NOVATEK CONFIDENTIAL

Novatek HDAL Design Specification - hd_debug

Copyright © 2018 Novatek Microelectronics Corp. All Rights Reserved.
With respect to the information represented in this document, Novatek makes no warranty, expressed or implied, including the warranties of merchantability, fitness for a particular purpose and non-infringement, and does not assume any legal liability or responsibility for the accuracy, completeness or usefulness of any such information.

Table of Content

NT9668x/NT98313 Design Specification - hd_debug 1
1 Introduction 3
2 Function and data structure definition. 4
2.1 Data structure definition for hd_debug 4
2.1.1 Structure for Debug function list 4
2.2 Function definition 4
2.2.1 Global function 4
2.3 Software control flow 6
2.3.1 hd_debug control flow 6
2.3.2 HDAL debug menu control flow 6
2.3.3 HDAL debug level message 7
3 Use cases 8
3.1 Example 8

NOVATEK CONFIDENTIAL

Copyright © 2018 Novatek Microelectronics Corp. All Rights Reserved.
With respect to the information represented in this document, Novatek makes no warranty, expressed or implied, including the warranties of merchantability, fitness for a particular purpose and non-infringement, and does not assume any legal liability or responsibility for the accuracy, completeness or usefulness of any such information.

1 Introduction

The major purpose of hd_debug is for HDAL level debugging. hd_debug is responsible for message printing, debug command switching, run module's test flow, and program trace controlling.

The Debug function list is as follow:

```
static HD_DBG_MENU root_menu[] = {
     {0x01, "AUDIOCAPTURE",   hd_audiocap_menu,   TRUE},
     {0x02, "AUDIOOUT",        hd_audioout_menu,   TRUE},
     {0x03, "AUDIOENC",        hd_audioenc_menu,   TRUE},
     {0x04, "AUDIODEC",        hd_audiodec_menu,   TRUE},
     {0x05, "VIDEOCAPTURE",   hd_videocap_menu,   TRUE},
     {0x06, "VIDEOOUT",        hd_videoout_menu,   TRUE},
     {0x07, "VIDEOPROCESS",   hd_videoproc_menu,  TRUE},
     {0x08, "VIDEOENC",        hd_videoenc_menu,   TRUE},
     {0x09, "VIDEODEC",        NULL,               TRUE},
     {0x0A, "OSG",              hd_osg_menu,         TRUE},
     {0x0B, "COMMON",          hd_common_menu,    TRUE},
     {0x0C, "UTIL",              hd_util_menu,         TRUE},
     {0x0D, "DEBUG",            hd_debug_menu,      TRUE},
     // escape must be last
     {HD_DEBUG_MENU_ID_LAST,  "", NULL,           FALSE},
};
```

NOVATEK CONFIDENTIAL

Copyright © 2018 Novatek Microelectronics Corp. All Rights Reserved.
With respect to the information represented in this document, Novatek makes no warranty, expressed or implied, including the warranties of merchantability, fitness for a particular purpose and non-infringement, and does not assume any legal liability or responsibility for the accuracy, completeness or usefulness of any such information.

2 Function and data structure definition.

2.1 Data structure definition for hd_debug

2.1.1 Structure for Debug function list

```
typedef struct _HD_DBG_MENU {
     int           menu_id;          ///< user command value
     const char    *p_name;          ///< command string
     HD_RESULT     (*p_func)(void);  ///< command function
     BOOL          b_enable;         ///< execution option
} HD_DBG_MENU;
```

2.2 Function definition

2.2.1 Global function

1. HD_RESULT hd_debug_init(void);

   Description:

      Initiate the requirement of debug, such as open a file handle to save the log to file

   Param:

      None

   Return value:

      HD_RESULT

2. HD_RESULT hd_debug_get(HD_DEBUG_PARAM_ID idx, void *p_data);

   Description:

      A universal interface to get data from hd_debug

   Param:

      idx : refers HD_DEBUG_PARAM_ID

      p_data : a pointer that address to returned data. The pointer type depends on what HD_DEBUG_PARAM_ID get to that has defined on description of HD_DEBUG_PARAM_ID.

   Return value:

      HD_RESULT

3. HD_RESULT hd_debug_set(HD_DEBUG_PARAM_ID idx, void *p_data);

   Description:

NOVATEK CONFIDENTIAL

Copyright © 2018 Novatek Microelectronics Corp. All Rights Reserved.
With respect to the information represented in this document, Novatek makes no warranty, expressed or implied, including the warranties of merchantability, fitness for a particular purpose and non-infringement, and does not assume any legal liability or responsibility for the accuracy, completeness or usefulness of any such information.

A universal interface to set data into hd_debug

Param:

idx : refers HD_DEBUG_PARAM_ID.

p_data : a pointer that address to returned data. The pointer type depends on what HD_DEBUG_PARAM_ID get to that has defined on description of HD_DEBUG_PARAM_ID.

Return value:

HD_RESULT

4. HD_RESULT hd_debug_uninit(void);

Description:

Un-initiate the debug, there is nothing to do in it, currently

Param:

None

Return value:

HD_RESULT

5. HD_RESULT hd_debug_run_menu(void);

Description:

Display an interactive menu on console and can,

1. Enable or disable message mask of each module.
2. Dump module's information.
3. Run module's test flow.

Param:

None

Return value:

HD_RESULT

NOVATEK CONFIDENTIAL

Copyright © 2018 Novatek Microelectronics Corp. All Rights Reserved.
With respect to the information represented in this document, Novatek makes no warranty, expressed or implied, including the warranties of merchantability, fitness for a particular purpose and non-infringement, and does not assume any legal liability or responsibility for the accuracy, completeness or usefulness of any such information.

2.3 Software control flow

2.3.1 hd_debug control flow

There are two operations in hd_debug: HDAL debug menu, and HDAL level message output.

2.3.2 HDAL debug menu control flow

Software procedures to switch to HDAL debug function.

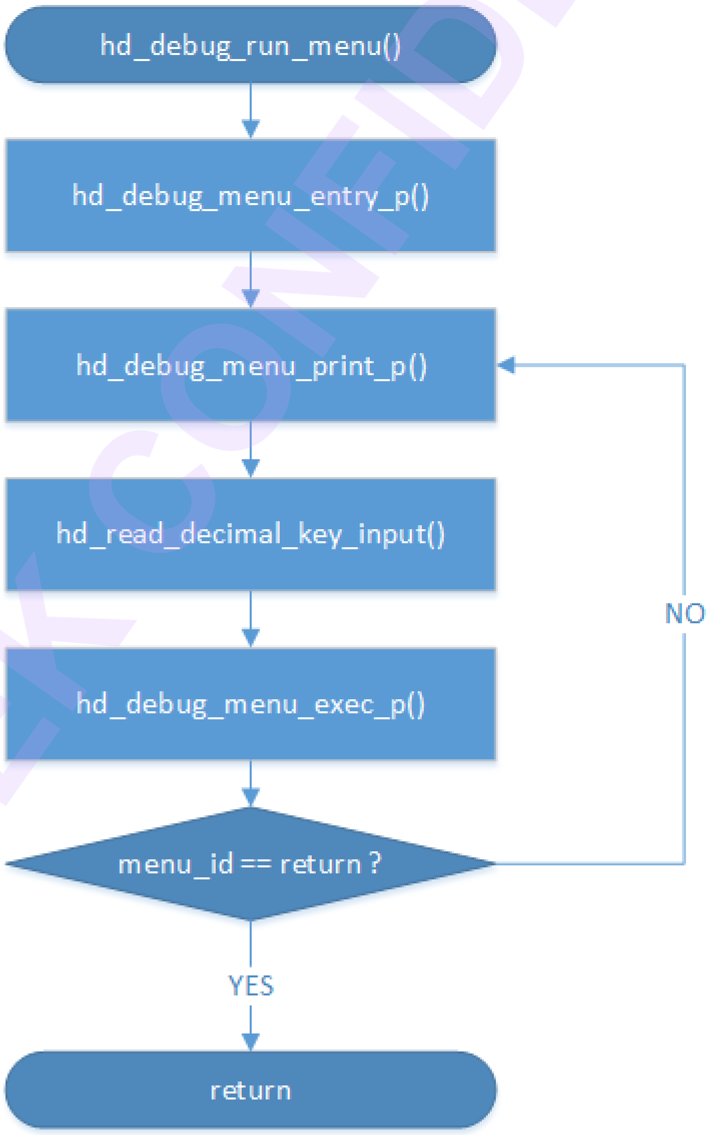

Figure 2-1 flow chart to open and start section filter

NOVATEK CONFIDENTIAL

Copyright © 2018 Novatek Microelectronics Corp. All Rights Reserved.
With respect to the information represented in this document, Novatek makes no warranty, expressed or implied, including the warranties of merchantability, fitness for a particular purpose and non-infringement, and does not assume any legal liability or responsibility for the accuracy, completeness or usefulness of any such information.

2.3.3 HDAL debug level message

Each type of message has its own unsigned integer and each its bit present a HDAL module. For example, a variable named g_hd_mask_err and its LSB indicate the enable or disable of hd_audiocapture's error message.

NOVATEK CONFIDENTIAL

Copyright © 2018 Novatek Microelectronics Corp. All Rights Reserved.
With respect to the information represented in this document, Novatek makes no warranty, expressed or implied, including the warranties of merchantability, fitness for a particular purpose and non-infringement, and does not assume any legal liability or responsibility for the accuracy, completeness or usefulness of any such information.

3 Use cases

3.1 Example

To disable all error level messages on debug menu, do this,

1. Type 13 enter => select 13.DEBUG
2. Type 2 enter => select 2. All ERR mask disable
3. Type 255 enter => return back to upper menu

To disable all error level messages by using API call, do this,

```
unsigned int disable_all = 0;

hd_debug_set(HD_DEBUG_PARAM_ERR_MASK, &disable_all);
```

NOVATEK CONFIDENTIAL

Copyright © 2018 Novatek Microelectronics Corp. All Rights Reserved.
With respect to the information represented in this document, Novatek makes no warranty, expressed or implied, including the warranties of merchantability, fitness for a particular purpose and non-infringement, and does not assume any legal liability or responsibility for the accuracy, completeness or usefulness of any such information.