Rockchip Buildroot Weston Developer Guide

ID: RK-KF-YF-326

Release Version: V1.0.1

Release Date: 2020-07-22

Security Level: □Top-Secret □Secret □Internal ■Public

Rockchip Electronics Co., Ltd.

No.18 Building, A District, No.89, software Boulevard Fuzhou, Fujian,PRC

Website: www.rock-chips.com

Customer service Tel: +86-4007-700-590

Customer service Fax: +86-591-83951833

Customer service e-Mail: fae@rock-chips.com

Preface

Overview

This document presents the basic configuration methods of Buildroot SDK Weston display service.

Product Version

Chipset	Kernel Version
All chipset	4.4

Intended Audience

This document (this guide) is mainly intended for:

Technical support engineers

Software development engineers

Revision History

Version	Author	Date	Revision History
V1.0.0	Jeffy Chen	2019-11-27	Initial version
V1.0.1	Ruby Zhang	2020-07-23	Update the company name and the format of the document

Contents

1. Introduction

1.1 Overview

Weston is the official implementation reference of Wayland open source display protocol, and Weston 3.0 drm back-end is used in Rockchip Buildroot SDK by default.

1.2 Configuration Methods

There are multiple ways to configure Rockchip Buildroot SDK Weston:

a. Command line parameters

That is, the parameters of the command when starting Weston, such as weston --tty=2.

b. weston.ini configuration file

It's located in /etc/xdg/weston/weston.ini, corresponding to the location of SDK code: buildroot/board/rockchip/common/base/etc/xdg/weston/weston.ini.

Please refer to: https://fossies.org/linux/weston/man/weston.ini.man.

c. Special environment variables

Generally, these environment variables are set in the startup script of Weston, which is located in /etc/init.d/S50launcher of the SDK firmware, for example:

```
    # /etc/init.d/S50launcher
      start)
                    ...
                    export WESTON_DRM_MIRROR=1 # should be set before
starting weston
                    ...
                    weston --tty=2 -B=drm-backend.so --idle-time=0&
```

d. Dynamic configuration file

For drm back-end, Buildroot SDK Weston provides some dynamic configuration support, such as dynamic display configuration files, the default path is /tmp/.weston_drm.conf, which can be specified by the environment variable WESTON_DRM_CONFIG.

e. udev rules

Part of the configurations of input devices in Weston should be set by `udev rules`.

2. Configuration

2.1 Status Bar Configuration

Weston supports setting the background color and position of status bar in the `shell` section of weston.ini configuration file, and setting the quick start program in the `launcher` section, for example:

```
# /etc/xdg/weston/weston.ini

[shell]
panel-color=0x90ff0000
# the color format is ARGB8888

panel-position=bottom
# top|bottom|left|right|none,none is disable

[launcher]
icon=/usr/share/icons/gnome/24x24/apps/utilities-terminal.png
# icon path

path=/usr/bin/gnome-terminal
# quick start command
```

Currently, Weston does not support setting the size of status bar. You have to modify in the code level when need some adjustments:

```
// weston-3.0.0/clients/desktop-shell.c

static void
panel_configure(void *data,
                struct weston_desktop_shell *desktop_shell,
                uint32_t edges, struct window *window,
                int32_t width, int32_t height)
{
        ...
        switch (desktop->panel_position) {
        case WESTON_DESKTOP_SHELL_PANEL_POSITION_TOP:
        case WESTON_DESKTOP_SHELL_PANEL_POSITION_BOTTOM:
                height = 32; # height
                break;
        case WESTON_DESKTOP_SHELL_PANEL_POSITION_LEFT:
        case WESTON_DESKTOP_SHELL_PANEL_POSITION_RIGHT:
                switch (desktop->clock_format) {
                case CLOCK_FORMAT_NONE:
                        width = 32;
                        break;
                case CLOCK_FORMAT_MINUTES:
                        width = 170;
                        break;
                case CLOCK_FORMAT_SECONDS:
                        width = 190;
                        break;
                }
                break;
        }
```

2.2 Background Configuration

Weston supports setting the background pattern and color in the `shell` section of the weston.ini configuration file, such as

```
    # /etc/xdg/weston/weston.ini

    [shell]
    background-image=/usr/share/backgrounds/gnome/Aqua.jpg
    # Background pattern (wallpaper) absolute path

    background-type=tile
    # scale|scale-crop|tile

    background-color=0xff002244
    # The color format is ARGB8888, effective when no background pattern is
set
```

2.3 Idle Time and Lock Screen Configuration

The idle timeout of Weston can be configured in the startup parameters or in the `core` section of weston.ini, such as:

```
    # /etc/init.d/S50launcher
      start)
                    ...
                    weston --tty=2 -B=drm-backend.so --idle-time=0&   # 0
means idle mode is disabled, in seconds
```

Or

```
    # /etc/xdg/weston/weston.ini

    [core]
    idle-time=10
```

Lock screen of Weston can be configured in the `shell` section of weston.ini, such as:

```
    # /etc/xdg/weston/weston.ini

    [shell]
    locking=false
    # lock screen is disabled

    lockscreen-icon=/usr/share/icons/gnome/256x256/actions/lock.png
    # unlock button icon

    lockscreen=/usr/share/backgrounds/gnome/Garden.jpg
    # background of lock screen
```

2.4 Color Format Configuration

The default display format of Weston in the Buildroot SDK is ARGB8888. For some low-performance platforms, you can configure RGB565 in the `core` section in the weston.ini, such as:

```
    # /etc/xdg/weston/weston.ini

    [core]
    gbm-format=rgb565
    # xrgb8888|rgb565|xrgb2101010
```

You can also configure the display format of each screen individually in the `output` section of weston.ini, such as:

```
    # /etc/xdg/weston/weston.ini

    [output]
    name=LVDS-1
    # output name can be seen in the Weston startup log, such as: Output LVDS-1, (connector 71, crtc 60)

    gbm-format=rgb565
    # xrgb8888|rgb565|xrgb2101010
```

2.5 Display Orientation Configuration

Display orientation of screens can be configured in the `output` section of weston.ini, such as

```
    # /etc/xdg/weston/weston.ini

    [output]
    name=LVDS-1

    transform=90
    # normal|90|180|270|flipped|flipped-90|flipped-180|flipped-270
```

If you want to configure the orientation dynamically, the dynamic configuration file can be used, such as:

```
    echo "output:all:rotate90" > /tmp/.weston_drm.conf # All screens rotate 90 degrees
    echo "output:eDP-1::rotate180" > /tmp/.weston_drm.conf # eDP-1 rotates 180 degrees
```

2.6 Resolution and Scale Configuration

Screen's resolution and scale of Weston can be configured in the `output` section of weston.ini, such as:

```
    # /etc/xdg/weston/weston.ini

    [output]
    name=LVDS-1

    mode=1280x800
    # should be an effective resolution supported by the screen

    scale=2
    # value must be an integer, support application level scaling
```

If you want to scale to a specific resolution, you can configure it through WESTON_DRM_VIRTUAL_SIZE environment variable, such as:

```
    # /etc/init.d/S50launcher
      start)
                    ...
                    export WESTON_DRM_VIRTUAL_SIZE=1024x768
                    weston --tty=2 -B=drm-backend.so --idle-time=0&
```

It requires the display driver to support hardware scaling. Some chip platforms do not support scaling with alpha transparency. Please refer to the above description to modify the display color format to XRGB8888 or other formats.

If you want to configure resolution and scaling dynamically, the dynamic configuration file can be used, for example:

```
    echo "output:HDMI-A-1:mode=800x600" > /tmp/.weston_drm.conf # change the resolution of HDMI-A-1 to 800x600
    echo "output:eDP-1:rect=<10,20,410,620>" > /tmp/.weston_drm.conf # eDP-1 display to the position of (10,20), the size is scaled to 400x600
```

This kind of scale depends on Rockchip's RGA 2D acceleration.

2.7 Freeze the Screen

When Weston is started, there will be a black screen that switches between boot logo and UI display temporarily. If you want to prevent this black screen, you can freeze the Weston screen content temporarily through the following dynamic configuration file:

```
    # /etc/init.d/S50launcher
      start)
                    ...
                    export WESTON_FREEZE_DISPLAY=/tmp/.weston_freeze # Set a special configuration file path
                    touch /tmp/.weston_freeze # Freeze the display
                    weston --tty=2 -B=drm-backend.so --idle-time=0&
                    ...
                    sleep 1 && rm /tmp/.weston_freeze& # unfreeze after 1 second
```

Or

```
    # /etc/init.d/S50launcher
      start)
                    ...
                    echo "output:all:freeze" > /tmp/.weston_drm.conf # Freeze
the display
                    weston --tty=2 -B=drm-backend.so --idle-time=0&
                    ...
                    sleep 1 && \
                        echo "output:all:unfreeze" > /tmp/.weston_drm.conf& #
unfreeze  after 1 second
```

2.8 Multi-screen Configuration

The Buildroot SDK Weston supports multi-screen with the same or different display and hot plug functions. You can distinction different screens based on the name of drm (obtained through Weston startup log or /sys/class/drm/card0-<name>). Some configurations can be set by environment variables, for example:

```
    # /etc/init.d/S50launcher
      start)
                    ...
                    export WESTON_DRM_PRIMARY=HDMI-A-1 # Specify HDMI-A-1 as
a main display
                    export WESTON_DRM_MIRROR=1 # In mirror mode (multi-screen
with the same display), without setting this environment variable will be
with different display
                    export WESTON_DRM_KEEP_RATIO=1 # In mirror mode, scaling
maintains the aspect ratio, without setting this variable will be full screen
by force
                    export WESTON_DRM_PREFER_EXTERNAL=1 # Turn off the built-
in monitor automatically when an external monitor is connected
                    export WESTON_DRM_PREFER_EXTERNAL_DUAL=1 # When an
external monitor is connected, keep the first external monitor as the main
display by default
                    weston --tty=2 -B=drm-backend.so --idle-time=0&
```

In mirror mode, scaling display content depends on Rockchip's RGA 2D acceleration.

It also supports disabling the specified screen individually in the `output` section of weston.ini:

```
    # /etc/xdg/weston/weston.ini

    [output]
    name=LVDS-1

    mode=off
    # off|current|preferred|<WIDTHxHEIGHT@RATE>
```

2.9 Configuration of Input Devices

The Weston service requires at least one input device by default. If there is no input device, the special settings in the `core` section of weston.ini is needed:

```
    # /etc/xdg/weston/weston.ini

    [core]
    require-input=false
```

If there are multiple screens in Weston, input devices should be bound to screens, you can configure it through the `output` section of weston.ini, such as:

```
    # /etc/xdg/weston/weston.ini

    [output]
    name=LVDS-1

    seat=default
    # The id for seat of the input device can be found through
buildroot/output/*/build/weston-3.0.0/weston-info tool
```

Input devices of Weston are based on libinput, so if you need to calibrate the touch screen, you can configure LIBINPUT_CALIBRATION_MATRIX in udev rules through the standard method of libinput, such as:

```
    # cat /etc/udev/rules.d/99-touch-cali.rules
    ATTRS{name}=="Fujitsu Component USB Touch Panel",
ENV{LIBINPUT_CALIBRATION_MATRIX}="1.013788 0.0 -0.061495 0.0 1.332709
-0.276154"
```

The calibration parameters can be obtained by Weston calibration tool: buildroot/output/<board>/build/weston/weston-calibrator. After running this tool, a number of random points will be generated, and then click them in sequence to output the calibration parameters, such as: calibration values: 1.013788 0.0- 78.713867 0.0 1.332709 -220.923355

The third and sixth parameters should be divided by the screen's width and height respectively. Taking the resolution of 1280x800 as an example, the final calibration parameter is 1.013788 0.0 -0.061495 (that is, -78.713867 divided by 1280) 0.0 1.332709 -0.276154 (that is -220.923355 divided by 800).

2.10 Configuration on the Platform without GPU

The Weston in the SDK uses GPU for render acceleration by default. For platforms without GPUs, Rockchip RGA 2D acceleration can also be used instead.

To enable this function, please ensure that the Buildroot repository is updated after this commit:

```
    commit 6873e04dd246c0b969c19bcc38549c3e012a4b20
    Author: Jeffy Chen <jeffy.chen@rock-chips.com>
    Date:   Fri Nov 1 18:44:36 2019 +0800

        pixman: pixman_image_composite32: Support rockchip RGA 2D accel

        Disabled by default, set env PIXMAN_USE_RGA=1 to enable.

        Change-Id: I674450da1fd713609cb7a1da790a5a3b8057d3c4
        Signed-off-by: Jeffy Chen <jeffy.chen@rock-chips.com>
```

The detailed configuration requires to enable BR2_PACKAGE_LINUX_RGA in the Buildroot SDK, and then configure PIXMAN_USE_RGA environment variable to 1, and add --use-pixman to Weston startup parameters, such as:

```
    # /etc/init.d/S50launcher
      start)
                    ...
                    export PIXMAN_USE_RGA=1
                    ...
                    weston --tty=2 -B=drm-backend.so --idle-time=0 --use-
pixman&
```