HOME COMPUTERS IN ARCHAEOLOGY

M J Rains

University of Durham, Dept. of Archaeology
46 Saddler St., Durham City, DH1 3NU

ABSTRACT

Most recent work on the use of microcomputers in archaeology has been orientated towards hardware in the ‘2000+ price range and has largely overlooked the potential of cheaper 'home' micros costing a tenth of this price or less which have often been dismissed as unsuitable for archaeological work largely because of their lack of disc storage facilities.

Use of a Sinclair Spectrum based system on recent excavations in Orkney has shown that this problem can be overcome by the use of data compaction techniques to allow over 500 context records or 1000 small find records to be held within the 48K RAM of the computer. Future hardware enhancements offer further scope for expansion.

The programs implemented to date are for the storage, handling and retrieval of site data and the analysis of stratigraphic information to facilitate matrix construction. Their development and use has demonstrated how a Sinclair Spectrum (or similar) can form the heart of a low cost/low risk computer system with wide capabilities eminently suited to use on small and medium sized archaeological projects.

Over recent years, one of the fastest growing sectors of the computer industry has been the so-called "home computer". This sector of the market really only came into existance with the introduction of the Sinclair Research ZX80 machine in 1980. Since then, development of the products and growth of the market have been spectacular. Prices have fallen as fast as the capabilities of the available machines have risen and there are now well over a million home computers in use in Britain.

Parallel to this growth in the home computer market there has been a similar growth in the use of computers in archaeology. This growth has been both in the number of archaeologists who are using computers and in the number of ways in which they are using them. However, most of the computers now in use in archaeology in Britain are what are often referred to as business machines (Apples of one variety or another, Sirius, still some Commodore Pets, etc) generally costing in excess of ‘2000. In general, archaeologists in common with most other so-called "serious" computer users have steered clear of the home computers costing a tenth of this amount or less.

This is unfortunate as home computers can have several advantages over their bigger brothers. Their very low (and falling) prices allow them to fit into budgets which would be totally swallowed up by a more expensive business machine. Because of their low price, they can be regarded as almost consumable or disposable. If one is broken, it is a minor annoyance not a major disaster. Furthermore, the fact that so many are in use across the country means that an increasing number of people are coming into archaeology who are familiar with these machines and for whom they hold no terrors, whereas business computers are still the preserve of the few.

With this in mind, a home computer based system was put into experimental use on recent excavations at Tuquoy, Westray, Orkney (Owen 1982; Owen 1983). The hardware configuration employed consisted of a Sinclair Research ZX Spectrum 48K RAM computer linked to a ZX printer, a portable cassette tape recorder and a domestic 12" black & white television, at a total cost of '260.

The system was viewed initially as simply an electronic filing cabinet or card index based on three files of data, namely a context catalogue, a small finds catalogue, and an environmental samples catalogue, these been the three main unique number sequences present in the site records. BASIC programs were designed which would allow the four basic operations of input, editing, searching and output to be performed. Before the programs could be written, however, the actual format of the data and the way in which it was to be stored had to be decided. In this, a trade off had to be made between the number of records and the size of each record.

The Spectrum computer has 48 kilobytes of random access memory (RAM) of which between seven and eight kilobytes are required by the machine operating system, the printer buffer and the video display, leaving forty kilobytes available for the user's program and data. This cannot be augmented by the use of external files without the addition of disc drives which at the time was neither technically or financially feasible on the Spectrum (1).

Based on an average program size of ten kilobytes, a total of thirty kilobytes is available for data. On the face of it this means thirty thousand characters which can represent a 5000 word long essay on one small find or 10 word long sentences on each of 600 small finds (approx.) This at first sight may seem to be a serious limitation on the viability of the system, but faced with this and the inability to easily increase the amount of available memory, techniques were developed to minimise the number of bytes required by each data record.

The first of these techniques concerned verbal descriptions of both contexts and small finds. These are often an important part of the site record, but in terms of computer memory are very wasteful. For example, the phrase "Dark brown humic sand with charcoal flecks" contains 42 characters (including spaces) and if stored in computer memory in a literal form would occupy at least 42 bytes (2). It was therefore necessary to find some memory efficient way of storing context descriptions without in any way affecting the amount of information conveyed by these descriptions.

A byte of computer memory can represent any one of 256 different things. These 'things' can be anything but in the case of the Spectrum are usually either numbers in the range 0 to 255, machine code instructions, or characters, symbols and keywords. In thislatter case, the Spectrum uses an expanded form of the ASCII character set (3). This includes all the letters (upper and lower case), digits, punctuation marks and symbols on the keyboard, along with various graphics characters, control codes and all the keywords of Sinclair BASIC. So one byte, depending on the value it holds, could represent an alphanumeric character, a BASIC keyword, or a machine code instruction, all of these been represented by a series of bytes each with a value between 0 and 255.

From this it can be seen that the computer can use one byte to represent not only a single character, but also a complete word or even a whole command, provided that there are only a limited number of such words and that they are all known in advance (Sinclair BASIC uses 91 keywords which are fixed and may not be expanded). For our purposes therefore, if we could limit our context description vocabulary to 256 or less pre-defined words, to each of which we could allocate a code from the ASCII table, then each word could be represented in memory by one byte and our description of "dark brown humic sand with charcoal flecks" could be stored in only seven bytes.

To find out if this was possible without in any way stifling the literary style of the site supervisor, a word count was taken of all the context descriptions in the previous seasons excavtion records. In the first instance, all the synonyms, link words (and, with, etc.) and most one-off occurances were discounted, leaving a list of only thirty-nine primary words from over 120 context descriptions. When most of the synonyms, link words and rare words were added back in, the total rose to only ninety-two words.

From the point of view of user-friendliness, it was felt that encoding of input should be avoided as far as possible. The computer program therefore includes a list of the vocabulary in alphabetical order. Context descriptions are typed in in plain English, and the computer checks each word against the vocabulary. If it is found, its code is stored in memory, if not then an error report is given. The maximum number of words allowed in a description has been fixed at twelve as this has been found to be sufficient to cover most needs, therefore twelve bytes of memory space are required for each context.

The second data compaction technique used was concerned with numbers in general and the distinctions between real and integer numbers and numbers used as labels. The Spectrum, unlike the BBC and other micro-computers, does not distinguish between real and integer numbers. It always regards the number 22 as being 22.0000000 and uses six bytes of memory to store it. For many archaeological applications we are dealing only with integer or whole numbers. For example, we may have 22 or 23 pot sherds from a particular context but we cannot have 22.4736285 sherds. In this example however, we are still using 22 as a number in that we can do arithmetic on it. For example, we can divide the 22 sherds by 2 and have 11 in each pile. In other cases though, we use numbers only as labels. For example, we could not meaningfully divide Plan 22 by 2 and get Plan 11 - in this case 22 is only a name used to distinguish one plan from the others. we could equally as well have called it Plan A or Plan Fred.

As mentioned above, any whole number between 0 and 255 can be stored in one byte and so, to conserve memory space, the program has to intercept the computer's normal number handling routines and in fact treat all such integer or 'label' numbers as single characters or groups of characters. Of course if one byte can represent up to 256 numbers then two bytes together can represent up to 65336 numbers (256 squared). Therefore, a further important factor in data compaction becomes the establishment of number ranges. In other words, what are the largest and smallest values that a number can take, what are the possible intervening values and can the number be negative. Once these limits have been decided, then the minimum number of bytes necessary to store that number can be set aside in memory.

Some of the information to be included in a context or small find record is neither verbal or numeric but belongs to a third data-type which may be called binary; that is data which has only two possible values such as 'yes' and 'no'. Examples of the use of binary data-types in archaeological records include such questions as 'Did this context produce any small finds?', 'Has this small find been photographed?', etc.

In relative terms, such binary data offers the most scope for the dramatic use of memory saving techniques, as without any data compaction, recording the words 'yes' and 'no' requires three and two bytes of memory respectively. The first step in compaction is simple and straightforward; a single character can be used for each of the two possible answers, perhaps 'y' and 'n' or 'x' and space. This however, is still very wasteful because, as we have seen, one byte can represent up to 256 different answers, not just two.

Each byte of computer memory is composed of eight 'bits' and each bit can be thought of as a switch which can be either open or closed. To record 'yes/no' type data, therefore, we need only use one bit and in theory, we can record eight such data items in one byte. In practice on the Spectrum, this is relatively easy to achieve although perhaps not as easy as it might be.

If the value held in a byte is expressed as a binary number, then each digit of that number represents one bit of the byte. So what is needed is a function which converts binary numbers to decimal and vice-versa to enable the program to read and write yes/no type data to the individual bits of a byte. Unfortunately, the Spectrum does not have such functions in-built but they can easily be implemented in either BASIC or machine code. If the bits of a byte, each destined to hold one piece of yes/no type data, are numbered 0 to 7 with the zero bit corresponding to two to the power zero (2^0) and the seventh bit to two to the power seven (2^7), then to construct the decimal value of the byte, it is only necessary to go through the bits adding in two to the power of the bit number (0-7) if the bit is set, in other words if the answer to the 'yes/no' question is 'yes' (fig 1). The total value arrived at can then be poked into the relevant byte of memory.

Working the other way, there are several possible methods of extracting the bit values from the decimal value stored in a byte. Perhaps the easiest way is to attempt multiple subtraction of the bit values (from 128 to 1) from the total byte value (eg: 141). If a subtraction is possible (eg: 141-128=13) then the bit is set, if not (eg: 13-64), then the bit is not set. These algorithms for number conversion are easily coded in either BASIC or machine code and are well worth the effort as they enable the answers to eight seperate questions of the yes/no or either/or type to be stored in one byte of memory.

A useful technique which can be used in certain special situations is worth mentioning. As we have seen, one byte can represent up to 256 different values. In some cases, however, our range of possible values may be less than this. For example, the section catalogue for an excavation may run to only 100 items. As a number up to 127 can be stored in only seven bits, one spare bit is available for storing a single yes/no type answer, for example, 'Has this section been inked?'. This spare bit (the seventh) can be set or read by adding or subtracting 128 from the main value, so a value of 50 would mean section number 50 un-inked, whereas a value of 178 would mean section number 50 - inked. This technique can be a very useful way of squeezing extra information into bytes and so avoiding any waste of space.

As has been stated, the initial use of the Spectrum computer was seen only as an electronic filing system or card index and was seen very much as experimental. It was soon appreciated however, that even such a small and relatively un-sophisticated system could be put to many more archaeological uses. The first of these to be explored was the handling of stratigraphic information with the aim of aiding the production of stratigraphic matrices, a task previously only attempted on considerably larger and more powerful computers (Bishop & Wilcock 1976; Wilcock 1981).

The resulting CONSORT program was developed jointly on the Sinclair Spectrum based system described above and on a BBC microcomputer based system with dual disc drives and an Epson MX100 dot-matrix printer. The program is now fully operational on both systems, but during program development, it was found that each had its advantages and disadvantages.

The underlying theory of the algorithm which the CONSORT program implements is that developed and fully described by Magnar Daland in a forthcoming volume of Scottish Archaeological Review (Daland 1984) and need not be repeated here. What follows is a brief description of the program's input and output and a discussion of some of the problems encountered in its development, particularly those related to the choice of computer hardware. The examples given are taken from the Tuquoy site mentioned above and from excavations on the Brough of Birsay, Orkney (Morris 1982; Morris forth.).

The CONSORT program requires as input, all the known stratigraphic relationships from the site (Fig 2) in the form of stratigraphic links. For example, 123/124 is a single link which expresses the two complimentary relationships "context 123 is later than (above) context 124" and "context 124 is earlier than (below) context 123". Relationships of equality (eg: 124=126) are not included as such but are implied by such sets of relationships as 123/124, 124/125, 123/126 & 126/125. If these are the only links in which contexts 124 and 126 appear then, as far as the recorded relationships go, they can be treated as equal in stratigraphic terms. The introduction of further links such as 124/127 and 127/126 would destroy this equality and imply the additional link 124/126. In general, this and other implied links (eg: 123/125) need not be included but no harm is done if they are. How this list of links is compiled will depend on the recording methods of the site concerned. In the case of the trial sites, this involved a thorough search of all site-books, plans and sections. The more relationships that can be fed in, the more closely detailed will be the picture which will emerge. One of the Birsay sites, where the level of recording was very variable, produced approximately 1500 links from 500 contexts.

Once all the known stratigraphic links have been entered, the program performs five main steps before output:-

1. All contexts are sorted into ascending and descending order and assigned to steps within a hierarchy based on the recorded links.
2. All repetitious links are removed leaving only those linking directly related contexts. For example, of the links 123/124, 124/125 and 123/125, the last would be discarded.
3. A list is drawn up showing for each context, which it is directly under and which it is directly over.
4. Using this list, all contexts are placed into chains of directly related contexts eg: 123/124/127/126/125. The head of each chain will be either a context with no recorded under-relationship such as turf, or one which has already been placed in another chain, thus linking the chains together.
5. All the chains are formatted ready for output.

The output of the CONSORT program takes the form of a matrix or hierarchy diagram (fig 3) in which all the chains are listed down the page in their correct vertical relationship to other chains. For example, chain 1 contains a continuous series of 21 contexts from 4 (sub-turf) to 153 (natural). Chain 2 also has at its head context 4 but only contains 10 contexts ending in 647 which appears in chain 1. Chain 5 only contains 4 contexts and links into chain 1 at both top (473) and bottom (520). Chain 12, also 4 contexts long, links into chain 5 through 275 at the top and into chain 1 through 714 at the bottom. It should be obvious that a more conventional matrix diagram can easily be constructed by adding horizontal and vertical lines to the program output. This has been done for chains 1,2,5 and 12 (fig 4).

The main problems encountered in programming CONSORT were those connected with memory size, mass data storage and printout facilities. The operation of the program requires that all the original data (ie: the stratigraphic links) be present in memory at the same time, as use of a random access disc file is not possible on the Spectrum and would increase run-times to an unnacceptable level on the BBC (4). Even using the data compaction techniques already described, each link requires 4 bytes of memory (2 if all context numbers are less than 255). In addition, the program sets up various number and character arrays in memory whose dimensions are dependant mainly on the highest context number present. The Spectrum is therefore able, within its 48K memory to handle, for example, 5000 links for 1000 contexts which would seem to be acceptable figures for many applications. On the other hand, the BBC with only 32K of memory, is limited to only 1600 links for 1000 contexts or 2500 links for 500 contexts which is quite restrictive. Even to achieve this the program has had to be split into three parts which together use four intermediate disc files (5). In this respect, therefore, the Spectrum seems much better suited to the task.

However, two further problems have arisen with respect to the Spectrum. As the original data in memory is destroyed during various stages of processing and then needs to be refreshed for the next, it needs to be loaded from mass storage three times during a run of the program. This presents no problem with disc drives, but with the cassette storage of the Spectrum it requires much manual starting, stopping and rewinding of the tape. This problem has been overcome by the use of a ZX Microdrive for mass storage. A further inconvenience is the narrow width (32 columns) and, to a lesser extent, poor quality of print-out from the ZX printer included in the original Spectrum configuration. However, with the addition of ZX Interface 1, required for Microdrive operation, it is possible to connect the Spectrum to any full size printer with an RS232 standard connection. The total cost of these Spectrum upgrades is under '100. In balance therefore, an enhanced Spectrum system would seem more suitable than the more expensive BBC based system for the implementation of this type of program.

Possible future applications for the Spectrum based system are currently under investigation. One of these is word-processing, again, an application usually only considered for much more expensive systems. However, several commercial word processing software packages are available for the Spectrum which, when used in conjunction with Microdrives and a full size printer via Interface 1 as already described, along with an add-on typewriter style keyboard, give the Spectrum most of the features of better known but much more expensive word processing packages.

A further application currently under investigation is the digitization of site plans. On the Tuquoy site, a system of single context plans was used and these seem to lend themselves ideally to computerization as most individual contexts cover a small enough area to appear on a TV screen at a reasonable scale. In post excavaation, much time will be spent in bringing together and superimposing various combinations of these single context plans, a task which can be accomplished easily on a computer screen where scales, positions and plan combinations can be altered at will. With the use of a simple screen dump program, a paper print-out of any combimation of plans at any scale can be obtained. Efficient computerization of plans requires the use of a digital tracer which is available for the Spectrum for under '60 and gives a tracing accuracy to within 1%.

In conclusion, it would seem that a home computer based system can bring to archaeologists involved in small and medium sized projects, many of the advantages of computerized data processing and analysis normally associated with much larger and more powerful machines, at a fraction of the cost - all the hardware mentioned has a total cost under '450 (6) - and therefore at a fraction of the risk. The capabilities of the system are sufficient to keep the average archaeologist, new to computing, occupied for years and at these sort of prices, if the whole thing proves to be a white elephant (in the case of the Spectrum, a black one), or if someone accidentally drops a Microdrive in the Elsan bucket, then it may be a disaster, but at least its a cheap one.

NOTES

1. Recently, disc drives for the Spectrum have come on the market but they have limited capacity, offer few file handling facilities and cost more than the computer itself.

2. In practice, the Spectrum would allocate at least two extra bytes of memory to indicate variable type and length.

3. ASCII - American Standard Code for Information Interchange.

4. Current versions of the CONSORT program give run-times averaging approximately ten minutes per hundred contexts on the BBC system and fifteen minutes per hundred contexts on the Spectrum systm. These timings could be considerably improved by the use of machine code for part or all of the program.

5. The dual disc drive in use with the BBC system is a Torch ZDP Disc Pack. Making full use of the extra memory (64K) which this brings to the BBC would give the system greater capacity than the Spectrum but at four times the price.

6. The Sinclair QL, shortly to come onto the market at under four hundred pounds, appears to offer facilities far in excess of those available on any computer under 2000 pounds. It seems set to revolutionize the market yet again and render similarly priced micros such as the BBC, largely obsolete.

REFERENCES

Bishop, S & Wilcock, J.D.	1976	Archaeological Context Sorting by Computer: the STRATA program Science & Arch. 17, 3-12.
Coll, J,	1982	The BBC Microcomputer User Guide. London
Cambridge Comm. Ltd.	1983	Sinclair ZX Microdrive and ZX Interface 1 Manual. Cambridge.
Daland, M,	1984	A Procedure for use in Stratigraphic Analysis. Scot. Arch. Review, 1984.
Morris, C,D,	1982	Brough of Birsay Excavations 1981 in Archaeological Reports For 1981. Durham
Morris, C,D,	forth.	Brough of Birsay: Excavations & Survey 1974-81. Soc. Antiq. Scot. Monograph.
Owen, O,	1982	Interim Report on the Archaeological Rescue Project at Tuquoy, Westray, Orkney. Durham.
Owen, O,	1983	An Interim Report on the Second Season of the Archaeological Rescue Project at Tuquoy, Westray, Orkney. Durham.
Simpson, R,J & Terrel, T,	1983	ZX Spectrum User's Handbook. London.
Vickers, S,	1982	Sinclair ZX Spectrum Basic Programming. Cambridge.
Wilcock, J,D,	1981	STRATA - the Microcomputer Version. Comp. Appl. Arch. 1981 112-114.

Bit Numbers:-	7	6	5	4	3	2	1	0
	1	0	0	0	1	1	0	1
Bit Values:-	128	64	32	16	8	4	2	1

Question	Answer	Bit No.	Bit Value	Decimal Value
1	Yes	0	1	1
2	No	1	0	-
3	Yes	2	1	4
4	Yes	3	1	8
5	No	4	0	-
6	No	5	0	-
7	No	6	0	-
8	Yes	7	1	128
			Total Byte Value:	141

FIG 1: Decimal/Binary Conversion & Bit Extraction

123/124	123/126	124/125	126/125
133/276	139/306	139/368	141/314
141/368	142/296	142/368	144/145
151/147	151/148	152/283	152/314
152/471	156/155	156/798	157/253
160/316	161/274	161/308	161/314
161/771	151/577	152/296	152/368
152/478	156/591	156/799	157/301
160/313	161/275	161/313	161/332
162/274			

FIG 2: Stratigraphic links - part of the CONSORT program input.

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
4	4	4				4	4								
101	8	27				12	25	25							
94	26	170				14	85	53		94					
277	226	226					159	159		263					
222	77	77	77		77	77	261		77						
99	162	161	165		178		244		160	99					
526	274	274	274		9		431								
599	313				308		430		313			313	313		
597	512				496		444					372	475		
647	647				495		451					474		495	
473				473			472						473	509	
184				275	184		452				275	184		523	
527				510			148				334				
520				520			621				580			520	
650							354								
714							379				714				379
768							401								626
769							400								400
779							631								
629															
153							153								

FIG 3: CONSORT program output.

```
          004
    _______:_______
  :                 :
 101               008
  :                 :
 094               026
  :                 :
 277               226
  :                 :
 222               077
  :                 :
 099               162
  :                 :
 526               274
  :                 :
 599               313
  :                 :
 597               512
  :_________________:
           :
          647
           :
          473
    _______:_______
  :                 :
 184               275
  :                 :________
  :                 :        :
 527               510       :
  :_________________:       334
           :                 :
          520                :
           :                580
          650                :
           :_________________:
                    :
                   714
                    :
                   768
                    :
                   769
                    :
                   779
                    :
                   629
                    :
                   153
```

FIG 4: CONSORT program output in conventional matrix format.