

Figure 14 PBASIC Help System – Command Reference

The screenshot shows a window titled "PBASIC Syntax Guide" with a navigation menu on the left containing "Welcome", "PBASIC Command Reference", and "Alphabetical Listing". The main content area is titled "BRANCH" and includes icons for different processor models (1, 2, 2E, 2SX, 2P) and an "Examples" link. The text describes the syntax for BS1 and other processors, the function of the command, and provides a table of quick facts. It also shows an example of how to use BRANCH to organize an IF-THEN statement.

**BRANCH**

Syntax : **BRANCH** *Offset* (*Address1*, *Address2*, ...*AddressN*)  
 Syntax : **BRANCH** *Offset* [*Address1*, *Address2*, ...*AddressN*]

**Function**  
 Go to the address specified by offset (if in range).

- **Offset** is a variable/constant/expression\* (0 - 255) that specifies the index of the address, in the list, to branch to (0 - N).
- **Addresses** are labels that specify where to go. **BRANCH** will ignore any list entries beyond offset 255.

\* **Note:** Expressions are not allowed as arguments on the BS1.

**Quick Facts**

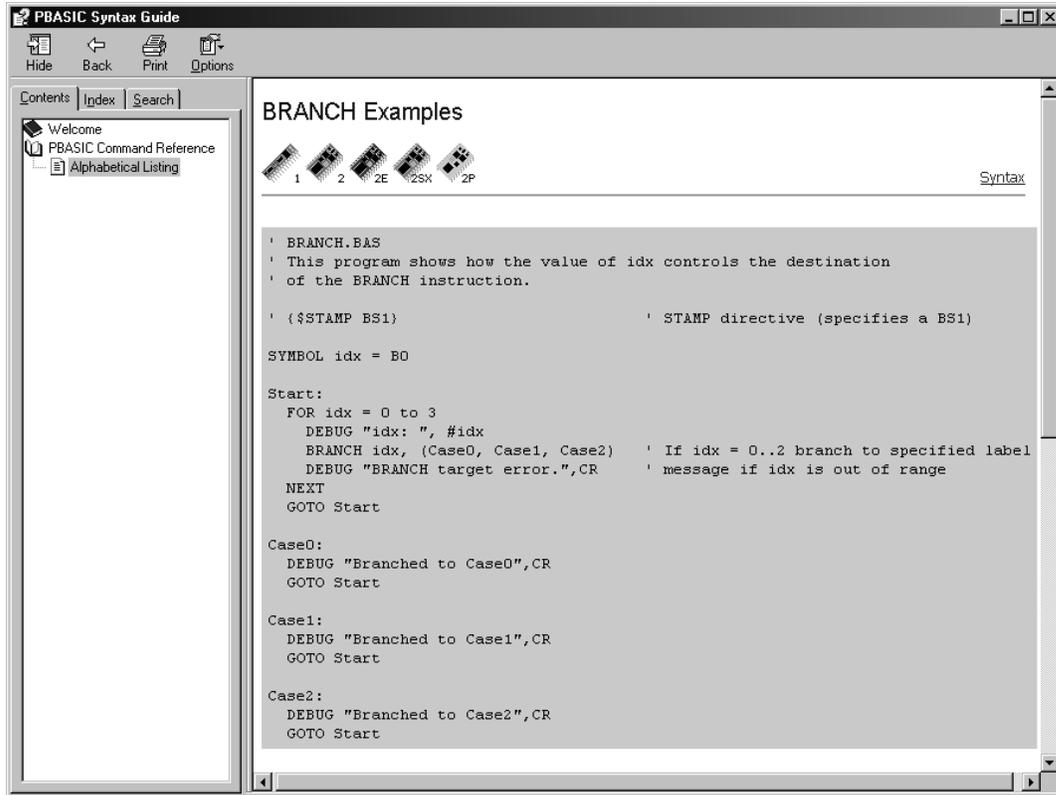
	BS1	BS2, BS2e, BS2sx and BS2p
Limit of <i>Address</i> entries	Limited only by memory	256

**Explanation**  
 The **BRANCH** instruction is useful when you want to write something like this:

```
IF value = 0 THEN case_0      ' value = 0: go to label "case_0"
IF value = 1 THEN case_1      ' value = 1: go to label "case_1"
IF value = 2 THEN case_2      ' value = 2: go to label "case_2"
```

You can use **BRANCH** to organize this into a single statement:

Figure 15 PBASIC Help System – Explanation BRANCH command



**Figure 16**  
**PBASIC Help System – Program example BRANCH**

In the program examples we will often use commands that have not been explained. It is normal in such a reference as this book to page forward and backward sometimes and use the on-line help for more detail.

To get a clear structure we will start with each PBASIC command on a new page. You can use the empty room for some comments.

The command for serial I/O SERIN and SEROUT will be directed to I/O pin 16. In these cases the serial I/O works over the programming connection and the serial I/O is redirected to the

---

**IOTERM BS2p-40**

IOTERM block#

Switching the I/O blocks of BS2p-40.

**It means:**

block# I/O block number (0=MAINIO; 1=AUXIO)

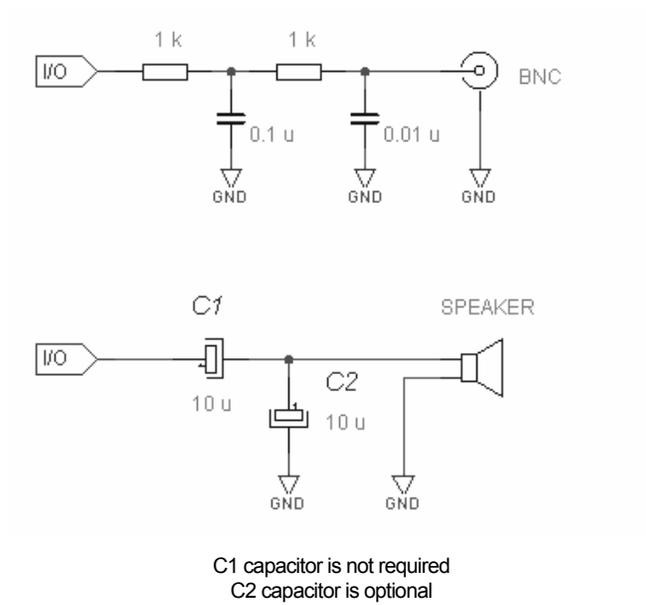
**Example:**

```
high 0      ` Output0 (Pin5) Hi
ioterm 1   ` Switch to Auxiliary I/O Pins
low 0       ` Output0 (Pin21) Lo
ioterm 0   ` Switch to Main I/O Pins
low 0       ` Output0 (Pin5) Lo
ioterm 1   ` Switch to Auxiliary I/O Pins
high 0      ` Output0 (Pin21) Hi
```

**Remark:**

Use the IOTERM command for switching between the BS2p's I/O blocks with a single parameter (0=MAINIO; 1=AUXIO).

This command only applies to the BS2p-40.



**Figure 84 BS2 Tone Output**

When connecting an external amplifier inserting a filter circuit is recommended. At the same time such a filter circuit protects the I/O pin against a short-circuit.