each line of 8 dots is a binary number.
2. Starting from the right hand dot and working to the left, each dot has a value related to the power of 2.

e.g. 2⁷ 2⁶ 2⁵ 2⁴ 2³ 2² 2¹ 2⁰

When these are calculated you will get the one number for each dot as follows:

128 64 32 16 8 4 2 1

In the book 'A guided tour of personal computing' an example of the Greek character Omega (Ω) is given. Here is how to make it.

Draw your own design using 1 for a dot ON and 0 for a dot OFF.

0 0 0 0 0 0 0 0	= 0+0+0+0+0+0+0+0	= 0
0 0 1 1 1 0 0 0	= 0+0+32+16+8+0+0+0	= 56
0 1 0 0 0 1 0 0	= 0+64+0+0+0+4+0+0	= 68
1 0 0 0 0 0 1 0	= 128+0+0+0+0+0+2+0	= 130
1 0 0 0 0 0 1 0	= 128+0+0+0+0+0+2+0	= 130
0 1 0 0 0 1 0 0	= 0+64+0+0+0+4+0+0	= 68
0 0 1 0 1 0 0 0	= 0+0+32+0+8+0+0+0	= 40
1 1 1 0 1 1 1 0	= 128+64+32+0+8+4+2+0	= 238

Which is: - DATA 0,56,68,130;130,68,40,238

Hereafter you find a small program that will help you in defining your own set of characters:

```

10 REM USER DEFINED GRAPHICS
15 ' THEO BARTLETT (RSA - 28/04/1984)
20 PRINT CHR$(12)
30 PRINT TAB(14); "PROGRAM FOR USER DEFINED GRAPHICS"
40 PRINT TAB(14); "=====
50 PRINT
60 :
70 FOR Y=1 TO 8: ' ===== FOR EVERY ROW =====
80 PRINT "ROW NUMBER ";Y
90 ROW1=0: N=8

```

(continued on next page)

```

100 :
110 FOR X=0 TO 7: ' ===== FOR EVERY DOT ON A ROW =====
120 N=N-1: N2=2^N
130 FOR Z=1 TO X
140 PRINT "- "; NEXT Z: ' ** GRAPHICS PROMPT FOR DOT REQUIRED
150 PRINT CHR$(177);" ";
160 FOR Z=X TO 6: PRINT "- "; NEXT Z
170 PRINT TAB(35); "If ON add "; SPC(14); N2
180 INPUT "Fpr the DOT position shown: Enter 1=ON or <CR> "; DOT
190 IF DOT=0 THEN 210
200 ROW1=ROW1+2^N
210 NEXT X
220 ROW(Y)=ROW1
230 PRINT: PRINT
240 PRINT "THE DATA FOR THAT ROW IS "; ROW1
250 PRINT
260 ' PRINT THE CHARACTER MADE SO FAR
270 A=0: FOR Z=-120 TO -113
280 A=A+1: POKE Z,ROW(A): NEXT Z
290 PRINT "Your character looks like this so far "; CHR$(241)
300 PRINT
310 INPUT "Do you want to do this row over (y/n) ";Y$
320 IF Y$="Y" OR Y$="y" THEN Y=Y-1: GOTO 70
330 PRINT: PRINT "THE DATA SO FAR IS ";
340 FOR X=1 TO 8: PRINT ROW(X);: NEXT X
350 PRINT: PRINT
360 NEXT Y

```

SOFTWARE EXCHANGE ON CASSETTE.

Again a contribution by Dany Rosseel from Westende (Belgium). This time a serial interface developed down south is the issue.

If software need to be exchanged between two computers of different make, there almost always are a number of problems:

- The disk format of both of the systems is not the same even when both users have the same operation system. This means that it is usually impossible to exchange software on disquette directly.
- Practically all systems have their own way to put information on cassette. This means also that exchange through 'built-in' cassette options of different systems are not possible.

Making the disk formats readable by other systems is almost impossible and creating a "standard" 5.25 inch format is also a risky business. Besides this puts the 8" users in the cold.

There is, however, a de facto cassette standard for exchange of ASCII files and that is the BASICODE format. As to yet it has been implemented on quite a lot of systems. Each Sunday NOS- Dutch National Broadcasting Systems, ed.- (Hilversum 2/4 -channel identification, ed.-) broadcasts one or more BASIC programs in that Basicode method. NOS only broadcasts Basic programs meeting the so-called Basicode-2 protocol. It is not relevant to enter this theme deeper (ref. HCC Newsletter number 61).

Of course it is possible to use that method of transmission for ASCII files other than Basic programs. I think for example of Pascal programs (did

not NOS already transmit such a program in the past?), Z80 sources, regular text, member data etc.. The possibilities are numerous. It is obvious one can not meet the requirements of the Basicode-2 protocol, when exchanging other files than Basic programs.

It is a pity that a lot existing Basicode reception programs try to offer text received, to the Basic interpreter with disastrous result especially when this text is not a Basic program! There is quite a lot of work to be done. The receiving programs that put text after reception on disk, are better of course.

Concerning the exchange of information, whatever source it may be, providing it is ASCII, this is a better chance for Basicode method. Someone from the CP/M users group has even suggested to rename the system into UNICODE.

Another question pops up now:

Could it be possible to exchange machinelanguage between systems that use the same operation system?

This would be a useful matter as different systems using the same operating system, in principle understand their ml-programs (when being written according to the conventions). This a.o. happens to be the matter for those computers that run under the CP/M disk-operating system. In case we like to transmit ml-programs we could do two things:

1. Convert the ml-program into ASCII (e.g. using UNLOAD for CP/M) and then transfer this file to cassette. Upon reading of this cassette the program has to be converted to ml (e.g. using LOAD for CP/M).
2. Use a slightly different method of broadcasting that also transmits the 8th bit, but being an equivalent of the normal Basicode (see further).

The first method has two major disadvantages:

- It requires a lot of work converting the file to be transmitted to ASCII first and then converting the received file to ml again. This all has to be performed manually, unless the transformation has been built-in in the transmitting and receiving routines.
- The ASCII file is twice in length as the binary file as each byte is being translated into two ASCII characters. This could cause serious problems for machines that could handle the received program normally regarding the memory size, but that can not load the ASCII file in memory in one piece. Besides this a transmission takes double time now.

The advantage of the use of ASCII is that the Basicode standard as it exists, can be used.

All of these matters (especially the memory problem and the doubling of the transmission time) made us choose for a slightly different approach using non-ASCII files. The method is this:

- The speed of 1200 Baud has been preserved.
- The 2400 and 1200 Hz 'beeps' and the 5 second lasting leader and trailer are also preserved.
- The data still consist of 8 bits, bit 0 being transmitted first.
- There still are 2 stopbits and no parity bit.
- The STX (start of text) remains 82H.
- After the STX-byte now are following two bytes holding the number of bytes to follow (not including the checksum). The first byte is the low-order byte of the number and the second the high-order byte.
- Thereafter follow all significant bytes also including the eighth bit.
- Finally there is the checksum that starts counting from the STX-byte

- (exclusive OR), just like in the regular BC standard.
- Of course there is no ETX-byte now.

ASCII files always can be transmitted with this method.

Furthermore it could be useful when some kind of "identification" has to be sent too. This information could be the name of the file and a small description. We therefore expand the specification a little and we are adding a HEADER *before* the STX byte.

This header starts with the SOH byte (81H) whereafter a (in principle) undefined number of ASCII characters may follow. The header automatically is concerned ended upon reception of the STX byte by the receiver. Whatever the receiver is deciding to do with the header is his business, ignore, put on screen or use as a filename for disk save operation.

Of course the header is not to contain the STX byte (this is not a regular ASCII value). Also one can note that the SOH byte as well as the header do not belong into the byte count for the checksum. The STX byte is the first byte influencing the checksum. This also means that a receiving program does not need capability to handle a header, thus leaving compatibility intact.

In our local CP/M users group we momentarily use a header in which the first eleven bytes hold the programname (8 bytes for the name, 3 bytes for the type). The receiving program puts the received file on disk using that name, unless another name is being specified while starting.

The remainder of the header is a description (0-60 characters) that could be inserted upon transmission of this file. That part is displayed by the receiving program.

Every group having a different operating system has to decide:

- On the way the header would look like;
- If after the 'number of' bytes have to be some special groups (I think of a.o. LOAD and GO address).

These groups ought and have to decide for themselves as the needs for information differ by systems and as exchange of ml programs between different operating systems is useless.

Remark that the header principle also is adaptable to the 'regular' Basicode standard, leaving all compatible. The reception routines are waiting for the STX byte and all before that is ignored.

This signifies that implementation of it is possible without loss of compatibility.

This is perhaps a gadget worth thinking over.

Conclusion: The Basicode method and perhaps the 'machine' method allow us to exchange information between different computers:

- without being connected physically (who is eager to drag his machinery around?);
- without need of simultaneous operation;
- without the need of equal cassette standard;
- without the need of equal disk format.

LET US USE THIS !!!

RS-232 - Cassette interface.

A lot of computers have a RS-232 interface that allows communication with other machines through serial port. The RS-232 is the standard for serial exchange.

If one likes to exchange information over this channel it is necessary the two machines are connected one way or another (if need be through modem

and telephonenumber) and that they respectively send or receive at the same time.

A very useful extra possibility exists when the two machines need not work at the same time and need not be connected. The information transmitted by one then of course needs storing on some medium, until the receiving machine is willing and able to read this information.

A suitable medium is the audio cassette.

The exchange of information will be a lot easier (exchange of cassettes at club evenings?) and one has the opportunity of cassette back-up for machines that do not have that option factory built-in.

The annexed information describes an interface between the RS-232 serial port of the computer and an audio cassette recorder (see editor's note).

The interface has the next options:

- Maximum speed is 1200 bps (Basiccode speed).
- A digital '1' renders 2400 Hz at cassette end.
- A digital '0' renders 1200 Hz (Basiccode tones).
- Full duplex, meaning simultaneous conversion of RS-232 signals to cassette signals and the other way around, is standard.
- RS-232 output levels are + and -12 Volt.
- There are no RS-232 control lines.
- Power supply comes from AC source.
- There is an external tuning facility to establish errorless reading of tapes with abnormal differing speed.

These options make this interface a flexible one because of the three standard signals or tensions:

1. At computer end the RS-232 interface.
2. At cassette end the Basiccode tones, allowing with this interface the conversion of Basiccode into RS-232 and vice versa. In this way the sending and receiving software can be simpler and your parallel connector is free again for your printer! Also your RS-232 connector is going to serve the way meant by the designer.
3. At the power supply end there is the AC tension, which could be called a standard gradually.

In the annexed documentation is the electronically diagram, the parts list and the parts ordering list (see editor's note).

The only thing needed adjusting is potentiometer R6. This meter has to be adjusted in such a way that, without RS-232 input signal, there is a frequency of 4800 Hz at pin 15 of IC3.

If you do not own a frequency counter you also can feed a known RS-232 signal and adjust R6 to produce as large a number of ONEs in a row to produce a correct number of periods in the cassette signal (a '1' gives TWO periods of 2400 Hz at 1200 bps=bits per second).

The adjustment for compensation of variation in the reproduction speed of cassette recorders is not critical (at correct speed), but it is advised to adjust to center position. Only when reading can not be obtained errorless, the potentiometer R28 can be turned from center position.

The connection to your equipment is easy:

- The RS-232 output from the interface is connected to the input of your computer (usually pin 3 of the serial connector).
- The RS-232 input of the interface is connected to the output of your computer (usually pin 2 of the serial port).
- One of the RS-232 ground signals at the interface is to be connected to

- the ground signal of your computer (usually pin 7 of the serial port).
- If your RS-232 system is a of the kind with 'control lines' (RS-232C), then you are to create the control line connections yourself. See the manual or as your fellow members.
 - The Cass IN signal of the interface is to be connected to pin 3 of the DIN connector or the IN/OUT CINCH plug of the cassette recorder (regular Line OUT). Use shielded wire! If your recorder has no combined AUX plug, use the equivalent EAR signal.
 - The Cass OUT signal of the interface has to be connected to pin 1 of the DIN connector or the IN/OUT CINCH plug of the cassette recorder. If your recorder has no combined AUX plug, use the equivalent of the MIC signal.
 - Both of the connections to the cassette recorder have their shield grounded at the interface end, whereas the shields at the recorder end are to be connected to DIN connector pin 2 or the shield of the CINCH plug.
 - If your recorder need a stronger or weaker input signal (recording too weak or too strong) you could exchange R9. A larger value gives a weaker cassette signal.

To built the interface one needs some experience with the next matters:

- Soldering of electronically components to a printboard.
- The right way to handle MOS-ICs.
- The identification of the components and the determination of the polarity of some parts.
- The way to handle a frequency counter or oscilloscope.

These matters however are no real part of this article. Fellow members or your users group undoubtedly are able to help you with a number of problems.

If your cassette recorder has the Dolby option, better not use it as this feature changes the wave form of the signal. You best find out for your situation what is working best.

Finally I have to mention that there is a print lay-out for this RS-232 cassette interface. If there should be sufficient interest for it, a number could be ordered for. I suggest you to contact your club. The board is invited to contact me for an eventually group order.

The lay-out for this print is also in the documentation. The estimated cost of the printboard, not drilled, single sided epoxy 10x16 cm, is about Dfl. 11,00 without mail.

Good luck and software exchange!!!

Postscript of the editor:

As the annexes take up so much space, we have decided publishing only the concerning intro and description of the hardware. If there is a need for such an interface, you are invited to mention this to your board. The P.O. Box of the editor is a proper entry to the board. May I suggest your proof of interest to be in this box within a fortnight from the release of this number of the periodical. The number of reactions can help to determine a point of view.

Of course you are free to write directly to the author, in order to get the description and material. The address of Dany Rosseel is: Lombardsijde 169 at B-8440 WESTENDE (Middelkerke), Belgium.

INPUT.

- * *Fred Knottenbelt has sent a letter to us, replying to the leader by the editor to the article about DEVSYS by Fedde Ringenaldus in ES66 number 15 (page 6). In his 'consideration' he states to be surprised by the remark from the editor that this article had been placed as an exception the article being commercially. He reacts to the idea that it is not right to write in the periodical about commercial products. In his opinion this point of view is clashing with the interest of the members.*

In his opinion the lack of attention to those matters could be acting as a brake to the unfolding of initiatives by companies due to the few information concerning their products.

He likes to see an opening through the periodical to poll for the need for such articles. He feels this kind of polling could be a support for the determination of the point of view concerning eventual production.

First of all the editor likes to clear up a misunderstanding. We certainly do not wish to keep articles about commercial products out of the periodical! The intro to the article of Fedde is in that regard obviously misunderstood.

In first instance we have created an opportunity for companies to have information concerning their products inserted in the periodical in a column called 'PRODUCT-INFO'. We regret this possibility has hardly been used. The intention of such a column is to offer the producer the space needed to add information to the product advertised. For such an information the advertisement is often too expensive a space. The information usually concerns new or improved products.

As to the polling for interest for products the producer could place a call asking parties interested to get in touch with him. Such a call could be offered as an advertisement.

- * *Fred Knottenbelt has some notes on remarks by the editor about the article by Henk Warnitz concerning the problems with MPI-drives.*

I like to answer to the editorial postscript to the article by Henk Warnitz, where the editor doubts the statement concerning the power supply connector not being resistant to the current through it. This statement however is correct. The current that is delivered by the transformer consists of very strong, very shortly lasting pulses (that is why the diodes are so strong!). These diodes namely only open at the time the transformer tension is higher than the tension at the capacitor. This is only so for a very short period and during that period the energy used by the computer has to be replenished. Especially the ground connector has a heavy life as this connection carries that high current twice a period, being only once for the tension carrying transformer connector pins (there has been made use of double sided rectification).

It is remarkable that there are two ground pins at the bottom (male) connector and that only one of them has been used; the upper connector has only one (female) ground contact. If the designer had been using the second contact too, the current could have been spread to two contacts thus decreasing the risk of trouble.

Whenever the contacts start to burn, the only solution is to solder the wires directly to the board (first remove the connector pins). One can see by the colour of the plastic near a contact that this connector had been 'sparking'. I had been looking for a soft cracking sound in my computer and only when the computer had been opened in darkness I saw lights in the connector. It seems a lot of Sorcerer have this problem and this therefore is the first thing service technicians look for!

By the way did you know that the power supply print can be turned 90 degrees (parts kept up!) and still can be connected to the Sorcerer print? That is very handy when having to check the underneath circuits.

You only have to watch out for resting the print to the metal studs at the bottom of the Sorcerer as you may cause a short-circuit between tracks.

The number of requests for the BIOS listing (ESGG 16) is up to now (end October) almost enough for starting printing. Remind that after December 1st 1984 no orders can be placed as then the exact number of this listing is being printed. Thank you for the nice remarks on the order cards.

* At this place we also like to correct a shortcoming in the previous number: The author of the article concerning the changes of the Baud rate of the serial port was Fred Knottenbelt. This had not been mentioned in the intro. We apologize.

* Mr. G. Wezer from Assendelft sent a letter stating having trouble with programs -no trouble up to now- in which a like POKE -16432,147 had been inserted. If so, the computer is totally tilted and even ESC/RESET does not help no more, or the computer returns an immediate READY and next only returns a READY to whatever command is given. A double-RESET is the only way out! He had found that this problem only occurs when CTRL-T (Token -BEXT8-) has been activated. He has tested the ROMs of the pack with the program CHKSM (Checksum) by Henk Warnitz. With the original ROMs the result had been all right. After having adapted some lines in the program for checking BEXT8, there were 144 bytes short! He wonders whatever could be wrong.

So do we, mr. Wezer! From the postscript I understood you having sent this problem to Henk Warnitz too, so we will wait for his comment.

CP/M MODIFICATIONS (3).

Or, as Fred Knottenbelt had titled his article "The continuing story of CP/M-BIOS (softsectored 1.9211.93)". In this part he has inserted and evaluated remarks and suggestions by Dany Rosseel from Westende (Belgium) and from Nora Schram from Amsterdam.

Fortunately there are still those that do not follow the doings of others in blind trust. From our Belgian friend Dany Rosseel I received a note that a calculation error had been made while determining the byte at address 2BDF in the relocation table. The value 20 has to be 10 (page 11, ESGG number 16, about halfway). Change this byte in the text of the article and in your systemtracks, for the cure might be worse than the disease! I have lost through this the contents of a disk, on which I had been preparing the text of this article!

Dany also hands a thoroughly thought over change to make the auto-wrap-around not take place after 63 but after 64 characters (Watch it, Dany: there has been changed something in it!).

To get this the next alterations need to be made:

```

2249: BB          CP   E          ;old and new col# equal?
224A: CC 05 E2   CALL Z,0E205H ;then CR&LF to outputdevice
224D: 79          LD    A,C        ;restore character
224E: C9          RET              ; and return
224F: FD 5E 6A   JMP: LD  E,(IY+6AH);save current column #
2252: E5 F5 E9   JP    0E9F5H    ;jump into video driver

225F: 18 EE

```

In the relocation table at 2C2A: fill with 00.

We, however, do not support this modification as the routine E205 gives a CR&LF via the outputvector of the monitor, while all other characters are sent directly to video! If you start the monitor and then change the outputvector, the characters and the CR&LF are sent to different devices.

We did try to send the characters to the outputvector as well, but we found that the Stack space reserved by CP/M is too small to be able to a back-space.

As the output can not be send through the outputvector, we do not agree with the modification by mr. Van der Ven from Groningen (ESGG number 13, page 13, bottom), suggesting to lead the input through the inputvector.

Dany Rosseel had been unable to reproduce the warm boot error (upon reset, followed by GO 0), but that has been caused by the known Belgian slowness: the error only occurs when one GOes 0 within approximately 15 seconds from reset (You have to speed up a little, brother!).

We are trying to accomplish a good (that is, a fully reliably) CP/M version. You have noticed that we are going over the modification that have been sent in, very critically. The alterations that have been published in previous issues all have been carefully examined and we only recommend in this column the generally usefull ideas! Of course you ought to know for yourself whatever you wish to do with your version. In this regard we consider the hint by Aad van Duijvenbode in ESGG number 13, one that never should have been published. Through this modification an essentially CP/M test, being a check on whether you have been changing disks without giving a CTRL-C, can be switched off. This could have a disastrous result as the directory might be blown away!

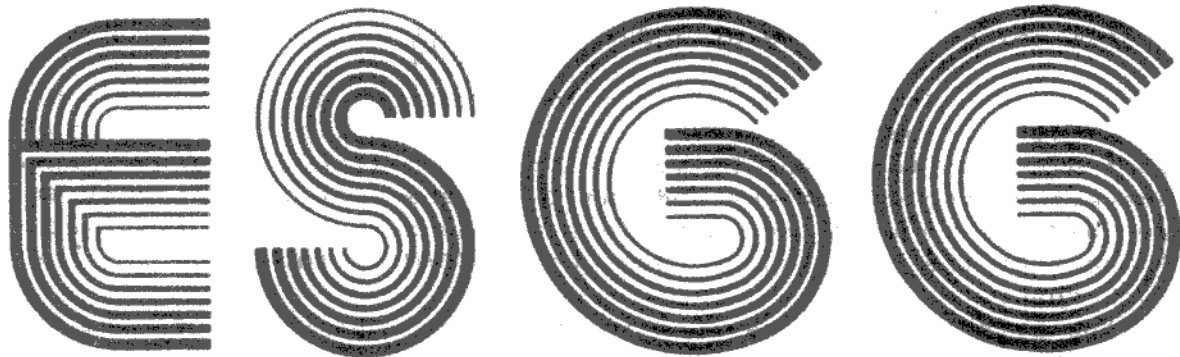
From Nora Schram we received note that she is creating, between her other computer work, an UNFORMAT program through which double sided written disks can be turned into single sided ones again. We already have accomplished this job (our version is going to the program library soon).

She handed a very useful hint to find out whether a disk had been formatted double sided (as you may recollect we were going to write a special program for this purpose!). The recipe is as follows: Put a double sided system disk in A: and the suspected disk in B:. Now command STAT B:DSK:. You now can see how many 128 byte records/track (erroneously stated as 'sectors/track) there are on the B: disk. This number can be 32 (for 256 bytes/physical sector and 16 physical sectors/track) or 40 (for 512 bytes/p.s. and 10 p.s./track). For a double sided formatted disk these numbers are doubled as now also the upper side of the disk is counted. In this way you immediately can see if a disk is double sided formatted, also when someone had put a single sided system on this disk!

It is obvious that this part of the article only is of importance to those that use one or more double sided drives and only now and then use a single sided system for communication through the disk to other members.

If you might have other alterations, please send them to the editor. Upon consultation with us (Fedde and I) they are to be published in this "continuing story", if they are of general interest and if they stand the test.

Question: Is there someone able to give information about the tutorial language LOGO being implemented on the Exidy Sorcerer? If you have this information, please inform the editor.



The LOGICAL 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 tries 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 you 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 remit the fee for the subscription either by money order or by postal check to postal check account number 5368539 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 secretary, mr. Charles Nettelers, Pr. Hendrikstraat 3d, 3071 LG Rotterdam.

