Application Note

Document No.: AN1082

APM32F4xx_SDRAM Application Note

Version: V1.0

1 Introduction

This application note provides a guide on how to configure and apply DMC interface on APM32F4xx series to access SDRAM.

Contents

2 SDRAM Introduction

The full name of SDRAM is Synchronous Dynamic Random Access Memory. Synchronous means that the transmission of data shares a clock line as the benchmark; dynamic means that the storage array needs to be constantly refreshed to ensure that data is not lost; random means that data is not stored in a linear sequence, but is freely assigned with an address for data reading and writing.

2.1 Storage structure

In SDRAM, each row and column address combination corresponds to a storage unit. The size of the storage unit is determined by the specification of SDRAM, and is usually 16 bits. A SDRAM usually has 2 Banks or 4 Banks. The capacity calculation of a SDRAM is: (2 ^ row address bits) x (2 ^ column address bits) x (storage unit size) x (number of Banks).

2.2 Brief description of SDRAM parameters

2.2.1 CAS Latency

CAS refers to the column address strobe signal. Addressing starts from the row address and then the column address. Therefore, after address strobe, the storage unit is determined; next the data will be transmitted. CAS Latency refers to the period of time from the sending of CAS to the first data output. The value of CAS Latency cannot exceed the design specification of the chip; otherwise, the memory will be unstable.

2.2.2 tRCD

tRCD is RAS to CAS Delay, which means the delay from RAS to CAS. RAS is the row address strobe signal, and CAS is the column address strobe signal. This delay is customized according to the response time of the electronic components of the chip storage array.

2.2.3 tRP

After reading and writing the SDRAM, if you want to address another row of the same Bank, you need to close the original valid row and resend the row / column address. The operation of closing an existing work row and preparing to open a new row is Precharge. After issuing the precharge command, it takes a period of time to allow sending RAS signal to open a new work line. This period of time is called tRP (RAS Precharge Time).

2.2.4 tRAS

tRAS, namely, Active to Precharge Command, is the interval from row activation to precharge command.

2.2.5 tWR

tWR, namely, Write Recovery Time, can ensure the reliable writing of data.

2.2.6 Burst length

Burst refers to the mode of continuous data transmission between the adjacent storage units in the same row. The number of storage units involved in continuous transmission is the burst length.

2.2.7 Refresh

Dynamic memory needs to be constantly refreshed to retain data, so it is the most important operation of SDRAM. The refresh operation has a fixed cycle, and the operation is performed on all rows in turn to retain the data in the memory that has not been rewritten for a long time.

How long interval is appropriate between two refreshes? At present, the recognized standard is the upper limit of the effective storage period of the capacitor data in the memory is 64ms, that is, the time to refresh all rows is 64ms, so the refresh speed is: number of rows /64ms.

There are two kinds of refresh operations: Auto Refresh and Self-Refresh. Neither of these two kinds of refresh needs to be provided with row address information externally, because this is an internal automatic operation.

For Auto Refresh, a row refresh counter inside SDRAM will automatically generate row addresses in turn. Because the refresh is specific to all storage banks in a row, there is no need for column addressing, or CAS is valid before RAS. Therefore, Auto Refresh is also called CBR refresh. Because the refresh involves all banks, all banks will stop working in the refresh process, and each refresh takes 9 clock cycles, and then it can enter the normal working state, which means that in these 9 clock cycles, all working instructions need to wait and cannot be executed. After 64ms, the same row will be refreshed again, and the refresh will cycle again and again.

Self-refresh is mainly used for data storage in sleep mode with low power. The most famous application is STR. When issuing the auto refresh command, the CKE will be set to the invalid state, and enter the self-refresh mode. At this time, it will no longer rely on the system clock to work, but perform the refresh operation according to the internal clock. During the self-refresh period, all external signals except CKE are invalid. Only when CKE is enabled again, can it exit the self-refresh mode and enter the normal operation state

3 DMC Introduction

DMC is a dynamic storage controller, which can be connected to an external SDRAM for SDRAM reading and writing. As shown in Figure 1 below, the SDRAM data can be transmitted with AHB bus through DMC.

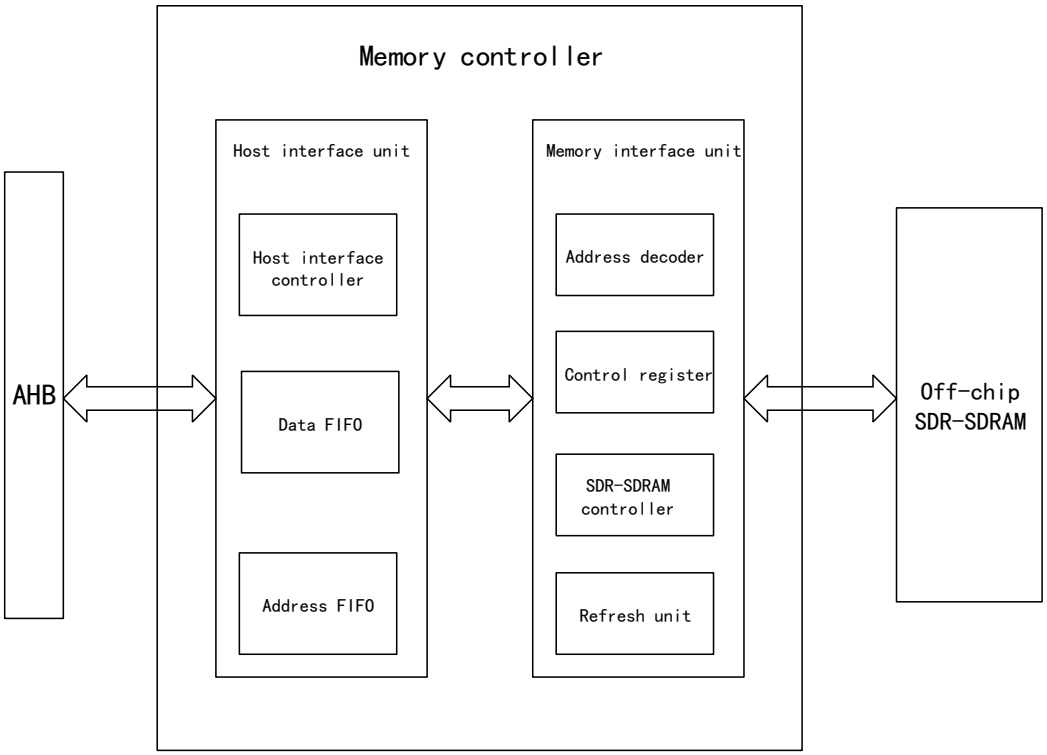

Figure 1 DMC Structure Block Diagram

3.1 Definitions of DMC pins

The definitions of DMC pins are shown in the following table:

Table 1 Definitions of DMC Pins

Signal Name	Input/output	Pin	Function
A0	Output	PF1	Address
A1	Output	PF2	Address
A2	Output	PF3	Address
A3	Output	PF4	Address
A4	Output	PF6	Address

Signal Name	Input/output	Pin	Function
A5	Output	PF7	Address
A6	Output	PF8	Address
A7	Output	PF9	Address
A8	Output	PF10	Address
A9	Output	PH3	Address
A10	Output	PF0	Address
D0	Input/output	PG3	Bidirectional data
D1	Input/output	PG4	Bidirectional data
D2	Input/output	PG5	Bidirectional data
D3	Input/output	PG6	Bidirectional data
D4	Input/output	PG8	Bidirectional data
D5	Input/output	PH13	Bidirectional data
D6	Input/output	PH15	Bidirectional data
D7	Input/output	PI3	Bidirectional data
D8	Input/output	PH8	Bidirectional data
D9	Input/output	PH10	Bidirectional data
D10	Input/output	PD10	Bidirectional data
D11	Input/output	PD12	Bidirectional data
D12	Input/output	PD13	Bidirectional data
D13	Input/output	PD14	Bidirectional data

Signal Name	Input/output	Pin	Function
D14	Input/output	PD15	Bidirectional data
D15	Input/output	PG2	Bidirectional data
BA	Output	PI11	Bank address
CKE	Output	PH5	Clock enable
CLK/CK	Output	PG1	Clock
LDQM	Input	PG15	16-bit data write
UNQM	Input	PF11	16-bit data read
NWE	Output	PI7	Write enable
NCAS	Output	PI8	Column address bit strobe command
NRAS	Output	PI9	Row address bit strobe command
NCS	Output	PI10	Chip selection

The row and column address lines share pins. Currently, APM32F407 supports one bank address line, and can support two banks.

3.2 Initialization of DMC pins

The initialization of DMC pins can refer to the following code example. GPIO needs to configure pin multiplexing:

```
GPIO_Config_T gpioConfig;

RCM_EnableAHB1PeriphClock(RCM_AHB1_PERIPH_GPIOD);

gpioConfig.speed = GPIO_SPEED_50MHz;
gpioConfig.mode = GPIO_MODE_AF;
gpioConfig.otype = GPIO_OTYPE_PP;
gpioConfig.pupd = GPIO_PUPD_NOPULL;
gpioConfig.pin = GPIO_PIN_10;
/** Take the PD10 pin as the configuration reference, and other pins are similar*/
GPIO_Config(GPIOD, &gpioConfig);

/** Pin multiplexing needs to be configured as EMMC*/
GPIO_ConfigPinAF(GPIOD, GPIO_PIN_SOURCE_10, GPIO_AF_EMMC);
```

4 DMC Initialization and SDRAM Access

Flow chart of SDRAM application:

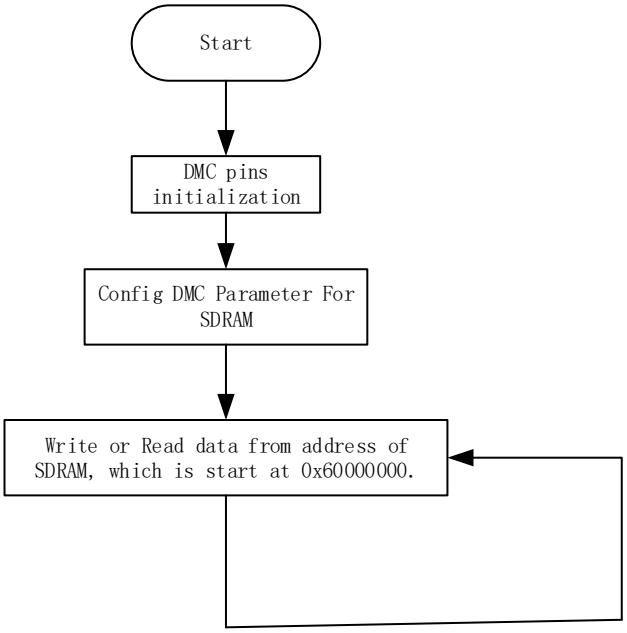

Figure 2 Program Flow Chart

4.1 Initialize DMC

4.1.1 DMC initialization structure

DMC_ Config_ T structure is defined in the document of APM32F4xx_dmc.h. The specific definitions are as follows:

```
/**
 * @brief DMC Config struct
 */
typedef struct
{
    DMC_BANK_WIDTH_T        bankWidth;      //!< Number of bank bits
    DMC_ROW_WIDTH_T         rowWidth;       //!< Number of row address bits
    DMC_COL_WIDTH_T         colWidth;       //!< Number of col address bits
    DMC_CLK_PHASE_T         clkPhase;       //!< Clock phase
    DMC_TimingConfig_T      timing;         //!< Timing
}DMC_Config_T;
```

Among the structure members:

bankWidth indicates the width of the bank address. When it is set to 1, it can support 2 banks, and when set to 2, it can support 4 banks;

rowWidth is the width of the row address

colWidth is the width of the column address

clkPhase is the clock phase

timing is the timing configuration of DMC, corresponding to DMC_ TimingConfig_ T structure members.

At present, the DMC of APM32F407 supports 11-bit row address, 8-bit column address, and 1-bit bank. When the storage unit is 16bit data, namely, 2 bytes, the DMC supports a capacity of up to 2^11 * 2^8 * 2 * 2 = 2Mbytes.

timing type is DMC_ TimingConfig_ T structure, and is defined in the document of APM32F4xx_dmc.h. The specific definitions are as follows:

```
/**
 * @brief Timing config definition
 */
typedef struct
{
    uint32_t    latencyCAS   : 2;        //!< DMC_CAS_LATENCY_T
    uint32_t    tRAS         : 4;        //!< DMC_RAS_MINIMUM_T
    uint32_t    tRCD         : 3;        //!< DMC_DELAY_TIME_T
    uint32_t    tRP          : 3;        //!< DMC_PRECHARGE_T
    uint32_t    tWR          : 2;        //!< DMC_NEXT_PRECHARGE_T
    uint32_t    tARP         : 4;        //!< DMC_AUTO_REFRESH_T
    uint32_t    tCMD         : 4;        //!< DMC_ATA_CMD_T
    uint32_t    tXSR         : 9;    //!< auto-refresh commands, can be 0x000 to 0x1FF
    uint16_t    tRFP         : 16;   //!< Refresh period, can be 0x0000 to 0xFFFF
}DMC_TimingConfig_T;
```

Among the structure members:

latencyCAS is the wait time of CAS;

tRAS is the minimum time between row activation and precharge;

tRCD is the delay time from RAS to CAS;

tRP is the precharge cycle;

tWR is the time from writing the last data to the next precharge;

tARP is the minimum time interval between two auto refresh commands;

tCMD is Active to active command cycle;

tXSR is the minimum time interval between exiting self-refresh and switching to active command or auto refresh read command;

tRFP is the interval between two continuous refreshes;

The unit of the above parameters is DMC clock cycle. The clock of DMC is obtained from AHB clock after DMC frequency division.

4.1.2 DMC configuration

DMC configuration can refer to the following codes:

```
uint32_t sdramCapacity;
DMC_Config_T dmcConfig;
DMC_TimingConfig_T timingConfig;

RCM_EnableAHB3PeriphClock(RCM_AHB3_PERIPH_EMMC);

timingConfig.latencyCAS   = DMC_CAS_LATENCY_3;
timingConfig.tARP         = DMC_AUTO_REFRESH_10;
timingConfig.tRAS         = DMC_RAS_MINIMUM_2;
timingConfig.tCMD         = DMC_ATA_CMD_1;
timingConfig.tRCD         = DMC_DELAY_TIME_1;
timingConfig.tRP          = DMC_PRECHARGE_1;
timingConfig.tWR          = DMC_NEXT_PRECHARGE_2;
timingConfig.tXSR         = 3;
timingConfig.tRFP         = 0x2F9;
dmcConfig.bankWidth       = DMC_BANK_WIDTH_1;
dmcConfig.clkPhase        = DMC_CLK_PHASE_REVERSE;
dmcConfig.rowWidth        = DMC_ROW_WIDTH_11;
dmcConfig.colWidth        = DMC_COL_WIDTH_8;
dmcConfig.timing          = timingConfig;

DMC_Config(&dmcConfig); //!< Configure DMC
DMC_EnableAccelerateModule(); //!<Turning on DMC buffer can improve SDRAM reading performance

DMC_Enable(); //!< Enable DMC
```

When configuring timing parameters, parameters shall be reasonably configured according to the datasheet of SDRAM, and the clock frequency allocated by the current system clock to DMC peripherals. The parameter configuration examples in the above code are for reference only.

4.2 SDRAM access

The start address of SDRAM is 0x60000000. Its end address corresponds to it according to the capacity. For example, the end address of SDRAM with a capacity of 2Mbytes is 0x60200000. In the program, we can access SDRAM directly through the address without unlocking.

For example, the following statement can be used to read data from the address 0x60000000 of SDARM, with read size of 1 byte:

```
uint8_t data = *(uint8_t*) (0x6000000);
```

For example, the following statement can be used to write a 0x55 byte to the address 0x60000000 of SDARM:

```
*(uint8_t*) (0x6000000) = 0x55;
```

5 Version History

Table 2 Document Version History

Date	Version	Change History
May 31, 2022	1.0	New

Statement

This document is formulated and published by Geehy Semiconductor Co., Ltd. (hereinafter referred to as “Geehy”). The contents in this document are protected by laws and regulations of trademark, copyright and software copyright. Geehy reserves the right to make corrections and modifications to this document at any time. Please read this document carefully before using Geehy products. Once you use the Geehy product, it means that you (hereinafter referred to as the “users”) have known and accepted all the contents of this document. Users shall use the Geehy product in accordance with relevant laws and regulations and the requirements of this document.

1. Ownership

This document can only be used in connection with the corresponding chip products or software products provided by Geehy. Without the prior permission of Geehy, no unit or individual may copy, transcribe, modify, edit or disseminate all or part of the contents of this document for any reason or in any form.

The “极海” or “Geehy” words or graphics with “®” or “™” in this document are trademarks of Geehy. Other product or service names displayed on Geehy products are the property of their respective owners.

2. No Intellectual Property License

Geehy owns all rights, ownership and intellectual property rights involved in this document.

Geehy shall not be deemed to grant the license or right of any intellectual property to users explicitly or implicitly due to the sale or distribution of Geehy products or this document.

If any third party’s products, services or intellectual property are involved in this document, it shall not be deemed that Geehy authorizes users to use the aforesaid third party’s products, services or intellectual property, unless otherwise agreed in sales order or sales contract.

3. Version Update

Users can obtain the latest document of the corresponding models when ordering Geehy products.

If the contents in this document are inconsistent with Geehy products, the agreement in thesales order or the sales contract shall prevail.

4. Information Reliability

The relevant data in this document are obtained from batch test by Geehy Laboratory or cooperative third-party testing organization. However, clerical errors in correction or errors caused by differences in testing environment may occur inevitably. Therefore, users should understand that Geehy does not bear any responsibility for such errors that may occur in this document. The relevant data in this document are only used to guide users as performance parameter reference and do not constitute Geehy’s guarantee for any product performance.

Users shall select appropriate Geehy products according to their own needs, and effectively verify and test the applicability of Geehy products to confirm that Geehy products meet their own needs, corresponding

standards, safety or other reliability requirements. If loses are caused to users due to the user's failure to fully verify and test Geehy products, Geehy will not bear any responsibility.

5. Legality

USERS SHALL ABIDE BY ALL APPLICABLE LOCAL LAWS AND REGULATIONS WHEN USING THIS DOCUMENT AND THE MATCHING GEEHY PRODUCTS. USERS SHALL UNDERSTAND THAT THE PRODUCTS MAY BE RESTRICTED BY THE EXPORT, RE-EXPORT OR OTHER LAWS OF THE COUNTIRIES OF THE PRODUCTS SUPPLIERS, GEEHY, GEEHY DISTRIBUTORS AND USERS. USERS (ON BEHALF OR ITSELF, SUBSIDIARIES AND AFFILIATED ENTERPRISES) SHALL AGREE AND PROMISE TO ABIDE BY ALL APPLICABLE LAWS AND REGULATIONS ON THE EXPORT AND RE-EXPORT OF GEEHY PRODUCTS AND/OR TECHNOLOGIES AND DIRECT PRODUCTS.

6. Disclaimer of Warranty

THIS DOCUMENT IS PROVIDED BY GEEHY "AS IS" AND THERE IS NO WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE, TO THE EXTENT PERMITTED BY APPLICABLE LAW.

GEEHY WILL BEAR NO RESPONSIBILITY FOR ANY DISPUTES ARISING FROM THE SUBSEQUENT DESIGN OR USE BY USERS.

7. Limitation of Liability

IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL GEEHY OR ANY OTHER PARTY WHO PROVIDE THE DOCUMENT "AS IS", BE LIABLE FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, DIRECT, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE DOCUMENT (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY USERS OR THIRD PARTIES).

8. Scope of Application

The information in this document replaces the information provided in all previous versions of the document.