* XB/PACKER V1.2A *
* XB/PACKER V1.2A *
* XB/PACKER V1.2A *
* XB/PACKER V1.2A *

FOR THE AMS 128 OR 512
FOR THE AMS 128 OR 512
FOR THE AMS 128 OR 512
FOR THE AMS 128 OR 512

BY BRAD SNYDER
BY BRAD SNYDER
BY BRAD SNYDER
BY BRAD SNYDER

V1.2A notes:

The only change is to set up the registers on the SuperAMS correctly for the low memory area. This prevents a lockup when running XBPacker on a SuperAMS, -if- ABOOT has not been previously run.

V1.2 notes:

This is the third release of XB/Packer. While all known bugs were fixed, it is not just a simple maintenance release! This version of XB/Packer is now able to auto detect the size of RAM on your AMS or SuperAMS, and if you have 256K or more you will be able to load up to ten extended basic programs at one time. If you only have a 128K ams, you can still load up to

five programs at once. Also new in this version all of the program's functions are enabled to start with.

V1.1B notes:

Corrected a bug in detecting the AMS card.

V1.1 notes:

Many new features have been added, making XB/Packer much more powerful and easier to use. XB/Packer can now save or load all 5 extended basic programs for you with only one command, a menu system has been added that a program can call up, there are 'hot' keys that call up the menu and let you change banks with a single key press, etc!

Distribution:

This version of XB/Packer is being released as fairware, and may be freely distributed.

To register, just send $5.00 to:

Snyder's Software
c/o Brad Snyder
4260 Cedar Drive
Walnutport, PA 18088

If you want me to provide you with the program on disk, please send $6.50 to cover the registration fee plus shipping and media. I will then send you the registered version of the program.

Concept:

This utility, when used with the Asgard Memory System, allows you to load up to five (10 on a 256K card) extended basic programs into the computer at one time. The programs that are loaded can be run one at a time, and a running program in any bank can call and run another program in any other bank. For the purposes of this utility, I have set aside five or ten different areas of memory in the AMS, and am going to call each area a bank. Note that this is not the way the AMS actually works, but is the way that the utility emulates banks. Note that XB/Packer does not request memory from the system, but takes direct control of the AMS card for maximum use of the available memory. Therefore, any programs that you may have loaded into your AMS via other utilities before loading XB/Packer will be lost.

Requirements:

- o TI-99/4A computer
- o TI Extended Basic or compatable, including XB3!
- o AMS 128K or 512K memory expansion
- o At least one disk drive controller

Thanks to Winfried Winkler (the author of Asgard's XB3), this version of XB/Packer should be compatable with any version of extended basic. It has been tested with TI extended basic,

XB3, and RICHGKXB. If you are using a different version of XB, I would love to hear from you!

Loading:

To load the utility, just run the program called "XBP" on the Packer disk. The assembly language will be loaded and the SETUP link will be called. The LOAD program on the disk is also the same as the XBP file, except that it has been modified to load and run the demo programs. If you either let XB auto load the LOAD program, or manually run it, you will see an example of how five programs can be set up to automatically load into all five banks and then run the program in the first bank. The demo will play 5 different music selections. Each one is a seperate program, all resident at the same time in your AMS. To stop the demo, just press [fctn 4] at any point.

Files on the Packer disk:

PACKDOCS - The documentation that you're reading.

MUSIC:x - XB music programs for demo purposes

(note that there will be several files starting with MUSIC: on your disk. They are all part of the MUSIC file.)

LOAD - Auto loads the demo programs

XBP - The main packer program

Links added to XB:

NOTE: THE FOLLOWING COMMANDS WERE PART OF V1.0:

```
CALL LINK("SETUP")  -  This  link gets the AMS ready for my
CALL LINK("SETUP")
CALL LINK("SETUP")
CALL LINK("SETUP")
```

banking scheme. You must use this link before switching banks is allowed. Also, this should be called only from bank #1, and setup will make sure that bank #1 is set on return to XB. Bank #1 is the default mode of the AMS. Your own programs in their exit routines should set bank #1 before exiting.

New to SETUP in V1.1: Setup is now more fool-proof, and if you call it from one of the XB banks, it will be ignored. A check for an AMS card in your system is made, and if none is found an error is returned. The high memory for XB in each bank is now fully initialized. The interrupt routine has been upgraded to automatically check if you already have an interrupt routine running, and if so will run your interrupt before the XB/Packer interrupt is executed. There is now a 5 second delay after returning from a running program to the command line until the status at the top of the screen appears. This was required to prevent the status from randomly appearing when a running program would call another program.

After you call SETUP, a status line will appear on the top 4 lines of your screen, if no program is running. The status line identifies the utility that you have loaded (packer) and

tells you what bank is currently active, and the bank's name.

CALL LINK("INSTAL") - Copies the program in bank #1 to all
CALL LINK("INSTAL")
CALL LINK("INSTAL")
CALL LINK("INSTAL")

other banks. Used for auto loading. This command is retained for compatability with V1.0, but is not needed with the new ASAVE and ALOAD commands of V1.1. It is likely that this command will be removed from future versions of the program.

Note: the INSTAL command is still available in V1.2 but is no longer being supported. It will only copy the program in bank 1 to banks 2 through 5. Banks 6 through 10 are not supported by INSTAL.

CALL LINK("BANK",n) - Selects bank n. If called from the
CALL LINK("BANK",n)
CALL LINK("BANK",n)
CALL LINK("BANK",n)

command line, the specified bank is selected and the status line updated to reflect the current bank and bank name. Whatever program may have been loaded in the current bank previously will still be there for you. If BANK is called from a running program, the bank will be selected and then whatever program is in the new bank will be "run". This allows you to chain programs in all banks. Remember when writing your own programs that the BANK link is the same as RUN except that it also changes the current bank. Any statements you have after BANK will not be executed.

New in V1.1: additional error checking has been added.

NEW COMMANDS IN V1.1 and higher:

```
CALL LINK("CSET")  - Reloads the lower case characters from
CALL LINK("CSET")
CALL LINK("CSET")
CALL LINK("CSET")
```

the TI-99/4A system GROM. Useful for chaining programs when the calling program has changed the lower case character definitions. This command was added because I found that CALL CHARSET in XB only restored the upper case characters.

CALL LINK("PASS") - Should be used when exiting from your
CALL LINK("PASS")
CALL LINK("PASS")
CALL LINK("PASS")

programs. Can only be called from bank 1, and returns the AMS card to the default 'pass' mode. The XB/Packer interrupt routine will be shut off and whatever program you had in bank 1 will still be in memory.

CALL LINK("MENU") - Calls up a menu where you can choose to
CALL LINK("MENU")
CALL LINK("MENU")
CALL LINK("MENU")

run a program in any bank by pressing a single key, or just exit back to the command line. Your own programs can call this link and use it as your own menu to select which program to run. Calling this menu will not mess up your XB screen, colors, definitions, sprites, etc., EXCEPT for the screen color. So if you are calling this from a running program, make sure that the called program knows to reset the screen color to the value it wants.

CALL LINK("NAME","bank_name"[,b]) - Gives a bank a name.
CALL LINK("NAME","bank_name"[,b])
CALL LINK("NAME","bank_name"[,b])
CALL LINK("NAME","bank_name"[,b])

The name will show up on the menu and status line. The names will be saved with ASAVE and reloaded with ALOAD. If no bank number is specified, the current bank will be named. (I.E.

CALL LINK("NAME","BANK #1")) Optionally, you can specify a bank number and that bank will be named no matter which bank you are calling the NAME link from. (I.E. CALL LINK("NAME","BANK #1",5)) The name you are giving to a bank can be up to 20 characters long.

CALL LINK("ASAVE","DSKx.filename") - Saves the contents of
CALL LINK("ASAVE","DSKx.filename")
CALL LINK("ASAVE","DSKx.filename")
CALL LINK("ASAVE","DSKx.filename")

all XB/Packer banks, and their names. The file will be written in 8K chunks, with the last character of the filename incremented for each segment. Only the amount of memory in each bank that is actually used will be saved, but be aware that even empty banks will have their pointers saved. When you reload the file, those banks will be empty again, even if you had previously loaded programs in them. ASAVE can only be called from the command line, and will let you know if you try calling it from a running program. The reason for this is that so much VDP memory is used for the save, the XB environment will be destroyed.

CALL LINK("ALOAD","DSKx.filename"[,x]) - Loads the
CALL LINK("ALOAD","DSKx.filename"[,x])
CALL LINK("ALOAD","DSKx.filename"[,x])
CALL LINK("ALOAD","DSKx.filename"[,x])

XB/Packer banks from disk. The filename specified must be the first file in a series saved with ASAVE, or an error will be returned. ALOAD is a powerful command and has several options when calling it. If you use the optional parameter "x", the file will be loaded and then the menu called up, or the bank specified will auto run. x=0 will call up the menu after the load, x=1 through 5 (or 10 if you have 256K) then that bank will be run after the load. If the optional parameter x isn't used, what happens after the load depends on whether ALOAD was called from the command line or from a running program. (Yes! ALOAD

can be called from a running program!) If called from command line, you will be returned to the command line. If called from a running program, the program in the current bank will be ran

after the load.

To summarize: if the optional parameter "x" is used, it takes priority no matter if ALOAD is called from the command line or from a running program. If x=0 the menu will be called. If x=1-10 then the bank specified will be run. If no optional parameters are used, if you call ALOAD from the command line, you will return to the command line. If called from a running program, the new program in the current bank (the bank that you called ALOAD from) will be ran after the load.

NEW HOTKEYS IN V1.1 and higher:

CNTL-SHIFT - If you press control and shift at the same
CNTL-SHIFT
CNTL-SHIFT
CNTL-SHIFT

time, the menu will appear. You may then choose to run any of the displayed banks or return to the command line.

FCTN-SHIFT - Pressing function and shift at the same time
FCTN-SHIFT
FCTN-SHIFT
FCTN-SHIFT

will increment the current bank number by one.

The hot keys do not function from a running program but only from the command line.

The AMS:

The Asgard Memory System is available from the SW99ERS computer group.

Questions or comments:

Please write to me!

Snyder's Software
c/o Brad Snyder
4260 Cedar Drive
Walnutport, PA 18088

Delphi: BRADSNYDER

Internet: BRADSNYDER@Delphi.COM or bls3@Lehigh.EDU

BBS: The First Floor, 610-760-0527

Delphi Signup Offer:

To check out Delphi, call 1-800-695-4002 with your modem. Once you are online press ENTER once or twice until you get a prompt. At the password prompt type in GALERIA and press ENTER. You will be given the option to sign on to Delphi to see how you like it. Delphi will give you 5 free hours and the option to cancel at any time during the first 30 days! It won't cost you a thing.

Once you are on Delphi, enter GO COM TI at any prompt to go to the TI NET. Hope to see you there!