ENABLING TECHNOLOGIES: DDR FUNDAMENTALS

JACQUES LANDRY

FIELD APPLICATIONS ENGINEERING
AUTOMOTIVE

AMF-AUT-T2785 | AUGUST 2017

SECURE CONNECTIONS
FOR A SMARTER WORLD

PUBLIC

AGENDA

- Industry Trends
- Basic DDR SDRAM Structure
- SDRAM Differences
- General Hardware and Software Design Guidelines and Tips
- i.MX DRAM Register Programming Aid
- Layerscape DRAM Configuration and Validation via QCVS Tools

Session Introduction

- Understanding the DDR and memory controller fundamentals is key to a successful selection and design of a DDR interface

- In this session you will learn about:
  - Industry trends and fundamentals
  - DRAM Comparisons
  - Factors to consider when deciding, designing and delivering DDR on your board
  - Available DDR tools

- Who would benefit by attending this session?
  - Hardware, software and system design engineers planning to implement a DDR interface in their design

- Session length is 2 hours

Two common types of DRAM Memory

- **DDR DRAM** (focus on DDR4)
  - Typically used in PCs, Servers and Embedded Applications
  - Available in Discrete components and Modules
  - Largest total capacity
  - ECC, CRC and parity protection for high reliability systems

- **LPDDR DRAM** (focus on LPDDR4)
  - Used in Mobile and Automotive Applications (often battery powered)
  - Discrete components (small packages)
  - Low Power
  - Multi-Channel Architecture
    - Highest Speed
    - Configuration Flexibility

Industry Trends

Industry Trend

- DDR4 DRAM pricing is lower or same as DDR3\3L. The pricing crossover occurred around Q4 of 2015. Similarly, LPDDR4 is expected to crossover LPDDR3 in 2018.

- Production DDR4 DRAM, DIMMs and LPDDR4 are available from most DRAM vendors.

- The first NXP device with DDR4 support, T104x product, taped out in Q42013.The LS1043A also supports DDR4. Nearly 4 years of product experience with DDR4.
  - Many current and all future QorIQ products including T1, LS1, and LS2 products will support DDR4.

- The first NXP device with LPDDR4 support is the i.MX8 Family.

DDR3, DDR4 and LPDDR – Major Vendors

Supported by all major memory vendors

SAMSUNG

SK hynix

Micron®

ELPIDA

winbond

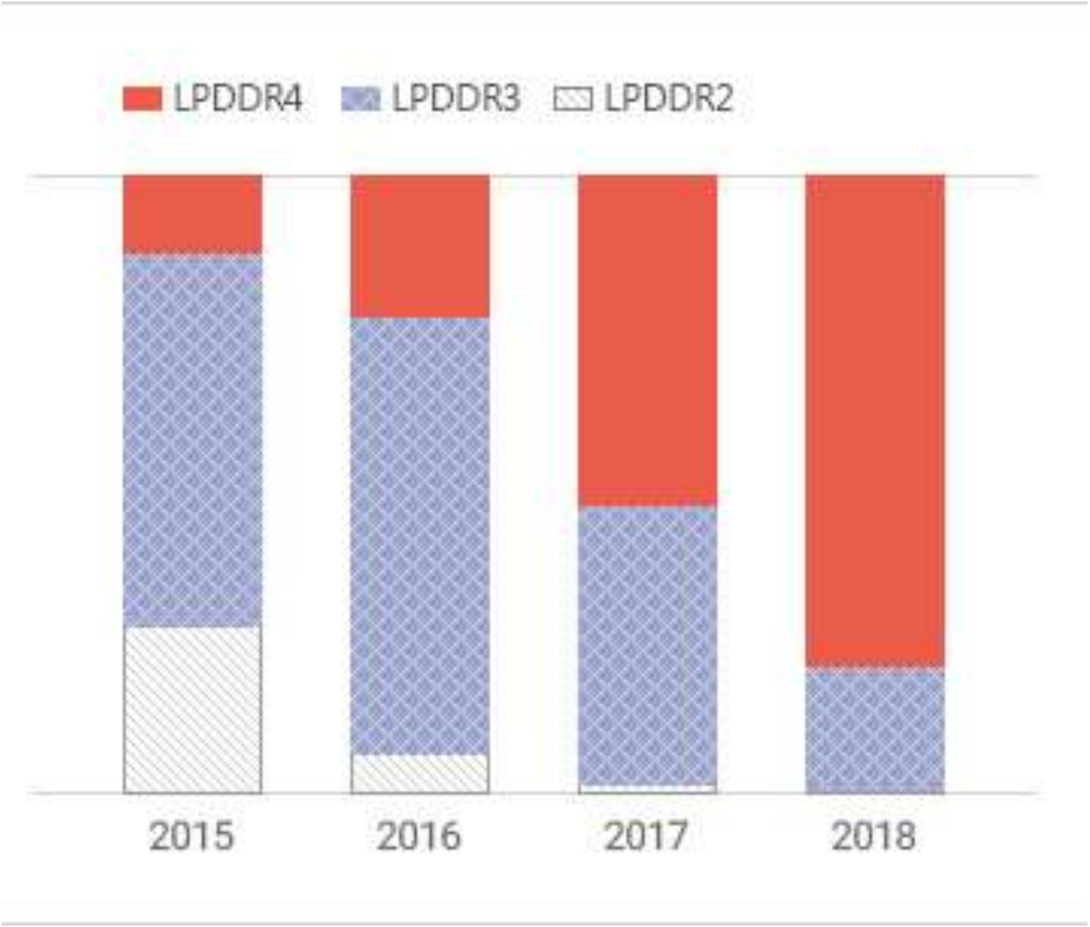
NANYA

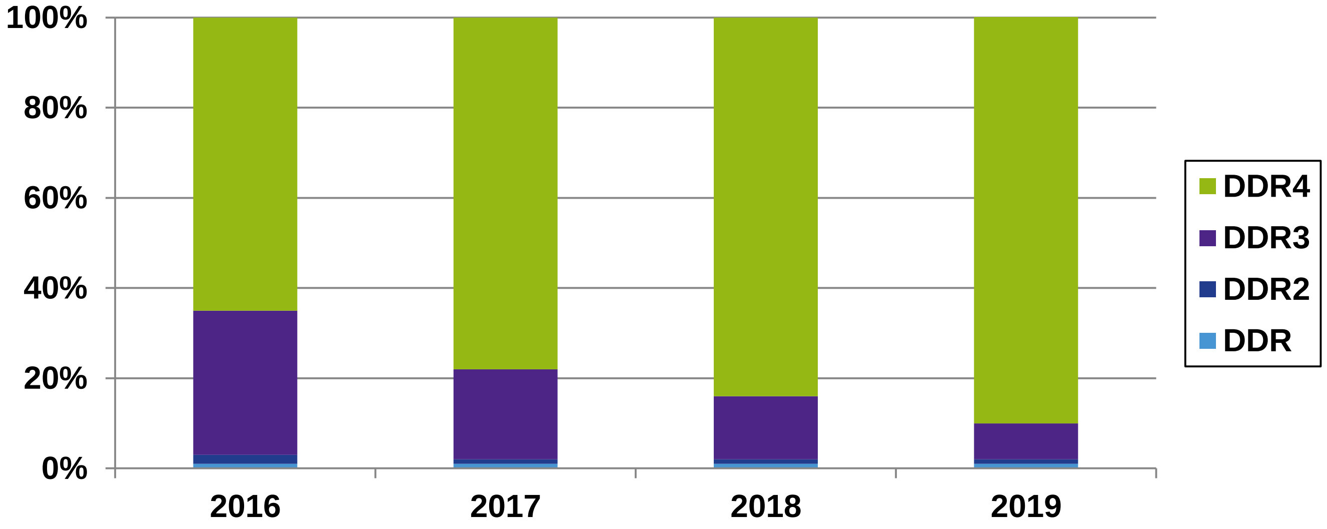
Etron 鈺創科技
Etron Technology, Inc.

ProMOS TECHNOLOGIES

Powerchip 力晶科技

DRAM Migration Roadmap





	2016	2017	2018	2019
DDR	1%	1%	1%	1%
DDR2	2%	1%	1%	1%
DDR3	32%	20%	14%	8%
DDR4	65%	78%	84%	91%

DDR3/DDR3L/DDR4 Power Saving

- DDR3 DRAM provides 20% power savings over DDR2
- DDR3L DRAM provides 10% power savings over DDR3
- DDR4 DRAM provides 37% power savings over DDR3L

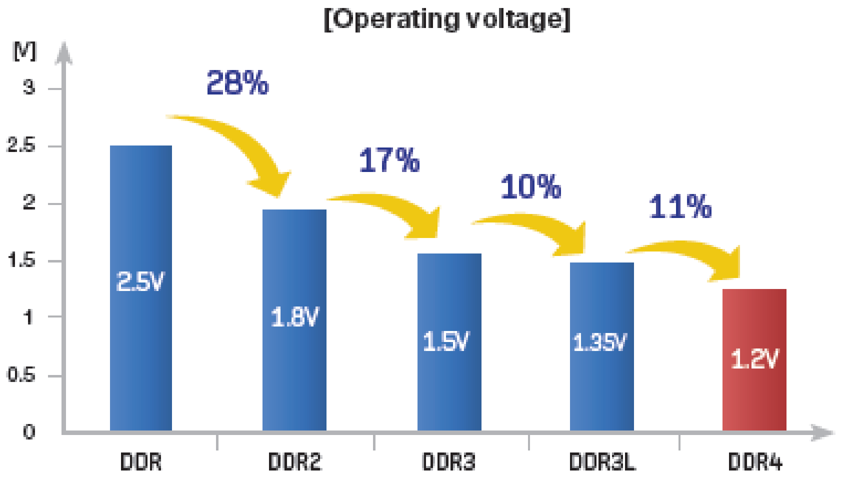


Figure 4. Reduced operating voltage requirements of DDR4 compared to DDR3L

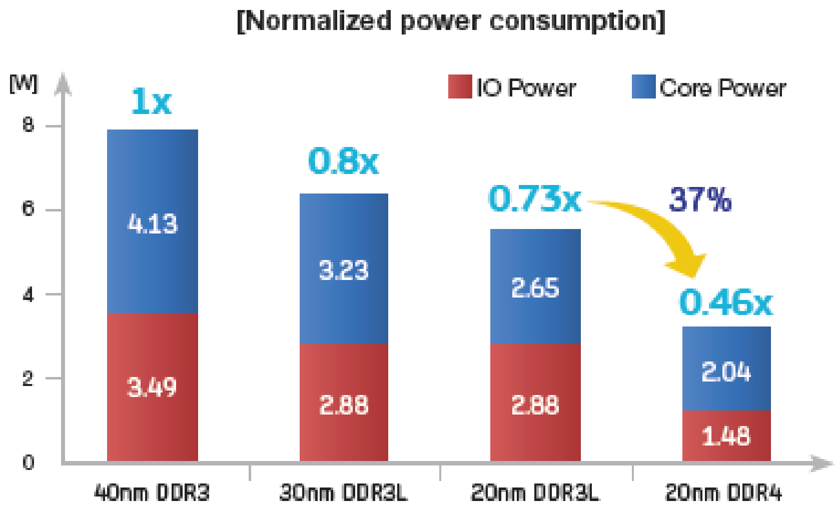


Figure 5. Reduced normalized power consumption requirements of DDR4 compared to DDR3L

LPDDR2/LPDDR3/LPDDR4 Power Saving

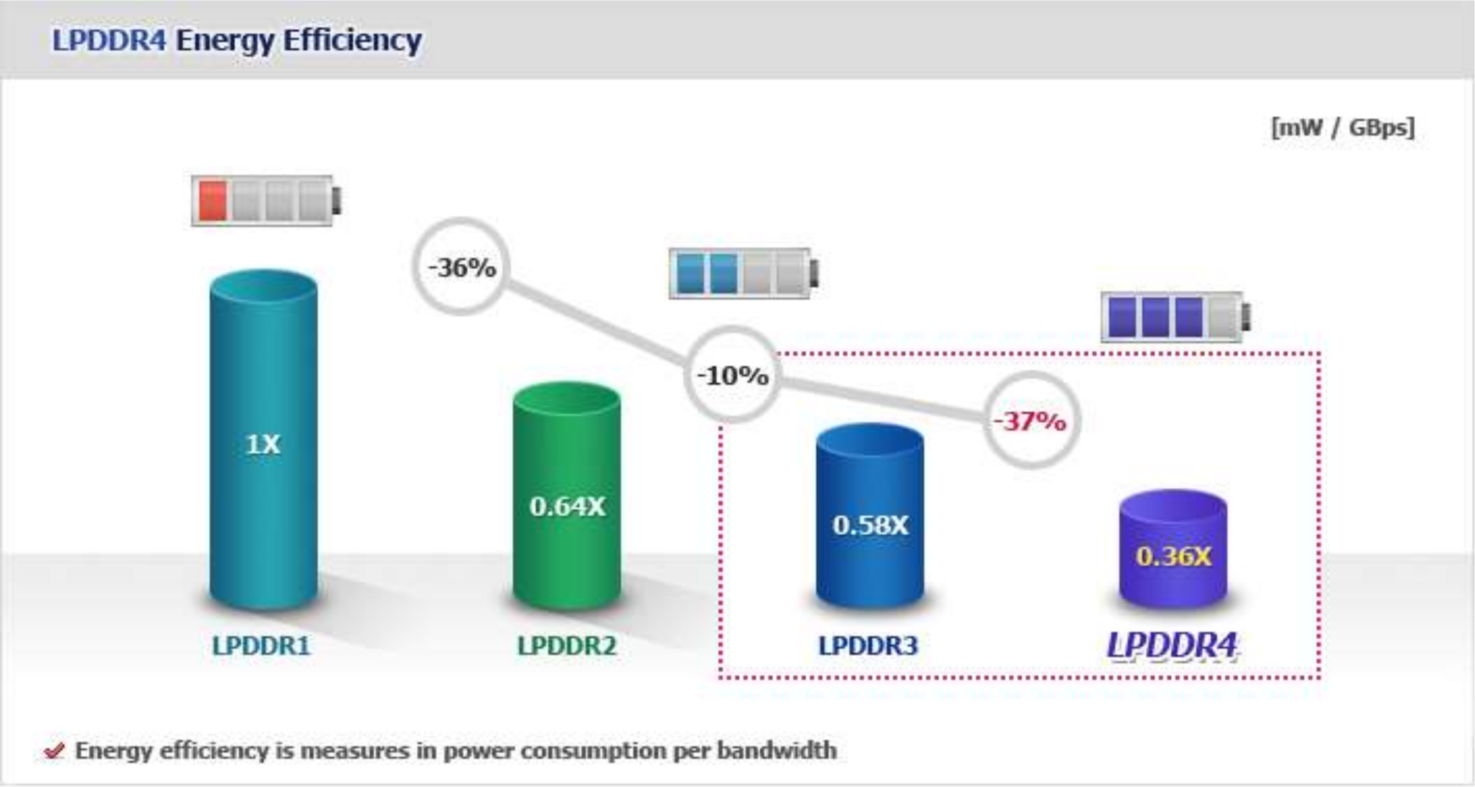
Reductions in operating voltage - LPDDR

1.8V (LPSDR, LPDDR)

1.2V (LPDDR2, LPDDR3)

1.1V, 0.6V (**LPDDR4**/LPDDR4X)

- LPDDR2 DRAM provides 36% power savings over LPDDR1
- LPDDR3 DRAM provides 10% power savings over LPDDR2
- PLDDR4 DRAM provides 37% power savings over LPDDR3



LPDDR4 Energy Efficiency

✓ Energy efficiency is measures in power consumption per bandwidth

*Source : SEC

DDR SDRAM Highlights and Comparison

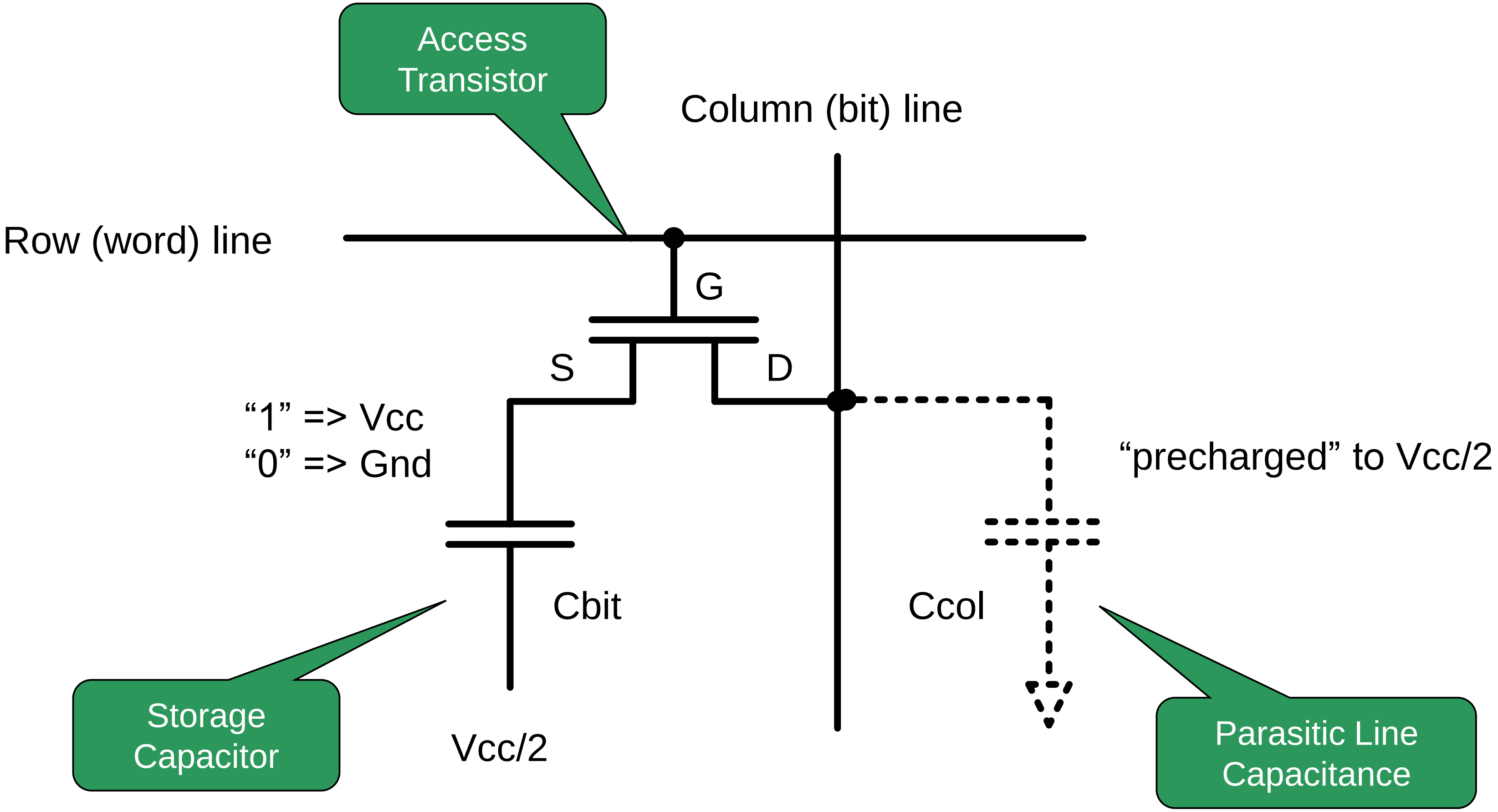
Feature/Category	DDR3	DDR4	LPDDR4
Package	BGA only	BGA only	BGA. PoP, Bare Die
Densities	512Mb -8Gb	2Gb -16Gb	2Gb to 16Gb per channel
Data Bus Organization	x4, x8, x16	x4, x8, x16	x32, (2 channels, x16), x64 (4 channels, x16)
Voltage	DDR3L:1.35V Core & I/O DDR3: 1.5V Core & I/O	1.2V Core 1.2V I/O, also 2.5V external VPP	LPDDR4: 1.1V, 1.8V Core & I/O LPDDR4X: 0.6V
Data I/O **CMD, ADDR I/O**	Center Tab Termination (CTT) CTT	Pseudo Open Drain (POD) CTT	LVSTL Programable voltage swing
Internal Memory Banks	8	16 for x4/x8 (2 BG), 8 for x16	8 per channel
Data Rate	DDR3/3L: up to2133/1866 MT/s	1600–3200 MT/s	1600-3200 MT/s (possible 4266 MT/s)
VREF	VREFCA & VREFDQ external	VREFCA external VREFDQ internal	CA Vref Internally Generated, Command Bus Training
Data Strobes/Prefetch/Burst Length/Burst Type	Differential/8-bits/BC4, BL8/ Fixed, OTF	Same as DDR3	BL16, BL32

DDR SDRAM Highlights and Comparison (cont'd)

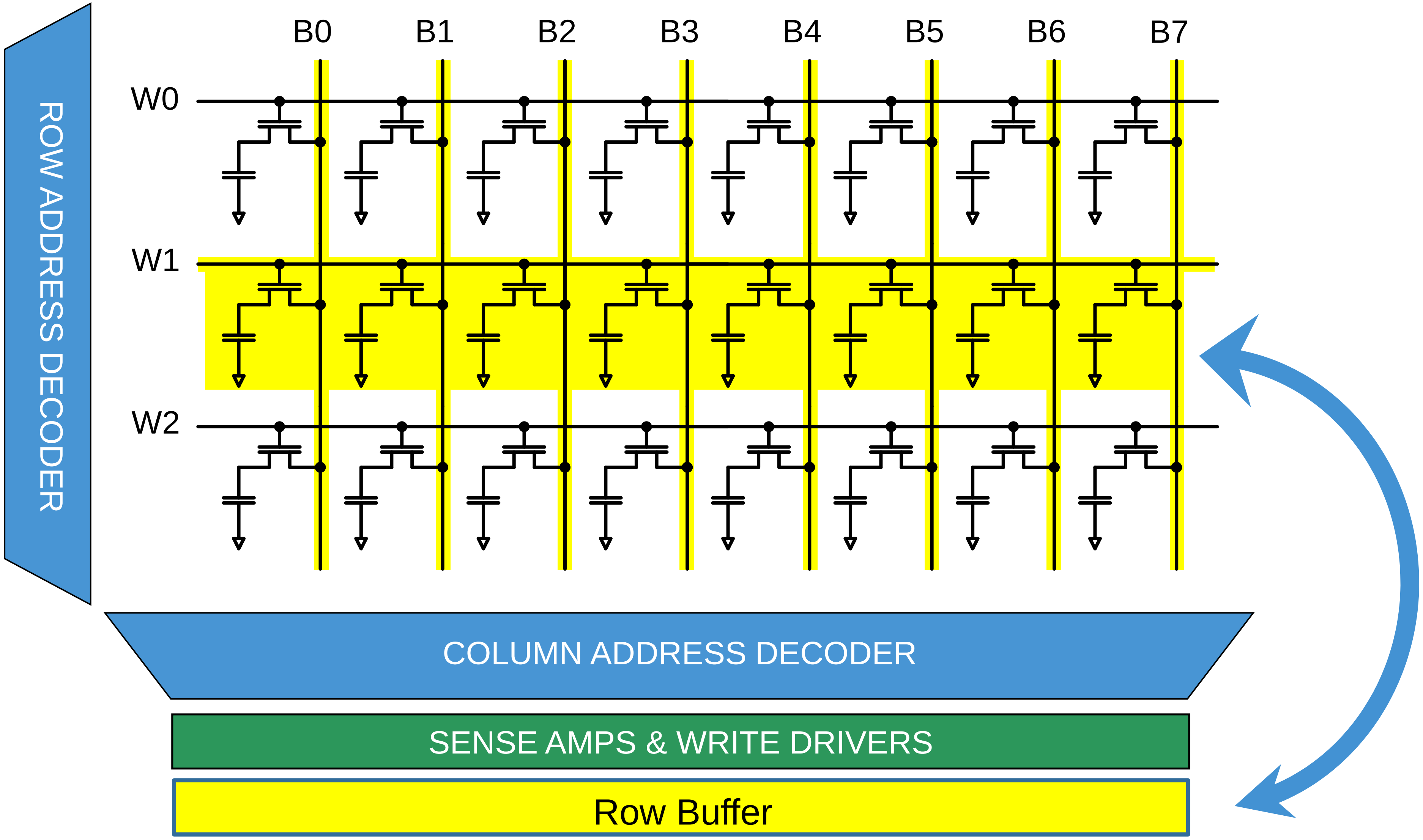
Feature/Category	DDR3	DDR4	LPDDR4
CRC Data Bus & C/A Parity	No	Yes	No
Connectivity test (TEN pin)	No	Yes	No
Bank Grouping	No	Yes	No
Data Bus Inversion	No	Yes	Yes
Write Leveling / ZQ / Reset	Yes	Yes	Yes
ACT_n new pin & command	No	Yes	No
Low power auto self-refresh	No	Yes	Yes
VREFDQ calibration	No	Yes	Yes

Basic DDR SDRAM Structure

Single Transistor Memory Cell

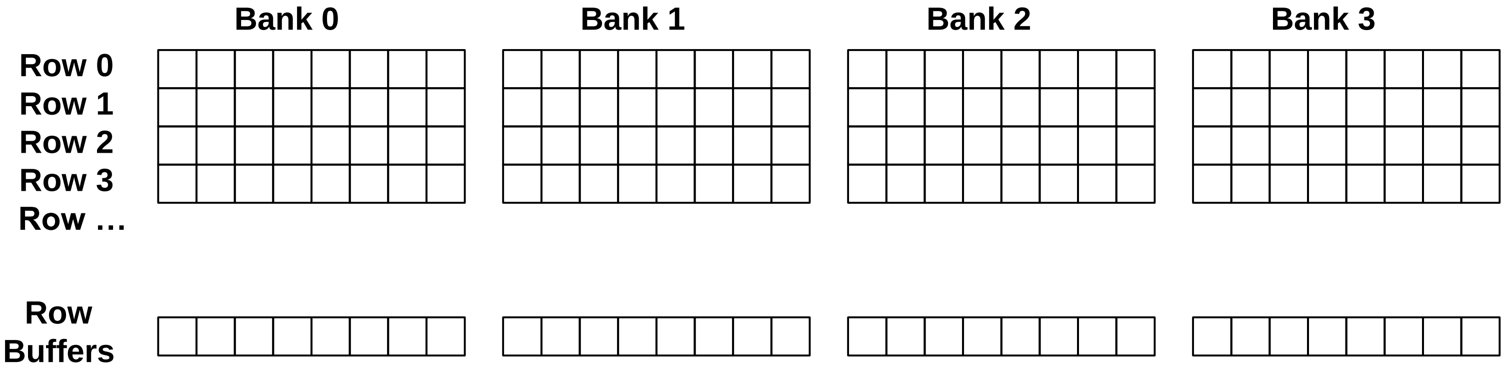


Memory Arrays



Internal Memory Banks

- Multiple arrays organized into banks
- Multiple banks per memory device
  - DDR3 – 8 banks, and 3 bank address (BA) bits
  - DDR4 – 16 banks with 4 banks in each of 4 sub bank groups
  - Can have one active row in each bank at any given time
- Concurrency
  - Can be opening or closing a row in one bank while accessing another bank

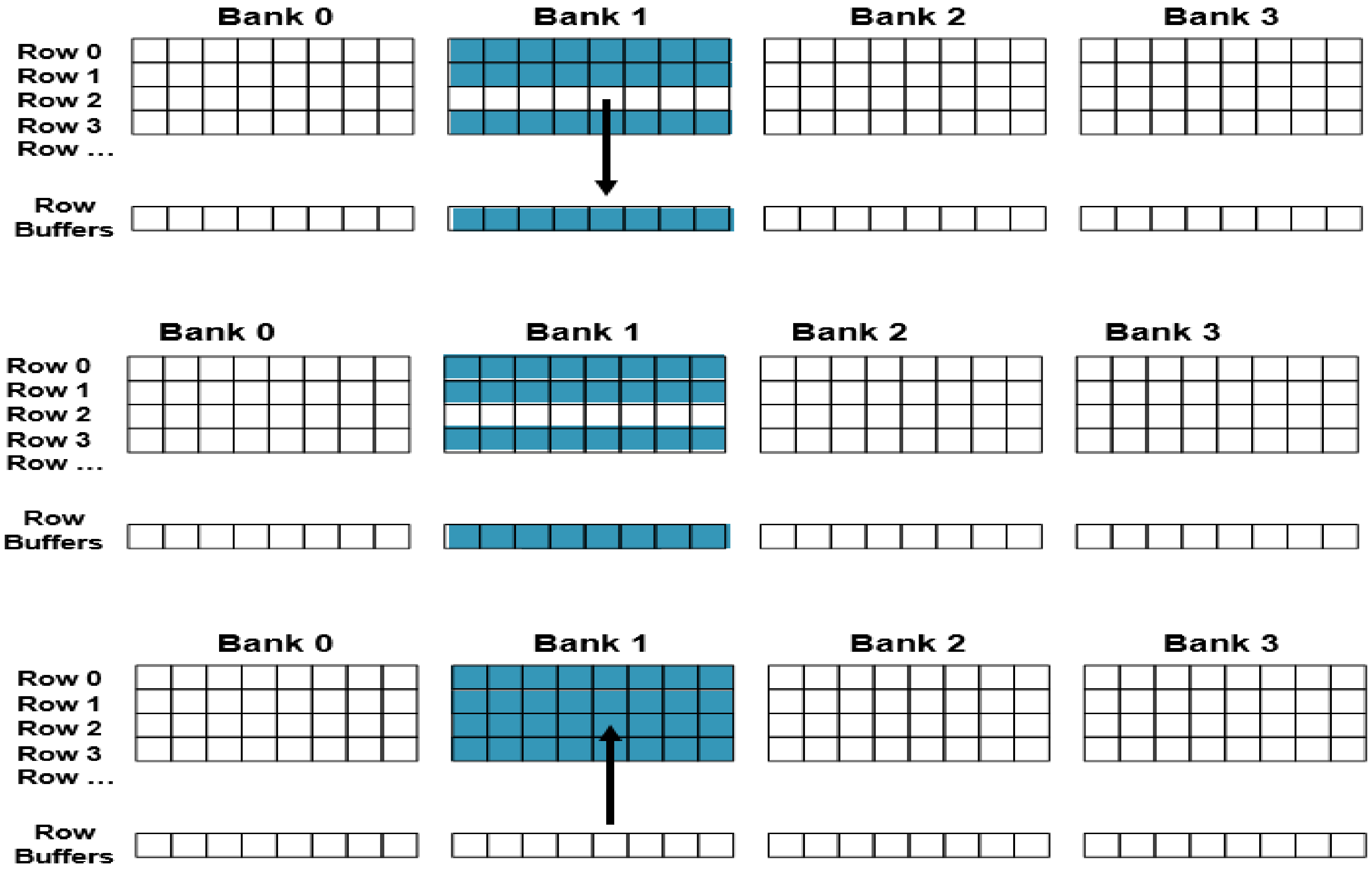


Memory Access

- A requested row is **ACTIVATED** and made accessible through the bank's row buffers

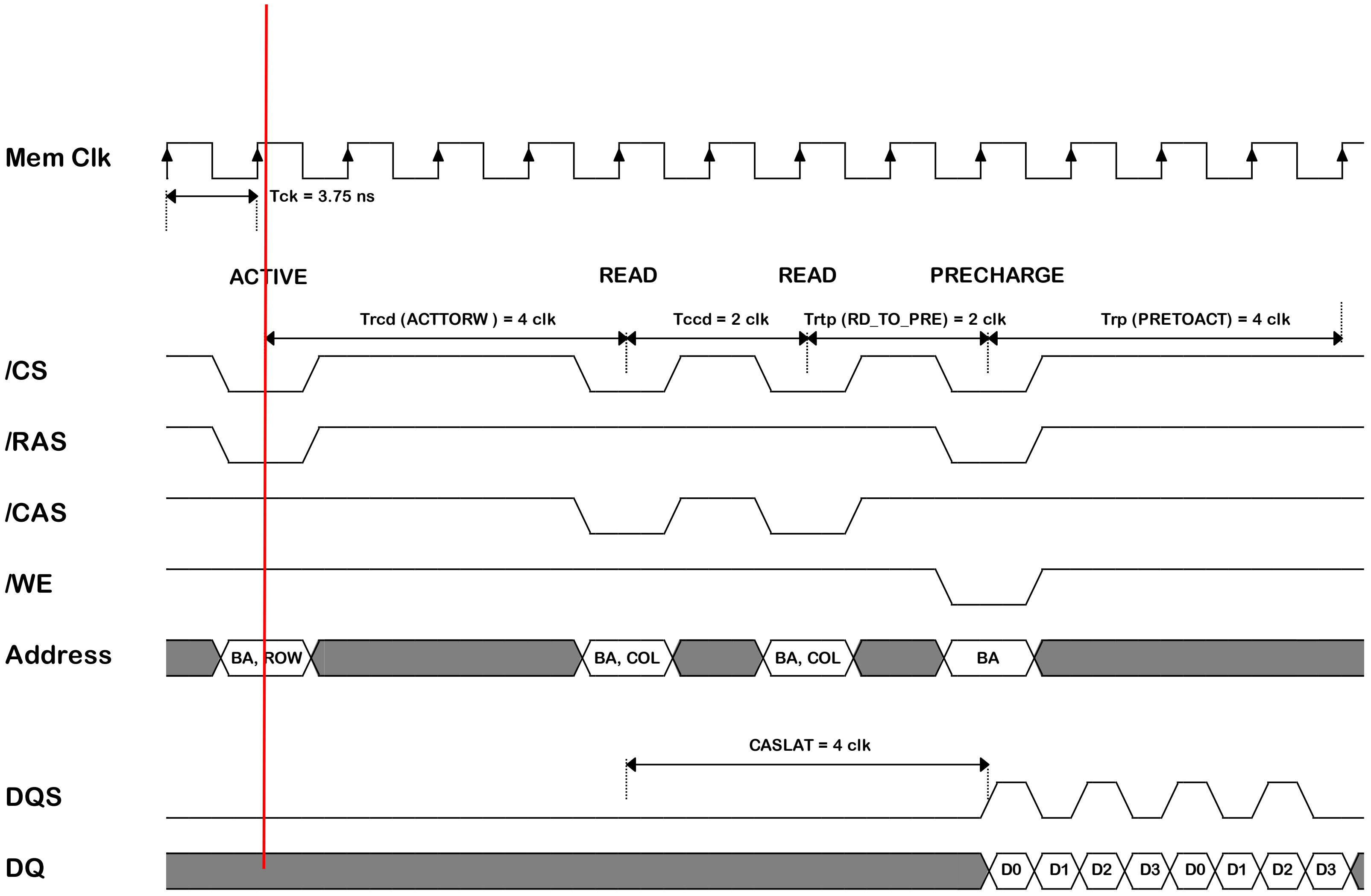
- **READ** and/or **WRITE** are issued to the active row in the row buffers

- The row is **PRECHARGED** and is no longer accessible through the bank's row buffers



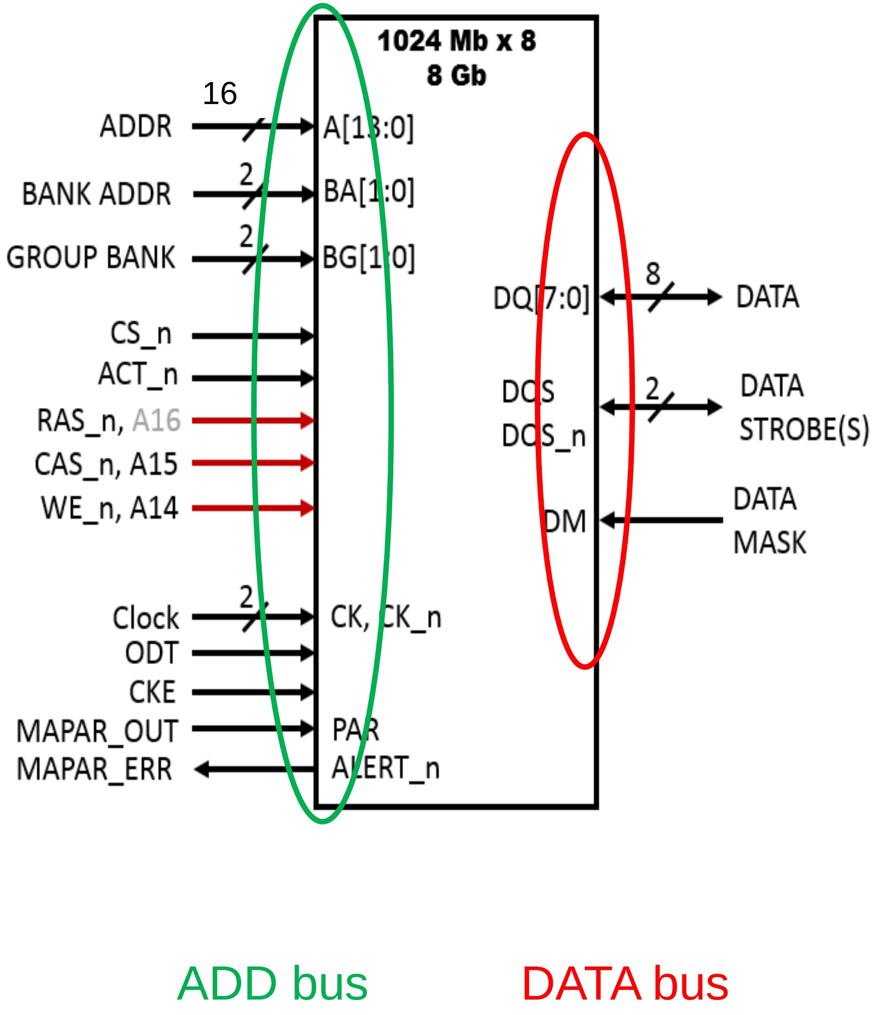
Example: DDR4-2133
Open Page = 2.133Gb/s maximum bandwidth
Closed Page = 199Mb/s maximum bandwidth
10x performance advantage to read and write from an open page

DDR2-533 Read Timing Example



Example – 8Gb DDR4 SDRAM

- Micron MT40A1G8
- 1024M x 8 (64M x 8 x 16 banks)
- 8 Gb total
- 16-bit row address
  - 64K rows
- 10-bit column address
  - 1K bits/row (1KB in x8 data with DRAM)
- 2-bit group and 2-bit bank address
- DATA bus: DQ, DQS, /DQS, DM (DBI)
- ADD bus: A, BA, GB, ACT, /CS, /RAS, /CAS, /WE, ODT, CKE, CK, /CK, PAR, /ALERT

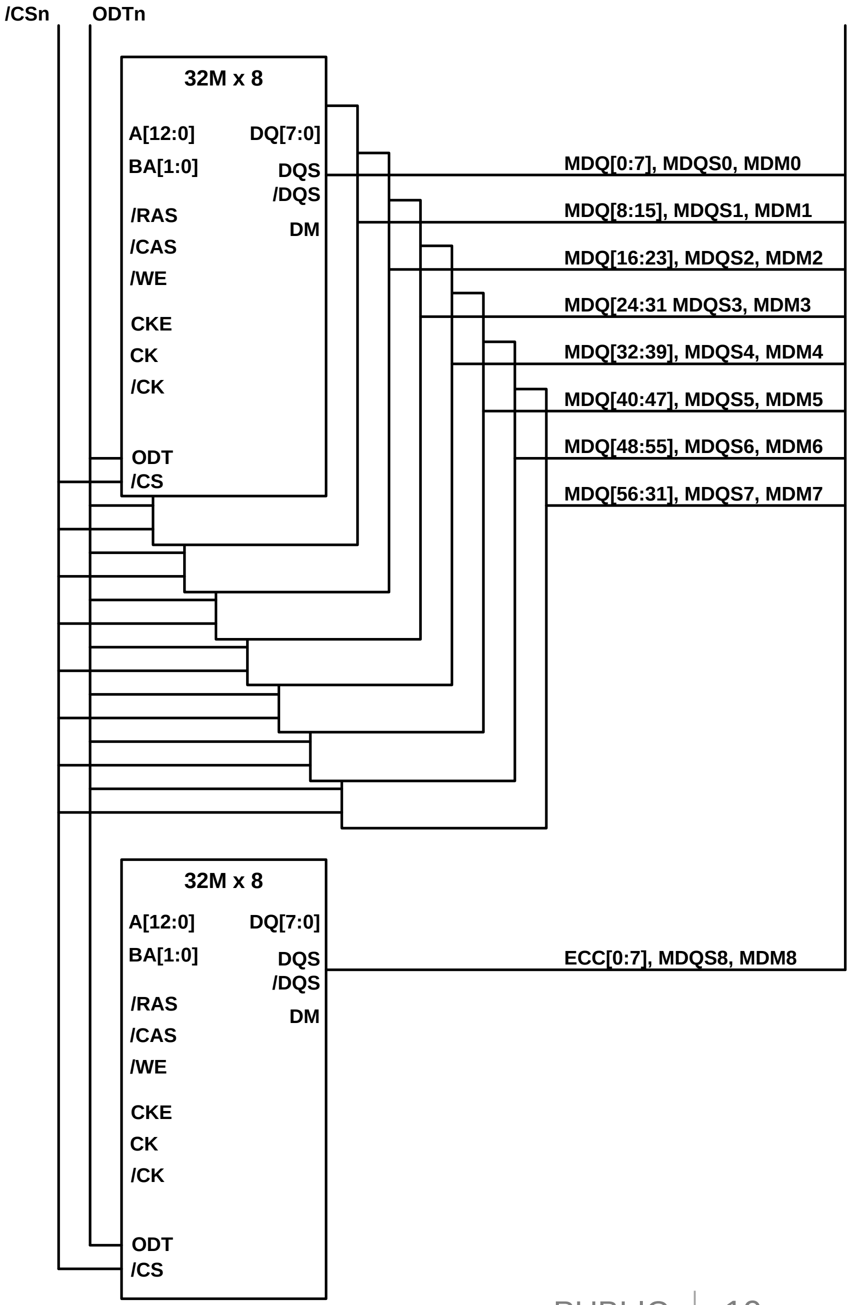


Example – DDR4 UDIMM

- Micron MTA9ASF51272AZ
- 9 each 512M x 8 DRAM devices
- 512M x 72 overall
- 4 GB total, single “rank”
- 9 “byte lanes”

Two Signal Bus

- 1- Address, command, control, and clock signals are shared among all 9 DRAM devices
- 2- Data, strobe, data mask not shared



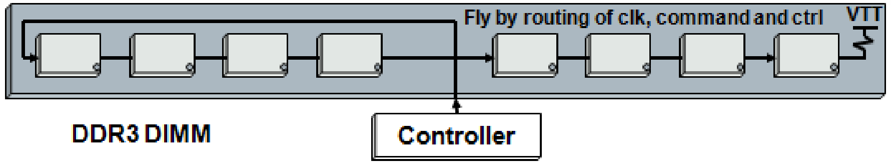
DRAM Module Type

UDIMM: Unbuffered Desktop standard

MiniDIMM:
Computing and Networking

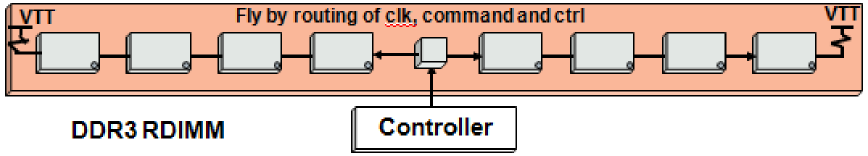
SODIMM: Notebook standard

VLP MiniDIMM:
Computing and Networking



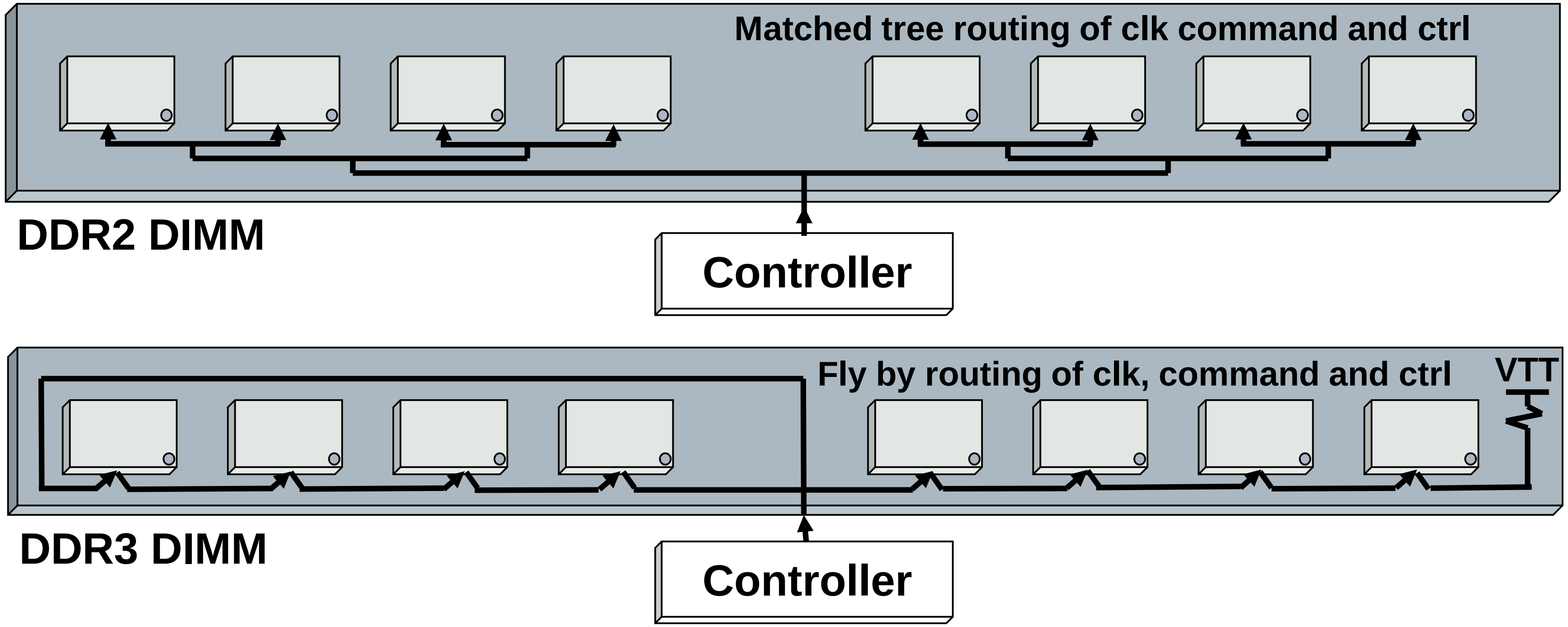
RDIMM: Registered Server standard

VLP RDIMM: Very Low Profile
Computing and Networking



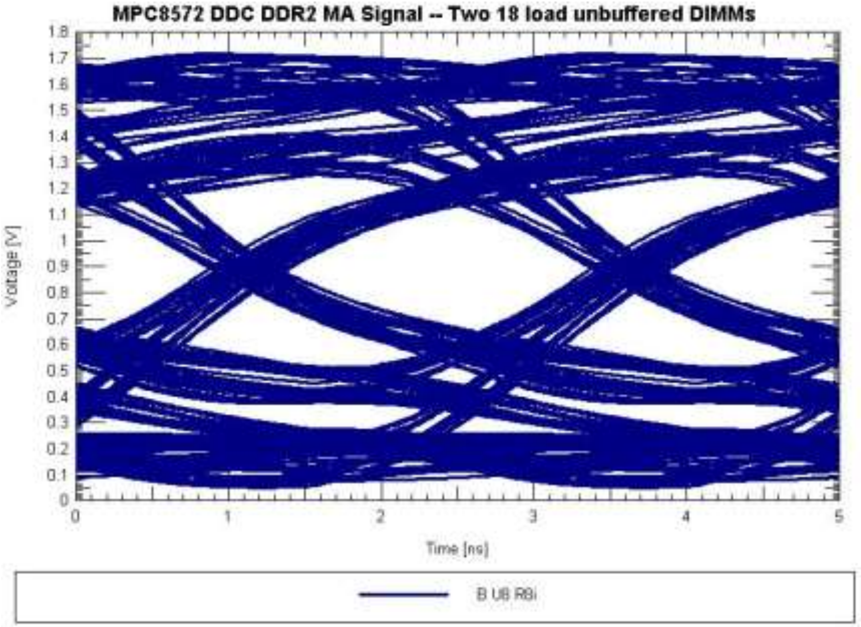
Fly-By Routing Topology

- **Introduction of “fly-by” architecture**
  - Address, command, control & clocks
  - Data bus (not illustrated below) remains unchanged, ie, direct 1-to-1 connection between the Controller bus lanes and the individual DDR devices.
  - Improved signal integrity…enabling higher speeds
  - On module termination

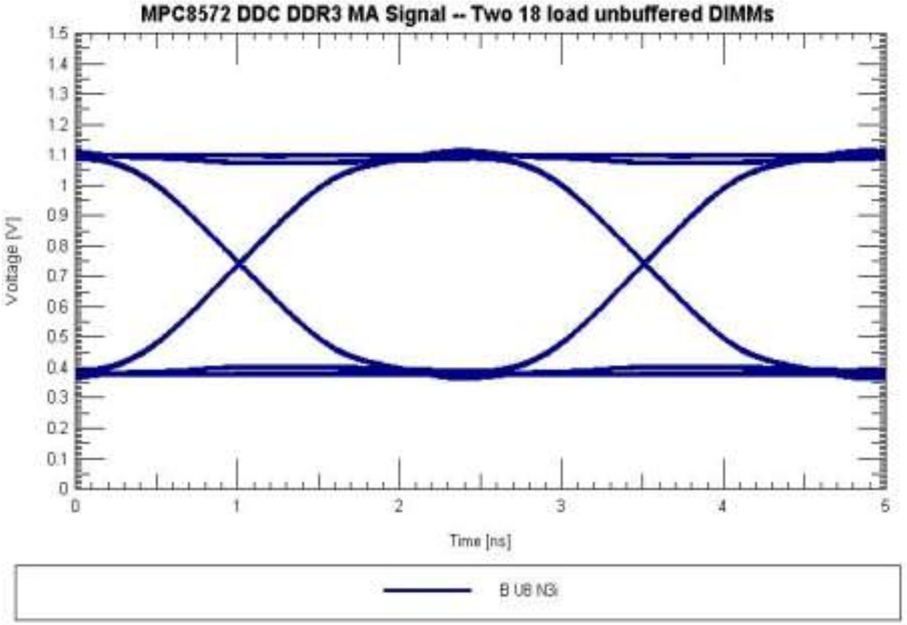

Matched tree routing of clk command and ctrl
DDR2 DIMM
Controller
Fly by routing of clk, command and ctrl
VTT
DDR3 DIMM
Controller

Fly-By Routing Improved SI

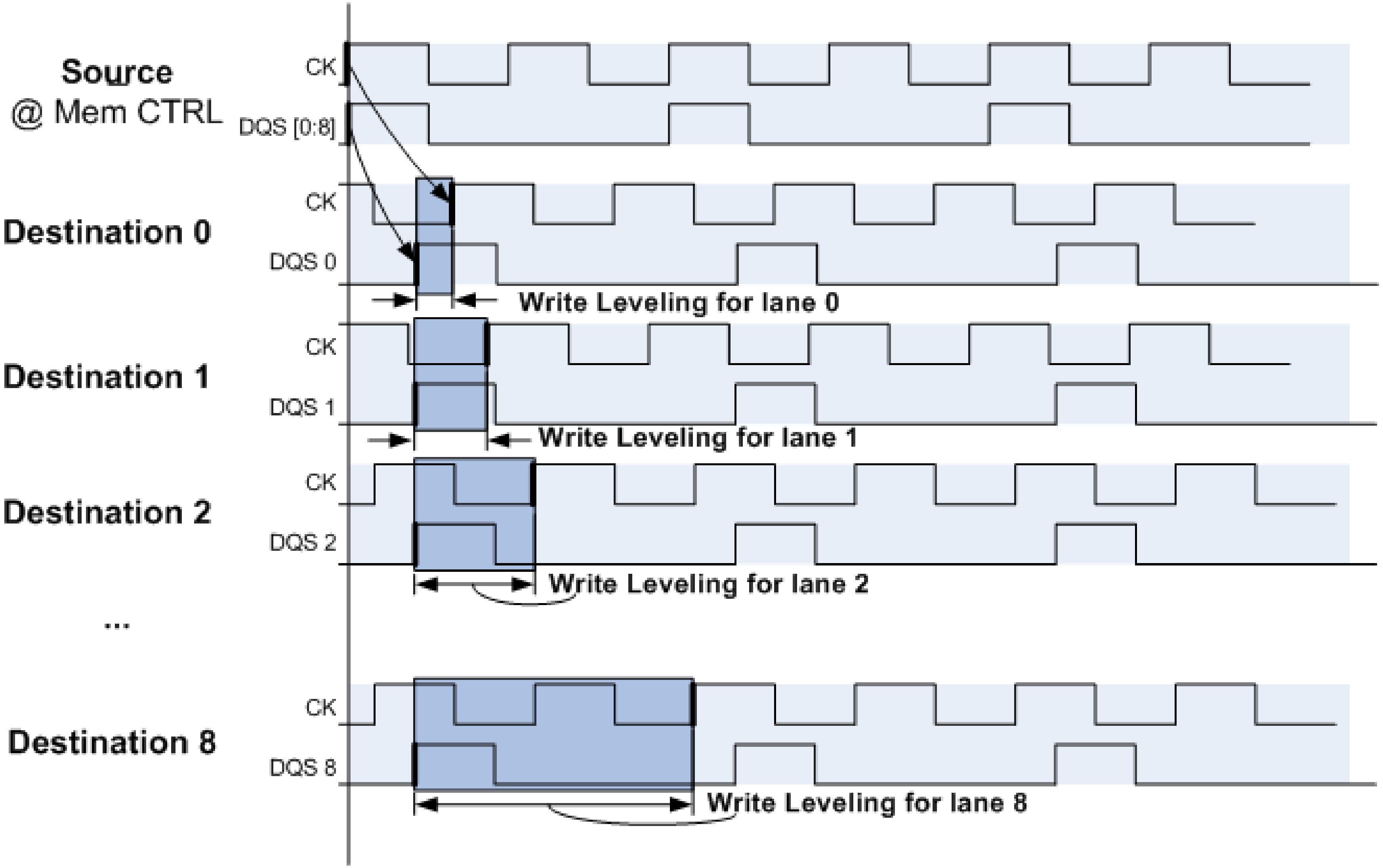
DDR2 Matched Tree Routing



DDR3 Fly By Routing

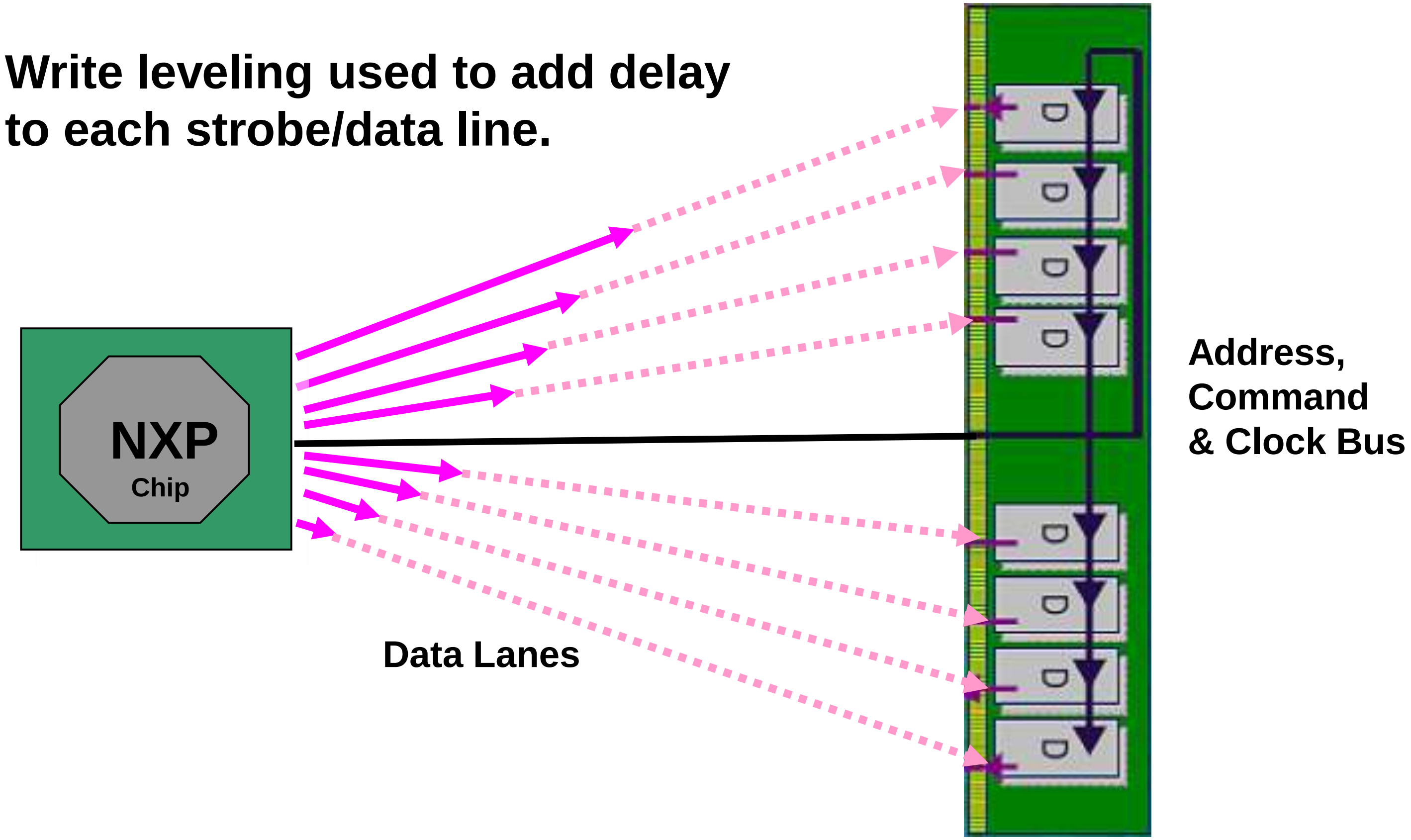


What Is Write Leveling?



Write Adjustment

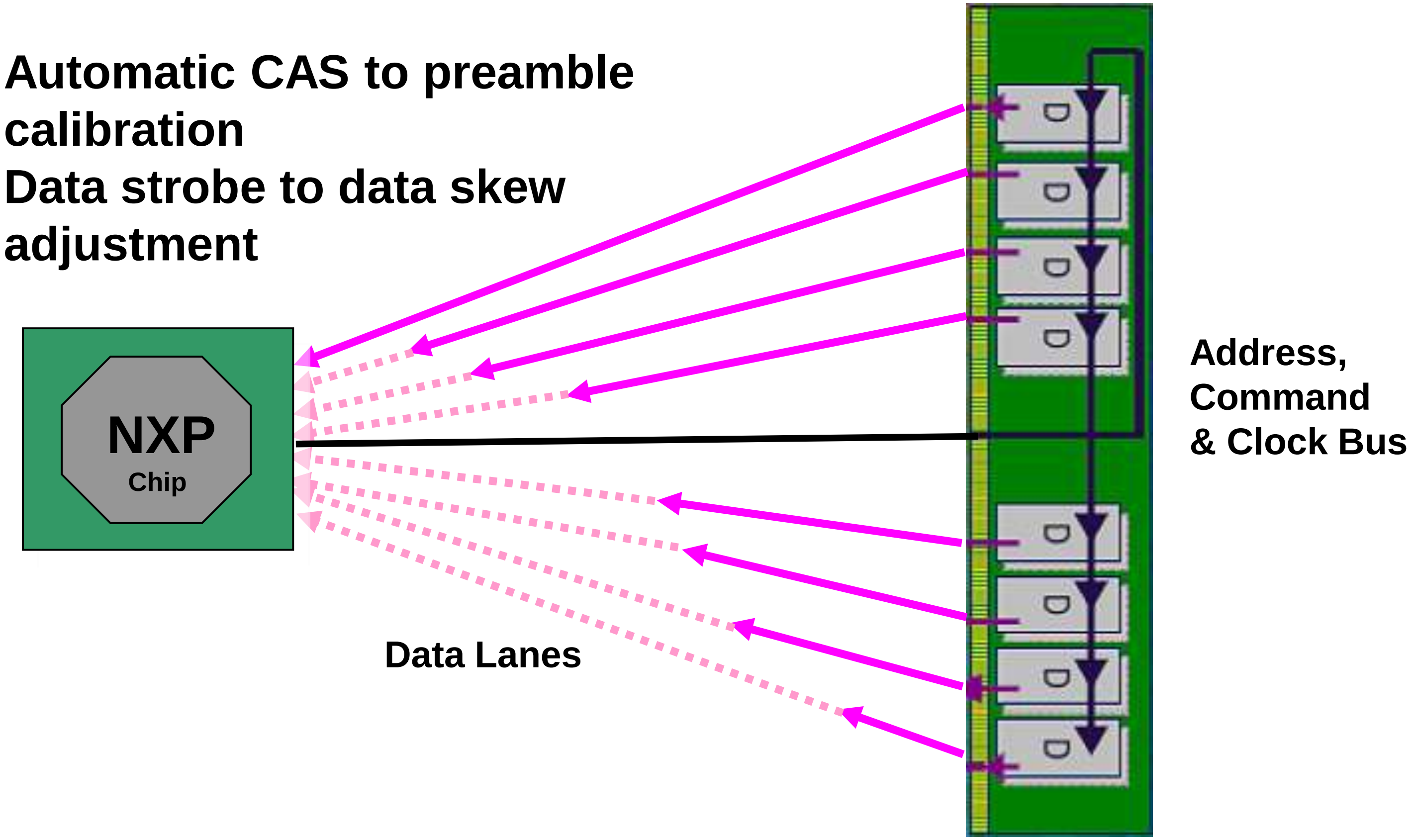
- **Write leveling used to add delay to each strobe/data line.**



Write leveling sequence during the initialization process will determine the appropriate delays to each data byte lane and add this delay for every write cycle.

Read Adjustment

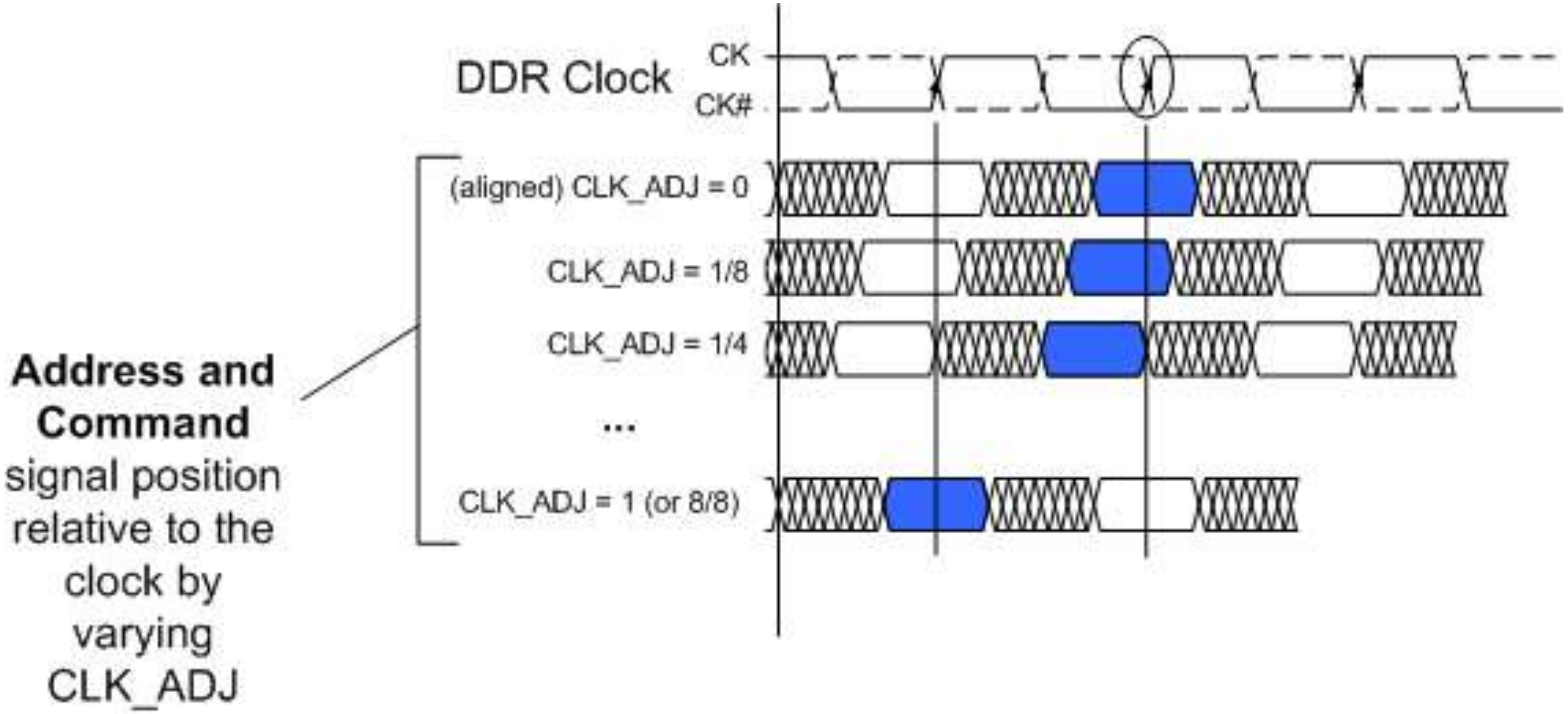
- **Automatic CAS to preamble calibration**
- **Data strobe to data skew adjustment**



Auto CPO will provide the expected arrival time of preamble for each strobe line of each byte lane during the read cycle to adjust for the delays cased by the fly-by topology.

CLK_ADJ - Clock Adjust

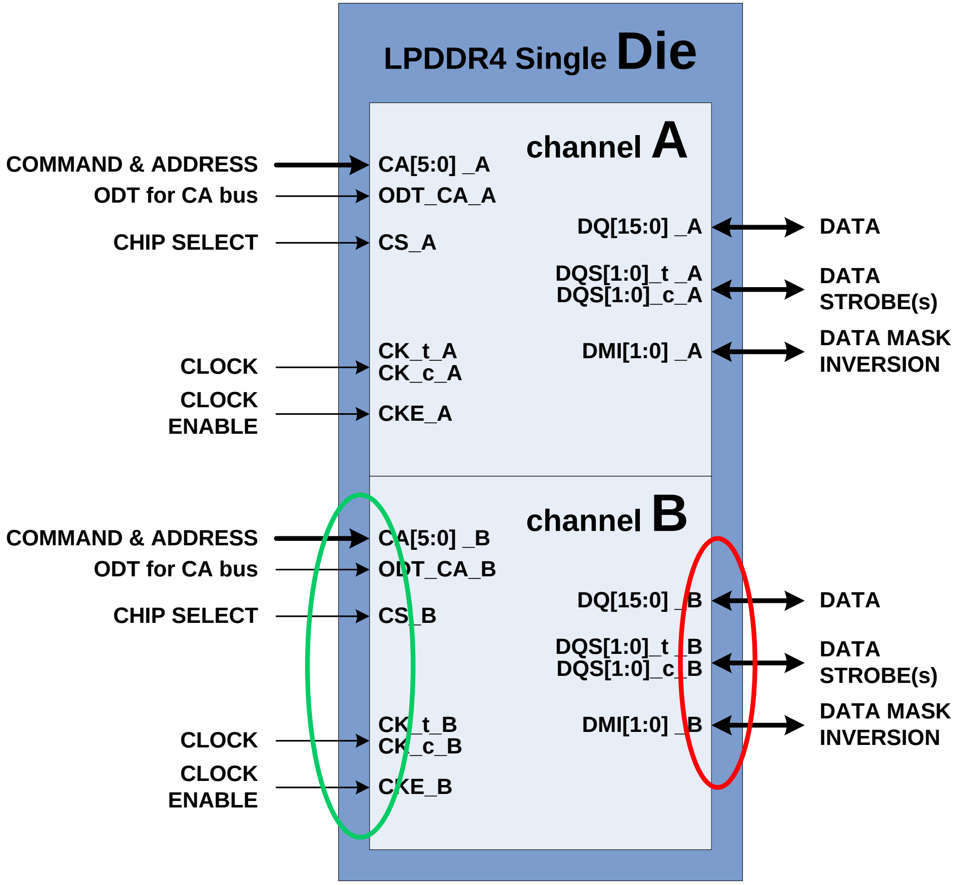
- CLK_ADJ defines the timing of the address and command signals relative to the DDR clock.



Example – generic LPDDR4 SDRAM

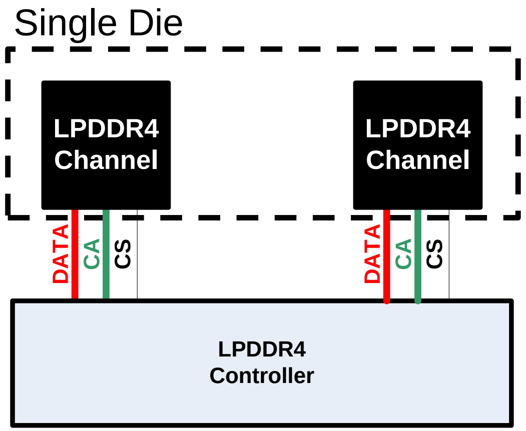
- 2 Channels per die
- 1, 2 or 4 Die per package
- 8 Banks per channel
- 2Gb to 16Gb Density range per channel

- DATA bus: DQ, DQS_t, DQS_c, DMI
- ADD bus: CA, OTD_CA, CS, CKE, CK_t, CK_c



ADD Bus DATA Bus

Example – LPDDR4 Arrangements – Single Die (2 Channels)



Multi-Channel

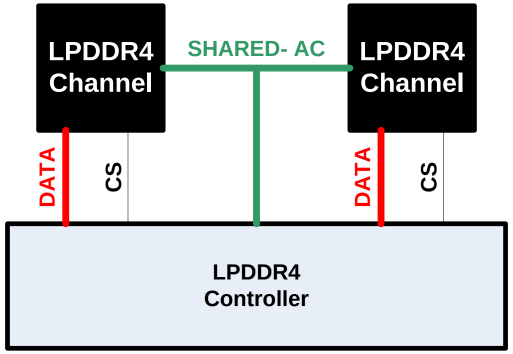
CA Pins: 12

DQ Pins: 32

CS Pins: 2

Banks: 16

Fetch (bytes): 32



Shared-AC

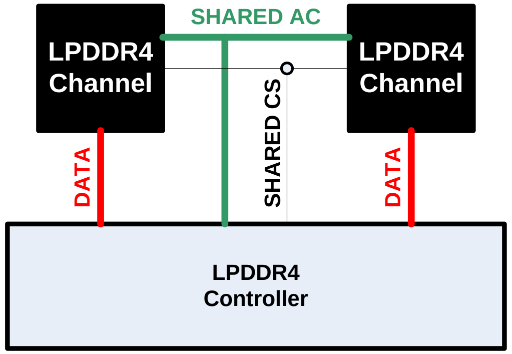
CA Pins: 6

DQ Pins: 32

CS Pins: 2

Banks: 16

Fetch(bytes): 32/64



DDR/Parallel

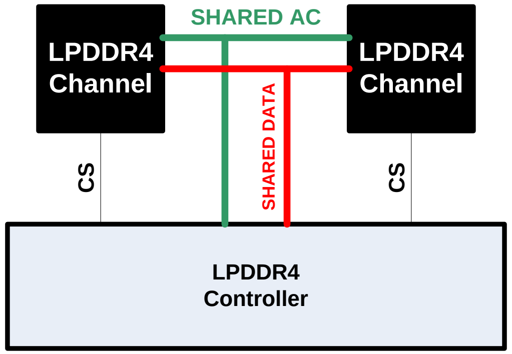
CA Pins: 6

DQ Pins: 32

CS Pins: 1

Banks: 8

Fetch(bytes): 64



Multi-Rank/Serial

CA Pins: 6

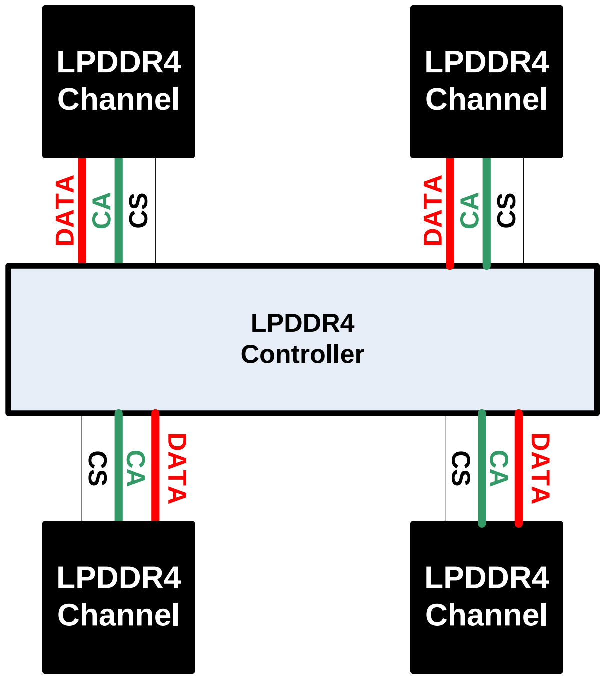
DQ Pins: 16

CS Pins: 2

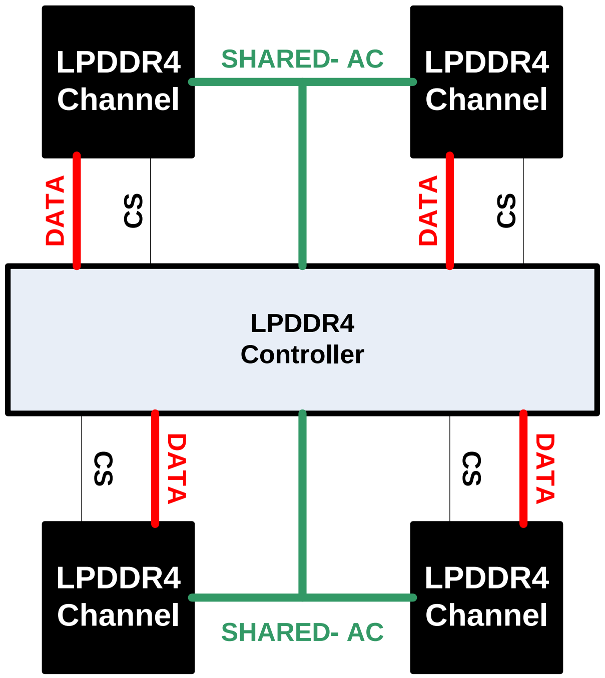
Banks: 8

Fetch(bytes): 32

Example – LPDDR4 Arrangements – 2 Die (4 Channels)

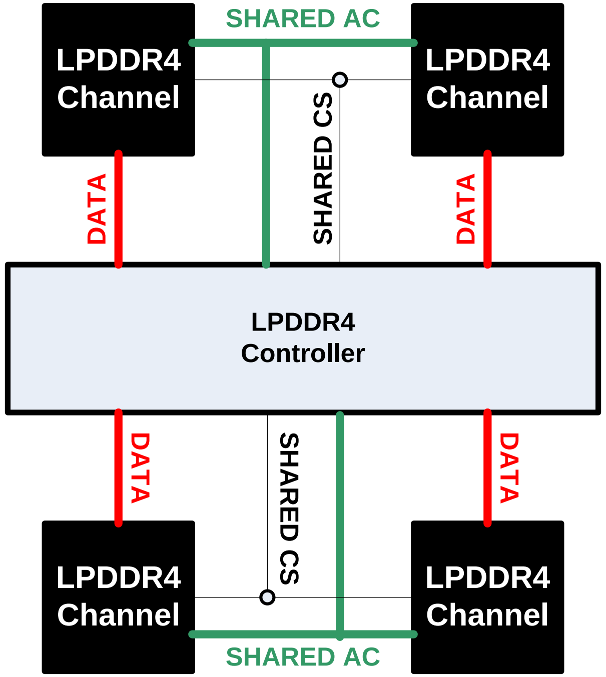


Multi-Channel

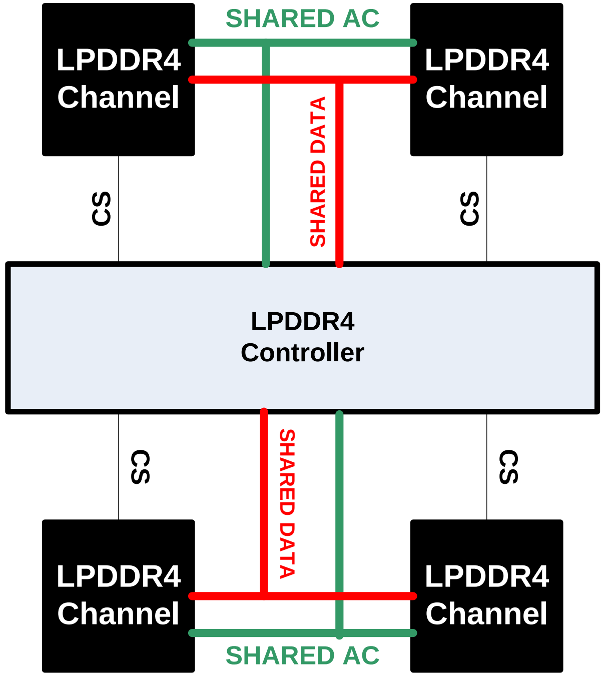


Shared-AC

Example – LPDDR4 Arrangements – 2 Die (4 Channels) cont'd

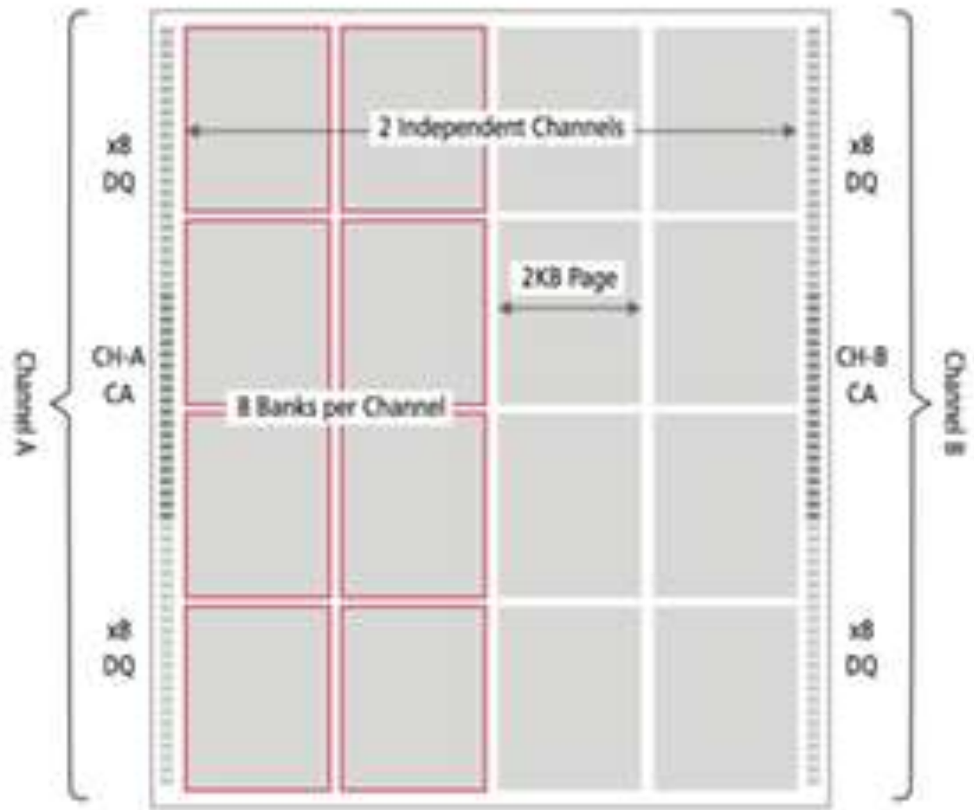


DDR/Parallel

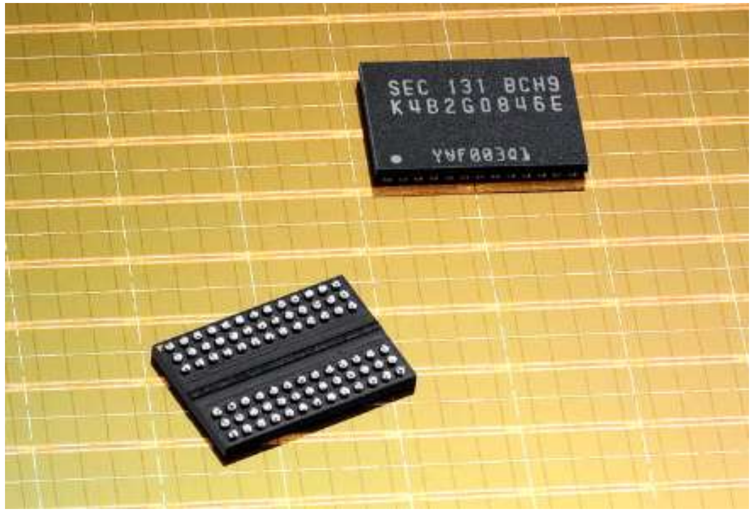


Multi-Rank/Serial

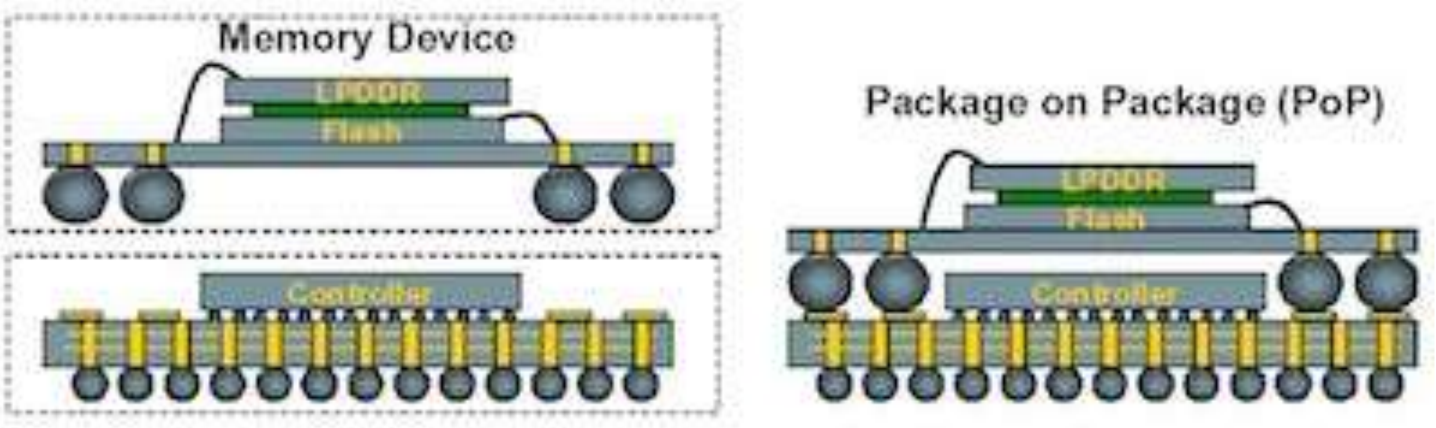
LPDDR4 Package



LPDDR4 Die Layout



Micron LPDDR4



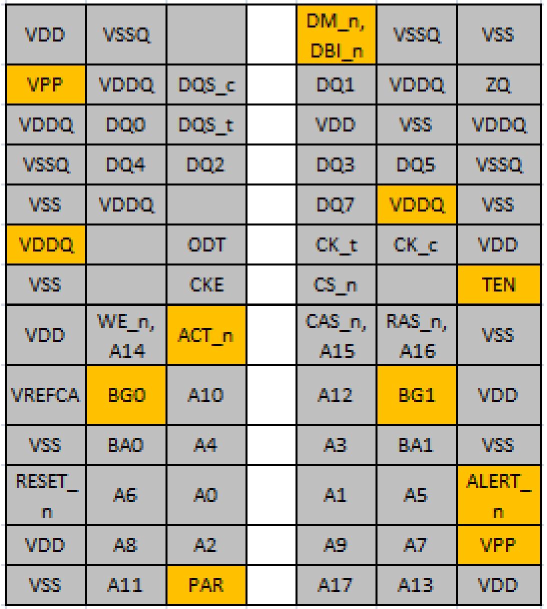
FLASH & LPDDR in PoP configuration

SDRAM Differences

DDR3 vs. DDR4 DRAM Pinouts

- DDR4 Pins Added
  - VDDQ (2) : 1.2V pins to DRAM
  - VPP (2): 2.5V external voltage source for DRAM internal word line driver
  - BG (2): Bank Group (2): pins to identify the bank groups
  - DBI_n: Data Bus Inversion
  - ACT_n: Active command
  - PAR: Parity error signal for address bus
  - ALERT_n: Both, Parity error on C\A and CRC error on data bus
  - TEN: Connectivity test mode

- DDR3 Pins Eliminated
  - VREFDQ
  - Bank Address (1): one less BA pin
  - VDD (1), VSS (3), VSSQ (1)



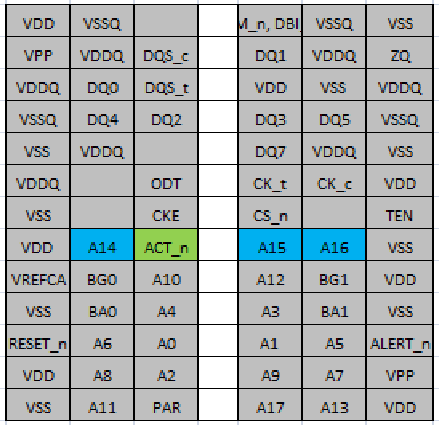
VDD	VSSQ			DM_n, DBI_n	VSSQ	VSS
VPP	VDDQ	DQS_c		DQ1	VDDQ	ZQ
VDDQ	DQ0	DQS_t		VDD	VSS	VDDQ
VSSQ	DQ4	DQ2		DQ3	DQ5	VSSQ
VSS	VDDQ			DQ7	VDDQ	VSS
VDDQ		ODT		CK_t	CK_c	VDD
VSS		CKE		CS_n		TEN
VDD	WE_n, A14	ACT_n		CAS_n, A15	RAS_n, A16	VSS
VREFCA	BG0	A10		A12	BG1	VDD
VSS	BA0	A4		A3	BA1	VSS
RESET_n	A6	A0		A1	A5	ALERT_n
VDD	A8	A2		A9	A7	VPP
VSS	A11	PAR		A17	A13	VDD

New Pin: ACT_n (Activate Command)

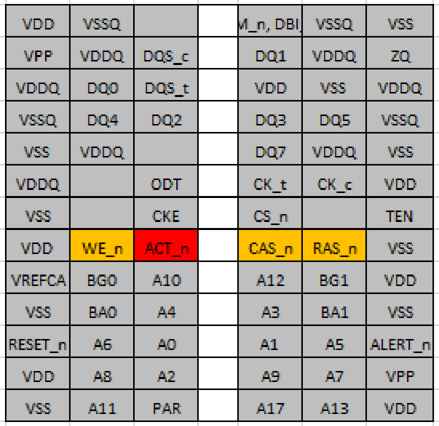
- ACT_n is a single pin for Active command input

- When ACT_n is low:
  - ACT Command is asserted
  - WE/CAS/RAS pins will be treated as address pins (A14:A16)

- When ACT_n is high
  - WE/CAS/RAS pins will be treated as command pins



VDD	VSSQ			M_n, DBI	VSSQ	VSS
VPP	VDDQ	DQS_c		DQ1	VDDQ	ZQ
VDDQ	DQ0	DQS_t		VDD	VSS	VDDQ
VSSQ	DQ4	DQ2		DQ3	DQ5	VSSQ
VSS	VDDQ			DQ7	VDDQ	VSS
VDDQ		ODT		CK_t	CK_c	VDD
VSS		CKE		CS_n		TEN
VDD	A14	ACT_n		A15	A16	VSS
VREFCA	BG0	A10		A12	BG1	VDD
VSS	BA0	A4		A3	BA1	VSS
RESET_n	A6	A0		A1	A5	ALERT_n
VDD	A8	A2		A9	A7	VPP
VSS	A11	PAR		A17	A13	VDD



VDD	VSSQ			M_n, DBI	VSSQ	VSS
VPP	VDDQ	DQS_c		DQ1	VDDQ	ZQ
VDDQ	DQ0	DQS_t		VDD	VSS	VDDQ
VSSQ	DQ4	DQ2		DQ3	DQ5	VSSQ
VSS	VDDQ			DQ7	VDDQ	VSS
VDDQ		ODT		CK_t	CK_c	VDD
VSS		CKE		CS_n		TEN
VDD	WE_n	ACT_n		CAS_n	RAS_n	VSS
VREFCA	BG0	A10		A12	BG1	VDD
VSS	BA0	A4		A3	BA1	VSS
RESET_n	A6	A0		A1	A5	ALERT_n
VDD	A8	A2		A9	A7	VPP
VSS	A11	PAR		A17	A13	VDD

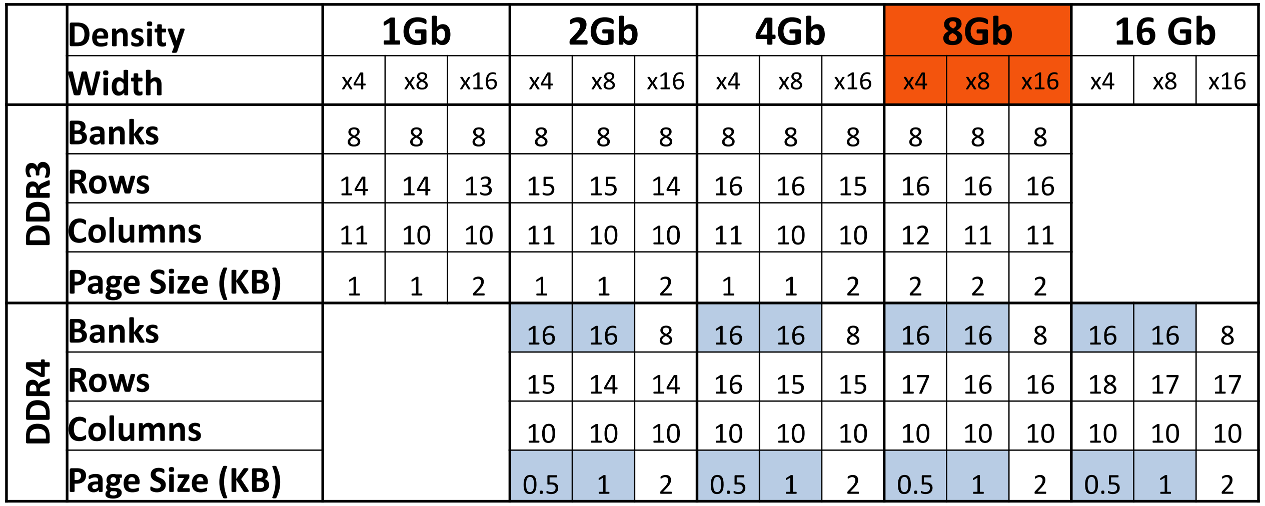
LPDDR3 vs. LPDDR4 DRAM Pinouts

- Change from a 10-bit DDR command/address bus to a 6-bit SDR bus
  - Address and Commands are encoded on CA lines
  - Addresses and Commands distributed over multiple cycle

- Change from one 32-bit wide bus to two independent 16-bit wide buses
  - x32, (2 channels, x16), x64 (4 channels, x16)

DRAM Densities DDR3 vs. DDR4

- 16 Banks for x4 and x8 DRAM DDR4, 8 Banks for x16
- 8Gb is DRAMs vendors choice for starting DDR4 density
- Larger memory size is one reason to use x4 vs. x8 vs. x16 DRAM
- Data mask or data bus inversion (DBI), not available in x4 DRAM



	Density	1Gb			2Gb			4Gb			8Gb			16 Gb		
	Width	x4	x8	x16	x4	x8	x16	x4	x8	x16	x4	x8	x16	x4	x8	x16
DDR3	Banks	8	8	8	8	8	8	8	8	8	8	8	8			
	Rows	14	14	13	15	15	14	16	16	15	16	16	16			
	Columns	11	10	10	11	10	10	11	10	10	12	11	11			
	Page Size (KB)	1	1	2	1	1	2	1	1	2	2	2	2			
DDR4	Banks				16	16	8	16	16	8	16	16	8	16	16	8
	Rows				15	14	14	16	15	15	17	16	16	18	17	17
	Columns				10	10	10	10	10	10	10	10	10	10	10	10
	Page Size (KB)				0.5	1	2	0.5	1	2	0.5	1	2	0.5	1	2

DRAM Densities progression in LPDDR

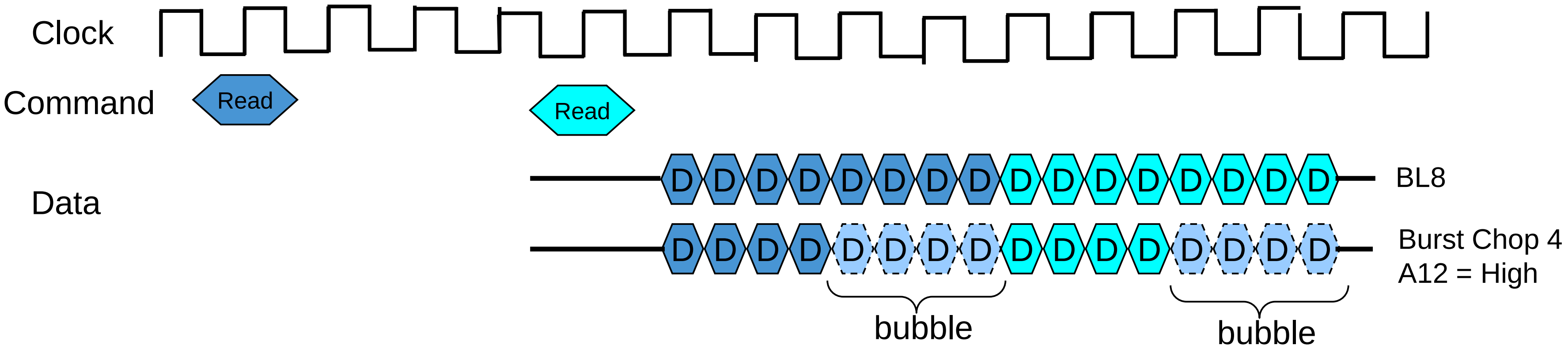
- LPDDR
  - 512Mb to 8Gb
- LPDDR2
  - 512Mb to 16Gb
- LPDDR3
  - 4Gb to 32Gb
- **LPDDR4**
  - 4Gb to 32Gb (2Gb to 16Gb per channel)

Modules DDR3 vs. DDR4

- U/RDIMM Pin count of 240 vs. 288, pin pitch of 1.0mm vs. 0.85mm
- Bottom edge flat vs. step ramp. Height & width increased by ~1mm
- DRAM ball count and ball pitch not changed
- DIMM topology of fly by for address/command bus not changed
- SoDIMM pin count of 204 vs. 260
- SoDIMM will have native ECC support vs. non-compatible pin out in DDR3

Burst Length

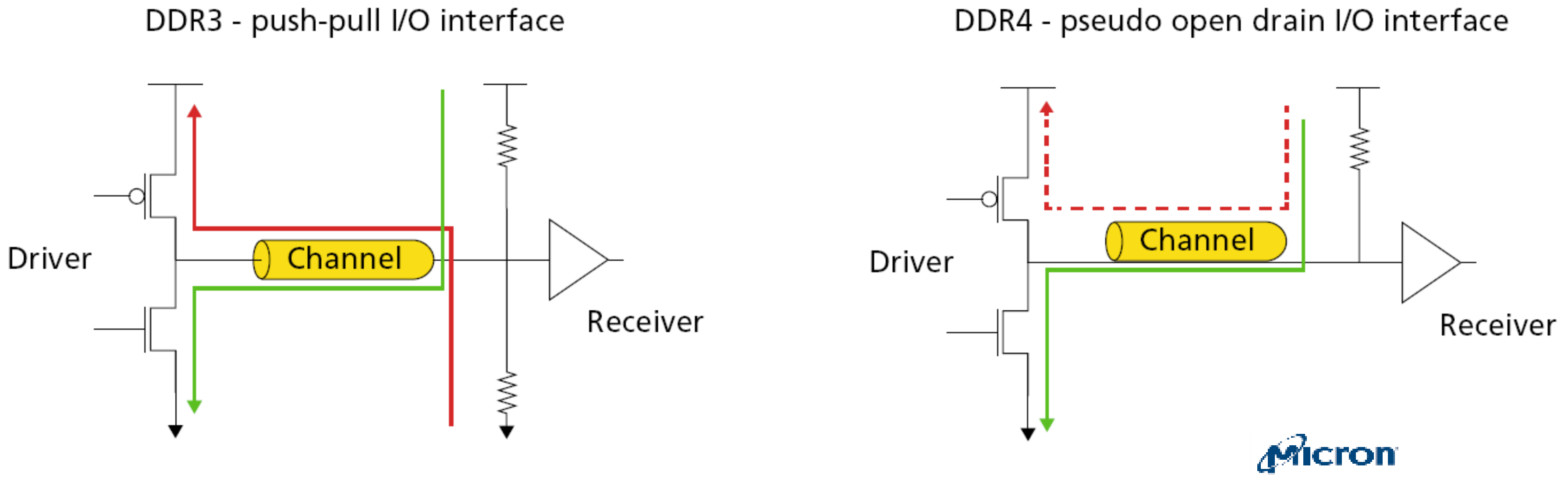
- DDR4 normally has Burst Length of 8 (BL8)
- DDR4 also support pseudo Burst Length of 4 (BC4, OTF BC)


Clock
Command
Read
Read
Data
D D D D D D D D D D D D D D D D
BL8
D D D D D D D D D D D D D D D D
Burst Chop 4
A12 = High
bubble
bubble

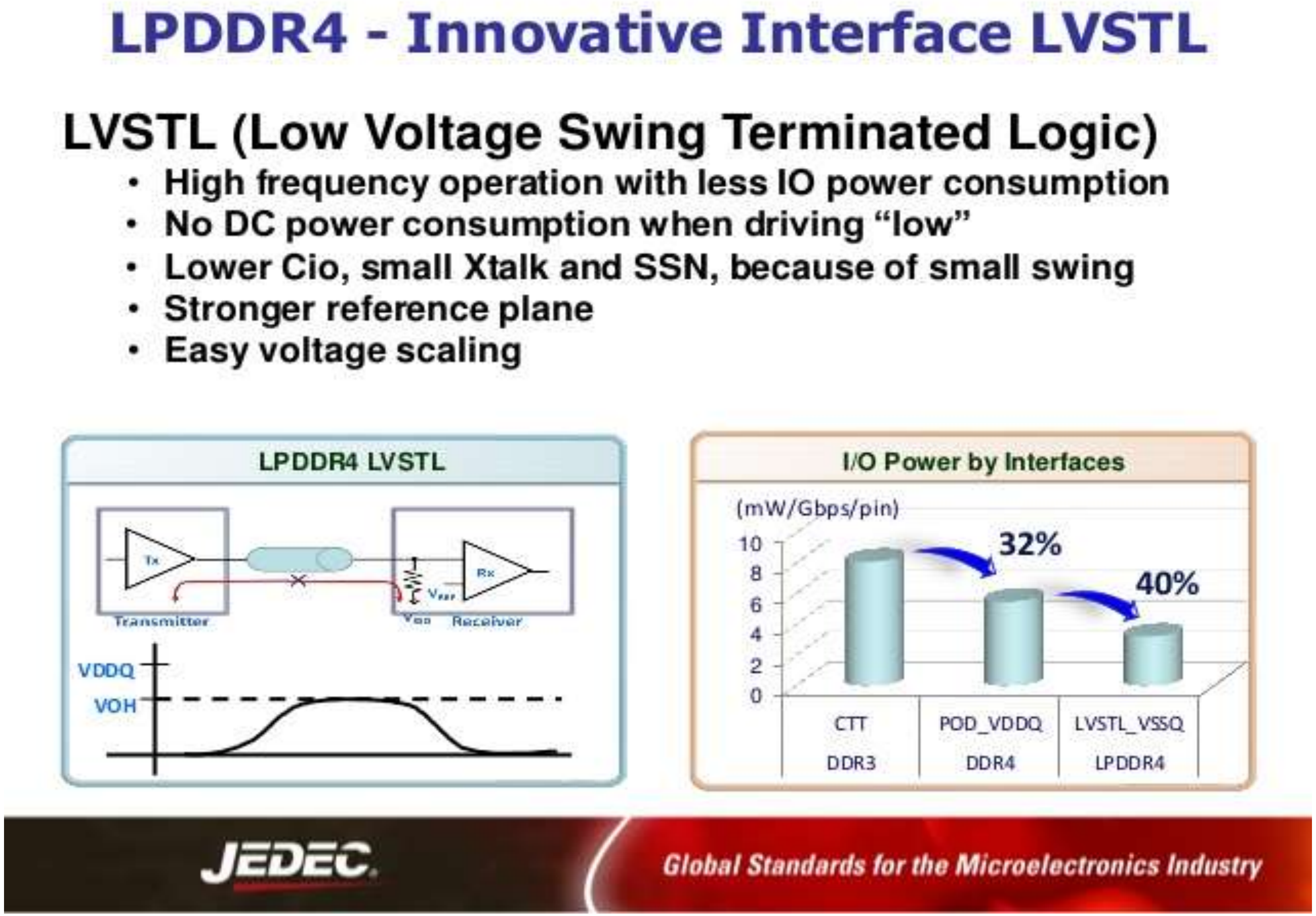
- LPDDR3 Burst Length of 8 (BL8)
- LPDDR4 Burst Length of 16 (BL16)

DDR4 Output Driver/Termination

- Center tap termination is used in DDR3 receiver
- POD termination or pull up is used in DDR4 receiver
- Push-Pull driver in DDR3 and POD driver in DDR4
- Less power is consumed using POD driver & termination


DDR3 - push-pull I/O interface
Driver
Channel
Receiver
DDR4 - pseudo open drain I/O interface
Driver
Channel
Receiver
Micron

LPDDR4 Output Driver/Termination



LPDDR4 - Innovative Interface LVSTL

LVSTL (Low Voltage Swing Terminated Logic)

- High frequency operation with less IO power consumption
- No DC power consumption when driving “low”
- Lower Cio, small Xtalk and SSN, because of small swing
- Stronger reference plane
- Easy voltage scaling

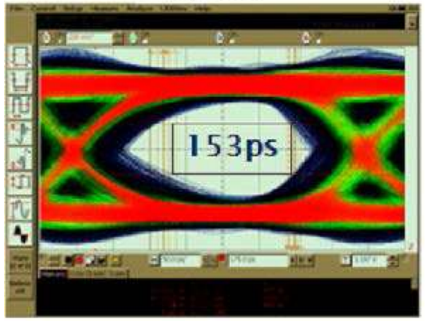
JEDEC. Global Standards for the Microelectronics Industry

New Pin: DBI_n (Data Bus Inversion)

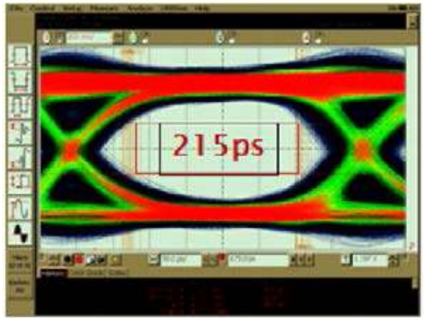
- Active low input/output for data bus inversion mode
- Available only on x8 and x16 DRAM
- **PROs:** Less noise, better data eye and lower power consumption
- **CONs:** Performance is affected due to data mask not being available and CAS_LAT is increased by 2 clocks.

	Controller				Interconnect				DRAM			
DQ0	0	1	0	0	1	1	0	1	0	1	0	0
DQ1	1	1	0	0	0	1	0	1	1	1	0	0
DQ2	0	0	0	0	1	0	0	1	0	0	0	0
DQ3	0	1	1	0	1	1	1	1	0	1	1	0
DQ4	0	1	0	0	1	1	0	1	0	1	0	0
DQ5	1	0	1	0	0	0	1	1	1	0	1	0
DQ6	1	1	1	0	0	1	1	1	1	1	1	0
DQ7	0	0	1	0	1	0	1	1	0	0	1	0
DBI_n					0	1	1	0				
# low bits	5	3	4	8	4	3	4	1				

No DBI



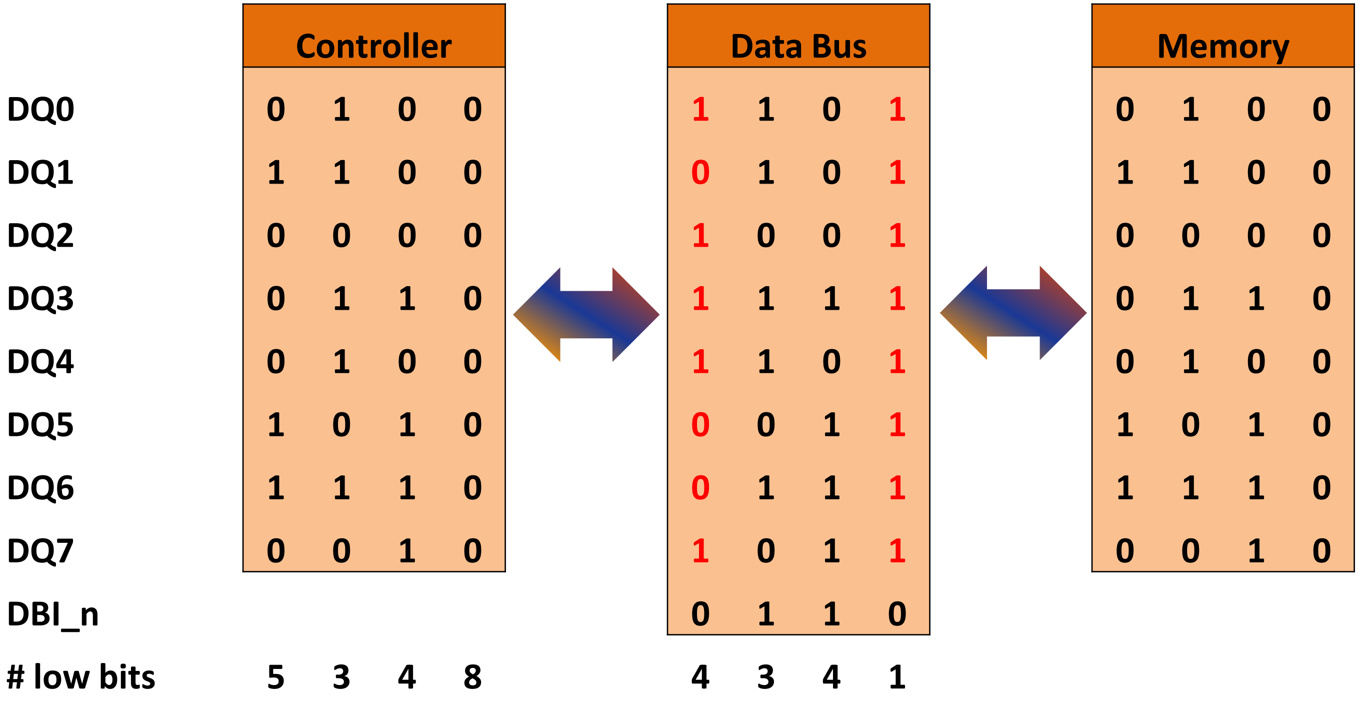
Minimum Zeros DBI



Micron

Data Bus Inversion - DBI

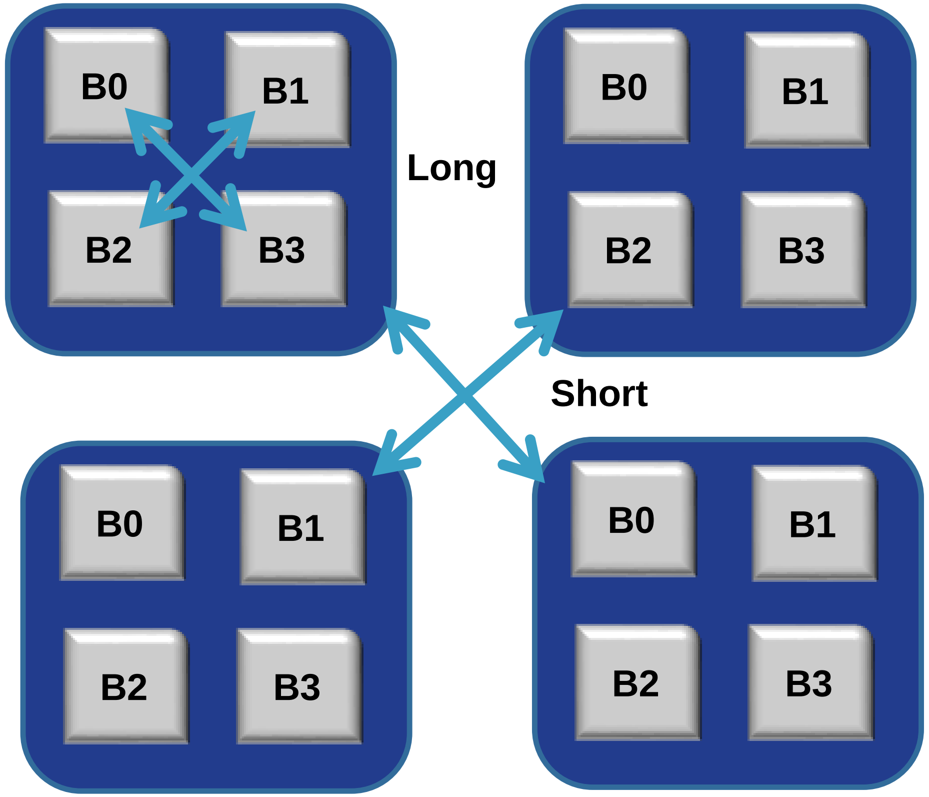
- If more than 4 bits of a byte lane are low, invert the data and drive the DBI_n pin low
- If 4 or less bits of a byte lane are low, do not invert the data and drive the DBI_n pin high



	Controller				Data Bus				Memory			
DQ0	0	1	0	0	1	1	0	1	0	1	0	0
DQ1	1	1	0	0	0	1	0	1	1	1	0	0
DQ2	0	0	0	0	1	0	0	1	0	0	0	0
DQ3	0	1	1	0	1	1	1	1	0	1	1	0
DQ4	0	1	0	0	1	1	0	1	0	1	0	0
DQ5	1	0	1	0	0	0	1	1	1	0	1	0
DQ6	1	1	1	0	0	1	1	1	1	1	1	0
DQ7	0	0	1	0	1	0	1	1	0	0	1	0
DBI_n					0	1	1	0				
# low bits	5	3	4	8	4	3	4	1				

New Pin: BGn (Bank Group Address)

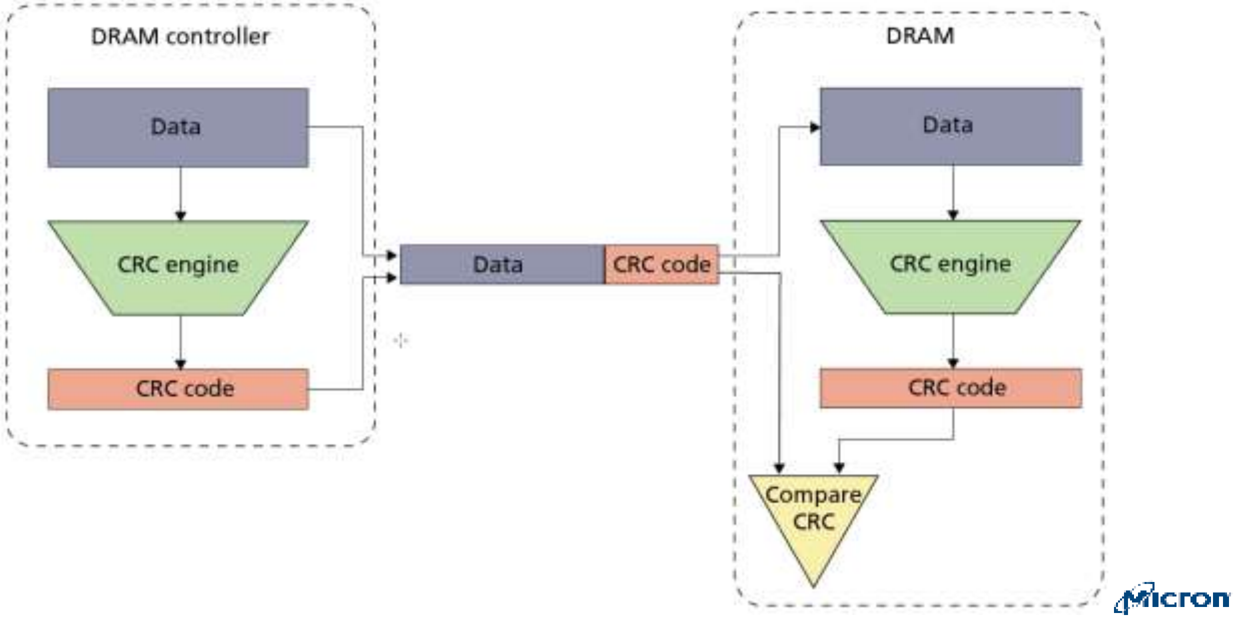
- Different timing within a group and between groups
  - Active to active (tRRD_L)
  - Write to read (tWTR_L)
  - Read to read (tCCD_L)
  - Write to write (tCCD_L)
- Controller to maintain timing requirements for both within a group (long) and between groups (short)



Data rate	1600	1866	2133	2400
tCCD_S	4	4	4	4
tCCD_L	5	5	6	6

New Pin: ALERT_n & Cyclic Redundancy Check (CRC)

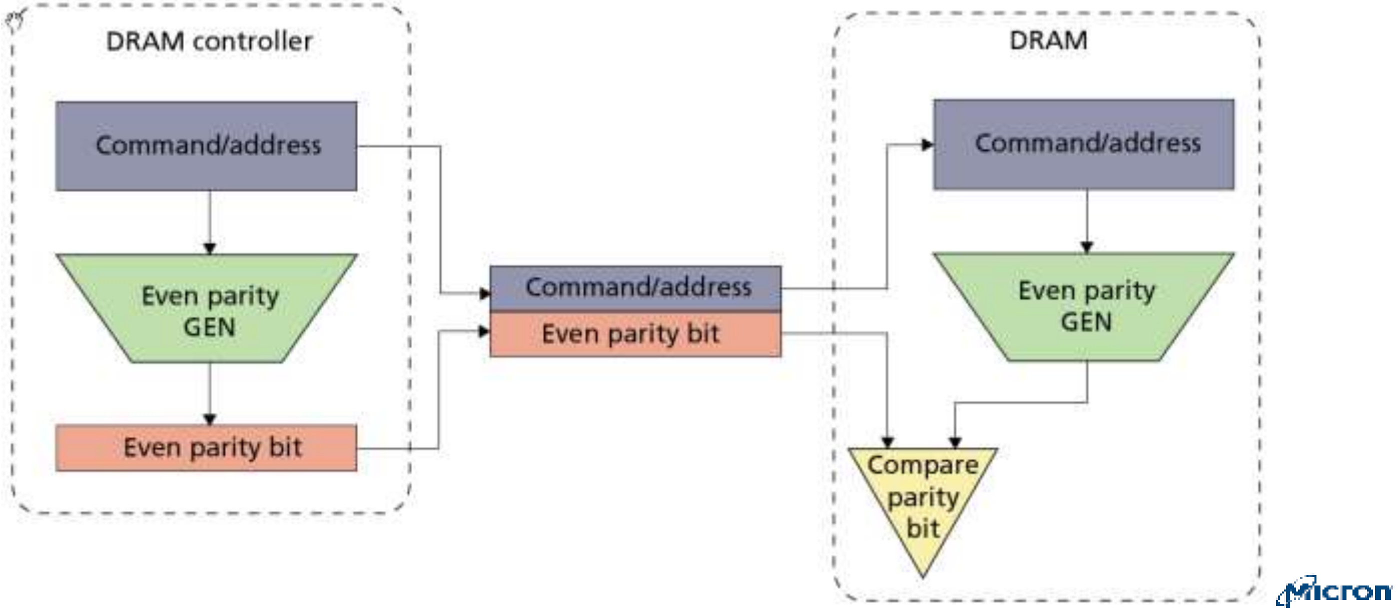
- Alert_n – Active low output signal that indicates an error event for reporting C/A parity and data write CRC errors
- Polynomial encoding is used to generate the CRC, 8-bits per write burst
- **PROs:** Better reliability by detecting data errors during write cycles
- **CONs:** Two beats added to the write burst to transfer the CRC header



*Not the same as ECC

New Pin: PAR (Address Bus Parity)

- C/A Parity signal (PAR) covers ACT_n, RAS_n, CAS_n, WE_n and the address bus. Control signals CKE, ODT, CS_n are not included
- Commands must be qualified by CS_n
- Alert_n used to flag error to memory controller
- **PROs:** Better reliability
- **CONs:** PL (4clk for 2133, 5clk for 2400) is added to read latency



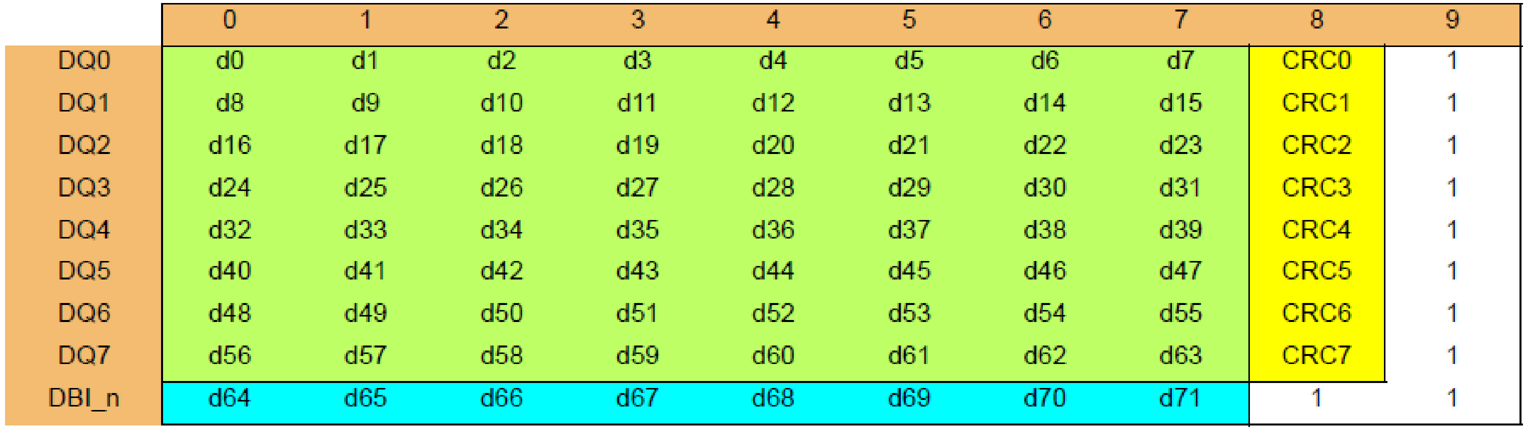
Cyclic Redundancy Check (CRC)

- Alert_n – Active low output signal that indicates an error event for both the C/A Parity Mode and the CRC Data Mode
- CRC Data mode:
  - To detect data errors during write cycles .
  - Polynomial encoding is used to generate the CRC for every 8-bit
  - Two beats added to the write burst to transfer the CRC header
  - DRAM generates a CRC checksum per each write burst and DQS lane
  - DRM compares the generated checksum to controllers checksum
  - If Data Mask is disabled, corrupt data is written, with the Alert_n flag sent to controller to retry the write.
  - If Data Mask is enabled, corrupt data is not written, with the Alert_n flag sent to controller to retry the write.

Data Write CRC

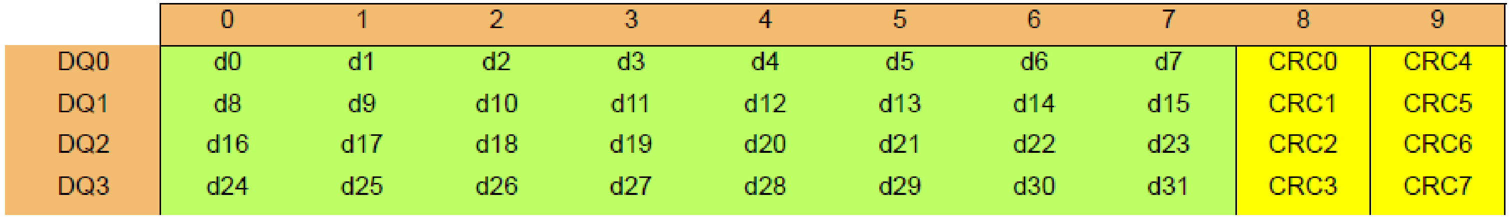
- Example data mapping with CRC for 8-bit, 4-bit and 16-bit devices
- Note: not the same as ECC

The following figure shows detailed bit mapping for a x8 device.



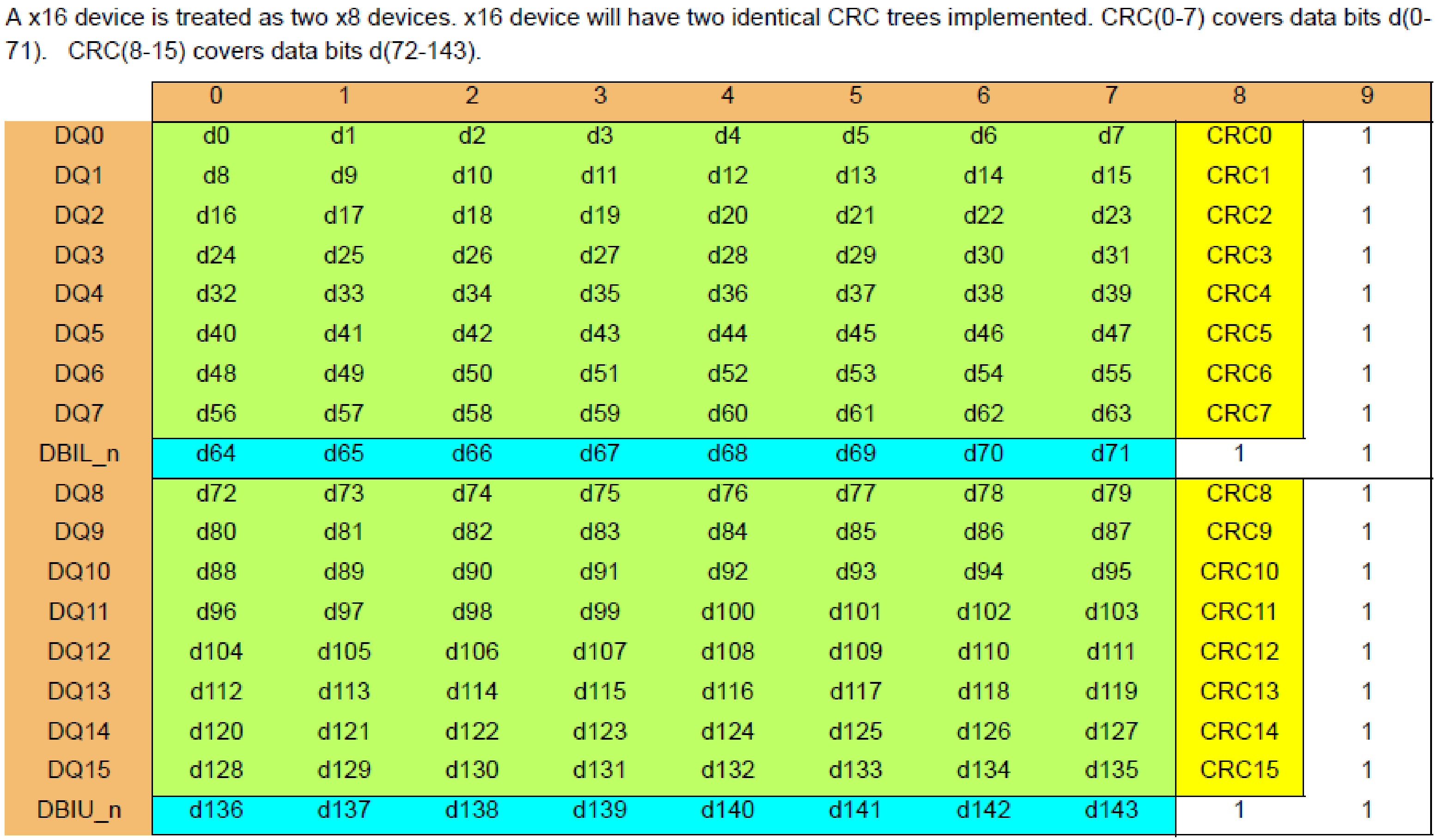
	0	1	2	3	4	5	6	7	8	9
DQ0	d0	d1	d2	d3	d4	d5	d6	d7	CRC0	1
DQ1	d8	d9	d10	d11	d12	d13	d14	d15	CRC1	1
DQ2	d16	d17	d18	d19	d20	d21	d22	d23	CRC2	1
DQ3	d24	d25	d26	d27	d28	d29	d30	d31	CRC3	1
DQ4	d32	d33	d34	d35	d36	d37	d38	d39	CRC4	1
DQ5	d40	d41	d42	d43	d44	d45	d46	d47	CRC5	1
DQ6	d48	d49	d50	d51	d52	d53	d54	d55	CRC6	1
DQ7	d56	d57	d58	d59	d60	d61	d62	d63	CRC7	1
DBI_n	d64	d65	d66	d67	d68	d69	d70	d71	1	1

The following figure shows detailed bit mapping for a x4 device.



	0	1	2	3	4	5	6	7	8	9
DQ0	d0	d1	d2	d3	d4	d5	d6	d7	CRC0	CRC4
DQ1	d8	d9	d10	d11	d12	d13	d14	d15	CRC1	CRC5
DQ2	d16	d17	d18	d19	d20	d21	d22	d23	CRC2	CRC6
DQ3	d24	d25	d26	d27	d28	d29	d30	d31	CRC3	CRC7

Data Write CRC (continued)

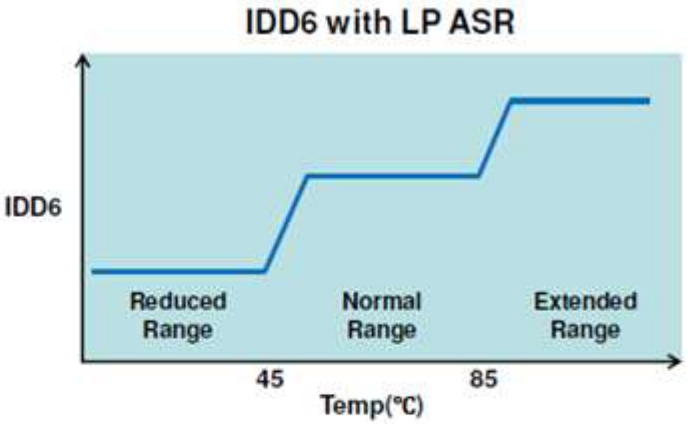


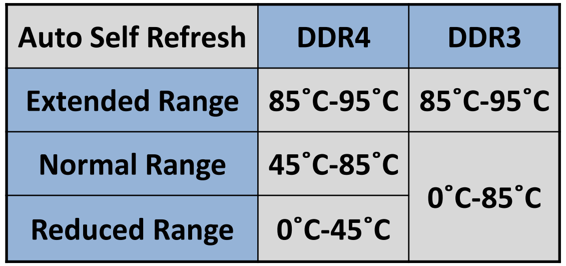
A x16 device is treated as two x8 devices. x16 device will have two identical CRC trees implemented. CRC(0-7) covers data bits d(0-71). CRC(8-15) covers data bits d(72-143).

	0	1	2	3	4	5	6	7	8	9
DQ0	d0	d1	d2	d3	d4	d5	d6	d7	CRC0	1
DQ1	d8	d9	d10	d11	d12	d13	d14	d15	CRC1	1
DQ2	d16	d17	d18	d19	d20	d21	d22	d23	CRC2	1
DQ3	d24	d25	d26	d27	d28	d29	d30	d31	CRC3	1
DQ4	d32	d33	d34	d35	d36	d37	d38	d39	CRC4	1
DQ5	d40	d41	d42	d43	d44	d45	d46	d47	CRC5	1
DQ6	d48	d49	d50	d51	d52	d53	d54	d55	CRC6	1
DQ7	d56	d57	d58	d59	d60	d61	d62	d63	CRC7	1
DBIL_n	d64	d65	d66	d67	d68	d69	d70	d71	1	1
DQ8	d72	d73	d74	d75	d76	d77	d78	d79	CRC8	1
DQ9	d80	d81	d82	d83	d84	d85	d86	d87	CRC9	1
DQ10	d88	d89	d90	d91	d92	d93	d94	d95	CRC10	1
DQ11	d96	d97	d98	d99	d100	d101	d102	d103	CRC11	1
DQ12	d104	d105	d106	d107	d108	d109	d110	d111	CRC12	1
DQ13	d112	d113	d114	d115	d116	d117	d118	d119	CRC13	1
DQ14	d120	d121	d122	d123	d124	d125	d126	d127	CRC14	1
DQ15	d128	d129	d130	d131	d132	d133	d134	d135	CRC15	1
DBIU_n	d136	d137	d138	d139	d140	d141	d142	d143	1	1

Low-Power Auto Self Refresh

- DDR4
  - While DRAM is in self-refresh mode, four refresh mode options available:
    - Manual mode, normal temperature (45C – 85C)
    - Manual mode, extended temperature (85C – 95C)
    - Manual mode, reduced temperature (0C – 45C)
    - Automatic mode: automatically switches between modes based on temperature sensor measurements
  - Power savings by reducing refresh rate when possible



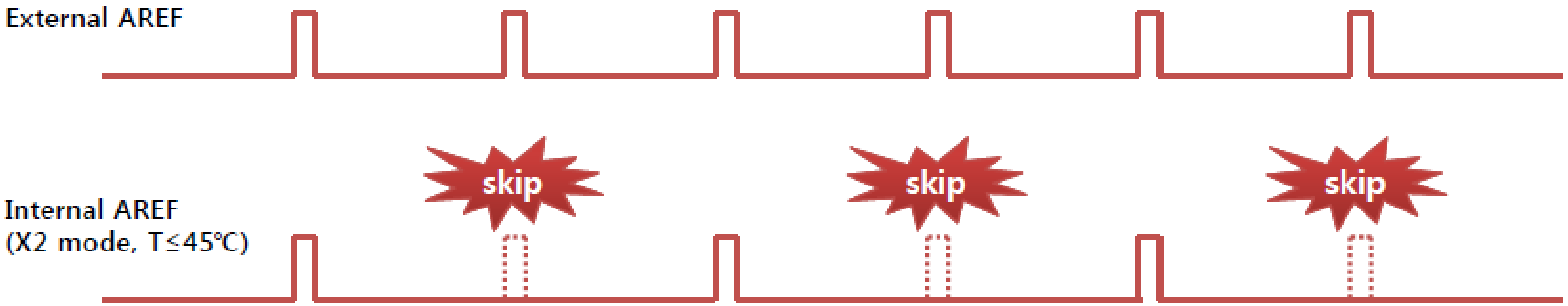


Auto Self Refresh	DDR4	DDR3
Extended Range	85˚C-95˚C	85˚C-95˚C
Normal Range	45˚C-85˚C	0˚C-85˚C
Reduced Range	0˚C-45˚C	

- LPDDR4
  - Temperature-compensated self refresh (TCSR) mode
  - Partial-array self refresh (PASR)

Temperature Controlled Auto Refresh

- Enabled or disabled in MR4
- In extended temp mode controller sends refresh commands every 3.9us
- DRAM based on the internal temp sensor will skip refresh commands automatically to save power



Command Address Latency (CAL)

- DDR4 supports CAL as a power savings feature
- In default mode, DRAM C/A input receivers are always on
- In CAL mode, only CS receiver is always on. All remaining C/A input receivers are kept in a low power state when not in use. CS signal is sent N number of cycles earlier to allow DRAM time to wake up C/A input receivers

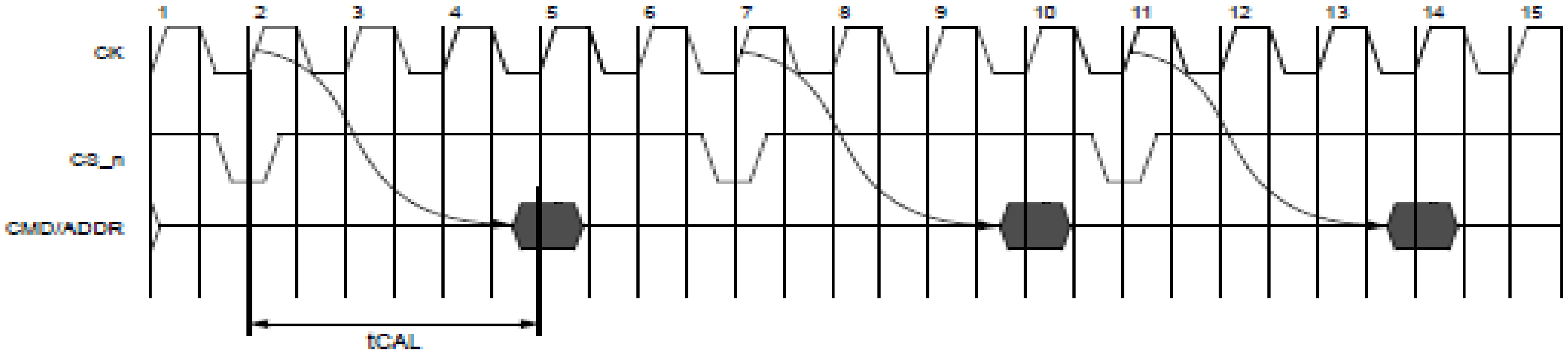


Figure 34 — Definition of CAL

Why DDR4 Over DDR3?

- Save on power and cost
  - DDR4 can reduce power by up to 37%
  - DDR4 price cross-over at the end of 2015

- Higher Data rates
  - DDR4 offers double the data rate
  - DDR4 doubles the number of internal banks, increased bandwidth
  - New options to increase performance

- Better reliability & manufacturing capabilities
  - Connectivity test
  - Data bus inversion (DBI)
  - Internal VREF calibration
  - CRC on DATA & parity on ADDRESS bus

- Larger densities

Why LPDDR4 Over PLDDR3?

- Save power
  - LPDDR4 can reduce overall power by up to 37% over LPDDR3
  - Many power savings mechanisms:
    - Temperature-compensated self refresh (TCSR), Partial-array self refresh (PASR), Deep power-down (DPD), Programmable drive strength (DS), Programmable VOH signal level

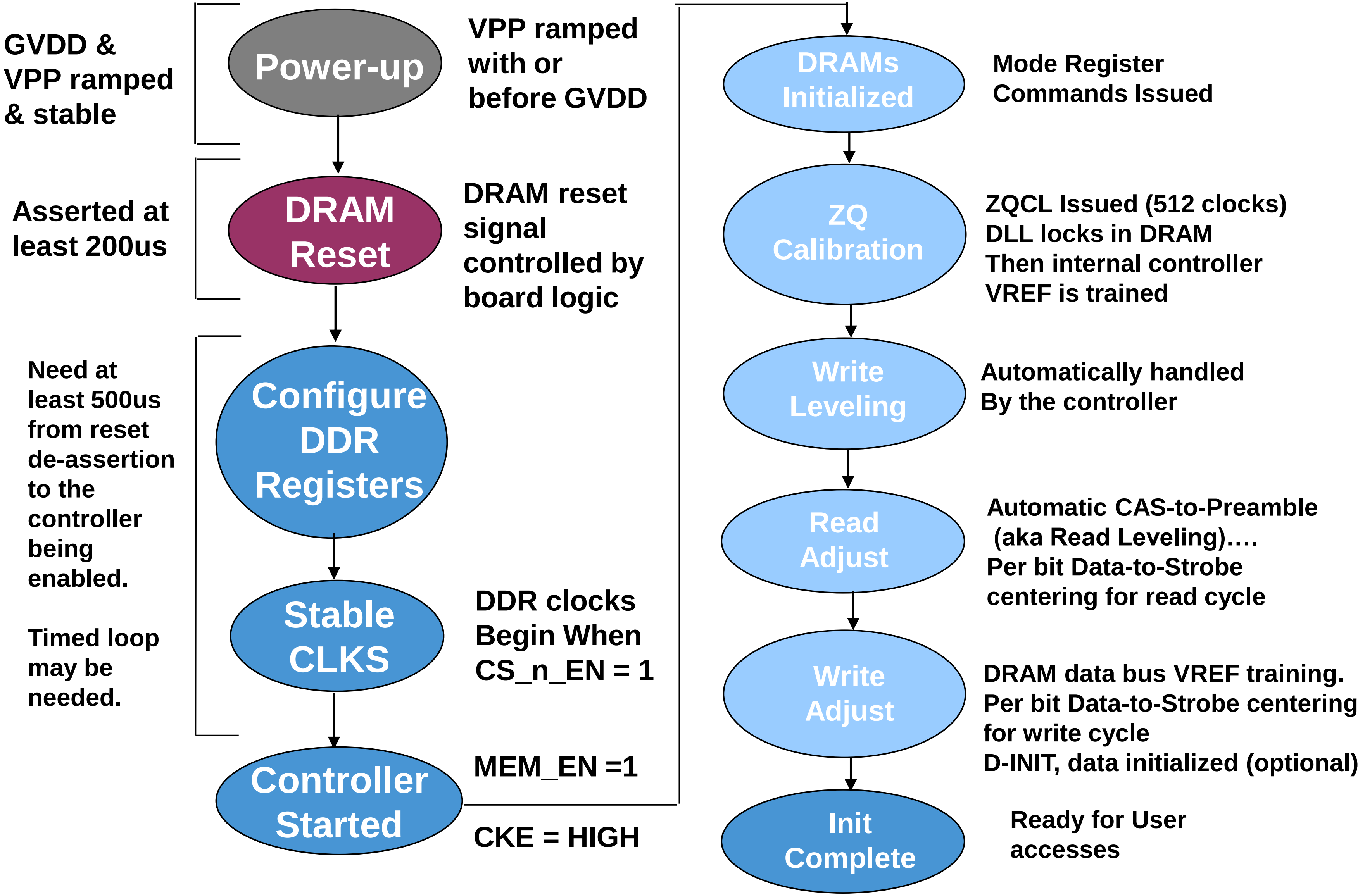
- Higher Data rates
  - Faster Clock rates
  - Two-channel architecture
    - Increased speed due dual channel die layout
    - Shorter signal distances

- Flexible memory configurations
  - Better suite the application

- Larger densities

General Hardware and Software Design Guidelines and Tips

DDR4 Initialization Flow



GVDD & VPP ramped & stable

Power-up

VPP ramped with or before GVDD

Asserted at least 200us

DRAM Reset

DRAM reset signal controlled by board logic

Need at least 500us from reset de-assertion to the controller being enabled.

Timed loop may be needed.

Configure DDR Registers

Stable CLKS

DDR clocks Begin When CS_n_EN = 1

Controller Started

MEM_EN =1

CKE = HIGH

DRAMs Initialized

Mode Register Commands Issued

ZQ Calibration

ZQCL Issued (512 clocks)
DLL locks in DRAM
Then internal controller VREF is trained

Write Leveling

Automatically handled By the controller

Read Adjust

Automatic CAS-to-Preamble (aka Read Leveling)….
Per bit Data-to-Strobe centering for read cycle

Write Adjust

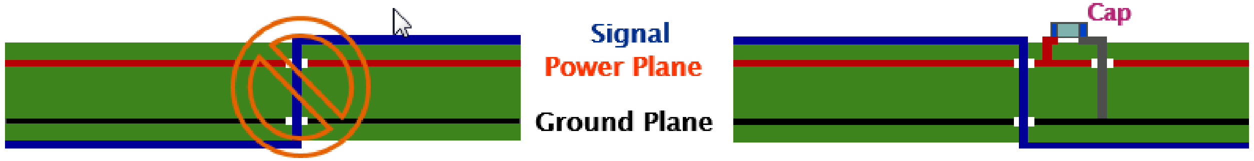
DRAM data bus VREF training.
Per bit Data-to-Strobe centering for write cycle
D-INIT, data initialized (optional)

Init Complete

Ready for User accesses

General Hardware Guidelines

- Examine the DDR4 Layout Guidelines for QorIQ devices App. Note (AN5097)
- Run pre and post board simulation
  - IBIS models are available for both controller and DRAM
- Employ industry standard practices
- Minimize Crosstalk, ISI, Vref noise, Impedance mismatches
- Eliminate return path discontinuities (RPD)
- Minimize the simultaneous switching output (SSO) effects
  - Proper distribution of power and ground planes
  - Proper capacitance decoupling
- Examine the reference design boards with DDR4 implemented
  - Both discrete and DIMM DDR4 are available for QorIQ devices



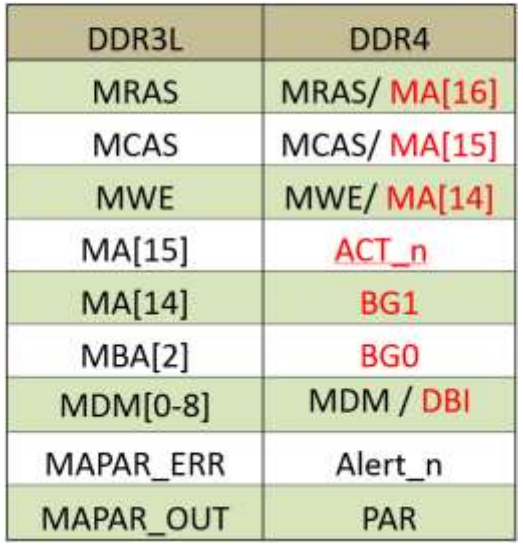
Important HW Considerations for DDR4 Transition

- VPP supply
  - VPP = 2.5V required for each DRAM
  - Follow DRAM vendor specification for power/current requirements
  - VPP ramped with or before GVDD

- VrefDQ reference input is removed

- New signals added to each DRAM
  - ACT_n
  - DBI
  - PAR
  - TEN (Pull to GND when not used)
  - ALERT

QorIQ with DDR3L\DDR4 memory controller

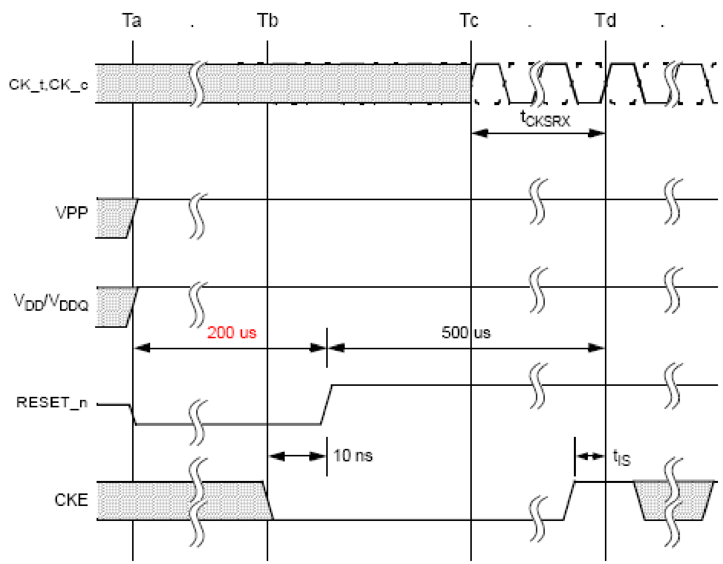


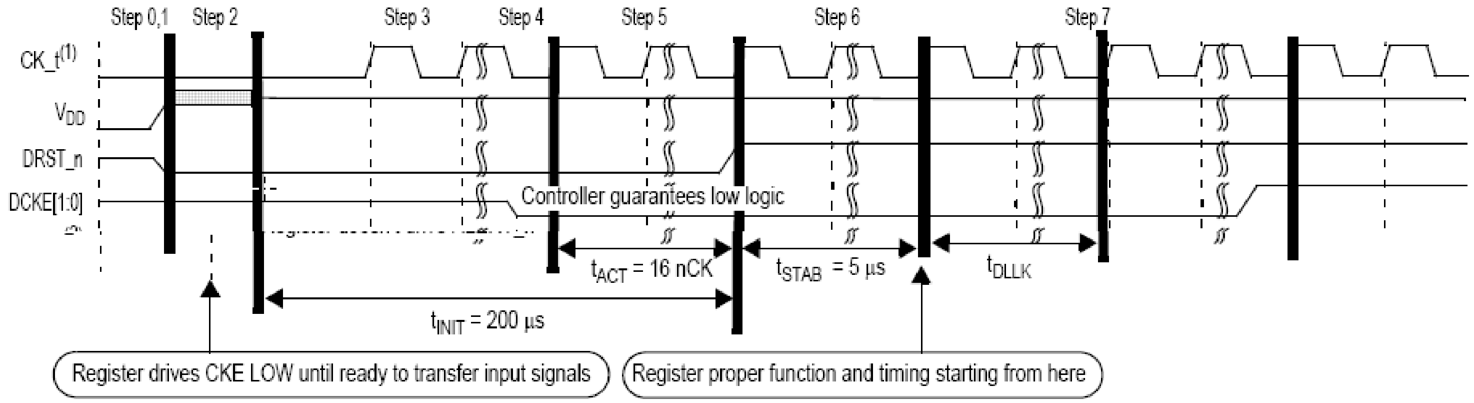
DDR3L	DDR4
MRAS	MRAS/ MA[16]
MCAS	MCAS/ MA[15]
MWE	MWE/ MA[14]
MA[15]	ACT_n
MA[14]	BG1
MBA[2]	BG0
MDM[0-8]	MDM / DBI
MAPAR_ERR	Alert_n
MAPAR_OUT	PAR

UDIMM vs. RDIMM DDR4 Reset

- UDIMM requires CKE to be low before RESET is de-asserted.

- RDIMM requires CKE to be low **and** clock to be present before RESET is de-asserted



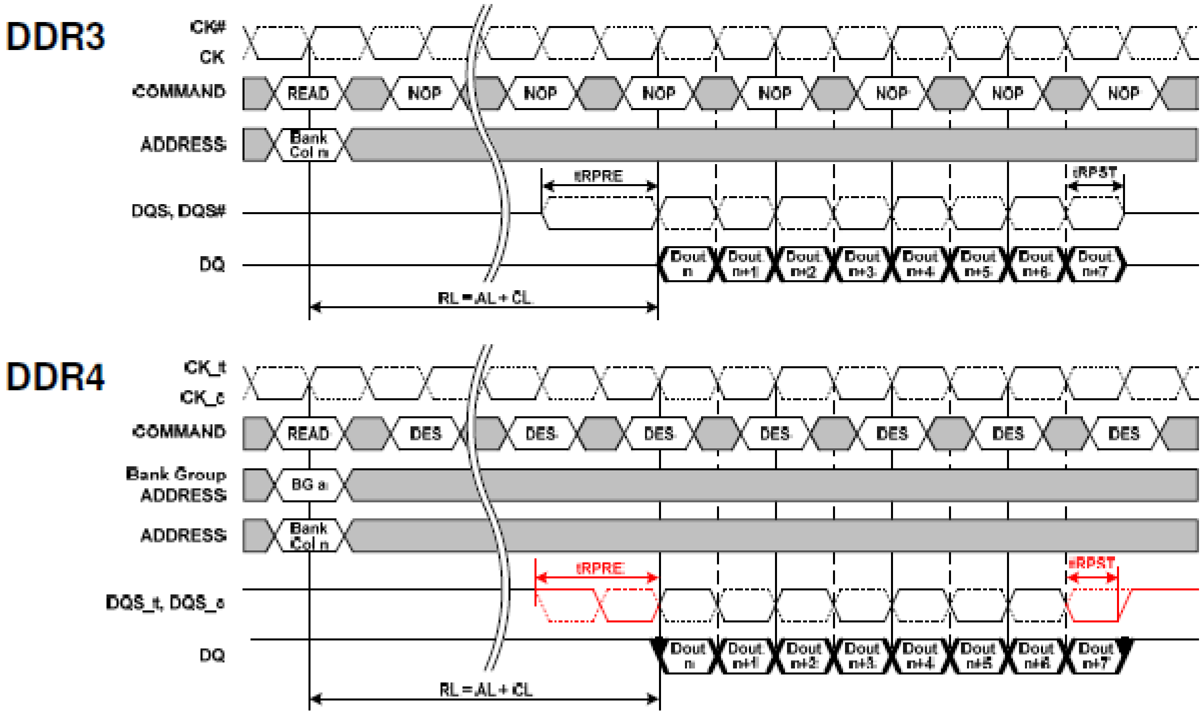


Register drives CKE LOW until ready to transfer input signals

Register proper function and timing starting from here

DDR3 and DDR4 Preamble and Idle Signal Differences

- DQ and DQS signals stay high during idle time
- DQS preamble is same for Read and Writes

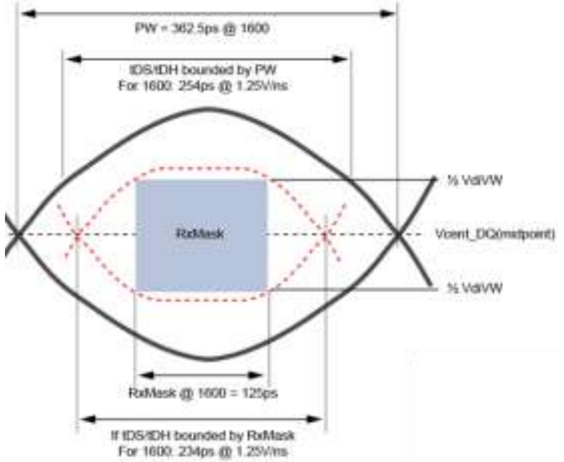


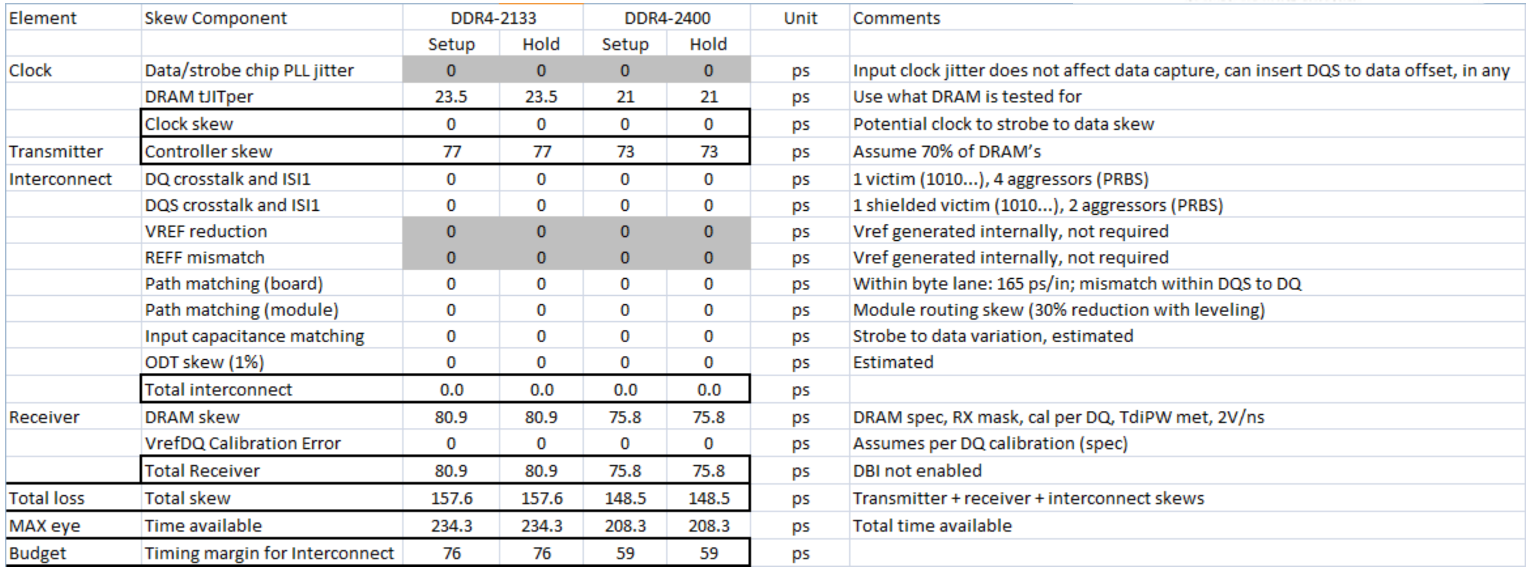
Timing Budget

- UI, the max theoretical timing budget is divided between controller, DRAM and the interconnect.
- When using Timing from the datasheet
  - Adjust datasheet values to simulated slew rates
    - Datasheet values are based on 1V/ns slew rates
- Convert from VIH/Lac and VIH/Ldc to Vref values (A/C bus)
  - Account for slew rate variation
  - Data eyes are based on Vref values
- Rx Mask modifies the READ and WRITE timing budgets calculation
  - Examples will clarifies the calculations
- tDVWp can be used for READ timing budget

Data Write Timing Budget Calculation

- The example shows 152ps available for the interconnect at 2133
  - For timing budget purpose, Rx mask can be the red dotted lines, but actual signal must be black lines

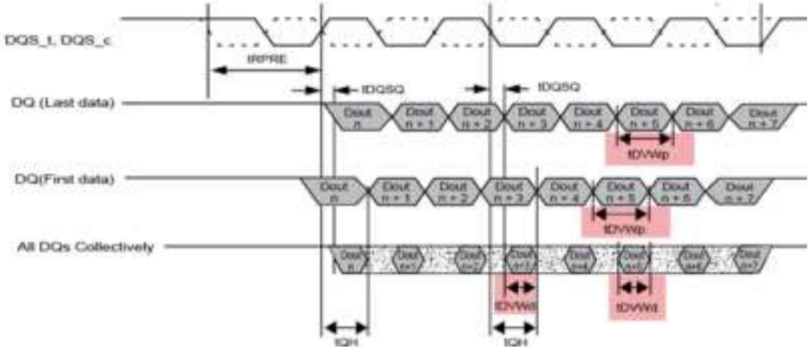


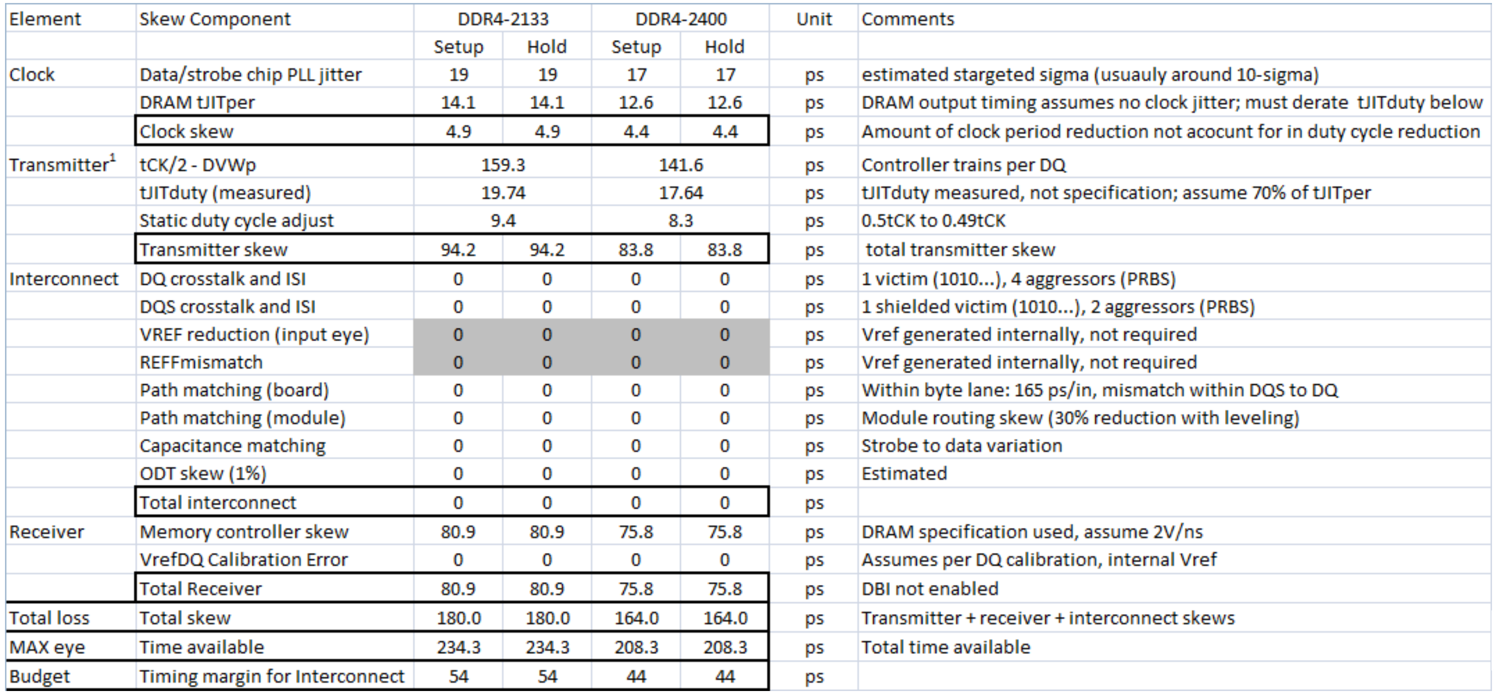


Element	Skew Component	DDR4-2133		DDR4-2400		Unit	Comments
		Setup	Hold	Setup	Hold		
Clock	Data/strobe chip PLL jitter	0	0	0	0	ps	Input clock jitter does not affect data capture, can insert DQS to data offset, in any
	DRAM tJITper	23.5	23.5	21	21	ps	Use what DRAM is tested for
	Clock skew	0	0	0	0	ps	Potential clock to strobe to data skew
Transmitter	Controller skew	77	77	73	73	ps	Assume 70% of DRAM's
Interconnect	DQ crosstalk and ISI1	0	0	0	0	ps	1 victim (1010...), 4 aggressors (PRBS)
	DQS crosstalk and ISI1	0	0	0	0	ps	1 shielded victim (1010...), 2 aggressors (PRBS)
	VREF reduction	0	0	0	0	ps	Vref generated internally, not required
	REFF mismatch	0	0	0	0	ps	Vref generated internally, not required
	Path matching (board)	0	0	0	0	ps	Within byte lane: 165 ps/in; mismatch within DQS to DQ
	Path matching (module)	0	0	0	0	ps	Module routing skew (30% reduction with leveling)
	Input capacitance matching	0	0	0	0	ps	Strobe to data variation, estimated
	ODT skew (1%)	0	0	0	0	ps	Estimated
	Total interconnect	0.0	0.0	0.0	0.0	ps	
Receiver	DRAM skew	80.9	80.9	75.8	75.8	ps	DRAM spec, RX mask, cal per DQ, TdiPW met, 2V/ns
	VrefDQ Calibration Error	0	0	0	0	ps	Assumes per DQ calibration (spec)
	Total Receiver	80.9	80.9	75.8	75.8	ps	DBI not enabled
Total loss	Total skew	157.6	157.6	148.5	148.5	ps	Transmitter + receiver + interconnect skews
MAX eye	Time available	234.3	234.3	208.3	208.3	ps	Total time available
Budget	Timing margin for Interconnect	76	76	59	59	ps	

Data Read Timing Budget Calculation

- Example shows 108ps available for the int
  - Uses new tDVWp specification
    - Requires per DQ calibration





Element	Skew Component	DDR4-2133		DDR4-2400		Unit	Comments
		Setup	Hold	Setup	Hold		
Clock	Data/strobe chip PLL jitter	19	19	17	17	ps	estimated stargeted sigma (usuauly around 10-sigma)
	DRAM tJITper	14.1	14.1	12.6	12.6	ps	DRAM output timing assumes no clock jitter; must derate tJITduty below
	Clock skew	4.9	4.9	4.4	4.4	ps	Amount of clock period reduction not acocunt for in duty cycle reduction
Transmitter[1]	tCK/2 - DVWp	159.3		141.6		ps	Controller trains per DQ
	tJITduty (measured)	19.74		17.64		ps	tJITduty measured, not specification; assume 70% of tJITper
	Static duty cycle adjust	9.4		8.3		ps	0.5tCK to 0.49tCK
	Transmitter skew	94.2	94.2	83.8	83.8	ps	total transmitter skew
Interconnect	DQ crosstalk and ISI	0	0	0	0	ps	1 victim (1010...), 4 aggressors (PRBS)
	DQS crosstalk and ISI	0	0	0	0	ps	1 shielded victim (1010...), 2 aggressors (PRBS)
	VREF reduction (input eye)	0	0	0	0	ps	Vref generated internally, not required
	REFFmismatch	0	0	0	0	ps	Vref generated internally, not required
	Path matching (board)	0	0	0	0	ps	Within byte lane: 165 ps/in, mismatch within DQS to DQ
	Path matching (module)	0	0	0	0	ps	Module routing skew (30% reduction with leveling)
	Capacitance matching	0	0	0	0	ps	Strobe to data variation
	ODT skew (1%)	0	0	0	0	ps	Estimated
	Total interconnect	0	0	0	0	ps	
Receiver	Memory controller skew	80.9	80.9	75.8	75.8	ps	DRAM specification used, assume 2V/ns
	VrefDQ Calibration Error	0	0	0	0	ps	Assumes per DQ calibration, internal Vref
	Total Receiver	80.9	80.9	75.8	75.8	ps	DBI not enabled
Total loss	Total skew	180.0	180.0	164.0	164.0	ps	Transmitter + receiver + interconnect skews
MAX eye	Time available	234.3	234.3	208.3	208.3	ps	Total time available
Budget	Timing margin for Interconnect	54	54	44	44	ps	

Micron

Address Decoding for DDR

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32	33	34	35

			8GB		2GB															64k			8k										8		
0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32	33	34	35

O Byte order in a 64-bit data bus width, 8 bytes

C Column order in a 10-bit column DRAM, 1k columns

B Bank order in a 3-bit bank DRAM, 8 banks

R Row order in a 15-bit row DRAM, 32k rows

S Chip select order in a memory controller w/ 4 CS

M Memory controller order in a part with 2 MC

DDR3			M 0	S 1	S 0	R 14	R 13	R 12	R 11	R 10	R 9	R 8	R 7	R 6	R 5	R 4	R 3	R 2	R 1	R 0	B 2	B 1	B 0	C 9	C 8	C 7	C 6	C 5	C 4	C 3	C 2	C 1	C 0	O 2	O 1	O 0
DDR4			M 0	S 1	S 0	R 13	R 12	R 11	R 10	R 9	R 8	R 7	R 6	R 5	R 4	R 3	R 2	R 1	R 0	BG 1	BG 0	B 1	B 0	C 9	C 8	C 7	C 6	C 5	C 4	C 3	C 2	C 1	C 0	O 2	O 1	O 0

Sent during the Active command

DDR4			M 0	S 1	S 0	R 13	R 12	R 11	R 10	R 9	R 8	R 7	R 6	R 5	R 4	R 3	R 2	R 1	R 0	BG 1	BG 0	B 1	B 0	C 9	C 8	C 7	C 6	C 5	C 4	C 3	C 2	C 1	C 0	O 2	O 1	O 0
DDR3			M 0	S 1	S 0	R 14	R 13	R 12	R 11	R 10	R 9	R 8	R 7	R 6	R 5	R 4	R 3	R 2	R 1	R 0	B 2	B 1	B 0	C 9	C 8	C 7	C 6	C 5	C 4	C 3	C 2	C 1	C 0	O 2	O 1	O 0

Sent during the read/write command

Software Use Case Considerations

- Look-Up tables
  - One write and many reads, 100% random access
  - Worst case is to access the data from the same bank
  - Solution: copy identical table in two or more banks

- Statistics Counters
  - Equal read to write ratio or read-modify write
  - To maintain millions of counters
  - Solution: Instead of reading and writing back to the same counter, read from many (a group of) counters and write back to many (a group of) counters.

i.MX DRAM
Register Programming Aid

Tools for DRAM Bring-up and Debug

DRAM Register Programming aid

Run basic DDR initialization and test memory	Assuming the use of a JTAG debugger, run the **DDR initialization** and open a debugger memory window pointing to the DDR memory map starting address. Try writing a few words and verify if they can be read correctly. If not, re-check the **DDR initialization** sequence and if the DDR has been correctly soldered onto the board. It is also recommended to re-check the schematic to ensure the DDR memory has been connected to the SoC correctly. In some cases, a DRAM calibration routine may need to be executed, see next row.
Run DRAM Stress test (some SoC's include a DRAM calibration routine)	**A unit test that focuses on the robustness of the DRAM interface**. Downloaded through JTAG debugger into internal RAM. Some SoC's **DRAM stress test**, like MX6Q, includes option to run DRAM calibration.

DRAM Stress Test

DRAM Register Programming Aid – Intro

- **Tool to help create DRAM init scripts for specific memory types**
  - Mainly used to help program JEDEC timing parameters
    - tRCD, tRC, tRFC, etc…
    - and for different DRAM parameters like rows, cols, and chip selects
  - Internal tool: Customers told to contact FAE for information
  - **Excel spread sheet based**, transparent, ease-of-use
  - **“Automatically” creates RVD init script** (.inc file)
    - To convert RVD to Lauterbach script format, Contact FAE.
    - How to modify *.inc script to *.ds for use with D5-Stream:
      - Change file name to *.ds
      - Substitute “mem set <reg_add> 32 <reg_val>”
      - For “set mem /32 <reg_add> = <reg_val>”

DRAM Register Programming Aid – Intro (Continued)

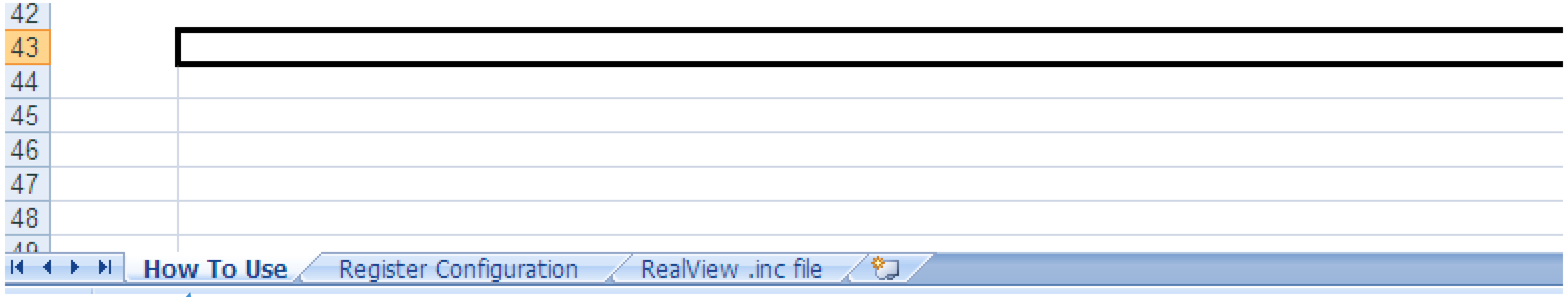
- **Based on scripts provided by design/validation**
- **Anyone can use it, change it, fix it, etc…**
- Each Programming Aid tool based on DRAM tech (DDR3, DDR2, LPDDR2, etc)
- What’s been created to date:
  - MX7D: DDR3, LPDDR3
  - DDR3, LPDDR2; MX6DL: DDR3,LPDDR2; MX6SL: DDR3,LMX6DQ:PDDR2; MX6SX: DDR3, LPDDR2
  - MX50: mDDR, LPDDR2, DDR2
  - MX28: mDDR, DDR2
  - What about other i.MX? No plans yet, need to resource this if enough interest

DRAM Register Programming Aid – Intro (Continued)

- Originated due to Denali controller on MX28 and MX50
  - Denali controller complex, many registers to program
  - Required use of Denali-specific tools available only to factory engineers (due to Denali license); burden on factory support
  - Even with Denali tools, the DRAM init scripts required further “tweaking” due to i.MX design integration
  - **Register programming aid takes into account any “tweaking” and incorporates i.MX design specifics (based on scripts from design/validation)**
  - **Register programming aid offers more visibility on how the DRAM controller is being programmed**
- Register programming aid tool concept carried over to i.MX FIL base controllers, starting with MX53 and MX6 series
- **Tools are available to customers through assigned FAE’s**

DRAM Register Programming Aid – Usage Overview

- Programming aid tool Excel Spreadsheet based
  - There are three tabs (worksheets)

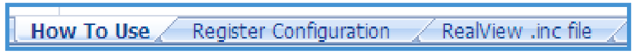


Start with “How To Use” tab for overview on how to use the tool

“Register Configuration” tab is where the user inputs various DRAM parameters

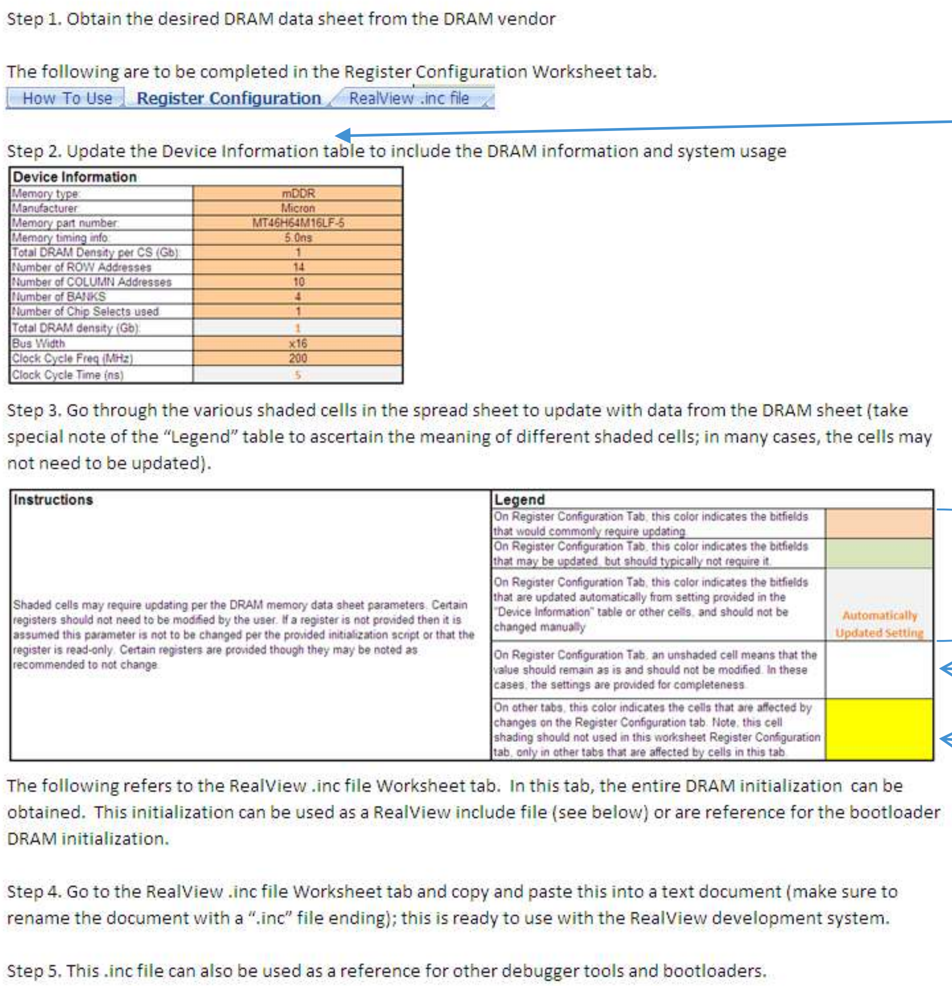
“RealView .inc file” tab contains the automatically generated DRAM initialization in the ARM RVD vernacular

DRAM Register Programming Aid – Usage Overview (Continued)



How To Use | Register Configuration | RealView .inc file

How to use the DRAM register programming aid outline



Step 1. Obtain the desired DRAM data sheet from the DRAM vendor

The following are to be completed in the Register Configuration Worksheet tab.

How To Use | Register Configuration | RealView .inc file

Step 2. Update the Device Information table to include the DRAM information and system usage

Device Information	
Memory type:	mDDR
Manufacturer:	Micron
Memory part number:	MT46H64M16LF-5
Memory timing info:	5.0ns
Total DRAM Density per CS (Gb):	1
Number of ROW Addresses	14
Number of COLUMN Addresses	10
Number of BANKS	4
Number of Chip Selects used	1
Total DRAM density (Gb):	1
Bus Width	x16
Clock Cycle Freq (MHz)	200
Clock Cycle Time (ns)	5

Step 3. Go through the various shaded cells in the spread sheet to update with data from the DRAM sheet (take special note of the "Legend" table to ascertain the meaning of different shaded cells; in many cases, the cells may not need to be updated).

Instructions	Legend	
Shaded cells may require updating per the DRAM memory data sheet parameters. Certain registers should not need to be modified by the user. If a register is not provided then it is assumed this parameter is not to be changed per the provided initialization script or that the register is read-only. Certain registers are provided though they may be noted as recommended to not change.	On Register Configuration Tab, this color indicates the bitfields that would commonly require updating	🟧
	On Register Configuration Tab, this color indicates the bitfields that may be updated, but should typically not require it.	🟩
	On Register Configuration Tab, this color indicates the bitfields that are updated automatically from setting provided in the "Device Information" table or other cells, and should not be changed manually	Automatically Updated Setting
	On Register Configuration Tab, an unshaded cell means that the value should remain as is and should not be modified. In these cases, the settings are provided for completeness.	
	On other tabs, this color indicates the cells that are affected by changes on the Register Configuration tab. Note, this cell shading should not used in this worksheet Register Configuration tab, only in other tabs that are affected by cells in this tab.	🟨

The following refers to the RealView .inc file Worksheet tab. In this tab, the entire DRAM initialization can be obtained. This initialization can be used as a RealView include file (see below) or are reference for the bootloader DRAM initialization.

Step 4. Go to the RealView .inc file Worksheet tab and copy and paste this into a text document (make sure to rename the document with a ".inc" file ending); this is ready to use with the RealView development system.

Step 5. This .inc file can also be used as a reference for other debugger tools and bootloaders.

Note, each spread sheet is for a specific i.MX SoC and DRAM memory type.

Pay attention to shaded cells – this is where you input data

Don't touch un-shaded cells

This is relevant to the "RealView .inc file" tab

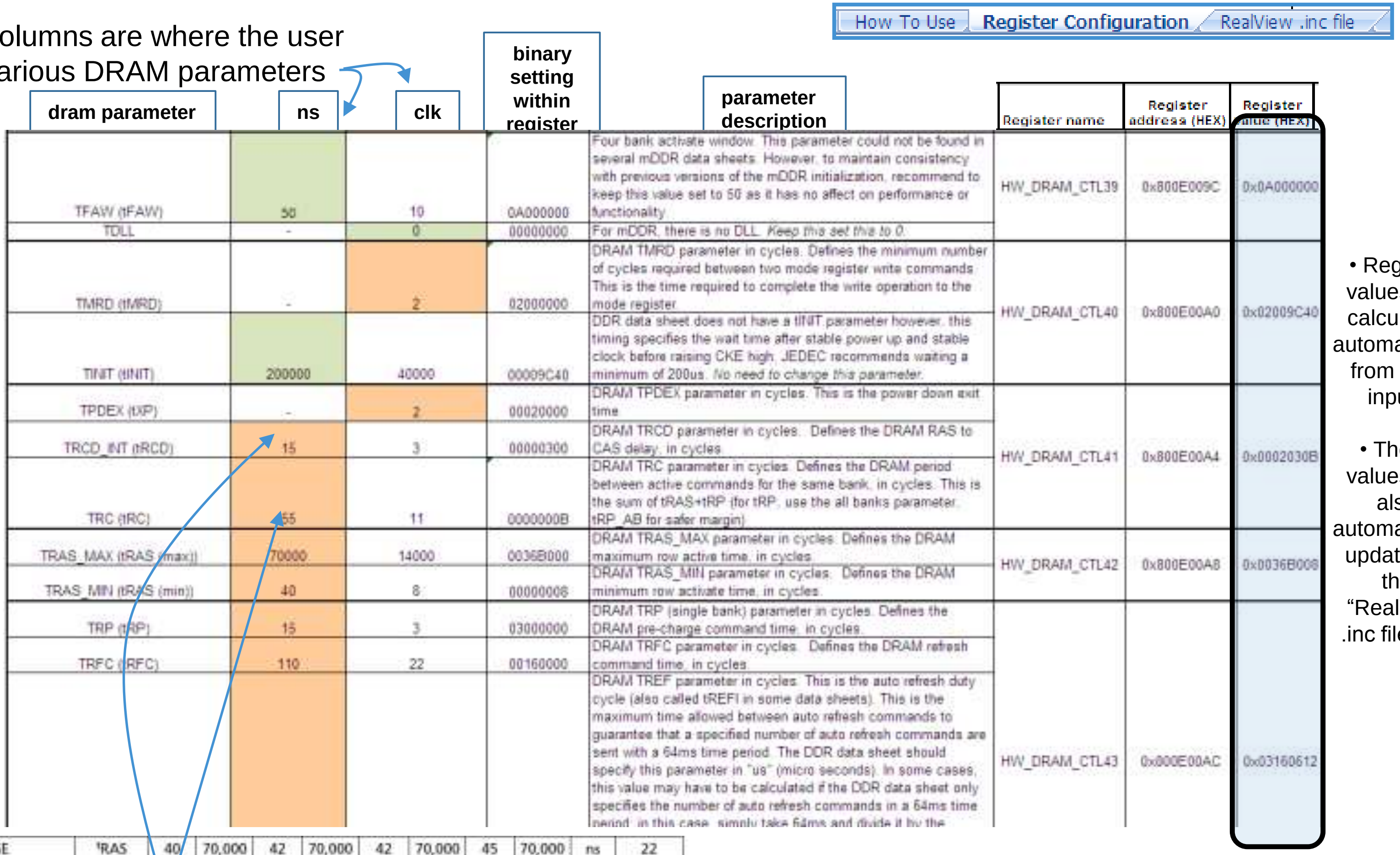
DRAM Register Programming Aid – Usage Overview (Continued)

- These columns are where the user inputs various DRAM parameters

Example tRC:
1. User inputs '55' (in 'ns')
2. Tool calculates that 55ns=11clks*
3. Then configures binary setting for register
4. Tool takes all binary settings to create final register value

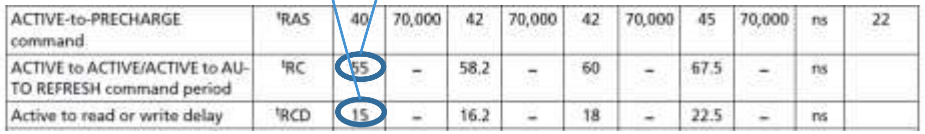
* Clock: 200MHz, clock period: 5ns

Example based on MX28



dram parameter	ns	clk	binary setting within register	parameter description	Register name	Register address (HEX)	Register value (HEX)
TFAW (tFAW)	50	10	0A000000	Four bank activate window. This parameter could not be found in several mDDR data sheets. However, to maintain consistency with previous versions of the mDDR initialization, recommend to keep this value set to 50 as it has no affect on performance or functionality	HW_DRAM_CTL39	0x800E009C	0x0A000000
TDLL	-	0	00000000	For mDDR, there is no DLL. *Keep this set this to 0.*			
TMRD (tMRD)	-	2	02000000	DRAM TMRD parameter in cycles. Defines the minimum number of cycles required between two mode register write commands. This is the time required to complete the write operation to the mode register.	HW_DRAM_CTL40	0x800E00A0	0x02009C40
TINIT (tINIT)	200000	40000	00009C40	DDR data sheet does not have a tINIT parameter however, this timing specifies the wait time after stable power up and stable clock before raising CKE high. JEDEC recommends waiting a minimum of 200us. *No need to change this parameter.*			
TPDEX (tXP)	-	2	00020000	DRAM TPDEX parameter in cycles. This is the power down exit time	HW_DRAM_CTL41	0x800E00A4	0x0002030B
TRCD_INT (tRCD)	15	3	00000300	DRAM TRCD parameter in cycles. Defines the DRAM RAS to CAS delay, in cycles.			
TRC (tRC)	55	11	0000000B	DRAM TRC parameter in cycles. Defines the DRAM period between active commands for the same bank, in cycles. This is the sum of tRAS+tRP (for tRP, use the all banks parameter tRP_AB for safer margin)			
TRAS_MAX (tRAS (max))	70000	14000	0036B000	DRAM TRAS_MAX parameter in cycles. Defines the DRAM maximum row active time, in cycles.	HW_DRAM_CTL42	0x800E00A8	0x0036B008
TRAS_MIN (tRAS (min))	40	8	00000008	DRAM TRAS_MIN parameter in cycles. Defines the DRAM minimum row activate time, in cycles.			
TRP (tRP)	15	3	03000000	DRAM TRP (single bank) parameter in cycles. Defines the DRAM pre-charge command time, in cycles.	HW_DRAM_CTL43	0x800E00AC	0x03160612
TRFC (tRFC)	110	22	00160000	DRAM TRFC parameter in cycles. Defines the DRAM refresh command time, in cycles.			
				DRAM TREF parameter in cycles. This is the auto refresh duty cycle (also called tREFI in some data sheets). This is the maximum time allowed between auto refresh commands to guarantee that a specified number of auto refresh commands are sent with a 64ms time period. The DDR data sheet should specify this parameter in "us" (micro seconds). In some cases, this value may have to be calculated if the DDR data sheet only specifies the number of auto refresh commands in a 64ms time period, in this case, simply take 64ms and divide it by the			

- Register values are calculated automatically from user inputs
- These values are also automatically updated to the "RealView .inc file" tab

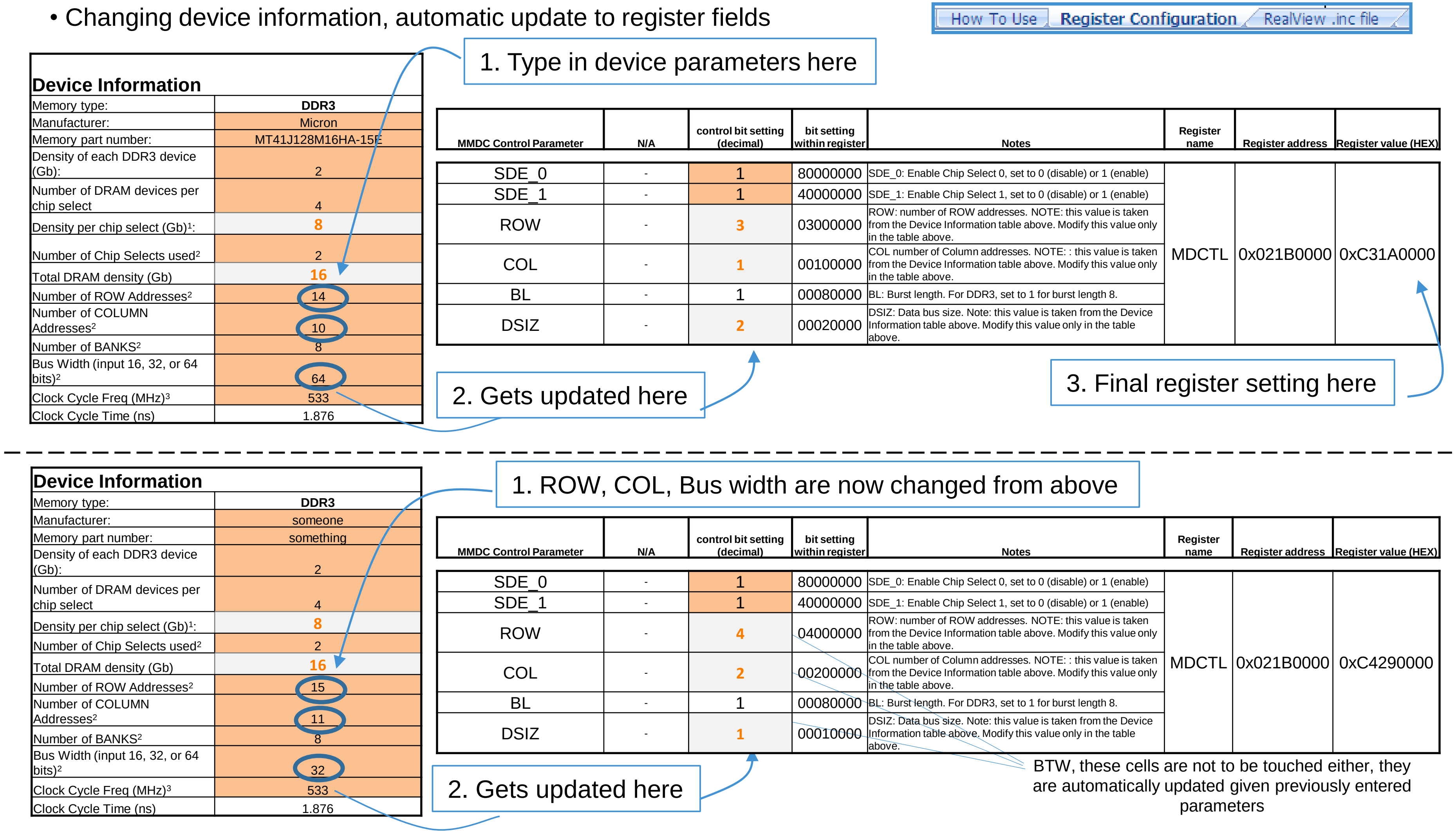


ACTIVE-to-PRECHARGE command	^{t}RAS	40	70,000	42	70,000	42	70,000	45	70,000	ns	22
ACTIVE to ACTIVE/ACTIVE to AU-TO REFRESH command period	^{t}RC	55	–	58.2	–	60	–	67.5	–	ns	
Active to read or write delay	^{t}RCD	15	–	16.2	–	18	–	22.5	–	ns	

Example Micron mDDR data sheet spec

DRAM Register Programming Aid – Usage Overview (Continued)

- Changing device information, automatic update to register fields



1. Type in device parameters here

Device Information	
Memory type:	DDR3
Manufacturer:	Micron
Memory part number:	MT41J128M16HA-15E
Density of each DDR3 device (Gb):	2
Number of DRAM devices per chip select	4
Density per chip select (Gb)[1]:	8
Number of Chip Selects used[2]	2
Total DRAM density (Gb)	16
Number of ROW Addresses[2]	14
Number of COLUMN Addresses[2]	10
Number of BANKS[2]	8
Bus Width (input 16, 32, or 64 bits)[2]	64
Clock Cycle Freq (MHz)[3]	533
Clock Cycle Time (ns)	1.876

MMDC Control Parameter	N/A	control bit setting (decimal)	bit setting within register	Notes	Register name	Register address	Register value (HEX)
SDE_0	-	1	80000000	SDE_0: Enable Chip Select 0, set to 0 (disable) or 1 (enable)	MDCTL	0x021B0000	0xC31A0000
SDE_1	-	1	40000000	SDE_1: Enable Chip Select 1, set to 0 (disable) or 1 (enable)			
ROW	-	3	03000000	ROW: number of ROW addresses. NOTE: this value is taken from the Device Information table above. Modify this value only in the table above.			
COL	-	1	00100000	COL number of Column addresses. NOTE: : this value is taken from the Device Information table above. Modify this value only in the table above.			
BL	-	1	00080000	BL: Burst length. For DDR3, set to 1 for burst length 8.			
DSIZ	-	2	00020000	DSIZ: Data bus size. Note: this value is taken from the Device Information table above. Modify this value only in the table above.			

2. Gets updated here

3. Final register setting here

1. ROW, COL, Bus width are now changed from above

Device Information	
Memory type:	DDR3
Manufacturer:	someone
Memory part number:	something
Density of each DDR3 device (Gb):	2
Number of DRAM devices per chip select	4
Density per chip select (Gb)[1]:	8
Number of Chip Selects used[2]	2
Total DRAM density (Gb)	16
Number of ROW Addresses[2]	15
Number of COLUMN Addresses[2]	11
Number of BANKS[2]	8
Bus Width (input 16, 32, or 64 bits)[2]	32
Clock Cycle Freq (MHz)[3]	533
Clock Cycle Time (ns)	1.876

MMDC Control Parameter	N/A	control bit setting (decimal)	bit setting within register	Notes	Register name	Register address	Register value (HEX)
SDE_0	-	1	80000000	SDE_0: Enable Chip Select 0, set to 0 (disable) or 1 (enable)	MDCTL	0x021B0000	0xC4290000
SDE_1	-	1	40000000	SDE_1: Enable Chip Select 1, set to 0 (disable) or 1 (enable)			
ROW	-	4	04000000	ROW: number of ROW addresses. NOTE: this value is taken from the Device Information table above. Modify this value only in the table above.			
COL	-	2	00200000	COL number of Column addresses. NOTE: : this value is taken from the Device Information table above. Modify this value only in the table above.			
BL	-	1	00080000	BL: Burst length. For DDR3, set to 1 for burst length 8.			
DSIZ	-	1	00010000	DSIZ: Data bus size. Note: this value is taken from the Device Information table above. Modify this value only in the table above.			

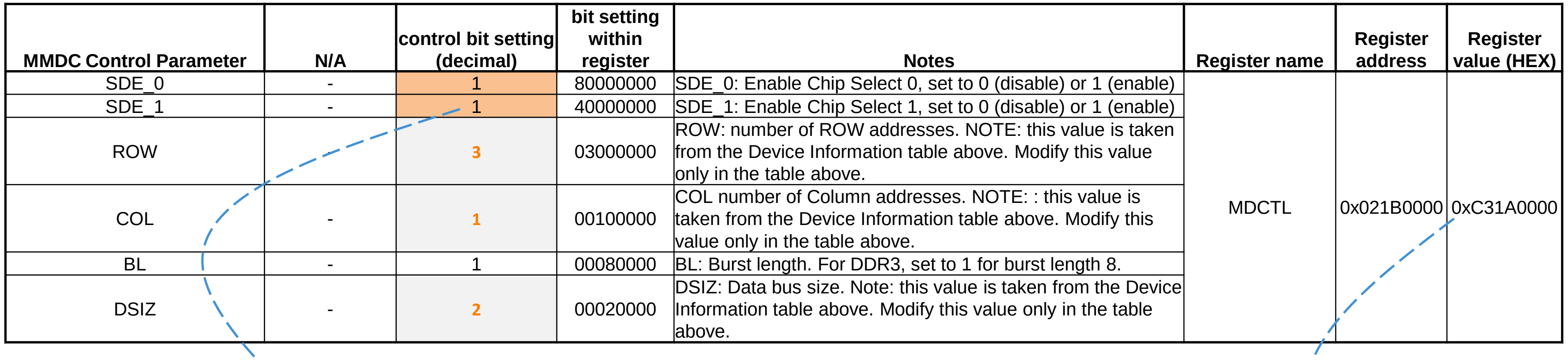
2. Gets updated here

BTW, these cells are not to be touched either, they are automatically updated given previously entered parameters

DRAM Register Programming Aid – Usage Overview (Continued)

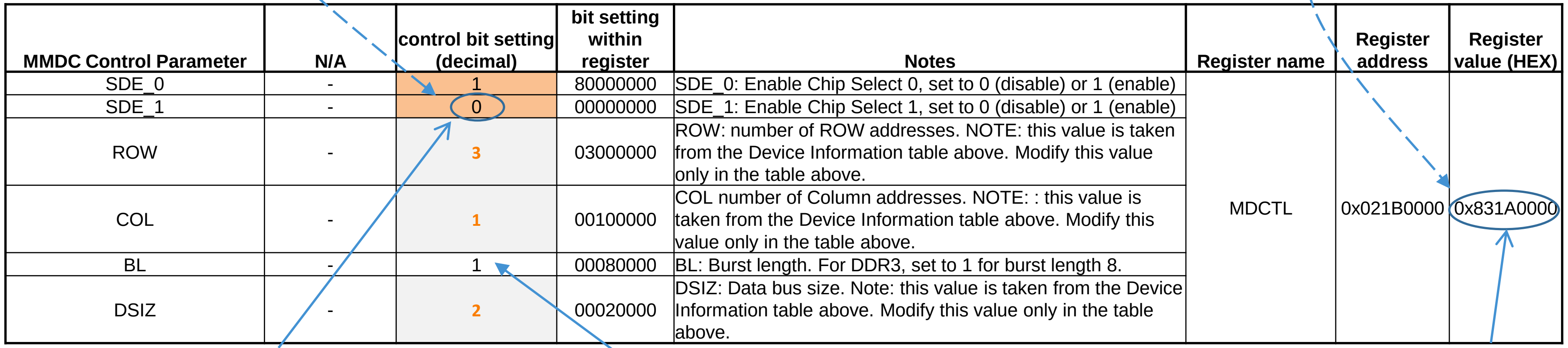
- Another detailed look…
- Let’s say on MX6DQ, you had only one chip select populated (CS0)

Before, the MX6DQ register was as follows for the validation board which has DDR3 on both chip selects



MMDC Control Parameter	N/A	control bit setting (decimal)	bit setting within register	Notes	Register name	Register address	Register value (HEX)
SDE_0	-	1	80000000	SDE_0: Enable Chip Select 0, set to 0 (disable) or 1 (enable)	MDCTL	0x021B0000	0xC31A0000
SDE_1	-	1	40000000	SDE_1: Enable Chip Select 1, set to 0 (disable) or 1 (enable)			
ROW		3	03000000	ROW: number of ROW addresses. NOTE: this value is taken from the Device Information table above. Modify this value only in the table above.			
COL	-	1	00100000	COL number of Column addresses. NOTE: : this value is taken from the Device Information table above. Modify this value only in the table above.			
BL	-	1	00080000	BL: Burst length. For DDR3, set to 1 for burst length 8.			
DSIZ	-	2	00020000	DSIZ: Data bus size. Note: this value is taken from the Device Information table above. Modify this value only in the table above.			

Now, after the change to enable CS0 only, here’s what the register looks like



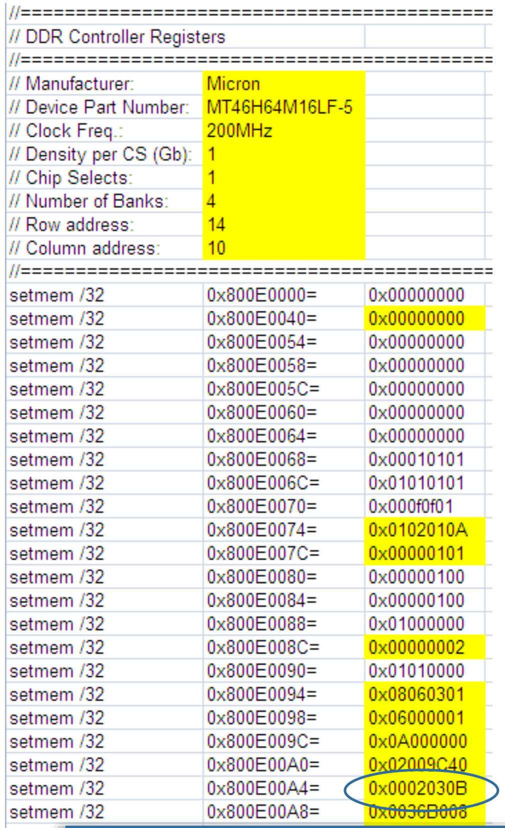
MMDC Control Parameter	N/A	control bit setting (decimal)	bit setting within register	Notes	Register name	Register address	Register value (HEX)
SDE_0	-	1	80000000	SDE_0: Enable Chip Select 0, set to 0 (disable) or 1 (enable)	MDCTL	0x021B0000	0x831A0000
SDE_1	-	0	00000000	SDE_1: Enable Chip Select 1, set to 0 (disable) or 1 (enable)			
ROW	-	3	03000000	ROW: number of ROW addresses. NOTE: this value is taken from the Device Information table above. Modify this value only in the table above.			
COL	-	1	00100000	COL number of Column addresses. NOTE: : this value is taken from the Device Information table above. Modify this value only in the table above.			
BL	-	1	00080000	BL: Burst length. For DDR3, set to 1 for burst length 8.			
DSIZ	-	2	00020000	DSIZ: Data bus size. Note: this value is taken from the Device Information table above. Modify this value only in the table above.			

Clearing this means we are not enabling chip select 1

Register setting gets automatically updated

Don’t touch. When BL cleared, burst length is 4 which is setting for LPDDR2

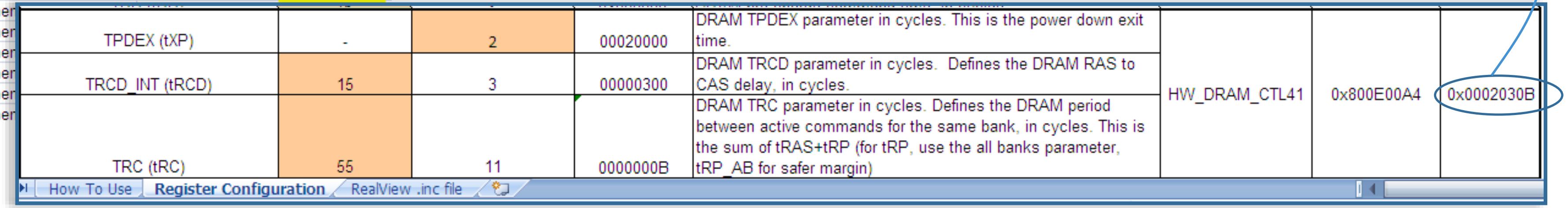
DRAM Register Programming Aid – Usage Overview (Continued)



//===		
// DDR Controller Registers		
//===		
// Manufacturer:	Micron	
// Device Part Number:	MT46H64M16LF-5	
// Clock Freq.:	200MHz	
// Density per CS (Gb):	1	
// Chip Selects:	1	
// Number of Banks:	4	
// Row address:	14	
// Column address:	10	
//===		
setmem /32	0x800E0000=	0x00000000
setmem /32	0x800E0040=	0x00000000
setmem /32	0x800E0054=	0x00000000
setmem /32	0x800E0058=	0x00000000
setmem /32	0x800E005C=	0x00000000
setmem /32	0x800E0060=	0x00000000
setmem /32	0x800E0064=	0x00000000
setmem /32	0x800E0068=	0x00010101
setmem /32	0x800E006C=	0x01010101
setmem /32	0x800E0070=	0x000f0f01
setmem /32	0x800E0074=	0x0102010A
setmem /32	0x800E007C=	0x00000101
setmem /32	0x800E0080=	0x00000100
setmem /32	0x800E0084=	0x00000100
setmem /32	0x800E0088=	0x01000000
setmem /32	0x800E008C=	0x00000002
setmem /32	0x800E0090=	0x01010000
setmem /32	0x800E0094=	0x08060301
setmem /32	0x800E0098=	0x06000001
setmem /32	0x800E009C=	0x0A000000
setmem /32	0x800E00A0=	0x02009C40
setmem /32	0x800E00A4=	0x0002030B
setmem /32	0x800E00A8=	0x0036B008
setmem		
setmem		
setmem		
setmem		
setmem		
setmem		

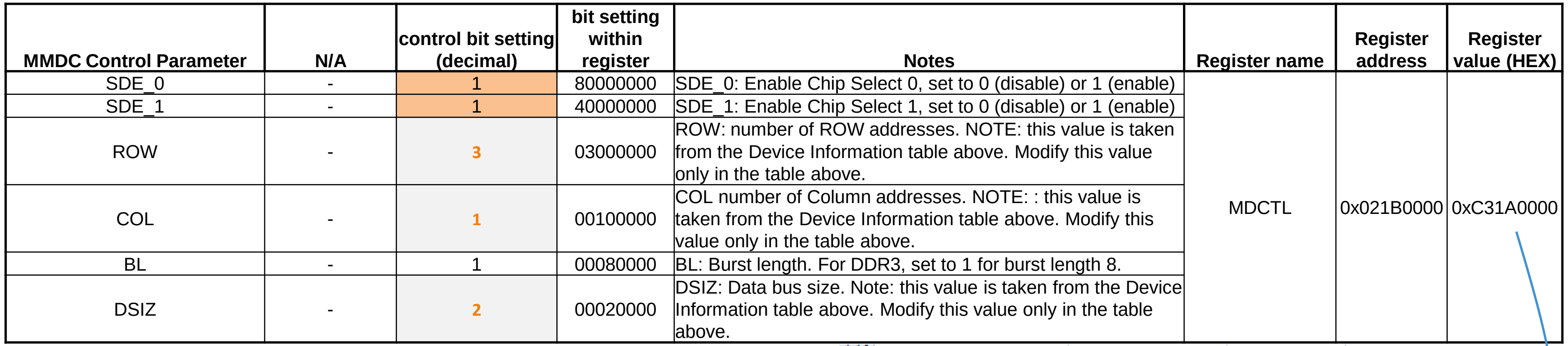
- DRAM initialization automatically generated per template
- Values in yellow are automatically updated from the “Register Configuration” tab (previous slide)
- No user input required in this tab

A few slides back, we saw an example of programming tRC and how the tool created the final value for this register address 0x800E00A4. The value is automatically transferred to this location.



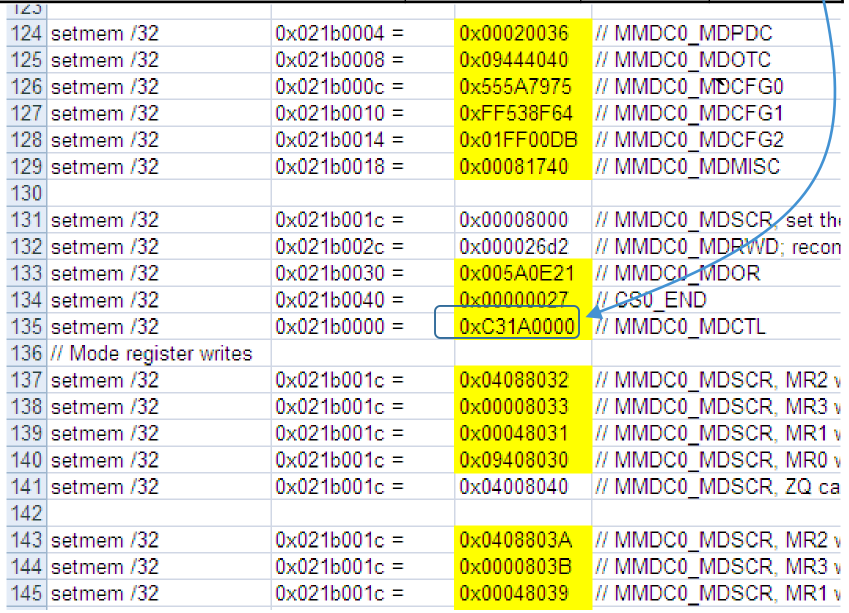
TPDEX (tXP)	-	2	00020000	DRAM TPDEX parameter in cycles. This is the power down exit time.	HW_DRAM_CTL41	0x800E00A4	0x0002030B
TRCD_INT (tRCD)	15	3	00000300	DRAM TRCD parameter in cycles. Defines the DRAM RAS to CAS delay, in cycles.			
TRC (tRC)	55	11	0000000B	DRAM TRC parameter in cycles. Defines the DRAM period between active commands for the same bank, in cycles. This is the sum of tRAS+tRP (for tRP, use the all banks parameter, tRP_AB for safer margin)			

DRAM Register Programming Aid – Usage Overview (Continued)



MMDC Control Parameter	N/A	control bit setting (decimal)	bit setting within register	Notes	Register name	Register address	Register value (HEX)
SDE_0	-	1	80000000	SDE_0: Enable Chip Select 0, set to 0 (disable) or 1 (enable)	MDCTL	0x021B0000	0xC31A0000
SDE_1	-	1	40000000	SDE_1: Enable Chip Select 1, set to 0 (disable) or 1 (enable)			
ROW	-	3	03000000	ROW: number of ROW addresses. NOTE: this value is taken from the Device Information table above. Modify this value only in the table above.			
COL	-	1	00100000	COL number of Column addresses. NOTE: : this value is taken from the Device Information table above. Modify this value only in the table above.			
BL	-	1	00080000	BL: Burst length. For DDR3, set to 1 for burst length 8.			
DSIZ	-	2	00020000	DSIZ: Data bus size. Note: this value is taken from the Device Information table above. Modify this value only in the table above.			

This example from the MX6DQ DDR3 register programming aid illustrates that the register value for MDCTL in the “RealView .inc file” tab is taken directly from the MDCTL Register value cell in the “Register Configuration” tab



```
setmem /32    0x021b0004 =    0x00020036    // MMDC0_MDPDC
setmem /32    0x021b0008 =    0x09444040    // MMDC0_MDOTC
setmem /32    0x021b000c =    0x555A7975    // MMDC0_MDCFG0
setmem /32    0x021b0010 =    0xFF538F64    // MMDC0_MDCFG1
setmem /32    0x021b0014 =    0x01FF00DB    // MMDC0_MDCFG2
setmem /32    0x021b0018 =    0x00081740    // MMDC0_MDMISC

setmem /32    0x021b001c =    0x00008000    // MMDC0_MDSCR, set th
setmem /32    0x021b002c =    0x000026d2    // MMDC0_MDRWD; recon
setmem /32    0x021b0030 =    0x005A0E21    // MMDC0_MDOR
setmem /32    0x021b0040 =    0x00000027    // CS0_END
setmem /32    0x021b0000 =    0xC31A0000    // MMDC0_MDCTL
// Mode register writes
setmem /32    0x021b001c =    0x04088032    // MMDC0_MDSCR, MR2 v
setmem /32    0x021b001c =    0x00008033    // MMDC0_MDSCR, MR3 v
setmem /32    0x021b001c =    0x00048031    // MMDC0_MDSCR, MR1 v
setmem /32    0x021b001c =    0x09408030    // MMDC0_MDSCR, MR0 v
setmem /32    0x021b001c =    0x04008040    // MMDC0_MDSCR, ZQ ca

setmem /32    0x021b001c =    0x0408803A    // MMDC0_MDSCR, MR2 v
setmem /32    0x021b001c =    0x0000803B    // MMDC0_MDSCR, MR3 v
setmem /32    0x021b001c =    0x00048039    // MMDC0_MDSCR, MR1 v
```

DRAM Stress Test – Overview and History

- Took years to develop, constantly evolving to catch elusive DRAM failures
- Compilation of various DRAM sub tests
  - Each sub test contains various data patterns/methods to stress the DRAM interface
  - Started with a few tests using memcpy and various data patterns (1’s and 0’s; A’s and 5’s; pseudo-random, etc)
  - Each new SoC/board presented new DRAM challenges/issues
    - New tests were created to debug
  - Historically, each test was run one-by-one, took time
  - Tests were compiled together into one overall test
    - Each test now called sub-test, executed through a function call
    - Marked beginning of ‘stress test’, sub-tests run in a loop, overnight
  - Cache enabled – important, needed to mimic OS-type transactions; more stress
  - Increment DRAM frequency – method to stress interface accounting for variations in PVT
    - How much above frequency max is considered ample?
      - Historically 30MHz or more seemed good
    - Useful for gathering statistical data; outliers may point to other issues

DRAM Stress Test – Overview and History (Continued)

- Non-OS test to exercise DRAM interface
  - Non-OS: easier than OS to catch/debug DRAM failures
  - Used by factory as part of DRAM validation
  - Helps diagnose but doesn’t fix DDR problems
- Purpose: Root out potential signal integrity issues due to inadequate board layout
  - Primarily uses sequential bursts of back-to-back data looking for simultaneous switching noise (SSN)
  - Validation vehicle that reports how robust DRAM interface is given current set of parameters (i.e. drive strength settings, timing parameters, board layout, etc)
- Runs from internal RAM
  - Device under test is DRAM itself, don’t execute out of same memory being tested
  - Download to IRAM via JTAG debugger tools (RVD, Lauterbach, Macraigor)
  - Now available in a USB version.
- FAE’s are able to provide tailored code if necessary.
  - Any debugger that supports specific SoC ARM core and elf should work
  - Factory not responsible to test every debugger or debug it if doesn’t work on other debuggers

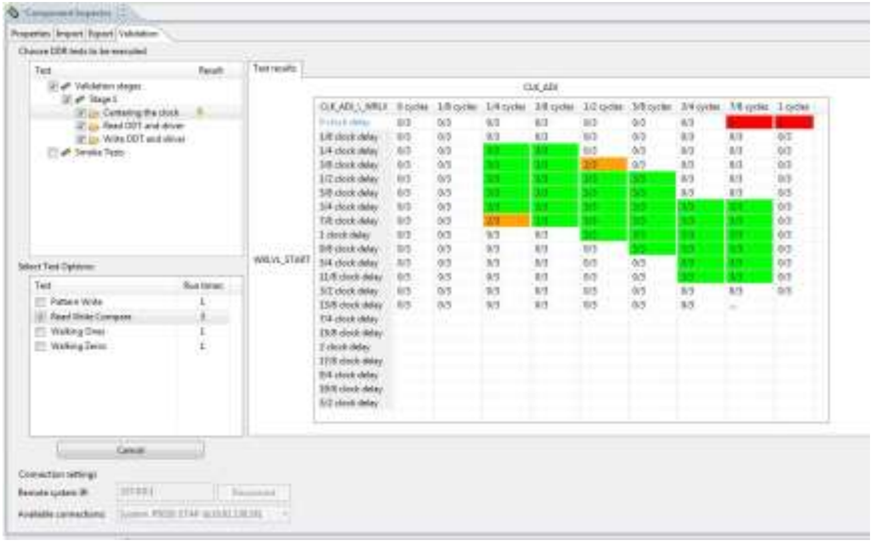
DRAM Stress Test – Overview and History (Continued)

- Once DRAM stress test passes with ample margin, are we guaranteed the OS will never fail due to DRAM issues?
  - High degree of confidence DRAM robust enough, but…
    - OS is still the most stressful, particularly an OS stress test like Bonnie++
    - Recommend to run any OS stress tests to double check
  - Currently Supported SoC:
    - MX28, MX508, MX51, MX53, MX6DQ (Arik) MX6DL (Rigel), MX6SL (Megrez), MX6SX (Pele), MX7D
  - No plans to back port to older legacy processors
    - Issues encountered as some only have 16KB of IRAM
- Challenges
  - Test becoming too big to fit inside IRAM (128KB becoming a limiting factor)
  - When new sub-tests are created, no plan in place to back port to older processors

Layerscape DRAM Configuration and Validation via QCVS Tools

Optimize/Validate the DDR Interface on your Board

- The board dependent parameters are optimized by connecting to your board and running targeted tests
- After this stage, the DDR interface in your board is optimized/validated



Choose DDR tests to be executed

Test	Result
☑ Validation stages	
☑ Stage 1	
☑ Centering the clock	
☑ Read ODT and driver	
☑ Write ODT and driver	
□ Smoke Tests	

Select Test Options:

Test	Run times
□ Pattern Write	1
☑ Read Write Compare	3
□ Walking Ones	1
□ Walking Zeros	1

Connection settings

Remote system IP: [illegible]

Available connections: [illegible]

CLK_ADJ

WRLVL_START

CLK_ADJ \ WRLV	0 cycles	1/8 cycles	1/4 cycles	3/8 cycles	1/2 cycles	5/8 cycles	3/4 cycles	7/8 cycles	1 cycles
0 clock delay	0/3	0/3	0/3	0/3	0/3	0/3	0/3	🟥	🟥
1/8 clock delay	0/3	0/3	0/3	0/3	0/3	0/3	0/3	0/3	0/3
1/4 clock delay	0/3	0/3	3/3	3/3	0/3	0/3	0/3	0/3	0/3
3/8 clock delay	0/3	0/3	3/3	3/3	2/3	0/3	0/3	0/3	0/3
1/2 clock delay	0/3	0/3	3/3	3/3	3/3	3/3	0/3	0/3	0/3
5/8 clock delay	0/3	0/3	3/3	3/3	3/3	3/3	0/3	0/3	0/3
3/4 clock delay	0/3	0/3	3/3	3/3	3/3	3/3	3/3	3/3	0/3
7/8 clock delay	0/3	0/3	2/3	3/3	3/3	3/3	3/3	3/3	0/3
1 clock delay	0/3	0/3	0/3	0/3	3/3	3/3	3/3	3/3	0/3
9/8 clock delay	0/3	0/3	0/3	0/3	0/3	3/3	3/3	3/3	0/3
5/4 clock delay	0/3	0/3	0/3	0/3	0/3	0/3	3/3	3/3	0/3
11/8 clock delay	0/3	0/3	0/3	0/3	0/3	0/3	3/3	3/3	0/3
3/2 clock delay	0/3	0/3	0/3	0/3	0/3	0/3	0/3	0/3	0/3
13/8 clock delay	0/3	0/3	0/3	0/3	0/3	0/3	0/3	...	
7/4 clock delay									
15/8 clock delay									
2 clock delay									
17/8 clock delay									
9/4 clock delay									
19/8 clock delay									
5/2 clock delay									

Register Configuration

Two general types of registers to be configured in the memory controller:

- First register type are set to the DRAM related parameter values, that are provided via SPD or DRAM datasheet. Over 100 register fields fall under this category.

- Second register type are the Non-SPD values that are set based on customer’s application. For example:
  - On-die-termination (ODT) settings for DRAM and controller
  - Driver impedance setting for DRAM and controller
  - Clock adjust value selection
  - Write leveling start value (WRLVL_START)

Using QCS DDRv Tool

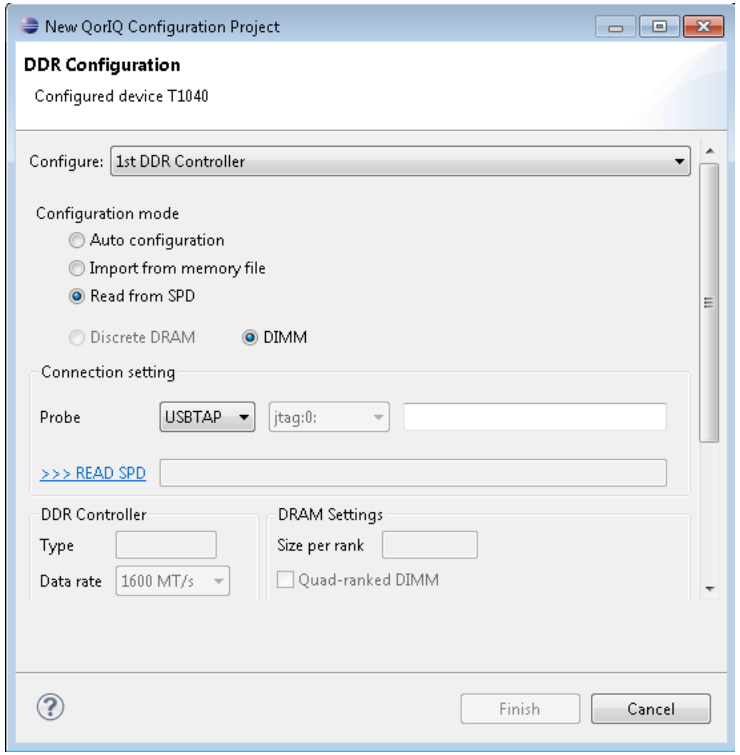
Configure and optimize your DDR interface in a matter of hours

1. Use the tool to generate the DDR register settings
    - Use the latest revision
    - Select the SPD option in configuration wizard when DIMM is used
    - Select Auto Configuration when Discrete DRAM is used
2. Optimize the DDR register setting on your QorIQ board
    - Run the clock centering test
    - Optimize the ODT and drive strength for read and write

DDRv DEMO: http://www.freescale.com/webapp/sps/site/prod_summary.jsp?code=PE_QORIQ_DDRV

Generate the DDR Register Settings

- Using DDR wizard, select the SPD option for DIMMs, or Auto configure for DIMMs or Discrete DRAM
- Press finish and you have generated DDR register settings



New QorIQ Configuration Project

DDR Configuration

Configured device T1040

Configure: 1st DDR Controller

Configuration mode

○ Auto configuration
○ Import from memory file
● Read from SPD

○ Discrete DRAM ● DIMM

Connection setting

Probe: USBTAP, jtag:0:

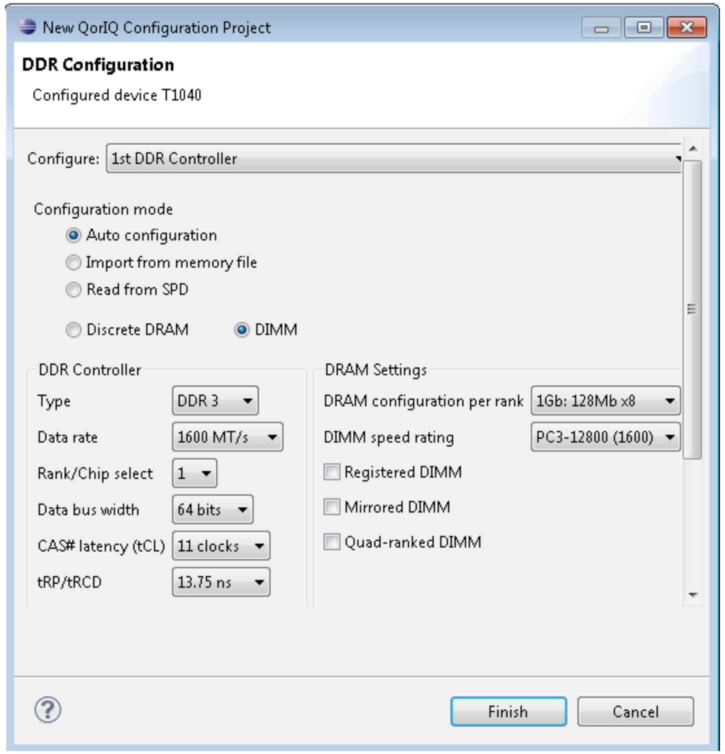
>>> READ SPD

DDR Controller

Type:
Data rate: 1600 MT/s

DRAM Settings

Size per rank:
□ Quad-ranked DIMM



New QorIQ Configuration Project

DDR Configuration

Configured device T1040

Configure: 1st DDR Controller

Configuration mode

● Auto configuration
○ Import from memory file
○ Read from SPD

○ Discrete DRAM ● DIMM

DDR Controller

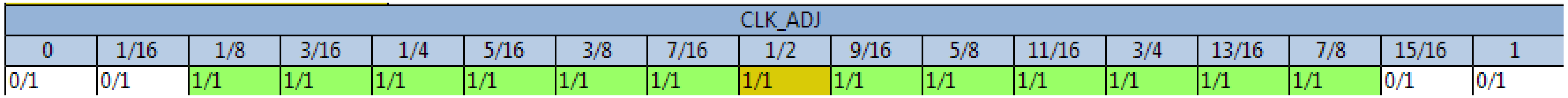
Type: DDR 3
Data rate: 1600 MT/s
Rank/Chip select: 1
Data bus width: 64 bits
CAS# latency (tCL): 11 clocks
tRP/tRCD: 13.75 ns

DRAM Settings

DRAM configuration per rank: 1Gb: 128Mb x8
DIMM speed rating: PC3-12800 (1600)
□ Registered DIMM
□ Mirrored DIMM
□ Quad-ranked DIMM

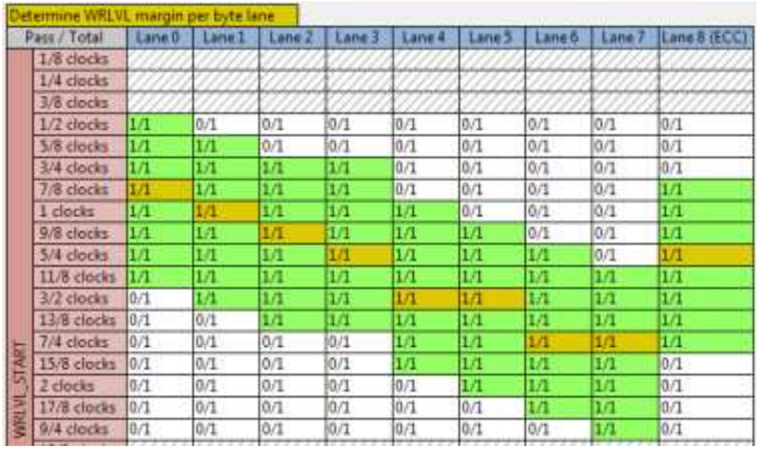
DDR Interface ADD/CMND Bus Margins via QCVS Tool

- Clock signal is stepped cross the address bus eye unit interval and tool regenerate a pass/fail address bus eye.
    - In the example below the address eye is passing from 1/8 clk to 7/8 of clock. This is 80% of open eye from maximum available address bus unit interval.



CLK_ADJ																
0	1/16	1/8	3/16	1/4	5/16	3/8	7/16	1/2	9/16	5/8	11/16	3/4	13/16	7/8	15/16	1
0/1	0/1	1/1	1/1	1/1	1/1	1/1	1/1	1/1	1/1	1/1	1/1	1/1	1/1	1/1	0/1	0/1

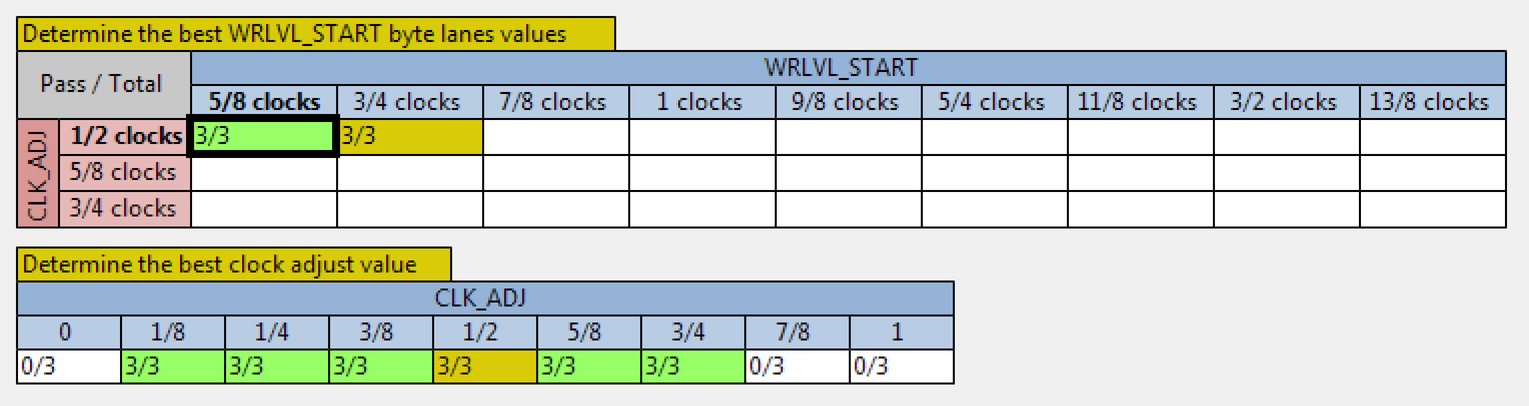
- Write level margin table provides the reconstruction pass fail margins for each byte lane.



Determine WRLVL margin per byte lane

Pass / Total		Lane 0	Lane 1	Lane 2	Lane 3	Lane 4	Lane 5	Lane 6	Lane 7	Lane 8 (ECC)
WRLVL_START	1/8 clocks	▨	▨	▨	▨	▨	▨	▨	▨	▨
	1/4 clocks	▨	▨	▨	▨	▨	▨	▨	▨	▨
	3/8 clocks	▨	▨	▨	▨	▨	▨	▨	▨	▨
	1/2 clocks	1/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1
	5/8 clocks	1/1	1/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1
	3/4 clocks	1/1	1/1	1/1	1/1	0/1	0/1	0/1	0/1	0/1
	7/8 clocks	1/1	1/1	1/1	1/1	0/1	0/1	0/1	0/1	1/1
	1 clocks	1/1	1/1	1/1	1/1	1/1	0/1	0/1	0/1	1/1
	9/8 clocks	1/1	1/1	1/1	1/1	1/1	1/1	0/1	0/1	1/1
	5/4 clocks	1/1	1/1	1/1	1/1	1/1	1/1	1/1	0/1	1/1
	11/8 clocks	1/1	1/1	1/1	1/1	1/1	1/1	1/1	1/1	1/1
	3/2 clocks	0/1	1/1	1/1	1/1	1/1	1/1	1/1	1/1	1/1
	13/8 clocks	0/1	0/1	1/1	1/1	1/1	1/1	1/1	1/1	1/1
	7/4 clocks	0/1	0/1	0/1	0/1	1/1	1/1	1/1	1/1	1/1
	15/8 clocks	0/1	0/1	0/1	0/1	1/1	1/1	1/1	1/1	0/1
	2 clocks	0/1	0/1	0/1	0/1	0/1	1/1	1/1	1/1	0/1
	17/8 clocks	0/1	0/1	0/1	0/1	0/1	0/1	1/1	1/1	0/1
	9/4 clocks	0/1	0/1	0/1	0/1	0/1	0/1	0/1	1/1	0/1

P2041RDB - DDRv Results

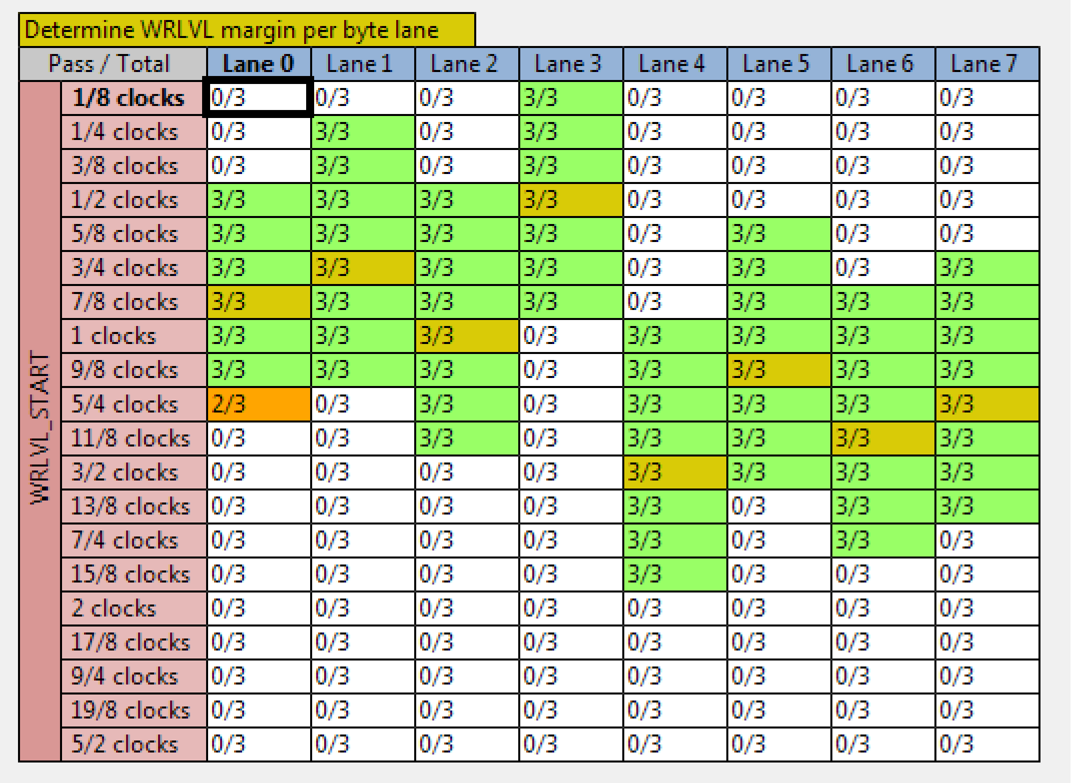


Determine the best WRLVL_START byte lanes values

Pass / Total		WRLVL_START								
		5/8 clocks	3/4 clocks	7/8 clocks	1 clocks	9/8 clocks	5/4 clocks	11/8 clocks	3/2 clocks	13/8 clocks
CLK_ADJ	**1/2 clocks**	3/3	3/3							
	5/8 clocks									
	3/4 clocks									

Determine the best clock adjust value

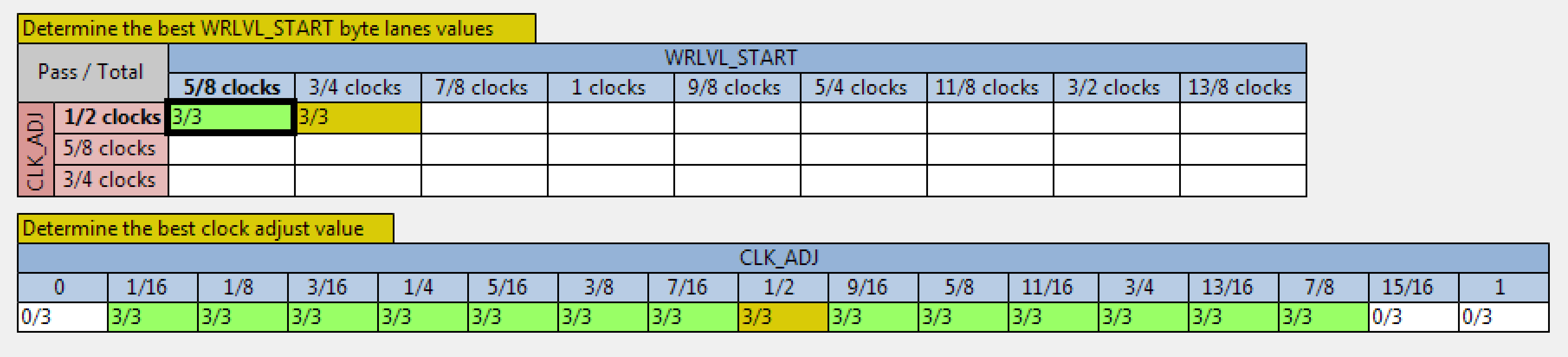
CLK_ADJ								
0	1/8	1/4	3/8	1/2	5/8	3/4	7/8	1
0/3	3/3	3/3	3/3	3/3	3/3	3/3	0/3	0/3



Determine WRLVL margin per byte lane

Pass / Total		**Lane 0**	Lane 1	Lane 2	Lane 3	Lane 4	Lane 5	Lane 6	Lane 7
WRLVL_START	**1/8 clocks**	0/3	0/3	0/3	3/3	0/3	0/3	0/3	0/3
	1/4 clocks	0/3	3/3	0/3	3/3	0/3	0/3	0/3	0/3
	3/8 clocks	0/3	3/3	0/3	3/3	0/3	0/3	0/3	0/3
	1/2 clocks	3/3	3/3	3/3	3/3	0/3	0/3	0/3	0/3
	5/8 clocks	3/3	3/3	3/3	3/3	0/3	3/3	0/3	0/3
	3/4 clocks	3/3	3/3	3/3	3/3	0/3	3/3	0/3	3/3
	7/8 clocks	3/3	3/3	3/3	3/3	0/3	3/3	3/3	3/3
	1 clocks	3/3	3/3	3/3	0/3	3/3	3/3	3/3	3/3
	9/8 clocks	3/3	3/3	3/3	0/3	3/3	3/3	3/3	3/3
	5/4 clocks	2/3	0/3	3/3	0/3	3/3	3/3	3/3	3/3
	11/8 clocks	0/3	0/3	3/3	0/3	3/3	3/3	3/3	3/3
	3/2 clocks	0/3	0/3	0/3	0/3	3/3	3/3	3/3	3/3
	13/8 clocks	0/3	0/3	0/3	0/3	3/3	0/3	3/3	3/3
	7/4 clocks	0/3	0/3	0/3	0/3	3/3	0/3	3/3	0/3
	15/8 clocks	0/3	0/3	0/3	0/3	3/3	0/3	0/3	0/3
	2 clocks	0/3	0/3	0/3	0/3	0/3	0/3	0/3	0/3
	17/8 clocks	0/3	0/3	0/3	0/3	0/3	0/3	0/3	0/3
	9/4 clocks	0/3	0/3	0/3	0/3	0/3	0/3	0/3	0/3
	19/8 clocks	0/3	0/3	0/3	0/3	0/3	0/3	0/3	0/3
	5/2 clocks	0/3	0/3	0/3	0/3	0/3	0/3	0/3	0/3

T1040RDB - DDRv Results

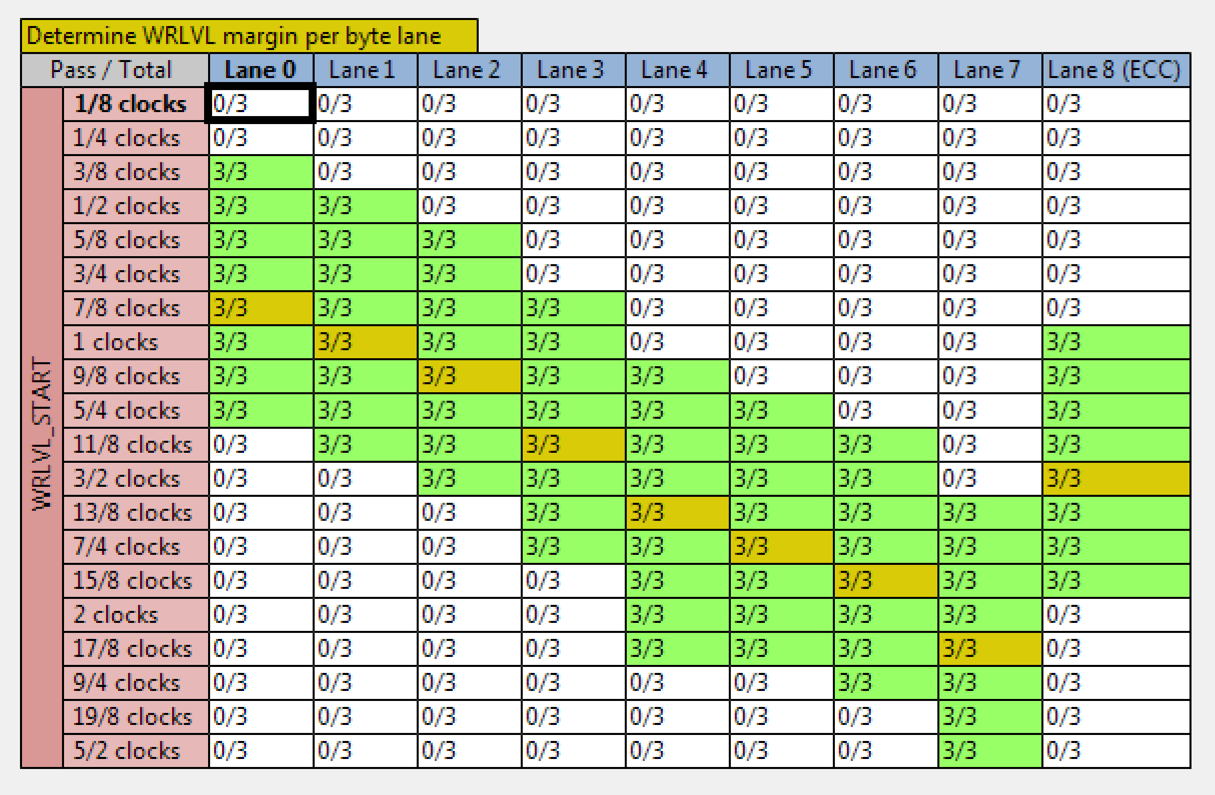


Determine the best WRLVL_START byte lanes values

Pass / Total		WRLVL_START								
		5/8 clocks	3/4 clocks	7/8 clocks	1 clocks	9/8 clocks	5/4 clocks	11/8 clocks	3/2 clocks	13/8 clocks
CLK_ADJ	**1/2 clocks**	3/3	3/3							
	5/8 clocks									
	3/4 clocks									

Determine the best clock adjust value

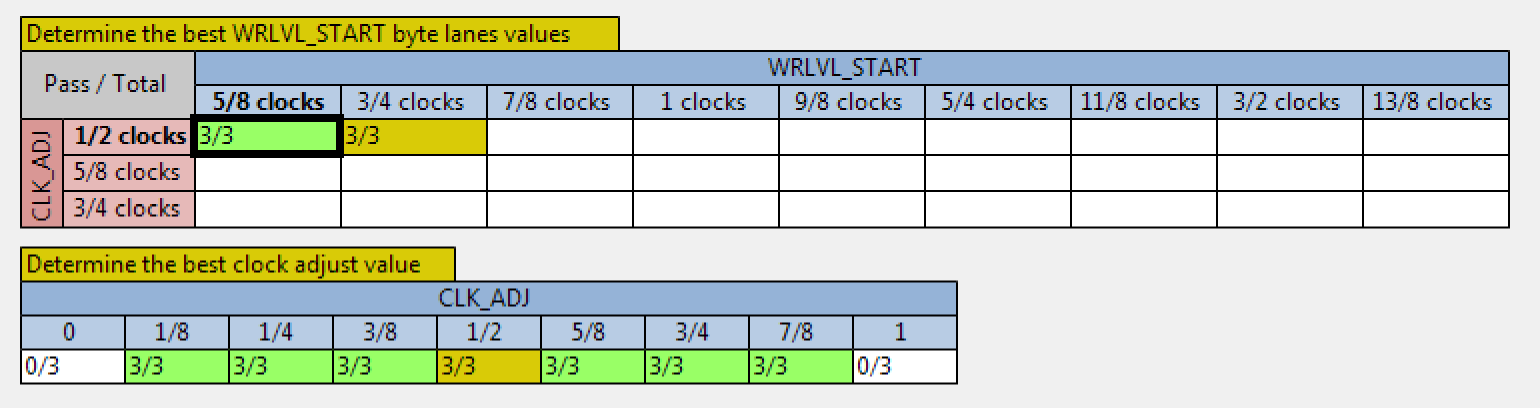
CLK_ADJ																
0	1/16	1/8	3/16	1/4	5/16	3/8	7/16	1/2	9/16	5/8	11/16	3/4	13/16	7/8	15/16	1
0/3	3/3	3/3	3/3	3/3	3/3	3/3	3/3	3/3	3/3	3/3	3/3	3/3	3/3	3/3	0/3	0/3



Determine WRLVL margin per byte lane

Pass / Total		**Lane 0**	Lane 1	Lane 2	Lane 3	Lane 4	Lane 5	Lane 6	Lane 7	Lane 8 (ECC)
WRLVL_START	**1/8 clocks**	0/3	0/3	0/3	0/3	0/3	0/3	0/3	0/3	0/3
	1/4 clocks	0/3	0/3	0/3	0/3	0/3	0/3	0/3	0/3	0/3
	3/8 clocks	3/3	0/3	0/3	0/3	0/3	0/3	0/3	0/3	0/3
	1/2 clocks	3/3	3/3	0/3	0/3	0/3	0/3	0/3	0/3	0/3
	5/8 clocks	3/3	3/3	3/3	0/3	0/3	0/3	0/3	0/3	0/3
	3/4 clocks	3/3	3/3	3/3	0/3	0/3	0/3	0/3	0/3	0/3
	7/8 clocks	3/3	3/3	3/3	3/3	0/3	0/3	0/3	0/3	0/3
	1 clocks	3/3	3/3	3/3	3/3	0/3	0/3	0/3	0/3	3/3
	9/8 clocks	3/3	3/3	3/3	3/3	3/3	0/3	0/3	0/3	3/3
	5/4 clocks	3/3	3/3	3/3	3/3	3/3	3/3	0/3	0/3	3/3
	11/8 clocks	0/3	3/3	3/3	3/3	3/3	3/3	3/3	0/3	3/3
	3/2 clocks	0/3	0/3	3/3	3/3	3/3	3/3	3/3	0/3	3/3
	13/8 clocks	0/3	0/3	0/3	3/3	3/3	3/3	3/3	3/3	3/3
	7/4 clocks	0/3	0/3	0/3	3/3	3/3	3/3	3/3	3/3	3/3
	15/8 clocks	0/3	0/3	0/3	0/3	3/3	3/3	3/3	3/3	3/3
	2 clocks	0/3	0/3	0/3	0/3	3/3	3/3	3/3	3/3	0/3
	17/8 clocks	0/3	0/3	0/3	0/3	3/3	3/3	3/3	3/3	0/3
	9/4 clocks	0/3	0/3	0/3	0/3	0/3	0/3	3/3	3/3	0/3
	19/8 clocks	0/3	0/3	0/3	0/3	0/3	0/3	0/3	3/3	0/3
	5/2 clocks	0/3	0/3	0/3	0/3	0/3	0/3	0/3	3/3	0/3

P5040 Customer Board - DDRv Results

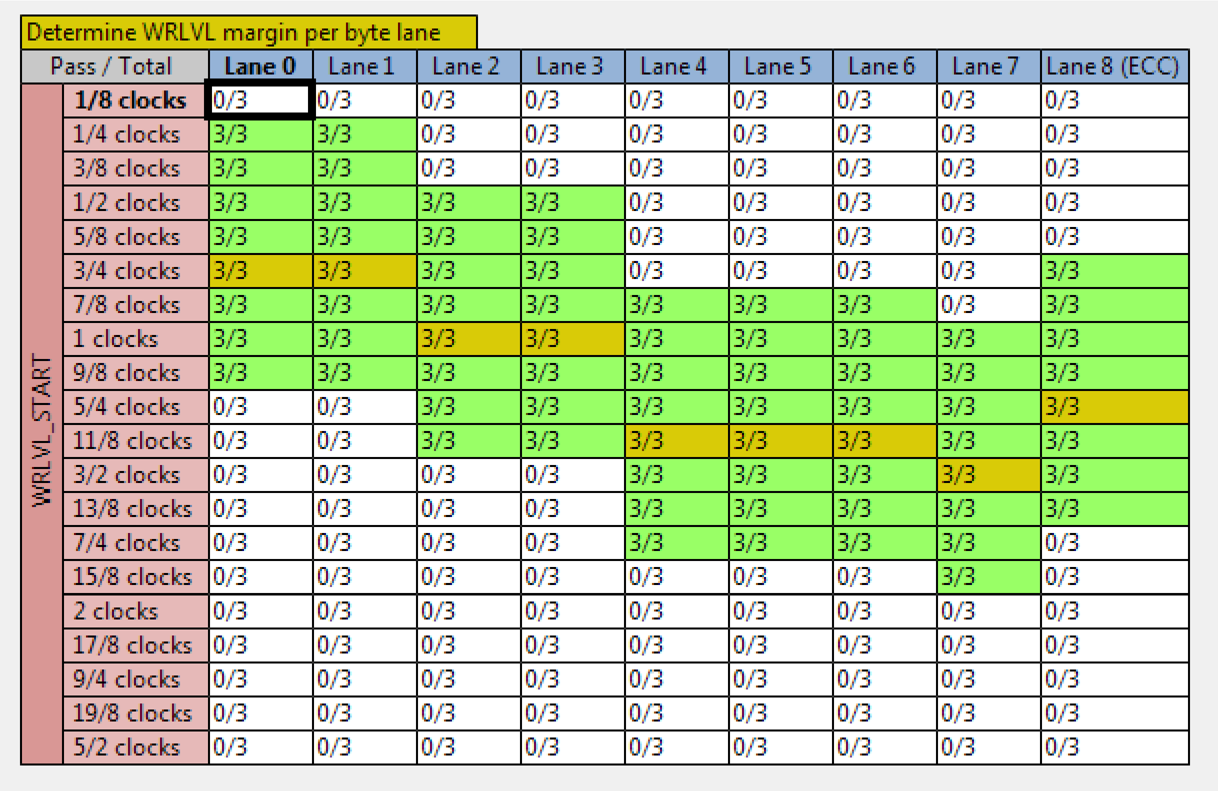


Determine the best WRLVL_START byte lanes values

	Pass / Total	WRLVL_START								
		5/8 clocks	3/4 clocks	7/8 clocks	1 clocks	9/8 clocks	5/4 clocks	11/8 clocks	3/2 clocks	13/8 clocks
CLK_ADJ	**1/2 clocks**	3/3	3/3							
	5/8 clocks									
	3/4 clocks									

Determine the best clock adjust value

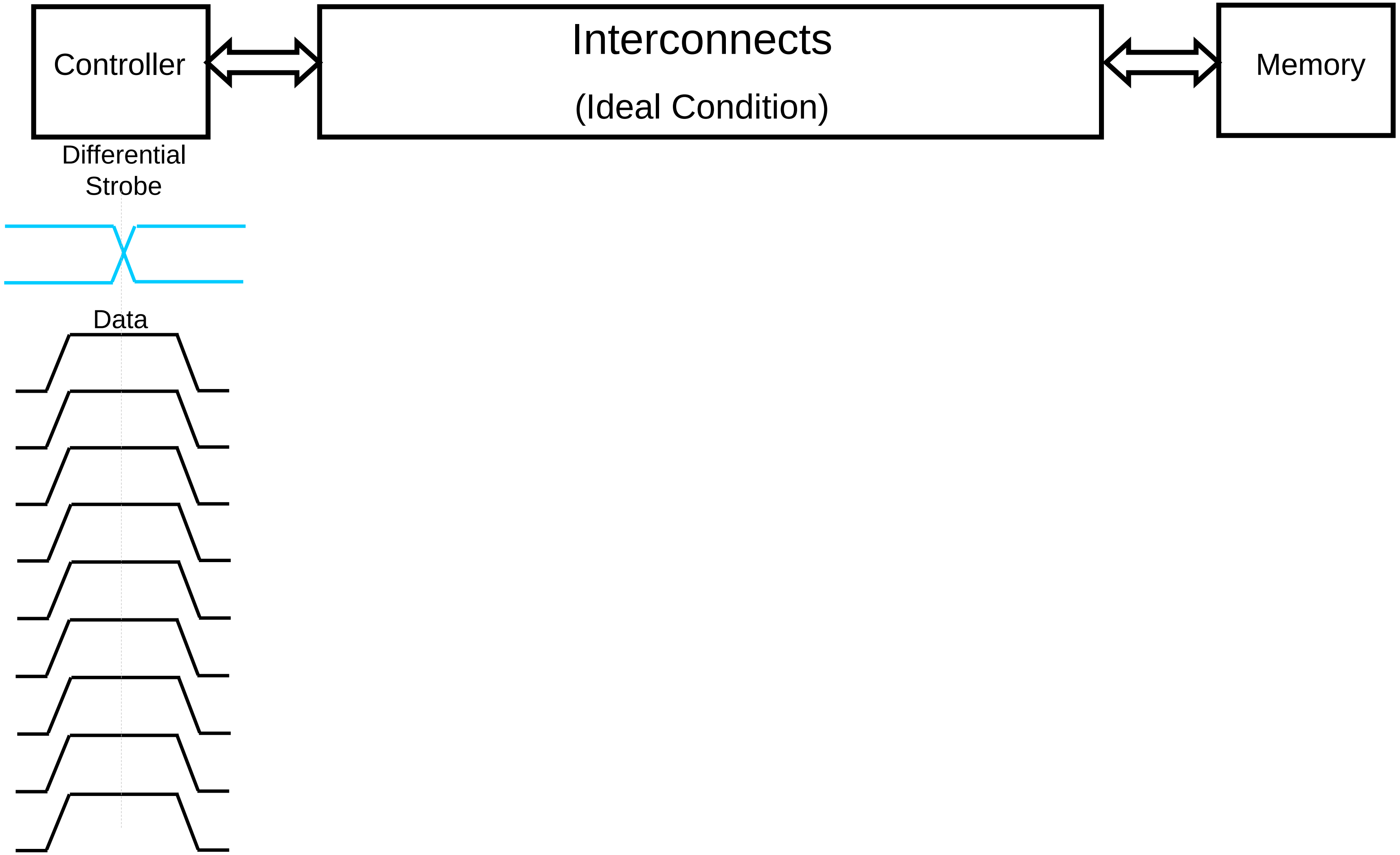
CLK_ADJ								
0	1/8	1/4	3/8	1/2	5/8	3/4	7/8	1
0/3	3/3	3/3	3/3	3/3	3/3	3/3	3/3	0/3



Determine WRLVL margin per byte lane

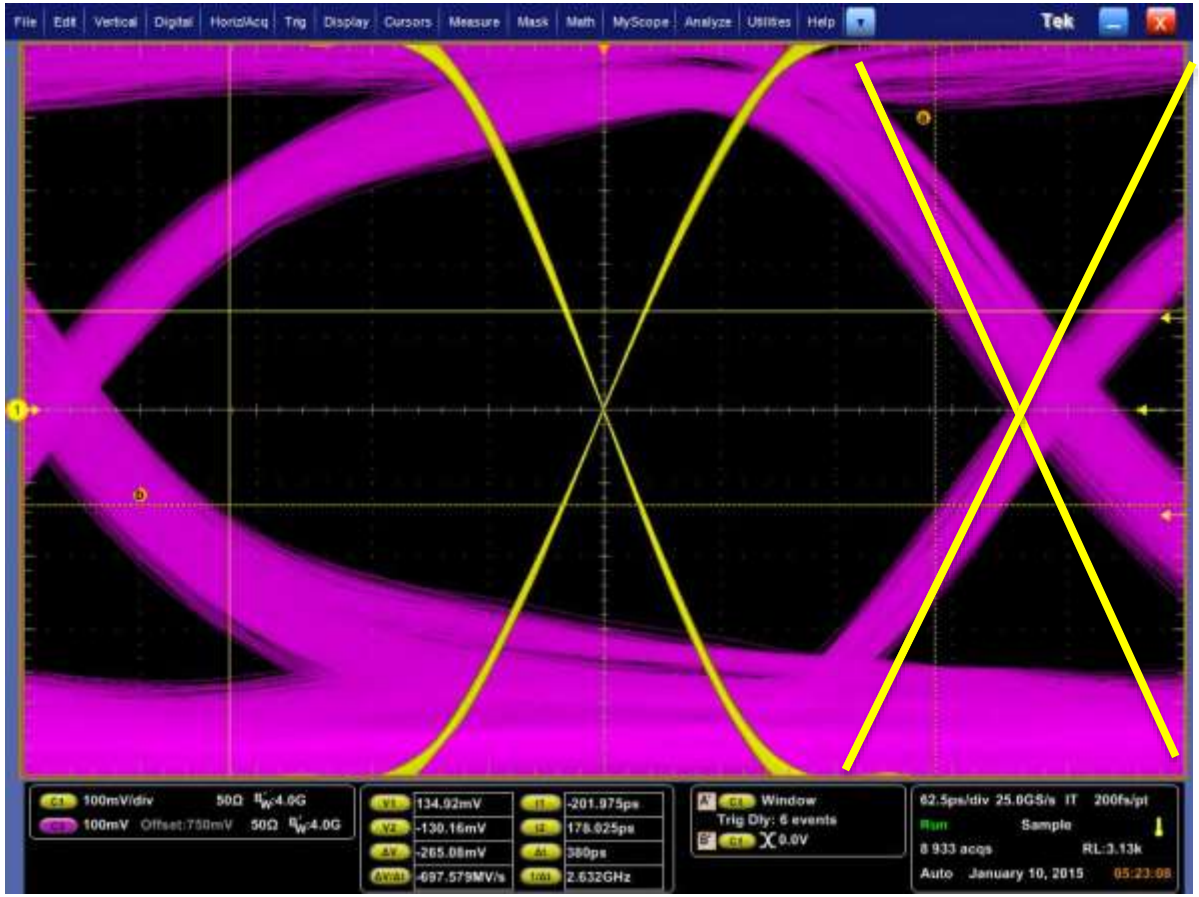
	Pass / Total	**Lane 0**	Lane 1	Lane 2	Lane 3	Lane 4	Lane 5	Lane 6	Lane 7	Lane 8 (ECC)
WRLVL_START	**1/8 clocks**	0/3	0/3	0/3	0/3	0/3	0/3	0/3	0/3	0/3
	1/4 clocks	3/3	3/3	0/3	0/3	0/3	0/3	0/3	0/3	0/3
	3/8 clocks	3/3	3/3	0/3	0/3	0/3	0/3	0/3	0/3	0/3
	1/2 clocks	3/3	3/3	3/3	3/3	0/3	0/3	0/3	0/3	0/3
	5/8 clocks	3/3	3/3	3/3	3/3	0/3	0/3	0/3	0/3	0/3
	3/4 clocks	3/3	3/3	3/3	3/3	0/3	0/3	0/3	0/3	3/3
	7/8 clocks	3/3	3/3	3/3	3/3	3/3	3/3	3/3	0/3	3/3
	1 clocks	3/3	3/3	3/3	3/3	3/3	3/3	3/3	3/3	3/3
	9/8 clocks	3/3	3/3	3/3	3/3	3/3	3/3	3/3	3/3	3/3
	5/4 clocks	0/3	0/3	3/3	3/3	3/3	3/3	3/3	3/3	3/3
	11/8 clocks	0/3	0/3	3/3	3/3	3/3	3/3	3/3	3/3	3/3
	3/2 clocks	0/3	0/3	0/3	0/3	3/3	3/3	3/3	3/3	3/3
	13/8 clocks	0/3	0/3	0/3	0/3	3/3	3/3	3/3	3/3	3/3
	7/4 clocks	0/3	0/3	0/3	0/3	3/3	3/3	3/3	3/3	0/3
	15/8 clocks	0/3	0/3	0/3	0/3	0/3	0/3	0/3	3/3	0/3
	2 clocks	0/3	0/3	0/3	0/3	0/3	0/3	0/3	0/3	0/3
	17/8 clocks	0/3	0/3	0/3	0/3	0/3	0/3	0/3	0/3	0/3
	9/4 clocks	0/3	0/3	0/3	0/3	0/3	0/3	0/3	0/3	0/3
	19/8 clocks	0/3	0/3	0/3	0/3	0/3	0/3	0/3	0/3	0/3
	5/2 clocks	0/3	0/3	0/3	0/3	0/3	0/3	0/3	0/3	0/3

Data Write Cycle



Write Data Eye on the Scope

- QCVS shifts the strobe in from right to left in small timing steps.
- At each step the a DMA write read compare test is performed and each cell is marked as pass or fail.
- This process is repeated for each byte lane.



C1 100mV/div 50Ω B_W:4.0G
C3 100mV Offset:750mV 50Ω B_W:4.0G

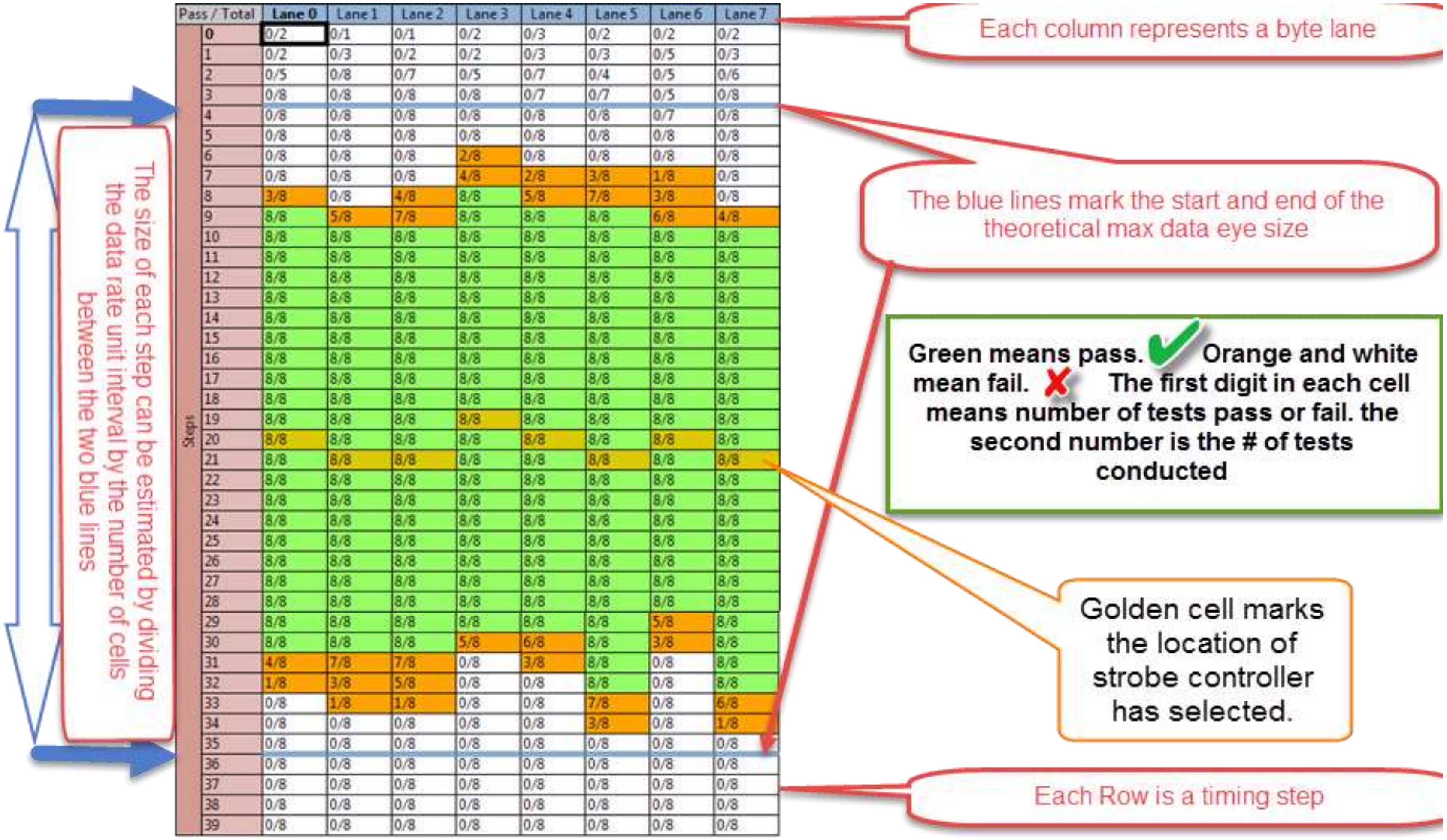
V1 134.92mV
V2 -130.16mV
ΔV -265.08mV
ΔV/Δt -697.579MV/s

t1 -201.975ps
t2 178.025ps
Δt 380ps
1/Δt 2.632GHz

A' C1 Window
Trig Dly: 6 events
B' C1 X 0.0V

62.5ps/div 25.0GS/s IT 200fs/pt
Run Sample
8 933 acqs RL:3.13k
Auto January 10, 2015 05:23:08

Write Margin Table in QCVS Tool



Pass / Total		Lane 0	Lane 1	Lane 2	Lane 3	Lane 4	Lane 5	Lane 6	Lane 7
Steps	0	0/2	0/1	0/1	0/2	0/3	0/2	0/2	0/2
	1	0/2	0/3	0/2	0/2	0/3	0/3	0/5	0/3
	2	0/5	0/8	0/7	0/5	0/7	0/4	0/5	0/6
	3	0/8	0/8	0/8	0/8	0/7	0/7	0/5	0/8
	4	0/8	0/8	0/8	0/8	0/8	0/8	0/7	0/8
	5	0/8	0/8	0/8	0/8	0/8	0/8	0/8	0/8
	6	0/8	0/8	0/8	2/8	0/8	0/8	0/8	0/8
	7	0/8	0/8	0/8	4/8	2/8	3/8	1/8	0/8
	8	3/8	0/8	4/8	8/8	5/8	7/8	3/8	0/8
	9	8/8	5/8	7/8	8/8	8/8	8/8	6/8	4/8
	10	8/8	8/8	8/8	8/8	8/8	8/8	8/8	8/8
	11	8/8	8/8	8/8	8/8	8/8	8/8	8/8	8/8
	12	8/8	8/8	8/8	8/8	8/8	8/8	8/8	8/8
	13	8/8	8/8	8/8	8/8	8/8	8/8	8/8	8/8
	14	8/8	8/8	8/8	8/8	8/8	8/8	8/8	8/8
	15	8/8	8/8	8/8	8/8	8/8	8/8	8/8	8/8
	16	8/8	8/8	8/8	8/8	8/8	8/8	8/8	8/8
	17	8/8	8/8	8/8	8/8	8/8	8/8	8/8	8/8
	18	8/8	8/8	8/8	8/8	8/8	8/8	8/8	8/8
	19	8/8	8/8	8/8	8/8	8/8	8/8	8/8	8/8
	20	8/8	8/8	8/8	8/8	8/8	8/8	8/8	8/8
	21	8/8	8/8	8/8	8/8	8/8	8/8	8/8	8/8
	22	8/8	8/8	8/8	8/8	8/8	8/8	8/8	8/8
	23	8/8	8/8	8/8	8/8	8/8	8/8	8/8	8/8
	24	8/8	8/8	8/8	8/8	8/8	8/8	8/8	8/8
	25	8/8	8/8	8/8	8/8	8/8	8/8	8/8	8/8
	26	8/8	8/8	8/8	8/8	8/8	8/8	8/8	8/8
	27	8/8	8/8	8/8	8/8	8/8	8/8	8/8	8/8
	28	8/8	8/8	8/8	8/8	8/8	8/8	8/8	8/8
	29	8/8	8/8	8/8	8/8	8/8	8/8	5/8	8/8
	30	8/8	8/8	8/8	5/8	6/8	8/8	3/8	8/8
	31	4/8	7/8	7/8	0/8	3/8	8/8	0/8	8/8
	32	1/8	3/8	5/8	0/8	0/8	8/8	0/8	8/8
	33	0/8	1/8	1/8	0/8	0/8	7/8	0/8	6/8
	34	0/8	0/8	0/8	0/8	0/8	3/8	0/8	1/8
	35	0/8	0/8	0/8	0/8	0/8	0/8	0/8	0/8
	36	0/8	0/8	0/8	0/8	0/8	0/8	0/8	0/8
	37	0/8	0/8	0/8	0/8	0/8	0/8	0/8	0/8
	38	0/8	0/8	0/8	0/8	0/8	0/8	0/8	0/8
	39	0/8	0/8	0/8	0/8	0/8	0/8	0/8	0/8

The size of each step can be estimated by dividing the data rate unit interval by the number of cells between the two blue lines

Each column represents a byte lane

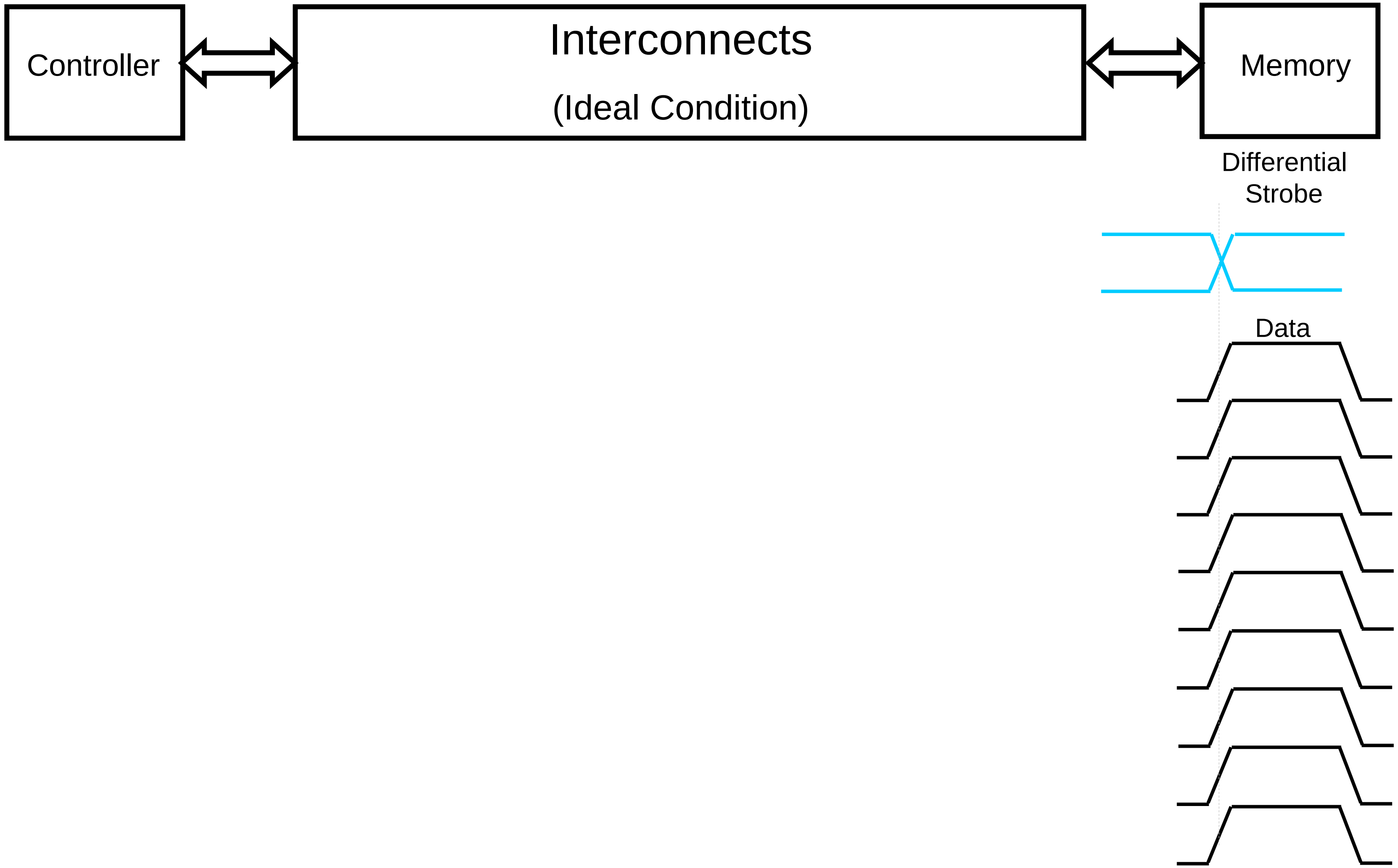
The blue lines mark the start and end of the theoretical max data eye size

Green means pass. ✓ Orange and white mean fail. ✗ The first digit in each cell means number of tests pass or fail. the second number is the # of tests conducted

Golden cell marks the location of strobe controller has selected.

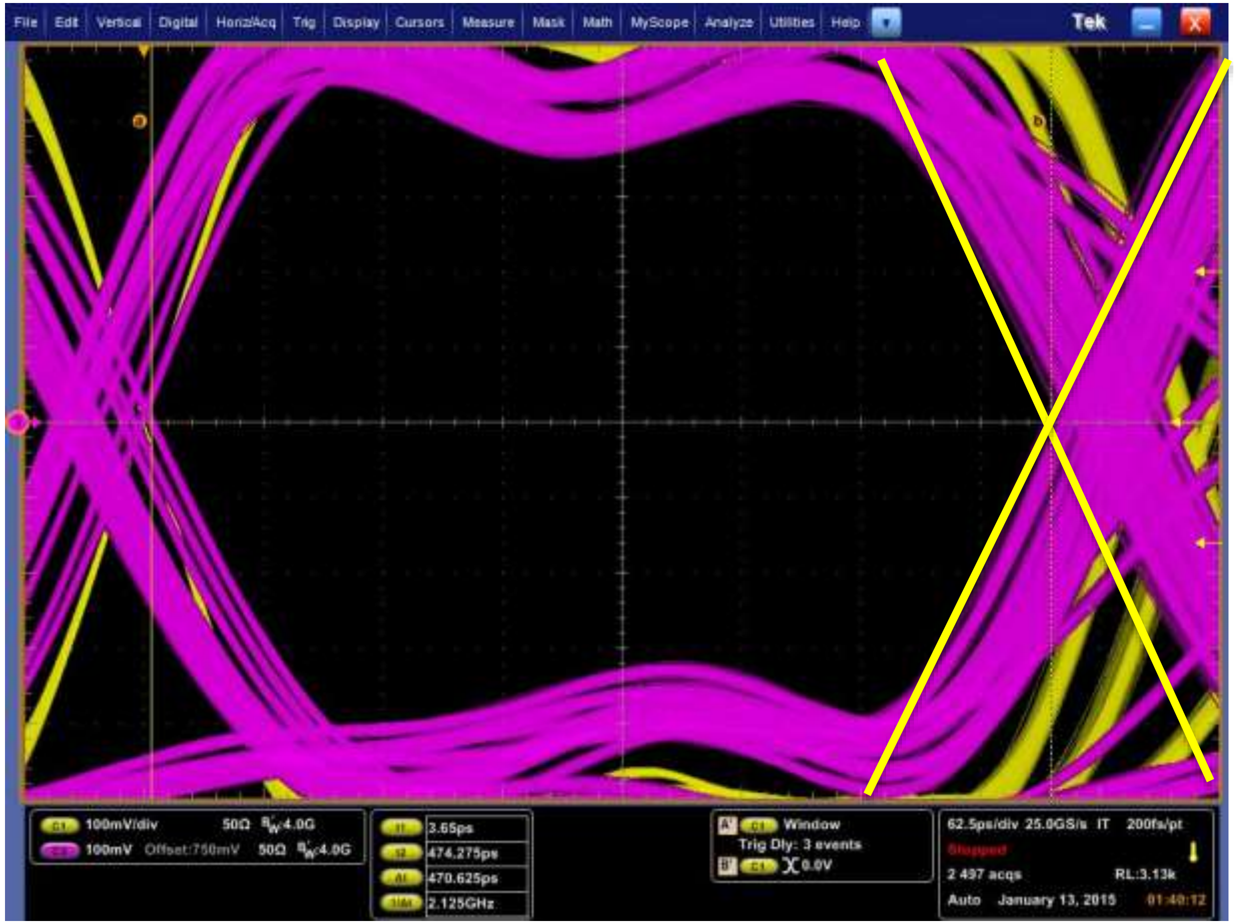
Each Row is a timing step

Data Read Cycle



Read Data Eye on the Scope

- Purple: data signal
- Yellow: strobe signal
- Probe is connected close to DRAM
- Strobe is aligned with the data eye
- Setup and hold can NOT be measured
- Approximate margin can be estimated by using a required functional mask



```
C1  100mV/div          50Ω  BW:4.0G
C2  100mV  Offset:750mV  50Ω  BW:4.0G

t1    3.65ps
t2    474.275ps
Δt    470.625ps
1/Δt  2.125GHz

A' C1  Window
   Trig Dly: 3 events
B' C1  ⵋ 0.0V

62.5ps/div 25.0GS/s  IT  200fs/pt
Stopped
2 497 acqs            RL:3.13k
Auto  January 13, 2015   01:40:12
```

Read Margin Table in QCVS Tool

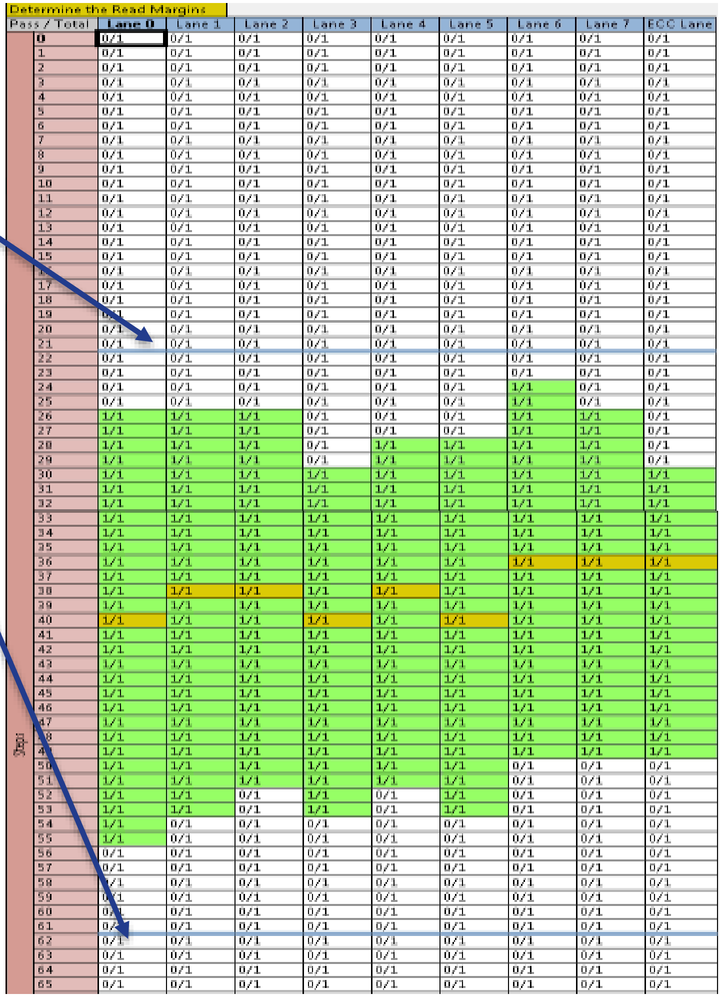
- Blue line indicates the beginning and end of the theoretical data eye
- Estimated timing for each step = theoretical-data-eye / number of steps

In this example:

Theoretical-data-eye = 536ps

Number of steps = 40

Estimated step = 13.4ps



Determine the Read Margins

Pass / Total		Lane 0	Lane 1	Lane 2	Lane 3	Lane 4	Lane 5	Lane 6	Lane 7	ECC Lane
Steps	0	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1
	1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1
	2	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1
	3	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1
	4	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1
	5	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1
	6	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1
	7	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1
	8	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1
	9	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1
	10	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1
	11	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1
	12	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1
	13	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1
	14	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1
	15	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1
	16	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1
	17	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1
	18	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1
	19	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1
	20	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1
	21	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1
	22	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1
	23	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1
	24	0/1	0/1	0/1	0/1	0/1	0/1	1/1	0/1	0/1
	25	0/1	0/1	0/1	0/1	0/1	0/1	1/1	0/1	0/1
	26	1/1	1/1	1/1	0/1	0/1	0/1	1/1	1/1	0/1
	27	1/1	1/1	1/1	0/1	0/1	0/1	1/1	1/1	0/1
	28	1/1	1/1	1/1	0/1	1/1	1/1	1/1	1/1	0/1
	29	1/1	1/1	1/1	0/1	1/1	1/1	1/1	1/1	0/1
	30	1/1	1/1	1/1	1/1	1/1	1/1	1/1	1/1	1/1
	31	1/1	1/1	1/1	1/1	1/1	1/1	1/1	1/1	1/1
	32	1/1	1/1	1/1	1/1	1/1	1/1	1/1	1/1	1/1
	33	1/1	1/1	1/1	1/1	1/1	1/1	1/1	1/1	1/1
	34	1/1	1/1	1/1	1/1	1/1	1/1	1/1	1/1	1/1
	35	1/1	1/1	1/1	1/1	1/1	1/1	1/1	1/1	1/1
	36	1/1	1/1	1/1	1/1	1/1	1/1	1/1	1/1	1/1
	37	1/1	1/1	1/1	1/1	1/1	1/1	1/1	1/1	1/1
	38	1/1	1/1	1/1	1/1	1/1	1/1	1/1	1/1	1/1
	39	1/1	1/1	1/1	1/1	1/1	1/1	1/1	1/1	1/1
	40	1/1	1/1	1/1	1/1	1/1	1/1	1/1	1/1	1/1
	41	1/1	1/1	1/1	1/1	1/1	1/1	1/1	1/1	1/1
	42	1/1	1/1	1/1	1/1	1/1	1/1	1/1	1/1	1/1
	43	1/1	1/1	1/1	1/1	1/1	1/1	1/1	1/1	1/1
	44	1/1	1/1	1/1	1/1	1/1	1/1	1/1	1/1	1/1
	45	1/1	1/1	1/1	1/1	1/1	1/1	1/1	1/1	1/1
	46	1/1	1/1	1/1	1/1	1/1	1/1	1/1	1/1	1/1
	47	1/1	1/1	1/1	1/1	1/1	1/1	1/1	1/1	1/1
	48	1/1	1/1	1/1	1/1	1/1	1/1	1/1	1/1	1/1
	49	1/1	1/1	1/1	1/1	1/1	1/1	1/1	1/1	1/1
	50	1/1	1/1	1/1	1/1	1/1	1/1	0/1	0/1	0/1
	51	1/1	1/1	1/1	1/1	1/1	1/1	0/1	0/1	0/1
	52	1/1	1/1	0/1	1/1	0/1	1/1	0/1	0/1	0/1
	53	1/1	1/1	0/1	1/1	0/1	1/1	0/1	0/1	0/1
	54	1/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1
	55	1/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1
	56	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1
	57	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1
	58	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1
	59	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1
	60	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1
	61	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1
	62	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1
	63	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1
	64	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1
	65	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1

Summary

Summary

- DDR4 is mainstream, LPDRR4 adoption is gaining momentum.
- QorIQ Layerscape and i.MX8 device families support latest DRAMs (DDR4 and LPDDR4)
- New DRAMs offer faster data rates, higher densities, reliability and low power consumption in comparison to previous generations.
- Layout guides, models and application notes are available.
- Available tools can help configure, validate and optimize complex DRAM interface to facilitate initial configuration and board bring-up.
  - i.MX Family → Spreadsheet and DRAM Stress Test program
  - Layerscape Family → Code Warrior QCVS and DDRv Applications

Useful References

- Books:
  - DRAM Circuit Design: A Tutorial, Brent Keeth and R. Jacob Baker, IEEE Press, 2001
- Freescale AppNotes:
  - AN2582 Hardware and Layout Design Considerations for DDR Memory Interfaces
  - AN2910 Hardware and Layout Design Considerations for DDR2 Memory Interfaces
  - AN2583 Programming the PowerQUICCIII / PowerQUICCII Pro DDR SDRAM Controller
  - AN3369 PowerQUICC DDR2 SDRAM Controller Register Setting Considerations
  - AN3939 PQ & QorIQ [illegible] ving
  - AN3940 Layout Design Considerations for DDR3 Memory Interface
  - AN4039 PowerQUICC DDR3 SDRAM Controller Register Setting Considerations
  - AN5097 Layout Design Considerations for DDR4 Memory Interface
- Micron AppNotes:
  - TN-41-07 DDR3 Power-Up, Initialization, and Reset
  - TN-41-08 DDR3 Design Guide
  - TN-40-03 DDR4 Design Guide
- JEDEC Specs:
  - JESD79-3F DDR3 SDRAM Specification
  - JESD79-4A DDR4 SDRAM Specification
- Tools
  - QCVS DDRV tool
  - http://www.freescale.com/webapp/sps/site/prod_summary.jsp?code=PE_QORIQ_SUITE&fsrch=1&sr=1&pageNum=1

Links to Useful DRAM Documents

- JEDEC

DDR3 Specification:

http://www.jedec.org/sites/default/files/docs/JESD79-3E.pdf

DDR3L Amendment:

http://www.jedec.org/sites/default/files/docs/JESD79-3-1_1.pdf

LPDDR3 Specification:

http://www.jedec.org/sites/default/files/docs/JESD209-3.pdf

WideIO SDR Specification:

http://www.jedec.org/sites/default/files/docs/JESD229.pdf

LPDDR2 Specification:

http://www.jedec.org/sites/default/files/docs/JESD209-2E.pdf

- Micron Documentation

DRAM Support Site:

http://www.micron.com/products/dram/ddr3-sdram#documentation_support

SECURE CONNECTIONS
FOR A SMARTER WORLD