DDR configuration and initilaization for Synopsis UMCTL-based platforms

Technical Note

TNxxxx
Rev. 11 (master@a510aa6)
Fri Jan 7 15:25:10 2022 +0100

Confidential

TERMS and ABBREVIATIONS

```
PUB                PHY utility block
SoC                System on a Chip
```

Introduction

Purpose of this document is to specify and document the design and requirements of the DDR customization for Synopsis UMCTL-based SoC's. This currently implies Fireant, Maserati and Laguna.

Configuring DDR for these platforms is a delicate process including a lot of configuration parameters. It is furthermore complicated by the needs for customer customization, sourcing changes and system tuning.

As we wish to provide a better method for enabling customers, many different boards and current/future SoC's, this document outlines a possible solution for this.

Primary target audience is:

- **SW Engineers:** This is the group of people who will do the software implementation of the proposed solution. Implementation and specification must be in alignment.
- **SW-Application:** Parts of this implementation will be directly exposed to customers, mainly the configuration tool and associated documentation. This group must be consulted to ensure catering for customer needs and being able to support the system solution as a whole.
- **Chip design:** This group will be consulted to ensure all relevant parameters are exposed to customization, and that the driver follow the necessary hardware initialization steps.

Requirements

The system must support the following high-level requirements:

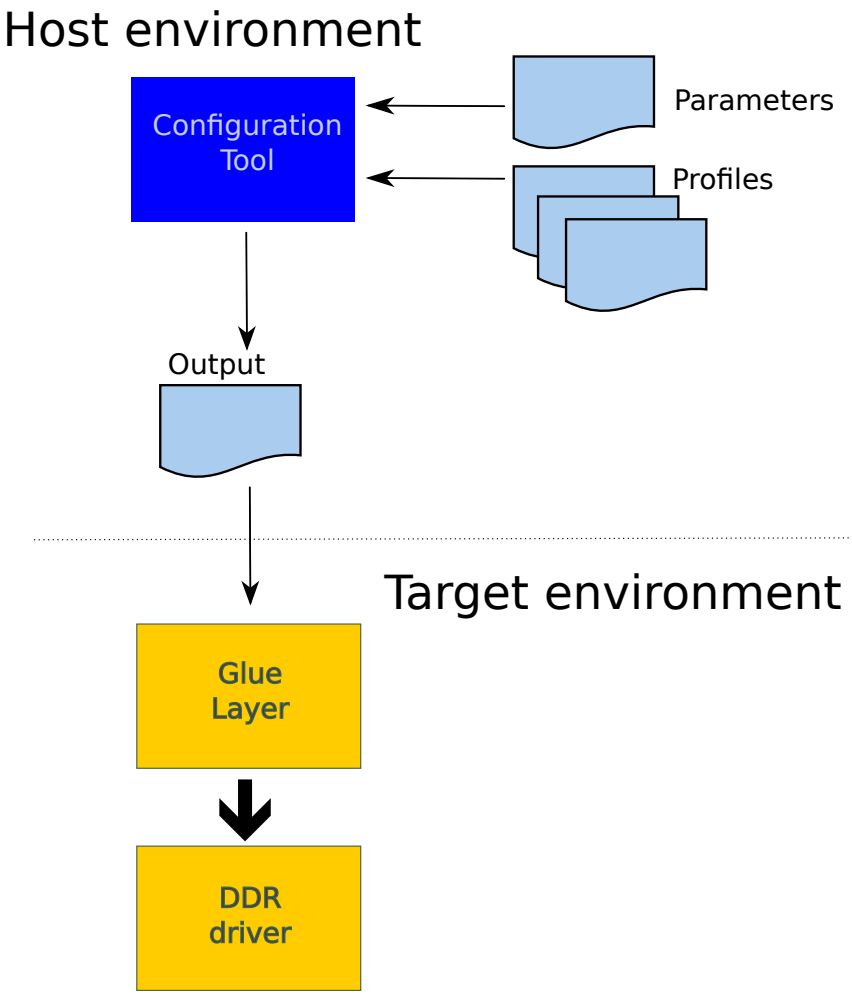
- The system must be able to support the current set of SoC's and boards using the UMCTL controller. This imply all Fireant boards, Maserati boards and Laguna boards.
- The system must be able to support new SoC's and their boards without imposing changes to exposed parameters.
- A set of high-level DDR configuration parameters must be exposed for customization of the 'common-case' changes a customer is expected to make.
- It should be possible to extend the configuration tool with profiles/parameter-sets to support new DDR chip types as required by sourcing needs.
- It should be possible to customize DDR parameters for a board without necessarily recompiling the firmware for the target system.
- The run-time parts of the system initialization should be as compact as possible, and require no floating-point support.

System design

To implement a system which feature the listed requirements, the following solution is proposed:

- A host configuration tool is created. This tool will take the actual values of the exposed parameters, the applicable DDR parameter sets and will produce a 'ready-to-use' DDR controller output configuration 'snippet'. All necessary calculations will be done in the configuration tool.
- The host configuration tool will be able to produce the output configuration snippets in different formats to support different platforms - binary, "C", YAML and device-tree (DT) format.
- The host configuration tool will operate on input data specified in YAML format, and DDR profiles/parameter-sets will be in the same format.
- A system-agnostic driver will be created. The driver will accept the DDR configuration 'snippet' in **binary** form, along with platform-defined DDR register access functions. Using this, the driver will be able to perform the full DDR initialization according to the UMCTL programmers guide.
- A system-specific 'glue' component will be created to extract the DDR configuration 'snippet' and convert it to binary form. It is the responsibility of this component to call the base driver, alongside with the register access functions. This component could optionally provide built-in fallback configuration profiles if initialization with the dynamically provided data fails.

The system architecture is outlined in the following diagram:



Configuration parameters

User-level configuration parameters

At the top level, the following user-level parameters are identified.

- Title: *text* (This text is accompanying the configuration for identification purposes)
- DDR type: `DDR3` , `DDR4`
- DDR speed: *integer* (KHz)
- DDR geometry:

  - Column bits
  - Row bits
  - Bank bits
  - Bank group bits
  - Active ranks 1 or 2
  - DQ bits: *x8/x16*
  - DQ bits used: *x16/x32/x40*
  - Density: *4G/8G*
- 2T mode (???): *enabled/disabled*
- ECC mode: *enabled/disabled*

NOTE Some platforms may not support all parameters.

DDR chip parameters

All DDR chip parameters are derived by the user-level parameters, primarily the DDR mode and speed (grade).

Output DDR configuration

The DDR output configuration data is as follows.

NOTE The following data/registers set is taken from drivers/st/ddr/stm32mp1_ddr.c (https://bitbucket.microchip.com/projects/UNGE/repos/sw-arm-trusted-firmware/browse/drivers/st/ddr/stm32mp1_ddr.c?at=refs%2Fheads%2Flaguna-v1) and the associated header file include/drivers/st/stm32mp1_ddr.h (https://bitbucket.microchip.com/projects/UNGE/repos/sw-arm-trusted-firmware/browse/include/drivers/st/stm32mp1_ddr.h?at=refs%2Fheads%2Flaguna-v1). The driver is going to be used as a reference to implement the Microchip equivalent, and it is expected to change the register set as required/desired.

Info

- Title: *text* (from input)
- Speed
- Memory size

Main control registers

registers	ddr tcl	fa ddr3	fa ddr4	stm32mp1	comments
mstr	yes	yes	yes	yes	

registers	ddr tcl	fa ddr3	fa ddr4	stm32mp1	comments
pwrctl	yes	yes	yes	yes	
rfshctl0	yes	yes	yes	yes	
rfshctl3	yes	yes	yes	yes	
dfitmg0	yes	yes	yes	yes	
dfitmg1	yes	yes	yes	yes	
dfiupd0	yes	yes	yes	yes	
dfiupd1	yes	yes	yes	yes	
pccfg	yes	-	-	yes	Only used for x16 width configurations

Timing configuration registers

registers	ddr tcl	fa ddr3	fa ddr4	stm32mp1	comments
rfshtmg	yes	yes	yes	yes	
dramtmg0	yes	yes	yes	yes	
dramtmg1	yes	yes	yes	yes	
dramtmg2	yes	yes	yes	yes	
dramtmg3	yes	yes	yes	yes	
dramtmg4	yes	yes	yes	yes	
dramtmg5	yes	yes	yes	yes	
dramtmg8	yes	yes	yes	yes	
dramtmg9	yes	-	yes	-	Should this be configured even for DDR3?
odtcfg	yes	yes	yes	yes	

Address map configuration registers

registers	ddr tcl	fa ddr3	fa ddr4	stm32mp1	comments
addrmap0	yes	yes	yes	-	
addrmap1	yes	yes	yes	yes	
addrmap2	yes	yes	yes	yes	
addrmap3	yes	yes	yes	yes	
addrmap4	yes	yes	yes	yes	
addrmap5	yes	yes	yes	yes	
addrmap6	yes	yes	yes	yes	
addrmap7	yes	yes	yes	-	
addrmap8	yes	yes	yes	-	

DDR PHY registers

registers	ddr tcl	fa ddr3	fa ddr4	stm32mp1	comments
dxccr	yes	yes	yes	yes	
dsgcr	yes	yes	yes	yes	
dcr	yes	yes	yes	yes	

DDR PHY timing registers

registers	ddr tcl	fa ddr3	fa ddr4	stm32mp1	comments
ptr0	yes	yes	yes	yes	
ptr1	yes	yes	yes	yes	
ptr2	-	-	-	yes	Should this be used?
dtpr0	yes	yes	yes	yes	
dtpr1	yes	yes	yes	yes	
dtpr2	yes	yes	yes	yes	
mr0	yes	-	-	yes	

registers	ddr tcl	fa ddr3	fa ddr4	stm32mp1	comments
mr1	yes	-	-	yes	
mr2	yes	-	-	yes	
mr3	yes	-	-	yes	

Sparx5 DDR registers not mapped to configuration registers

register	ddr tcl	fa ddr3	fa ddr4	comments
bistar0	-	-	yes	Newly added (by JSA) for VREF training
bistar1	-	-	yes	-same-
bistar3	-	-	yes	-same-
bistudpr	-	-	yes	Newly added for VREF training
crcparctl1	yes	-	yes	**Add config**: write_crc, ca_parity_en
dbictl	yes	-	yes	**Add config**: dbi_en
dfimisc	yes	yes	yes	**Add config**: dbi_en
dramtmg12	yes	-	-	Only used with PDA_EN, do need this feature?
dtcr0	yes	yes	yes	**Add config**: depends on #lanes (dq_bits_used)
dtcr1	yes	yes	yes	**Add config**: depends params_active_ranks
dtpr3	yes	yes	yes	**Add config**: depends params_tDLLKc
dtpr4	yes	yes	yes	**Add config**: depends params_tXPc / params_tXPDLLc
dtpr5	yes	yes	yes	**Add config**: depends params_tWTRc / params_tRCDc / params_tRCc

register	ddr tcl	fa ddr3	fa ddr4	comments
dx0bdlr0	-	yes	yes	Used for data training (fixed value)
dx0bdlr1	-	yes	yes	Used for data training (fixed value)
dx0bdlr2	-	yes	yes	Used for data training (fixed value)
dx0gcr5	-	-	yes	Fixed value: Used for VDDQ
dx0gtr0	yes	yes	yes	Fixed value: DGSL = 2 used, should it ever changed?
dx1bdlr0	-	yes	yes	Used for data training (fixed value)
dx1bdlr1	-	yes	yes	Used for data training (fixed value)
dx1bdlr2	-	yes	yes	Used for data training (fixed value)
dx1gcr5	-	-	yes	Fixed value: Used for VDDQ
dx1gtr0	yes	yes	yes	Fixed value: DGSL = 2 used, should it ever changed?
dx2bdlr0	-	yes	yes	Used for data training (fixed value)
dx2bdlr1	-	yes	yes	Used for data training (fixed value)
dx2bdlr2	-	yes	yes	Used for data training (fixed value)
dx2gcr0	yes	-	-	Only used for x16 width configurations
dx2gcr5	-	-	yes	Fixed value: Used for VDDQ
dx2gtr0	yes	yes	yes	Fixed value: DGSL = 2 used, should it ever changed?

register	ddr tcl	fa ddr3	fa ddr4	comments
dx3bdlr0	-	yes	yes	Used for data training (fixed value)
dx3bdlr1	-	yes	yes	Used for data training (fixed value)
dx3bdlr2	-	yes	yes	Used for data training (fixed value)
dx3gcr0	yes	-	-	Only used for x16 width configurations
dx3gcr5	-	-	yes	Fixed value: Used for VDDQ
dx3gtr0	yes	yes	yes	Fixed value: DGSL = 2 used, should it ever changed?
dx4bdlr0	-	yes	yes	Used for data training (fixed value)
dx4bdlr1	-	yes	yes	Used for data training (fixed value)
dx4bdlr2	-	yes	yes	Used for data training (fixed value)
dx4gcr0	yes	-	-	Only used for x16 width configurations
dx4gcr5	-	-	yes	Fixed value: Used for VDDQ
dx4gtr0	yes	yes	yes	Fixed value: DGSL = 2 used, should it ever changed?
ecccfg0	yes	yes	yes	**Add config**: ecc_mode
eccpoisonaddr0	yes	-	-	*Ignore*: Only used for ECC injection
eccpoisonaddr1	yes	-	-	*Ignore*: Only used for ECC injection
init0	yes	yes	yes	**Add config**: pre_cke / post_cke
init1	yes	yes	yes	**Add config**: DRAM_RSTN_X1024

register	ddr tcl	fa ddr3	fa ddr4	comments
init3	yes	yes	yes	**Add config**: params_reg_ddrc_mr / ddrc_emr
init4	yes	yes	yes	**Add config**: params_reg_ddrc_emr3 / ddrc_emr2
init5	yes	yes	yes	**Add config**: params_tZQinitc
init6	yes	-	yes	**Add config**: params_reg_ddrc_mr5
init7	yes	-	yes	**Add config**: params_reg_ddrc_mr6
iovcr0	yes	yes	yes	Fixed settings
iovcr1	yes	yes	yes	Fixed settings
mr0_ddr4	yes	-	-	**Add config**: Used - params_reg_ddrc_mr
mr0_lpddr3	-	yes	yes	Alias for mr0_ddr4
mr1_ddr4	yes	-	-	**Add config**: Used - ddrc_emr
mr1_lpddr3	-	yes	yes	Alias for mr1_ddr4
mr2_ddr4	yes	-	-	**Add config**: Used - ddrc_emr2
mr2_lpddr3	-	yes	yes	Alias for mr2_ddr4
mr3_ddr4	yes	-	-	**Add config**: Used - params_reg_ddrc_emr3
mr3_lpddr3	-	yes	yes	Alias for mr3_ddr4
mr4_ddr4	yes	-	-	**Add config**: Used - params_reg_ddrc_mr4
mr4_lpddr3	-	-	yes	Alias for mr4_ddr4
mr5_ddr4	yes	-	-	**Add config**: Used - dbi_en / params_dm_en / params_reg_ddrc_mr5
mr5_lpddr3	-	-	yes	Alias for mr5_ddr4

register	ddr tcl	fa ddr3	fa ddr4	comments
mr6_ddr4	yes	-	-	**Add config**: Used - params_reg_ddrc_mr6
mr6_lpddr3	-	-	yes	Alias for mr6_ddr4
pctrl_0	-	yes	yes	Control only (ECC scrubbing)
pgcr0	yes	yes	yes	Control only
pgcr1	yes	yes	yes	Fixed settings
pgcr2	yes	yes	yes	**Add config**: Used - params_tRASc_max
pgcr3	yes	yes	yes	Fixed settings, but **used** to hold params params_rd_dbi_en / params_wr_dbi_en (TCL script)
pgcr7	yes	yes	yes	Fixed settings
pir	yes	yes	yes	Fixed settings / depending on DDR3/ DDR4
ptr3	yes	yes	yes	**Add config**: Used - params_tdinit0 / params_tXS_tRFCc
ptr4	yes	yes	yes	**Add config**: Used - params_tdinit2 / params_tZQinitc
rankctl	yes	yes	yes	Fixed settings
rankidr	yes	yes	yes	Index register for `DX*GTR0` etc
rfshctl1	yes	yes	yes	Fixed settings
sbrctl	-	yes	yes	Used for ECC scrubbing init
sbrwdata0	-	yes	yes	-same-
schcr1	yes	-	-	**Add config**: Used - params_active_ranks > 1
swctl	yes	yes	yes	Control only

register	ddr tcl	fa ddr3	fa ddr4	comments
vtcr0	-	-	yes	Fixed settings
vtcr1	-	-	yes	Fixed settings
zq0pr	-	yes	yes	**Add config**: Used - ddrconf→ca_ln_drv : ddrconf→zqdiv
zq1pr	-	yes	yes	**Add config**: Used - ddrconf→zqdiv
zq2pr	-	yes	yes	**Add config**: Used - ddrconf→zqdiv
zqcr	yes	yes	yes	**Add config**: Used - asym_drv_pd/pu and params_tCK_min