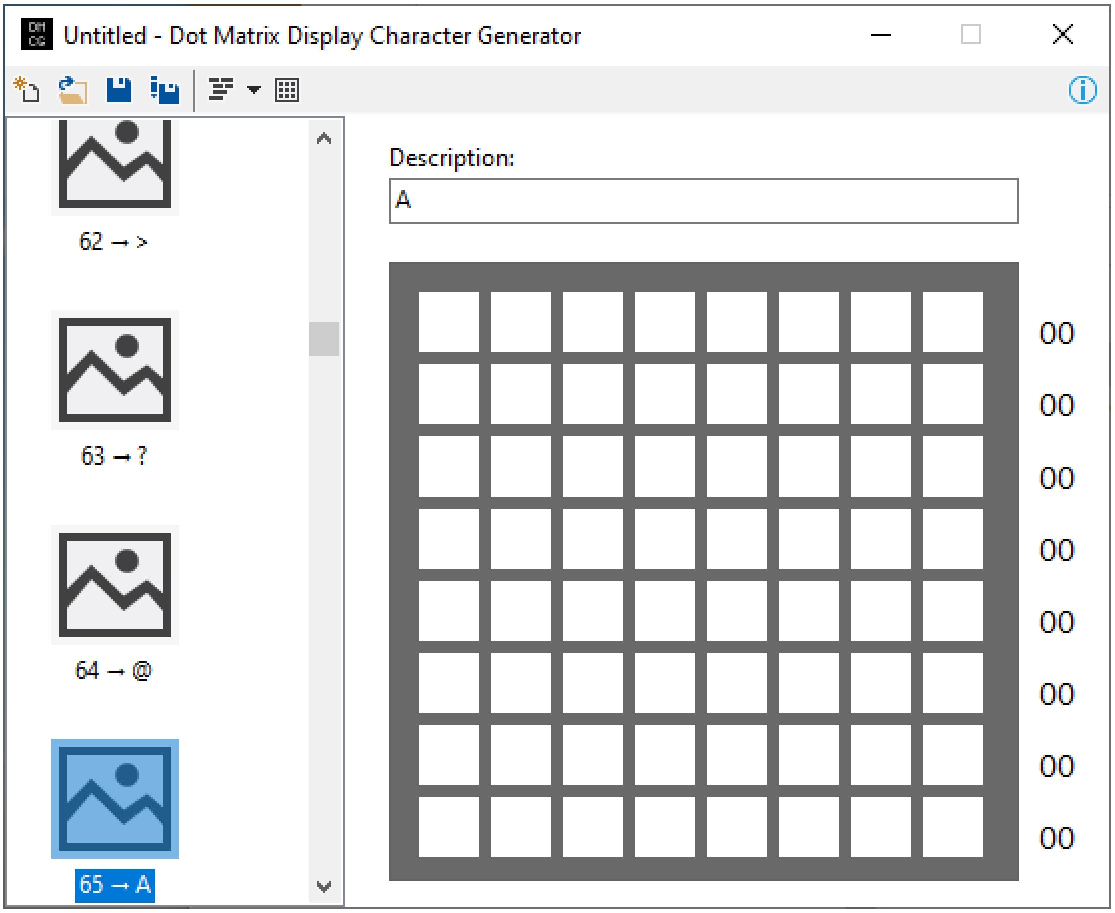
# Dot Matrix Character Generator

## Overview

This program assists in the creation of character set fonts for use on 8x8 LED dot-matrix displays and provides the code for any microprocessor that can be programmed in C.

The window contains a list on the left that contains thumbnails of the characters created, and the character matrix on the right. The list contains all character codes from 0 to 254. If a character pattern contains no data, a default thumbnail is used.



Untitled - Dot Matrix Display Character Generator

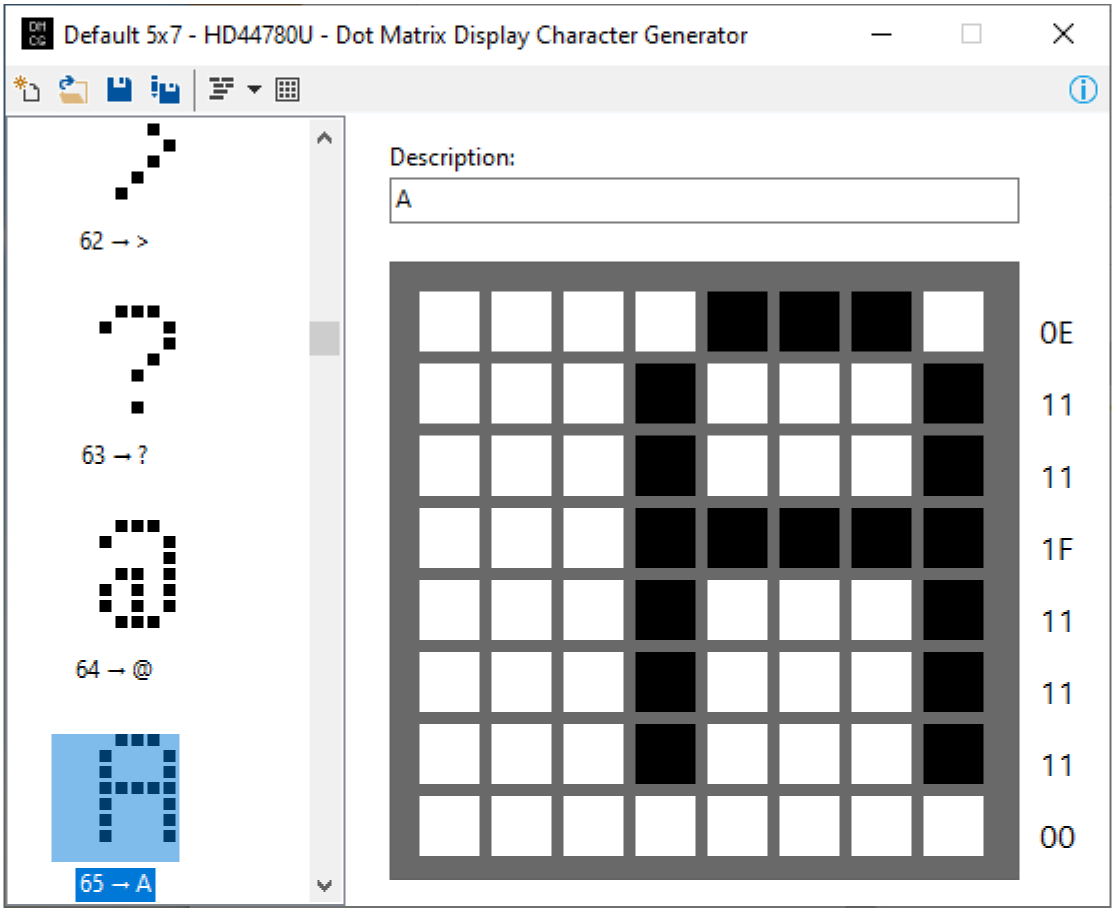
Description: A

```
00
00
00
00
00
00
00
00
```

## Creating character patterns

To create a character pattern for a code point, select the character code point in the list on the left side. Click on the squares in the matrix to toggle the dot on or off. The byte code for each row is displayed to the right of the matrix.

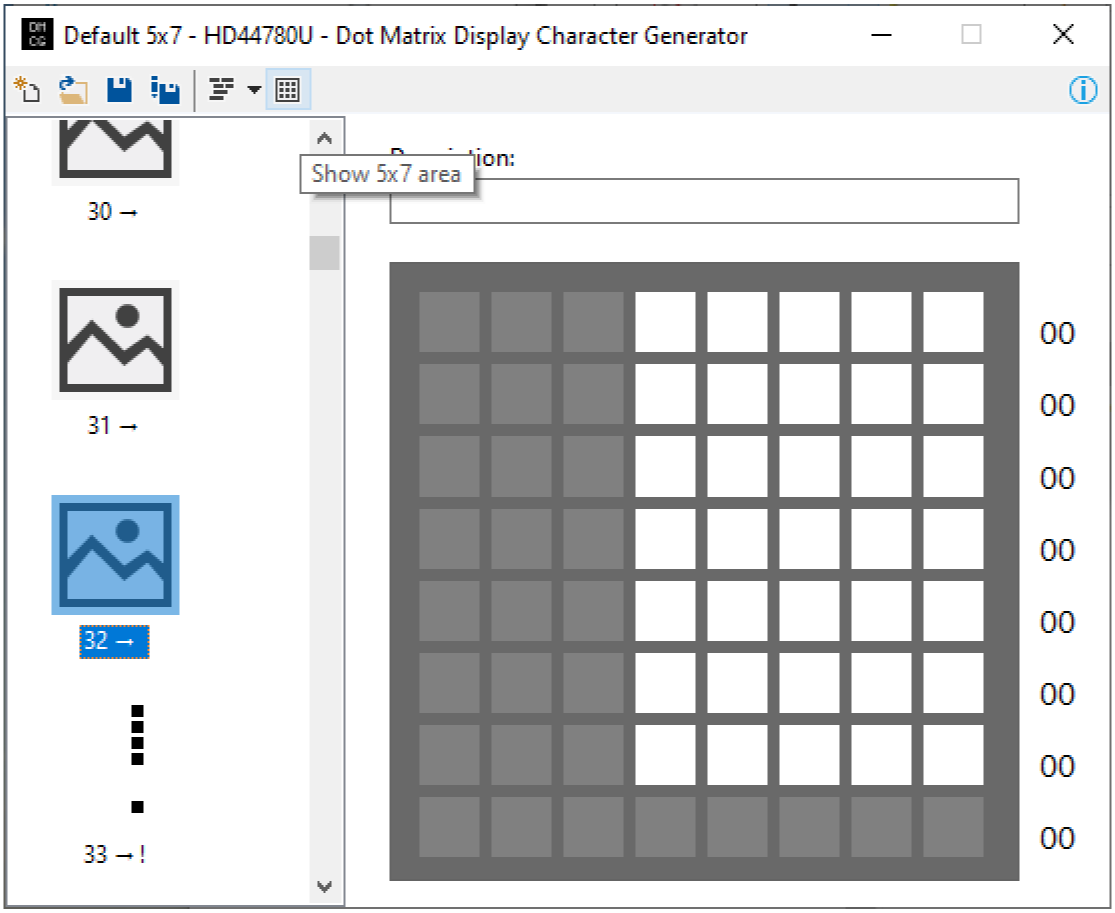
A description of the code point can be edited with the text box above the matrix.



Default 5x7 - HD44780U - Dot Matrix Display Character Generator

Description: A

If a 5x7 character matrix is needed, click on the **Show 5x7 area** button to mask out this area.

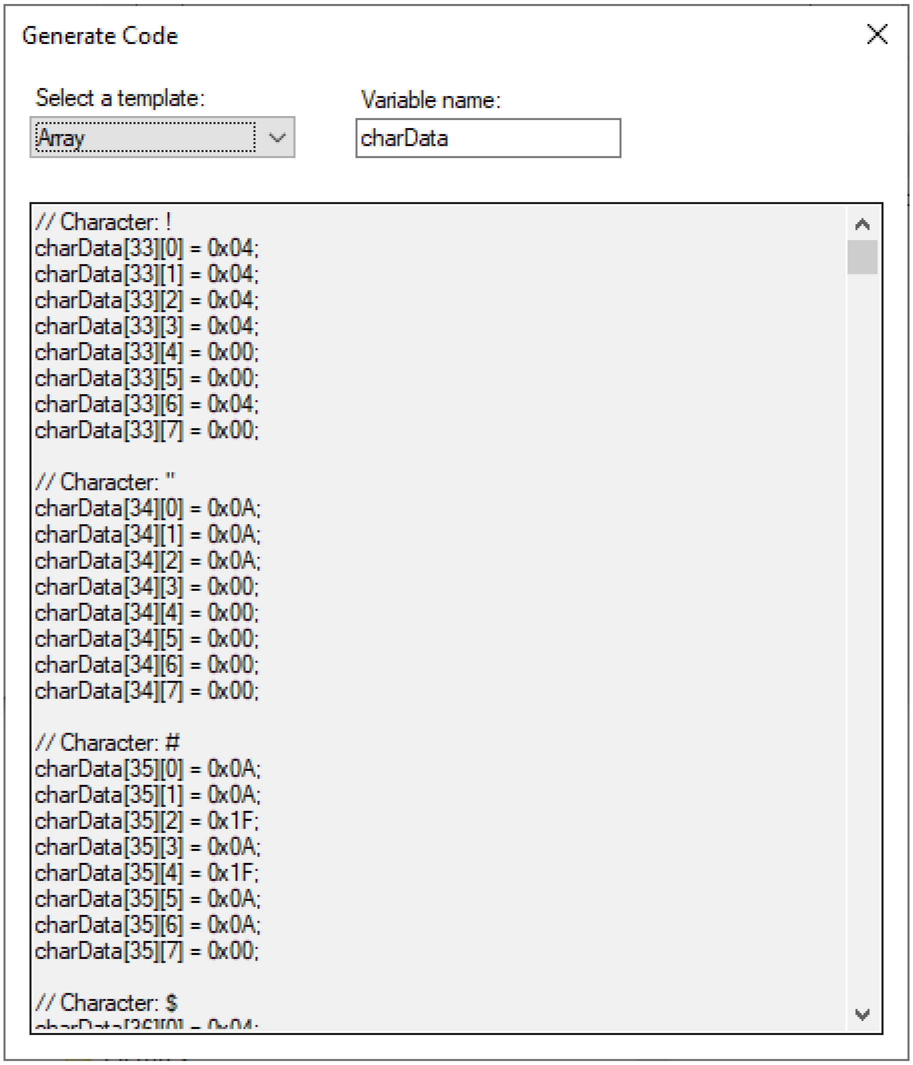

Default 5x7 - HD44780U - Dot Matrix Display Character Generator
Show 5x7 area
ion:
30 →
31 →
32 →
33 → !
00
00
00
00
00
00
00
00

## Generating Code

To generate the C code for the character(s) that have been defined, click on the Generate Code button:

To generate the code for the selected item only, click on the drop-down arrow, and select Selected Item from the menu.

A dialog opens that displays the generated code. The code can be implemented as an array or as a function. Select the template from the drop-down:

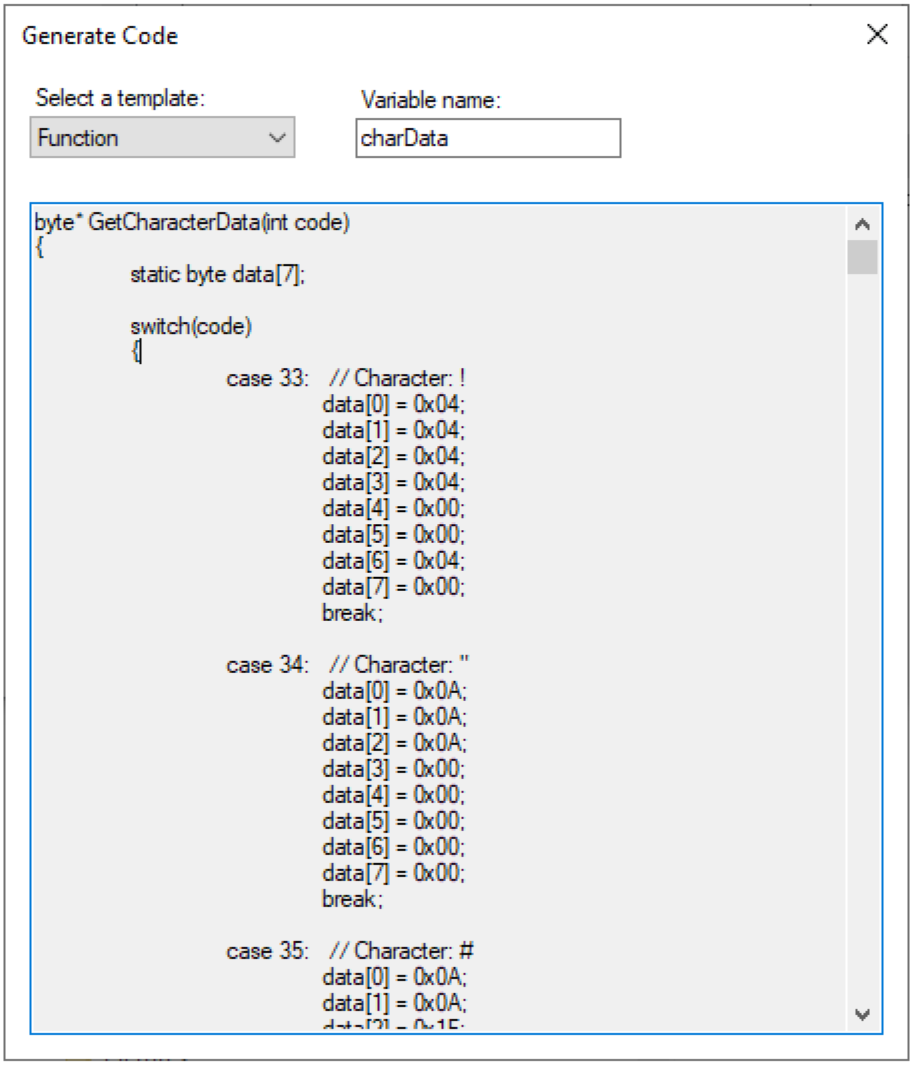


```
// Character: !
charData[33][0] = 0x04;
charData[33][1] = 0x04;
charData[33][2] = 0x04;
charData[33][3] = 0x04;
charData[33][4] = 0x00;
charData[33][5] = 0x00;
charData[33][6] = 0x04;
charData[33][7] = 0x00;

// Character: "
charData[34][0] = 0x0A;
charData[34][1] = 0x0A;
charData[34][2] = 0x0A;
charData[34][3] = 0x00;
charData[34][4] = 0x00;
charData[34][5] = 0x00;
charData[34][6] = 0x00;
charData[34][7] = 0x00;

// Character: #
charData[35][0] = 0x0A;
charData[35][1] = 0x0A;
charData[35][2] = 0x1F;
charData[35][3] = 0x0A;
charData[35][4] = 0x1F;
charData[35][5] = 0x0A;
charData[35][6] = 0x0A;
charData[35][7] = 0x00;

// Character: $
charData[36][0] = 0x04;
```



Select a template: Function

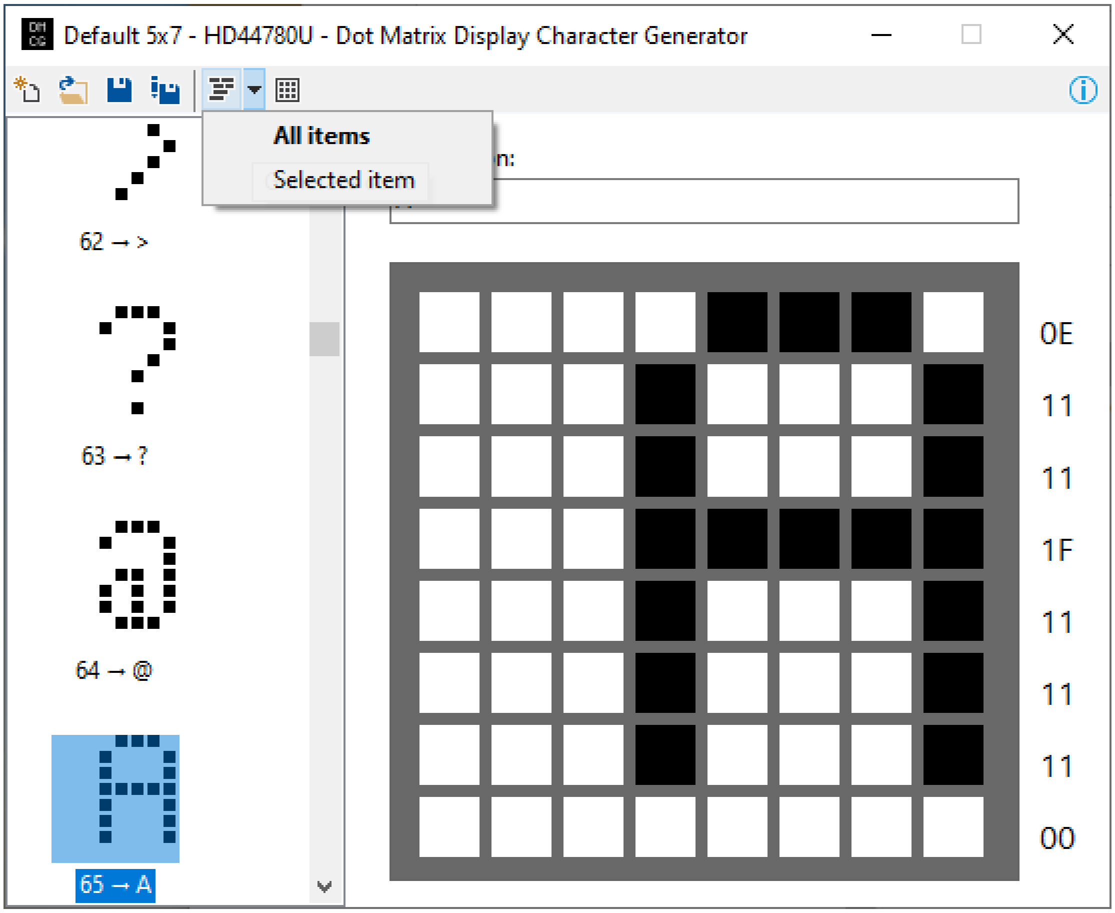
Variable name: charData

```
byte* GetCharacterData(int code)
{
        static byte data[7];

        switch(code)
        {
                case 33:  // Character: !
                        data[0] = 0x04;
                        data[1] = 0x04;
                        data[2] = 0x04;
                        data[3] = 0x04;
                        data[4] = 0x00;
                        data[5] = 0x00;
                        data[6] = 0x04;
                        data[7] = 0x00;
                        break;

                case 34:  // Character: "
                        data[0] = 0x0A;
                        data[1] = 0x0A;
                        data[2] = 0x0A;
                        data[3] = 0x00;
                        data[4] = 0x00;
                        data[5] = 0x00;
                        data[6] = 0x00;
                        data[7] = 0x00;
                        break;

                case 35:  // Character: #
                        data[0] = 0x0A;
                        data[1] = 0x0A;
                        data[2] = 0x1F;
```


Default 5x7 - HD44780U - Dot Matrix Display Character Generator
All items
Selected item
n:
62 → >
63 → ?
64 → @
65 → A

```
0E
11
11
1F
11
11
11
00
```