

SDK6 AN: SD Card Tuning

Draft Version 1.0

August 22, 2014



Confidentiality Notice:

Copyright © 2014 Ambarella Inc.

The contents of this document are proprietary and confidential information of Ambarella Inc.

The material in this document is for information only. Ambarella assumes no responsibility for errors or omissions and reserves the right to change, without notice, product specifications, operating characteristics, packaging, ordering, etc. Ambarella assumes no liability for damage resulting from the use of information contained in this document. All products, brand names, and company names are trademarks of their respective owners.

US

3101 Jay Street
Ste.110
Santa Clara, CA 95054, USA
Phone: +1.408.734.8888
Fax: +1.408.734.0788

Hong Kong

Unit A&B, 18/F, Spectrum Tower
53 Hung To Road
Kwun Tong, Kowloon
Phone: +85.2.2806.8711
Fax: +85.2.2806.8722

Korea

6 Floor, Hanwon-Bldg.
Sunae-Dong, 6-1, Bundang-Gu
SeongNam-City, Kyunggi-Do
Republic of Korea 463-825
Phone: +031.717.2780
Fax: +031.717.2782

China - Shanghai

9th Floor, Park Center
1088 Fangdian Road, Pudong New District
Shanghai 201204, China
Phone: +86.21.6088.0608
Fax: +86.21.6088.0366

Taiwan

Suite C1, No. 1, Li-Hsin Road 1
Science-Based Industrial Park
Hsinchu 30078, Taiwan
Phone: +886.3.666.8828
Fax: +886.3.666.1282

Japan - Yokohama

Shin-Yokohama Business Center Bldg. 5th Floor
3-2-6 Shin-Yokohama, Kohoku-ku,
Yokohama, Kanagawa, 222-0033, Japan
Phone: +81.45.548.6150
Fax: +81.45.548.6151

China - Shenzhen

Unit E, 5th Floor
No. 2 Finance Base
8 Ke Fa Road
Shenzhen, 518057, China
Phone: +86.755.3301.0366
Fax: +86.755.3301.0966

I Contents

II	Preface	ii
1	Overview	1
1.1	Overview: Introduction of SD Tuning Tool	1
2	Choosing Proper SD Tuning Parameter from the SD Shmoo Result	3
2.1	Choosing Proper SD Tuning Parameter from SD Shmoo Result: Overview ..	3
3	Adding of API Functions to Apply SD Tuning Parameter	4
3.1	Adding API Functions to Apply SD Tuning Parameter: Overview	4
Appendix 1	Additional Resources	A1
Appendix 2	Important Notice	A2
Appendix 3	Revision History	A3

II Preface

This document provides technical details using a set of consistent typographical conventions to help the user differentiate key concepts at a glance.

Conventions include:

Example	Description
AmbaGuiGen, DirectUSB Save, File > Save Power, Reset, Home	Software names GUI commands and command sequences Computer / Hardware buttons
Flash_IO_control da, status, enable	Register names and register fields. For example, Flash_IO_control is the register for global control of Flash I/O, and bit 17 (da) is used for DMA acknowledgement.
GPIO81, CLK_AU	Hardware external pins
VIL, VIH, VOL, VOH	Hardware pin parameters
INT_O, RXDATA_I	Hardware pin signals
amb_performance_t amb_operating_mode_t amb_set_operating_mode()	API details (e.g., functions, structures, and type definitions)
/usr/local/bin success = amb_set_operating_ mode (amb_hal_base_address, & operating_mode)	User entries into software dialogues and GUI windows File names and paths Command line scripting and Code

Table II-1. *Typographical Conventions for Technical Documents.*

Additional Ambarella typographical conventions include:

- Acronyms are given in UPPER CASE using the default font (e.g., AHB, ARM11 and DDRIO).
- Names of Ambarella documents and publicly available standards, specifications, and databooks appear in *italic* type.

1 Overview

This document provides information on SD card tuning.

1.1 Overview: Introduction of SD Tuning Tool

1. Use the shell command “**t sd_shmoo**” and the system would show the usage as below:

```
a:\> t sd_shmoo
Usage: sd_shmoo [slot] all
Usage: sd_shmoo [slot] [all_str|3mA|6mA|12mA|18mA|] [set|all_dly] [8 delay value]
all - test all the driving strength & all the delay ctrl settings
all_str - test all the driving strength
3mA|6mA|12mA|18mA - choose one of driving strength
all_dly - test all the delay ctrl settings

Example : t sd_shmoo c 12mA all_dly

set - set 8 delay ctrl settings of start(0-7) & count(1-8) as following order
[data_output_delay_start data_output_delay_count
data_input_delay_start data_input_delay_count
clk_output_delay_start clk_output_delay_count
clk_input_delay_start clk_input_delay_count]

Example : t sd_shmoo c 12mA set 0 1 1 1 1 1 0 8
```

Figure 1-1. The Usage of Shell Command “**t sd_shmoo**” from the UART Log.

2. Insert the SD card, and key in the sd shmoo command with proper settings.

3. The system performs the shmoo test and shows the result on UART log as given below:

```
a:\> t sd_shmoo c 12mA all_dly
--- driving strength : 12mA ---
[Success] data_output_delay : 0, clk_output_delay : 0, data_input_delay : 0, clk_input_delay : 0
[Success] data_output_delay : 0, clk_output_delay : 0, data_input_delay : 0, clk_input_delay : 1
[Success] data_output_delay : 0, clk_output_delay : 0, data_input_delay : 0, clk_input_delay : 2
[Success] data_output_delay : 0, clk_output_delay : 0, data_input_delay : 0, clk_input_delay : 3
[Success] data_output_delay : 0, clk_output_delay : 0, data_input_delay : 0, clk_input_delay : 4
[Success] data_output_delay : 0, clk_output_delay : 0, data_input_delay : 0, clk_input_delay : 5
[Failed (-1)] data_output_delay : 0, clk_output_delay : 0, data_input_delay : 0, clk_input_delay : 6
[Failed (-1)] data_output_delay : 0, clk_output_delay : 0, data_input_delay : 0, clk_input_delay : 7
[Success] data_output_delay : 0, clk_output_delay : 0, data_input_delay : 1, clk_input_delay : 0
[Success] data_output_delay : 0, clk_output_delay : 0, data_input_delay : 1, clk_input_delay : 1
...
[Success] data_output_delay : 7, clk_output_delay : 7, data_input_delay : 7, clk_input_delay : 5
[Failed (-1)] data_output_delay : 7, clk_output_delay : 7, data_input_delay : 7, clk_input_delay : 6
[Failed (-1)] data_output_delay : 7, clk_output_delay : 7, data_input_delay : 7, clk_input_delay : 7
[sd_shmoo] set back to default driving strength & delay ctrl setting to init card
[sd_shmoo] save sd_shmoo.txt to sd card
[sd_shmoo] test sd shmoo completed!
```

Figure 1-2. The UART Log of the Shell Command “t sd_shmoo c 12mA all_dly”.

4. The result of successful SD driving strength and SD delay control settings would be saved as “sd_shmoo.txt” on the SD card.

```
(c:\sd_shmoo.txt)
--- driving strength : 12mA ---
[Success] data_output_delay : 0, clk_output_delay : 1, data_input_delay : 1, clk_input_delay : 0
[Success] data_output_delay : 0, clk_output_delay : 1, data_input_delay : 1, clk_input_delay : 1
[Success] data_output_delay : 0, clk_output_delay : 1, data_input_delay : 1, clk_input_delay : 2
[Success] data_output_delay : 0, clk_output_delay : 1, data_input_delay : 1, clk_input_delay : 3
[Success] data_output_delay : 0, clk_output_delay : 1, data_input_delay : 1, clk_input_delay : 4
[Success] data_output_delay : 0, clk_output_delay : 1, data_input_delay : 1, clk_input_delay : 5
```

Figure 1-3. The Successful Results of sd shmoo Test (saved as “sd_shmoo.txt”).

2 Choosing Proper SD Tuning Parameter from the SD Shmoo Result

2.1 Choosing Proper SD Tuning Parameter from SD Shmoo Result: Overview

This chapter provides the steps for choosing proper SD tuning parameters from the SD Shmoo result.

1. Use at least 5 different SD cards to perform SD shmoo test. (including UHS-I SD cards)
2. Compare the SD shmoo test results to find the common successful parameter settings.
3. If there are many common successful parameter settings, please choose the setting closest to the middle value as the golden SD tuning parameter. (Please refer to the following example)

The common parameter settings by 10 SD cards is provided as below:

--- driving strength : 12mA ---

[Success] data_output_delay : 1, clk_output_delay : 1, data_input_delay : 1, clk_input_delay : 2
[Success] data_output_delay : 1, clk_output_delay : 1, data_input_delay : 2, clk_input_delay : 3
[Success] data_output_delay : 1, clk_output_delay : 1, data_input_delay : 3, clk_input_delay : 4
[Success] data_output_delay : 1, clk_output_delay : 1, data_input_delay : 4, clk_input_delay : 5
[Success] data_output_delay : 1, clk_output_delay : 1, data_input_delay : 5, clk_input_delay : 6
[Success] data_output_delay : 1, clk_output_delay : 2, data_input_delay : 1, clk_input_delay : 0

The user can choose the setting in red as the golden SD tuning parameter.

4. For more SD tuning tests, the user can use the shell command to set specific delay control range. Please check the following example:

```
a:\> t sd_shmoo c 12mA set 0 1 1 1 2 2 0 5
data_output_delay_start : 0, data_output_delay_count : 1
data_input_delay_start : 1, data_input_delay_count : 1
clk_output_delay_start : 2, clk_output_delay_count : 2
clk_input_delay_start : 0, clk_intput_delay_count : 5
--- driving strength : 12mA ---
[Success] data_output_delay : 0, clk_output_delay : 1, data_input_delay : 2, clk_input_delay : 0
[Success] data_output_delay : 0, clk_output_delay : 1, data_input_delay : 2, clk_input_delay : 1
[Success] data_output_delay : 0, clk_output_delay : 1, data_input_delay : 2, clk_input_delay : 2
[Failed (-1)] data_output_delay : 0, clk_output_delay : 1, data_input_delay : 2, clk_input_delay : 3
[Failed (-1)] data_output_delay : 0, clk_output_delay : 1, data_input_delay : 2, clk_input_delay : 4
[Success] data_output_delay : 0, clk_output_delay : 1, data_input_delay : 3, clk_input_delay : 0
[Success] data_output_delay : 0, clk_output_delay : 1, data_input_delay : 3, clk_input_delay : 1
[Success] data_output_delay : 0, clk_output_delay : 1, data_input_delay : 3, clk_input_delay : 2
[Failed (-1)] data_output_delay : 0, clk_output_delay : 1, data_input_delay : 3, clk_input_delay : 3
[Failed (-1)] data_output_delay : 0, clk_output_delay : 1, data_input_delay : 3, clk_input_delay : 4
...
```

Figure 2-1. The UART Log of Shell Command “t sd_shmoo c 12mA set 0 1 1 1 2 2 0 5”.

3 Adding of API Functions to Apply SD Tuning Parameters

3.1 Adding API Functions to Apply SD Tuning Parameters: Overview

This chapter provides various methods of adding API functions to apply SD tuning parameters.

API Function to Set the SD Driving Strength:

API Syntax:

```
void AmbaSD_SetDrivingStrength (int Id, AMBA_SD_DRIVING_STRENGTH_TYPE_e Type,
AMBA_SD_DRIVING_STRENGTH_VALUE_e Driving);
```

Function Description:

This API is used to set the SD driving strength.

Parameters:

Type	Parameter	Description
int	Id	0: SD_HOST_0 1: SD_HOST_1
AMBA_SD_DRIVING_STRENGTH_TYPE_e	Type	AMBA_SD_DRIVING_STRENGTH_CLK AMBA_SD_DRIVING_STRENGTH_DATA AMBA_SD_DRIVING_STRENGTH_CMD AMBA_SD_DRIVING_STRENGTH_CDWP AMBA_SD_DRIVING_STRENGTH_ALL
AMBA_SD_DRIVING_STRENGTH_VALUE_e	Driving	AMBA_SD_DRIVING_STRENGTH_CLK AMBA_SD_DRIVING_STRENGTH_DATA AMBA_SD_DRIVING_STRENGTH_CMD AMBA_SD_DRIVING_STRENGTH_CDWP AMBA_SD_DRIVING_STRENGTH_ALL

Table 3-1. Parameters for API Functions *AmbaSD_SetDrivingStrength()*.

Returns:

None

API Function to Set the SD Delay Control:

API Syntax:

void **AmbaSD_SetDelayControl** (int Id, AMBA_SD_DELAY_CTRL_TYPE_e Type, UINT8 DelayValue);

Function Description:

This API is used to set the SD delay control.

Parameters:

Type	Parameter	Description
int	Id	0: SD_HOST_0 1: SD_HOST_1
AMBA_SD_DELAY_CTRL_TYPE_e	Type	AMBA_SD_CLK_INPUT_DELAY AMBA_SD_DATA_INPUT_DELAY AMBA_SD_CLK_OUTPUT_DELAY AMBA_SD_DATA_OUTPUT_DELAY
UINT8	DelayValue	8 different delay control settings: 0, 1, 2, 3, 4, 5, 6, 7

Table 3-2. Parameters for API Functions **AmbaSD_SetDelayControl()**.

Returns:

None

Apply the Golden SD Tuning Parameters by API Functions to Ambarella A9.

(Please refer to AmbaRelease\A9\Workshop\A9EVK\src\AmbaUserSysCtrl.c)

Example: driving strength: 12mA
data_output_delay: 1, clk_output_delay: 1, data_input_delay: 3, clk_input_delay: 4

```
#include "AmbaSD.h"
...
void AmbaUserSysInitPostOS(void) {
    ...
    AmbaSD_SetDrivingStrength(SD_HOST_0, AMBA_SD_DRIVING_STRENGTH_ALL,
                              AMBA_SD_DRIVING_STRENGTH_12MA);
    AmbaSD_SetDelayControl(SD_HOST_0, AMBA_SD_DATA_OUTPUT_DELAY, 1);
    AmbaSD_SetDelayControl(SD_HOST_0, AMBA_SD_CLK_OUTPUT_DELAY, 1);
    AmbaSD_SetDelayControl(SD_HOST_0, AMBA_SD_DATA_INPUT_DELAY, 3);
    AmbaSD_SetDelayControl(SD_HOST_0, AMBA_SD_CLK_INPUT_DELAY, 4);

    AmbaSD_Init(1, SD_UHS_MAX_FREQ); /* SD Initializations */
    ...
}
```

Figure 3-1. The Example to Set SD Driving Strength and SD Delay Control.

[Note: SD driving strength and SD delay control setting should be applied before **AmbaSD_Init()**]

Appendix 1 Additional Resources

Please contact an Ambarella representative for digital copies.

Confidential
For PROTRULY Only

Appendix 2 Important Notice

All Ambarella design specifications, datasheets, drawings, files, and other documents (together and separately, “materials”) are provided on an “as is” basis, and Ambarella makes no warranties, expressed, implied, statutory, or otherwise with respect to the materials, and expressly disclaims all implied warranties of noninfringement, merchantability, and fitness for a particular purpose. The information contained herein is believed to be accurate and reliable. However, Ambarella assumes no responsibility for the consequences of use of such information.

Ambarella Incorporated reserves the right to correct, modify, enhance, improve, and otherwise change its products and services at any time and to discontinue any product or service without notice. Customers should obtain the latest relevant information before placing orders and should verify that such information is current and complete.

All products are sold subject to Ambarella’s terms and conditions of sale supplied at the time of order acknowledgment. Ambarella warrants performance of its hardware products to the specifications applicable at the time of sale in accordance with its standard warranty. Testing and other quality control techniques are used to the extent Ambarella deems necessary to support this warranty.

Ambarella assumes no liability for applications assistance or customer product design. Customers are responsible for their products and applications using Ambarella components. To minimize the risks associated with customer products and applications, customers should provide adequate design and operating safeguards.

Ambarella does not warrant or represent that any license, either expressed or implied, is granted under any Ambarella patent right, copyright, mask work right, or other intellectual property right relating to any combination, machine, or process in which Ambarella products or services are used. Information published by Ambarella regarding third-party products or services does not constitute a license from Ambarella to use such products or services or a warranty or endorsement thereof. Use of such information may require a license from a third party under the patents or other intellectual property of the third party, or a license from Ambarella under the patents or other intellectual property of Ambarella.

Reproduction of information from Ambarella documents is not permissible without prior approval from Ambarella.

Ambarella products are not authorized for use in safety-critical applications (such as life support) where a failure of the product would reasonably be expected to cause severe personal injury or death, unless officers of the parties have executed an agreement specifically governing such use. Customers acknowledge and agree that they are solely responsible for all legal, regulatory and safety-related requirements concerning their products and any use of Ambarella products in such safety-critical applications, notwithstanding any applications-related information or support that may be provided by Ambarella. Further, Customers must fully indemnify Ambarella and its representatives against any damages arising out of the use of Ambarella products in such safety-critical applications.

Ambarella products are neither designed nor intended for use in automotive and military/aerospace applications or environments. Customers acknowledge and agree that any such use of Ambarella products is solely at the Customer’s risk, and they are solely responsible for compliance with all legal and regulatory requirements in connection with such use.

Appendix 3 Revision History

NOTE: Page numbers for previous drafts may differ from page numbers in the current version.

Version	Date	Comments
1.0	22 August 2014	Formatting.

Table A3-1. Revision History.

Confidential
For PROTRULY Only