



K

KEYPUNCH

Keypunch is a data entry system that relies on holes punched in specified locations on a rigid paper card. Using the Hollerith code (see page 769), individual letters and numbers can be represented by a series of strategically punched holes on a card divided into 12 rows of 80 columns. The keypunch system was the primary means of entering large amounts of data into a computer system, and is still widely used in data processing applications.

KEYWORD

To resemble human language, and thereby make computer programming more 'user-friendly', higher-level programming languages contain a vocabulary of words whose precise meaning is a part of the structure of the language. When the language processor (compiler or interpreter) encounters a *keyword*, it accesses a predefined procedure or routine. In BASIC, keywords form the command structure, so that words like PRINT, LET, GOTO, and IF...THEN have specific functions. Because of their specialised meanings, keywords cannot be found in, or used as, variable names.

KLUDGE

A *kludge* is a colourful term for a solution to a problem — usually a hardware problem — which arises after a system has been designed and largely executed. Rather than redesign the entire system, a kludge is hastily constructed from available parts and added to the system as cleanly and quietly as possible. Forgivable, and often creatively-inspired, in the hobbyist, kludges found in manufactured hardware on the market are inexcusable. On early versions of the Sinclair QL, reviewed on page 501, the company found that the operating system would not fit on the ROMs provided. They therefore introduced a kludge, in the form of an EPROM, which contained the extra operating system.

instructions. These names can be used to mean the address of those instructions; for example:

| Label Field | Instruction Field | Operand Field |
|-------------|-------------------|---------------|
| LOOP | LDA | TEST |
| EXIT | BNE | LOOP |

Here, LOOP, TEST and EXIT are all labels; they are Assembly language's equivalent of variables. When the program is passed to the assembler for conversion to machine code, the labels are replaced by the addresses they represent.

One form of label is a statement, such as a REMark statement in BASIC or a coded phrase within the body of a computer program, which identifies the specific task to be carried out by the portion of the program to which it is attached. Labelling in this sense is a key element of structured programming in that it helps the programmer to keep track of what the program is actually doing at every stage.

Structured programming also makes use of procedures. These are essentially subroutines that are labelled and which can be referred to and acted on by their labels. For example, the commands required to create a graphics backdrop in a game program might form a procedure called SCREEN. When the program encounters the word SCREEN, it executes all the commands pointed to by that label.

A label is also a small file on a larger magnetic tape or disk data file that identifies the nature of the data held in the file. A volume label identifies the contents of the entire tape or disk file. Header labels directly precede and characterise individual data files.

Finally, some sophisticated spreadsheet programs have the facility of labelling related regions of cells, then performing calculations on the entire region simply by incorporating the label into a formula. For example, actual sales figures of a product are recorded for a one-year period. To predict what sales will be for the same period one year in the future, a formula calculates a given percentage of growth, or decline, and is applied to the actual sales numbers. Without labelling, making this happen would require including the row and column number for each actual figure, like this:

$$A4 * 110\% = B4$$

and the formula must then be copied to the relevant portions of the spreadsheet, requiring several steps. With a labelling facility, however, the same formula looks like this:

$$\text{Actual Sales} * 110\% = \text{Projected Sales}$$

with labels identifying the significant regions. This formula now applies automatically to all of the cells in the area labelled 'Actual Sales', with the results automatically being assigned to the region labelled 'Projected Sales'. This speeds up the creation and operation of the spreadsheet and makes the results much clearer to read.

Flying Kludge

The obtrusive wires attached to the Spectrum PCB connect an additional reset switch into the logic, thus turning the Spectrum into the Spectrum+ — a fine example of kludging.

The extremely expensive alternative would have been to redesign the PCB entirely, incorporating the new wires into the copper track patterns on the board



CHRIS STEVENS

LABEL

Names, or *labels*, serve to identify an object and set it apart from other objects. Labels occupy a variety of positions in computer operations, so the term has several distinct usages, the most important of which is in Assembly language, where labels are names attached to individual