by Stephen Adams, London, England

ZX Memory Map

When Clive Sinclair designed his basic computer, it was to contain only three things within the memory map (the area storing all the instructions to operate the computer). The area is easily accessible by the Z80A microprocessor and it is easy to program uses for it. The memory area is divided into 65535 locations (64K) by the 16 binary address lines (A0-A15) that indicate the location at which the processor wants to look.

Sinclair required only a place to store users' programs (RAM), a place to hold instructions to operate the computer when it was first switched on (ROM) and a location in memory which would translate the TV picture. So he divided the memory map into three sections: 0-16K for the ROM, 16-32K for the program RAM and 32-64K for the TV picture. He used only the top two address lines A14 and A15 to tell which section was operating at the time.

This divided the memory map into the four sections in Figure 1 with the ROM and RAM repeating themselves again and again. Address line A14 is used to turn on the ROM when the condition is binary 0. When it is binary 1 the RAM is turned on. Address line A15 is used to operate the hardware that puts out the TV picture; it operates only when A15 is binary 1.

Because A14 can also change state from 0 to 1 while A15 is operating the TV hardware, the RAM can be used to store the screen display. The ROM software uses that fact and has only to pick up the address where the screen is— between 16K and 32K— and then add 32K to it to put it on the TV screen.

The only problem with this system is that ROM between 32K and 48K could not be used because it would upset the TV picture. It also meant that the ROM appeared in the 8K to 16K section, blocking out any other uses for that region because it could not be turned off.

That was the situation which existed in the ZX80 and only internal modifications to the computer circuitry could sort it out. The only place left to put new devices produced by other manufacturers was to steal some of the RAM space to fit in the new device. By operating a pin called $\overline{\text{RAM CS}}$ on the expansion board, the RAM could be turned off. Sinclair included this feature in their design because some method was needed to turn off the 1K of internal RAM when the 16K RAM pack was fitted on the back.

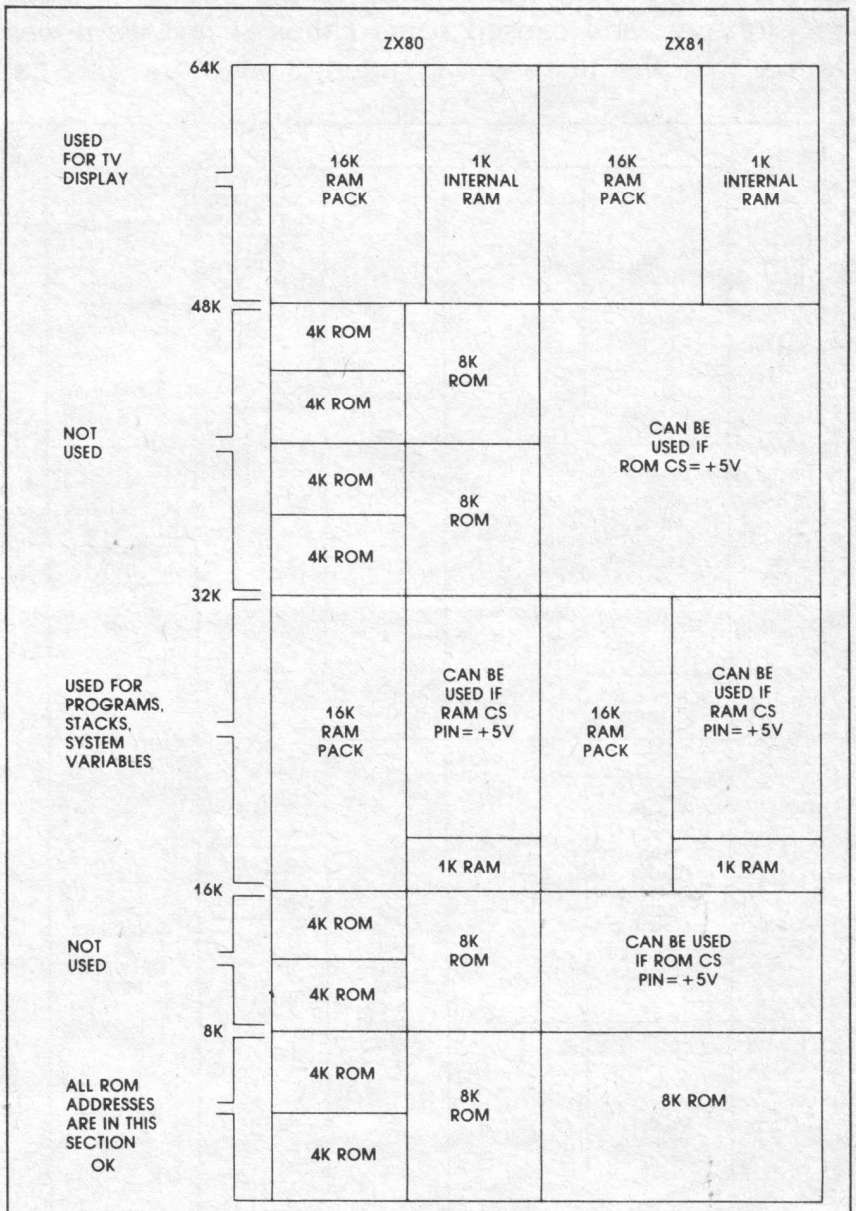

Figure 1. Sinclair's Intended Memory Map

Without turning off the 1K internal RAM, the 16K RAM could not operate. The 1K RAM repeated itself through the entire 16K section and got in the way when 16 separate 1Ks were installed instead. Sinclair's 16K RAM avoided this problem because when attached to the computer it not only blocked off any connection to the expansion port by covering it, but also used the only free memory locations. It also did all of its address decoding inside the pack, so could not be turned off by operating the $\overline{\text{RAM CS}}$ line.

When Sinclair saw the growing market for his computer and that other manufacturers wanted to use it as a basis for experimenting with computers, he redesigned it, giving greater flexibility to the memory map. The result was the ZX81.

The internal design was much the same as the ZX80—Figure 1— but this time an extra connection in the expansion port had been put to good use. It was called the $\overline{\text{ROM CS}}$ pin. By operating it in the same way as the $\overline{\text{RAM CS}}$ pin, the ROM could be taken out of the memory map. This was located on pin 23B on the expansion port and caused some chaos at first, as it was found that the first Sinclair attempt at more external memory, the 3K RAM pack, did not work with the ZX81. This was because inside the pack the pin was permanently set to binary 0, preventing the ROM from being turned on. The only solution to the problem was to cut the connection to the pin within the pack, leaving it unconnected to anything within the pack.

People could then free some space in the memory map for their use from the outside of the computer. That was a great advantage, as many of the people who had bought ZX80s or ZX81s did not want to have to fiddle inside their precious computers with a soldering iron for fear of damaging them.

Now many firms that had cut their teeth in the ZX80 could greatly expand the ZX81's flexibility and use with add-on boards.

This change also led every manufacturer to try to use the same areas for different uses. Also because ZX customers were not willing to pay a great deal for any extras to their machines, costs had to be cut to fit the market. So they took a tip from Sinclair and reduced the number of lines they used for addressing.

But then we had the same problem before—that one port covered more than one memory location; some cover up as much as 8K. That, of course, means that when one device is using that area, no other devices can use the same address.

Figures 2, 3 and 4 show many available add-ons for ZX and TS computers, plus their addresses. These charts also show, by putting into different columns, the devices

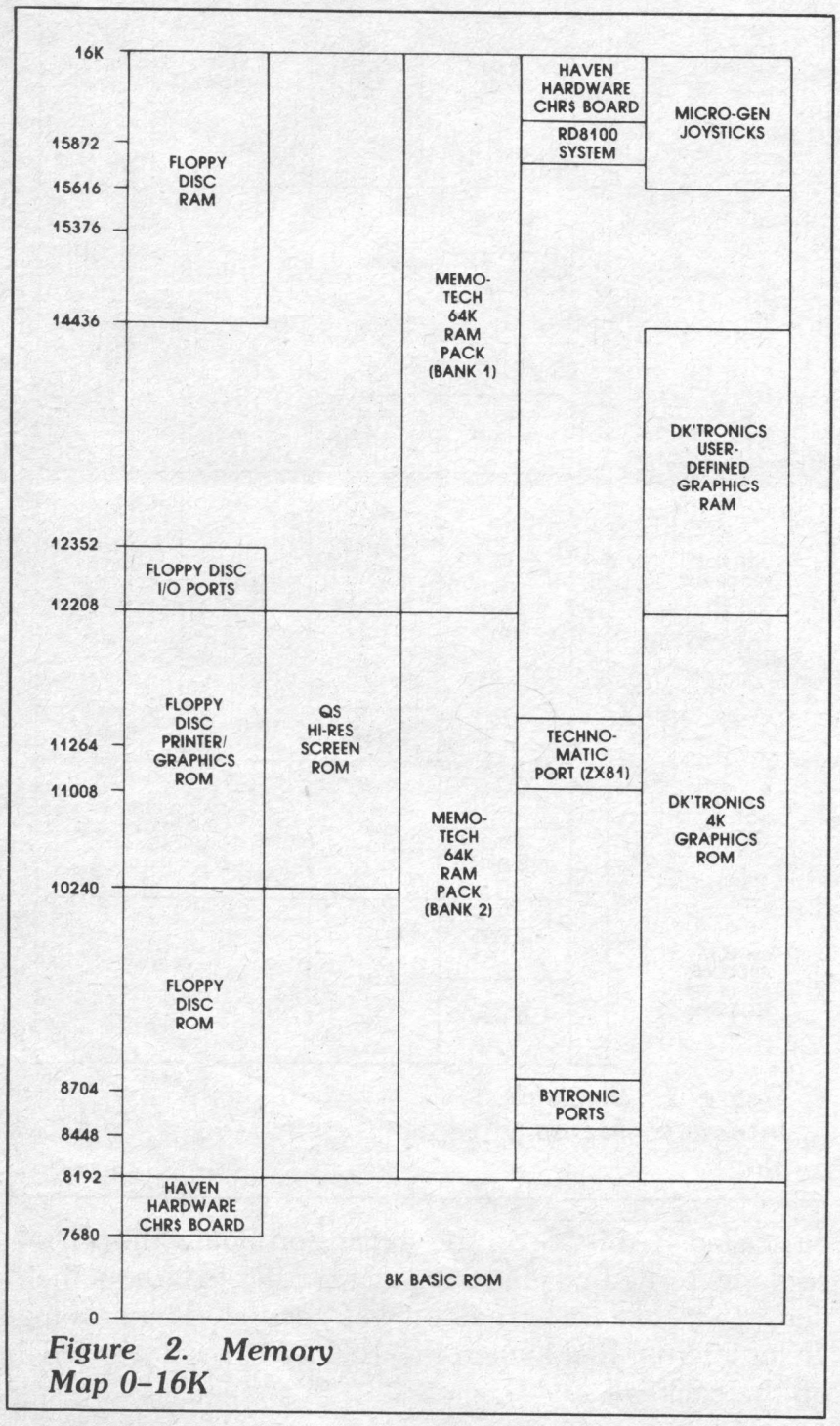

Figure 2. Memory Map 0–16K

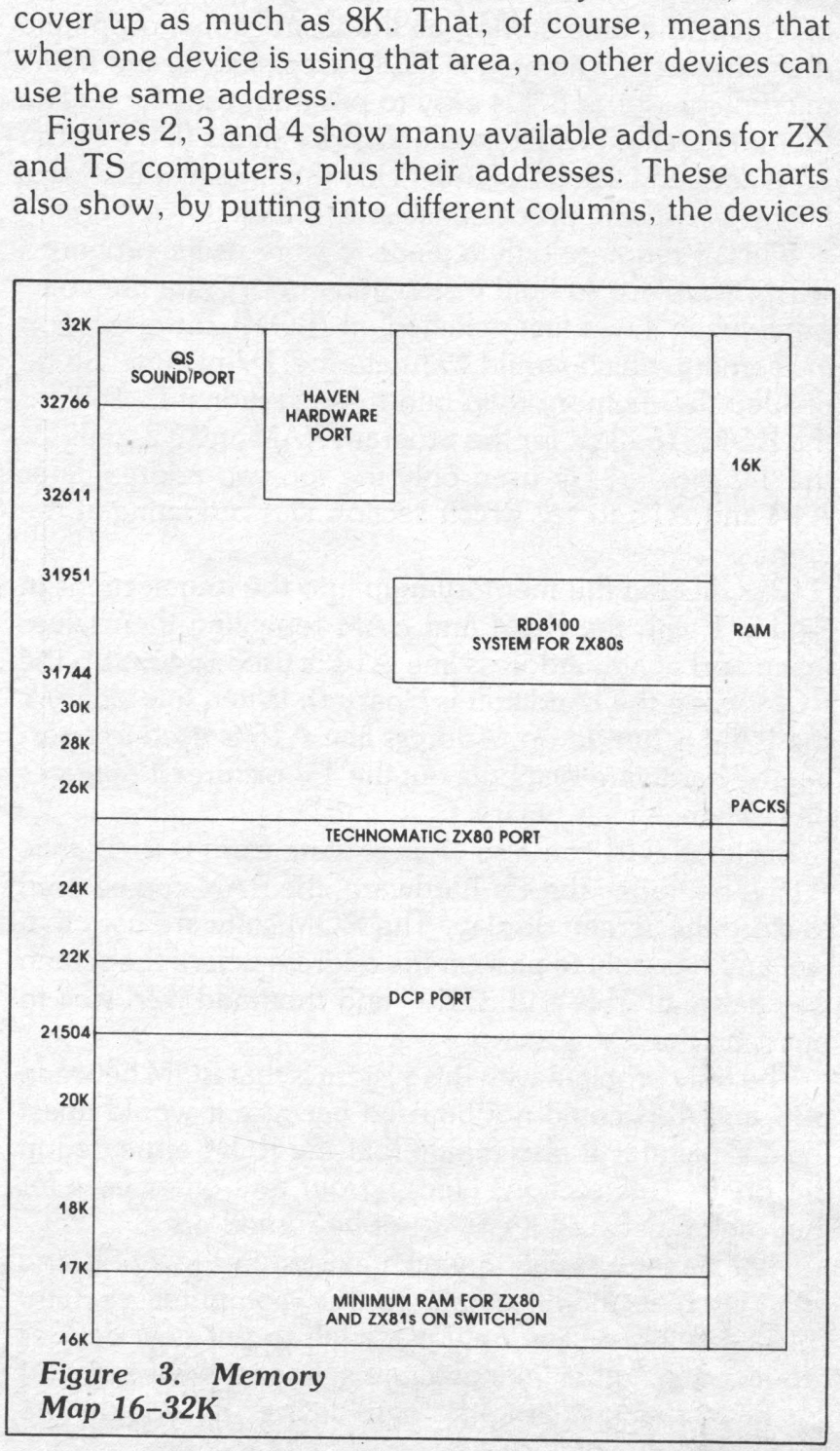

Figure 3. Memory Map 16–32K

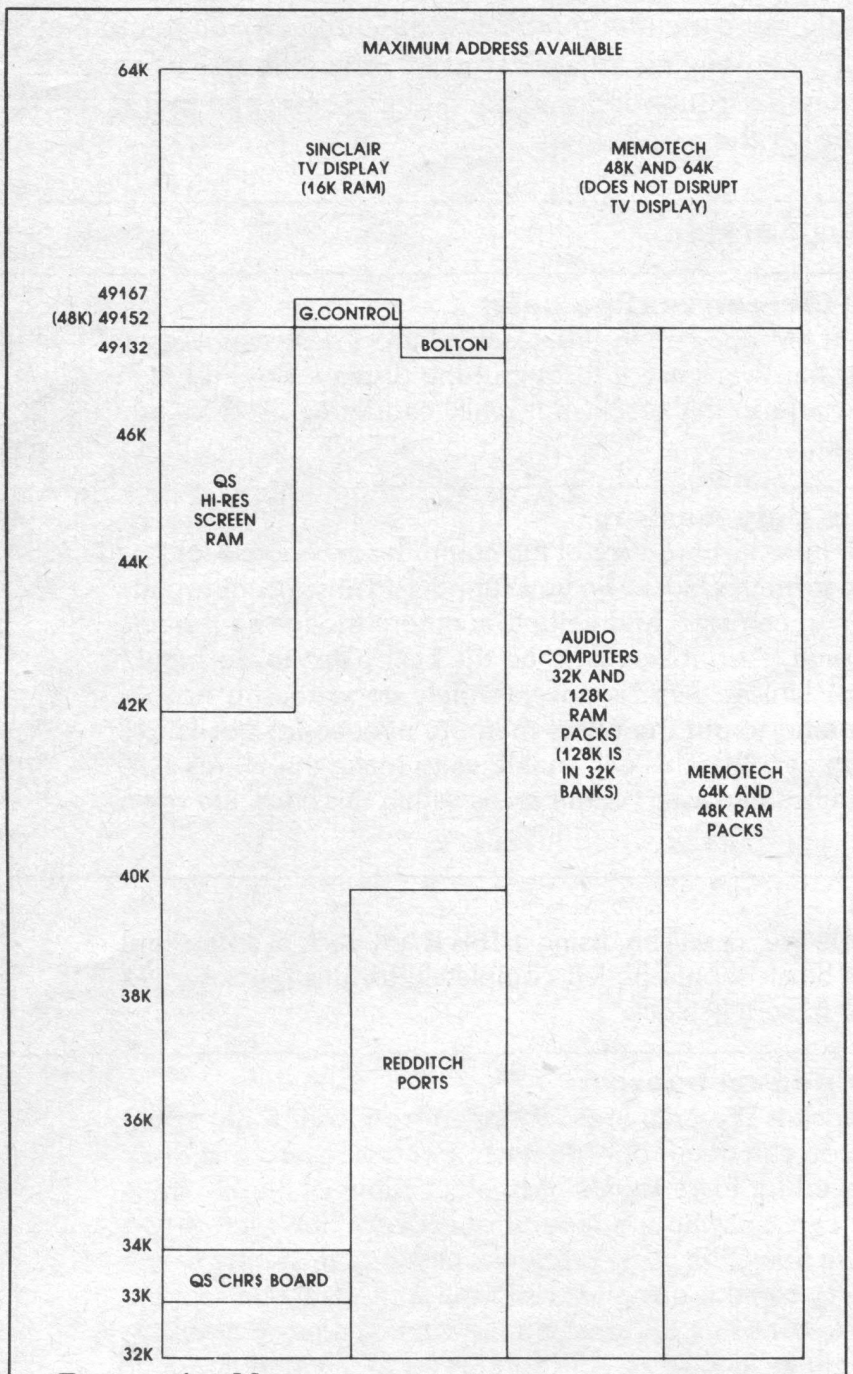

Figure 4. Memory Map 32–64K

which cannot be used together. If I have omitted any devices which fit either the ZX80 or the ZX81, please send me, care of SQ, the name and address of the manufacturer, what the device does and how many addresses it covers, including all the ports if it contains more than one.

I have included two RAM expansions as they all have to start at 16K and work their way upwards. That memory must be continuous for BASIC, as the program would crash if it tried to store memory in a RAM location which was not there.

For more details on these devices, contact the manufacturer (see accompanying list). **SQ**

*In addition to the British companies in Steve Adam's **ZX Memory Map**, these U.S. suppliers have provided SQ with the following memory data.*

Company	Device	Addresses Used
Byte-Back Co.	BB1 control module	32764-32767
	Modem-RS232	38 & 39 in ROM
	M64 memory	8-16K deselected in 2K increments
Memotech	Hi-res graphics board	8-10K
	Centronics parallel interface	10-10.5K
	RS232 interface	10.5-11K
	Balance of addresses open so far, but disk drive I/F *may* run 11-13 or 14K region	
CAI Instruments Inc.	C-40 printer	8-10K plus 2K paging from 14-16K
	Stringy floppy	10-12K
	Modem/ASCII converter	13-14K
Data-assette	ZX99 tape controller	ZX/TS1000's own ROM in memory address space (i.e., from 8192 to 10239) but since the address is not fully decoded, ZX99 will respond to any address in range 8192-16383

New peripherals for ZX/TS computers have proliferated rapidly since the computer's introduction. Not all peripherals, or add-ons, are compatible with each other, as Steve Adam's accompanying article describes. Because of this problem, Nick Lambert of Quicksilva in England proposed this memory map for future peripherals. If designers agree to this sort of convention, you may be able to use all your ZX/TS accessories at the same time!

ZX Computer Memory Map

64K 48K	Screen Display Area	**User RAM (Screen routine only)** The user RAM appears in this block where the screen display routines and hardware use it to output the display. D_FILE can appear anywhere in this block so it really cannot be used for any other purpose.
32K	More I/O & Memory Blocks Software Switches I/O Space QUICKSILVA Hi-res memory	**Read/write only memory** Due to the internal hardware of the computer no opcode fetches can be made from this block. So you cannot run any machine code from here. You can read and write to memory though so it is fine for data storage. Also it seems to be the best place to do Input/Output from. Unless there is an extremely good reason not to. Quicksilva intend to put the 6K of memory needed for our Hi-res board starting at 32K. This will enable us to make the Hi-res software run significantly faster. Other areas within this block are open to debate.
16K	User RAM	**User RAM** Most people are, or will be, using a 16K RAM pack of some kind so really this block should be left completely for this purpose. No ROM or I/O from this block.
14K 12K 10K 8K	? Voice Graphics & Extended BASIC D.O.S. & Network	**ROMs (for add-on boards)** This 8K block is the only area left, apart from user RAM space from which you can do an opcode fetch. For this reason this block should be saved for extra ROMs. Actual functions of the 2K areas in this block can be swapped around of course. However, if two companies are doing the same product which requires some of this ROM space, to avoid wasting valuable space, it would be sensible for both to use the same 2K area. So here it would be advisable to settle for a definite layout, i.e., 8K to 10K for Disc Operating Systems, etc.
0K	BASIC	**ROM (Sinclair BASIC ROM)**

List of Suppliers and Addresses:

Audio Computers
87 Bournemouth Park Rd.
Southand on the Sea, Essex
England

Byte-Back Co.
Rt. 3 Box 147
Brodie Road
Leesville, SC 29070
(803) 532-5812

CAI Instruments Inc.
2559 Arbutus Court
Midland, MI 48640
(517) 687-7343

DCP Port
Microdevelopments
2 Station Close
Lingwood, Norwich
MR 12 4AX, England

Bolton Electronics
44 Newlands Dr.
Bolton, Lancs BL5
BL5 1DP, England

Bytronic
88 Russell Bank Rd.
Sutton Cold Field
W. Midlands C74 4RJ
England

Data-assette
52 South 3d Street
Oxford, PA 19363
(215) 932-3626

Dk'tronics
23 Sussex Road,
Gorleston
Great Yarmouth, Norfolk
England

Ground Control
Alfreda Ave.
Hullbridge, Essex
SS5 6LT
England

Memotech Ltd.
3 Collins Street
Oxford OX4 1XL
Tel. 722102/3/4/5
England

Micro-Gen
24 Agar Crescent
Bracknell, Berks.
England

Redditch
21 Ferney Hill Ave.
Redditch, Worcs.
England

Haven Hardware
4 Asby Rd.
Asby Workington
Cumbria, CA14 4RR
England

Memotech
7550 West Yale Ave.
Suite 200
Denver, CO 80227
(303) 986-1516

Quicksilva
95 Upper Brownhill
Maybush, Suton, Hants.
England

Technomatic
17 Burnley Rd.
London, NW 10
England