**Secret Status：Top Secret ( √ ) Secret** **( √ ) Restrict ( √ ) Public ( √ )**

**RKNanoD Font Format**

**User Manual**

**(Version 1.0)**

**RKNano Team**

**Rockchip Electronics Co.，Ltd**

**www.rock-chips.com**

|  |  |  |  |
| --- | --- | --- | --- |
| **Status：**  **[√] development**  **[√]** **public**  **[√] modify** | **File Tag：** | **Font Format User Manual** | |
| **Version：** | **1.0** | |
| **Author：** | **Ma Longchang** | |
| **Data：** | **2016年6月6日** | |
| **Verify:** |  |  |

Version History

|  |  |  |  |
| --- | --- | --- | --- |
| **Version** | **Author** | **Date** | **Remarks** |
| **1.0** | **Ma Longchang** | **2016-06-6** | **Original** |
|  |  |  |  |
|  |  |  |  |

目录

[1 简介 4](#_Toc453056189)

[2 字库文件结构说明 5](#_Toc453056190)

[2.1 Font12.bin 5](#_Toc453056191)

[2.1.1 文件结构 5](#_Toc453056192)

[2.1.2 字形结构 6](#_Toc453056193)

[2.2 Font16.bin文件结构 8](#_Toc453056194)

[2.2.1 文件结构 8](#_Toc453056195)

[2.2.2 字形结构 9](#_Toc453056196)

[3 用户自定义字库 11](#_Toc453056197)

# 简介

RKNanoD MP SDK目前支12×12,16×16 点阵大小的UNICODE字符的显示。本文档将对字库文件中的组成结构，字符在字库文件中的数据组成结构做以详细说明，并结合SDK中的显示代码作为辅助说明，以便用户开发过程中的使用。

# 字库文件结构说明

对EMMC flash的SDK而已，字库文件目录放在如下路径中：..\SDK\_160\_128\Development\firmware\_generate\_eMMC。

字库文件分别命名为:

12×12 dot:Font12.bin。

16×16 dot:Font16.bin。

### Font12.bin

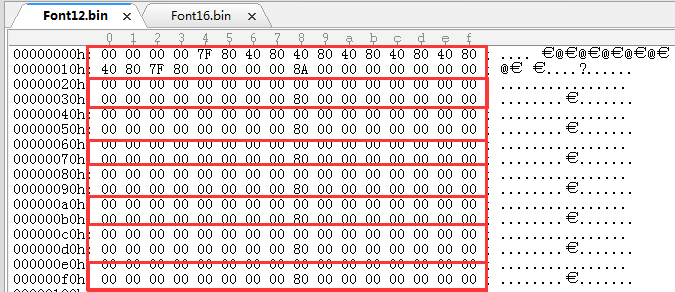
### 文件结构

在该文件中包含了所有Unicode字符的字形数据，包括可显示与不可显示的，即从0x0000

~0xFFFF，每个Unicode字符的字形数据占用32个bytes，其中有效数据为26个bytes，其他

剩余6个字节为0，为无效字节，起补位作用，方便在文件中定位。

使用UltraEdit或WinHex二进制工具打开该字库文件，可以查看该文件的组成结构：如下图所示：



Font12.bin文件结构

其中26个bytes有效数据中前24个字节为Unicode字符的字形数据，第25,26个字节为字符的类型，最重要的是第25个字节，该类型包含了字符的字形种类及宽度信息，第26个字节默认为0。

具体描述如下表：

|  |  |
| --- | --- |
| iType（第25个bytes） | 备注 （单位：pixels） |
| 0x10 | 仅当语言为俄语时，字符宽度为 8 |
| 0x20 | 字符宽度为12 |
| 0x80 | 字符Unicode大于0x7F时，字符宽度为12 |
| 0x80 – 0xFF | 字符宽度为（iType – 0x80） |

iType类型含义

代码详见**LcdChar.c**中**LCD\_NFDispUnicodeChar**接口。

### 字形结构

每个Unicode字符的字形数据是按照从左→右，从上↓下的扫描顺序获取出来保存的， 每

行12个bit（每个pixel占一个bit），用2个字节表示，有12行，共24个字节的数据。

→ → → → → → → →

→ → → → → → → →

……

→ → → → → → → →

→ → → → → → → →

以大写的字符“W”举例说明：

“W”字符在font12.bin中的位置：0x57×0x20（0x57 << 5）= 0xAE0

其字形数据为：从左到右，从上到下。

十六进制表示 二进制表示

1. 00： 0 0 0 0 0 0 0 0 0 0 0 0
2. 00： 0 0 0 0 0 0 0 0 0 0 0 0

88 80： 1 0 0 0 1 0 0 0 1 0 0 0

88 80： 1 0 0 0 1 0 0 0 1 0 0 0

88 80： 1 0 0 0 1 0 0 0 1 0 0 0

55 00： 0 1 0 1 0 1 0 1 0 0 0 0

55 00： 0 1 0 1 0 1 0 1 0 0 0 0

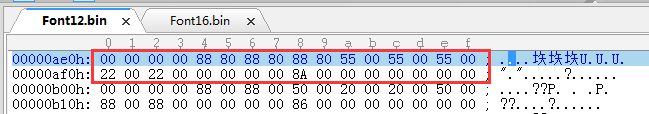
55 00： 0 1 0 1 0 1 0 1 0 0 0 0

22 00： 0 0 1 0 0 0 1 0 0 0 0 0

22 00： 0 0 1 0 0 0 1 0 0 0 0 0

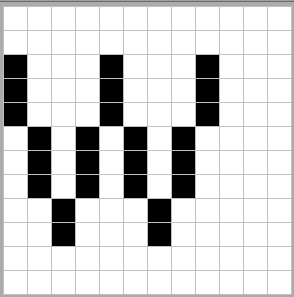
1. 00： 0 0 0 0 0 0 0 0 0 0 0 0

00 00： 0 0 0 0 0 0 0 0 0 0 0 0



“W”在Font12.bin中的位置与内容

“W”的显示效果图：



“W”字符显示，宽度为10个pixel（0x8A – 0x80）

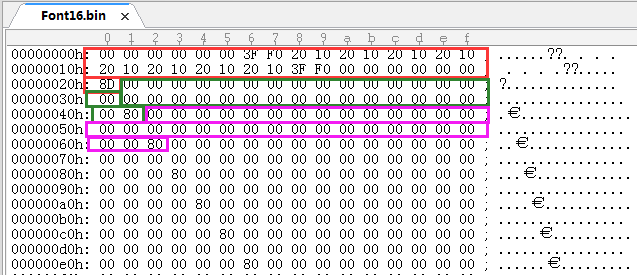
### Font16.bin文件结构

### 文件结构

在该文件中包含了所有Unicode字符的字形数据，包括可显示与不可显示的，即从

0x0000~0xFFFF，每个Unicode字符的字形数据占用33个bytes，其中有效数据为32个bytes，其他剩余1个字节为iType，包含字符类型及宽度信息。

使用UltraEdit或WinHex二进制工具打开该字库文件，可以查看该文件的组成结构：如下图所示：



Font16.bin文件结构

如上图所示前32个字节为Unicode字符的字形数据，最后1个字节为字符的类型，该类型包含了字符的字形种类及宽度信息。

具体描述如下表：

|  |  |
| --- | --- |
| iType（第33个字节） | 备注 （单位：pixels） |
| 0x10 | 仅当语言为俄语时，字符宽度为 8 |
| 0x20 | 字符宽度为12 |
| 0x80 | 字符Unicode大于0x7F时，字符宽度为12 |
| 0x80 – 0xFF | 字符宽度为（iType – 0x80） |

iType类型含义

代码详见**LcdChar.c**中**LCD\_NFDispUnicodeChar**接口。

### 字形结构

同2.1.2节中描述的一样，每个Unicode字符的字形数据是按照从左→右，从上↓下的扫描顺序获取出来保存的， 每行16个bit（每个pixel占一个bit），用2个字节表示，有16行，共32个字节的数据。

→ → → → → → → →

→ → → → → → → →

……

→ → → → → → → →

→ → → → → → → →

以大写的字符“W”举例说明：“W”字符在font16.bin中的位置：0x57×33 = 0xB37

其字形数据为：从左到右，从上到下。

十六进制表示 二进制表示

00 00： 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0

00 00： 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0

00 00： 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0

00 00： 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0

84 20： 1 0 0 0 0 1 0 0 0 0 1 0 0 0 0 0

84 20： 1 0 0 0 0 1 0 0 0 0 1 0 0 0 0 0

8A 20： 1 0 0 0 1 0 1 0 0 0 1 0 0 0 0 0

8A 20： 1 0 0 0 1 0 1 0 0 0 1 0 0 0 0 0

4A 40： 0 1 0 0 1 0 1 0 0 1 0 0 0 0 0 0

51 40： 0 1 0 1 0 0 0 1 0 1 0 0 0 0 0 0

51 40： 0 1 0 1 0 0 0 1 0 1 0 0 0 0 0 0

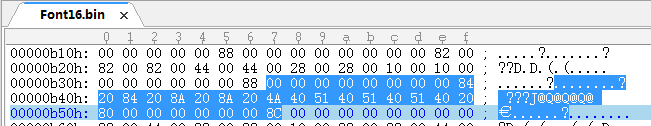
51 40： 0 1 0 1 0 0 0 1 0 1 0 0 0 0 0 0

20 80： 0 0 1 0 0 0 0 0 1 0 0 0 0 0 0 0

00 00： 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0

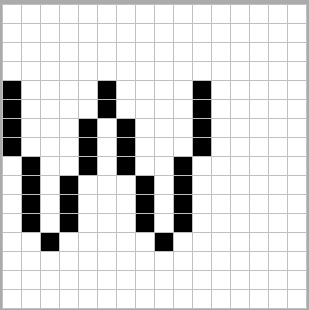
00 00： 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0

00 00： 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0



“W”在文件中的位置与内容

“W”的显示效果图：



“W”字符显示，宽度为12个pixel（0x8C – 0x80）

# 用户自定义字库

用户如需使用自定义的字库，需按照以上章节描述的组织适当的字库文件和所有Unicode字符字形数据，同时需要命名为Font12.bin或Font16.bin的文件名，否则无法顺便打包进固件文件。