success. Listeners as far away as Belgium, France, England, Germany and Denmark reported success in loading the programs. This international attention was increased when the Dutch broadcasting service began transmitting BASICODE-2 on its external network.

BASICODE is founded on the 42 keywords and 11 symbols that most of the machines able to make use of the language have in common. A BASIC keyword is not stored as its component characters but is held in the form of a single-byte token, a symbol which represents the keyword. For example, the keyword LEFT$ is represented on the Commodore 64 by a single byte containing the value 200, rather than by five bytes containing the ASCII values for L, E, F, T and $. This makes the work of the BASIC interpreter much more efficient and uses far less RAM. However, although every computer uses tokenisation for storing and interpreting a BASIC program, each machine uses different values for its tokens. The problem was solved by providing two translation programs, BASICODE-Save and BASICODE-Load. After a program has been written in BASIC, it is SAVEd by using the BASICODE-Save program, which substitutes BASICODE standard tokens for the computer's own BASIC tokens and produces on tape a standard BASICODE program. This program can then be loaded onto another machine using the BASICODE-Load program, which substitutes the machine's BASIC tokens for the BASICODE version.

This raises a major question — how to ensure that the various types of computer read and write to tape in the same way. Once again, although all the machines use the same principle for LOADing and SAVEing programs on tape, in practice a program tape produced by one computer might be very different from another machine's tape. Not only may the data be written and read to the tape at different transmission rates, but also the start and stop bits (the markers that tell the computer where the data begins and ends), and the checksum methods (the system whereby the machine checks that data has been transmitted correctly), may also differ radically. The solution adopted was to suppress the individual machine's own tape handling methods and impose a common *audiocode* format for transmission.

In this format, data is transmitted at 1,200 bits per second. Each byte of data, preceded by a start bit (value 0), is transmitted least significant bit first, and followed by two stop bits (both with a value of 1). For example, the ASCII value of 'A' is 65 — 01000001 in binary — and this would be transmitted in audiocode as 01000001011. A leader marker, consisting of a sequence of stop bits transmitted for five seconds, indicates the start of a BASICODE program. This is then followed by the code for 'start text' (82 in hexadecimal). The BASICODE program is followed by a checksum byte, which enables the computer to check the accuracy of the transmitted data. Another five-second stop bit sequence indicates the end of data transmission.

Although nearly all machines can be adapted to BASICODE by software alone, the TRS-80 models I and III and the Video Genie require a small interface to be added to them in order to allow tapes to be read in correctly. The handbook provided with the Basicode-2 cassette gives full details on how to construct the interface. For the less DIY-inclined, a printed circuit board can be obtained from the Netherlands TRS-80 users group.

In order to write a BASICODE program, you must first load the BASICODE-Save program. This program not only allows the newly-written code to be SAVEd on cassette in a standard format, but also provides a list of subroutines for the computer that are unique to that particular machine. These routines are stored between lines 0 and 999 and, therefore, are unavailable to the programmer.

The reason why these routines are provided by the BASICODE-2 translation program is because a command common to several machines — such as the instruction to clear the screen (CLS) — may be executed in different ways. Instead of using the CLS command, the programmer uses GOSUB 100, which refers to the BASICODE subroutine that performs this function.

The first line a programmer writes should be in the form:

```
1000 A=(value): GOTO 20: REM program name
```

where (value) is the maximum number of characters used by all the strings together. From this point onwards, the user is given a free hand to program as desired. There are, however, a number of

This diagram shows how each of the machines catered for in BASICODE conforms to the format. Those machines with specifications that do not measure up to the standard are marked with a cross. Machines marked with a tick, have facilities beyond what is allowed by BASICODE

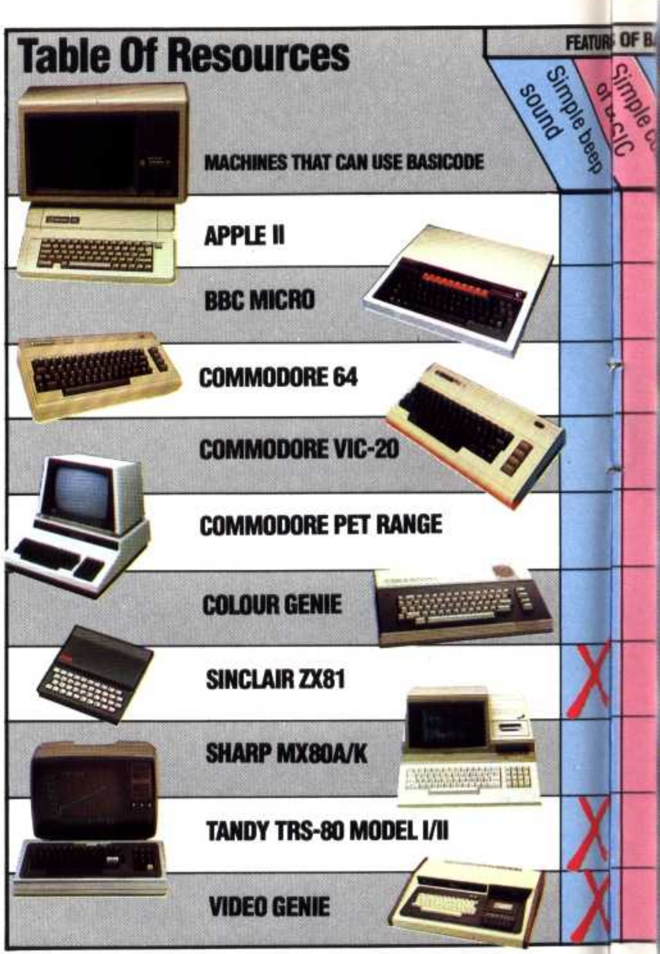

Table Of Resources

MACHINES THAT CAN USE BASICODE	Simple beep sound
APPLE II	
BBC MICRO	
COMMODORE 64	
COMMODORE VIC-20	
COMMODORE PET RANGE	
COLOUR GENIE	
SINCLAIR ZX81	✗
SHARP MX80A/K	
TANDY TRS-80 MODEL I/II	✗
VIDEO GENIE	✗

FEATUR OF B
Simple of BASIC
Simple c

Operation Commands
This is a list of commands and operations that are allowed by BASICODE. Note that many machines will have a far greater number of keywords within the BASIC interpreter that are not recognised within BASICODE

ABS
AND
ASC
ATN
CHRS
COS
DATA
DIM
END
EXP
FOR
GOSUB
GOTO
IF
INPUT
INT
LEFTS
LEN
LET
LOG
MIDS
+
-
*
/
∧
NEXT
NOT
ON
OR
PRINT
READ
REM
RESTORE
RETURN
RIGHTS
RUN
SGN
SIN
SQR
STEP
STOP
TAB
TAN
THEN
TO
VAL
<
>
<>
<=
>=