Lab on
STM32CubeMX DDR tool suite

For an easier DDR configuration
an easier DDR signals tuning

STM32 MP1

life.augmented

Agenda

Presentation

25 min

- DDR tool suite forewords
- Technical information
- Demo
  - DDR configuration
  - Connection to the target
  - DDR registers loading
  - DDR tuning
  - DDR testing

DEMO (presenter Only)

Forewords

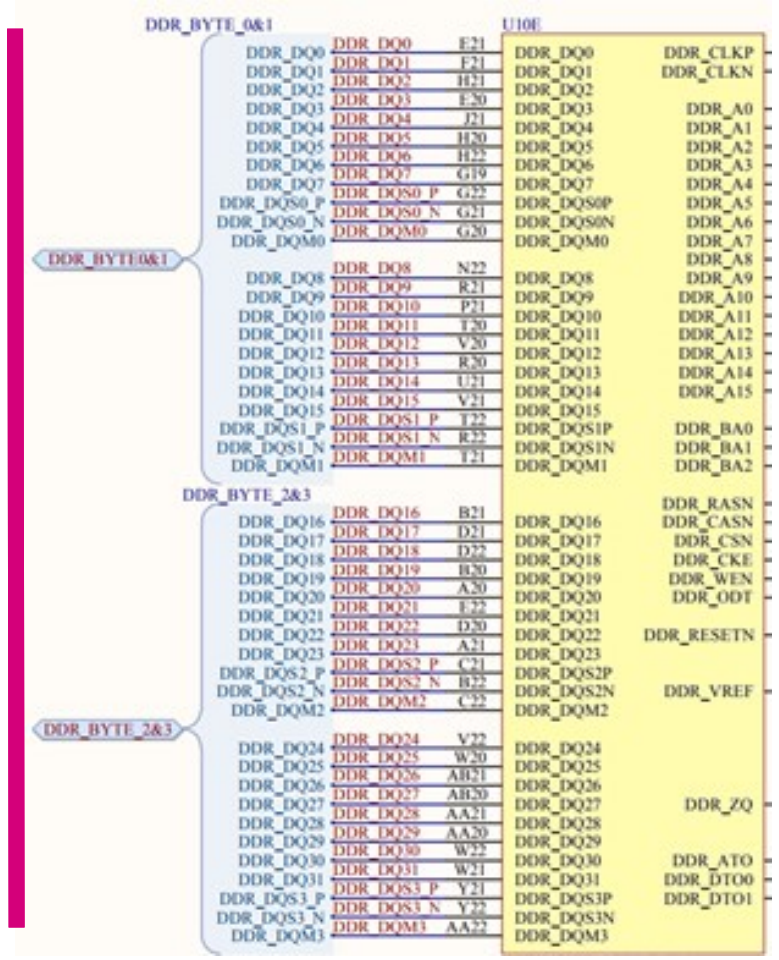
- STM32CubeMX comes along with an exhaustive **DDR tool suite** for DDR subsystem. Helps to setup the DDR controller.
- **Configuration of DDR** controller and **PHY** registers is managed automatically based on reduced set of editable parameters
- **DDR Testing** is offered based on a rich tests list (basic, stress tests). User can also develop its own tests.
- **Tuning** of byte lanes delays is proposed to compensate design imperfection

Technical inputs

Time 10 min

Technical inputs

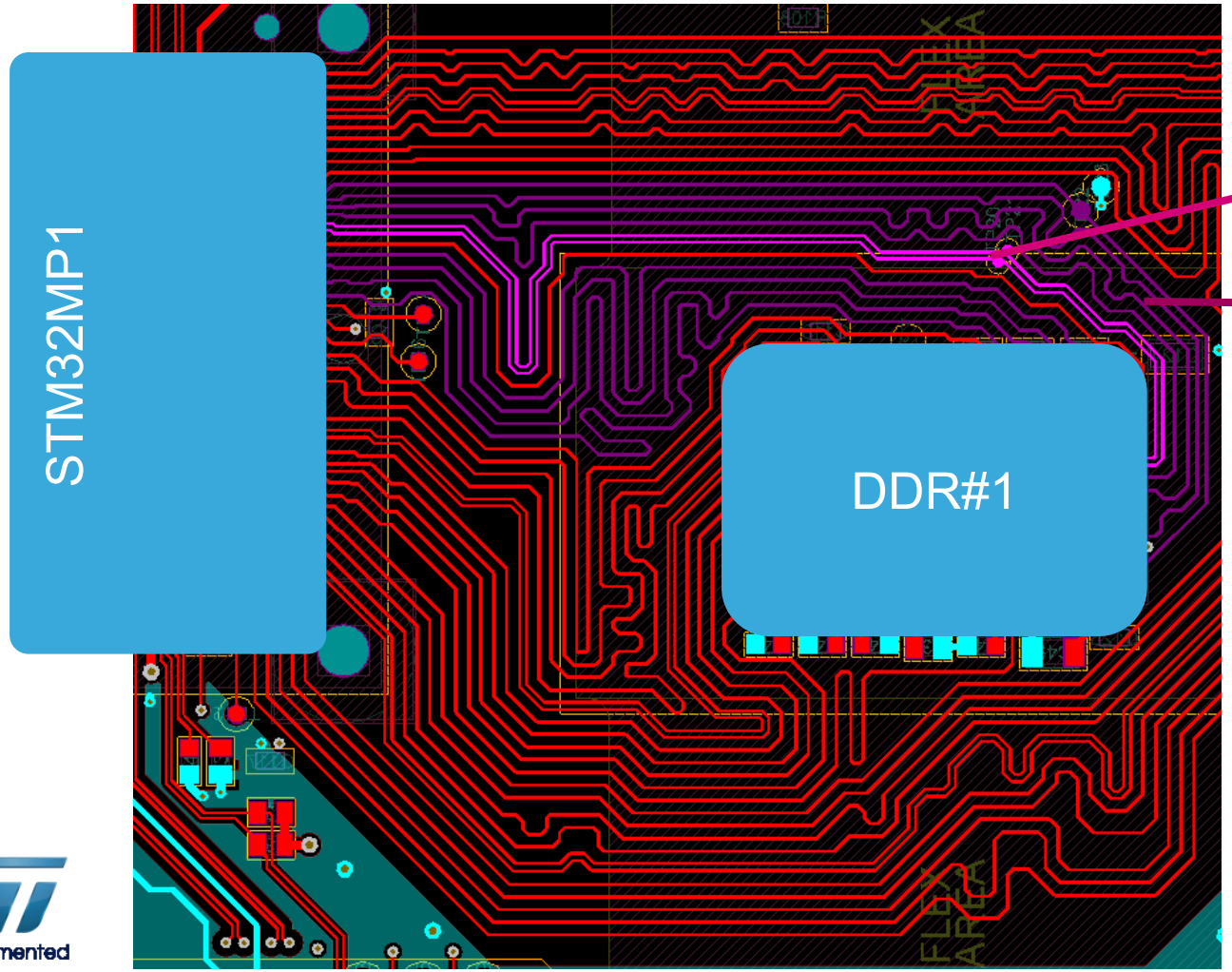
- Some signals routed between STM32MP15 and the DRAM are sensitive to design imperfection. See AN5122 for DDR design routing guidelines
- STM32MP15 PHY offers possibility to fine tune the those signals and thus compensate design imperfection
- **Tunable signals** are
  - DQS signals
  - DQ bits
- Tuning is a **sequence in 3 steps**
  - DQS gating
  - Bit deskew
  - Eye training



Tunable signals for the 4 data bytes (except the DQM one)

Technical inputs

- Example of DDR Byte0 tunable signals



DQS0N / DQS0P signals

DQ[0:7] byte lane

Constrained signals length
(e.g. +/-1mm within all signals
in a byte vs DQS)

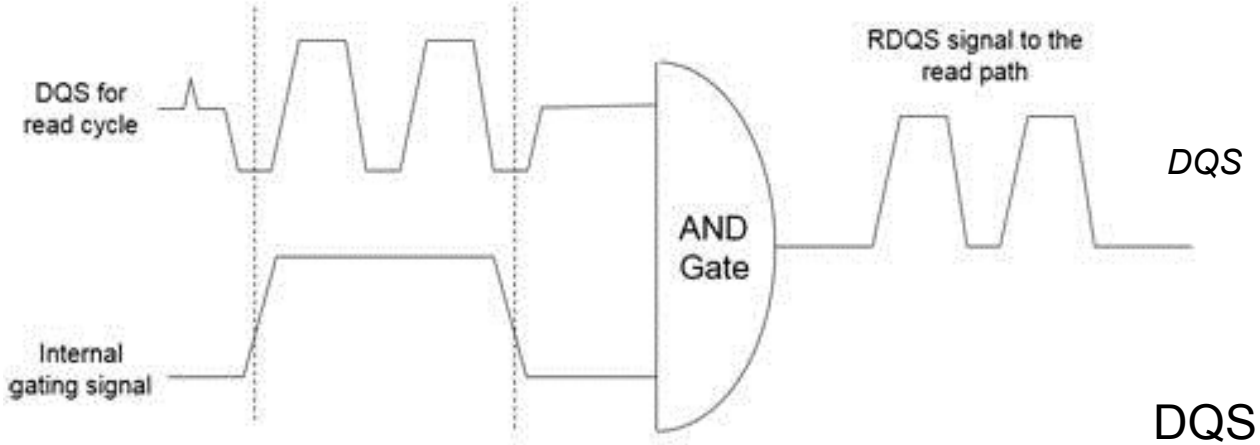
Length influences
the round trip delay

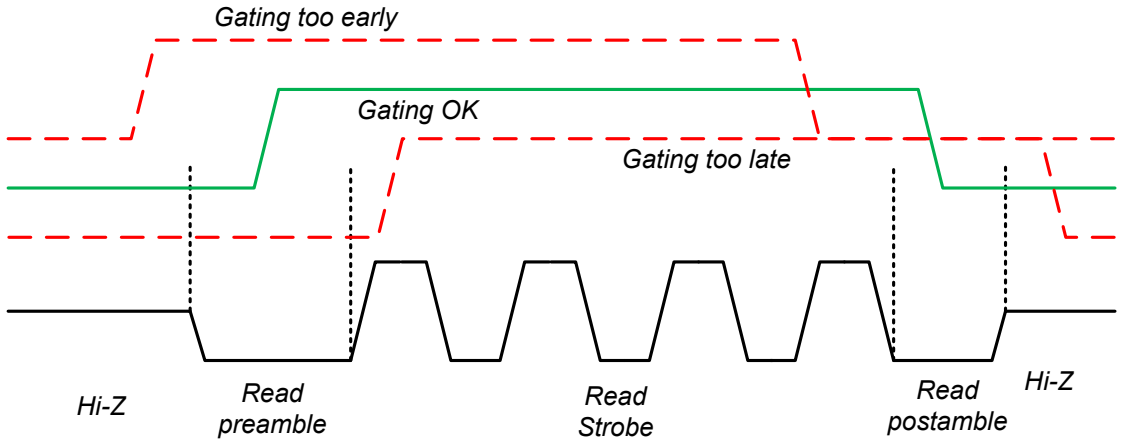
Technical inputs

- DDR tuning: **Read DQS gating training**

Read DQS gating

➢ *The purpose read DQS gating training is to adjust the timing of internal gating signal for DQS signal so that pseudo edges may occur due to glitches, post/preamble periods are eliminated.*



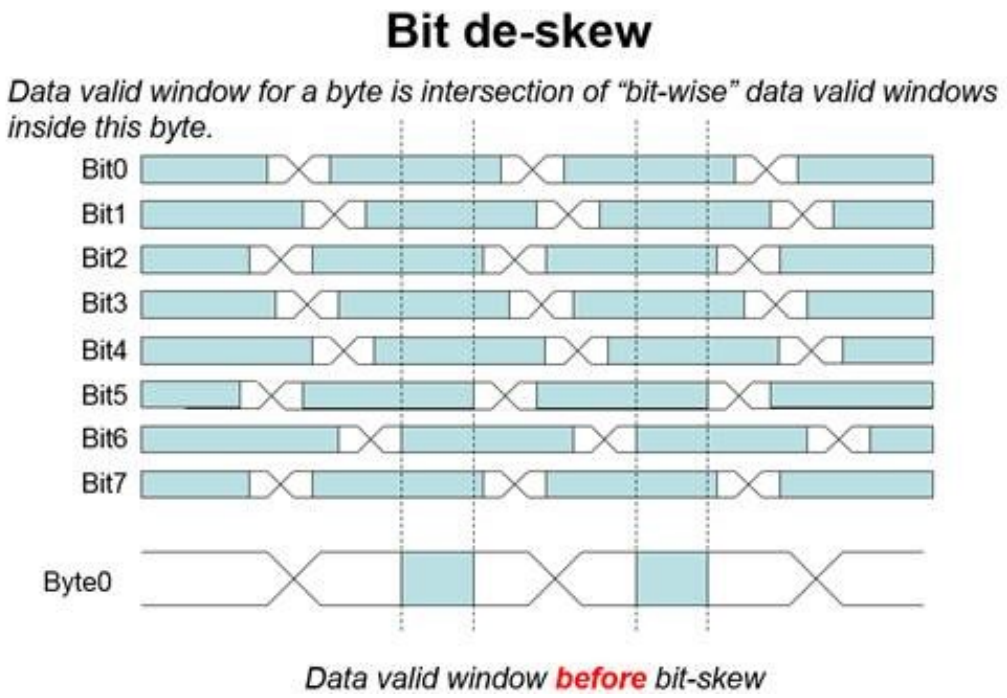


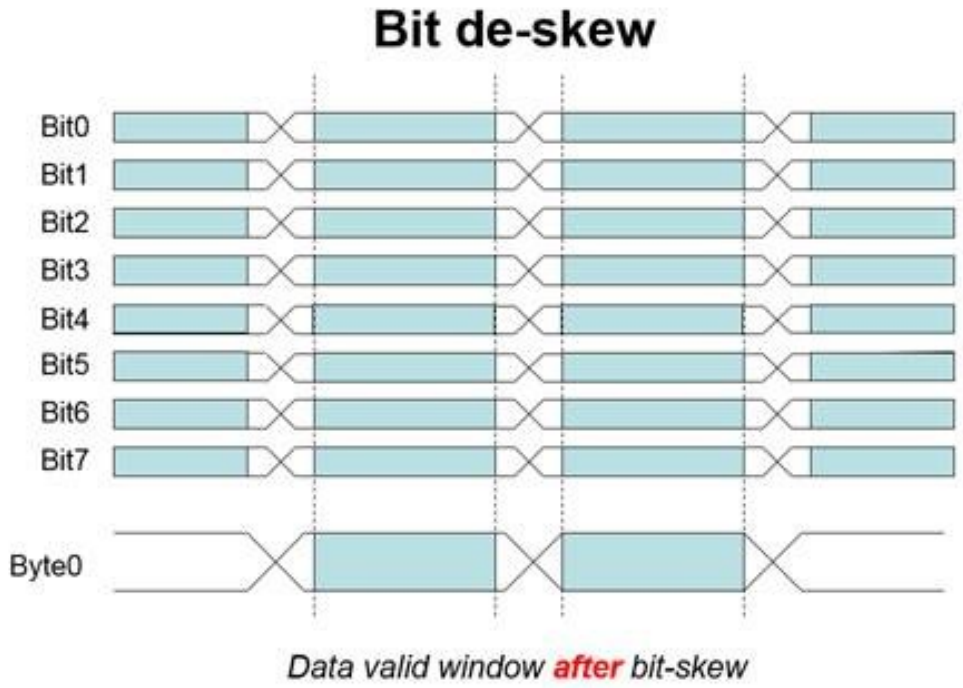
DQS gating position is depending on the round trip delay.

Read DQS signal is an input signal

Technical inputs

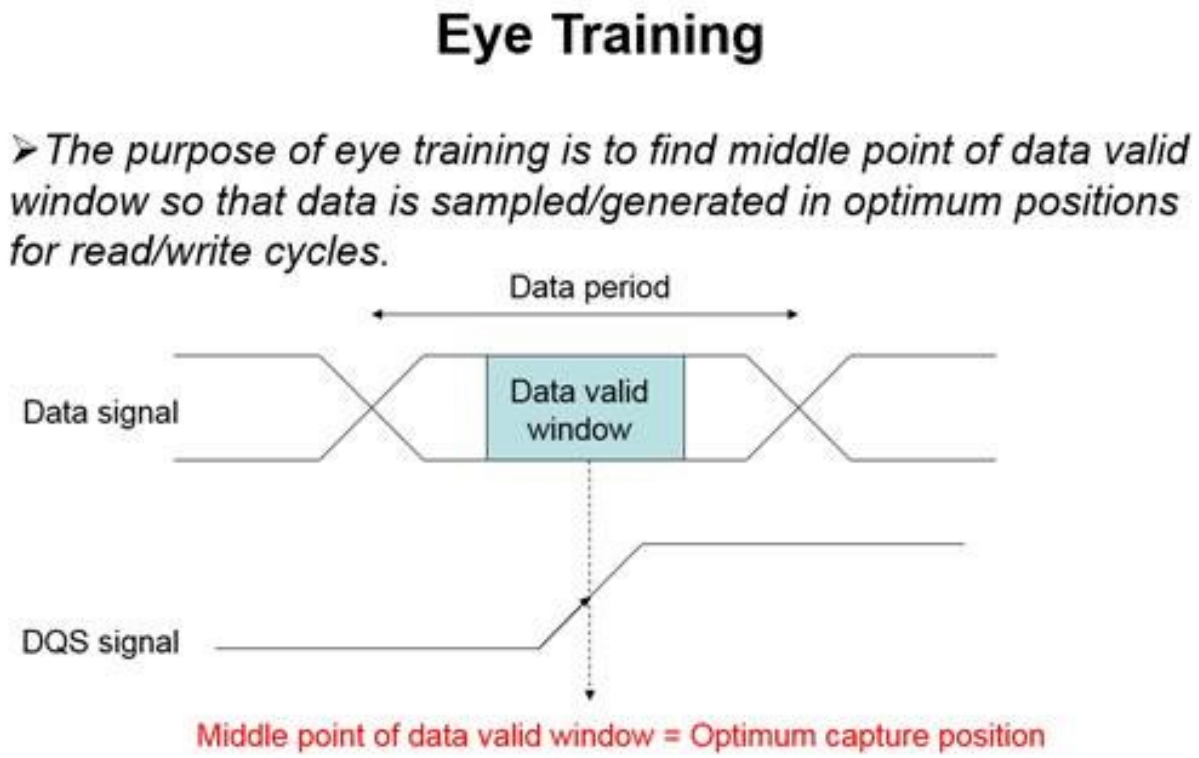
- DDR tuning: **Data Bit deskew**





  - Delay to be applied to each data line is computed based on results of several iteration DDR tool suite. This explains the step duration.

- DDR tuning: **Read Eye Training**



Read Eye training is intended for *Read access*

DEMO (presenter Only)

DDR Configuration Demo

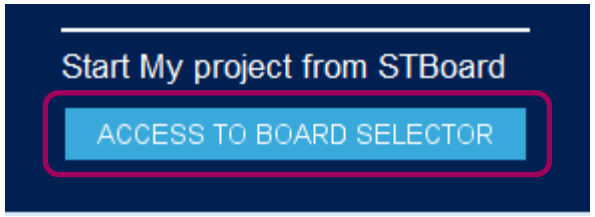
15 min

Lab Pre-requisites

- STM32CubeMX
- SD card Uboot-SPL from basic boot chain
- Discovery board

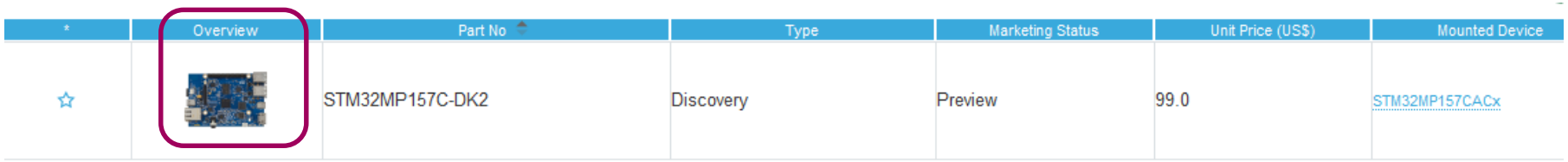
DDR configuration

1. First, create a STM32CubeMX project
2. Start my project from STBoard

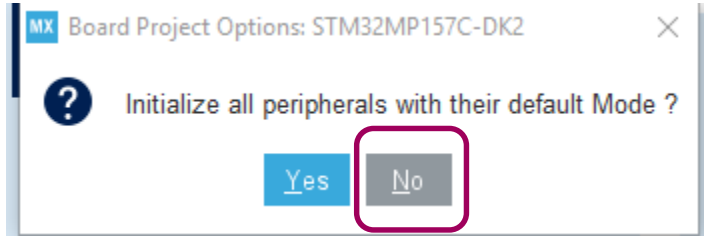


3. Select

4. Click on overview



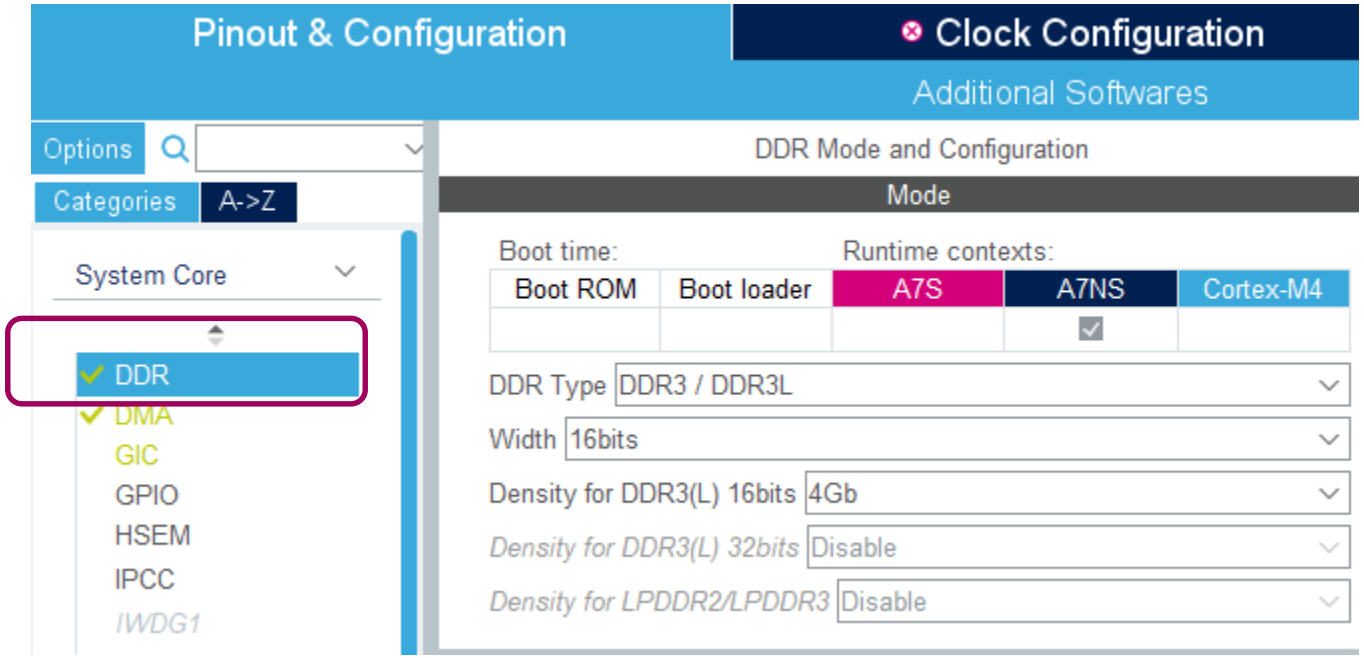
5. Create project



6. Initialize all peripherals : no

DDR configuration

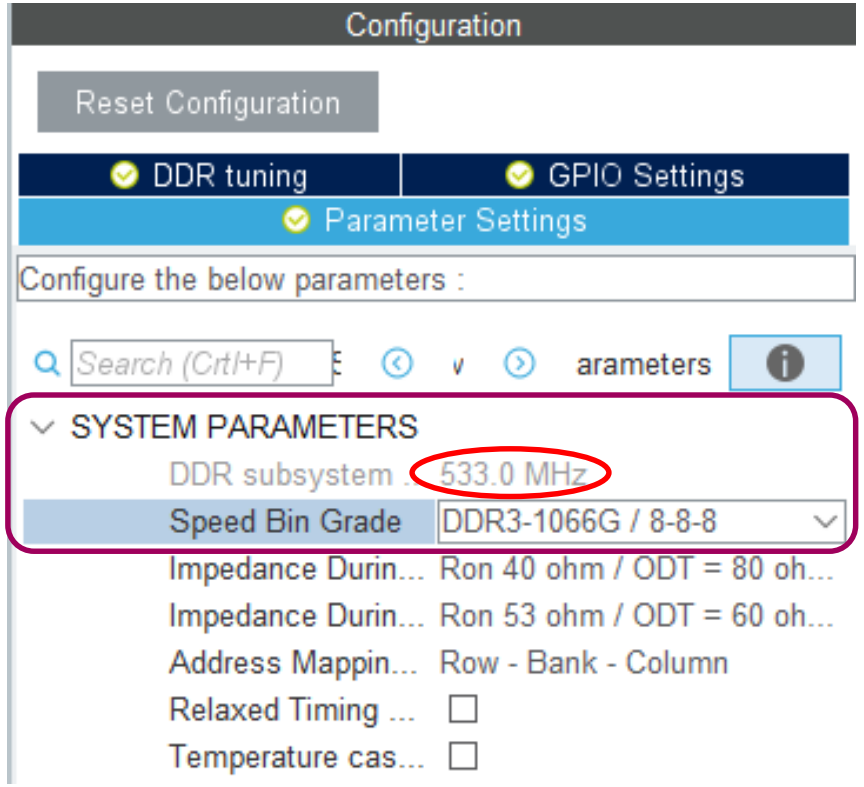
- First, need to set DDR parameters influencing the pinout from “Pinout” tab



7. Select according to DK2 discovery board memory: 16 bits 4Gb DDR3 DRAM

DDR configuration

- For DDR3, the configuration is made easier.

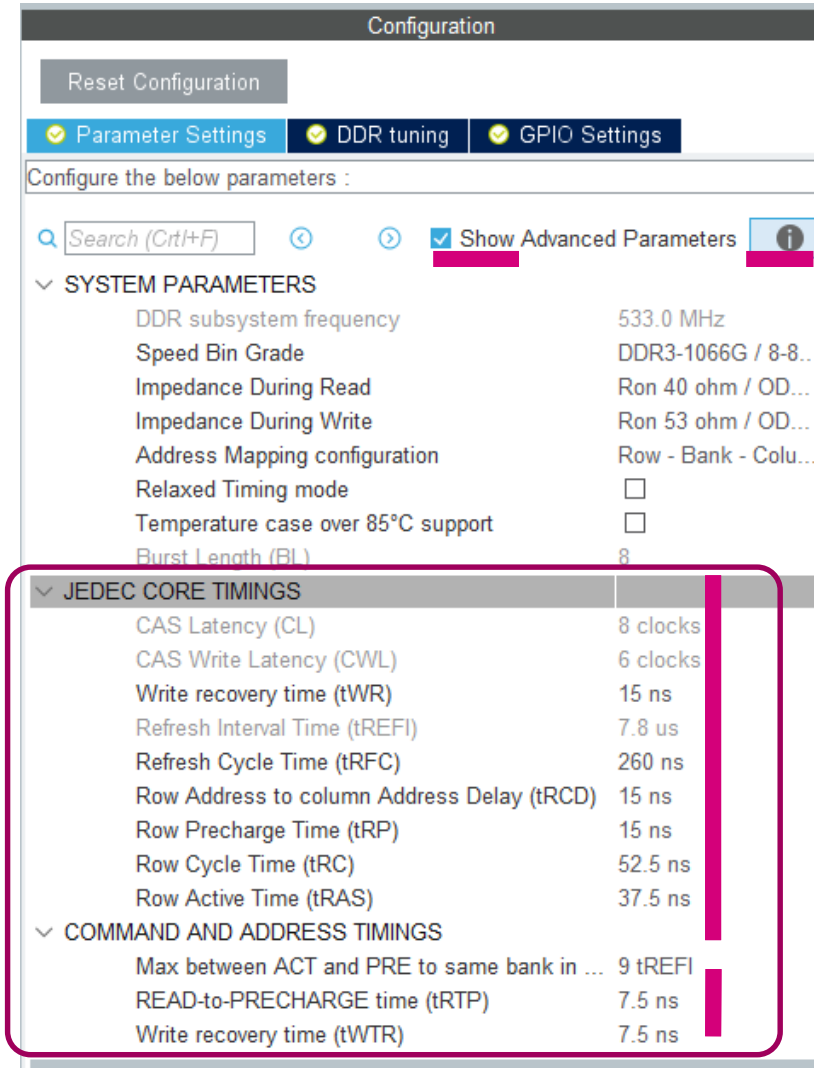


8. Select DDR3-1066G / 8-8-8

- User can select directly the Speed Bin / Grade according to the DDR RAM
  to avoid manual timing parameters edition.
  JEDEC defines DDR timings
- AN5168 for further details about DDR DRAM parameters
- For other DDR type (LPDDR2), DDR parameters must be picked up from DRAM data sheet and computed manually.

DDR configuration

- Observe other parameters from the « Parameter Settings » tab



- Timing parameters are retrieved by user from its DDR datasheet
- Some parameters, depending of DDR types, are read only. They are displayed for user information only
- “i” icone, details are provided in the window footer.

Relaxed Timing mode
RELAXED_TIMING_MODE
Parameter Description:
Add one supplementary clock cycle to some key DDR parameters.
This ease the DDR bring-up by relaxing some key timings.
The relaxed timing parameters are: tRC, tRCD and tRP.
The +1 is made before register computation, thus not visible in the parameter edition interface

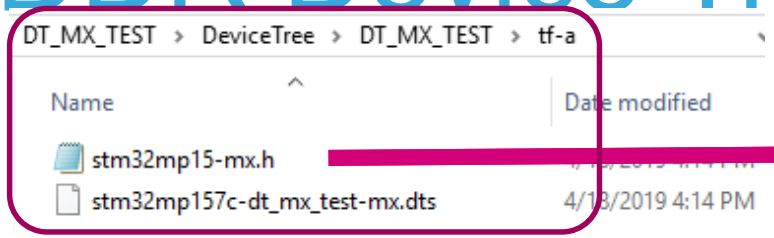
DDR configuration

- Tuning parameters

- As DDR tuning has not run, byte line delays = 0xF

- User can check modification to the Tuning parameters via the ‘DDR tuning’ tab
- Those parameters related to Read DQS position and DQ Line delay are read only in the DDR configuration panel
- Those parameters are modified after tuning operation (See DDR tuning slides)

DDR Device Tree configuration



- **First Stage boot loader (FSBL) initializes the DDR Controller**.
  All DDR controller registers values stand in FSBL device tree. STM32CubeMx generates the device tree files

- When trusted boot chain is in use, FSBL is TF-A firmware

- STM32CubeMX device tree files are to be copied in the TF-A source code location before device tree compilation
  - Refer to Wiki page related to device tree structure and device tree compilation [*Wiki: STM32MP15 Device tree*]

- DDR AN5168 provides details about STM32MP15 DDR registers

```
/*
 * Copyright (C) 2015-2018, STMicroelectronics - All Rights Reserved
 *
 * SPDX-License-Identifier: GPL-2.0+    BSD-3-Clause

 */

/*
 * File generated by STMicroelectronics STM32CubeMX DDR Tool for MPUs
 * DDR type: DDR3 / DDR3L
 * DDR width: 32bits
 * DDR density: 8Gb
 * System frequency: 400Mhz
 * Relaxed Timing Mode: false
 * Address mapping type: RBC
 *
 * Save Date: 2018.06.19, save Time: 13:18:42
 */

#define DDR_MEM_NAME    "DDR3-DDR3L 32bits 400Mhz"
#define DDR_MEM_SPEED   400
#define DDR_MEM_SIZE    0x40000000

#define DDR_MSTR 0x00040401
#define DDR_MRCTRL0 0x00000010
#define DDR_MRCTRL1 0x00000000
#define DDR_DERATEEN 0x00000000
#define DDR_DERATEINT 0x00800000
#define DDR_PWRCTL 0x00000000
#define DDR_PWRTMG 0x00400010
#define DDR_HWLPCTL 0x00000000
#define DDR_RFSHCTL0 0x00210000
#define DDR_RFSHCTL3 0x00000000
#define DDR_RFSHTMG 0x00610068
#define DDR_CRCPARCTL0 0x00000000
#define DDR_DRAMTMG0 0x0F141B0F
#define DDR_DRAMTMG1 0x000A0415
#define DDR_DRAMTMG2 0x0506080E
#define DDR_DRAMTMG3 0x0050400C
#define DDR_DRAMTMG4 0x06040406
#define DDR_DRAMTMG5 0x05050403
#define DDR_DRAMTMG6 0x02020002
#define DDR_DRAMTMG7 0x00000202
#define DDR_DRAMTMG8 0x00001005
```

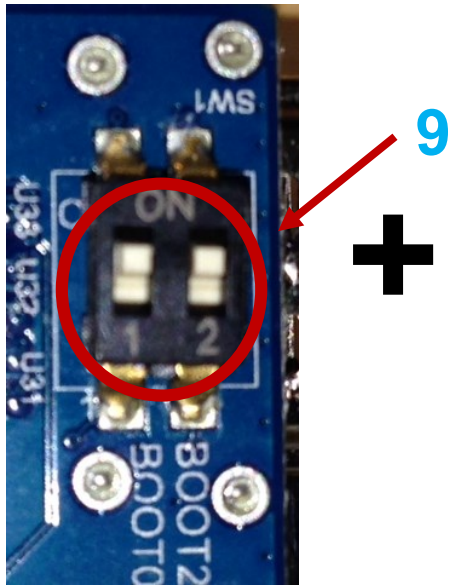
Connection to the target

Connection to Target

- **DDR tool suite** is made of: DDR interactive protocol, DDR tests and tuning process.
- **DDR tool suite** is part of U-Boot SPL binary
- To manage DDR tests and tuning, STM32CubeMX establishes a connection to **U-Boot SPL** via a dedicated **DDR interactive** protocol over UART
- Two connection options
  1. U-Boot SPL binary is available in Flash memory (Today Option used in this lab)
  2. U-Boot SPL needs to be loaded in SYSRAM
- Last option is required when Flash memory can be programmed only through STM32MP15 chip. Indeed, Flash programming requires a fully functional DDR (and thus test and tuning before hand).

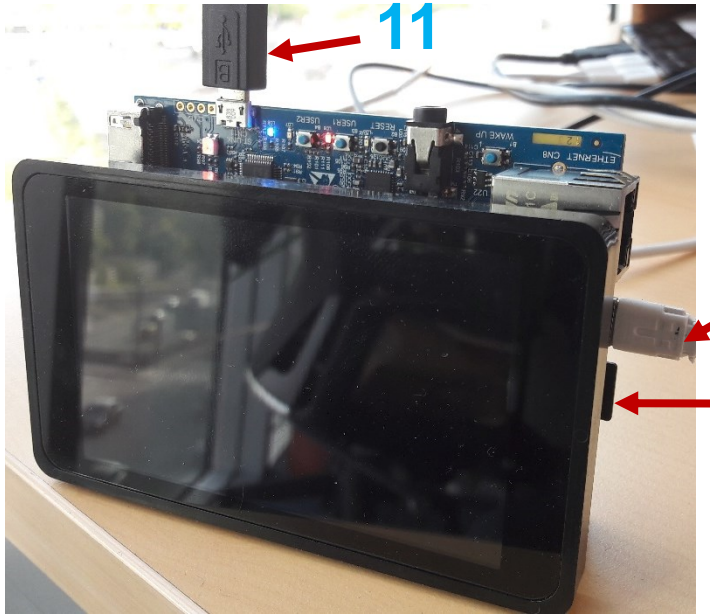
Connect to the Target

9. Make the MPU Boot from SDCard: set the boot switch to ‘11’ (On)
10. Insert the yellow SDCard in the board **with Uboot-SPL** (basic boot chain)
11. Connect the host PC to the ST-LINK-v2 port of the discovery (Micro-Usb cable)
12. Power-On the discovery with USB (USB C cable - power supply)



9

+

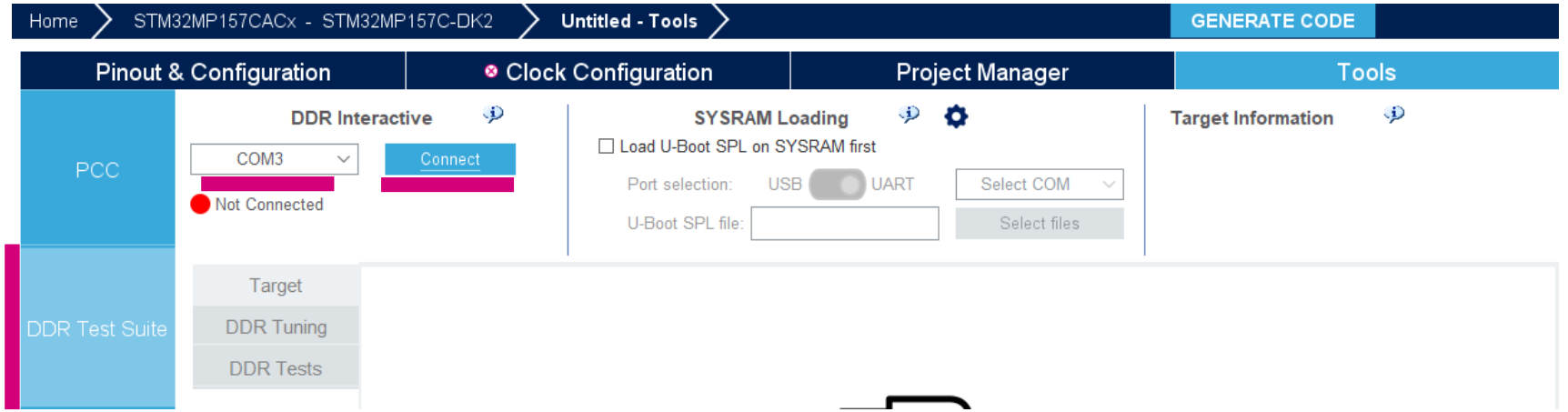


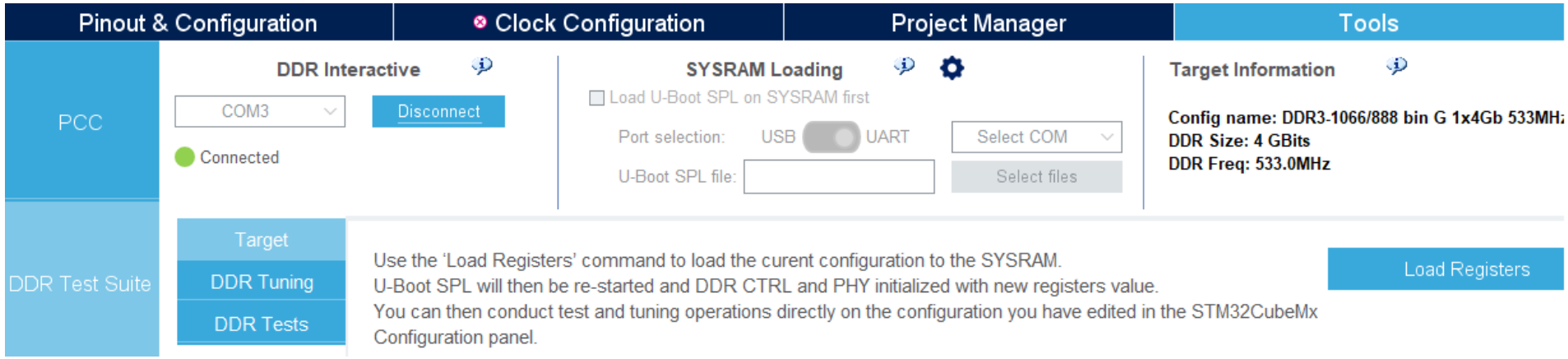
life.augmented

Connection to Target

- **U-Boot SPL binary is available in Flash memory (option 1)**

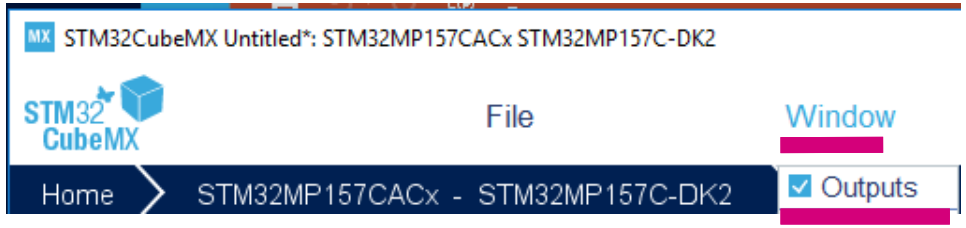
13. Press Connect icon
14. Press reset black button on the board



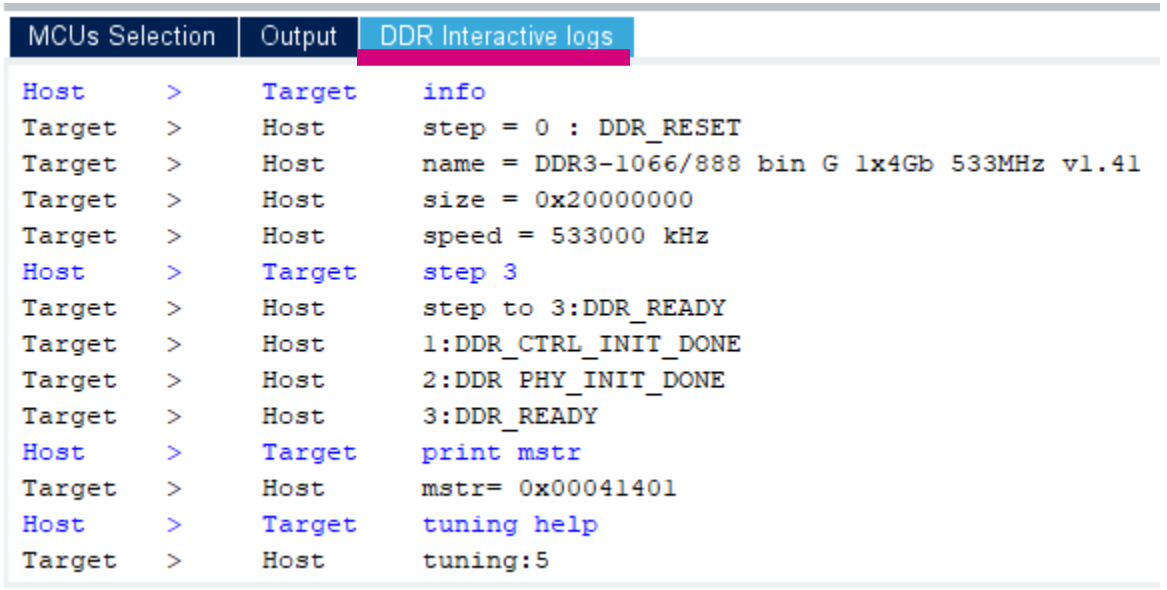


Connection to Target

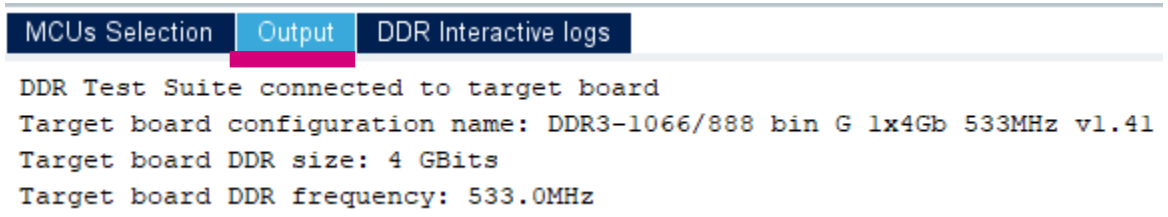
15. Display output window



16. Observe Connection log



17. Check activity logs

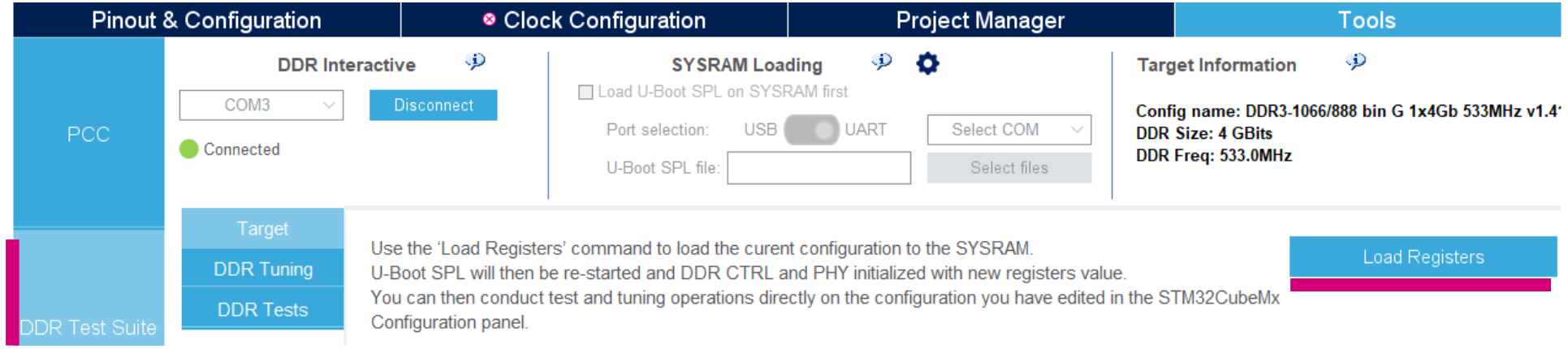


DDR registers loading

DDR registers loading

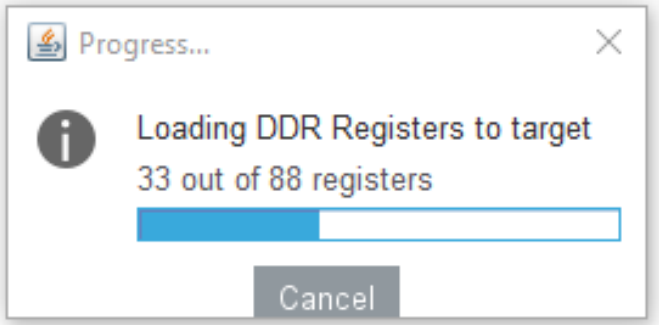
18. Once connected by DDR interactive, **user can load the current DDR configuration in SYSRAM**

- This step is **optional** if the used U-Boot SPL already contains the required DDR configuration
- It allows user to quickly test a configuration without generating device tree and dedicated U-Boot SPL binary file



DDR registers loading

- The DDR Timings parameters set in CubeMX are loaded into internal RAM.
- DDR controller is initialized.
- DDR signal are not yet fine tuned.



MCUs Selection | Output | DDR Interactive logs

```
Host     >     Target    info
Target   >     Host      step = 0 : DDR_RESET
Target   >     Host      name = DDR3-1066/888 bin G 1x4Gb 533MHz v1.41
Target   >     Host      size = 0x20000000
Target   >     Host      speed = 533000 kHz
Host     >     Target    step 3
Target   >     Host      step to 3:DDR_READY
Target   >     Host      1:DDR_CTRL_INIT_DONE
Target   >     Host      2:DDR PHY_INIT_DONE
Target   >     Host      3:DDR_READY
Host     >     Target    print mstr
Target   >     Host      mstr= 0x00041401
Host     >     Target    tuning help
Target   >     Host      tuning:5
```

DDR tuning

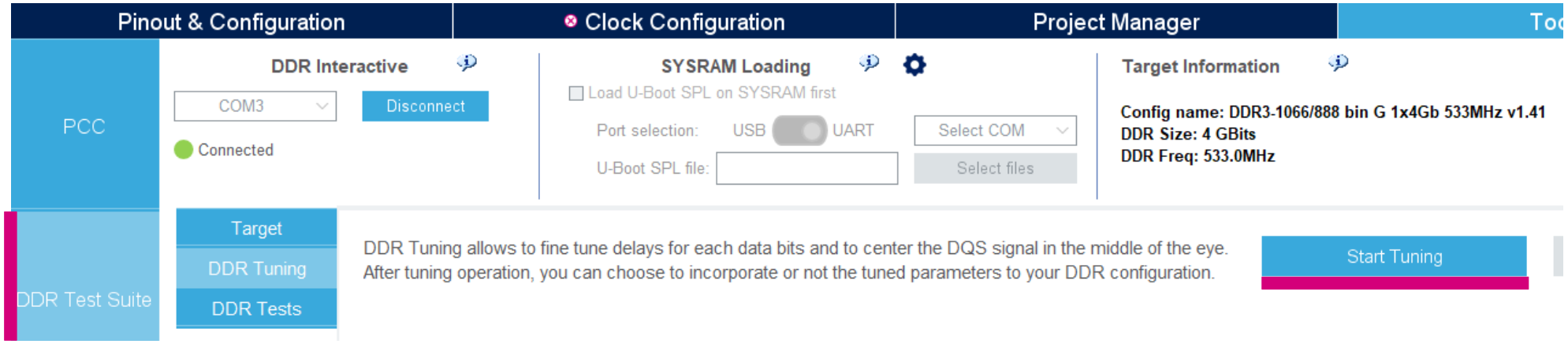
DDR tuning

- Compensate HW design slight imperfections for best operations.
- Tuning is a sequence in 3 steps
  - Read DQS gating
  - Bit deskew
  - Eye training
- Once Tuning is completed, tuned parameters can be propagated to the current DDR configuration

DDR tuning

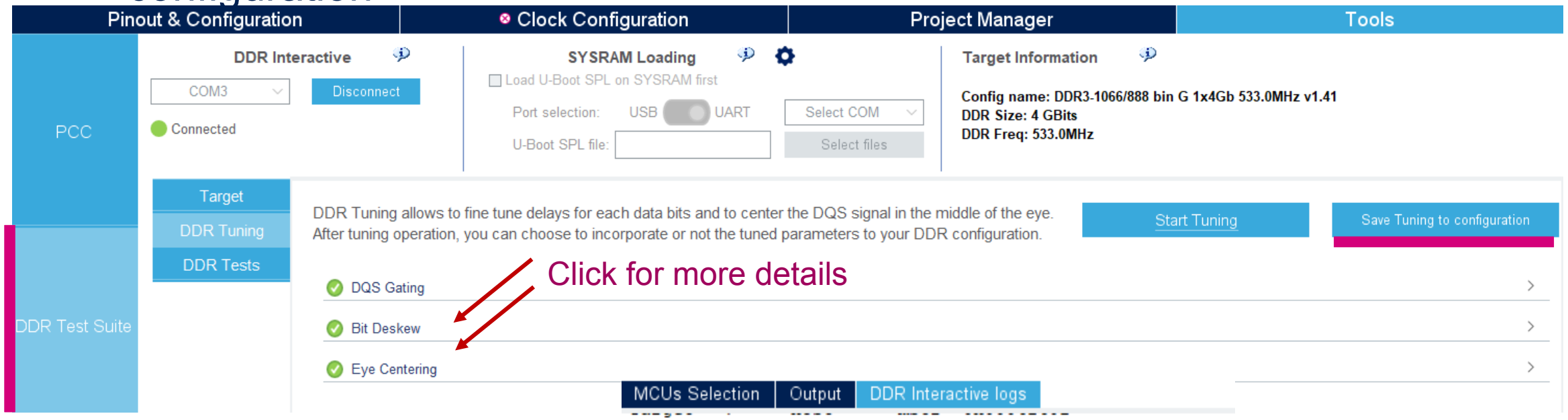
- DDR tuning operation requires :
  - DDR tool suite in connected state
  - a valid DDR configuration to be available in memory
    - either an U-Boot SPL with valid DDR configuration (DDR register file in Device Tree)
    - either the current DDR configuration written in memory (see ‘DDR registers loading’ slides)

19. Press on a Start Tuning icon



DDR tuning

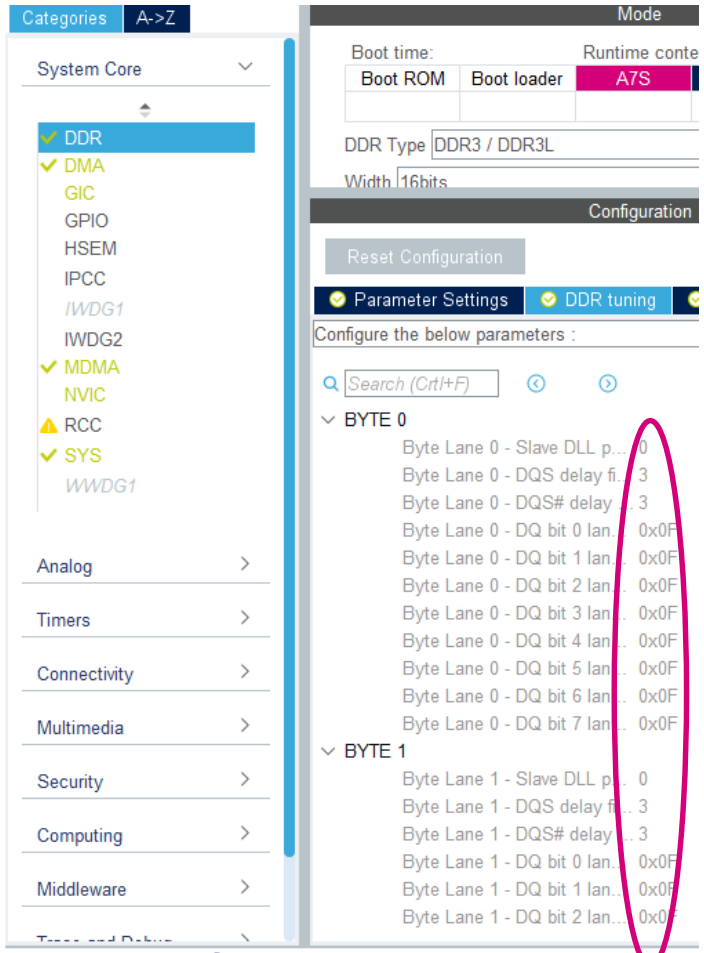
20. Once Tuning is completed, with the 3 steps in sequence, the **user is proposed to save the current tuning outcome** to the edited DDR configuration



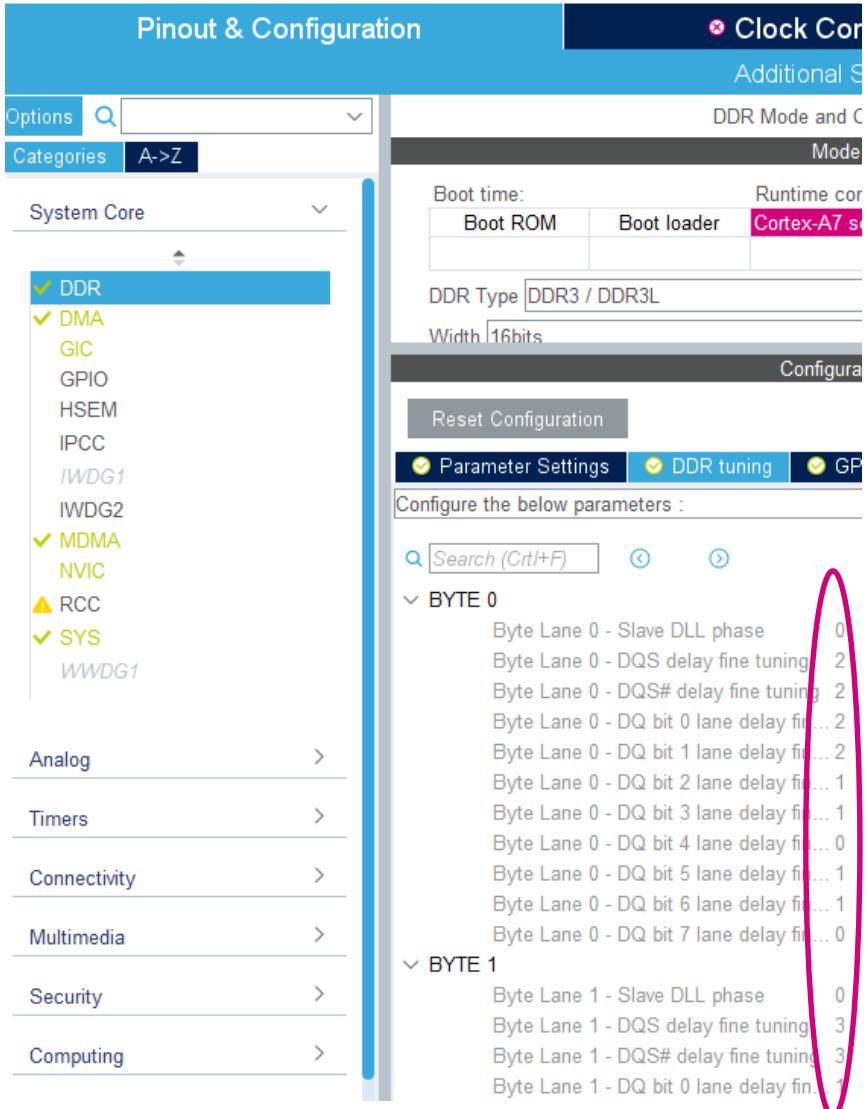
life.augmented

DDR tuning

- Observe the DDR tuned parameters

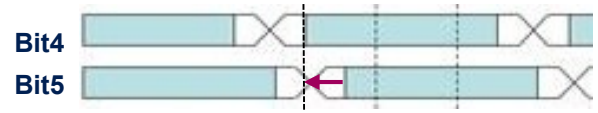


Before tuning



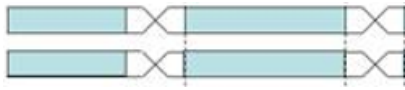
After tuning

DDR Tuning

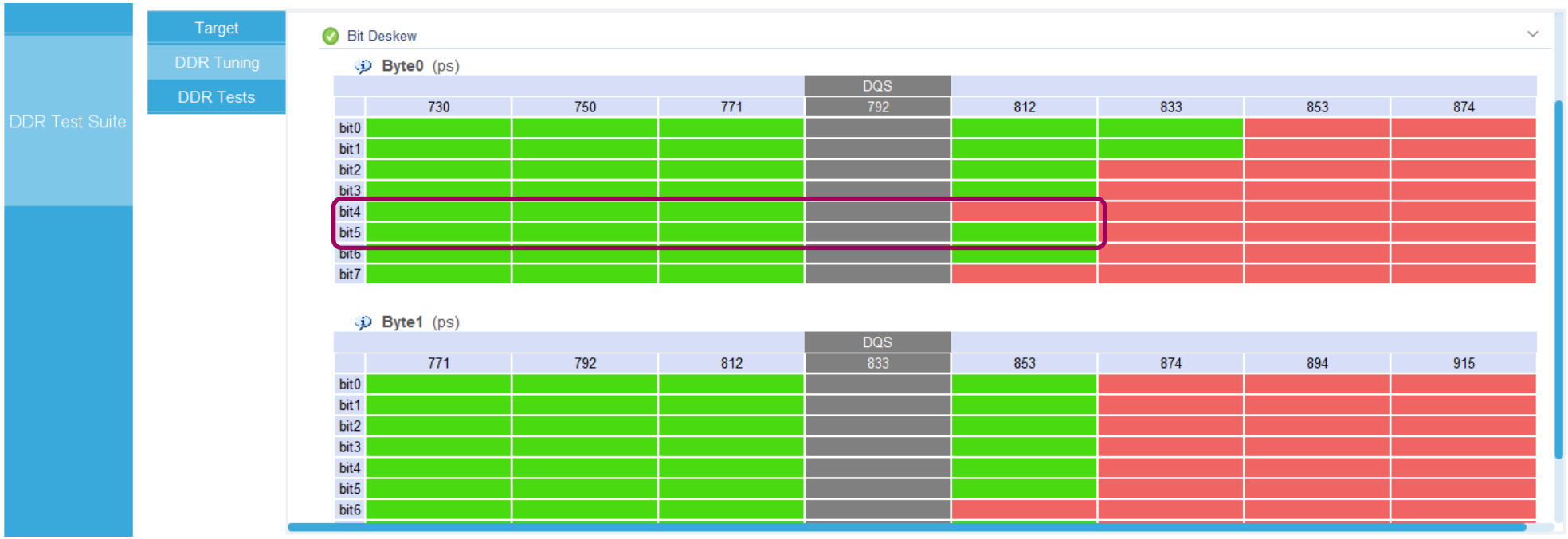


Bit4 delay = 1 step (+20 ps)
Bit5 delay = 0 (longest line, late)

Internally ==>



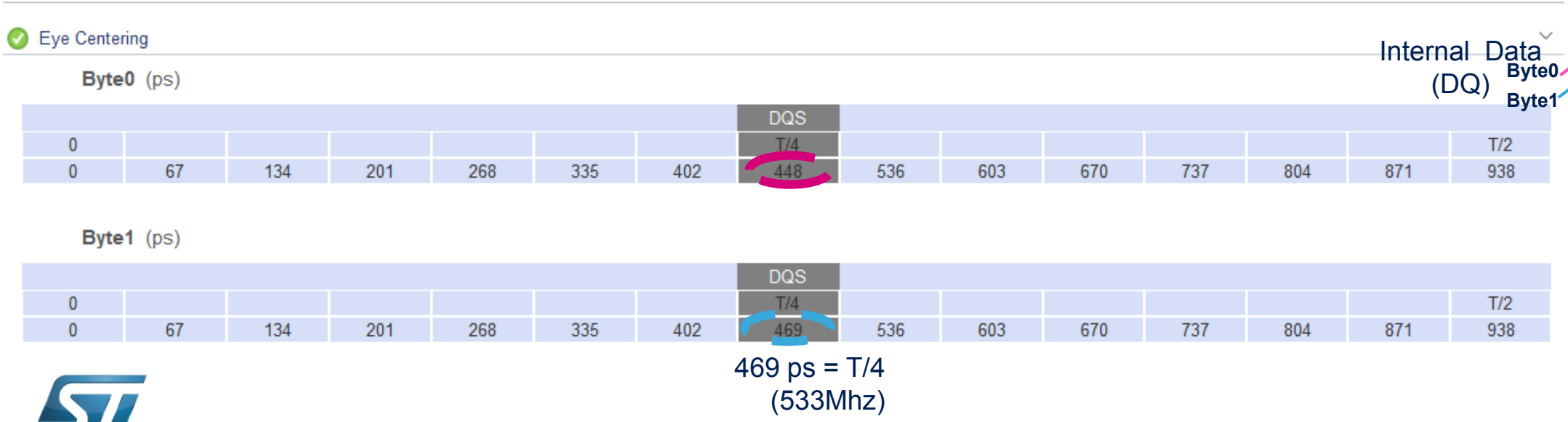
- **Bit deskew** panel gives a graphical representation of
  - Best Read DQS signal position (in grey) for the given byte
  - The delay to apply for each DQ line regarding this Read DSQ position (0 to +61.68ps in 3 steps of 20.56ps)



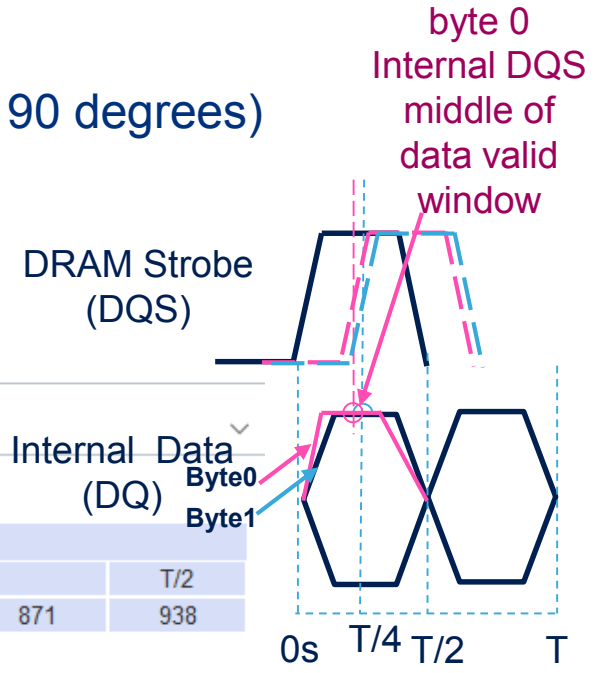
life.augmented

DDR Tuning

- **Read Eye training** panel gives the final optimum DQS position regarding DQ line eye in the half-period for each byte
  - DQS position can vary coarsely from 36 degrees to 144 degrees (quarter period is 90 degrees)
  - DQS position can then vary finely about the coarse position with 8 steps, from -61.68ps to +82.24ps



469 ps = T/4
(533Mhz)



life.augmented

DDR testing

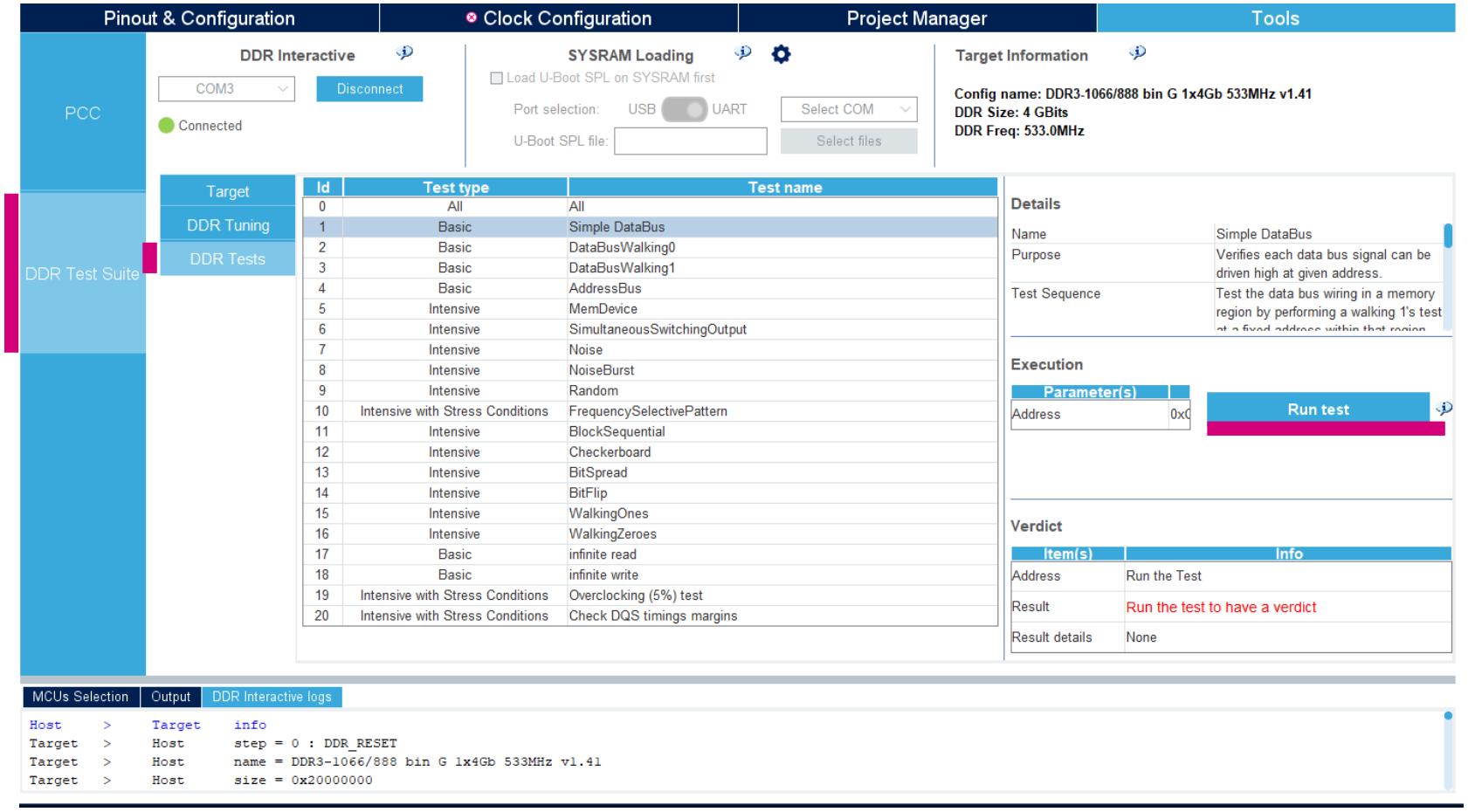
DDR testing

- Similarly to DDR Tuning, DDR testing operation requires :
  - DDR tool suite in connected state
  - a valid DDR configuration to be available in memory
    - either an U-Boot SPL with valid DDR configuration (DDR register file in Device Tree)
    - either the current DDR configuration written in memory (see ‘DDR registers loading’ slides)
- DDR tests are part of U-Boot SPL
- DDR tests are standard tests
- User can add its own DDR test by modifying U-Boot SPL

DDR testing

- **Test list is retrieved from U-Boot SPL**

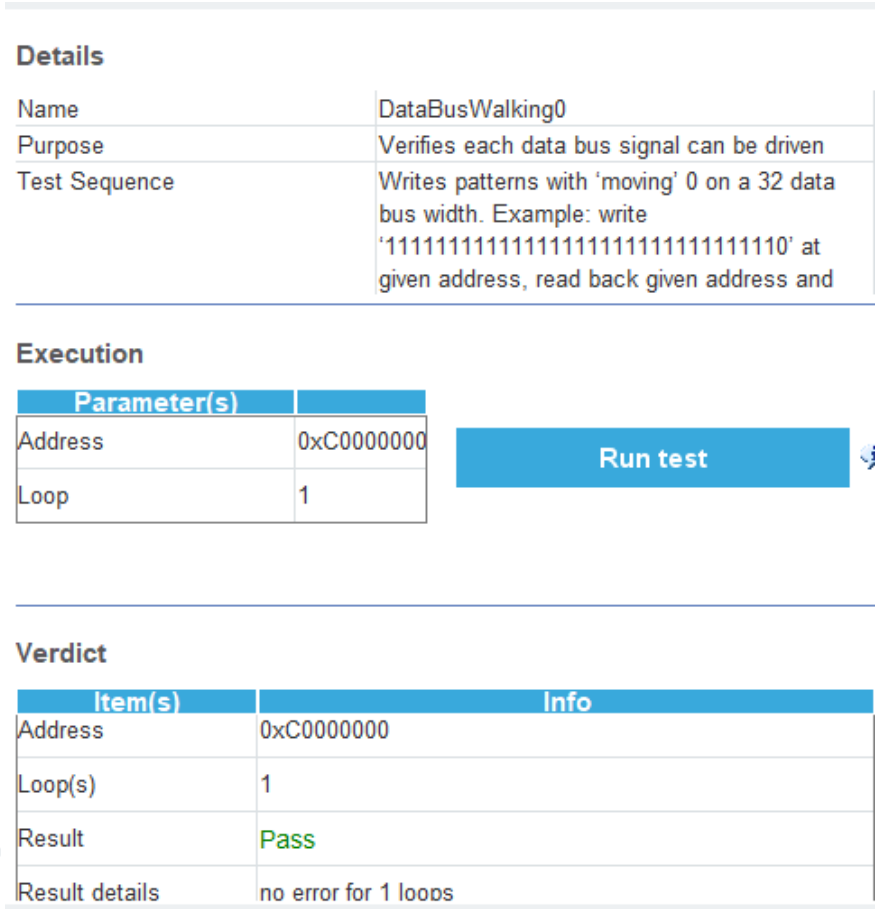
21. Press on a Run test icon



life.augmented

DDR testing

- Test **verdict is reported by U-Boot SPL**
- Parameters used for test are recalled, along with Pass/Fail status and results details
- Test history is available in the output panel



Details

Name	DataBusWalking0
Purpose	Verifies each data bus signal can be driven
Test Sequence	Writes patterns with 'moving' 0 on a 32 data bus width. Example: write '11111111111111111111111111111110' at given address, read back given address and

Execution

Parameter(s)	
Address	0xC0000000
Loop	1

Run test

Verdict

Item(s)	Info
Address	0xC0000000
Loop(s)	1
Result	Pass
Result details	no error for 1 loops

MCUs Selection | Output | DDR Interactive logs

```
Target   >   Host     Byte 1, DQS unit = 3, phase = 3
Target   >   Host     Result: Pass []
Host     >   Target   test 2 1 0xC0000000
Target   >   Host     execute 2:DataBusWalking0
Target   >   Host     running 1 loops at 0xc0000000
Target   >   Host     Result: Pass [no error for 1 loops]
```

Thanks