



# **Tina Linux GPU 开发指南**

**版本号: 1.3  
发布日期: 2020-09-04**

## 版本历史

| 版本号 | 日期         | 制/修订人   | 内容描述                                   |
|-----|------------|---------|--|
| 1.0 | 2019.07.11 | AWA1359 | 初始版本。                                  |
| 1.1 | 2020.02.24 | AWA1359 | 添加 GE8300 型号 GPU。                      |
| 1.2 | 2020.05.12 | AWA1359 | 修改查看 GE8300 频率命令。                      |
| 1.3 | 2020.06.04 | AWA1359 | 添加 MR813 和 R818 平台, 修改查看 GE8300 使用率路径。 |



# 目 录

|                      |           |
|----------------------|-----------|
| <b>1 概述</b>          | <b>1</b>  |
| 1.1 编写目的             | 1         |
| 1.2 适用范围             | 1         |
| 1.3 相关人员             | 1         |
| <b>2 平台 GPU 支持情况</b> | <b>2</b>  |
| <b>3 图形框架</b>        | <b>3</b>  |
| <b>4 GPU 配置</b>      | <b>4</b>  |
| 4.1 GPU+fbdev 配置     | 4         |
| 4.1.1 GPU 库以及驱动配置    | 4         |
| 4.1.2 fbdev 配置       | 4         |
| 4.2 GPU+wayland 配置:  | 6         |
| 4.2.1 GPU 库以及驱动配置    | 6         |
| 4.2.2 wayland 配置     | 7         |
| 4.2.3 配置 DRM         | 9         |
| 4.3 GPU demo 配置      | 10        |
| <b>5 GPU 目录结构</b>    | <b>12</b> |
| 5.1 添加新平台适配 gpu      | 12        |
| <b>6 GPU 性能测试</b>    | <b>13</b> |
| 6.1 测试工具 glmark2     | 13        |
| 6.2 glmark2 配置       | 13        |
| 6.3 测试方法             | 13        |
| 6.4 查看 GPU 使用率       | 16        |
| 6.4.1 mali400        | 16        |
| 6.4.2 mali-t760      | 16        |
| 6.4.3 sgx544         | 16        |
| 6.4.4 ge8300         | 16        |
| 6.5 查看 GPU 频率        | 17        |
| 6.5.1 mali400        | 17        |
| 6.5.2 mali-t760      | 18        |
| 6.5.3 sgx544         | 18        |
| 6.5.4 ge8300         | 18        |
| 6.6 查看 GPU 电压        | 18        |
| 6.7 查看 GPU 温度        | 18        |
| 6.8 glmark2 参数分析     | 19        |
| <b>7 GPU 问题解决</b>    | <b>21</b> |

## 插 图

|   |    |
|---|----|
| 3-1 支持 gpu 框架 . . . . .                         | 3  |
| 4-1 Console display driver support 选项 . . . . . | 5  |
| 4-2 Video support for sunxi 选项 . . . . .        | 5  |
| 4-3 Kernel modules 配置 . . . . .                 | 7  |
| 4-4 Cairo 选项 . . . . .                          | 8  |
| 4-5 Wayland 选项 . . . . .                        | 8  |
| 4-6 Weston 选项 . . . . .                         | 9  |
| 4-7 DRM 选项 . . . . .                            | 10 |
| 5-1 添加新平台适配 gpu . . . . .                       | 12 |
| 6-1 glmark2 测试数据 . . . . .                      | 15 |
| 6-2 gpu 使用率 . . . . .                           | 16 |
| 6-3 status 节点内容 1 . . . . .                     | 17 |
| 6-4 status 节点内容 2 . . . . .                     | 17 |
| 7-1 gpu 错误 bad alloc . . . . .                  | 21 |



# 1 概述

---

本文档简介 Tina Linux 平台 GPU 不同型号的支持，GPU 如何使用，如何查看 GPU 性能，以及如何配置 & 添加一款 GPU。

## 1.1 编写目的

本文主要服务于使用 Tina 软件平台的广大客户，以冀帮助客户使用 Tina 平台的 GPU。

## 1.2 适用范围

Allwinner 软件平台 Tina。

## 1.3 相关人员

适用 Tina 平台的广大客户和关心 GPU 的相关人员。

## 2 平台 GPU 支持情况

目前 tina 中支持 4 种 gpu，分别是 MALI-400, MALI-T760, PowerVR SGX544,GE8300

| GPU 架构       | GPU 型号    | opengles 版本     | opencl 版本   | vulkan 版本 |
|--------------|-----------|-----------------|-------------|-----------|
| Mali Utgard  | Mali-400  | 1.1/2.0         | 不支持         | 不支持       |
| Mali Midgard | Mali-T760 | 1.1/2.0/3.0/3.1 | 1.1/1.2     | 1.0       |
| IMG SGX      | SGX544    | 1.1/2.0         | 1.1         | 不支持       |
| IMG RGX      | GE8300    | 1.1/2.0/3.0/3.1 | 1.0/1.1/1.2 | 1.0/1.1   |

| GPU 型号    | 适配硬件平台              | 对接的窗口类型                      |
|-----------|---------------------|------------------------------|
| Mali-400  | R40/R16/R18/R311/T7 | fbdev, 只有 R18 平台同时支持 wayland |
| Mali-T760 | R30                 | fbdev                        |
| SGX544    | R58                 | fbdev                        |
| GE8300    | MR813/R818          | fbdev                        |
| 没有 GPU    | R331/R11/R6/R7/R328 |                              |

### 3 图形框架

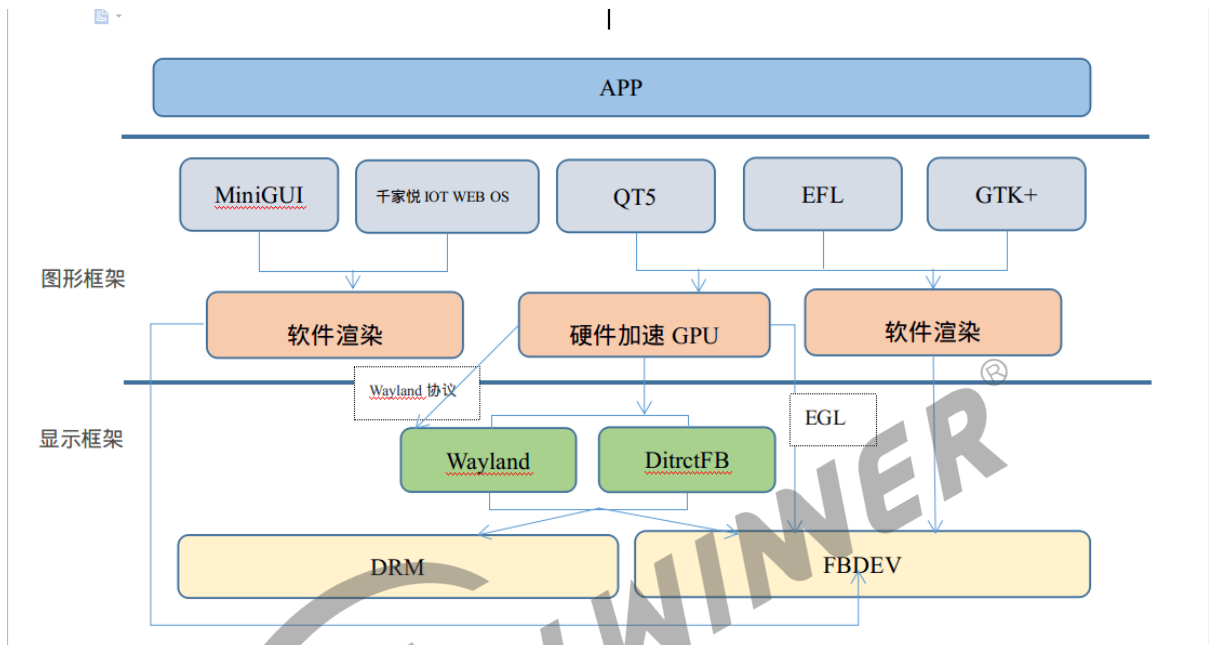


图 3-1: 支持 gpu 框架

从上图可以看出，目前 tina 中支持的图形框架，只有 qt5,efl gtk+ 支持 gpu 加速。显示框架 wayland,drm 只有在 r18 上面支持。

## 4 GPU 配置

### 4.1 GPU+fbdev 配置

#### 4.1.1 GPU 库以及驱动配置

```
make menuconfig
--<*>Libraries
  --<*>GPU Libraries
    -->*- mali400-um/mali-t760-um/sgx544-um/ge8300-um /*gpu的标准库*/
  -->Kernel modules
    -->Video Support
      -->*- kmod-mali-utgard-km. /*gpu驱动*/
      不同的gpu型号对应kmod-mali-midgard-km/kmod-sgx544-km/kmod-ge8300-km
```

MR813 方案比较特殊，如果此方案既使用 OpenGL ES 也使用 OpenCL，则按照上面的配置。如果此方案不需要使用 GPU 的 OpenGL ES 渲染功能，只需要使用 OpenCL 计算功能，因此针对此方案，做了特殊配置处理，其配置如下：

```
make menuconfig
--<*>Libraries
  --<*>GPU Libraries
    --<*> libgpu-openc1 /*gpu的标准库*/

  -->Kernel modules
    -->Video Support
      -->*- kmod-ge8300-km-openc1 /*gpu驱动*/
```

#### 4.1.2 fbdev 配置

选上 Framebuffer Console Support(sunxi)、DISP Driver Support(sunxi-disp2)、Framebuffer Console support 与 Transform Driver Support(sunxi)，默认 fbdev 是配置好的，这里只是了解，方便后面更换成 drm。

```
make kernel_menuconfig
Device Drivers
  -->Graphics support
```

```

-->Frame buffer Devices
  --><*> Support for frame buffer devices
    -->Video support for sunxi
      -->[*] Framebuffer Console Support(sunxi)
      --><*> DISP Driver Support(sunxi-disp2)
    -->Console display driver support
      --><*> Framebuffer Console support
-->Character devices
  --><*> Transform Driver Support(sunxi)

```

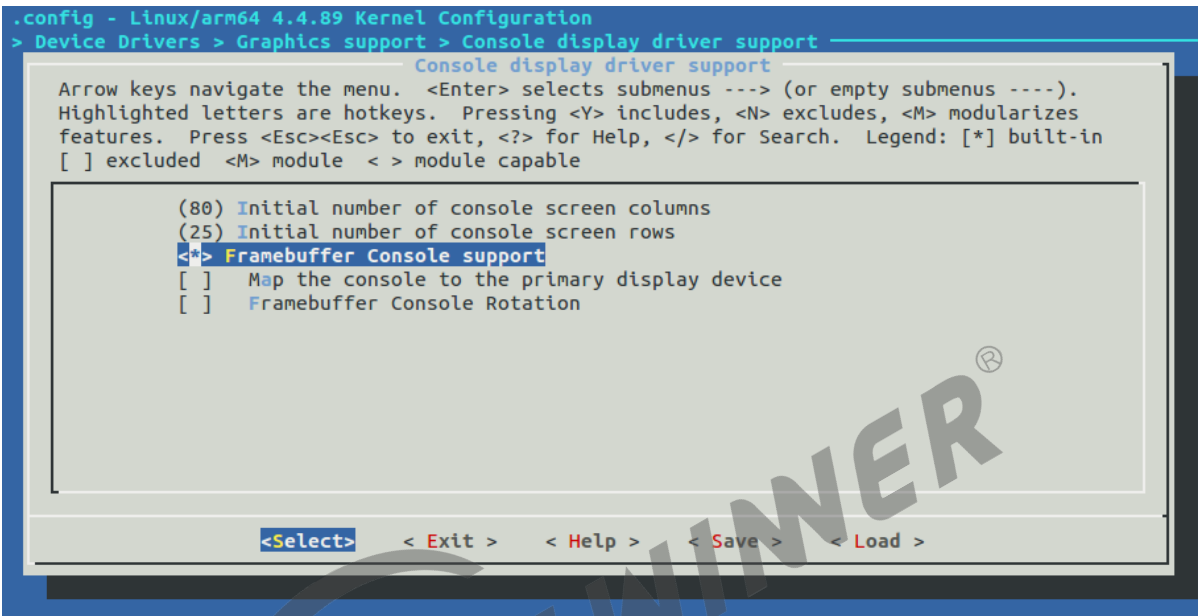


图 4-1: Console display driver support 选项

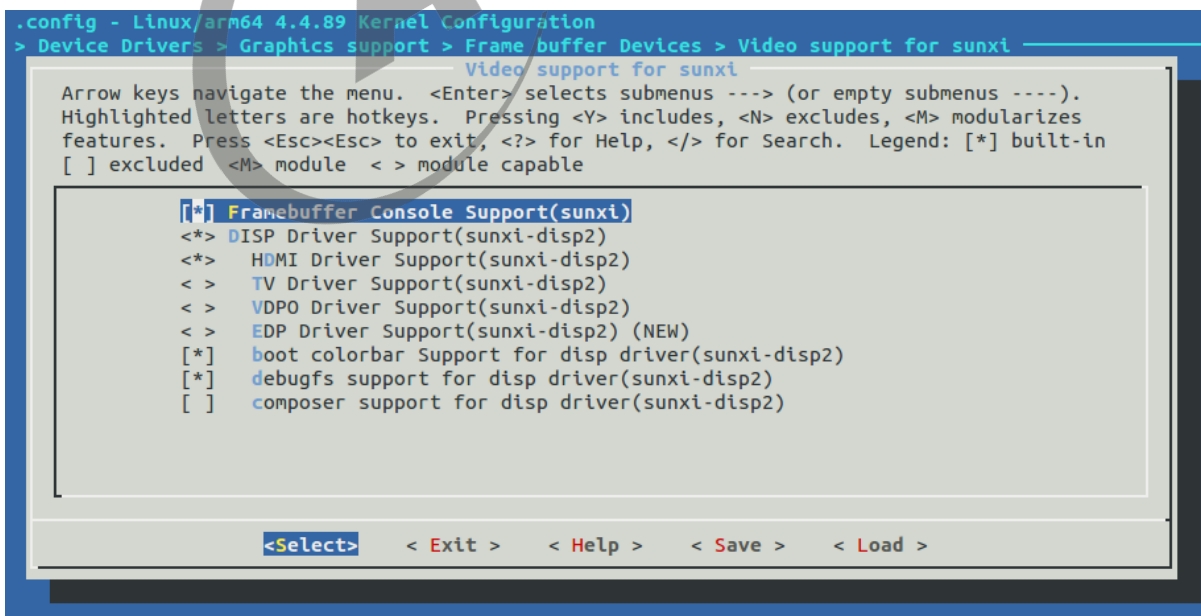


图 4-2: Video support for sunxi 选项

针对 GE8300 型号的 GPU,(目前使用此型号的 gpu 只有 mr813 和 r818 平台) 还需要配置上如

下的选项:

```
make kernel_menuconfig
Device Drivers
  --> Graphics support
    -->GPU support for sunxi
      -->[*] GPU PM OPP
```

默认的 gpu 配置对接的显示框架是 fbdev, 如果要使用 wayland 显示框架, 需要进行下面配置 (注意: **wayland** 只在 **r18** 平台支持)。

## 4.2 GPU+wayland 配置:

以 R18 平台为例, 主要配置项如下: cairo 是 wayland 依赖的库,

### 4.2.1 GPU 库以及驱动配置

```
make menuconfig
  --<*>Libraries
    --<*>GPU Libraries
      -->*- mali400-um/mali-t760-um/sgx544-um/ge8300-um /*gpu的标准库*/
    -->Kernel modules
Kernel modules
  -->Video Support
    --<*> kmod-mali-utgard-km /*gpu驱动*/
    --> > kmod-sunxi-disp /*gpu使用fbdev的驱动*/
    --<*> kmod-sunxi-drm /*配置DRM*/
```

针对 GE8300 型号的 GPU,(目前使用此型号的 gpu 只有 mr813 和 r818 平台) 还需要配置上如下的选项:

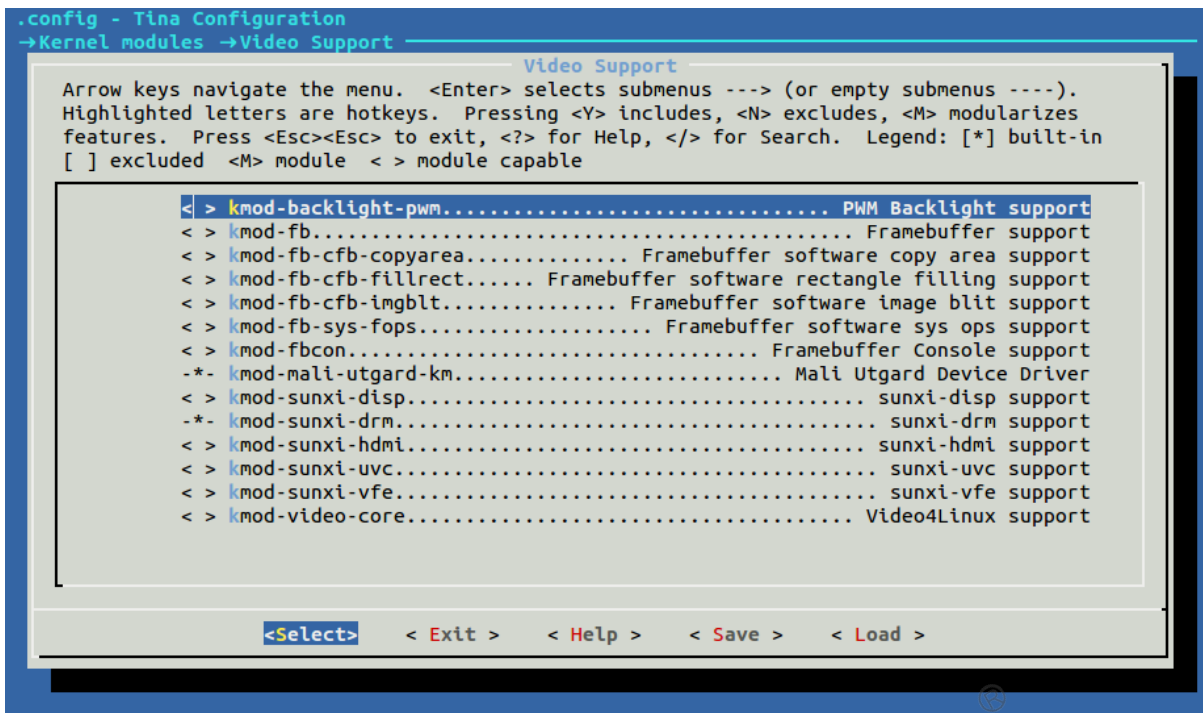


图 4-3: Kernel modules 配置

```
make kernel_menuconfig
Device Drivers
--> Graphics support
-->GPU support for sunxi
-->[*] GPU PM OPP
```

## 4.2.2 wayland 配置

```
Libraries
-->cairo
-->*- libcairo
-->[*] Enable cairo postscript support
-->[*] Enable cairo pdf support
-->[*] Enable cairo png support
-->[ ] Enable script support
-->[*] Enable cairo svg support
-->[ ] Enable cairo tee support
-->[ ] Enable cairo xml support
-*Wayland
-*wayland-protocols
<*>weston
-->[ ] Enabel dbus support
-->[ ] Enabel weston-launch linux pam support
-->[*] Enabel opengl es support
-->[ ] Enabel fbdev compositor support
```

```

-->[*] Enabel drm compositor support
-->[ ] Enabel lcms supports support
-->[ ] Enabel junit xml support
-->[ ] Enabel demo clients install

```

上面的配置项如下图所示:

```

.config - Tina Configuration
->Libraries ->cairo ->libcairo..... Cairo is a 2D graphics library
libcairo..... Cairo is a 2D graphics library
Arrow keys navigate the menu. <Enter> selects submenus ---> (or empty submenus ----).
Highlighted letters are hotkeys. Pressing <Y> includes, <N> excludes, <M> modularizes
features. Press <Esc><Esc> to exit, <?> for Help, </> for Search. Legend: [*] built-in
[ ] excluded <M> module < > module capable

-- libcairo..... Cairo is a 2D graphics library
[ ] Enable cairo postscript support
[*] Enable cairo pdf support
[*] Enable cairo png support
[ ] Enable script support
[ ] Enable cairo svg support
[ ] Enable cairo tee support
[ ] Enable cairo xml support

<Select> <Exit > <Help > <Save > <Load >

```

图 4-4: Cairo 选项

```

.config - Tina Configuration
->Wayland
Wayland
Arrow keys navigate the menu. <Enter> selects submenus ---> (or empty submenus ----).
Highlighted letters are hotkeys. Pressing <Y> includes, <N> excludes, <M> modularizes
features. Press <Esc><Esc> to exit, <?> for Help, </> for Search. Legend: [*] built-in
[ ] excluded <M> module < > module capable

<*> gmark2..... gmark2 is an OpenGL 2.0 and ES 2.0 benchmark
[ ] Enable fbdev support (NEW)
[*] Enable wayland support (NEW)
-* wayland..... wayland
-* wayland-protocols..... wayland-protocols
<*> weston... Weston is the reference implementation of a Wayland compositor --

<Select> <Exit > <Help > <Save > <Load >

```

图 4-5: Wayland 选项

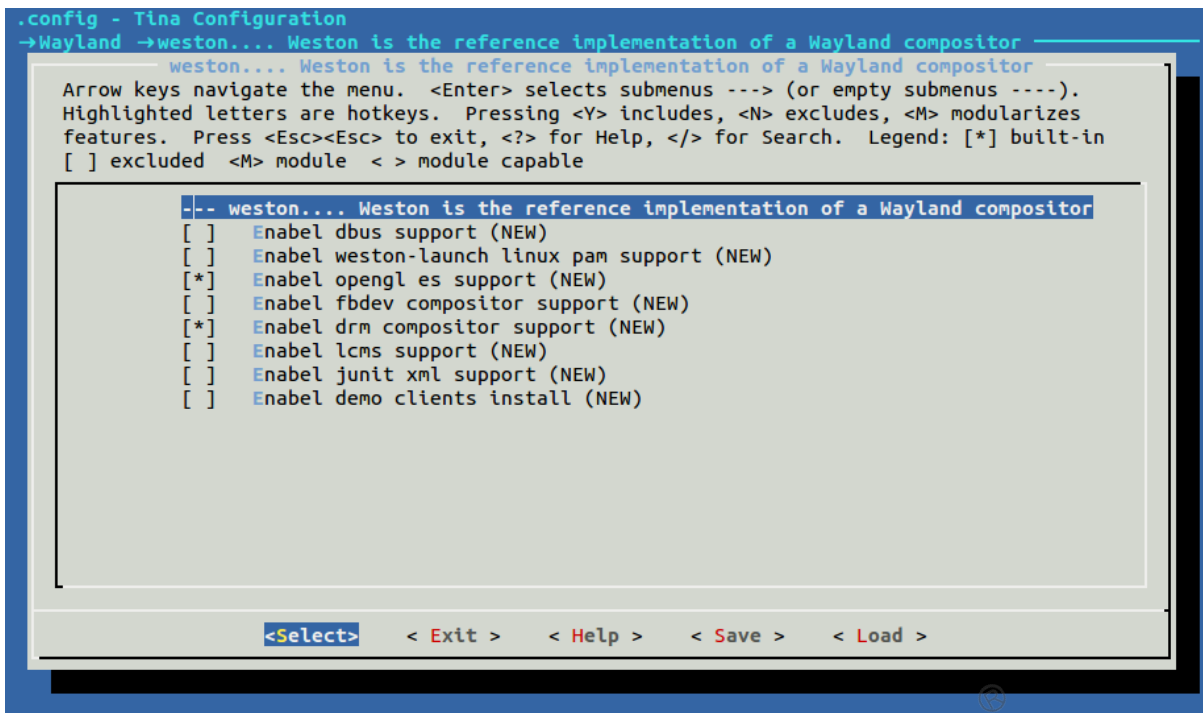


图 4-6: Weston 选项

### 4.2.3 配置 DRM

如果 menuconfig 选择的是使用 DRM 作为后端，由于内核中默认使用 FBDEV，所以先要取消原本的配置，再选择上 DRM 的配置，在 menuconfig 的配置中取消 kmod-sunxi-disp，选上 kmod-sunxi-drm，R18 平台会自动配置下面的选项，不用在执行这一小节的步骤，其他平台暂未实现自动配置。执行以下命令，以 R18 的为例。现阶段只有 R18 支持 DRM。

则取消选择 Framebuffer Console Support(sunxi)、DISP Driver Support(sunxi-disp2)、Framebuffer Console support 与 Transform Driver Support(sunxi)

```
make kernel_menuconfig
Device Drivers
-->Graphics support
-->Frame buffer Devices
-->< > Support for frame buffer devices
-->Video support for sunxi
-->[ ] Framebuffer Console Support(sunxi)
-->< > DISP Driver Support(sunxi-disp2)
-->Console display driver support
-->< > Framebuffer Console support
-->Character devices
-->< > Transform Driver Support(sunxi)
```

选上 DRM 配置

```
Device Drivers
-->Graphics support
--><*> Direct Rendering Manager (XFree86 4.1.0 and higher DRI support)
--><*> DRM Support for Allwinnertech SoC A and R Series
```

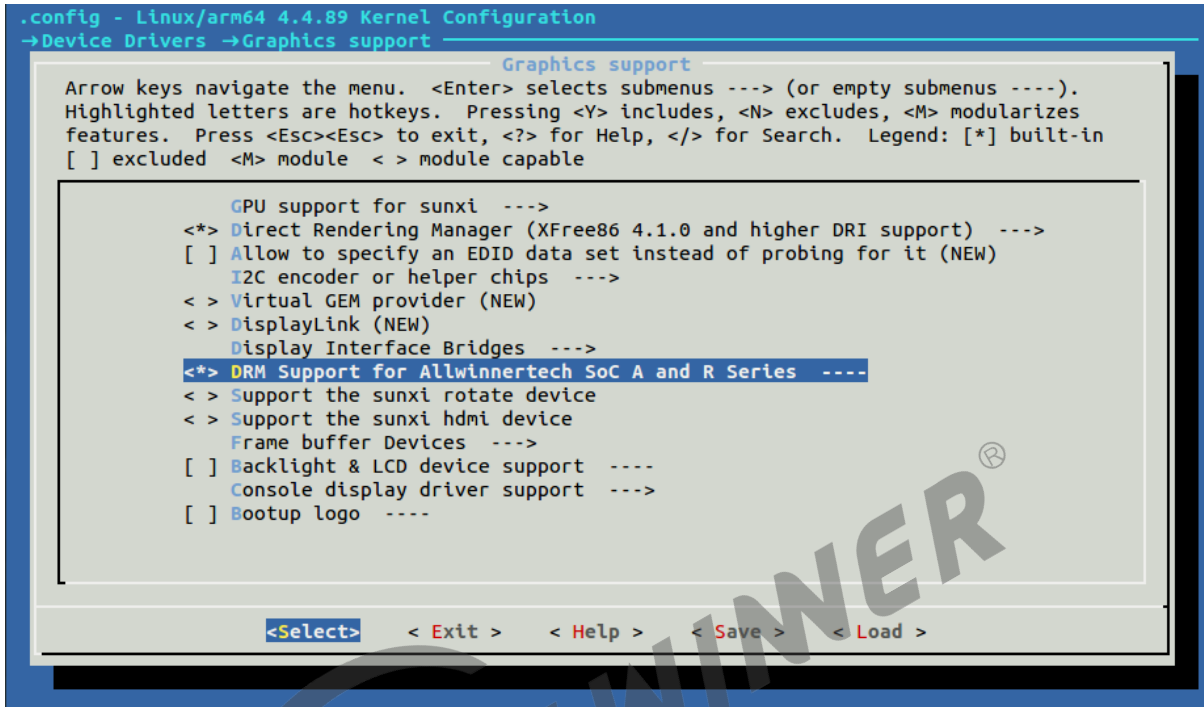


图 4-7: DRM 选项

选上 DRM 之后，要修改 tina/target/allwinner/方案名称/configs/sys\_config.fex 文件中 output\_type 的值，1 使用 framebuffer，0 不使用。使用 DRM 则修改成 0。

```
[boot_disp]
output_disp = 0
output_type = 0
output_mode = 4
```

## 4.3 GPU demo 配置

```
make menuconfig
--> <*>Utilities
--><*> gpu-sample /*gpu示例*/
--><*> opengl_demo /*opengl测试用例*/
--><*> opengles_demo /*opengles测试用例*/
```

```
make menuconfig
--<*>Wayland
--<*> glmark2 /*可用做gpu压力测试应用, 测gpu渲染性能, 即测试opengle性能*/
-->[*]Enabel fbdev support (NEW) /*fbdev和wayland二选一即可, wayland只能在r18上面使用*/
-->[]Enabel wayland support (NEW)
```



## 5 GPU 目录结构

package/kernel/gpu-km/ 是用来编译 GPU 内核驱动代码。

package/libs/libgpu 是用来管理 gpu 库, 这个文件夹里面提供了两种编译工具连 glibc 或 musl 以及/lib 或 lib64, 基于 fbdev 或 wayland 的 gpu 库。MR813 平台见: package/libs/libgpu-openc1 是用来管理 gpu 库。

### 5.1 添加新平台适配 gpu

```
ifeq ($(TARGET_PLATFORM), $(filter $(TARGET_PLATFORM), r16 r40 r18 r311 t7))
    GPU_TYPE:=mali400
else ifeq ($(TARGET_PLATFORM), $(filter $(TARGET_PLATFORM), r30))
    GPU_TYPE:=mali-t760
else ifeq ($(TARGET_PLATFORM), $(filter $(TARGET_PLATFORM), r58))
    GPU_TYPE:=sgx544
else ifeq ($(TARGET_PLATFORM), $(filter $(TARGET_PLATFORM), mr813 r818))
    GPU_TYPE:=ge8300
endif
```

图 5-1: 添加新平台适配 gpu

如上图, 在 package/libs/libgpu/Makefile 中上图位置添加平台的名称即可。

## 6 GPU 性能测试

### 6.1 测试工具 glmark2

gpu 压力测试，主要是通过测试工具 glmark2 来绘制图形，并查看 gpu 的使用率。

Tina 中不同平台可以选择的测试方法：在没有 wayland 的平台，只能使用 glmark2+fbdev 进行测试，对于有 wayland 的平台，既可以使用 glmark2+fbdev 进行测试，也可以通过 glmark2+wayland+drm 进行测试。

glmark 会运行一系列测试，在屏幕上呈现不同种类的 2D 和 3D 图形和动画，然后以 FPS（每秒帧数）的方式测量输出性能。然后在所有测试中平均出 fps 来计算 gpu 的分数。在测试结束时，glmark 会显示一个分数。较高的分数应该表示更强大的 GPU。最终得分不固定，每次进行测试时都会有所不同。但它确实保持在近距离范围内。所以你需要多次运行 glmark，并取平均分数

请注意，**glmark 分数不是图形性能的决定性测试**。从第三次测试中可以看出，分数有时会产生误导。例如，具有高 CPU 和没有硬件加速的机器可能与具有中等 CPU 和低 GPU 的机器相同

### 6.2 glmark2 配置

如下面的配置，glmark2 是 tina 中移植的测试 gpu 性能或者压力测试的应用。

```
make menuconfig
  --><*>Wayland
    --><*> glmark2 /*可用做gpu压力测试应用，测gpu性能*/
      -->[*]Enabel fbdev support (NEW) /*fbdev和wayland二选一即可，wayland只能在r18上面使用*/
      -->[]Enabel wayland support (NEW)
```

### 6.3 测试方法

在小机端运行应用程序 glmark2。

```
./glmark2-es2-fbdev
```

或者，

```
./glmark2-es2-wayland
```

查看应用支持哪些配置。

```
glmark2-es2-fbdev --help
```

设置分辨率：如果不是离屏渲染这里的分辨率不能大于 framebuffer 的大小。

```
./glmark2-es2-fbdev --resolution XXX XXX
```

设置离屏渲染：这里由于 ddr 的限制，目前只能设置的最大的分辨率位 4096x4096，要实现 GPU 达到高负载可以使用离屏渲染且将分辨率设置位最大。

```
./glmark2-es2-fbdev --off-screen
```

```
./glmark2-es2-fbdev --off-screen --resolution XXX XXX
```

下图是结果，可以看出 glmark2 score 的数值是测试的结果，其值越大，gpu 性能越好。

```

=====
OpenGL Information
GL_VENDOR:      ARM
GL_RENDERER:    Mali-400 MP
GL_VERSION:     OpenGL ES 2.0
=====
[build] use-vbo=false: FPS: 59 FrameTime: 16.949 ms
[build] use-vbo=true: FPS: 59 FrameTime: 16.949 ms
[texture] texture-filter=nearest: FPS: 59 FrameTime: 16.949 ms
[texture] texture-filter=linear: FPS: 59 FrameTime: 16.949 ms
[texture] texture-filter=mipmap: FPS: 60 FrameTime: 16.667 ms
[shading] shading=gouraud: FPS: 59 FrameTime: 16.949 ms
[shading] shading=blinn-phong-inf: FPS: 59 FrameTime: 16.949 ms
[shading] shading=phong: FPS: 60 FrameTime: 16.667 ms
[shading] shading=cgl: FPS: 59 FrameTime: 16.949 ms
[bump] bump-render=high-poly: FPS: 60 FrameTime: 16.667 ms
[bump] bump-render=normals: FPS: 59 FrameTime: 16.949 ms
[bump] bump-render=height: FPS: 59 FrameTime: 16.949 ms
[effect2d] kernel=0,1,0;1,-4,1;0,1,0;: FPS: 59 FrameTime: 16.949 ms
[effect2d] kernel=1,1,1;1,1,1;1,1,1;1,1,1;: FPS: 26 FrameTime: 38.462 ms
[pulsar] light=false:quads=5:texture=false: FPS: 60 FrameTime: 16.667 ms
[desktop] blur-radius=5:effect=blur:passes=1:separable=true:windows=4: FPS: 20 FrameTime: 50.000 ms
[desktop] effect=shadow:windows=4: FPS: 59 FrameTime: 16.949 ms
Error: Requested MapBuffer VBO update method but GL_OES_mapbuffer is not supported!
[buffer] columns=200:interleave=false:update-dispersion=0.9:update-fraction=0.5:update-method=map: Unsupported
[buffer] columns=200:interleave=false:update-dispersion=0.9:update-fraction=0.5:update-method=subdata: FPS: 45 FrameTime: 22.222 ms
Error: Requested MapBuffer VBO update method but GL_OES_mapbuffer is not supported!
[buffer] columns=200:interleave=true:update-dispersion=0.9:update-fraction=0.5:update-method=map: Unsupported
[ideas] speed=duration: FPS: 59 FrameTime: 16.949 ms
[jellyfish] <default>: FPS: 59 FrameTime: 16.949 ms
Error: SceneTerrain requires Vertex Texture Fetch support, but GL_MAX_VERTEX_TEXTURE_IMAGE_UNITS is 0
[terrain] <default>: Unsupported
[shadow] <default>: FPS: 59 FrameTime: 16.949 ms
[refract] <default>: FPS: 18 FrameTime: 55.556 ms
[conditionals] fragment-steps=0:vertex-steps=0: FPS: 59 FrameTime: 16.949 ms
[conditionals] fragment-steps=5:vertex-steps=0: FPS: 59 FrameTime: 16.949 ms
[conditionals] fragment-steps=0:vertex-steps=5: FPS: 60 FrameTime: 16.667 ms
[function] fragment-complexity=low:fragment-steps=5: FPS: 59 FrameTime: 16.949 ms
[function] fragment-complexity=medium:fragment-steps=5: FPS: 59 FrameTime: 16.949 ms
[loop] fragment-loop=false:fragment-steps=5:vertex-steps=5: FPS: 59 FrameTime: 16.949 ms
[loop] fragment-steps=5:fragment-uniform=false:vertex-steps=5: FPS: 60 FrameTime: 16.667 ms
[loop] fragment-steps=5:fragment-uniform=true:vertex-steps=5: FPS: 59 FrameTime: 16.949 ms
=====
glmark2 Score: 54
=====
root@TinaLinux:/# glmark2-es2-fbdev

```

图 6-1: glmark2 测试数据

使用 wayland 显示后端，需要先运行 weston。

```

chmod 0700 /dev/shm/
export XDG_RUNTIME_DIR=/dev/shm
export XDG_CONFIG_HOME=/etc/xdg
weston --backend=drm-backend.so --tty=1 --idle-time=0 &
或者
weston --backend=fbdev-backend.so --tty=1 --idle-time=0 &
如果没有/dev/shm/文件夹，手动创建即可

```

运行 glmark2，显示后端使用 wayland

```
./glmark2-es2-wayland
```

## 6.4 查看 GPU 使用率

### 6.4.1 mali400

在小机端的/sys/kernel/debug/mali路径下,执行while true; do cat utilization\_gp\_pp; sleep 1; done  
这个命令可以实时的对gpu进行监控。

如下图所示：

```
root@TinaLinux:/sys/kernel/debug/mali# while true; do cat utilization_gp_pp; sleep 1; done
27%
28%
27%
25%
28%
26%
26%
26%
```

图 6-2: gpu 使用率

### 6.4.2 mali-t760

在小机端/sys/kernel/debug/mali0 路径下,执行 while true; do cat gpu\_utilisation; sleep 1; done  
这个命令可以实时的对 gpu 进行监控。

### 6.4.3 sgx544

目前内核没有透出 gpu 使用率的接口，后面会完善。

### 6.4.4 ge8300

在小机端/sys/kernel/debug/pvr 路径下 status 节点，执行：

```
while true; do cat status; sleep 1; done
```

如下图此节点里面的内容，如果没有运行 gpu 的测试用例，则此节点的内容如下图 1，信息比较少。运行 gpu 测试用例之后，此节点的内容如下图 2，会有使用率的信息。

```
root@TinaLinux:/sys/kernel/debug/pvr# cat status
Driver Status:    OK
Firmware Status: OK
APM Event Count: 0
root@TinaLinux:/sys/kernel/debug/pvr#
```

图 6-3: status 节点内容 1

```
root@TinaLinux:/sys/kernel/debug/pvr# cd /usr/bin/OpenCLDemo/
root@TinaLinux:/usr/bin/OpenCLDemo# ./openc1_demo
1 + 4 -> expected result: 5, openc1 result: 5
2 + 5 -> expected result: 7, openc1 result: 7
3 + 6 -> expected result: 9, openc1 result: 9
root@TinaLinux:/usr/bin/OpenCLDemo# cd -
/sys/kernel/debug/pvr
root@TinaLinux:/sys/kernel/debug/pvr# cat status
Driver Status:    OK
Firmware Status: OK
HWR Event Count: 0
CRR Event Count: 0
FWF Event Count: 0
APM Event Count: 2
GPU Utilisation: 0%
root@TinaLinux:/sys/kernel/debug/pvr#
```

图 6-4: status 节点内容 2

## 6.5 查看 GPU 频率

### 6.5.1 mali400

```
cat /kernel/debug/clk/pll_gpu/clk_rate
```

## 6.5.2 mali-t760

```
cat /kernel/debug/clk/pll_gpu/clk_rate
```

## 6.5.3 sgx544

```
cat /kernel/debug/clk/hosc/pll_gpu/clk_rate
```

## 6.5.4 ge8300

```
cat /sys/kernel/debug/sunxi_gpu/dump
```

## 6.6 查看 GPU 电压

```
cat sys/class/regulator/dump
```

## 6.7 查看 GPU 温度

```
while true; do cat /sys/class/thermal/thermal_zone0/temp; sleep 1; done
```

## 6.8 glmark2 参数分析

```
(1)[build] use-vbo=true: FPS: 59 FrameTime: 16.949 ms
```

创建顶点对象 (vbo) 成功, 每秒传输帧数为 59, 传输一帧 16.949ms

```
(2)[texture] texture-filter=nearest: FPS: 59 FrameTime: 16.949 ms  
[texture] texture-filter=linear: FPS: 59 FrameTime: 16.949 ms
```

分别表示纹理过滤的两种模式的速度, texture-filter=nearest 最近点采样 texture-filter=linear 线性采样

```
(3)[texture] texture-filter=mipmap: FPS: 60 FrameTime: 16.667 ms
```

纹理映射速率

```
(4)[shading] shading=gouraud: FPS: 59 FrameTime: 16.949 ms  
[shading] shading=blinn-phong-inf: FPS: 59 FrameTime: 16.949 ms  
[shading] shading=phong: FPS: 60 FrameTime: 16.667 ms  
[shading] shading=cel: FPS: 59 FrameTime: 16.949 ms
```

表示定点着色器的着色方式, shading=gouraud: 对三角面上每个顶点颜色进行线性插值计。shading=blinn-phong-inf: 对三角面上每个顶点法线进行插值, 跟 phone 相似。shading=phong: 对三角面上每个顶点法线进行插值 shading=cel: 卡通渲染

```
(5)[bump] bump-render=high-poly: FPS: 60 FrameTime: 16.667 ms  
[bump] bump-render=normals: FPS: 59 FrameTime: 16.949 ms  
[bump] bump-render=height: FPS: 59 FrameTime: 16.949 ms
```

表示凹凸映射的速率, 就是把粗糙信息加到图形上。

```
(6)[effect2d] kernel=0,1,0;1,-4,1;0,1,0;: FPS: 59 FrameTime: 16.949 ms  
[effect2d] kernel=1,1,1,1;1,1,1,1;1,1,1,1;: FPS: 26 FrameTime: 38.462 ms
```

表示 2D 纹理渲染

```
(7)[pulsar] light=false:quads=5:texture=false: FPS: 60 FrameTime: 16.667 ms
[desktop]blur-radius=5:effect=blur:passes=1:separable=true:windows=4:FPS:59FrameTime
:16.949ms
[desktop] effect=shadow:windows=4: FPS: 212 FrameTime: 4.717 ms
[buffer] columns=200:interleave=false:update-dispersion=0.9:update-fraction=0.5:update-
method=map: FPS:52 FrameTime: 19.231 ms
```

```
(8)[shadow] <default>: FPS: 59FrameTime: 16.949 ms
```

顶点着色器。

```
(9)[refract] <default>: FPS: 18 FrameTime:55.556 ms
```

表示反射，折射速。

```
(10)[conditionals] fragment-steps=0:vertex-steps=0: FPS: 427 FrameTime: 2.342 ms
[function]fragment-complexity=medium:fragment-steps=5:FPS:51FrameTime:19.608ms
[loop] fragment-loop=false:fragment-steps=5:vertex-steps=5: FPS: 59 FrameTime: 16.949 ms
[loop]fragment-steps=5:fragment-uniform=false:vertex-steps=5:FPS:60FrameTime:16.667ms
[loop] fragment-steps=5:fragment-uniform=true:vertex-steps=5: FPS: 59 FrameTime: 16.949 ms
```

表示片段操作速率。

## 7 GPU 问题解决

### **eglGetError() = 0x3003:bad alloc**

问题：执行 gpu 的测试用例，报错提示信息为：**eglGetError() = 0x3003:bad alloc**

解决方法：查看 tina/target/allwinner/astar-parrot/configs/sys\_config.fex 文件设置的 lcd 的大小 lcd\_x 和 lcd\_y，如图：

```
[lcd0_para]
lcd_used           = 1
lcd_driver_name    = "st7701s_lcd"
lcd_if             = 4
lcd_x              = 480
lcd_y              = 800
lcd_width          = 52
```

图 7-1: gpu 错误 bad alloc

运行 `opengles_demo -width 480 -height 800` 设置的 width 和 height 不能大于 lcd 的分辨率。

问题原因：是因为设置的 surface 比本身 lcd 的分辨率大导致的申请 surface 失败。

### **bad native windows**

解决方法：

此问题一般由于显示驱动未加载，判断显示驱动加载是否成功 1. 看开机 logo 是否正常，2. 看启动 log，DISP 模块是否报错。也有可能出错的原因是 display 属性初始化错误，导致 opengles 在获取 display 的属性时候出错。之前在 R30 上面遇到过这种情况，是 R30 的开机 logo 那张图片导致 framebuffer 的属性发生变化，opengles\_demo 获取 fb 的信息错误。

其它问题：

- 1. 检查 mali.ko 是否加载。
- 2. 给板子加上移动电源，因为有时候跑 gpu，不加移动电源应用跑不起来。
- 3. 跟 disp 显示有关的驱动检查。

- 4. 目前 tina 库中只有 mali400 型号的 gpu 支持 wayland, 其它型号 gpu 没有对接 wayland, 且 wayland 只在 R18 平台支持。






## 著作权声明

版权所有 © 2020 珠海全志科技股份有限公司。保留一切权利。

本档及内容受著作权法保护，其著作权由珠海全志科技股份有限公司（“全志”）拥有并保留一切权利。

本档是全志的原创作品和版权财产，未经全志书面许可，任何单位和个人不得擅自摘抄、复制、修改、发表或传播本档内容的部分或全部，且不得以任何形式传播。

## 商标声明

、 **全志科技** （不完全列举）均为珠海全志科技股份有限公司的商标或者注册商标。在本档描述的产品中出现的其它商标，产品名称，和服务名称，均由其各自所有人拥有。

## 免责声明

您购买的产品、服务或特性应受您与珠海全志科技股份有限公司（“全志”）之间签署的商业合同和条款的约束。本档中描述的全部或部分产品、服务或特性可能不在您所购买或使用的范围内。使用前请认真阅读合同条款和相关说明，并严格遵循本档的使用说明。您将自行承担任何不当使用行为（包括但不限于如超压，超频，超温使用）造成的不利后果，全志概不负责。

本档作为使用指导仅供参考。由于产品版本升级或其他原因，本档内容有可能修改，如有变更，恕不另行通知。全志尽全力在本档中提供准确的信息，但并不确保内容完全没有错误，因使用本档而发生损害（包括但不限于间接的、偶然的、特殊的损失）或发生侵犯第三方权利事件，全志概不负责。本档中的所有陈述、信息和建议并不构成任何明示或暗示的保证或承诺。

本档未以明示或暗示或其他方式授予全志的任何专利或知识产权。在您实施方案或使用产品的过程中，可能需要获得第三方的权利许可。请您自行向第三方权利人获取相关的许可。全志不承担也不代为支付任何关于获取第三方许可的许可费或版税（专利税）。全志不对您所使用的第三方许可技术做出任何保证、赔偿或承担其他义务。