



MAY 1985

I HAD OCCASION TO LOAD THE CHARACTER SETS FROM THE GROM THE OTHER DAY . USUALLY THIS IS DONE BY LINKING TO GPLLNK IN ASSEMBLY LANGUAGE. I WANTED TO DO IT WITHOUT USING GPLLNK. THE IDEA WAS TO JUST MOVE THE PATTERNS IN GROM TO THE VDP PATTERN TABLE. THREE ADDRESS IN GROM CHIP 0 CONTAIN THE ADDRESSES OF THE START ADDRESSES OF THE PATTERNS. THESE LOCATIONS ARE 004C,004E,0050 . THE ADDRESS AT 004C IS THE START OF THE PATTERNS FOR THE TITLE SCREEN CHARACTERS. THE ADDRESS AT 004E IS THE START OF THE PATTERNS FOR REGULAR UPPER CASE CHARACTERS. THE ADDRESS AS 0050 IS THE START OF THE PATTERNS FOR LOWER CASE CHARACTERS.

WHEN I FIRST TRIED LOADING THESE PATTERNS ONLY THE TITLE CHARACTERS SEEMED TO WORK PROPERLY. I WAS USING THE SAME TECHNIQUE , MOVING 8 BYTES FROM GROM TO THE CORRECT LOCATION IN VDP. AS IT TURNS OUT THE REGULAR UPPER AND LOWER CASE CHARACTERS ARE STORED IN GROM AS 7 BYTES. THE PROGRAM THAT MOVES THE PATTERNS TO VDP HAS TO ADD THE EXTRA BYTE AT THE START OF THE PATTERN WHICH IS ALWAYS A 00 (TO LEAVE A SPACE BETWEEN EACH CHARACTER LINE). BYE DOING THIS I SAVED 192 BYTES OF MEMORY IN GROM. IN FACT THE GROM PROGRAM THAT LOADS THE PATTERNS INTO VDP WOULD USE UP SOME OF THAT SAVINGS.

I HAVE BEEN WORKING ON A NATIVE CODE COMPILER OF HIGH LEVEL FORTH WORDS. THIS SEEMS TO OFFER THE OPPORTUNITY OF SPEEDING UP SMALL AREAS OF CODE AS REQUIRED FOR GRAPHICS OR SORTING APPLICATIONS . THE SPEED UP IS TYPICALLY FROM 600 % TO 1000 % OR 6 TO 10 TIMES AS FAST. THE DOWN SIDE IS THAT THE SPACE NEEDED BY THE NATIVE CODE IS USUALLY FROM 2 TO 4 TIMES AS LARGE. THIS FIGURE COULD BE MUCH MORE IF YOU WERE TO COMPILE WILLY NILLY WITHOUT SOME INTELLIGENT DIRECTION.

INCLUDED IN THIS NEWSLETTER ARE 3 SCREENS OF FORTH. SCREEN #6 IS A MESSAGE SCREEN WHICH CONTAINS ADDITIONAL COMPILER MESSAGES. THIS MUST BE KEPT ON SCREEN # 6 TO WORK PROPERLY. THE OTHER SCREENS CAN BE LOADED BEFORE LOADING THE REST OF YOUR SYSTEM. THESE TWO SCREEN WILL PROVIDE SOME MEASURE OF COMPILER SECURITY FOR FETCH AND STORE. AN ERROR OF WARNING MESSAGE WILL BE PRODUCED AT COMPILE TIME IF FETCH (@) OR STORE (!) ARE USED IMPROPERLY. NO RUNTIME CHECKING IS DONE AND NO EXTRA CODE IS PRODUCED AT COMPILE TIME. THE ONLY OVERHEAD IS THE TWO SCREENS OF COMPILER EXTENSIONS. ONCE A NEW PROGRAM IS WORKING THE PROGRAM CAN BE RECOMPILED WITHOUT THE COMPILER EXTENSIONS IN THE DICTIONARY.

BOB CROSBY IS SELLING SOME GAMES AND CARTRIDGES INCLUDING SHAMUS AND ALPINE . CALL BOB AT 382-8384 IF YOU ARE INTERESTED.

```

BCR #26
0 ( DEVELOPMENT COMPILER MESSAGES 00 - 47 )
1 # ' to a constant. #
2 # : to a colon definition. #
3 # ' to a stack value.
4 # @ from a constant. #
5 # @ from a colon definition. #
6 # @ from a stack value.
7 Loading screen #
8 Bytes
9
10
11
12
13
14
15

```

```

BCR #29
0 ( DEVELOPMENT COMPILER EXTENSIONS ) ( :DEF CODE @ 33588 )
1 ( VARIABLE CODE @ 47650 CONSTANT CODE @ 47594 )
2 : DERR ( n-- ) CR ." ** ERROR ** SCR # " IN @ BLK @
3 B/SCR /MOD . ." Ln " + 64 / . CR MESSAGE :
4 : DWARN ( n-- ) CR ." * WARNING * SCR # " IN @ BLK @
5 B/SCR /MOD . ." Ln " + 64 / . CR MESSAGE :
6 : ?STATE ( -- F ) STATE @ 0= IF 0 ELSE 1 THEN :
7 : ! ?STATE IF HERE 2- @ @
8 CASE
9 47650 OF ( VAR ! ) ENDOF
10 47594 OF ( CON ! ) 33 DERR ENDOF
11 33588 OF ( :DEF ! ) 34 DERR ENDOF
12 ( DEFAULT ) 35 DWARN
13 ENDCASE COMPILE ! ELSE
14 ! ( NOT COMPILE TIME ) THEN ; IMMEDIATE --
15

```

```

BCR #30
0 ( DEVELOPMENT COMPILER EXTENSIONS ) ( :DEF CODE @ 33588 )
1 ( VARIABLE CODE AT 47650 CONSTANT CODE AT 47594 )
2 : @ ?STATE IF HERE 2- @ @
3 CASE
4 47650 OF ( VAR @ ) ENDOF
5 47594 OF ( CON @ ) 36 DERR ENDOF
6 33588 OF ( :DEF @ ) 37 DERR ENDOF
7 ( DEFAULT ) 38 DWARN
8 ENDCASE COMPILE @ ELSE
9 @ ( NOT COMPILE TIME )
10 THEN ; IMMEDIATE
11
12
13
14
15

```

JOHAN VAN IMSCHOT HAS PREPARED A PRODUCT EVALUATION OF THE CORCOMP DS/DC DISK CONTROLLER CARD. THERE ARE NOW TWO OF THESE CARDS OWNED BY CLUB MEMBERS.

THE CORCOMO 9940 DISK CONTROLLER

Review by: Johan Van Impehoet.

This controller was brought to the April meeting for demonstration purposes. Unfortunately it could not be made to function properly at that time. It was later found that this was not the fault of the controller: it was set for faster drives with which it had been working, but the TI drives used at the meeting were not capable of working well at that speed. The demonstration should go well at the May meeting.

For those who might miss the meeting, a short review follows.

The card appears to be totally compatible with TI files and file handling and formats. The exception is the double density formatter on the P-Code system. (It wasn't useable with the TI card either, of course) However, the P-Code system will use double-density diskettes formatted by the CorComo disk manager, and so this is not a major problem (although a new P-system dformat.code would be appreciated!). In fact, I can report that all four disks worth of system stuff will fit onto one disk with plenty of room to spare for program space:

The advantages this controller offers as compared with the TI controller are as follows:

- will handle up to 4 drives (instead of 2).
- can handle both single and double density (not just single)
- both this controller and the newer TI one handle double sided drives
- uses a disk-based disk manager as well as being compatible with the TI cartridge-based manager; the CorComo manager is somewhat fancier and easier to use; also it is searched for on all drives, not just DSK1.
- the card manager make it possible to load object code from disks without a cartridge in place as would normally be required
- CorComo includes a TI-FORTH version compatible with their card; it adapts TI-FORTH so the latter is no longer restricted to the E/A cartridge (as TI in its 'wisdom' originally designed it). Documentation for the Forth is of course not included.
- there are resident in the hardware a number of low-level access programs to seek,peek,move memory, and execute machine programs; these routines are accessible both from Basic and Extended Basic. (No cartridge is needed.)
- the hardware can be configured to take advantage of faster drives
- the manager can be configured to know ahead of time the drive configuration; this seems to be there primarily to prevent you from accidentally getting single and double-sided diskettes confused.

There seem to be few disadvantages. One is that this is definitely a card intended for use with extra memory. (It can be used without xmem if the TI disk manager is available.) Early versions of the controller and/or the manager were not compatible with some of the third party hardware, and the controller was initially not compatible with early console models.

Also, apparently early versions of the disk controller did not do a good job of initializing certain cartridges, at least if the rumour I have heard is true. The version I have does seem to handle the problem alright. From rumours in print, and personal experience, I would say that CorComo has experienced a number of teething problems such as this with the card, which are probably solved by now.

The major disadvantage of the card may relate to CorComo's viability as a company. Though TI has quit making its various 99/4A products, it still repairs/replaces them at reasonable rates. As CorComo may now be having financial difficulties, it's future may not be a safe bet, and future repairs of CorComo cards may thus not be a safe bet. On the other hand, I do not wish to discourage potential supporters of a company such as CorComo, which has supported TI99/4A users with a variety of hardware, especially the RS232 card, and their products generally seem to aim at excellence. TI users need companies such as CorComo, and they deserve our support.

Some of you may know that when I first received the CorComo disk controller, it was defective in that it did not operate well in double density. Although it took some time, I did get it replaced under warranty and it now functions well. Apparently there was a bad run of chips, and the design has been changed to help avoid the problem.

The disk manager is almost excellent. All normal file handling procedures are handled well and more conveniently than by the TI manager. Only a few minor improvements could be suggested. One advantage of the disk-based manager is that CorCom has been able to supply its customers with corrections and updates. I am not as impressed with the disk test portion of the manager: I am not certain that it always handles errors properly; it sometimes starts on an endless run of error finding without moving the head again (but I may be having drive problems...on the other hand the TI manager never does this.) Another problem of the manager seems to be that initial access to drives 2 and 3 does not work; one must try twice. (Yet from Basic the access works first time, thus I suspect the manager rather than the card or the drives, but I cannot be sure.)

The manager includes an object code loader, so you don't need a cartridge to run assembly programs. Although this is great in itself, I don't like the implementation: one must first load the manager, which is large, and then select the loader from it: I would have preferred a quick-loading loader to have been immediately selectable from the hardware menu. The manager can be called from Basic, but this offers no advantages, as any program resident is removed: again it would have been better if this had been a loader one could call from Basic!

Of course, one could load one's own loader via the disk manager, and then run it from Basic, but all this is much less convenient.

The possibility of fooling the card into loading one's own program instead of the CorCom disk manager obviously exists, but CorCom hasn't made it easy, as there is no documentation supporting such an approach. This would certainly have been appreciated.

The documentation is generally quite good, even going to the point of covering TI file handling conventions. Also there are sample programs on diskette demonstrating the use of the utility routines built into the hardware. My only beef was that it should have stated up front that the only way to copy the disk manager (as I always make a backup first thing) was TO USE THAT VERY DISK MANAGER and INITIALIZE a new disk: this took me a while to find out. Some sample programs are included on diskette to show use of the built-in utility routines.

So that's about it, for a quick synopsis. Is it worth buying one? It costs somewhat more than a TI one, but it does far more. Only the possibility of better TI servicing would make one suggest buying a TI one instead. If you already have a TI controller, you will of course be less likely to spend the \$250 Canadian or so required to obtain the extra advantages of the CorCom card.

Should any member wish more detailed information, or even a personal demonstration, please contact me. (John -- 479-7503)

EDMONTON J. G.
P.O. Box 11983
EDMONTON ALTA
T5T 3L1

