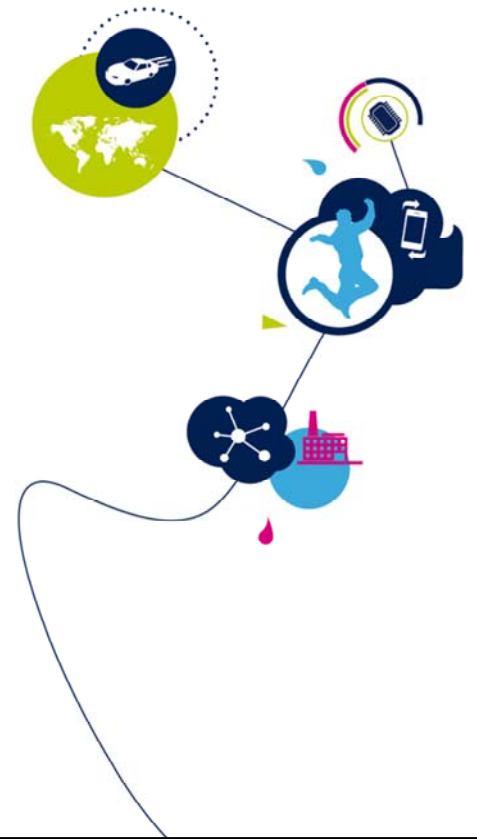


# STM32MP1 – DDRSS

DDR Subsystem  
Revision 1.0



Hello, and welcome to this presentation of the STM32MP1  
DDR memory subsystem.

- Multi Standard: DDR3/3L, LPDDR2/3 (\*)
- Frequency:
  - DDR3/3L  $\leq$  533MHz (1066Mbps)
  - LPDDR2/3  $\leq$  533MHz (1066Mbps)
- Interface Width: either fully populated (x32) or half populated (x16)
- Density limited to 1GByte due to the memory map address range
- Single rank only

(\*) LPDDR3 is supported without On Die Terminal (ODT)



The key features of the DDR subsystem are

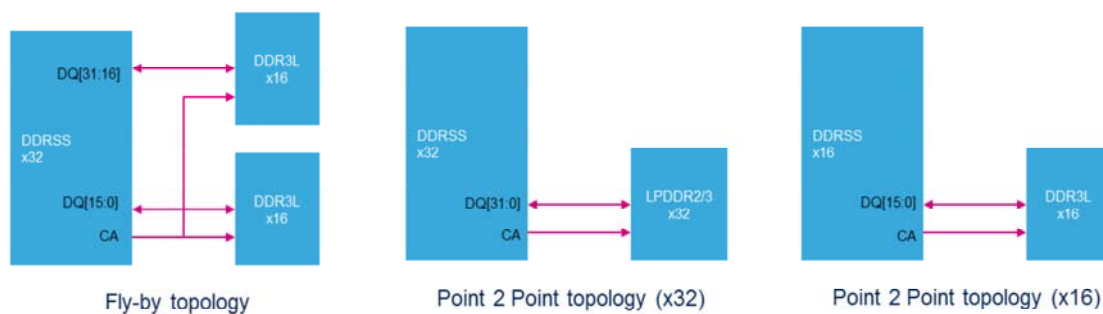
- Support multiple DDR standards: DDR3, DDR3L, LPDDR2, LPDDR3
- For all standards, the maximum frequency is 533MHz (DDR at 1066Mbps)
- Note LPDDR3 is supported without On Die Terminal (ODT), similar to LPDDR2 support.
- 32-bit Interface width can be either fully populated (32bit ) or half populated (16 bit)
- Note the device may also be packaged with the 16-bit DDR interface only
- Density limited to 1GByte due to the memory map address range
- DDRSS is supporting Single rank only
- DDR3 and DDR3L memories are similar except for the

supply voltage (1.5V for DDR3 / 1.35V for DDR3L). For the sake of simplification we refer only to DDR3L .

# SDRAM Topology and density selection

3

- SDRAM Topologies are:
  - Point to point (p2p) with single SDRAM device
  - Fly-by with two SDRAM devices
- The Topology is set according to SDRAM and PHY interface widths



DRAM topologies can be:

- Point to point (p2p) applicable with a single SDRAM device on the board , This is the case with 16 bit DDR3L or 32 bit LPDDR2/3 memories.
- Fly-by applicable with two DDR3L devices (16 bit) on the board and a 32 bit (full populated) interface.

For the Fly-by topology, the Command/Address bus (CA bus) is sequentially routed to the two DDR3L devices. Byte lanes groups (DQ/DQS) are connected point to point. It is also recommended to use termination on the Command /Address bus to VTT.

The suitable topology is determined by the SDRAM memory device and the DDR interface width, resulting in a number of SDRAM devices considering their density.

LPDDR2/3 devices are mostly available with 32-bit data bus width while DDR33/3L devices are mostly available with a 16-bit data bus width.

- DDR Subsystem (DDRSS) is composed of:
  - DDR controller (DDCTRL)
    - Converts AXI bus transactions to SDRAM commands at the DFI interface
    - Does AXI Port arbitration for from 64-bit AXI
    - Schedule DRAM commands according to traffic classes (QoS) for optimal DDR utilization
    - Schedule DRAM refreshes.
  - DDR PHY (DDRPHYC)
    - Command/address (CA bus) and Write data (DQ/DQS) to SDRAM according to JEDEC
    - Read data (DQ/DQS) from SDRAM according to JEDEC
    - Support initialization of PHY and SDRAM for DDR3 and LPDDR2/3 standards



The DDR Subsystem (DDRSS) is composed of 2 main parts:

- The DDR Controller (DDCTRL)
- The DDR PHY (DDRPHYC)

DFI is the standard interface between the DDR Controller and the DDR PHY

The DDCTRL is in charge of:

- Convert AXI bus transactions to DRAM transfers at DFI interface
- AXI Port arbitration, DDCTRL is equipped with a dual 64-bit AXI
- Schedule DRAM commands according to traffic classes

(QoS) for optimal DDR utilization

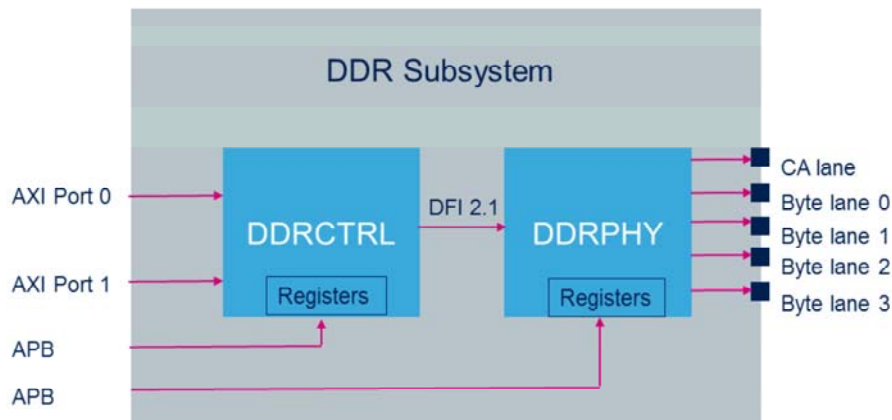
- Schedule DRAM refreshes

The DDR PHY (DDRPHYC) is in charge of:

- Drive Command/Address (CA bus) and write data (DQ/DQS) according to JEDEC timing
- FIFO read data (DQ/DQS) according to JEDEC timing
- Programmable initialization of PHY and DRAM for DDR3 and LPDDR2/3 standards

## DDRSS diagram

5



This is a simplified Block diagram of the DDR subsystem (DDRSS)

DDRCTRL is a multi standard DDR controller connected to the SoC backbone and which generates DDR commands at the DFI interface.

DFI specification defines a generic interface protocol between a memory controller and PHY interfaces.

DDRPHYC is a multi standard PHY interface which supports the actual signaling of SDRAM memory device.

DDRPHYC also supports the required initialization sequence and some fine timing control required by the SDRAM.

DDRCTRL and DDRPHYC have their own Control registers. DDRCTRL is connected to the SoC backbone by Two AXI ports.

DDRPHYC has a slice based architecture with a Command/Address bus and 4 Byte lanes.



## DDRCTRL features (1/3)

6

- Two 64-bit AXI4 ports interface (XPI) asynchronous to the controller
- DFI 2.1 compliant interface to DDRPHYC
- Single rank configuration
- 1:1 frequency ratio between the DDRC clock and the DDR PHY
- Advanced scheduler for SDRAM command generator
- QoS support with 3 traffic classes on Read and 2 traffic classes on Write
- Options to avoid starvation of lower priority traffic
- Programmable burst length options (4, 8, 16)
- Write combine to allow multiple write accesses to the same address to be combined into a single write access
- Supports the automatic SDRAM power-down entry and exit caused by a lack of transaction arrival for a programmable time
- Supports the automatic Clock stop (LPDDR2/3) entry and exit caused by the lack of transaction arrival
- Supports the automatic low-power mode operation caused by a lack of transaction arrival for a programmable time via hardware low-power interface



The main DDRCTRL features are:

- Two 64-bit AMBA 4 AXI4 ports interface (XPI) asynchronous to the controller
- 64-bit with DFI 2.1 compliant interface to DDRPHYC
- Advanced scheduler and SDRAM command generator
- 1:1 frequency ratio between the DDRC clock and the DDR PHY (aka SDR mode)
- Advanced QoS support with 3 traffic classes on read and 2 traffic classes on write
- Options to avoid starvation of lower priority traffic
- Guaranteed coherency for write-after-read (WAR) and read-after-write (RAW) on the AXI ports
- Programmable support for burst length options (4, 8, 16)
- Write combine to allow multiple write accesses to the same address to be combined into a single write access
- Single rank configuration
- Supports the automatic SDRAM power-down entry and exit caused by a lack of transaction arrival for a

programmable time

- Supports the automatic clock stop (LPDDR2/3) entry and exit caused by the lack of transaction arrival
- Supports the automatic low-power mode operation caused by a lack of transaction arrival for a programmable time via hardware low-power interface.

## DDRCTRL features (2/3)

7

- Programmable paging policy selectable among any of the following options:
  - Leave page open after accesses (Open page policy)
  - Close page when there are no further access available in the controller for that page
  - Auto-precharge with each access, with an optimization for page-close mode which leaves the page open after a flush for read-write and write-read collision cases
- Supports the self-refresh entry and exit as follows:
  - Automatic self-refresh entry and exit triggered by a lack of transaction for a programmable time
  - Self-refresh entry and exit under software control
- Support for deep power-down entry and exit under software control (LPDDR2)
- Support for explicit SDRAM mode register updates under software control
- Flexible address mapper logic to allow application specific mapping of row, column, bank bits



Programmable paging policy can be as any of the following options:

- Leave page open after accesses (Open page policy)
- Close page when there are no further access available in the controller for that page
- Auto-precharge with each access, with an optimization for page-close mode which leaves the page open after a flush for read-write and write-read collision cases

The DDR controller supports the self-refresh entry and exit as follows:

- Automatic self-refresh entry and exit triggered by a lack of transaction for a programmable time
- Self-refresh entry and exit under software control

Support for deep power-down entry and exit under software control (LPDDR2)

Support for explicit SDRAM mode register updates under software control

Flexible address mapper logic to enable application specific

mapping of row, column and bank bits.

- Programmable support for 1T or 2T timing
- User-selectable refresh control options:
  - Controller-generated auto-refreshes at programmable average intervals
  - Ability to group up to 8 controller-generated refreshes together to be consecutively issued (this reduces the number of page closings, increasing the overall efficiency)
  - When the controller-generated refreshes are grouped, some refreshes can be speculatively issued when the controller is idle for a programmable period of time
  - Ability to disable controller-generated auto-refreshes
  - Ability to issue a refresh through direct software request
  - User-selectable ability to perform per-bank refreshes rather than all-banks refreshes for LPDDR2/3 devices



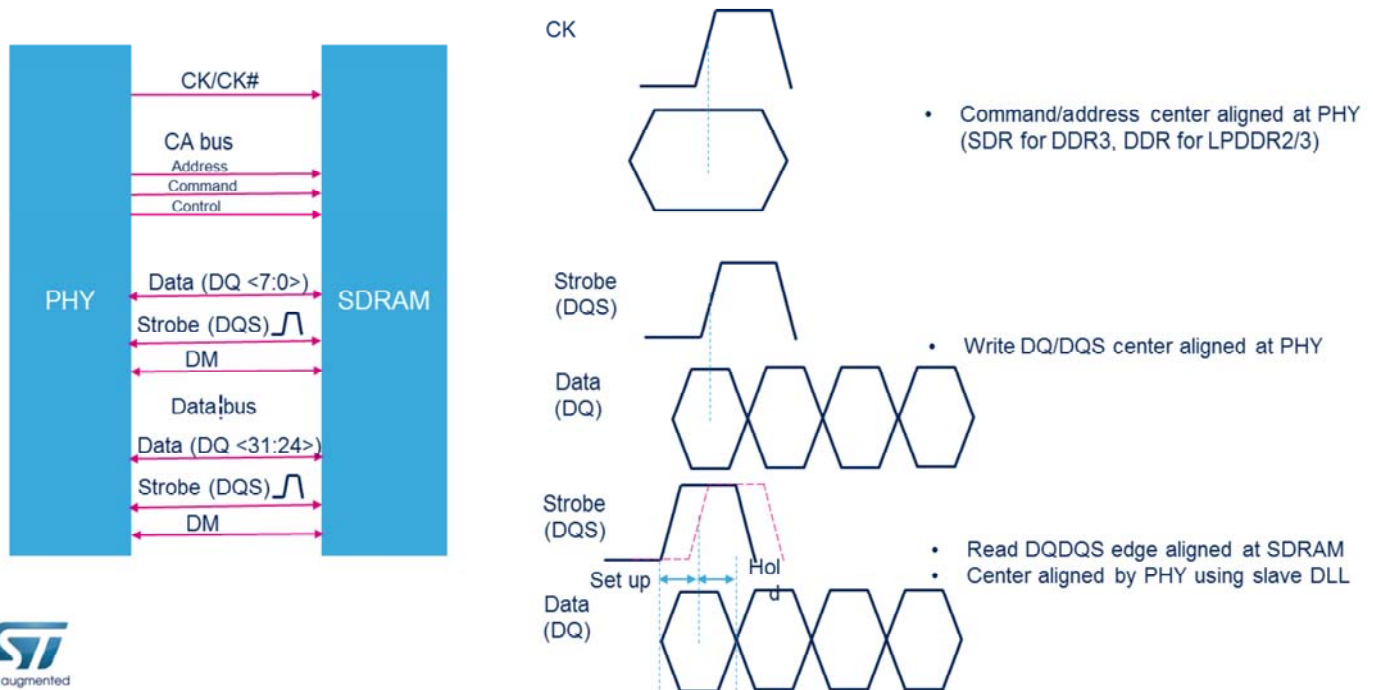
The DDR controller supports programmable 1T or 2T timing. Refresh control mode can be selected among the following options:

- Controller-generated auto-refreshes at programmable average intervals
- Ability to group up to 8 controller-generated refreshes together to be consecutively issued (this reduces the number of page closings, increasing the overall efficiency)
- When the controller-generated refreshes are grouped, some refreshes can be speculatively issued when the controller is idle for a programmable period of time
- Ability to disable controller-generated auto-refreshes
- Ability to issue a refresh through direct software request
- User-selectable ability to perform per-bank refreshes rather than all-banks refreshes for LPDDR2/3 devices.



# DDR Source Synchronous timings

9



This is a simplified view of a DDR interface using source synchronous signaling. The interface has several signal groups.

- The CA bus group which consists in Command/Address signals, they are unidirectional and related to the differential clock signals CK/CK#. CA signals are center aligned at PHY output by the delay-locked-loop circuit (MDLL). The CA bus uses Single edge (SDR) for DDR3 and Dual edge (DDR) for LPDDR2/3.
- The DQ/DQS group per byte lane: these signals are bidirectional according to Write and Read command. The data group includes data (DQ) and data mask (DM) signals, they are related to the DQS/DQS# strobe.
  - For Write access, DQ/DQS are center aligned at PHY output by the MDLL's
  - For Read access, DQ/DQS are edge aligned at SDRAM

edge and the PHY will gate in/out signals and realign DQS/DQS# for optimal sampling using the SDLL.

Furthermore fine step delay can be applied to individual signals in the group for optimal timing margins. This is covered by the tuning procedure.

- DDRPHYC uses a Byte lane architecture for best signal delay matching.
- DDRPHYC uses a DLL based architecture with an MDLL for CA bus and one MSDLL per byte lane.
- The DLL's are used to:
  - Delay CK/CK# signals by 180° (SDR) and 90° (DDR) to align CA bus to DDR timings
  - Delay output DQS/DQS# signals by 90° to align byte lane during a write access
  - Delay input DQS/DQS# signals by 90° to capture byte lane during a read access



DDRPHYC uses a Byte lane architecture so that all signals in a group are closely matched.

It uses a Master DLL (MDLL) for Command/Address signals to the SDRAM memory and a Master/Slave DLL (MSDLL) for each Byte lane group.

MDLL helps tune the CK/CK# delay by 180 degrees (SDR mode) in the case of DDR3L devices and by 90 degrees (DDR mode) in the case of LPDDR2/3 devices.

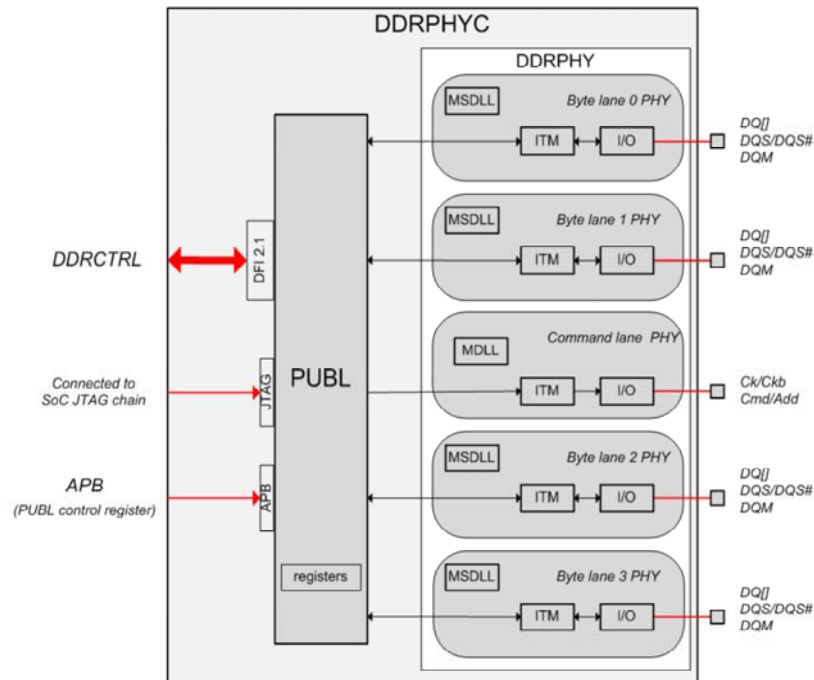
MSDLL is used to delay DQS/DQS# output signals by 90 degrees to center align Write data.

MSDLL is used to delay DQS/DQS# input signals by 90 degrees to center align Read data.



## DDRPHYC diagram

11



This is a simplified diagram of DDRPHY controller. The Physical Utility Block (PUBL) includes the control registers and several state machines. PUBL is in charge of the sequence of the PHY activity, including the DDR initialization sequence. PUBL supports impedance calibration and several 'built-in' tuning with programmable execution. The MDLL and MSDLL are generating clock phases to the Internal Timing Modules (ITM's).

- PHY Initialization with DLL's Reset and locking
- SRAM initialization including MRS and ZQCL commands according to SDRAM standards
- Output Drive and/or ODT Impedance calibration (ZCAL) using external 240 ohm
- Support DQS Gate Training (DQSTRN) to position optimal gating of Read data.
- Support Read DQ/DQS Eye Centering delay tuning (software)
- Support Read DQ delay tuning (software)
- PUBL supports a BIST engine for at-speed loopback testing.



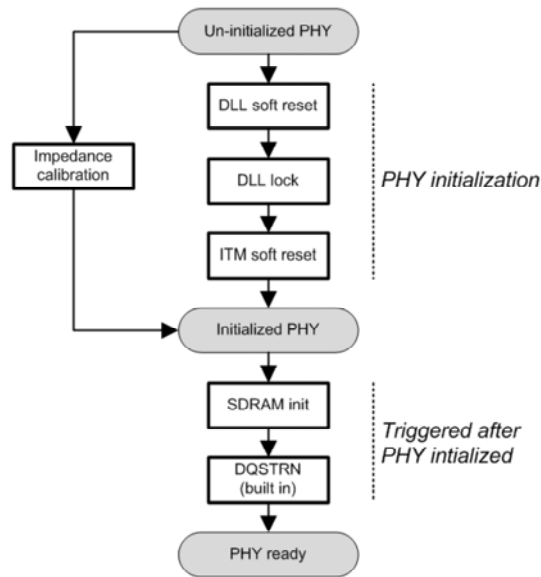
This is a summary of the DDRPHYC features:

- PHY Initialization with DLL's Reset and locking
- SRAM initialization including MRS and ZQCL commands according to SDRAM standards
- Output Drive and/or ODT Impedance calibration (ZCAL)
- DQS Gate Training (DQSTRN) to position optimal gating of Read data
- Support Read DQ/DQS Eye Centering with fine step delay by software
- Support Read DQ delay tuning with fine step delay by software
- PUBL also supports a BIST engine for at-speed loopback testing



# PHY Initialization sequence

13



- Above steps are launched via DDRPHYC\_PIR register
- PHY tuning is done by software after PHY ready
- tuning parameters are saved and restored

The Complete PHY initialization including SDRAM initialization and DQSTRN is shown in this diagram. The sequence is defined and controlled by the PIR register. This sequence is launched after a system reset. Tuning parameters determined by software during the bring up, are saved and restored.

- DQS gate is the window during which Read DQS propagates to PHY read logic to sample incoming read data.
- DQS gate position is depending of the round trip delay.
- DQS gate needs to be positioned to Quarter bit accuracy.
- PUBL is supporting a built-in DQS Gate Training sequence (DQSTRN), it is launched by default during initialization.
- DQSTRN is using the SDRAM array, instead of DDR3 MPR register, by default Row/Bank/Column zero's are selected.



DQS gate is the window during which Read DQS propagates to PHY read logic to sample incoming read data.

DQS gate position is depending of the round trip delay.

DQS gate needs to be positioned to Quarter bit accuracy.

PUBL is supporting a built-in DQS Gate Training sequence (DQSTRN), it is launched by default during initialization.

DQSTRN is using the SDRAM array, instead of the DDR3 MPR register, by default Row/Bank/Column zero's are selected.

- DQ and DQS can be delayed in fine steps ( $\sim\pm 20\text{pS}$ ).
  - DQ bit delays are used to compensate for DQ bit mismatch (DQ bit deskew)
  - DQS/DQS# delay are used for optimal placement for DQ sampling (DQS Read Eye centering)
- DQ/DQS Read Tuning is a software procedure supported by STM32CubeMx tool to find optimal settings.
- Both DQ bit deskew and DQS Read Eye centering should be done during the board bring up and the determined optimal settings saved to the DDR configuration registers.



DQ and DQS can be delayed in fine steps ( $\sim\pm 20\text{pS}$ ) (pico second]

- DQ bit delays are used to compensate for DQ bit mismatch (DQ bit deskew) .
- DQS/DQS# delays help optimize the placement for DQ sampling (DQS Read Eye centering)

DQ/DQS Read Tuning is a software procedure supported by the STM32CubeMx tool to find optimal settings.

Both DQ bit deskew and DQS Read Eye centering should be done during the board bring up and the determined optimal settings saved to the DDR configuration registers.

- Power saving scheme implemented at various locations of the DDR subsystem:
  - SDRAM: Precharge Power down, **self-refresh** , clock disabling.
  - DDRPHYC: DLL and I/O's low power modes
  - DDRCTRL and SoC: clock gating
- Self-refresh is the main power saving feature supported by DDRCTRL. It can be entered in 3 ways:
  - Explicit control from Software (SSR)
  - Based on idle timeout (ASR)
  - Control from hardware (HSR)



Power saving scheme is implemented at various locations of the DDR subsystem:

- SDRAM: with Precharge Power down, self-refresh and clock disabling.
- DDRPHYC: with DLL and I/O's low power modes
- DDRCTRL and SoC: with clock gating

Self-refresh is the main power saving feature supported by DDRCTRL. It can be entered in 3 ways:

- Explicit control from Software (SSR)
- Based on idle timeout (ASR)
- Control from hardware (HSR)



## DDRSS configuration 17

- The DDRCTRL and DDRPHYC registers are programmed according to SDRAM timings and configuration prior to the DDRSS initialization.
- The DDRSS configuration is requiring minimal information from the user through STM32CubeMx DDR panel.
- The DDRSS is ready after the DDRSS initialization. Enabling the AXI port is the final step.
- Beside the predetermined configuration parameters, the tuning parameters are determined during the board bring up, using the STM32CubeMx tool.
- Once the DDR memory is configured and the PHY is tuned for optimized timing, the configuration robustness must be tested. A complete test suite is proposed by the STM32CubeMx tool.
- Configuration parameters are saved and restored before regular DDR operations.



The DDRCTRL and DDRPHYC registers are programmed according to SDRAM timings and configuration prior to DDRSS initialization.

The DDRSS configuration is requiring minimal information from the user through the STM32CubeMx DDR panel.

The DDRSS is ready after the DDRSS initialization. Enabling the AXI port is the final step.

Beside the predetermined configuration parameters, the tuning parameters are determined during the board bring up, using the STM32CubeMx tool.

Once the DDR memory is configured and the PHY is tuned for optimized timing, the configuration robustness must be tested. A complete test suite is proposed by the STM32CubeMx tool.



Configuration parameters are saved and restored before regular DDR operations.

- DDR3 SDRAM Standard: JEDEC JESD79-3F
- LPDDR2 SDRAM Standard: JEDEC JESD209-2F
- LPDDR3 SDRAM Standard: JEDEC JESD209-3C
- RM0436 Reference manual STM32MP15xxx advanced Arm®-based 32-bit MPUs
- AN5168: DDR Subsystem initialization and configuration.
- AN5122: DDR PCB Design guidelines
- STM32CubeMX tool



For more details you may refer to the following documents:

- DDR3 SDRAM Standard: JEDEC JESD79-3F
- LPDDR2 SDRAM Standard: JEDEC JESD209-2F
- LPDDR3 SDRAM Standard: JEDEC JESD209-3C
- RM0436 Reference manual STM32MP15xxx advanced Arm®-based 32-bit MPUs
- AN5168: DDR Subsystem initialization and configuration.
- AN5122: DDR PCB Design guidelines
- STM32CubeMX tool