



**4D Systems**

---

## **Application Note: 4D-AN-1005**

# **Importing & Customizing a Windows Font into a 4DGL Application using FONT Tool**

Document Date: 1<sup>st</sup> November 2011

Document Revision: 1.0

---

## Description

This application note is dedicated to illustrating how to use Font Tool to convert Windows Fonts into the Bitmap format required by 4D displays. To exercise this application, the following items are required;

- Any 4D GFX Screen Module
- 4D Programming Cable
- FONT Tool Software Tool

---

## Application Overview

When programming any one of the 4D GFX modules with a 4DGL application program, the user can customize the Font that is displayed on the screen. This can be one of the default Windows fonts or a customized font if special characters are required. It is necessary to use the FONT Tool Software Tool to develop these fonts and create the file to be imported into the application.



---

## Setup Procedure

Firstly, you will need to download FONT Tool, which can be found from the 4D Systems website below:

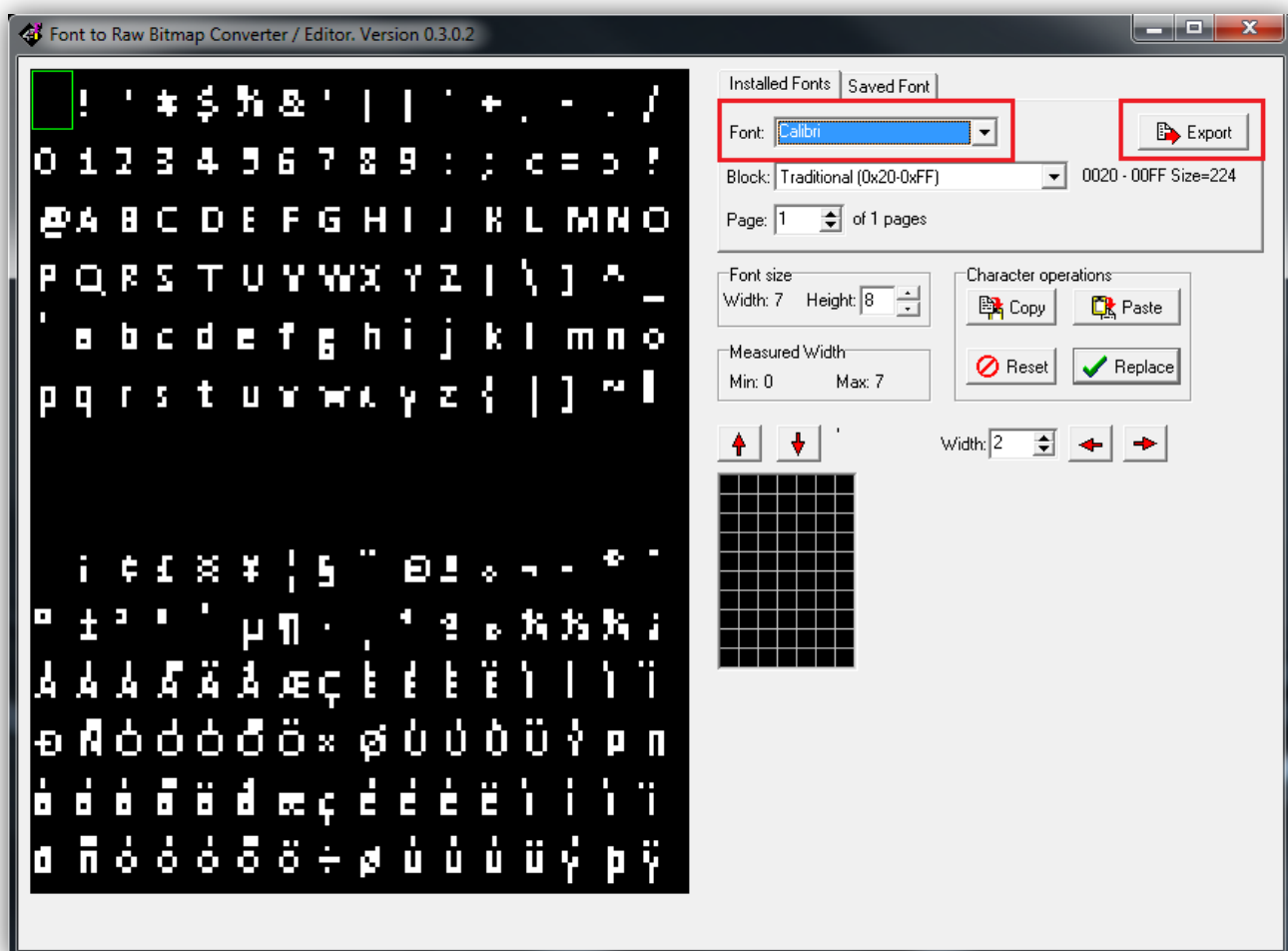
<http://www.4dsystems.com.au/prod.php?id=80>

## Simulation Procedure

### Select a Font Set

Open the FONT Tool and start by selecting the desired font Set by clicking on the dropdown box located in the middle top area of the screen. All the characters within that font set will be displayed, which can then be individually modified. For a detailed description on how to customize a font set, a step by step guide can be found at the link below:

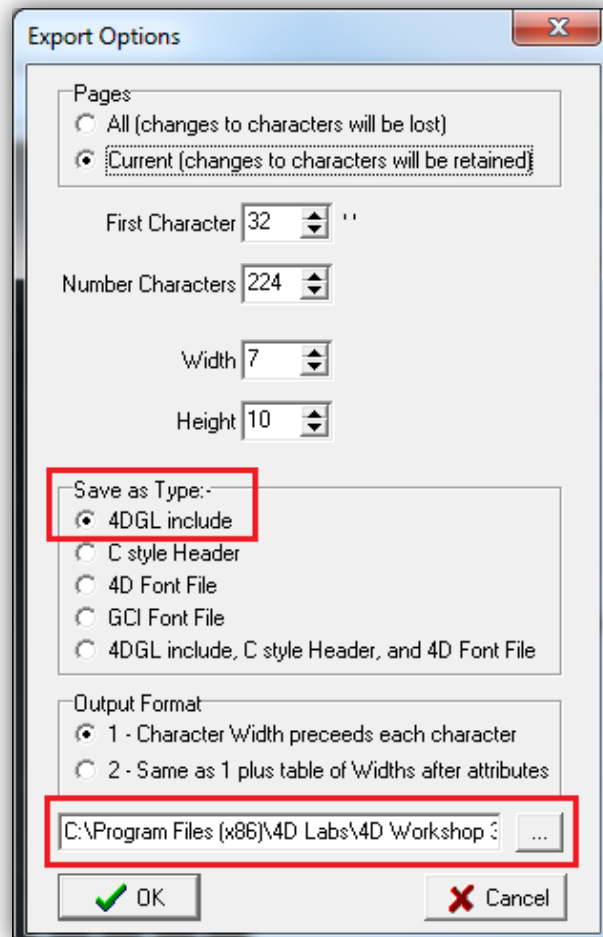
[http://www.4dsystems.com.au/downloads/Software/Font-Tool/User\\_Manual\\_Font\\_Tool.pdf](http://www.4dsystems.com.au/downloads/Software/Font-Tool/User_Manual_Font_Tool.pdf)



Click the **Export** button when you have finished editing, or are happy with a default Windows font set.

## Export Options

An Export Options window will appear with various configurable settings. Select **4DGL include** as the save type. It must be saved in the following directory: **\4D Labs\4D Workshop 3 IDE\INCLUDE**. The file name chosen will be the **Font ID** used in the 4DGL application. Click **OK**, and an .inc file will be created.



## Importing the Font Set

In order to use the desired Font Set, it needs to be inherited at the beginning of the application using the following command;

```
#inherit "newFont.inc"
```

## Example Application

An example snapshot is shown of using the imported Font Set once it is inherited.

```
txt_FontID(newFont);
```

## Sample Code

Below is an extract that illustrates the use of the imported font set. This sample code can be copied directly into a new 4DGL application to see the results of the new fonts.

```
#platform "uOLED-32028-P1_GFX2"

/*****
 * Filename: ImportFont.4dg
 * Created: 3rd November 2011
 * Author: 4D team
 * Description: Importing a Custom Font Set
 *****/

#inherit "4DGL_16bitColours.fnc"

#inherit "newFont.inc"    // 'newFont.inc' is the file name created from
the Export Options

func main()

    var n;
    n:=0;
    gfx_Cls();
    txt_FontID(newFont);
    print([HEX] newFont, " newFont");
    pokeW(SYSTEM_TIMER_LO, 0);    // reset timer
    while (n++<16)
        print(n, " HELLO WORLD");
    wend
    print(peekW(SYSTEM_TIMER_LO), "Msec");
    pause(1000);

    repeat
    forever

endfunc
```

## Proprietary Information

The information contained in this document is the property of 4D Systems Pty. Ltd. and may be the subject of patents pending or granted, and must not be copied or disclosed without prior written permission.

4D Systems endeavours to ensure that the information in this document is correct and fairly stated but does not accept liability for any error or omission. The development of 4D Systems products and services is continuous and published information may not be up to date. It is important to check the current position with 4D Systems.

All trademarks belong to their respective owners and are recognised and acknowledged.

---

## Disclaimer of Warranties & Limitation of Liability

4D Systems makes no warranty, either expresses or implied with respect to any product, and specifically disclaims all other warranties, including, without limitation, warranties for merchantability, non-infringement and fitness for any particular purpose.

Information contained in this publication regarding device applications and the like is provided only for your convenience and may be superseded by updates. It is your responsibility to ensure that your application meets with your specifications.

In no event shall 4D Systems be liable to the buyer or to any third party for any indirect, incidental, special, consequential, punitive or exemplary damages (including without limitation lost profits, lost savings, or loss of business opportunity) arising out of or relating to any product or service provided or to be provided by 4D Systems, or the use or inability to use the same, even if 4D Systems has been advised of the possibility of such damages.

4D Systems products are not fault tolerant nor designed, manufactured or intended for use or resale as on line control equipment in hazardous environments requiring fail – safe performance, such as in the operation of nuclear facilities, aircraft navigation or communication systems, air traffic control, direct life support machines or weapons systems in which the failure of the product could lead directly to death, personal injury or severe physical or environmental damage ('High Risk Activities'). 4D Systems and its suppliers specifically disclaim any expressed or implied warranty of fitness for High Risk Activities.

Use of 4D Systems' products and devices in 'High Risk Activities' and in any other application is entirely at the buyer's risk, and the buyer agrees to defend, indemnify and hold harmless 4D Systems from any and all damages, claims, suits, or expenses resulting from such use. No licenses are conveyed, implicitly or otherwise, under any 4D Systems intellectual property rights.