

Figure 53 Connecting LEDs to PortB

### 4.3.2 Seven-Segment Displays

Seven-segment displays can display the figures of our numbering system and a couple of special characters.

There are many types of seven-segment displays from different manufacturers. Basically, this type of display consists of a number of LEDs with connected anodes or cathodes.

In our application example, the type SA03-11 display made by Kingbright is used. Figure 54 depicts such a display.

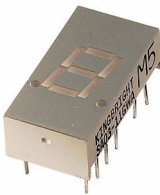


Figure 54 Seven-Segment Display SA03-11

**Caution:** To connect an LED display to the port of any microcontroller, adhere to the connecting diagram of the display used.

Figure 55 shows the segment assignment and pin configuration of an SA03-11 display.

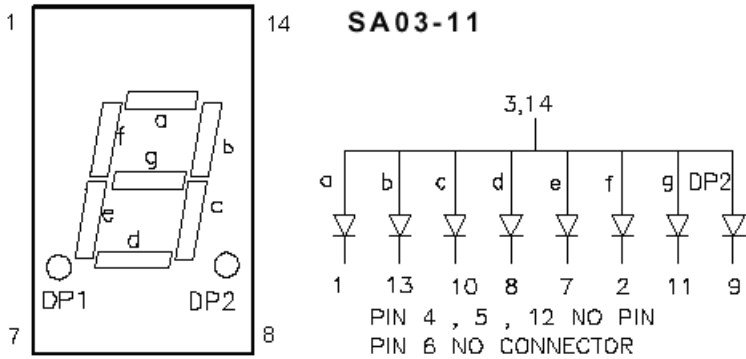


Figure 55 Segment Assignment and Pin Configuration of SA03-11

To display alphanumeric data with such a seven-segment display, it is necessary to define the control scheme.

As can be seen in Figure 55, the anodes of the individual LEDs are interconnected. If a cathode resistor is connected to a microcontroller pin, the LED can be switched on and off. Lo at the controlling pin switches the LED on, and Hi switches it off.

Table 4 shows the segment control for characters 0 to 9 and A to F as is required for displaying hexadecimal numbers.

Character	Segments							Output
	g	f	e	d	c	b	a	
0	1	0	0	0	0	0	0	&H40
1	1	1	1	1	0	0	1	&H79
2	0	1	0	0	1	0	0	&H24
3	0	1	1	0	0	0	0	&H30
4	0	0	1	1	0	0	1	&H19
5	0	0	1	0	0	1	0	&H12
6	0	0	0	0	0	1	0	&H02
7	1	1	1	1	0	0	0	&H78
8	0	0	0	0	0	0	0	&H00
9	0	0	1	0	0	0	0	&H10
A	0	0	0	1	0	0	0	&H08
B	0	0	0	0	0	1	1	&H03
C	1	0	0	0	1	1	0	&H46
D	0	1	0	0	0	0	1	&H21
E	0	0	0	0	1	1	0	&H06
F	0	0	0	1	1	1	0	&H0E

Table 4 Segment Control

Chapter 0 lists the complete character set of a seven-segment display. This table permits a lot more characters to be defined. No more than seven bits of the control byte are required for the complete character set. The MSB can control the decimal point.

Listing 24 shows a program example that displays the characters 0 to F on a seven-segment display continuously.

```

Config Porta = Output          ' PortA Output
Porta = 255                    ' all segments off

Dim I As Byte

Dim X(16) As Byte              ' Array for controlling bit patterns

Restore Value_table

For I = 1 To 16                 ' Read data in array
    Read X(i)
Next

Do
    For I = 1 To 16
        Porta = X(i)           ' Display character
        Waitms 250             ' Wait .5 seconds
        Waitms 250
    Next

```

---

```
Loop
```

```
End
```

```
Value_table:
```

```
Data &H40 , &H79 , &H24 , &H30 , &H19 , &H12 , &H02 , &H78
```

```
Data &H00 , &H10 , &H08 , &H03 , &H46 , &H21 , &H06 , &H0E
```

#### Listing 24 Control of Seven-Segment-Display by AVR (7SEGMENT.BAS)

The controlling bit patterns (see Table 4) are stored in the ROM in a table named `Value_table`.

Upon program start, the ROM table is copied to array X which makes access to the array (indexed variable) quite simple.

Characters 0 to F are displayed in an endless loop. The two instructions `waitms 250` generate a waiting time of half a second. Two wait instructions are needed because the argument has byte format and is limited to 255!

With the exception of port I/O, program 7SEGMENT.BAS has no AVR specific instructions. To port this program to 8051, all that is required is to adapt the I/O related instructions. Listing 25 shows the modified program for 8051 microcontrollers. The modifications are marked in bold characters.

```
' Seven-segment control by AT89C2051
$sim                               ' comment for normal operation

P1 = 255

Dim I As Byte
Dim X(16) As Byte

Restore Value_table

For I = 1 To 16
  Read X(i)
Next

Do
  For I = 1 To 16
    P1 = Not X(i)           ' inverted for simulation only
    Waitms 250
    Waitms 250
  Next
Loop
```

---

End

Value\_table:

Data &H40 , &H79 , &H24 , &H30 , &H19 , &H12 , &H02 , &H78

Data &H00 , &H10 , &H08 , &H03 , &H46 , &H21 , &H06 , &H0E

### Listing 25 Control of Seven-Segment-Display by 8051 (7SEGMENT.BAS)

In Listing 25 the output instruction was enhanced by the operator `not`. The reason is the BASCOM-8051 simulator which was used for program testing. Pressing the LCD button causes a display window to appear that contains a seven-segment display, too. Figure 56 shows the open window. As the segments of this display are switched on at Hi, the polarity had to be changed.

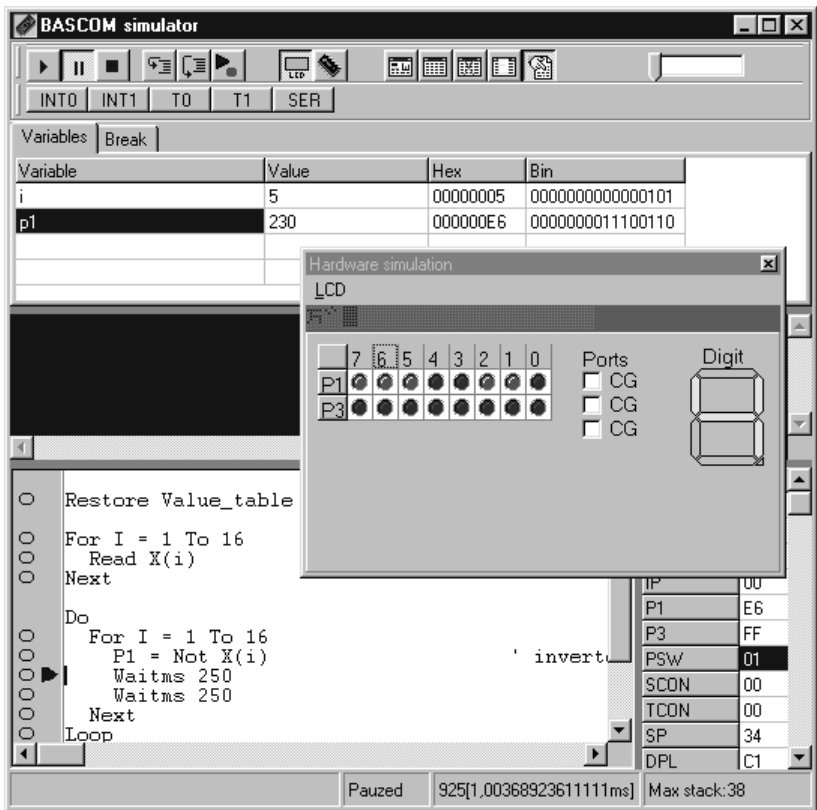


Figure 56 Seven-Segment Display in BASCOM-8051 Simulator

BASCOM-8051 offers the flexibility to assign the segments to any available pin. Right-click the seven-segment display to edit the properties of this display. Figure 57 shows how to edit the digit properties.

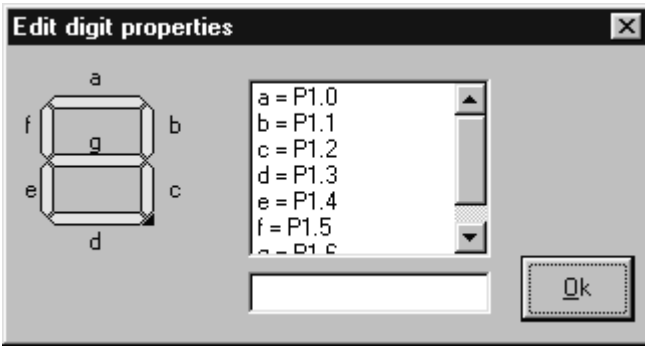


Figure 57 Pin Assignment

### 4.3.3 Dot-Matrix Displays

In most cases, a dot-matrix display uses a 5 x 7 LED matrix for display purposes. As is common with LCDs, a lot more characters can be displayed.

As an example, Figure 58 shows a dot-matrix TA07-11 made by Kingbright.

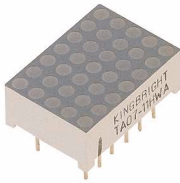


Figure 58 Dot-Matrix Display TA07-11

To control such a dot-matrix display, the assignment of these 35 LEDs to the pins of the display must be known. Figure 59 shows the internal circuit diagram of the TA07-11.

Column connections C1 to C5 link up the anodes of all LEDs in a certain column. Row connections R1 to R7 do the same for all LEDs in a certain row.

To switch a LED in the first column and third row, for example, line C1 must be connected to  $V_{CC}$  and line R3 via a series resistor to GND.