KAS03(4)XXD

KOWIN eMMC(4/8GB) Products

eMMC 5.1 Specification compatibility

Datasheet

KOWIN RESERVES THE RIGHT TO CHANGE PRODUCTS INFORMATION AND SPECIFICATION WITHOUT ANY NOTICE.

Products and specification discussed herein are for reference purpose only. All information discussed herein is proven on an “AS IS” basis without warranties of any kind.

This documents and all information does not constitute any license, express or implied, by estoppel or otherwise, to any such patents, trademarks, copyrights, or other intellectual property rights.

KOWIN products are not intended for use in medical, life saving, life sustaining, critical control or safety systems, or in nuclear facility or similar applications where product failure could result in loss of life or personal or physical harm, or any military or defense application, or governmental procurement to which special items or provisions may apply.

KOWIN assumes no responsibility or liability for any errors or inaccuracies that may appear in this document that may be provided in association with this document. Except as permitted by such license, no part of this document may be reproduced, stored in a retrieval system, or transmitted in any form or by any means without the express written consent of KOWIN.

Contact your nearest KOWIN sales office or your distributor to obtain the latest specifications and before placing your product order.

Revision History

Revision No.	History	Draft Date	Remark	Editor
0.0	Initial Issue	Feb.2020	Draft Version	Brian. Kim
1.0	First Version	Oct.2020	First Version	Barry. Qiu

IF THERE IS ANY OTHER OPERATION TO IMPLEMENT IN ADDITION TO SPECIFICATION IN DATASHEET OR JEDEC STANDARD, PLEASE CONTATC EACH NEAREST SALES OFFICE OR HEADQUARTERS OF KOWIN

Table of Contents

IF THERE IS ANY OTHER OPERATION TO IMPLEMENT IN ADDITION TO SPECIFICATION IN DATASHEET OR JEDEC STANDARD, PLEASE CONTATC EACH NEAREST SALES OFFICE OR HEADQUARTERS OF KOWIN

INTRODUCTION

The KOWIN eMMC is an embedded MMC solution designed in a BGA package type form and a highly integrated solution which combines a feature-wise flash controller and NAND flash memory. Its high performance and low power make the KOWIN eMMC a fabulous solution for embedded and portable applications.

The KOWIN eMMC leverages industry leading technology and experience in NAND management. In addition, the KOWIN eMMC supports the standard eMMC interface as well as the newly introduced eMMC features such as HS400 mode and FFU. By integrating all the advanced techniques, the KOWIN eMMC is able to further enhance the data transferring efficiency and optimizes the overall performance for embedded systems.

Available in various densities, the KOWIN eMMC offers the features, performance, and flexibility exactly for mobile handset, navigation, multi-function printer, and next-generation consumer applications.

1.0 PRODUCT LIST

[Table 1] Product List

Capacity	Part ID	Nand	User Density	Power System	Pin Configuration	Package size
4GB	KAS0411D	32Gb x 1	89%	- Interface power : VCCQ(1.70V~1.95 V or 2.7V ~ 3.6V) - Memory power: VCC(2.7V~3.6V)	153FBGA	11.5mm x 13mm x 1.0mm
8GB	KAS0311D	64Gb x1	90%			

IF THERE IS ANY OTHER OPERATION TO IMPLEMENT IN ADDITION TO SPECIFICATION IN DATASHEET OR JEDEC STANDARD, PLEASE CONTATC EACH NEAREST SALES OFFICE OR HEADQUARTERS OF KOWIN

2.0 KEY FEATUERS

- Industrial Standard Interface : JEDEC / eMMC Standard Version 5.1 Compliant
- eMMC 5.1 Enhanced Features
  - 11-wire bus(clock,Data Strobe,1 bit command,8 bit data bus) and a hardware reset
  - Programmable bus width: 1-bit, 4-bit, and 8-bit
  - Supports HS400 high speed interface timing mode up to 400MB/s data rate
  - Up to 200MHz clock frequency
  - Supports eMMC Field firmware update (FFU)
  - Supports eMMC production state awareness (PSA)
  - Supports eMMC device health report
  - High-speed, Dual Data Rate Boot support
  - Supports Boot and Alternative Boot Mode
  - Replay Protected Memory Block (RPMB)
  - Trim, Sanitize, Discard, Secure Erase, Secure Trim
  - Enhanced Partition Attributes
  - High Priority Interrupt (HPI)
  - Background Operations
  - Enhanced Reliable Write
  - Supports Command Queuing
  - Supports Enhanced Strobe in HS400 Mode
  - Supports eMMC Background Operation Control
- Robust Data Protection and Endurance
  - Configurable BCH ECC engine with zero overhead pipeline greatly reduces data loss rates and increases data endurance
  - Enhanced Write Protection with Permanent, Temporary and Power-On protection options
  - Read disturbance protection technologies ensure the data reliability
  - Sudden power-off protection and embedded voltage detection support power-down data protection
  - Global wear leveling maximizes product lifespan with minimal wear leveling and write amplification overhead
- Supply Voltage
  - eMMC Interface Power(VCCQ) : 1.70–1.95V or 2.7–3.6V [1)]
  - NAND Memory Power(VCC) : 2.7–3.6V
- Dynamic power management technology enables multiple power saving modes
- Multiple Densities and Packages
  - Available in 4GB, 8GB
  - 153-ball standard BGA packages
  - Green Package and RoHS Compliant
- Temperature : Operation(-40°C ~ +85°C) , Storage without operation(-40°C ~ 85°C)

[Note] 1) HS200 and HS400 mode are not supported when VCCQ is in 2.7–3.6V.

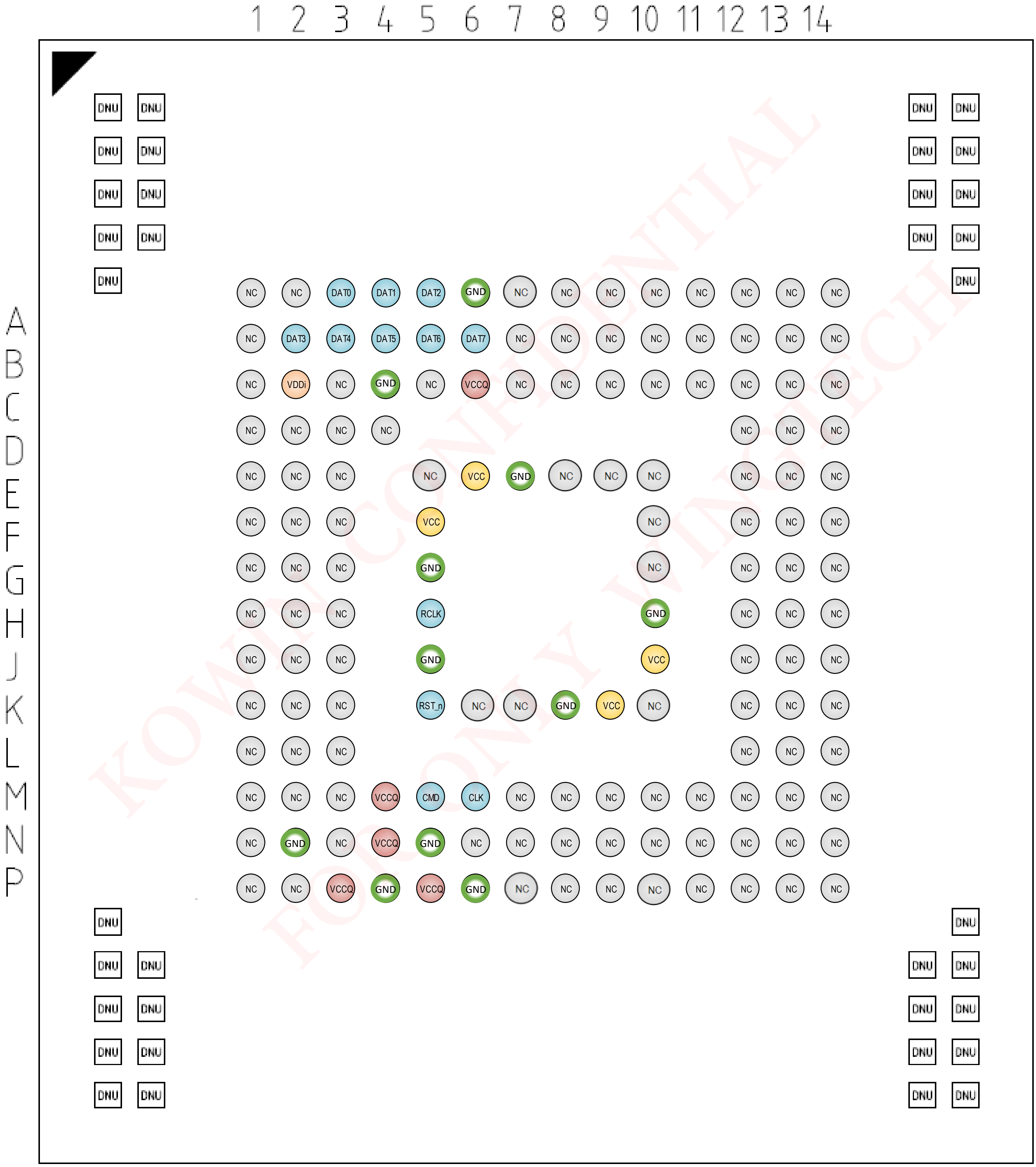
2.1 eMMC New Features List

No	Function	e•MMC	Implement (KOWIN)	Remark (eMMC Ver)
1	Boot	Mandatory	Yes	
2	RPMB	Mandatory	Yes	
3	Write Protection(including Perm & Temp)	Mandatory	Yes	
4	1.2 V I/O	Optional	No	
5	Dual Data Rate timing	Optional	Yes	
6	HS200	Optional	Yes	
7	Multi Partitioning	Mandatory	Yes	
8	Secure Erase/Secure Trim	Optional	Yes	
9	Trim	Mandatory	Yes	
10	High Priority Interrupt	Mandatory	Yes	
11	Background Operation	Mandatory	Yes	
12	Enhance Reliable Write	Mandatory	Yes	
13	Discard Command	Mandatory	Yes	
14	Security Features	Mandatory	Yes	
15	Partition types	Mandatory	Yes	
16	Context ID	Mandatory	Yes	
17	Data Tag	Mandatory	Yes	
18	Packed commands	Mandatory	Yes	
19	Real Time Clock	Mandatory	Yes	
20	Dynamic Device Capacity	Mandatory	Yes	
21	Power Off Notification	Mandatory	Yes	
22	Thermal Spec	Mandatory	Yes	
23	Minimum Sector Size = 4 KB (≤256 GB)	Optional	Yes	
24	Minimum Sector Size = 4 KB (>256 GB)	Mandatory	Yes	
25	Cache	Optional	Yes	
26	Extended Security Protocols	Optional	No	
27	HS400	Optional	Yes	
28	Field Firmware Update	Optional	Yes	
29	Product State Awareness	Optional	Yes	
30	Secure Removal Type	Optional	Yes	
31	Device Health Report	Optional	Yes	
32	Command Queuing	Optional	Yes	Ver. 5.1
33	Enhanced Strobe	Optional	Yes	Ver.5.1
34	Cache Flushing Report	Mandatory	Yes	Ver. 5.1
35	BKOPS Control	Mandatory	Yes	Ver. 5.1
36	Cache Barrier	Optional	Yes	Ver. 5.1
37	RPMB Throughput Improve	Optional	Yes	Ver. 5.1
38	Secure Write Protection	Optional	Yes	Ver. 5.1

IF THERE IS ANY OTHER OPERATION TO IMPLEMENT IN ADDITION TO SPECIFICATION IN DATASHEET OR JEDEC STANDARD, PLEASE CONTATC EACH NEAREST SALES OFFICE OR HEADQUARTERS OF KOWIN

3.0 PACKAGE CONFIGURATION

3.1 153 Ball Pin Configuration



	1	2	3	4	5	6	7	8	9	10	11	12	13	14
A	NC	NC	DAT0	DAT1	DAT2	GND	NC	NC	NC	NC	NC	NC	NC	NC
B	NC	DAT3	DAT4	DAT5	DAT6	DAT7	NC	NC	NC	NC	NC	NC	NC	NC
C	NC	VDDi	NC	GND	NC	VCCQ	NC	NC	NC	NC	NC	NC	NC	NC
D	NC	NC	NC	NC								NC	NC	NC
E	NC	NC	NC		NC	VCC	GND	NC	NC	NC		NC	NC	NC
F	NC	NC	NC		VCC					NC		NC	NC	NC
G	NC	NC	NC		GND					NC		NC	NC	NC
H	NC	NC	NC		RCLK					GND		NC	NC	NC
J	NC	NC	NC		GND					VCC		NC	NC	NC
K	NC	NC	NC		RST_n	NC	NC	GND	VCC	NC		NC	NC	NC
L	NC	NC	NC									NC	NC	NC
M	NC	NC	NC	VCCQ	CMD	CLK	NC	NC	NC	NC	NC	NC	NC	NC
N	NC	GND	NC	VCCQ	GND	NC	NC	NC	NC	NC	NC	NC	NC	NC
P	NC	NC	VCCQ	GND	VCCQ	GND	NC	NC	NC	NC	NC	NC	NC	NC

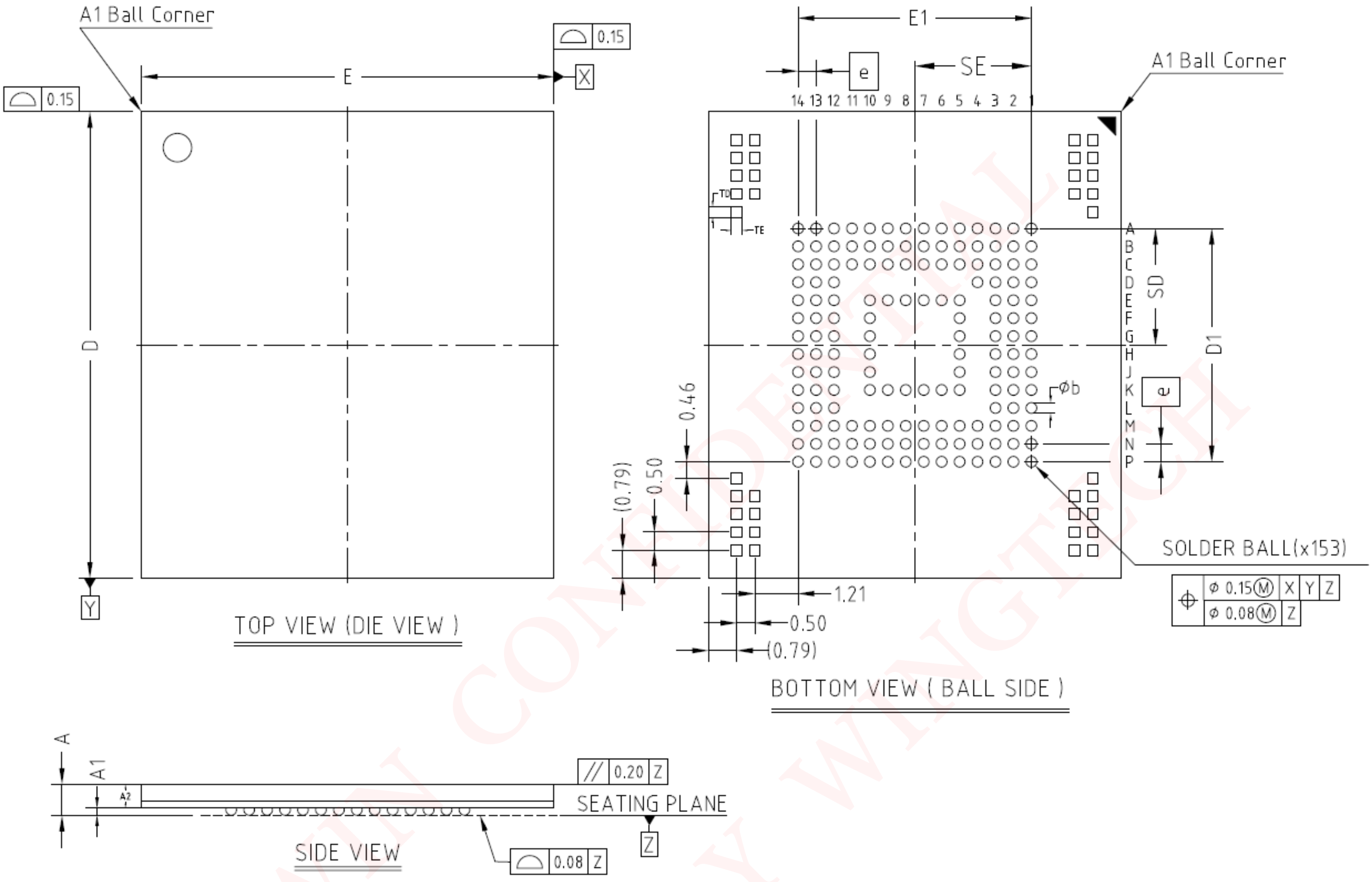
IF THERE IS ANY OTHER OPERATION TO IMPLEMENT IN ADDITION TO SPECIFICATION IN DATASHEET OR JEDEC STANDARD, PLEASE CONTATC EACH NEAREST SALES OFFICE OR HEADQUARTERS OF KOWIN

[Table 3] Signal Descriptions (Functional Signals)

Signal	Type	Description
CLK	Input	Clock. Each cycle of the clock directs a transfer on the command line and on the data line(s). The frequency can vary between the minimum and the maximum clock frequency.
RCLK	Output	eMMC interface data strobe (HS400 mode)
CMD	I/O	Command. This signal is a bidirectional command channel used for command and response trans fers. The CMD signal has two bus modes: open-drain mode and push-pull mode. Com mands are sent from the MMC host to the device, and responses are sent from the d evice to the host.
DAT0 – DAT7	I/O	Data I/O. These are bidirectional data signals. The DAT signals operate in push-pull mode. By default, after power-up or assertion of the RST_n signal, only DAT0 is used for dat a transfer. The MMC controller can configure a wider data bus for data transfer using either DAT[3:0] (4-bit mode) or DAT[7:0] (8-bit mode). eMMC includes internal pull-up resistors for data lines DAT[7:1]. Immediately after e ntering the 4-bit mode, the device disconnects the internal pull-up resistors on the D AT[3:1] lines. Correspondingly, immediately after entering the 8-bit mode, the device disconnects the internal pull-ups on the DAT[7:1] lines.
RST_n	Input	Reset. The RST_n signal is used for host resetting device, moving the device to pre-idle state . By default, the RST_n signal is temporary disabled in device. The host must set bits[1:0] in the extended CSD register [162] to 0x1 to enable this functionality before the host can use it.
VCC	Supply	NAND interface I/O and NAND Flash power supply.
VCCQ	Supply	eMMC controller core and eMMC interface I/O power supply.
GND	Supply	NAND interface I/O and NAND Flash ground connection. eMMC controller core and eMMC interface ground connection.
VDDi	-	Internal voltage node A 1.0uF capacitor is required for VDDi for core power stabilization. Do not tie to supply voltage or ground.

IF THERE IS ANY OTHER OPERATION TO IMPLEMENT IN ADDITION TO SPECIFICATION IN DATASHEET OR JEDEC STANDARD, PLEASE CONTATC EACH NEAREST SALES OFFICE OR HEADQUARTERS OF KOWIN

3.1.2 11.5mm x 13mm x 1.0mm Package Dimension



[Figure 3] 11.5mm x 13mm x 1.0mm Package Dimension

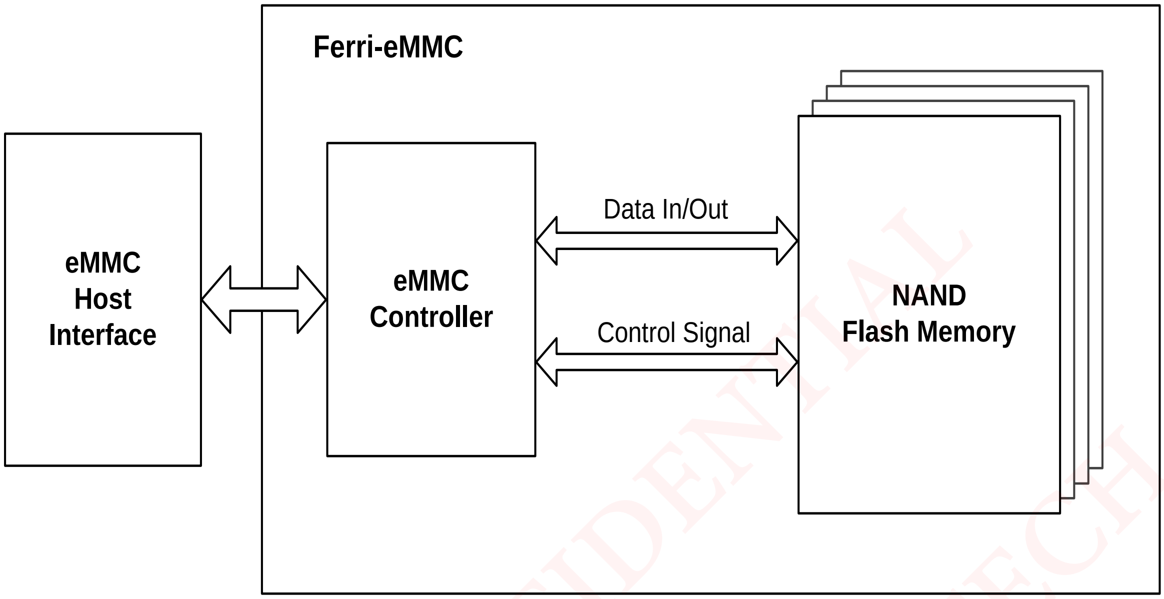
Symbol	MIN	NOM	MAX	Symbol	MIN	NOM	MAX
A	---	---	1.00	TD	0.25	0.30	0.35
A1	0.15	0.20	0.25	TE	0.25	0.30	0.35
A2	---	0.73	---	D1	---	6.50	---
b	0.25	0.30	0.35	E1	---	6.50	---
D	12.90	13.00	13.10	SD	---	3.25	---
E	11.40	11.50	11.60	SE	---	3.25	---
e	---	0.50	---				

Notes:

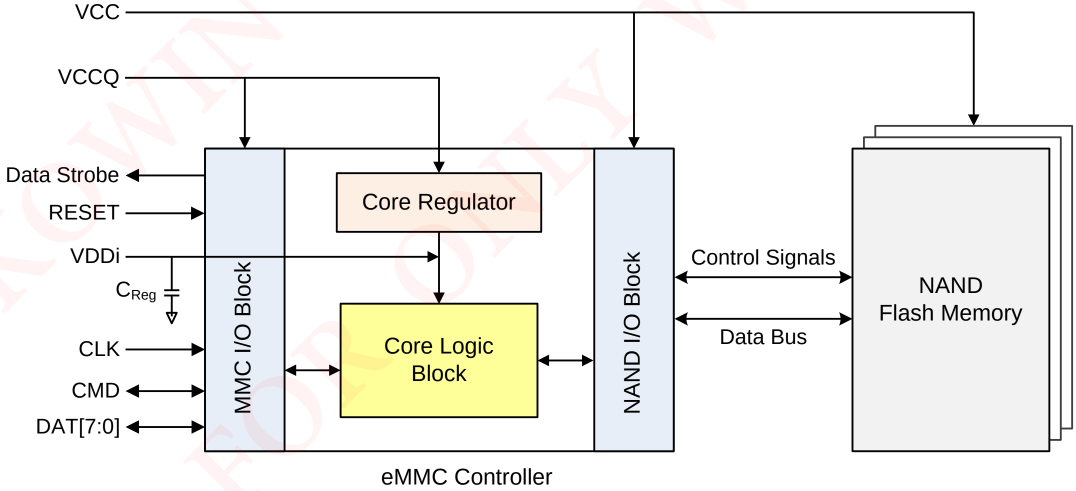
1. **Controlling dimension: Millimeter.**
2. **Primary datum c and seating plane are defined by the spherical crowns of the solder balls.**
3. **Dimension b is measured at the maximum solder ball diameter, parallel to primary datum C.**
4. **Special characteristics C class: bbb, ddd.**
5. **The pattern of Pin 1 fiducial is for reference only.**

IF THERE IS ANY OTHER OPERATION TO IMPLEMENT IN ADDITION TO SPECIFICATION IN DATASHEET OR JEDEC STANDARD, PLEASE CONTATC EACH NEAREST SALES OFFICE OR HEADQUARTERS OF KOWIN

3.2 eMMC Block Diagram



[Figure 4] eMMC Block Diagram



- eMMC consist of NAND Flash and Controller. VCCQ is for Controller power and VCC is for flash power.
- A C_{Reg} capacitor must be connected to the VDDi terminal to stabilize regulator output on the system (See. 8.4 Bus Signal Line Load)

[Figure 5] eMMC Block Diagram(Internal Power Diagram)

IF THERE IS ANY OTHER OPERATION TO IMPLEMENT IN ADDITION TO SPECIFICATION IN DATASHEET OR JEDEC STANDARD, PLEASE CONTATC EACH NEAREST SALES OFFICE OR HEADQUARTERS OF KOWIN

4.0 eMMC 5.1 Features

4.1 Command Queuing.

To facilitate command queuing in eMMC, the device manages an internal task queue that the host can queue data transfer tasks.

Initially the task queue is empty. Every task is issued by the host and initially queued as pending. The device controller works to prepare pending tasks for execution. When a task is ready for execution its state changes to “ready for execution”. The exact meaning of “ready for execution” is left for device implementation.

The host tracks the state of all queued tasks and may order the execution of any task, that is marked as “ready for execution”, by sending a command indicating its task ID. When the execute command is received (CMD46/CMD47) the device executes the data transfer transaction.

For example, in order to queue a write transaction the host sends a CMD44 indicating the task’s parameters. The device responds and the host sends a CMD45, indicating the start block address.

The device regards the two commands as a single task in the queue and sends a response indicating success if no error is detected. This exchange may be executed on the CMD line while a data transfer, or busy state, is ongoing on the DAT lines. The host tracks the state of the queue using CMD13.

At a later time, when data transfer is not in progress, the host issues a CMD47, ordering the device to execute a task from the queue, providing the Task ID in its argument. The device responds with an R1 response and the data transfer starts.

NOTE If hosts need to access RPMB partition, the host should disable the Command Queue mechanism and access RPMB partition not through the command queue.

General Purpose partitions may be accessed when command queuing is enabled. The queue must be empty when CMD6 is sent (to switch partitions or to disable command queuing). Sending CMD6 while the queue is not empty shall be regarded as illegal command (see 4.1.8 Supported Commands).

Prior to enabling command queuing, the block size shall be set to 512 B. Device may respond with an error to CMD46/CMD47 if block size is not 512 B.

The device does not guarantee the order that queued tasks are processed. In cases where ordering is important (e.g., commands with overlapping LBAs), the host is responsible to ensure it.

IF THERE IS ANY OTHER OPERATION TO IMPLEMENT IN ADDITION TO SPECIFICATION IN DATASHEET OR JEDEC STANDARD, PLEASE CONTATC EACH NEAREST SALES OFFICE OR HEADQUARTERS OF KOWIN

4.1.1 QUEUED_TASK_PARAMS - CMD44

CMD44 is the first step in queuing a data transfer task. The command encodes parameters that are necessary for queuing the task and executing the operation. The arguments are: Block Count, Task ID, Priority, Data Direction, and additional parameters (i.e., Tag Request, Context ID, Reliable Write, Forced Programming).

The encoded Task ID must not be already in use by another task in the queue. A Task ID is considered available for reuse if the task identified by it is completed. A data read task is considered completed when the last data block has been fully transferred over the eMMC bus. A data write task is considered completed when the busy signal following the last data block is released.

There are 2 priority levels: high priority tasks and simple priority tasks. The device is required to give priority to queued high-priority tasks by making the preparations for their execution before it prepares simple tasks. As such, a high priority task should normally transfer to "ready for execution" state before simple tasks that were queued before it. However, it is acceptable that, at the time of receiving the high-priority task request, the device may have started preparing simple tasks for execution, but has not marked them as "ready for execution" in the QSR. The device may mark such tasks as "ready for execution" before the new high-priority task.

The handling of error cases related to CMD44 is listed in Table 5.
CMD44 should always be followed by a CMD45 to complete the queuing operation. If the next command issued by the host is not CMD45, or CMD45 is issued but an error condition is detected, device behavior is undefined and host is expected to resend CMD44 and CMD45.
CMD44 may be issued while data transfer is on-going on the DAT lines, when the lines are idle (e.g., during NAC time) or when the device indicates BUSY state, except when the device is BUSY due to execution of CMD48.

4.1.2 QUEUED_TASK_ADDRESS - CMD45

CMD45 shall be issued directly after (and only after) CMD44. The argument of a CMD45 is the start block address for the related transaction (similar to CMD18, for example). When receiving CMD45, the device queues the task with the following parameters (extracted from the two commands CMD44 and CMD45): Task ID, Block Count, Block Address, Priority, Data Direction, Tag Request, Context ID, Reliable Write, and Forced Programming.

The handling of error cases related to CMD45 is listed in Table 5.
CMD45 may be issued while data transfer is on-going on the DAT lines or when the device indicates BUSY state or when the lines are idle (e.g., during NAC time). CMD45 should not be issued when device executes CMD48.

IF THERE IS ANY OTHER OPERATION TO IMPLEMENT IN ADDITION TO SPECIFICATION IN DATASHEET OR JEDEC STANDARD, PLEASE CONTATC EACH NEAREST SALES OFFICE OR HEADQUARTERS OF KOWIN

4.1.3 EXECUTE_READ_TASK - CMD46

In order to execute a data read task that is already queued the host issues CMD46. The argument of CMD46 is the Task ID of the requested task. The designated task must be a data read task (Data Direction = 1), that is marked as “ready for execution” in the Queue Status Register.

If the requested task (as indicated by the Task ID) is indeed a read task and is ready for execution, the device shall respond with R1 response and start the data transfer.
When the last data block has been fully transferred over the eMMC bus, the device should clear "ready for execution" for the task.

The handling of error cases related to CMD46 is listed in Table 5.

4.1.4 EXECUTE_WRITE_TASK - CMD47

In order to execute a data write task that is already queued the host issues CMD47. The argument of CMD47 is the Task ID of the requested task. The designated task must be a data write task (Data Direction = 0), that is marked as “ready for execution” in the Queue Status Register.

If the requested task (as indicated by the Task ID) is indeed a write task and is ready for execution, the device shall respond with R1 response and the host will start the data transfer.
When the busy signal following the last data block is released, the device should clear "ready for execution" for the task.

The handling of error cases related to CMD47 is listed in Table 5.

4.1.5 CMDQ_TASK_MGMT - CMD48

CMD48 is used for various Command Queuing task management requests. The argument includes a TM op-code, which encodes the requested operation and a TaskID when relevant to the op-code. The list of valid TM op-codes and the respective descriptions of the operations is given in Table 4.

CMD48 may only be issued when no data transfer is taking place.

[Table 4] Task Management op-codes

TM op-code	Description	Task ID Required?
0h	Reserved	N/A
1h	**Discard entire queue**: Device shall discard all tasks in the queue[1] and QSR shall be cleared.	no
2h	**Discard Task: Device** shall discard designated task[2] and QSR designated task shall be cleared.	yes
3h-Fh	Reserved	

NOTE 1 If the queue is empty, device shall execute command without an error.
NOTE 2 If Task ID does not exist, device shall execute command without an error.

IF THERE IS ANY OTHER OPERATION TO IMPLEMENT IN ADDITION TO SPECIFICATION IN DATASHEET OR JEDEC STANDARD, PLEASE CONTATC EACH NEAREST SALES OFFICE OR HEADQUARTERS OF KOWIN

4.1.6 SEND_STATUS - CMD13

An option is added to CMD13 for reading the Queue Status Register (QSR) by the host. If bit [15] in CMD13's argument is set, then the device shall send an R1 Response with the QSR instead of the Device Status.

When the device, in its response to CMD13, indicates that one or more tasks are 'ready for execution', the host should select one of these tasks for execution, and not send additional CMD13s in expectation that additional tasks would become 'ready for execution'.

4.1.7 Error handling

A number of error cases are introduced by the command queuing mechanism. The new error cases are listed in Table 5, along with a description of how each case is handled by the eMMC device.

[Table 5] Error handling for Command Queue

	Command	Error(s) Description	Response	Error Handling
1	CMD44/CMD45/ CMD46/CMD47/ CMD48	Received when Command Queuing is not enabled	No Response	Illegal Command
2	CMD44	Task ID already in use; Task ID exceeds CMDQ_DEPTH; Block Count equals zero	No Response	Illegal Command
3	CMD44	Error in other parameters (e.g., ContextID)	OK (no error indication)	Type 'X' errors
4	CMD45	Sent not immediately after CMD44	No Response	Illegal Command
5	CMD45	Type 'R' errors (e.g., OOR)	Error in Response	Error returned in Response
6	CMD45	Type 'X' errors (e.g., WP violation)	OK (no error indication)	Type 'X' errors
7	CMD46/CMD47	Error in parameters: non-existent Task ID; Task ID is not ready for execution; Wrong direction	No Response	Illegal Command
8	CMD48	Invalid TM op-code	OK (no error indication)	Type 'X' error (ERROR)

4.1.8 Supported Commands

Command queuing can be enabled and disabled using CMDQ_MODE_EN field in the extended CSD. The queue must be empty prior to disabling it. Trial of disabling non-empty queue shall be regarded as illegal command.

When command queuing is enabled (CMDQ Mode En bit in CMDQ_MODE_EN field is set to '1') class 11 commands are the only method through which data transfer tasks can be issued. Existing data transfer commands, namely CMD18/CMD17 and CMD25/CMD24, are not supported when command queuing is enabled.

When command queuing is disabled, CMD17/18 and CMD24/25 are supported, as they are in previous revisions of eMMC.

When the task queue is enabled and empty host should only send class11 commands in Trans state. However, if the device is in Trans state and processes a sequence of commands (e.g., erase sequence) host should complete the sequence before sending class11 commands.

When command queuing mode is enabled, some legacy commands may still be used, while other are treated as illegal. The legality status of commands in different queuing modes is listed in Table 6.
If CMD12 is issued during CMD46 data transfer, the device moves back to Transfer state and the operation can be considered completed (QSR[i] = 1 → 0).

If CMD12 is issued during CMD47 data transfer, the device moves to Programming state and the operation can be considered completed (QSR[i] = 1 → 0) only when all the transferred data have been programmed.

[Table 6] Supported Commands for Command Queue

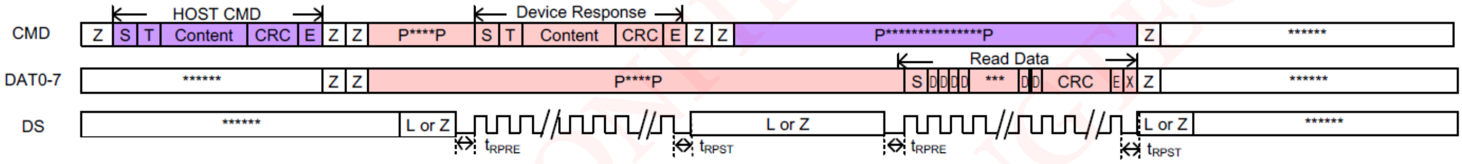
Mode	Task Queue status	Commands	Status
CMDQ_MODE_EN is set to 0.	N/A	Class 11 commands	Illegal
		All other commands	Legal (same as eMMC5.0)
CMDQ_MODE_EN is set to 1.	Empty	CMD17, CMD18, CMD 24, CMD 25	Illegal
		Class 11 commands	Legal
		All other commands	Legal (same as eMMC5.0)
	Non-empty	Class 11 commands	Legal
		CMD0	Legal (all tasks in the queue and QSR will reset)
		CMD12	Legal
		CMD13	Legal
		All other commands	Illegal

4.2 Enhanced Strobe in HS400 Mode

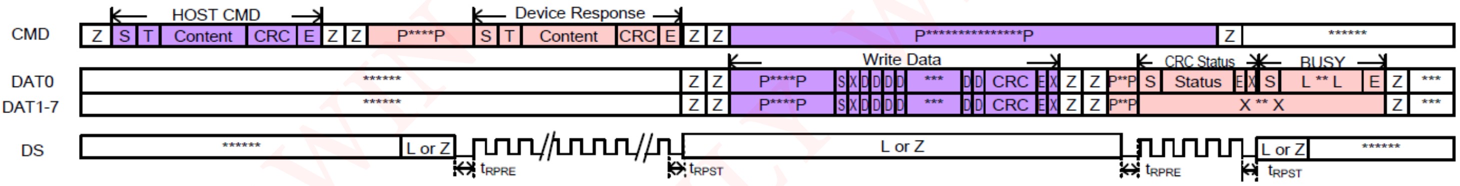
Following a WRITE_MULTIPLE_BLOCK command (CMD25), the device updates the Support of Enhanced Strobe mode is indicated by STROBE_SUPPORT[184] register of EXT_CSD. The following explanation relates to devices that supports Enhanced Strobe mode.

In HS400 a STROBE signal is provided by the device to the host. The STROBE signal is provided only in a limited time periods. The time periods that STROBE signal is provided while in HS400 mode are defined by Enhanced Strobe bit in BUS_WIDTH [183] register of EXT_CSD.
While Enhanced Strobe bit is set to “0” the STROBE is provided during DATA Out and CRC Response time periods. Data Out and CRC response are synched to the STROBE clock signals (as described in Figure 6).

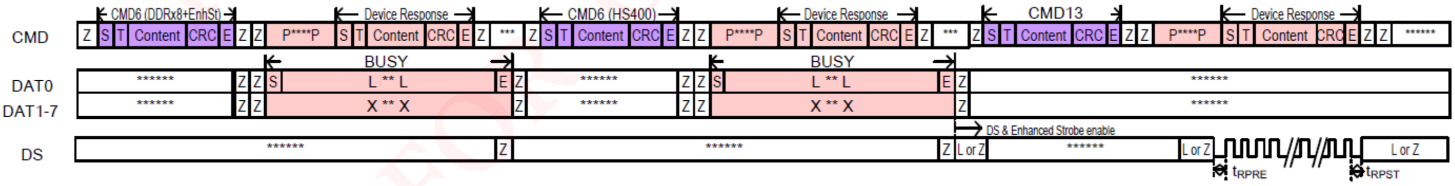
While Enhanced Strobe bit is set to “1” the STROBE is provided during DATA Out, CRC Response and also during CMD Response.


HOST CMD
Device Response
CMD
Z S T Content CRC E Z Z P****P S T Content CRC E Z Z P***************P Z ******
Read Data
DAT0-7
****** Z Z P****P S D D D D *** D D CRC E X Z ******
DS
****** L or Z L or Z L or Z ******
tRPRE
tRPST
tRPRE
tRPST

[Figure 6] Enhanced Strobe signals for CMD Response and Data Out (Read operation)


HOST CMD
Device Response
CMD
Z S T Content CRC E Z Z P****P S T Content CRC E Z Z P***************P Z ******
Write Data
CRC Status
BUSY
DAT0
****** Z Z P****P S X D D D D *** D D CRC E X Z Z P**P S Status E X S L ** L E Z ***
DAT1-7
****** Z Z P****P S X D D D D *** D D CRC E X Z Z P**P X ** X Z ***
DS
****** L or Z L or Z L or Z ******
tRPRE
tRPST
tRPRE
tRPST

[Figure 7] Enhanced Strobe signals for CMD Response and CRC Response (Write operation)


CMD6 (DDRx8+EnhSt)
Device Response
CMD6 (HS400)
Device Response
CMD13
Device Response
CMD
Z S T Content CRC E Z Z P***P S T Content CRC E Z *** Z S T Content CRC E Z Z P***P S T Content CRC E Z *** Z S T Content CRC E Z Z P***P S T Content CRC E Z Z ******
BUSY
BUSY
DAT0
****** Z Z S L ** L E Z ****** Z Z S L ** L E Z ******
DAT1-7
****** Z Z X ** X Z ****** Z Z X ** X Z ******
DS & Enhanced Strobe enable
DS
****** Z ****** Z L or Z ****** L or Z L or Z
tRPRE
tRPST

[Figure 8] HS400 mode change with Enhanced Strobe

In Enhanced Strobe mode DATA OUT, CRC Response and CMD Response are all synched to STROBE clocks. The timing relation between CMD Response output signals and STROBE clocks is the same as defined for DATA Out to STROBE clocks. Refer to HS400 timing diagrams (Figure 6).
For detailed tRPRE and tRPST timing refer to Table 23 in 7.3.3.

IF THERE IS ANY OTHER OPERATION TO IMPLEMENT IN ADDITION TO SPECIFICATION IN DATASHEET OR JEDEC STANDARD, PLEASE CONTATC EACH NEAREST SALES OFFICE OR HEADQUARTERS OF KOWIN

4.3 Cache Enhancement Barrier

Barrier function provides a way to perform a delayed in-order flushing of a cached data. The main motivation for using barrier commands is to avoid the long delay that is introduced by flush commands. There are cases where the host is not interested in flushing the data right away, however it would like to keep an order between different cached data batches. The barrier command enables the host achieving the in-order goal but without paying the flush delay, since the real flushing can be delayed by the device to some later idle time. The formal definition of the barrier rule is as follows:
Denote a sequence of requests Ri, i=0,..,N. Assuming a barrier is set between requests Rx and Rx+1 (0<x<N) then all the requests R0..Rx must be flushed to the nonvolatile memory before any of the requests Rx+1..RN.
Between two barriers the device is free to write data into the nonvolatile memory in any order. If the host wants to preserve a certain order it shall flush the cache or set another barrier at a point where order is important.
The barrier is set by writing to the BARRIER bit of the FLUSH_CACHE byte (EXT_CSD byte [32]). Any error resulted can be read from the status register by CMD13 after the completion of the programming as defined for a normal write request. The error could affect any data written to the cache since the previous flush operation.
The device shall support any number of barrier commands between two flush commands. In case of multiple barrier commands between two flush commands a subset of the cached data may be committed to the nonvolatile memory according to the barrier rule. Internally, a device may have an upper limit on the barrier amount it can absorb without flushing the cache. That is, if the host exceeds this barrier amount, the device may issue, internally, a normal flush.
The device shall expose its barrier support capability via the BARRIER_SUPPORT byte (EXT_CSD byte [486]). If a device does not support barrier function this register shall be zero. If a device supports barrier function this register shall be one.
Assuming the device supports barrier function, if the BARRIER bit of the FLUSH_CACHE byte is set, a barrier operation shall be executed.
If the cache gets totally full and/or the cache is not able to receive the data of the next access (per block count indicated in CMD23 or per initiated single / open ended multiple block write in general) then it shall still be the responsibility of the eMMC device to store the data of the next access within the timeouts that are specified elsewhere in this specification. The actual algorithm to handle the new data and possible flush of some older cached data is left for the implementation.

NOTE When issuing a force-programming write request (CMD23 with bit 24 on) or a reliable write request (CMD23 with bit 31 on), the host should be aware that the data will be written to the nonvolatile memory, potentially, before any cached data, even if a barrier command was issued. Therefore, if the writing order to the nonvolatile memory is important, it is the responsibility of the host to issue a flush command before the force-programming or the reliable-write request.

In order to use the barrier function, the host shall set bit 0 of BARRIER_EN (EXT_CSD byte [31]).
The barrier feature is optional for an eMMC device.

IF THERE IS ANY OTHER OPERATION TO IMPLEMENT IN ADDITION TO SPECIFICATION IN DATASHEET OR JEDEC STANDARD, PLEASE CONTATC EACH NEAREST SALES OFFICE OR HEADQUARTERS OF KOWIN

4.4 Cache Flushing Policy

Any application running on the host may issue write protection by updating fields of write protection The host may require the device to flush data from the cache in an in-order manner. From time to time, to guarantee in-order flushing, the host may command the device to flush the device cache or may use a barrier command.

However, if the eMMC device flushing policy is to flush data from the cache in an in-order manner, cache barrier commands or flush commands operations (In case goal is to guarantee the flushing order) are redundant and impose a needless overhead to the device and host.

FIFO bit in CACHE_FLUSH_POLICY field (EXT_CSD byte [240]) is used by the device to indicate to the host that the device cache flushing policy is First-In-First-Out; this means that the device guarantees that the order of the flushing of data would be the in same order that data was written to the cache. When the FIFO bit is set it is recommended for the host not to send cache barrier commands or flush operations which goal is to guarantee the flushing order as they are redundant and impose a burden to the system.

However, if the FIFO bit is set to 1b and the device supports the cache barrier mechanism, the host may still send barrier commands without getting an error. Sending these commands will not change the device behavior as device flushes cache in-order anyway.
The CACHE_FLUSH_POLICY field is read-only field and never change its value either by the host or device.

4.5 Secure Write Protect Mode

Any application running on the host may issue write protection by updating fields of write protection related EXT_CSD, like USER_WP[171], BOOT_WP[173], by issuing CMD6, CMD8, CMD28 and CMD29. To prevent un-authorized changes of such write protection related EXT_CSD fields, host should enter the secure write protect mode by setting the SECURE_WP_EN field in SECURE_WP_MODE_CONFIG to 0x1.

If host enters the secure write protection mode, those write protection related fields like USER_WP[171] , BOOT_WP[173], TMP_WRITE_PROTECT[12] and PERM_WRITE_PROTECT[13] are updated only when SECURE_WP_MASK is set as 0x1. Those two SECURE_WP_MODE_CONFIG and SECURE_WP_MODE_ENABLE fields are defined in Device Configuration area shall be updated only by Authenticated Device Configuration write request.

[Table 7] Device Configuration Area

Name	Field	Size (Bytes)	Cell Type	Address
Reserved		253	TBD	[255~3]
Secure Write Protect Configuration	SECURE_WP_MODE_CONFIG	1	R/W/E_P	[2]
Secure Write Protect Enable	SECURE_WP_MODE_ENABLE	1	R/W/E	[1]
Reserved		1	TBD	[0]

5.0 eMMC Enhanced Features

5.1 High Priority Interrupt (HPI)

In some scenarios, different types of data on the device may have different priorities for the host. For example, writing operation may be time consuming and therefore there might be a need to suppress the writing to allow demand paging requests in order to launch a process when requested by the user.

The high priority interrupt (HPI) mechanism enables servicing high priority requests, by allowing the device to interrupt a lower priority operation before it is actually completed, within OUT_OF_INTERRUPT_TIME timeout. Host may need to repeat the interrupted operation or part of it to complete the original request. The HPI command may have one of two implementations in the device:

- CMD12 – based on STOP_TRANSMISSION command when the HPI bit in its argument is set.
- CMD13 – based on SEND_STATUS command when the HPI bit in its argument is set.

Host shall check the read-only HPI_IMPLEMENTATION bit in HPI_FEATURES (EXT_CSD byte [503]) and use the appropriate command index accordingly.

If CMD12 is used with HPI bit set, it differs from the non-HPI command in the allowed state transitions. See JEDEC - Device state transition - , for the specific transitions for both cases).

HPI shall only be executed during prg-state. Then, it indicates the device that a higher priority command is pending and therefore it should interrupt the current operation and return to tran-state as soon as possible, with a different timeout value.

If HPI is received in states other than prg-state, the device behavior is defined in JEDEC - Device state transition - . If the state transition is allowed, response is sent but the HPI bit is ignored. If the state transition is not allowed, the command is regarded as an illegal command.

HPI command is accepted as a legal command in prg-state. However, only some commands may be interrupted by HPI. If HPI is received during commands that are not interruptible, a response is sent but the HPI command has no effect and the original command is completed normally, possibly exceeding the OUT_OF_INTERRUPT_TIME timeout. Table 8 shows the commands are interruptible and those that are not.

IF THERE IS ANY OTHER OPERATION TO IMPLEMENT IN ADDITION TO SPECIFICATION IN DATASHEET OR JEDEC STANDARD, PLEASE CONTATC EACH NEAREST SALES OFFICE OR HEADQUARTERS OF KOWIN

CMD Index	Name	Is interruptible?
CMD24	WRITE_BLOCK	Yes
CMD25	WRITE_MULTIPLE_BLOCK	Yes
CMD38	ERASE	Yes
CMD6	SWITCH, byte BKOPS_START, any value	Yes
CMD6	SWITCH, byte SANITIZE_START, any value	Yes
CMD6	SWITCH, byte POWER_OFF_NOTIFICATION, value POWER_OFF_LONG or SLEEP_NOTIFICATION	Yes
CMD6	SWITCH, byte POWER_OFF_NOTIFICATION, other values	No
CMD6	CACHE_CTRL when used for turning the cache OFF	Yes
CMD6	FLUSH_CACHE	Yes
CMD6	SWITCH, other bytes, any value	No
All others		No

Following a WRITE_MULTIPLE_BLOCK command (CMD25), the device updates the CORRECTLY_PRG_SECTORS_NUM field (EXT_CSD bytes [245:242]) with the number of 512B sectors successfully written to the device. Host may use this information when repeating an interrupted write command – it does not need to re-write again all data sectors, it may skip the correctly programmed sectors and continue writing only the rest of the data that wasn't yet programmed.

In case HPI is interrupting a CMD25 that is part of a packed write command (see JEDEC - Packed Commands -), the CORRECTLY_PRG_SECTORS_NUM field will reflect the accumulated packed sectors that were transferred (plus header) – host may calculate from this number the index of the interruptible individual command and the offset of interruption for it.

If HPI is received during a reliable-write command, the first CORRECTLY_PRG_SECTORS_NUM sectors shall contain the new data and all the rest of the sectors shall contain the old data.

The HPI mechanism shall be enabled before it may be used, by setting the HPI_EN bit in the HPI_MGMT field (EXT_CSD byte [161]). The HPI feature is mandatory for this specification. All devices shall have the HPI_SUPPORT bit in HPI_FEATURES field (EXT_CSD byte [503]) set.

5.2 Background Operations

Devices have various maintenance operations need to perform internally. In order to reduce latencies during time critical operations like read and write, it is better to execute maintenance operations in other times - when the host is not being serviced. Operations are then separated into two types:

- Foreground operations – operations that the host needs serviced such as read or write commands;
- Background operations – operations that the device executes while not servicing the host; Depending on how they can be initiated, there are two types of background operations: manually initiated background operations and autonomously initiated background operations.

Manually initiated method: In order for the device to know when the host does not need it and it can execute background operations, host shall write any value to BKOPS_START (EXT_CSD byte [164]) to manually start background operations. Device will stay busy till no more background processing is needed. Since foreground operations are of higher priority than background operations, host may interrupt on-going background operations using the High Priority Interrupt mechanism (see 5.1 High Priority Interrupt). In order for the device to know if host is going to periodically start background operations, host shall set bit 0 (MANUAL_EN) of BKOPS_EN (EXT_CSD byte [163]) to indicate that it is going to write to BKOPS_START periodically. The device may then delay some of its maintenance operations to when host writes to BKOPS_START.
The device reports its background operation status in bits [1:0] of BKOPS_STATUS (EXT_CSD byte [246]), that can be in one of four possible levels: 0x0: No operations required 0x1: Operations outstanding – non critical 0x2: Operations outstanding – performance being impacted 0x3: Operations outstanding – critical Host shall check the status periodically and start background operations as needed, so that the device has enough time for its maintenance operations, to help reduce the latencies during foreground operations. If the status is at level 3 ("critical"), some operations may extend beyond their original timeouts due to maintenance operations that cannot be delayed anymore. The host should give the device enough time for background operations to avoid getting to this level in the first place.
To allow hosts to quickly detect the higher levels, the URGENT_BKOPS bit in the EXCEPTION_EVENTS_STATUS is set whenever the levels is either 2 or 3. That automatically sets the EXCEPTION_BIT in Device Status. This allows hosts to detect urgent levels on every R1 type response. Hosts shall still read the full status from the BKOPS_STATUS byte periodically and start background operations as needed.
The background operations feature is mandatory for this specification. Bit 0 of BKOPS_SUPPORT (EXT_CSD byte [502]) shall be set.
Autonomously initiated method; a Host that wants to enable the device to perform background operations during device idle time, should signal the device by setting AUTO_EN in BKOPS_EN field [EXT_CSD byte 163] to 1b. When this bit is set, the device may start or stop background operations whenever it sees fit, without any notification to the host.

When AUTO_EN bit is set, the host should keep the device power active. The host may set or clear this bit at any time based on its power constraints or other considerations.

5.3 Cache

Cache is a temporary storage space in an eMMC device. The cache should in typical case reduce the access time (compared to an access to the main nonvolatile storage) for both write and read. The cache is not directly accessible by the host. This temporary storage space may be utilized also for some implementation specific operations like as an execution memory for the memory controller and/or as storage for an address mapping table etc. however that definition is out of scope of this standard.

The cache is expected to be volatile by nature. Implementation of such volatile cache has some significant effects on some of the requirements of this standard as follows:

- Caching of data shall apply only for the single block read/write (CMD17/24), pre-defined multiple block read/write (CMD23+CMD18/25) and open ended multiple block read/write (CMD18/25+CMD12) commands and excludes any other access e.g., to the register space (e.g., CMD6).
- Data0 busy and status response bit [8] after CMD24, CMD23+CMD25 or CMD25+CMD12 does not anymore necessarily indicate the programming status to the nonvolatile memory but may indicate programming status to the volatile cache (exceptions defined later).
- A Flush operation refers to the requirement, from the host to the device, to write the cached data to the nonvolatile memory. Prior to a flush, the device may autonomously write data to the nonvolatile memory, but after the flush operation all data in the volatile area must be written to nonvolatile memory.
- Data in the cache may (and most probably will) be lost due to a sudden power down. If there was a flush operation ongoing when the power was lost then also any such data may be lost.
- Accesses to the RPMB and Boot partitions while the cache is ON shall still be directed to the nonvolatile storage with same requirements as defined elsewhere in this standard.
- When the cache is turned ON it applies to the eMMC device as whole (When flushing the cache the data for all partitions shall be flushed, this operation is independent of the active partition).
- There is no requirement for flush due to switching between the partitions. (Note: This also implies that the cache data shall not be lost when switching between partitions). Cached data may be lost in SLEEP state, so host should flush the cache before placing the device into SLEEP state.
- The device may invalidate the data in the cache in case of RST_n or CMD0 received.

Support of the cache function and size of the cache are indicated in the CACHE_SIZE byte (EXT_CSD byte [252:249]).

The cache shall be OFF by default after power up, RST_n assertion or CMD0. All accesses shall be directed to the nonvolatile storage like defined elsewhere in this specification. The cache function can be turned ON and OFF by writing to the CACHE_CTRL byte (EXT_CSD byte [33]). Turning the cache ON shall enable behavior model defined in this section. Turning the cache OFF shall trigger flushing of the data to the nonvolatile storage.

The cache may be flushed to the nonvolatile storage by writing to FLUSH_CACHE byte (EXT_CSD byte [32]). The R1b response result shall reflect the status of programming of cached data to the nonvolatile storage.

be directed to the nonvolatile storage like defined elsewhere in this specification. The cache function can be turned ON and OFF by writing to the CACHE_CTRL byte (EXT_CSD byte [33]). Turning the cache ON shall enable behavior model defined in this section. Turning the cache OFF shall trigger flushing of the data to the nonvolatile storage.

The cache may be flushed to the nonvolatile storage by writing to FLUSH_CACHE byte (EXT_CSD byte [32]). The R1b response result shall reflect the status of programming of cached data to the nonvolatile storage.

The cache may be flushed to the nonvolatile storage by writing to FLUSH_CACHE byte (EXT_CSD byte [32]). The R1b response result shall reflect the status of programming of cached data to the nonvolatile storage. Any error resulted can be read from the status register by CMD13 after the completion of the programming as defined for normal write. If a flush error occurs during the execution of the FLUSH_CACHE or while turning off the cache using the CACHE_CTRL operation, the device shall set the generic error bit, STATUS BIT[19]. If an error occurs as a result of flush operation the device has no responsibility to isolate the error to a specific data area. The error could affect any data written to the cache since the previous flush operation.

There is no maximum timeout for flushing of the cache as flushing a large amount of cached data may take very unpredictably long time. This applies both for the FLUSH_CACHE and CACHE_CTRL (turning the cache OFF) operations. Host may use the HPI function to interrupt the flush operation. In this case the cache shall not be considered as fully flushed and host should re-initiate the flush.

There are two ways to force data to be programmed directly to the nonvolatile storage while the cache is turned ON:

1) A Reliable Write access shall force the data to be written to the nonvolatile storage and Data0 busy indication reflects the status of the programming as defined elsewhere in this standard,
2) Set the argument bit [24] in the CMD23 in prior to the actual write command. In case the cache is turned OFF then this bit shall be ignored by the memory device. If this bit is set together with the Reliable Write bit in CMD23 then this bit shall be ignored also by the device.

A logical block may or may not be removed from the cache after the flush operation. The data may also be stored to the cache in case of Reliable Write access and access with CMD23 bit [24] set. These are left for the implementation.

If the cache gets totally full and/or the cache is not able to receive the data of the next access (per block count indicated in CMD23 or per initiated single/open ended multiple block write in general) then it shall still be responsibility of the e•MMC device to store data of the next access within timeouts specified elsewhere in this specification. The actual algorithm to handle the new data and possible flush of some older cached data is left for the implementation.

ERASE, TRIM and DISCARD commands shall have effect on data in the cache accordingly so that any following read will reflect the Erased, Trimmed or Discarded state of data. There may be copies of an old data in the cache and these are expected to be cleared during power down. The SANITIZE operation shall clean up the unmapped data in the cache.

All functions related to a certain address should still be considered as sequential and ordered, just like with a device without the cache. For example if there is some old data in an address and then some new data is written (potentially cached) to this address. Next there is the write protection set to the very same address and write protect is not a cached operation as such. Still the write protection shall in all cases apply to the new data, not to the old data that may still have been in the nonvolatile memory in this particular time. There is no requirement that the cache in general should be first-in-first-out -type, this is left for the implementation.

In between consecutive flush operations the device is free to write data into the nonvolatile memory out of order. If the host wants to preserve the order it must flush the cache at the point where order is important; the device can only ensure data was written in order after the flush operation has completed.
It shall never be possible to read stale data from the eMMC device due to the cached operation.

IF THERE IS ANY OTHER OPERATION TO IMPLEMENT IN ADDITION TO SPECIFICATION IN DATASHEET OR JEDEC STANDARD, PLEASE CONTATC EACH NEAREST SALES OFFICE OR HEADQUARTERS OF KOWIN

A signed access to a Replay Protected Memory Block is provided. This function provides means for the system to store data to the specific memory area in an authenticated and replay protected manner. This is provided by first programming authentication key information to the eMMC memory (shared secret).

As the system cannot be authenticated yet in this phase the authentication key programming have to take in a secure environment like in an OEM production. Further on the authentication key is utilized to sign the read and write accesses made to the replay protected memory area with a Message Authentication Code (MAC).

Usage of random number generation and count register are providing additional protection against replay of messages where messages could be recorded and played back later by an attacker.

[Table 9] Data Frame Files for RPMB

Start	Stuff Bytes	Key/ (MAC)	Data	Nonce	Write Counter	Address	Block Count	Result	Req/ Resp	CRC16	End
1bit	196Byte	32Byte (256b)	256Byte	16Byte	4Byte	2Byte	2Byte	2Byte	2Byte	2Byte	1bit
	[511:316]	[315:284]	[283:28]	[27:12]	[11:8]	[7:6]	[5:4]	[3:2]	[1:0]		

[Note] See JEDEC - Replay Protected Memory Block -

- The Data Frame for Replay Protected Memory Block Access
  - RPMB Request/Response Message Types
  - RPMB Operation Results data structure
  - RPMB Operation Results
- Memory Map of the Replay Protected Memory Block
- Message Authentication Code Calculation
- Accesses to the Replay Protected Memory Block
- Programming of the Authentication Key
  - Result Register Read Request Packet
  - Response for Key Programming Result Request
- Reading of the Counter Value
  - Counter Read Request Packet
  - Counter Value Response
- Authenticated Data Write
  - Program Data Packet
  - Result Register Read Request Packet
  - Response for Data Programming Result Request
- Authenticated Data Read
- Authenticated Device Configuration Write
- Authenticated Device Configuration Read

IF THERE IS ANY OTHER OPERATION TO IMPLEMENT IN ADDITION TO SPECIFICATION IN DATASHEET OR JEDEC STANDARD, PLEASE CONTATC EACH NEAREST SALES OFFICE OR HEADQUARTERS OF KOWIN

The Trim operation is similar to the default erase operation described in 6.6.9. The Trim function applies the erase operation to write blocks instead of erase groups. The Trim function allows the host to identify data that is no longer required so that the Device can erase the data if necessary during background erase events. The contents of a write block where the trim function has been applied shall be '0' or '1' depending on different memory technology. This value is defined in the EXT_CSD.

Once the trim command completes successfully, the mapped device address range that was trimmed shall behave as if it was overwritten with all ‘0’ or all ‘1’ depending on the different memory technology. The impact of the trim command should be simply moving the mapped host address range to the unmapped host address range.
NOTE In some cases other flash management tasks may also be completed during the execution of this command.

Completing the TRIM process is a three steps sequence. First the host defines the start address of the range using the ERASE_GROUP_START (CMD35) command, next it defines the last address of the range using the ERASE_GROUP_END (CMD36) command and finally it starts the erase process by issuing the ERASE (CMD38) command with argument bit 0 set to one and the remainder of the arguments set to zero. In the case of a TRIM operation both CMD35 and CMD36 identify the addresses of write blocks rather than erase groups.

If an element of the Trim command (CMD35, CMD36 or CMD38) is received out of the defined erase sequence, the device shall set the ERASE_SEQ_ERROR bit in the status register and reset the whole sequence.

If the host provides an out of range address as an argument to CMD35 or CMD36, the Device will reject the command, respond with the ADDRESS_OUT_OF_RANGE bit set and reset the whole erase sequence. If a “non erase” command (neither of CMD35, CMD36, CMD38 or CMD13) is received, the Device shall respond with the ERASE_RESET bit set, reset the erase sequence and execute the last command. Commands not addressed to the selected Device do not abort the erase sequence.

If the trim range includes write protected blocks, they shall be left intact and only the non-protected blocks shall be erased. The WP_ERASE_SKIP status bit in the status register shall be set. As described above for block write, the Device will indicate that a Trim command is in progress by holding DAT0 low. The actual erase time may be quite long, and the host may issue CMD7 to deselect the Device.

The host should execute the Trim command with caution to avoid unintentional data loss.
Resetting a Device (using CMD0, CMD15, or hardware reset for e•MMC) or power failure will terminate any pending or active Trim command. This may leave the data involved in the operation in an unknown state

The Sanitize operation is a feature, in addition to TRIM and Erase that is used to remove data from the device according to Secure Removal Type (see JEDEC - SECURE_REMOVAL_TYPE [16] -). The use of the Sanitize operation requires the device to physically remove data from the unmapped user address space. A Sanitize operation is initiated by writing a value to the extended CSD[165] SANITIZE_START. While the device is performing the sanitize operation, the busy line is asserted. The device will continue the sanitize operation, with busy asserted, until one of the following events occurs:

- Sanitize operation is complete,
- An HPI is used to abort the operation,
- A power failure, or
- A hardware reset.

After the sanitize operation is completed, no data should exist in the unmapped host address space.
If the sanitize operation is interrupted, either by HPI, power failure, CMD0 or hardware reset, the state of the unmapped host address space cannot be guaranteed. The host must re-initiate the sanitize operation by writing to the SANITIZE_START[165] and allow the operation to complete to be sure that unmapped host address space is clear.

Since the region being operated on is not accessible by the host, applications requiring this feature must work with individual device manufacturers to ensure this operation is performing properly and to understand the impact on device reliability.

5.7 Discard

The Discard is similar operation to TRIM. The Discard function allows the host to identify data that is no longer required so that the device can erase the data if necessary during background erase events. The contents of a write block where the discard function has been applied shall be ‘don’t care’. After discard operation, the original data may be remained partially or fully accessible to the host dependent on device. The portions of data that are no longer accessible by the host may be removed or unmapped just as in the case of TRIM. The device will decide the contents of discarded write block.

The distinction between Discard and TRIM, is that a read to a region that was discarded may return some or all of the original data. However, in the case of Trim the entire region shall be unmapped or removed and will return ‘0’ or ‘1’ depending on the memory technology.

When Sanitize is executed, only the portion of data that was unmapped by a Discard command shall be removed by the Sanitize command. The device cannot guarantee that discarded data is completely removed from the device when Sanitize is applied.

Completing the Discard process is a three steps sequence. First the host defines the start address of the range using the ERASE_GROUP_START (CMD35) command, next it defines the last address of the range using the ERASE_GROUP_END (CMD36) command and finally it starts the erase process by issuing the ERASE (CMD38) command with argument bit 0 and bit 1 set to one and the remainder of the arguments set to zero. In the case of a Discard operation both CMD35 and CMD36 identify the addresses of write blocks rather than erase groups.

If an element of the Discard command (CMD35, CMD36 or CMD38) is received out of the defined erase sequence, the device shall set the ERASE_SEQ_ERROR bit in the status register and reset the whole sequence.

If the host provides an out of range address as an argument to CMD35 or CMD36, the device will reject the command, respond with the ADDRESS_OUT_OF_RANGE bit set and reset the whole erase sequence.
If a “non erase” command (neither of CMD35, CMD36, CMD38 or CMD13) is received, the device shall respond with the ERASE_RESET bit set, reset the erase sequence and execute the last command. Commands not addressed to the selected device do not abort the erase sequence.

If the discard range includes write protected blocks, they shall be left intact and only the non-protected blocks shall be erased. The WP_ERASE_SKIP status bit in the status register shall be set.
As described above for block write, the device will indicate that a Discard command is in progress by holding DAT0 low. The actual erase time may be quite long, and the host may issue CMD7 to deselect the device.

The host should execute the Discard command with caution to avoid unintentional data loss.
Resetting a device (using CMD0, CMD15, or hardware reset for e•MMC) or power failure will terminate any pending or active Discard command. This may leave the data involved in the operation in an unknown state.

NOTE Secure Erase is included for backwards compatibility. New system level implementations (based on v4.51 devices and beyond) should use Erase combined with Sanitize instead of secure erase.

In addition to the standard Erase command there is also an optional Secure Erase command. The Secure Erase command differs from the basic Erase command (outlined in 6.6.9) in that it requires the device to execute the erase operation on the memory array when the command is issued and requires the device and host to wait until the operation is complete before moving to the next device operation. Also, the secure erase command requires the device to do a secure purge operation, according to Secure Removal Type in EXT_CSD, outlined in 7.4.120, on the erase groups, and any copies of items in those erase groups, identified for erase.

This command allows applications that have high security requirements to request that the device perform secure operations, while accepting a possible erase time performance impact. The Secure Erase command is executed in the same way the erase command outlined in 6.6.9, except that the Erase (CMD38) command is executed with argument bit 31 set to 1 and the other argument bits set to zero. See JEDEC - Erase command (CMD38) Valid arguments - for details on the argument combinations supported with the Erase (CMD 38) command and Table 10 for the definition of the argument bits associated with the ERASE (CMD38) command.

The host should execute the Secure Erase command with caution to avoid unintentional data loss. Resetting a device (using CMD0, CMD15, or hardware reset for eMMC) or power failure will terminate any pending or active Secure Erase command. This may leave the data involved in the operation in an unknown state.
[Table 10] Erase command (CMD38) Valid arguments

Bit	Argument	Arguments
31	Secure Request	'1' - Secure form of the command must be performed. '0' - Default in Secure Erase command is performed
15	Force Garbage Collection	'1' -– CMD35 & 36 are ignored. Erase is performed on previously identified write blocks. '0' - Command uses the erase groups identified by CMD35&36
1	Discard Enable	‘1’ – Discard write blocks identified by CMD35 & 36. ‘0’ – Execute an erase. NOTE Bit[0] is also required to be set to 0x1 (in addition to bit[1] being
0	Identify Write Blocks for Erase (or TRIM Enable)	‘1' -– Mark write block identified by CMD35&36 for erase (bit 31 set) or execute Trim operation (bit 31 cleared). Bit[0] shall be set to 0x1 for the Discard write blocks operation. '0' - Execute an erase.

NOTE: Secure TRIM is included for backwards compatibility. New system level implementations (based on v4.51 devices and beyond) should use TRIM combined with Sanitize instead of secure trim.

The Secure Trim command is very similar to the Secure Erase command. The Secure Trim command performs a secure purge operation on write blocks instead of erase groups. To minimize the impact on the device's performance and reliability the Secure Trim operation is completed by executing two distinct steps. In Secure Trim Step 1 the host defines the range of write blocks that it would like to mark for the secure purge. This step does not perform the actual purge operation. The blocks are marked by defining the start address of the range using the ERASE_GROUP_START (CMD35) command, followed by defining the last address of the range using the ERASE_GROUP_END (CMD36) command. In the case of Secure Trim, both ERASE_GROUP_START and ERASE_GROUP_END arguments are identifying write block addresses. Once the range of blocks has been identified the ERASE (CMD 38) with argument bit 31and 0 set to 1 and the remainder of the argument bits set to 0 (See JEDEC - Erase command (CMD38) Valid arguments - for the allowable arguments) is applied. This completes Secure Trim Step 1.

Secure Trim Step 1 can be repeated several times, with other commands being allowed in between, until all the write blocks that need to be purged have been identified. It is recommended that the Secure Trim Step 1 is done on as many blocks as possible to improve its efficiency of the secure trim operation.

Secure Trim Step 2 issues the ERASE_GROUP_START (CMD35) and ERASE_GROUP_END (CMD36) with addresses that are in range. Note the arguments used with these commands will be ignored. Then the ERASE (CMD 38) with bit 31 and 15 set to 1 and the remainder of the argument bits to 0 is sent. This step actually performs the secure purge on all the write blocks, according to Secure Removal Type, outlined in 7.4.120, as well as any copies of those blocks, that were marked during Secure Trim Step 1 and completes the secure trim operation. Other commands can be issued to the device in between Secure Trim Step 1 and Secure Trim Step 2.

The host may issue Secure Trim Step 2 without issuing Secure Trim Step 1. This may be required after a power failure event in order to complete unfinished secure trim operations. If Secure Trim Step 2 is done and there are no write blocks marked for erase then Secure Trim Step 2 again will have no impact on the device.

Once a write block is marked for erase using Secure Trim Step 1, it is recommended that the host consider this block as erased. However, if the host does write to a block after it has been marked for erase, then the last copy of the block, that occurred as a result of the modification, will not be marked for erase. All previous copies of the block will remain marked for erase.

If the host application wishes to use the secure TRIM command as the method to remove data from the device, then the host should make sure that it completes secure trim step 1 for a write block before using a write command to overwrite the block. This will ensure that the overwritten data is removed securely from the device the next time that secure Trim step 2 is issued.

If either CMD35, CMD36 or CMD38 is received out of the defined erase sequence, the device shall set the ERASE_SEQ_ERROR bit in the status register and reset the whole sequence for an individual step. If the host provides an out of range address as an argument to CMD35 or CMD36, the device will reject the command, respond with the ADDRESS_OUT_OF_RANGE bit set and reset the whole erase sequence.

If a 'non erase' command (neither of CMD35, CMD36, CMD38 or CMD13) is received, while the device is performing the operations in Secure Trim Step 1 or Secure Trim Step 2, the device shall respond with the ERASE_RESET bit set, and reset the individual step without completing the operation and execute the last command. Commands not addressed to the selected device do not abort the sequence. Other commands may occur in between the multiple iterations of Secure Trim Step 1 and/or before Secure Trim Step 2 is sent. However, the sequence during each of the steps cannot be interrupted.

If a power failure or reset occurs in between Secure Trim Step 1 and Secure Trim Step 2. The blocks that were identified for the secure purge operation will remain marked. The next time the device sees Secure Trim Step 2 it will purge the blocks that were marked prior to the power failure or reset and along with any blocks that have been identified since that point.

The host should execute the Secure Trim command with caution to avoid unintentional data loss. Resetting a device (using CMD0, CMD15, or hardware reset for eMMC) or power failure during either step 1 or step2 will terminate any pending or active secure trim command. This may leave the data involved in the operation in an unknown state. If the erase range includes write protected blocks, they shall be left intact and only the non-protected blocks shall be erased. The WP_ERASE_SKIP status bit in the status register shall be set.

If the device needs a write block that is marked for Secure Erase and Secure Trim Step 2 has not been issued, the device can issue a secure purge operation on that write block as a background task. As described above for block write, the device will indicate that either Secure Trim Step 1 or Secure Trim Step 2 are in progress by holding DAT0 low. The actual time for the operation may be quite long, and the host may issue CMD7 to deselect the device. If the write block size is changed in between Secure Trim Step 1 and Secure Trim Step 2 then the write block size used during step 1 of the operation will apply.

5.10 Field Firmware Update

Field Firmware Updates (FFU) enables features enhancement in the field. Using this mechanism the host downloads a new version of the firmware to the eMMC device and, following a successful download, instructs the eMMC device to install the new downloaded firmware into the device.

In order to start the FFU process the host first checks if the eMMC device supports FFU capabilities by reading SUPPORTED_MODES and FW_CONFIG fields in the EXT_CSD. If the eMMC device supports the FFU feature the host may start the FFU process. The FFU process starts by switching to FFU Mode in MODE_CONFIG field in the EXT_CSD. In FFU Mode host should use closed-ended or open ended commands (CMD17/CMD18/CMD24/CMD25) for downloading the new firmware and reading vendor proprietary data. In this mode, the host should set the argument of these commands to be as defined in FFU_ARG field. In case these commands have a different argument the device behavior is not defined and

the FFU process may fail. The host should set Block Length to be DATA_SECTOR_SIZE. Downloaded firmware bundle must be DATA_SECTOR_SIZE size aligned (internal padding of the bundle might be required).
Once in FFU Mode the host may send the new firmware bundle to the device using one or more write commands.

The host could regain regular functionally of write and read commands by setting MODE_CONFIG field in the EXT_CSD back to Normal state. Switching out of FFU Mode may abort the firmware download operation. When switched back-in to FFU Mode, the host should check the FFU Status to get indication about the number of sectors that were downloaded successfully by reading the NUMBER_OF_FW_SECTORS_CORRECTLY_PROGRAMMED in the extended CSD. In case the number of sectors that were downloaded successfully is zero the host should re-start downloading the new firmware bundle from its first sector. In case the number of sectors that were downloaded successfully is positive the host should continue the download from the next sector, that would resume the firmware download operation.

In case MODE_OPERATION_CODES field is supported by the device (defined in SUPPORTED_MODE_OPERATION_CODES bit in FFU_FEATURES field in extended CSD) the host may abort the firmware download process by setting MODE_OPERATION_CODES to FFU_ABORT to inform the device the FFU operation has been aborted. In response the device shall set NUMBER_OF_FW_SECTORS_CORRECTLY_PROGRAMMED field to zero and be ready to receive a new firmware bundle. If the host finished downloading successfully the firmware bundle to the device it may set MODE_OPERATION_CODES to FFU_INSTALL, the device shall set NUMBER_OF_FW_SECTORS_CORRECTLY_PROGRAMMED field to zero, install the new firmware and set MODE_CONFIG to Normal state that would regain regular operation of read and write commands. The host should set MODE_OPERATION_CODES to FFU_INSTALL. If the host performs a CMD0/HW_Reset/Power cycle through the FFU process, prior to successful execution of the FFU_INSTALL command, device may abort the download process.

In case MODE_OPERATION_CODES field is not supported by the device the host sets to NORMAL state and initiates a CMD0/HW_Reset/Power cycle to install the new firmware. In such case the device doesn't need to use NUMBER_OF_FW_SECTORS_CORRECTLY_PROGRAMMED.

In both cases occurrence of a CMD0/HW_Reset/Power cycle before the host successfully downloaded the new firmware bundle to the device may cause the firmware download process to be aborted.
When in FFU_MODE and host sends other commands that are not part of the recommended flow, device behavior may be undefined.

[Note] For the other eMMC Device features , See JEDEC

6.0 REGISTER VALUE

6.1 OCR Register

The 32-bit operation conditions register (OCR) stores the VDD voltage profile of the Device and the access mode indication. In addition, this register includes a status information bit. This status bit is set if the Device power up procedure has been finished. The OCR register shall be implemented by all Devices.

[Table 11] OCR Register

OCR bit	VCCQ voltage window[2]	Register Value
[6 : 0]	Reserved	000 0000b
[7]	1.70 – 1.95V	1b
[14 : 8]	2.0 – 2.6V	000 0000b
[23 : 15]	2.7 – 3.6V	1 1111 1111b
[28 : 24]	Reserved	0 0000b
[30 : 29]	Access Mode	10b (sector mode)-[High than 2GB]
[31]	eMMC power up status bit (busy)[1]	

NOTE 1 This bit is set to LOW if the Device has not finished the power up routine.
NOTE 2 The voltage for internal flash memory should be 2.7~3.6v regardless of OCR Register value.

6.2 CID Register

The Device IDentification (CID) register is 128 bits wide. It contains the Device identification information used during the Device identification phase (eMMC protocol). Every individual flash or I/O Device shall have an unique identification number. Every type of eMMC Device shall have a unique identification number.

[Table 12] CID Register

Name	Field	Width	CID-Slice	CID Value
Manufacturer ID	MID	8	[127:120]	0xEA
Bank Index Number	BIN	6	[119:114]	0X0B
Device/BGA	CBX	2	[113:112]	0x01
OEM/Application ID	OID	8	[111:104]	0x00
Product name	PNM	48	[103:56]	See Product Name Table
Product revision	PRV	8	[55:48]	Composed of the revision count of controller and the revision count of F/W patch
Product serial number	PSN	32	[47:16]	A 32-bits unsigned binary integer(Random Number)
Manufacturing date	MDT	8	[15:8]	Same as eMMC JEDEC Standard
CRC7 checksum	DRC	7	[7:1]	Same as eMMC JEDEC Standard
Not used always “1”	-	1	[0:0]	

IF THERE IS ANY OTHER OPERATION TO IMPLEMENT IN ADDITION TO SPECIFICATION IN DATASHEET OR JEDEC STANDARD, PLEASE CONTATC EACH NEAREST SALES OFFICE OR HEADQUARTERS OF KOWIN

Part ID	Density	Product Name in CID Register(PND)
KAS0411D	4GB	0x533034313144
KAS0311D	8GB	0x533033313144

6.3 CSD Register

The Device-Specific Data (CSD) register provides information on how to access the Device contents. The CSD defines the data format, error correction type, maximum data access time, data transfer speed, whether the DSR register can be used etc. The programmable part of the register (entries marked by W or E below) can be changed by CMD27. The type of the CSD Registry entries below is coded as follows:

- R : Read only.
- W: One time programmable and not readable.
- R/W: One time programmable and readable.
- W/E: Multiple writable with value kept after power failure, H/W reset assertion and any CMD0 reset and not readable.
- R/W/E: Multiple writable with value kept after power failure, H/W reset assertion and any CMD0 reset and readable.
- R/W/C_P: Writable after value cleared by power failure and HW/rest assertion (the value not cleared by CMD0 reset) and readable.
- R/W/E_P: Multiple writable with value reset after power failure, H/W reset assertion and any CMD0 reset and readable.
- W/E_P: Multiple writable with value reset after power failure, H/W reset assertion and any CMD0 reset and not readable.

IF THERE IS ANY OTHER OPERATION TO IMPLEMENT IN ADDITION TO SPECIFICATION IN DATASHEET OR JEDEC STANDARD, PLEASE CONTATC EACH NEAREST SALES OFFICE OR HEADQUARTERS OF KOWIN

Name	Field	Bit	Type	Slice	Value	Note
CSD structure	CSD_STRUCTURE	2	R	[127:126]	3h	
MultiMediaCard protocol version	SPEC_VERS	4	R	[125:122]	4h	
Reserved	-	2	R	[121:120]	-	
Data read access-time-1	TAAC	8	R	[119:112]	27h	
Data read access-time-2 in CLK cycles (NSAC*100)	NSAC	8	R	[111:104]	01h	
Max. data transfer rate	TRAN_SPEED	8	R	[103:96]	32h	
Command classes	CCC	12	R	[95:84]	0F5h	
Max. read data block length	READ_BL_LEN	4	R	[83:80]	9h	
Partial blocks for read allowed	READ_BL_PARTIAL	1	R	[79:79]	0h	
Write block misalignment	WRITE_BLK_MISALIGN	1	R	[78:78]	0h	
Read block misalignment	READ_BLK_MISALIGN	1	R	[77:77]	0h	
DSR implemented	DSR_IMP	1	R	[76:76]	0h	
Reserved	-	2	R	[75:74]	-	
Device size	C_SIZE	12	R	[73:62]	FFFh	
Max. read current at VCC(min)	VDD_R_CURR_MIN	3	R	[61:59]	7h	
Max. read current at VCC(max)	VDD_R_CURR_MAX	3	R	[58:56]	7h	
Max. write current at VCC(min)	VDD_W_CURR_MIN	3	R	[55:53]	7h	
Max. write current at VCC(max)	VDD_W_CURR_MAX	3	R	[52:50]	7h	
Device size multiplier	C_SIZE_MULT	3	R	[49:47]	7h	
Erase group size	ERASE_GRP_SIZE	5	R	[46:42]	1Fh	
Erase group size multiplier	ERASE_GRP_MULT	5	R	[41:37]	1Fh	
Write protect group size	WP_GRP_SIZE	5	R	[36:32]	See Note	≦64GB: 0x0F. Others: 0x1F
Write protect group enable	WP_GRP_ENABLE	1	R	[31:31]	1h	
Manufacturer default ECC	DEFAULT_ECC	2	R	[30:29]	0h	
Write speed factor	R2W_FACTOR	3	R	[28:26]	2h	
Max. write data block length	WRITE_BL_LEN	4	R	[25:22]	9h	
Partial blocks for write allowed	WRITE_BL_PARTIAL	1	R	[21:21]	0h	
Reserved	-	1	-	[20:20]	-	
Content protection application	CONTENT_PROT_APP	1	R	[16:16]	0h	
File format group	FILE_FORMAT_GRP	1	R/W	[15:15]	0h	
Copy flag (OTP)	COPY	1	R/W	[14:14]	1h	
Permanent write protection	PERM_WRITE_PROTECT	1	R/W	[13:13]	0h	
Temporary write protection	TMP_WRITE_PROTECT	1	R/W/E	[12:12]	0h	
File format	FILE_FORMAT	2	R/W	[11:10]	0h	
ECC code	ECC	2	R/W/E	[9:8]	0h	
CRC	CRC	7	R/W/E	[7:1]	TBD	
Not used, always '1'	-	1	-	[0:0]	TBD	

[Note]

1. The Device-Specific Data (CSD) register defines the behavior of eMMC devices. The eMMC behavior is related to the controller design. The following table shows a typical CSD definition of KOWIN eMMC based eMMC. If users need to add on more features, firmware or hardware modifications may be necessary.
2. Reserved bits should be read as “0”.

The Extended CSD register defines the Device properties and selected modes. It is 512 bytes long. The most significant 320 bytes are the Properties segment, that defines the Device capabilities and cannot be modified by the host. The lower 192 bytes are the Modes segment, that defines the configuration the Device is working in. These modes can be changed by the host by means of the SWITCH command. Multi bytes field is interpreted in little endian byte order.

- R: Read only.
- W: One time programmable and not readable.
- R/W: One time programmable and readable.
- W/E: Multiple writable with value kept after power failure, H/W reset assertion and any CMD0 reset and not readable.
- R/W/E: Multiple writable with value kept after power failure, H/W reset assertion and any CMD0 reset and readable.
- R/W/C_P: Writable after value cleared by power failure and HW/rest assertion (the value not cleared by CMD0 reset) and readable.
- R/W/E_P: Multiple writable with value reset after power failure, H/W reset assertion and any CMD0 reset and readable.
- W/E_P: Multiple writable with value reset after power failure, H/W reset assertion and any CMD0 reset and not readable.

[Table 15] Extended CSD Register

Name	Field	Byte	Type	Slice	Value	Note
Reserved	-	6	-	[511:506]	-	
Extended Security Commands Error	EXT_SECURITY_ERR	1	R	[505]	0h	
Supported command sets	S_CMD_SET	1	R	[504]	1h	
HPI features	HPI_FEATURES	1	R	[503]	1h	
Background operations support	BKOPS_SUPPORT	1	R	[502]	1h	
Max packed read commands	MAX_PACKED_READS	1	R	[501]	3Fh	
Max packed write commands	MAX_PACKED_WRITES	1	R	[500]	3Fh	
Data Tag Support	DATA_TAG_SUPPORT	1	R	[499]	1h	
Tag Unit Size	TAG_UNIT_SIZE	1	R	[498]	0h	
Tag Resources Size	TAG_RES_SIZE	1	R	[497]	0h	
Context management capabilities	CONTEXT_CAPABILITIES	1	R	[496]	78h	

Name	Field	Byte	Type	Slice	Value	Note
Large Unit size	LARGE_UNIT_SIZE_M1	1	R	[495]	1h	
Extended partitions attribute support	EXT_SUPPORT	1	R	[494]	3h	
Supported modes	SUPPORTED_MODES	1	R	[493]	1h	
FFU features	FFU_FEATURES	1	R	[492]	0h	
Operation codes timeout	OPERATION_CODE_TIMEOUT	1	R	[491]	17h	
FFU Argument	FFU_ARG	4	R	[490:487]	FFFAFFF0h	
Barrier support	BARRIER_SUPPORT	1	R	[486]	1h	
Reserved	-	177	-	[485:309]	-	
CMD Queuing Support	CMDQ_SUPPORT	1	R	[308]	1h	
CMD Queuing Depth	CMDQ_DEPTH	1	R	[307]	1Fh	
Reserved	-	1	TBD	[306]	-	
Number of FW sectors correctly programmed	NUMBER_OF_FW_SECTORS_CORRECTLY_PROGRAMMED	4	R	[305:302]	0000h	
Vendor proprietary health report	VENDOR_PROPRIETARY_HEALTH_REPORT	32	R	[301:270]	0h	
Device life time estimation type B	DEVICE_LIFE_TIME_EST_TYP_B	1	R	[269]	1h	
Device life time estimation type A	DEVICE_LIFE_TIME_EST_TYP_A	1	R	[268]	1h	
Pre EOL information	PRE_EOL_INFO	1	R	[267]	1h	
Optimal read size	OPTIMAL_READ_SIZE	1	R	[266]	40h	
Optimal write size	OPTIMAL_WRITE_SIZE	1	R	[265]	40h	
Optimal trim unit size	OPTIMAL_TRIM_UNIT_SIZE	1	R	[264]	7h	
Device version	DEVICE_VERSION	2	R	[263:262]	see Note	4G: 0x3405 8G: 0x3805
Firmware version	FIRMWARE_VERSION	8	R	[261:254]	See Note	[254] = PRV value in CID, [261:255] are all zero

IF THERE IS ANY OTHER OPERATION TO IMPLEMENT IN ADDITION TO SPECIFICATION IN DATASHEET OR JEDEC STANDARD, PLEASE CONTATC EACH NEAREST SALES OFFICE OR HEADQUARTERS OF KOWIN

Name	Field	Byte	Type	Slice	Value	Note
Power class for 200MHz, DDR at VCC=3.6V	PWR_CL_DDR_200_360	1	R	[253]	0h	
Cache size	CACHE_SIZE	4	R	[252:249]	0400h	
Generic CMD6 timeout	GENERIC_CMD6_TIME	1	R	[248]	5h	
Power off notification (long) timeout	POWER_OFF_LONG_TIME	1	R	[247]	64h	
Background operations status	BKOPS_STATUS	1	R	[246]	0h	
Number of correctly programmed sectors	CORRECTLY_PRG_SECTORS_NUM	4	R	[245:242]	0h	
First initialization time after partitioning	INI_TIMEOUT_AP	1	R	[241]	0Ah	
Cache Flushing Policy	CACHE_FLUSH_POLICY	1	R	[240]	1h	
Power class for 52MHz, DDR at 3.6V	PWR_CL_DDR_52_360	1	R	[239]	0h	
Power class for 52MHz, DDR at 1.95V	PWR_CL_DDR_52_195	1	R	[238]	0h	
Power class for 200MHz at VCCQ=1.95V, VCC=3.6V	PWR_CL_200_195	1	R	[237]	0h	
Power class for 200MHz, at VCCQ=1.3V, VCC=3.6V	PWR_CL_200_130	1	R	[236]	0h	
Minimum write performance for 8bit at 52 MHz in DDR mode	MIN_PERF_DDR_W_8_52	1	R	[235]	0h	
Minimum read performance for 8bit at 52 MHz in DDR mode	MIN_PERF_DDR_R_8_52	1	R	[234]	0h	
Reserved	-	1	-	[233]	-	
TRIM multiplier	TRIM_MULT	1	R	[232]	02h	
Secure feature support	SEC_FEATURE_SUPPORT	1	R	[231]	55h	
Secure erase multiple	SEC_ERASE_MULT	1	R	[230]	See Note	4G: FFh 8G: FFh
Secure trim multiple	SEC_TRIM_MULT	1	R	[229]	FFh	
Boot information	BOOT_INFO	1	R	[228]	7h	
Reserved	-	1	-	[227]	-	

IF THERE IS ANY OTHER OPERATION TO IMPLEMENT IN ADDITION TO SPECIFICATION IN DATASHEET OR JEDEC STANDARD, PLEASE CONTATC EACH NEAREST SALES OFFICE OR HEADQUARTERS OF KOWIN

Name	Field	Byte	Type	Slice	Value	Note
Boot partition size	BOOT_SIZE_MULTI	1	R	[226]	20h	
Access size	ACC_SIZE	1	R	[225]	6h	
High-capacity erase unit size	HC_ERASE_GRP_SIZE	1	R	[224]	1h	
High-capacity erase timeout	ERASE_TIMEOUT_MULT	1	R	[223]	2h	
Reliable write sector count	REL_WR_SEC_C	1	R	[222]	1h	
High-capacity write protect group size	HC_WP_GRP_SIZE	1	R	[221]	see Note	≦64GB: 0x10. Others: 0x20
Sleep current (VCC)	S_C_VCC	1	R	[220]	7h	
Sleep current (VCCQ)	S_C_VCCQ	1	R	[219]	7h	
Production state awareness timeout	PRODUCTION_STATE_AWARENESS_TIMEOUT	1	R	[218]	17h	
Sleep/awake timeout	S_A_TIMEOUT	1	R	[217]	13h	
Sleep Notification Timeout	SLEEP_NOTIFICATION_TIME	1	R	[216]	0Ch	
Sector count	SEC_COUNT	4	R	[215:212]	see Note	4GB:0x720000 8GB: 0xE68000
Secure Write Protect Information	SECURE_WP_INFO	1	R	[211]	1h	
Minimum write performance for 8bit at 52MHz	MIN_PERF_W_8_52	1	R	[210]	0h	
Minimum read performance for 8bit at 52MHz	MIN_PERF_R_8_52	1	R	[209]	0h	
Minimum write performance for 8bit at 26MHz, for 4bit at 52MHz	MIN_PERF_W_8_26_4_52	1	R	[208]	0h	
Minimum read performance for 8bit at 26MHz, for 4bit at 52MHz	MIN_PERF_R_8_26_4_52	1	R	[207]	0h	
Minimum write performance for 4bit at 26MHz	MIN_PERF_W_4_26	1	R	[206]	0h	
Minimum read performance for 4bit at 26MHz	MIN_PERF_R_4_26	1	R	[205]	0h	
Reserved	-	1	-	[204]	-	
Power class for 26MHz at 3.6V	PWR_CL_26_360	1	R	[203]	0h	
Power class for 52MHz at 3.6V	PWR_CL_52_360	1	R	[202]	0h	
Power class for 26MHz at 1.95V	PWR_CL_26_195	1	R	[201]	0h	
Power class for 52MHz at 1.95V	PWR_CL_52_195	1	R	[200]	0h	

IF THERE IS ANY OTHER OPERATION TO IMPLEMENT IN ADDITION TO SPECIFICATION IN DATASHEET OR JEDEC STANDARD, PLEASE CONTATC EACH NEAREST SALES OFFICE OR HEADQUARTERS OF KOWIN

Name	Field	Byte	Type	Slice	Value	Note
Partition switching timing	PARTITION_SWITCH_TIME	1	R	[199]	6h	
Out-of-interrupt busy timing	OUT_OF_INTERRUPT_TIME	1	R	[198]	5h	
I/O Driver Strength	DRIVER_STRENGTH	1	R	[197]	1Fh	
Card type	CARD_TYPE	1	R	[196]	57h	
Reserved	-	1	-	[195]	-	
CSD STRUCTURE	CSD_STRUCTURE	1	R	[194]	2h	
Reserved	-	1	-	[193]	-	
Extended CSD revision	EXT_CSD_REV	1	R	[192]	8h	
Command set	CMD_SET	1	R/W/E_P	[191]	0h	
Reserved	-	1	-	[190]	-	
Command set revision	CMD_SET_REV	1	R	[189]	0h	
Reserved	-	1	-	[188]	-	
Power class	POWER_CLASS	1	R/W/E_P	[187]	0h	
Reserved	-	1	-	[186]	-	
High-speed interface timing	HS_TIMING	1	R/W/E_P	[185]	1h	
Strobe Support	STROBE_SUPPORT	1	R	[184]	1h	
Bus width mode	BUS_WIDTH	1	W/E_P	[183]	1h	
Reserved	-	1	-	[182]	-	
Erased memory content	ERASED_MEM_CONT	1	R	[181]	0h	
Reserved	-	1	-	[180]	-	
Partition configuration	PARTITION_CONFIG	1	R/W/E R/W/E_P	[179]	0h	
Boot configuration protection	BOOT_CONFIG_PROT	1	R/W R/W/C_P	[178]	0h	
Boot bus conditions	BOOT_BUS_CONDITIONS	1	R/W/E	[177]	0h	
Reserved	-	1	-	[176]	-	
High-density erase group definition	ERASE_GROUP_DEF	1	R/W/E	[175]	0h	
Boot write protection status regist ers	BOOT_WP_STATUS	1	R	[174]	0h	
Boot area write protection register	BOOT_WP	1	R/W R/W/C_P	[173]	0h	
Reserved	-	1	-	[172]	-	
User area write protection register	USER_WP	1	R/W R/W/C_P R /W/E_P	[171]	0h	
Reserved	-	1		[170]	-	
FW configuration	FW_CONFIG	1	R/W	[169]	0h	
RPMB size	RPMB_SIZE_MULT	1	R	[168]	20h	
Write reliability setting register	WR_REL_SET	1	R/W	[167]	1Fh	
Write reliability parameter register	WR_REL_PARAM	1	R	[166]	15h	

IF THERE IS ANY OTHER OPERATION TO IMPLEMENT IN ADDITION TO SPECIFICATION IN DATASHEET OR JEDEC STANDARD, PLEASE CONTATC EACH NEAREST SALES OFFICE OR HEADQUARTERS OF KOWIN

Name	Field	Byte	Type	Slice	Value	Note
Start Sanitize operation	SANITIZE_START	1	W/E_P	[165]	0h	
Manually start background operations	BKOPS_START	1	W/E_P	[164]	0h	
Enable background operations handshake	BKOPS_EN	1	R/W	[163]	0h	
H/W reset function	RST_n_FUNCTION	1	R/W	[162]	0h	
Reserved	-	1	-	[161]	0h	
Partitioning support	PARTITIONING_SUPPORT	1	R	[160]	7h	
Max enhanced area size	MAX_ENH_SIZE_MULT	3	R	[159:157]	see Note	*4GB:TBD* 8GB:0x1CD
Partitions attribute	PARTITIONS_ATTRIBUTE	1	R/W	[156]	0h	
Partitioning setting	PARTITION_SETTING_COMPLETED	1	R/W	[155]	0h	
General purpose partition size	GP_SIZE_MULT	12	R/W	[154:143]	0h	
Enhanced user data area size	ENH_SIZE_MULT	3	R/W	[142:140]	0h	
Enhanced user data start address	ENH_START_ADDR	4	R/W	[139:136]	0h	
Reserved	-	1	-	[135]	-	
Secure bad block management	SEC_BAD_BLK_MGMNT	1	R/W	[134]	0h	
Production state awareness	PRODUCTION_STATE_AWARENESS	1	R/W/E	[133]	0h	
Package Case Temperature is controlled	TCASE_SUPPORT	1	W/E_P	[132]	0h	
Periodic Wake-up	PERIODIC_WAKEUP	1	R/W/E	[131]	0h	
Program CID/CSD in DDR mode support	PROGRAM_CID_CSD_DDR_SUPPORT	1	R	[130]	0h	
Reserved	-	2	TBD	[129:128]	-	
Vendor Specific Fields	VENDOR_SPECIFIC_FIELD	64	<vendor specific>	[127:64]	see Note	Internal Used.
Native sector size	NATIVE_SECTOR_SIZE	1	R	[63]	1h	
Sector size emulation	USE_NATIVE_SECTOR	1	R/W	[62]	0h	
Sector size	DATA_SECTOR_SIZE	1	R	[61]	0h	
1st initialization after disabling sector size emulation	INI_TIMEOUT_EMU	1	R	[60]	0Ah	
Class 6 commands control	CLASS_6_CTRL	1	R/W/E_P	[59]	0h	
Number of addressed group to be Released	DYNCAP_NEEDED	1	R	[58]	0h	
Exception events control	EXCEPTION_EVENTS_CTRL	2	R/W/E_P	[57:56]	00h	
Exception events status	EXCEPTION_EVENTS_STATUS	2	R	[55:54]	00h	

Name	Field	Byte	Type	Slice	Value	Note
Extended Partitions Attribute	EXT_PARTITIONS_ATTRIBUTE	2	R/W	[53:52]	00h	
Context configuration	CONTEXT_CONF	15	R/W/E_P	[51:37]	0h	
Packed command status	PACKED_COMMAND_STATUS	1	R	[36]	0h	
Packed command failure index	PACKED_FAILURE_INDEX	1	R	[35]	0h	
Power Off Notification	POWER_OFF_NOTIFICATION	1	R/W/E_P	[34]	0h	
Control to turn the Cache ON/OFF	CACHE_CTRL	1	R/W/E_P	[33]	0h	
Flushing of the cache	FLUSH_CACHE	1	W/E_P	[32]	0h	
Control to turn the Barrier ON/OFF	BARRIER_CTRL	1	R/W	[31]	0h	
Mode config	MODE_CONFIG	1	R/W/E_P	[30]	0h	
Mode operation codes	MODE_OPERATION_CODES	1	W/E_P	[29]	0h	
Reserved	-	1	-	[28:27]	-	
FFU status	FFU_STATUS	1	R	[26]	0h	
Pre loading data size	PRE_LOADING_DATA_SIZE	4	R/W/E_P	[25:22]	0h	
Max pre loading data size	MAX_PRE_LOADING_DATA_SIZE	4	R	[21:18]	see Note	4GB:0x720000 8G:0xE68000
Product state awareness enablement	PRODUCT_STATE_AWARENESS_ENABLEMENT	1	R/W/E & R	[17]	01h	
Secure Removal Type	SECURE_REMOVAL_TYPE	1	R/W & R	[16]	3Bh	
Command Queue Mode Enable	CMDQ_MODE_EN	1	R/W/E_P	[15]	0h	
Reserved	-	15	-	[14:0]	-	

IF THERE IS ANY OTHER OPERATION TO IMPLEMENT IN ADDITION TO SPECIFICATION IN DATASHEET OR JEDEC STANDARD, PLEASE CONTATC EACH NEAREST SALES OFFICE OR HEADQUARTERS OF KOWIN

[Table 16] Timing Parameter(Time out)

Timing Parameter		Max.Value	Unit
Initialization Timeout(tINIT)	Normal [1]	1	S
	After partition setting [2]	1	S
Read Timeout		150	ms
Write Timeout		300	ms
Erase Timeout		600	ms
Force Erase Timeout		180000	ms
Secure Erase Timeout		60	S
Secure Trim Step1 Timeout		60	S
Secure Trim Step2 Timeout		60	S
Trim Timeout		600	ms
Partition Switching Timeout(after Init.)		50000	us
Power Off Notification (Short) Timeout		50	ms
Power Off Notification (Long) Time Out		1000	ms

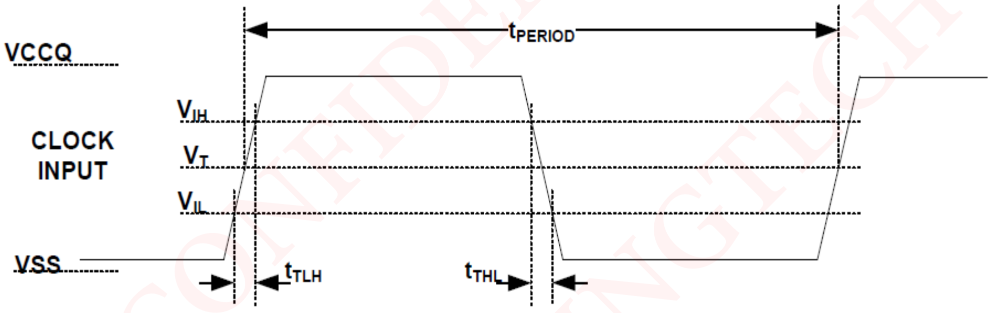
[Note]
1) Normal Initialization Time without partition setting
2) Initialization Time after partition setting, refer to INI_TIMEOUT_AP in 6.4 EXT_CSD register
3) Timeout value are measured based on KOWIN test pattern
4) Under server user case EXCEPTION_EVENT occurs and Timeout values have possibility to increase

7.2 Bus Timing Specification in HS200 mode

7.2.1 Previous Bus Timing Parameters for DDR52 are defined by JEDEC standard

7.2.2 HS200 Clock Timing

Host CLK Timing in HS200 mode shall conform to the timing specified in Figure 9 and Table 17.
CLK input shall satisfy the clock timing over all possible operation and environment conditions. CLK input parameters should be measured while CMD and DAT lines are stable high or low, as close as possible to the Device.
The maximum frequency of HS200 is 200 MHz. Hosts can use any frequency up to the maximum that HS200 mode allows.



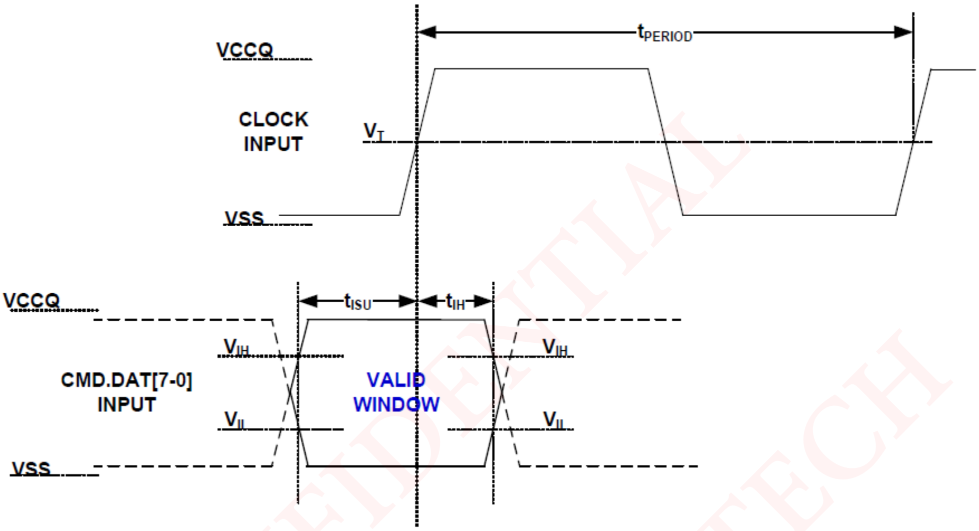
NOTE 1 V_{IH} denote V_{IH}(min.) and V_{IL} denotes V_{IL}(max.).
NOTE 2 $V_T = 50\%$ of V_{CCQ}, indicates clock reference point for timing measurements.

[Figure 9] HS200 Clock signal timing

[Table 17] HS200 Clock signal timing

Symbol	Min	Max	Unit	Remark
tPERIOD	5	-	ns	200 MHz (max), between rising edges
tTLH, tTHL	-	0.2· tPERIOD	ns	tTLH, tTHL < 1ns (max) at 200 MHz, CDEVICE = 6 pF, The absolute maximum value of tTLH, tTHL is 10ns regardless of clock frequency.
Duty Cycle	30	70	%	

7.2.3 HS200 Device Input Timing



NOTE 1 t_{ISU} and t_{IH} are measured at V_{IL}(max) and V_{IH}(min).

NOTE 2 V_{IH} denote V_{IH}(min) and V_{IL} denotes V_{IL}(max).

[Figure 10] HS200 Device input timing

[Table 18] HS200 Device input timing

Symbol	Min	Max	Unit	Remark
tISU	1.40	-	ns	CDEVICE ≤6 pF
tIH	0.8		ns	CDEVICE ≤6 pF

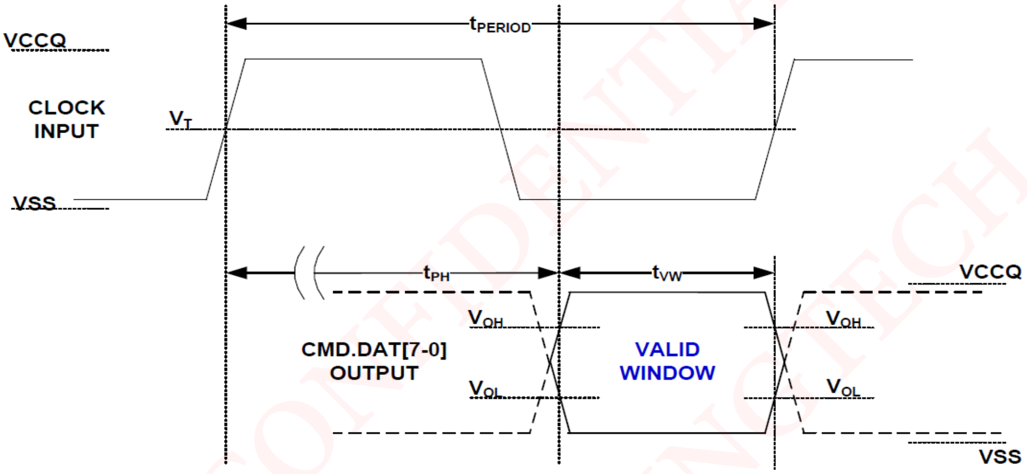
IF THERE IS ANY OTHER OPERATION TO IMPLEMENT IN ADDITION TO SPECIFICATION IN DATASHEET OR JEDEC STANDARD, PLEASE CONTATC EACH NEAREST SALES OFFICE OR HEADQUARTERS OF KOWIN

7.2.4 HS200 Device Output Timing

tPH parameter is defined to allow device output delay to be longer than tPERIOD. After initialization, the tPH may have random phase relation to the clock. The Host is responsible to find the optimal sampling point for the Device outputs, while switching to the HS200 mode.

Figure 11 and Table 19 define Device output timing.

While setting the sampling point of data, a long term drift, which mainly depends on temperature drift, should be considered. The temperature drift is expressed by ΔTPH. Output valid data window (tVW) is available regardless of the drift (ΔTPH) but position of data window varies by the drift, as describes in Figure 12.



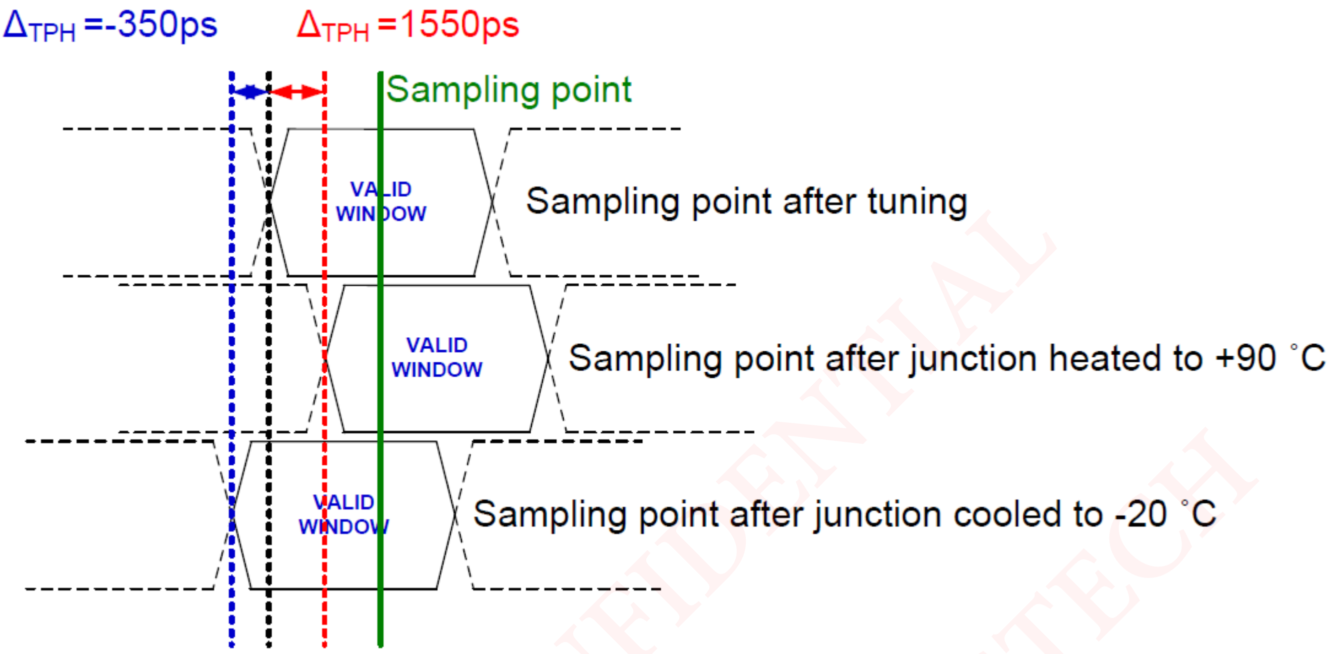
NOTE V_{OH} denotes V_{OH}(min) and V_{OL} denotes V_{OL}(max).

[Figure 11] HS200 Device output timing

[Table 19] HS200 Output timing

Symbol	Min	Max	Unit	Remark
tPH	0	2	UI	Device output momentary phase from CLK input to CMD or DAT lines output. Does not include a long term temperature drift.
ΔTPH	-350 (ΔT= -40 °C)	+1550 (ΔT= 90 °C)	ps	Delay variation due to temperature change after tuning. Total allowable shift of output valid window (TVW) from last system Tuning procedure ΔTPH is 2600ps for ΔT from -40 °C to 125 °C during operation.
tVW	0.575	-	UI	tVW = 2.88ns at 200 MHz Using test circuit in JEDEC - Outputs test circuit for rise/fall time measurement - including skew among CMD and DAT lines created by the Device. Host path may add Signal Integrity induced noise, skews, etc. Expected TVW at Host input is larger than 0.475UI.

NOTE Unit Interval (UI) is one bit nominal time. For example, UI=5ns at 200 MHz


ΔTPH =-350ps
ΔTPH =1550ps
Sampling point
VALID WINDOW
Sampling point after tuning
VALID WINDOW
Sampling point after junction heated to +90 ˚C
VALID WINDOW
Sampling point after junction cooled to -20 ˚C

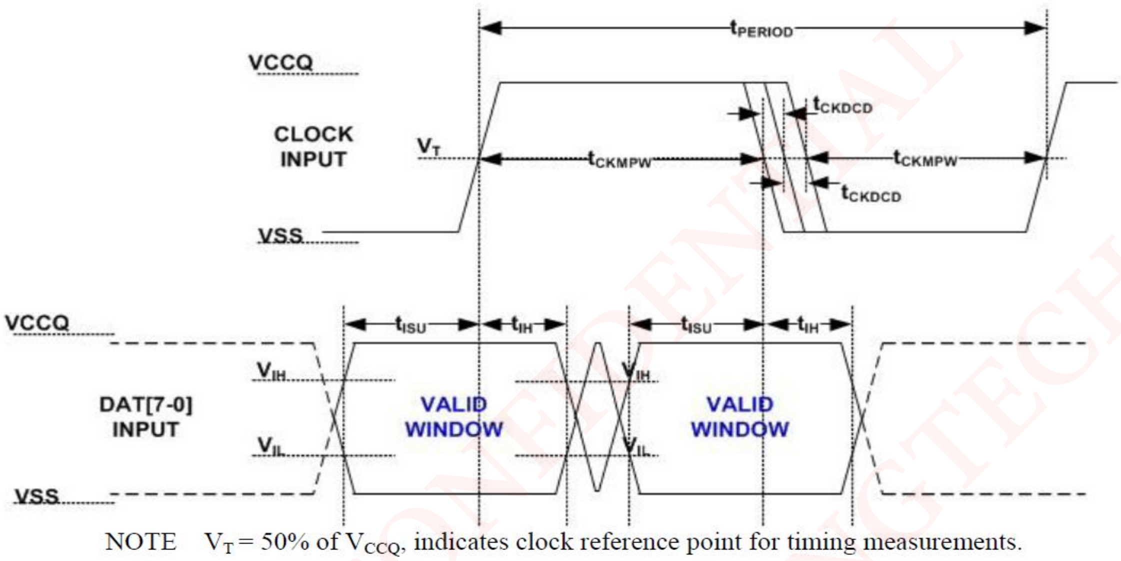
[Figure 12] ΔTPH consideration

Implementation Guide:
Host should design to avoid sampling errors that may be caused by the ΔTPH drift.
It is recommended to perform tuning procedure while Device wakes up, after sleep.
One simple way to overcome the ΔTPH drift is by reduction of operating frequency.

7.3 Bus Timing Specification in HS400 mode

7.3.1 HS400 Device Input Timing

The CMD input timing for HS400 mode is the same as CMD input timing for HS200 mode. Figure 13 and Table 20 show Device input timing.



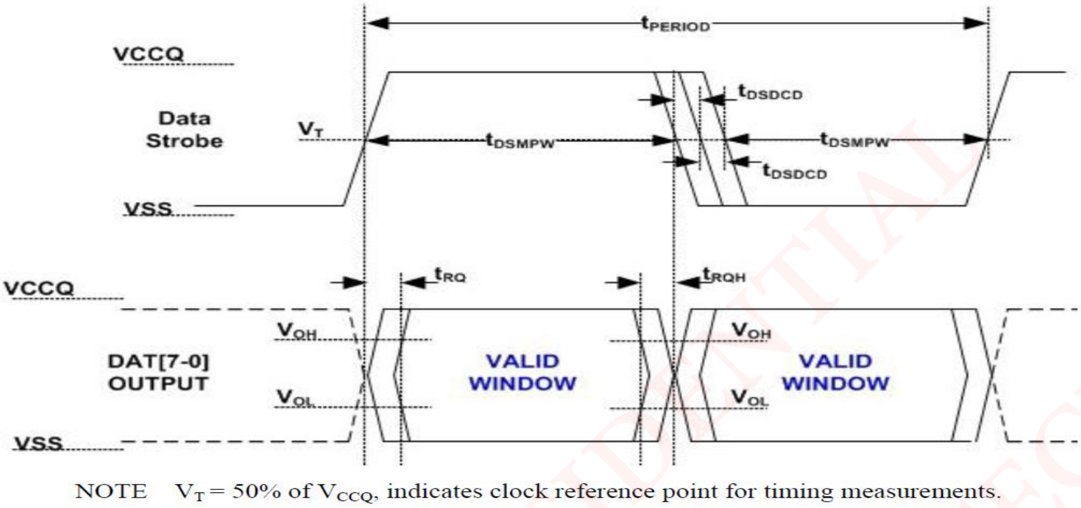
[Figure 13] HS400 Device Data input timing

[Table 20] HS400 Device input timing

Parameter	Symbol	Min	Max	Unit	Remark
Input CLK					
Cycle time data transfer mode	tPERIOD	5			200 MHz(max), between rising edges With respect to VT.
Slew rate	SR	1.125		V/ns	With respect to VIH/VIL.
Duty cycle distortion	tCKDCD	0.0	0.3	ns	Allowable deviation from an ideal 50% duty cycle. With respect to VT. Includes jitter, phase noise
Minimum pulse width	tCKMPW	2.2		ns	With respect to VT.
Input DAT (referenced to CLK)					
Input set-up time	tISUddr	0.4		ns	CDevice ≤ 6 pF With respect to VIH/VIL.
Input hold time	tIHddr	0.4		ns	CDevice ≤ 6 pF With respect to VIH/VIL.
Slew rate	SR	1.125		V/ns	With respect to VIH/VIL.

7.3.2 HS400 Device Output Timing

The Data Strobe is used to read data in HS400 mode. The Data Strobe is toggled only during data read or CRC status response.



NOTE V_T = 50% of V_{CCQ}, indicates clock reference point for timing measurements.

[Figure 14] HS400 Device output timing

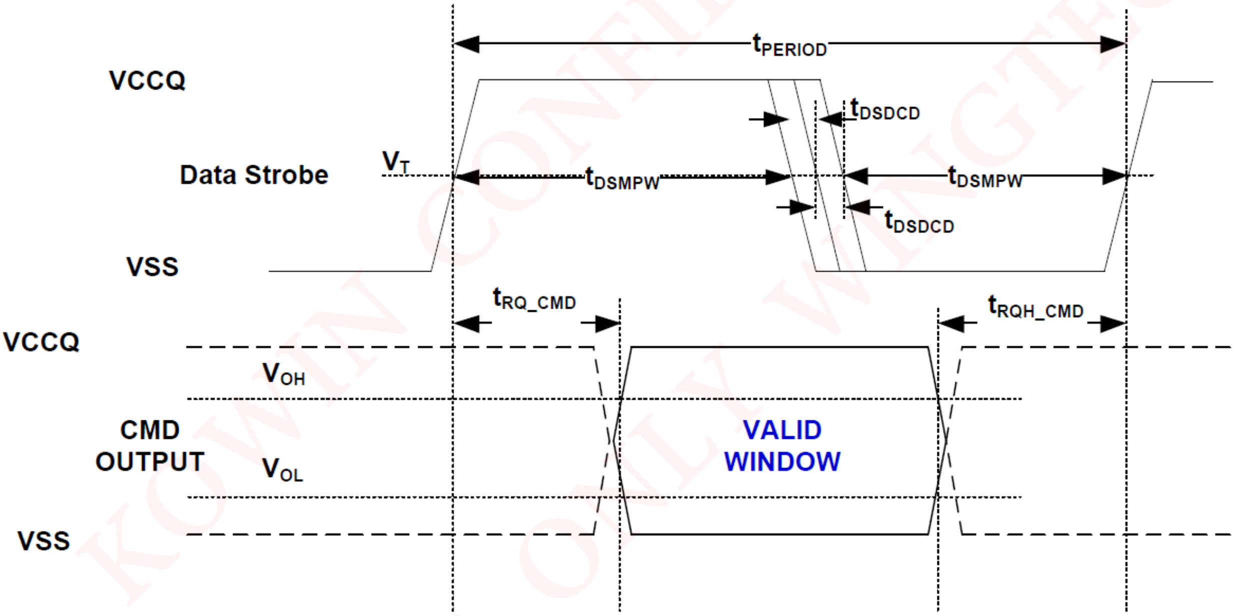
[Table 21] HS400 Device input timing

Parameter	Symbol	Min	Max	Unit	
Data Strobe					
Cycle time data transfer mode	tPERIOD	5			200 MHz(max), between rising edges With respect to VT
Slew rate	SR	1.125		V/ns	With respect to VOH/VOL and HS400 reference load
Duty cycle distortion	tDSDCD	0.0	0.2	ns	Allowable deviation from the input CLK duty cycle distortion (tCKDCD) With respect to VT Includes jitter, phase noise
Minimum pulse width	tDSMPW	2.0		ns	With respect to VT
Read pre-amble	tRPRE	0.4	-	tPERIOD	Max value is specified by manufacturer. Value up to infinite is valid
Read post-amble	tRPST	0.4	-	tPERIOD	Max value is specified by manufacturer. Value up to infinite is valid
Output DAT (referenced to Data Strobe)					
Output skew	tRQ		0.4	ns	With respect to VOH/VOL and HS400 reference load
Output hold skew	tRQH		0.4	ns	With respect to VOH/VOL and HS400 reference load.
Slew rate	SR	1.125		V/ns	With respect to VOH/VOL and HS400 reference load

[Table 22] HS400 Capacitance and Resistors

Parameter	Symbol	Min	Type	Max	Unit	Remark
Pull-up resistance for CMD	RCMD	4.7		100 (1)	kΩ	
Pull-up resistance for DAT0-7	RDAT	10		100 (1)	kΩ	
Pull-down resistance for Data Strobe	RDS	10		100 (1)	kΩ	
Internal pull up resistance DAT1-DAT7	Rint	10		150	kΩ	
Single Device capacitance	CDevice			6	pF	
NOTE 1 Recommended maximum value is 30 kΩ for 1.2 V and 50 kΩ for 1.8 V interface supply voltages.						

7.3.3 HS400 Device Command Output Timing



NOTE V_T = 50% of V_{CCQ}, indicates clock reference point for timing measurements.

[Figure 15] HS400 CMD Response timing

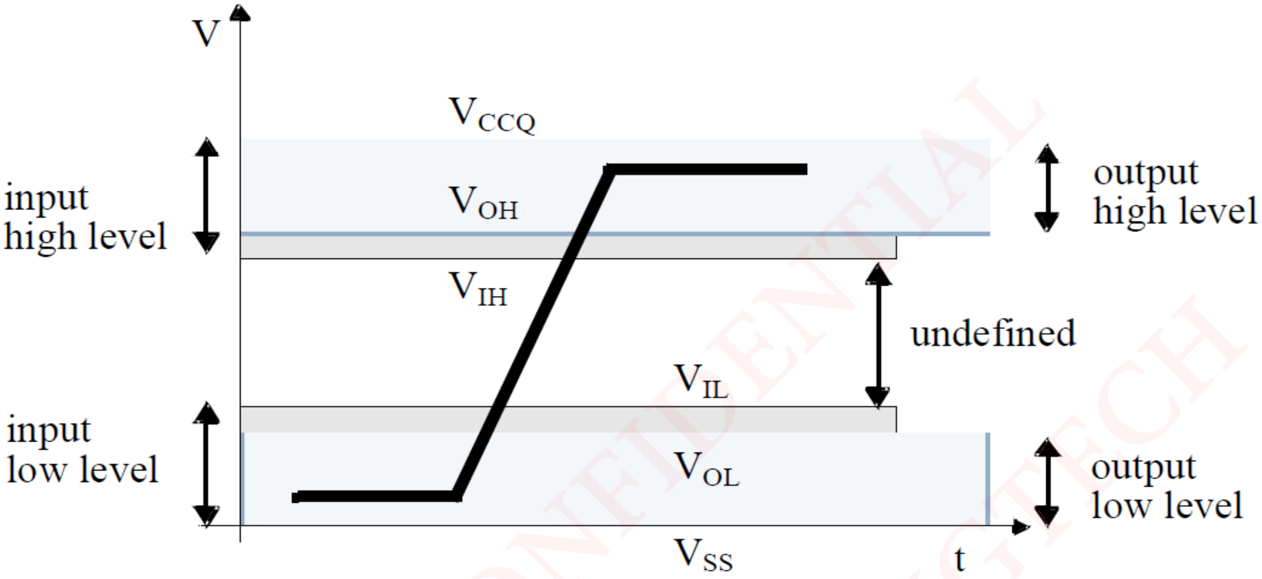
IF THERE IS ANY OTHER OPERATION TO IMPLEMENT IN ADDITION TO SPECIFICATION IN DATASHEET OR JEDEC STANDARD, PLEASE CONTATC EACH NEAREST SALES OFFICE OR HEADQUARTERS OF KOWIN

[Table23] HS400 CMD Response timing

Parameter`	Symbol	Min	Max	Unit	Remark
Data Strobe					
Cycle time data transfer mode	tPERIOD	5			200 MHz(max), between rising edges With respect to VT
Slew rate	SR	1.125		V/ns	With respect to VOH/VOL and HS400 reference load
Duty cycle distortion	tDSDCD	0.0	0.2	ns	Allowable deviation from the input CLK duty cycle distortion (tCKDCD) With respect to VT Includes jitter, phase noise
Minimum pulse width	tDSMPW	2.0		ns	With respect to VT
Read pre-amble	tRPRE	0.4	-	tPERIOD	Max value is specified by manufacturer. Value up to infinite is valid
Read post-amble	tRPST	0.4	-	tPERIOD	Max value is specified by manufacturer. Value up to infinite is valid
CMD Response (referenced to Data Strobe)					
Output skew(CMD)	tRQ_CMD		0.4	ns	With respect to VOH/VOL and HS400 reference load
Output hold skew(CMD)	tRQH_CMD		0.4	ns	With respect to VOH/VOL and HS400 reference load
Slew rate	SR	1.125		V/ns	With respect to VOH/VOL and HS400 reference load

7.4 Bus signal levels

As the bus can be supplied with a variable supply voltage, all signal levels are related to the supply voltage.



[Figure 16] Bus signal levels

7.4.1 Open-drain mode bus signal level

[Table 24] Open-drain bus signal level

Parameter	Symbol	Min	Max	Unit	Conditions
Output HIGH voltage	VOH	VCCQ - 0.2		V	NOTE 1
Output LOW voltage	VOL		0.3	V	IOL = 2 mA

NOTE 1 Because Voh depends on external resistance value (including outside the package), this value does not apply as device specification. Host is responsible to choose the external pull-up and open drain resistance value to meet Voh Min value.

The input levels are identical with the push-pull mode bus signal levels.

7.4.2 Push-pull mode bus signal level eMMC

The device input and output voltages shall be within the following specified ranges for any VCCQ of the allowed voltage range

[Table 25] Push-pull signal level—high-voltage eMMC

Parameter	Symbol	Min	Max	Unit	Conditions
Output HIGH voltage	VOH	0.75 * VCCQ		V	IOH = -100 μA @ VCCQ min
Output LOW voltage	VOL		0.125 * VCCQ	V	IOL = 100 μA @ VCCQ min
Input HIGH voltage	VIH	0.625 * VCCQ	VCCQ + 0.3	V	
Input LOW voltage	VIL	VSS - 0.3	0.25 * VCCQ	V	

[Table 26] Push-pull signal level—1.70 V -1.95 V VCCQ voltage Range

Parameter	Symbol	Min	Max	Unit	Conditions
Output HIGH voltage	VOH	VCCQ - 0.45V		V	IOH = -2mA
Output LOW voltage	VOL		0.45V	V	IOL = 2mA
Input HIGH voltage	VIH	0.65 * VCCQ (1)	VCCQ + 0.3	V	
Input LOW voltage	VIL	VSS - 0.3	0.35 * VCCQ (2)	V	

NOTE 1 0.7 * VDD for MMC4.3 and older revisions.
NOTE 2 0.3 * VDD for MMC4.3 and older revisions.

7.4.3 Bus Operating Conditions for HS200 and HS400

The bus operating conditions for HS200 and HS400 devices is the same as specified in 7.4.1 through 7.4.2. The only exception is that VCCQ = 3.3 V is not supported.

IF THERE IS ANY OTHER OPERATION TO IMPLEMENT IN ADDITION TO SPECIFICATION IN DATASHEET OR JEDEC STANDARD, PLEASE CONTATC EACH NEAREST SALES OFFICE OR HEADQUARTERS OF KOWIN

8.0 DC POWER PARAMETER

8.1 Supply Voltage and Operating Power Condition

[Table 27] Operating Conditions

Parameter	Symbol	Min	Max	Unit
Supply Voltage (NAND)	VCC	2.7	3.6	V
Supply Voltage (I/O)	VCCQ	1.7	1.95	V
		2.7	3.6	V
Vss	Vss	-0.5	0.5	V
Supply Power-up for 3.3V	tPRUH		35	ms
Supply Power-up for 1.8V	tPRUL		25	ms
Operating Temperature	T_A	-40	+85	°C
Junction Temperature	T_J		+125	°C

8.2 Power Consumption in all mode

[Table 28] Power Consumption in all mode

eMMC		Max operating current RMS(100ms) [*1]		Standby [*2]		Sleep [*3]	
Density	Operation	ICC(mA)	ICCQ(mA)	ICC(uA)	ICCQ(uA)	ICC(uA)	ICCQ(uA)
4GB	*Read*	50	70	35	80	30	80
	Write	40	35				
8GB	Read	60	85	35	80	30	80
	Write	50	35				

- NAND type, 4GB: 32Gb x 1, 8GB: 64Gb x 1
- Typical value is measured at Vcc=3,3V, TA=25’C.

[Note1]

- Active Power Measure condition : Bus configuration = x8 @ 200MHz
- The measurement for max RMS current is the average RMS current consumption over a power period of 100ms

[Note2]

- Standby Power Measure condition : Bus configuration = x8 No CLK

[Note3]

- Sleep Power Measure condition : Bus configuration = x8 No CLK
- In auto power saving mode, NAND power is alive
- In sleep mode NAND power can be switched off. If NAND power is alive, it is same with the value in standby state

IF THERE IS ANY OTHER OPERATION TO IMPLEMENT IN ADDITION TO SPECIFICATION IN DATASHEET OR JEDEC STANDARD, PLEASE CONTATC EACH NEAREST SALES OFFICE OR HEADQUARTERS OF KOWIN

9. Bus Signal Line Load

The total capacitance CL of each line of the eMMC bus is the sum of the bus master capacitance CHOST, the bus capacitance CBUS itself, and the capacitance CDEVICE of the Device connected to this line,

CL = CHOST + CBUS + CDEVICE

and requiring the sum of the host and bus capacitances not to exceed 20 pF

[Table 32] Capacitance and Resistors

Parameter	Symbol	Min	Typ	Max	Unit	Remark
Pull-up resistance for CMD	R_{CMD}	4.7		100 (1)	kΩ	to prevent bus floating
Pull-up resistance for DAT0–7	R_{DAT}	10		100 (1)	kΩ	to prevent bus floating
Internal pull up resistance DAT1–DAT7	R_{int}	10		150	kΩ	to prevent unconnected lines floating
Bus signal line capacitance	C_L			30	pF	Single Device
Single Device capacitance	C_{DEVICE}			6	pF	
Maximum signal line inductance				16	nH	fPP ≤ 52 MHz
VDDi capacitor value	C_{REG}(2)	0.1			uF	To stabilize regulator output when target device bus speed mode is either backward-compatible, high speed SDR, high speed DDR, or HS200.
		1			uF	To stabilize regulator output when target device bus speed mode is HS400.
VDDi2 capacitor value (e^2MMC)	C_{REG3}	1			uF	To stabilize internal regulated voltage
VCCQ decoupling capacitor	C_{H1}	1			uF	(3), (4), (5)

NOTE 1 Recommended maximum pull-up is 30 kΩ for 1.2 V and 50 kΩ for 1.8 V interface supply voltages. A 3 V part, may use the whole range up to 100 kΩ.

NOTE 2 Recommended value for CREG, CREG2 and CREG3 might be different between *e*MMC device vendors.

NOTE Confirm the maximum value and the accuracy of the capacitance with *e*MMC vendor because the electrical characteristics of the regulator inside *e*MMC is affected by the fluctuation of the capacitance.

NOTE 3 CH1 is VCCQ-VSSQ decoupling capacitor required for HS200&HS400 *e*MMC device.

NOTE 4 CH1 should be placed adjacent to VCCQ-VSSQ balls (around DAT[7..0] balls), It should be located as close as possible to the balls defined in order to minimize connection parasitics.

NOTE 5 *e*MMC device vendor may have more specific requirements for CH1 placement. Please confirm such requirements with specific *e*MMC device vendor.

IF THERE IS ANY OTHER OPERATION TO IMPLEMENT IN ADDITION TO SPECIFICATION IN DATASHEET OR JEDEC STANDARD, PLEASE CONTATC EACH NEAREST SALES OFFICE OR HEADQUARTERS OF KOWIN

Appendix : Part Number Information

K	A	S	X	X	X	X	X

Digit No.	Description	Value
1	Company Name 1	K
2	Company Name 2	A
3	I/F & Version	**. S : eMMC5.1**
4-5	eMMC Density & Nand Information	. **04** : 4GB eMMC with 1st Gen(32Gb MLC) . **03** : 8GB eMMC with 1st Gen(64Gb MLC)
6	Controller	*Varies*
7	PKG Type/Size	. **1** : 153FBGA(11.5x13x1.0)
8	Operating Temp	. **1** : C-Temp . **D** : Industrial-Temp

IF THERE IS ANY OTHER OPERATION TO IMPLEMENT IN ADDITION TO SPECIFICATION IN DATASHEET OR JEDEC STANDARD, PLEASE CONTATC EACH NEAREST SALES OFFICE OR HEADQUARTERS OF KOWIN