

# N1 Smart Module Adding LCD Driver

Version 1.0



**Copyright © Neoway Technology Co., Ltd 2016. All rights reserved.**

No part of this document may be reproduced or transmitted in any form or by any means without prior written consent of Shenzhen Neoway Technology Co., Ltd.

**Neoway 有方** is the trademark of Neoway Technology Co., Ltd.

All other trademarks and trade names mentioned in this document are the property of their respective holders.

**Notice**

This document provides guide for users to use the N1.

This document is intended for system engineers (SEs), development engineers, and test engineers.

The information in this document is subject to change without notice due to product version update or other reasons.

Every effort has been made in preparation of this document to ensure accuracy of the contents, but all statements, information, and recommendations in this document do not constitute a warranty of any kind, express or implied.

Neoway provides customers complete technical support. If you have any question, please contact your account manager or email to the following email addresses:

Sales@neoway.com

Support@neoway.com

Website: <http://www.neoway.com>

<b>Revision Record</b>		
<b>Issue</b>	<b>Changes</b>	<b>Date</b>
V1.0	Initial draft	2016-06



# Contents

1 Overview.....	1
2 Adding LCD Driver to Bootloader.....	1
3 Adding LCD Driver to Kernel.....	4



# 1 Overview

This document describes the process of adding LCD driver for the N1 smart module. NT35517 is used as an example to detail how to add the driver to bootloader and kernel. You can add other LCD drivers by referring to this document.

## 2 Adding LCD Driver to Bootloader

- bootable\bootloader\lk\target\msm8909\jsr\_feature.mk

Add a macro to the item selected for compiling.

```
LCD_NT35517_SUPPORT_PROJECTS := E45T E45T_TEST AM809_TEST AM809
ifneq ($(filter $(TARGET_PRODUCT), $(LCD_NT35517_SUPPORT_PROJECTS)),)
  DEFINES += FEATURE_JSRLCD_NT35517_VIDEO_MODE=1
  DEFINES += FEATURE_NEO_AUTOMATIC_STARTUP=1
endif
```

- bootable\bootloader\lk\dev\gcdb\display\include\panel\_nt35517\_qhd\_video.h

Add the driver IC file of your LCD to the **include** fold. You can choose cmd or video as the MIPI mode and name the .h file.

Open the header file and note the following key information when you add.

```
48: static struct panel_config nt35517_qhd_video_panel_data = {
49:     "qcom,mdss_dsi_nt35517_qhd_video", "dsi:0:", "qcom,mdss-dsi-panel",
50:     10, 0, "DISPLAY_1", 0, 0, 60, 0, 0, 0, 1, 7000, 0, 0, 0, 0, 0
51: };
```

Ensure that you enter correct information in the quotation marks. The information will be used in kernel and the MIPI mode should be set to video or cmd.

Enter the actual resolution data.

```
153: /*-----*/
154: /* Panel resolution */
155: /*-----*/
156: static struct panel_resolution nt35517_qhd_video_panel_res = {
157:     540, 960, 100, 100, 8, 0, 20, 20, 1, 0, 0, 0, 0, 0, 0, 0, 0
158: };
```

Check the LCD you use and then set the lane configurations.

```
103: /*-----*/
104: /* Lane Configuration */
105: /*-----*/
106:
107: static struct lane_configuration nt35517_qhd_video_lane_config = {
108:     2, 1, 1, 1, 0, 0
109: };
```

For other data, please check the header file for details.

The following figure shows the initialization data sets.

```

41: static char nt35517_goworld_video_set_01[12] = {0x05,0x00,0x39,0xc0,0xFF,0xAA,0x55,0x25,0x01,0xff,0xff,0xff};
42: static char nt35517_goworld_video_set_02[12] = {0x06,0x00,0x39,0xc0,0xF0,0x55,0xAA,0x52,0x08,0x00,0xff,0xff};
43: static char nt35517_goworld_video_set_03[8] = {0x02,0x00,0x39,0xc0,0xB1,0xFC,0xff,0xff};
44: static char nt35517_goworld_video_set_04[12] = {0x05,0x00,0x39,0xc0,0xB8,0x01,0x02,0x02,0x02,0xff,0xff,0xff};
45: static char nt35517_goworld_video_set_05[8] = {0x02,0x00,0x39,0xc0,0xBB,0x33,0xff,0xff};
46: static char nt35517_goworld_video_set_06[12] = {0x07,0x00,0x39,0xc0,0xC9,0x63,0x06,0x0D,0x1A,0x17,0x00,0xff};
47: static char nt35517_goworld_video_set_07[8] = {0x03,0x00,0x39,0xc0,0xE0,0x01,0x03,0xff};
48: static char nt35517_goworld_video_set_08[12] = {0x06,0x00,0x39,0xc0,0xF0,0x55,0xAA,0x52,0x08,0x01,0xff,0xff};
49: static char nt35517_goworld_video_set_09[8] = {0x04,0x00,0x39,0xc0,0xB0,0x05,0x05,0x05}; // 5.5v 0a
50: static char nt35517_goworld_video_set_10[8] = {0x04,0x00,0x39,0xc0,0xB2,0x02,0x02,0x02}; // 5.5v 0a

```

In the first line, according to the MIPI protocol,

- 0x05 indicates that there are five types of information following;
- 0x39 is generic long write;
- 0xC0 is checksum;
- 0xFF is the address to which data is written;
- 0xAA, 0x55, 0x25, 0x01 are the data written in the register;
- 0xff is used to fill the vacancy of the data set. (because the data sent by MIPI is encapsulated at certain length, and for the N1 platform you must ensure that the data set length is a multiple of 4.)

### 3. bootable\bootloader\lk\target\msm8909\oem\_panel.c

Quote the above .h file.

```

)53: #if FEATURE_JSR_LCD_NT35517_VIDEO_MODE
)54: #include "include/panel_nt35517_qhd_video.h"
)55: #include "include/panel_nt35517_qhd_video_success_init.h"
)56: #include "include/panel_nt35517_qhd_video_goworld_init.h"
)57:
)58: #endif

```

Add NT35517\_QHD\_VIDEO\_PANEL.

```

)82: #if FEATURE_JSR_LCD_NT35517_VIDEO_MODE
)83: NT35517_QHD_VIDEO_PANEL,
)84: LCD_PANEL_NT35517_GOWORLD,
)85: LCD_PANEL_NT35517_SUCCESS,
)86: #endif

```

Add LCD commissioning data to the initialization data function.

```

)285: static int init_panel_data(struct panel_struct *panelstruct,
)286:                          struct msm_panel_info *pinfo,
)287:                          struct mdss_dsi_phy_ctrl *phy_db)
)288: {
)289:     int pan_type = PANEL_TYPE_DSI;

```



```

1297: #if FEATURE_JSR_LCD_NT35517_VIDEO_MODE
1298:     panel_id=NT35517_QHD_VIDEO_PANEL;
1299: #endif
1300:     dprintf(CRITICAL, "lcd panel_id=%d\n", panel_id);
1301:
1302:     switch (panel_id) {

1384:     #if FEATURE_JSR_LCD_NT35517_VIDEO_MODE
1385:     case NT35517_QHD_VIDEO_PANEL:
1386:         panelstruct->paneldata = &nt35517_qhd_video_panel_data;
1387:         panelstruct->panelres = &nt35517_qhd_video_panel_res;
1388:         panelstruct->color = &nt35517_qhd_video_color;
1389:         panelstruct->videopanel = &nt35517_qhd_video_video_panel;
1390:         panelstruct->commandpanel = &nt35517_qhd_video_command_panel;
1391:         panelstruct->state = &nt35517_qhd_video_state;
1392:         panelstruct->laneconfig = &nt35517_qhd_video_lane_config;
1393:         panelstruct->paneltiminginfo
1394:             = &nt35517_qhd_video_timing_info;
1395:         panelstruct->panelresetseq
1396:             = &nt35517_qhd_video_panel_reset_seq;
1397:         panelstruct->backlightinfo = &nt35517_qhd_video_backlight;
1398:
1399:         {
1400:             pinfo->mipi.panel_cmds= nt35517_goworld_qhd_panel_video_mode_init;
1401:             pinfo->mipi.num_of_panel_cmds = NT35517_GOWORLD_VIDEO_CMDON_NUM;
1402:         }
1403:         memcpy(phy_db->timing,
1404:             nt35517_qhd_video_timings, TIMING_SIZE);
1405:         pinfo->mipi.signature = NT35517_QHD_VIDEO_SIGNATURE;
1406:         break;
1407:     #endif

```

The data sets transmitted are that modified in the .h file.

### 3 Adding LCD Driver to Kernel

1. kernel\drivers\video\msm\mdss\Kconfig

Add a macro label of LCD.

```
53 config JSR_LCD_NT35517_SUPPORT
54     depends on FB_MSM_MDSS
55     bool 'JSR NT35517 Panel'
56     default n
57     ---help---
58     Support for NT35517 panel.
```

2. kernel\arch\arm\configs\msm8909-1gb\_N1\_defconfig

msm8909-1gb-perf\_N1\_defconfig kernel

Add and open the macro.

```
629: #CONFIG_JSJ_LCD_HX8379C_FWVGA_VIDEO_SUPPORT=y
630: #CONFIG_JSJ_LCD_ILI9806E_FWVGA_VIDEO_SUPPORT=y
631: CONFIG_JSJ_LCD_NT35517_SUPPORT=y
```

3. kernel\arch\arm\boot\dts\jsr\N1\_msm8909-qrd-skue.dtsi

Add the following information to the file.

```
499: &dsi_nt35517_qhd_vid {
500:     qcom,cont-splash-enabled;
501: };

524: &mdss_dsi0 {
525:     qcom,dsi-pref-prim-pan = <&dsi_nt35517_qhd_vid>;
526:     pinctrl-names = "mdss_default", "mdss_sleep";
527:     pinctrl-0 = <&mdss_dsi_active &mdss_te_active>;
528:     pinctrl-1 = <&mdss_dsi_suspend &mdss_te_suspend>;
529:
530:     qcom,platform-reset-gpio = <&msm_gpio 25 0>;
531: };
```

4. kernel\arch\arm\boot\dts\jsr\dsi-panel-nt35517-qhd-video.dtsi

Add a .dtsi file to the dts folder. The content of the file is the same as that in the panel\_nt35517\_qhd\_video.h file.

#### NOTE

The **Panel-name** should be the same as that in the bootloader. Note how to fill the resolution, video/cmd mode, and vfp/hfp.



```

18: &mdss_mdp {
19:     dsi_nt35517_qhd_vid: qcom,mdss-dsi-nt35517-qhd-video {
20:         qcom,mdss-dsi-panel-name = "nt35517 qhd video mode dsi panel";
21:         qcom,mdss-dsi-panel-controller = <&mdss_dsi0>;
22:         qcom,mdss-dsi-panel-type = "dsi_video_mode";
23:         qcom,mdss-dsi-panel-destination = "display_1";
24:         qcom,mdss-dsi-panel-framerate = <60>;
25:         qcom,mdss-dsi-virtual-channel-id = <0>;
26:         qcom,mdss-dsi-stream = <0>;
27:         qcom,mdss-dsi-panel-width = <540>;
28:         qcom,mdss-dsi-panel-height = <960>;
29:         qcom,mdss-dsi-h-front-porch = <100>;
30:         qcom,mdss-dsi-h-back-porch = <100>;
31:         qcom,mdss-dsi-h-pulse-width = <8>;
32:         qcom,mdss-dsi-h-sync-skew = <0>;
33:         qcom,mdss-dsi-v-back-porch = <20>;
34:         qcom,mdss-dsi-v-front-porch = <20>;
35:         qcom,mdss-dsi-v-pulse-width = <1>;
36:         qcom,mdss-dsi-h-left-border = <0>;
37:         qcom,mdss-dsi-h-right-border = <0>;
38:         qcom,mdss-dsi-v-top-border = <0>;
39:         qcom,mdss-dsi-v-bottom-border = <0>;
40:         qcom,mdss-dsi-bpp = <24>;
41:         qcom,mdss-dsi-color-order = <0>;
42:         qcom,mdss-dsi-underflow-color = <0xff>;
43:         qcom,mdss-dsi-border-color = <0>;
44:         qcom,ulps-enabled;

```

The values of the LCD IC register are the same. And you do not have to add 0xff to fill the vacancy

```

45:         qcom,mdss-dsi-on-command-nt35517-goworld = [
46:             39 01 00 00 00 00 05
47:             FF AA 55 25 01
48:             39 01 00 00 00 00 06
49:             F0 55 AA 52 08 00
50:             39 01 00 00 00 00 02
51:             B1 FC
52:             39 01 00 00 00 00 05
53:             B8 01 02 02 02
54:             39 01 00 00 00 00 07
55:             C9 63 06 0D 1A 17 00
56:             39 01 00 00 00 00 03
57:             E0 01 03
58:             39 01 00 00 00 00 06
59:             F0 55 AA 52 08 01
60:             39 01 00 00 00 00 04
61:             B0 05 05 05

```

For details about parameter description, please refer to files in the

**\kernel\Documentation\devicetree\bindings\fb** directory:

**\kernel\Documentation\devicetree\bindings\fb\mdss-dsi-panel.txt**

**\kernel\Documentation\devicetree\bindings\fb\mdss-dsi-ctrl.txt.**

- kernel/arch/arm/boot/dts/jsr/N1\_msm8909-qrd-panels.dtsi

Add the above DTSI file to **N1\_msm8909-qrd-panels.dtsi** to facilitate quotation.

```

13: #include "dsi-panel-hx8379c-fwvga-video.dtsi"
14: #include "dsi-panel-ili9806e-fwvga-video.dtsi"
15: #include "dsi-panel-nt35517-qhd-video.dtsi"

```

6. kernel/drivers/video/msm/mdss/mdss\_dsi\_panel.c

Add your LCD driver evoking to the function parsing the DTSI file. (Modify the **display on** command to point to the **on** commands in the above DTSI file.)

```

1512: static int mdss_panel_parse_dt(struct device_node *np,
1513:                                struct mdss_dsi_ctrl_pdata *ctrl_pdata)
1514: {
1515:     ...
1857: #elif defined CONFIG_JSR_LCD_NT35517_SUPPORT
1858:     if (lcd_panel_module_type == LCD_PANEL_NT35517_GOWORLD){
1859:         mdss_dsi_parse_dcs_cmds(np, &ctrl_pdata->on_cmds,
1860:                                "qcom,mdss-dsi-on-command-nt35517-goworld", "qcom,mdss-dsi-on-command-state");
1861:     }
1862:     else{
1863:         mdss_dsi_parse_dcs_cmds(np, &ctrl_pdata->on_cmds,
1864:                                "qcom,mdss-dsi-on-command-nt35517-goworld", "qcom,mdss-dsi-on-command-state");
1865:     }

```