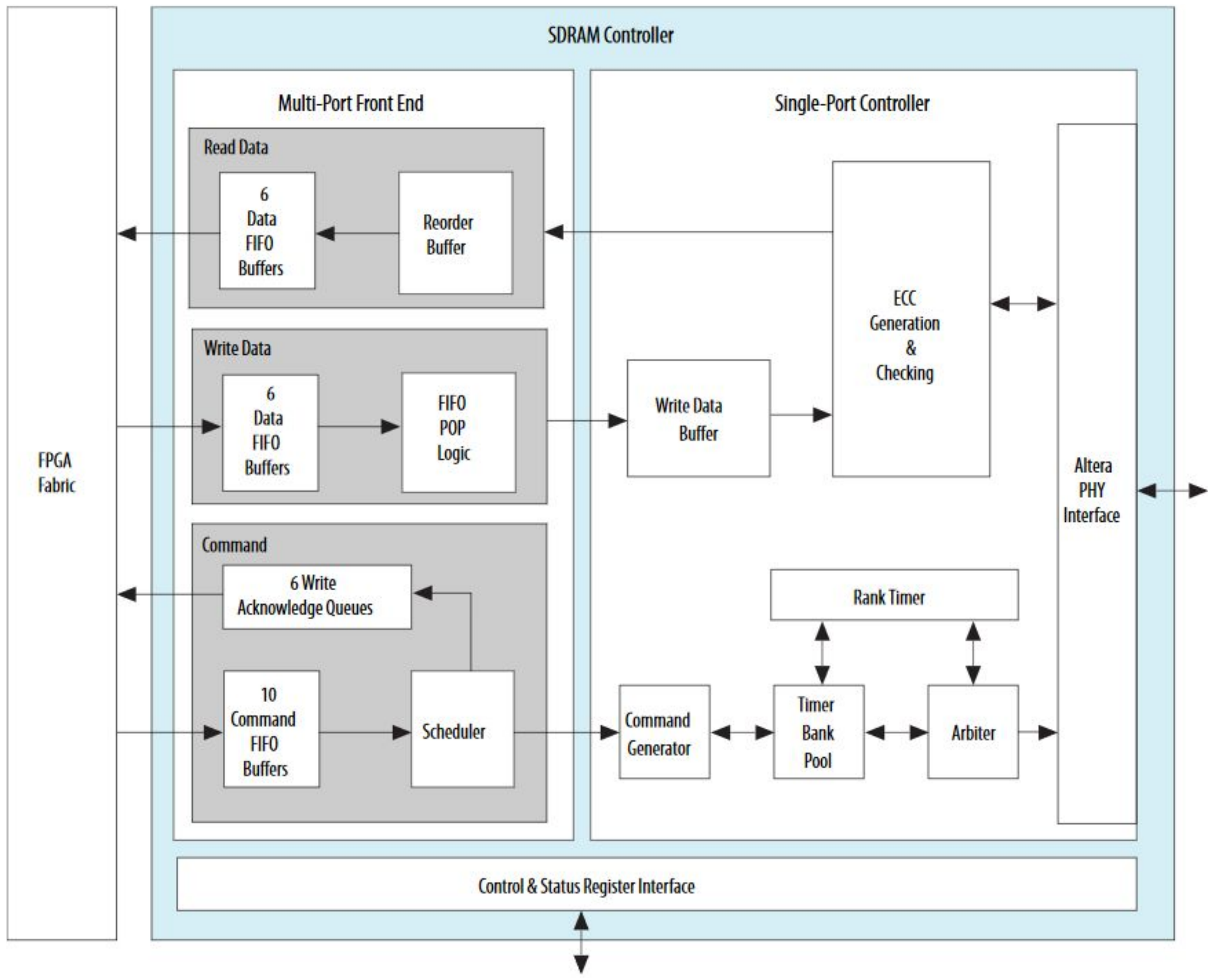
Tile Engine DDR3 SDRAM Timing:

- Very Nontrivial.
- The DDR3 we use has a programmable CAS latency (read address to output data timing in clock cycles) (read about CAS latency here [0]) (find SDRAM datasheet in CD or here [1]). We need to know what this is programmed for.



Above: Block diagram of the HPS's SDRAM Controller [2, 12-7]

Vocab:

- MPFE: Multi-Port Front End.

MPFE Multi-Port Arbitration:

- This is a "scheduler" of sorts. It is a logic block under the HPS SDRAM Controller which decides which ports get Single-Port Controller access when multiple requests are made.
- Two criteria: Priority and weight.
    - Priority: Set by programmable registers. Higher priority ports always win over lower priority ports. Ports with the same priority have weighted round-robin access.

- Weight: Set by programmable registers. Higher weight has more time/representation in round-robin.
- By setting static weights correctly, you effectively allocate a percentage of bandwidth to a particular port.

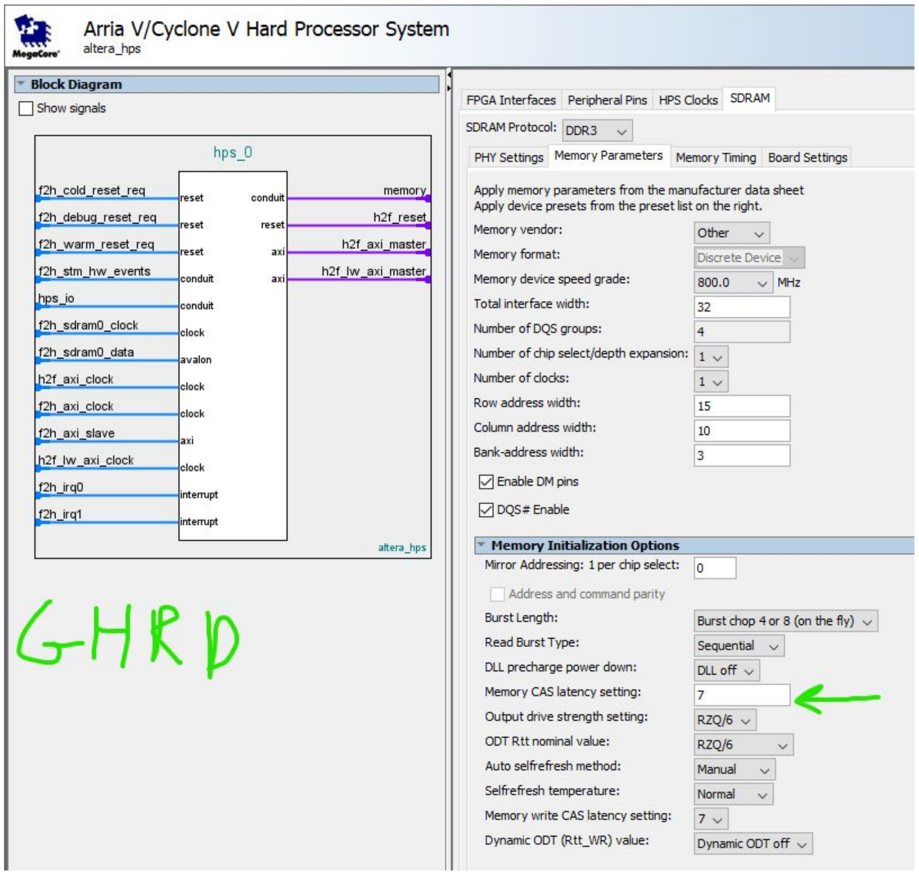
Set all sources to the same priority and same weight. The worst case is the following sequence of commands **(assuming command FIFO is empty)** executed all at once:

- HPS read
- HPS write
- PPU - BG Tile Engine read
- PPU - FG Tile Engine read
- APU read

What is the size of the FIFOs? Need to know how many can commands can be queued up to calculate actual worst case read latency.

Notes 3-10-21:

- GHRD uses 400MHz clock for SDRAM



Arria V/Cyclone V Hard Processor System
altera_hps

FPGA Interfaces | Peripheral Pins | HPS Clocks | SDRAM

SDRAM Protocol: DDR3

PHY Settings | Memory Parameters | Memory Timing | Board Settings

Apply memory parameters from the manufacturer data sheet
Apply device presets from the preset list on the right.

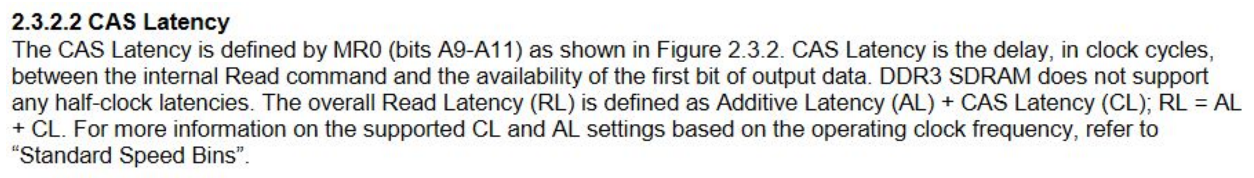
Memory vendor:	Other
Memory format:	Discrete Device
Memory device speed grade:	800.0 MHz
Total interface width:	32
Number of DQS groups:	4
Number of chip select/depth expansion:	1
Number of clocks:	1
Row address width:	15
Column address width:	10
Bank-address width:	3

☑ Enable DM pins
☑ DQS# Enable

Memory Initialization Options

Mirror Addressing: 1 per chip select:	0
☐ Address and command parity	
Burst Length:	Burst chop 4 or 8 (on the fly)
Read Burst Type:	Sequential
DLL precharge power down:	DLL off
Memory CAS latency setting:	7
Output drive strength setting:	RZQ/6
ODT Rtt nominal value:	RZQ/6
Auto selfrefresh method:	Manual
Selfrefresh temperature:	Normal
Memory write CAS latency setting:	7
Dynamic ODT (Rtt_WR) value:	Dynamic ODT off

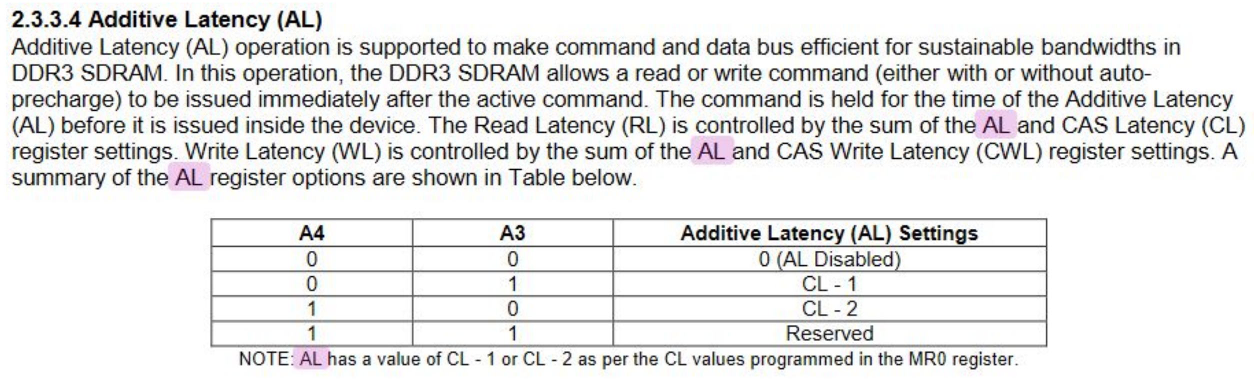
- Notably, the GHRD is configured for a memory CAS latency of 7 cycles.



2.3.2.2 CAS Latency

The CAS Latency is defined by MR0 (bits A9-A11) as shown in Figure 2.3.2. CAS Latency is the delay, in clock cycles, between the internal Read command and the availability of the first bit of output data. DDR3 SDRAM does not support any half-clock latencies. The overall Read Latency (RL) is defined as Additive Latency (AL) + CAS Latency (CL); RL = AL + CL. For more information on the supported CL and AL settings based on the operating clock frequency, refer to "Standard Speed Bins".

Above: Source is [1, pg. 12].



2.3.3.4 Additive Latency (AL)

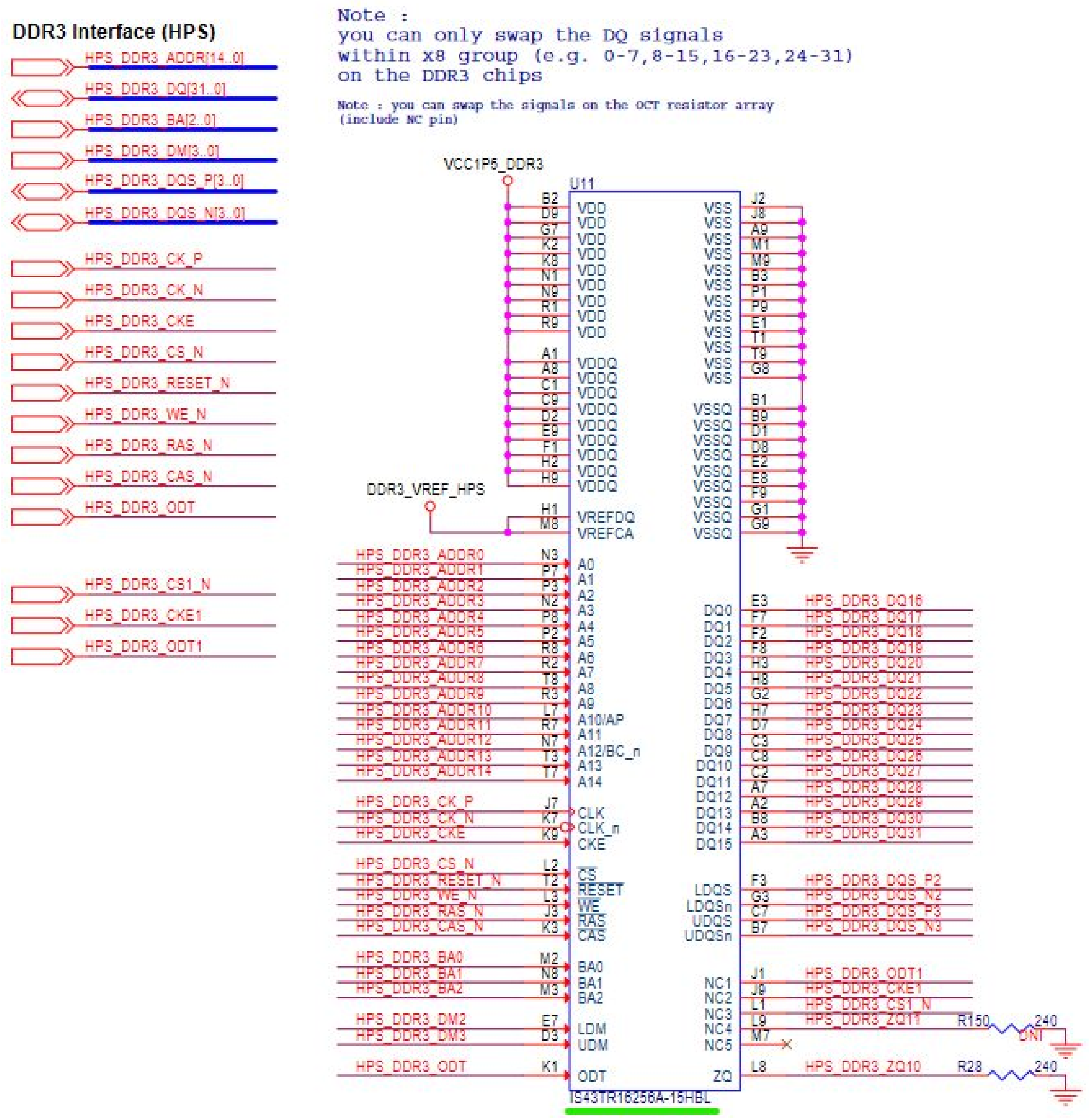
Additive Latency (AL) operation is supported to make command and data bus efficient for sustainable bandwidths in DDR3 SDRAM. In this operation, the DDR3 SDRAM allows a read or write command (either with or without auto-precharge) to be issued immediately after the active command. The command is held for the time of the Additive Latency (AL) before it is issued inside the device. The Read Latency (RL) is controlled by the sum of the AL and CAS Latency (CL) register settings. Write Latency (WL) is controlled by the sum of the AL and CAS Write Latency (CWL) register settings. A summary of the AL register options are shown in Table below.

A4	A3	Additive Latency (AL) Settings
0	0	0 (AL Disabled)
0	1	CL - 1
1	0	CL - 2
1	1	Reserved

NOTE: AL has a value of CL - 1 or CL - 2 as per the CL values programmed in the MR0 register.

Above: Source is [1, pg. 14].

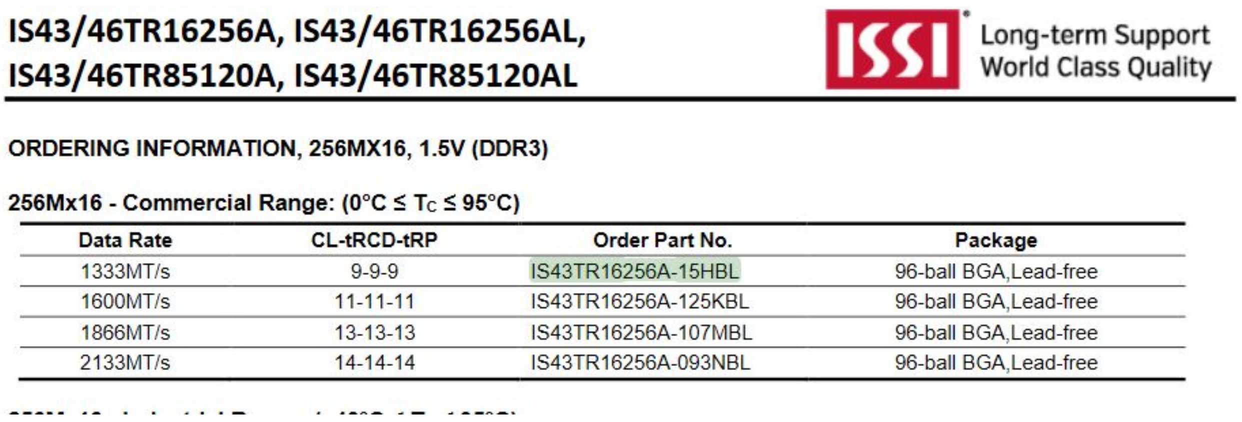
- Need to find the corresponding AL for a CL of 7. The speed bins are organized by the exact chip number, but the datasheet is for a collection of chips. Time to consult the schematics (or physical board) to find out the exact chip!
- Worst case would be AL = CL-1. Then total Read Latency RL (HPS_command->SDRAM->HPS_read) = CL + CL - 1 = 2CL - 1. With CL = 7 this means RL = 13.
- Best case would be AL = CL + 0 = CL = 7.
- Note that AL is set somewhere. According to [1, pg. 9], the control registers in our SDRAM have no default values and must be fully initialized after power-up/rst.


Note :
you can only swap the DQ signals
within x8 group (e.g. 0-7,8-15,16-23,24-31)
on the DDR3 chips

Note : you can swap the signals on the OCT resistor array
(include NC pin)

Above: Schematic from DE10-Nano Online CD.

- The chip number is IS43TR16256A-15HBL.


IS43/46TR16256A, IS43/46TR16256AL, IS43/46TR85120A, IS43/46TR85120AL

Long-term Support
World Class Quality

ORDERING INFORMATION, 256MX16, 1.5V (DDR3)

256Mx16 - Commercial Range: (0°C ≤ T_C ≤ 95°C)

Data Rate	CL-tRCD-tRP	Order Part No.	Package
1333MT/s	9-9-9	IS43TR16256A-15HBL	96-ball BGA,Lead-free
1600MT/s	11-11-11	IS43TR16256A-125KBL	96-ball BGA,Lead-free
1866MT/s	13-13-13	IS43TR16256A-107MBL	96-ball BGA,Lead-free
2133MT/s	14-14-14	IS43TR16256A-093NBL	96-ball BGA,Lead-free

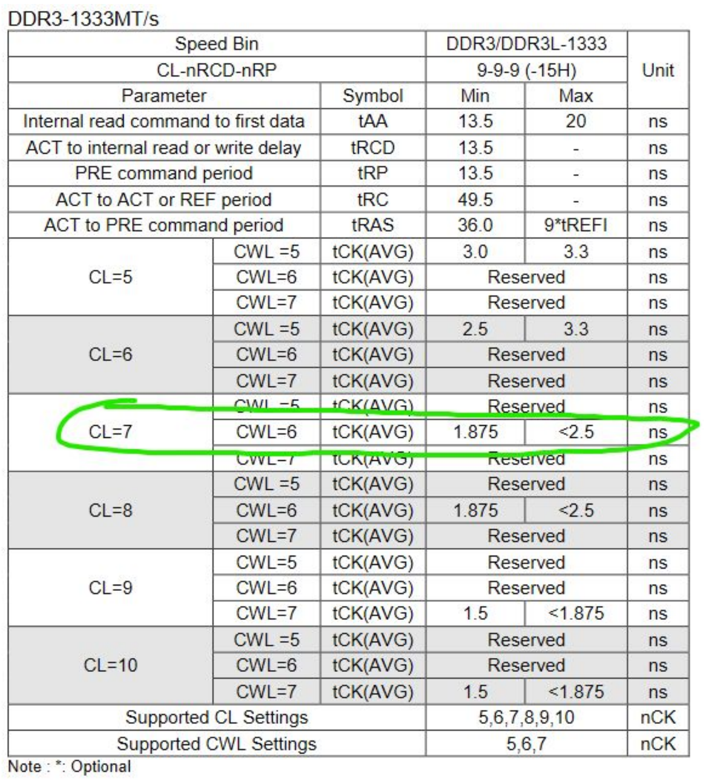
Above: [1, pg. 82]

- The speed bins refer to this part as 1333MT/s

It turns out not very many embedded memory controllers support the Additive Latency feature and automatically program this value to be 0.
https://e2e.ti.com/support/processors/f/processors-forum/442397/additive-latency-value-for-ddr3-memory

- The portion of the Cyclone V that controls this is the DDR PHY [2, 12-25]. The DDR PHY handles memory device initialization and reinitialization after reset.
- Based on the lack of information here, I think I could assume that DDR PHY doesn't support the Additive Latency Feature. The worst case RL = 7.

Memory Clock Speed and Can I Lower CL?


DDR3-1333MT/s

Speed Bin			DDR3/DDR3L-1333		Unit
CL-nRCD-nRP			9-9-9 (-15H)		
Parameter		Symbol	Min	Max	
Internal read command to first data		tAA	13.5	20	ns
ACT to internal read or write delay		tRCD	13.5	-	ns
PRE command period		tRP	13.5	-	ns
ACT to ACT or REF period		tRC	49.5	-	ns
ACT to PRE command period		tRAS	36.0	9*tREFI	ns
CL=5	CWL =5	tCK(AVG)	3.0	3.3	ns
	CWL=6	tCK(AVG)	Reserved		ns
	CWL=7	tCK(AVG)	Reserved		ns
CL=6	CWL =5	tCK(AVG)	2.5	3.3	ns
	CWL=6	tCK(AVG)	Reserved		ns
	CWL=7	tCK(AVG)	Reserved		ns
CL=7	CWL =5	tCK(AVG)	Reserved		ns
	CWL=6	tCK(AVG)	1.875	<2.5	ns
	CWL=7	tCK(AVG)	Reserved		ns
CL=8	CWL =5	tCK(AVG)	Reserved		ns
	CWL=6	tCK(AVG)	1.875	<2.5	ns
	CWL=7	tCK(AVG)	Reserved		ns
CL=9	CWL=5	tCK(AVG)	Reserved		ns
	CWL=6	tCK(AVG)	Reserved		ns
	CWL=7	tCK(AVG)	1.5	<1.875	ns
CL=10	CWL =5	tCK(AVG)	Reserved		ns
	CWL=6	tCK(AVG)	Reserved		ns
	CWL=7	tCK(AVG)	1.5	<1.875	ns
Supported CL Settings			5,6,7,8,9,10		nCK
Supported CWL Settings			5,6,7		nCK

Note : *: Optional

Above: Speed bin for our SDRAM chip [1, pg. 56]. Contains recommended clock rates for a given CL value. I've circled the GHRD in green.

- 400MHz corresponds to the upper bound tCK < 2.5 (clock period average) for CL=7.
- This tells me that we should choose a clock rate which matches the MAX TCK(AVG) period recommended by the speed bin.

How to read memory specs:
The spec says "9-9-9 (-15H)". What does this mean?
The first column is CAS latency (CL): 9
The second column is RAS to CAS delay: nRCD
The third column is row precharge time: nRP
See [3] for more information.

The actual metric most people use to determines memory speed is the tCLK * CL ~ the actual delay between read command and response data.

CL	tCK(AVG) (min)	tCK(AVG) (MAX)	Min Freq.	Max Freq.	Min Response	Max Response
5	3	3.3	0.303030303	0.3333333333	15	16.5
6	2.5	3.3	0.303030303	0.4	15	19.8
7	1.875	2.5	0.4	0.5333333333	13.125	17.5
8	1.875	2.5	0.4	0.5333333333	15	20
9	1.5	1.875	0.5333333333	0.6666666667	13.5	16.875
10	1.5	1.875	0.5333333333	0.6666666667	15	18.75

Above: Calculations of Min. Freq., Max Freq., Min. Response, and Max Response from data in [1].

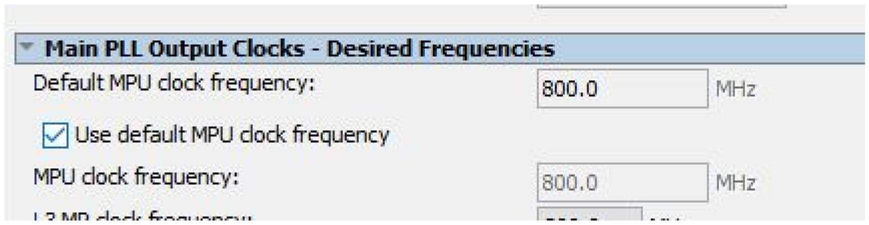
- Clearly, the best setting for lowest read latency is CL=7 at 400MHz. The GHRD makes sense.

TLDR 3/10/21 (Joe rambles too much):

- The read latency from the PHY Interface (SDRAM Controller) to the SDRAM is 7 clock cycles, with a clock rate of 400MHz. This should be treated as the theoretical best-case (likely unreachable) latency for our system.
- The next piece of research is to determine:
  - Worst-case travel time for read-command -> SDRAM controller subsystem (across h2f_sdram bridge)
  - Worst-case travel time for read-command in SDRAM controller subsystem to be output to the actual SDRAM (through any FIFOs/buffers/scheduling/etc).

    ○ Worst-case travel time for read data in SDRAM controller subsystem to be sent back to the PPU/APU (through h2f_sdram and any FIFOs/buffers/etc.)

Notes 3/12/21:



Main PLL Output Clocks - Desired Frequencies

Default MPU clock frequency: 800.0 MHz

☑ Use default MPU clock frequency

MPU clock frequency: 800.0 MHz

- MPU clocked at 800 MHz. I'm going to be using this in my assumptions about the latency due to clocked logic (if any) inside of the HPS's SDRAM controller.

To compute an upper bound on the time between sending the read command from the PPU/APU and receiving the data, I have to make some assumptions and simplifications:

1. The "10 Command FIFO Buffers" simply store 10 commands.
    a. There is no "First-in-First-out" guarantee, since the scheduler+arbiter picks which operations happen in which order.
    b. Worst case, your command is received by the SDRAM controller along with 9 others in parallel - and your command is sent to the SDRAM chip last.
2. The latency from the PPU/APU Avalon Master sending a command to the SDRAM controller receiving that command is 1 clock cycle. Similarly, for SDRAM data to PPU/APU Avalon Master.
    a. In other words, any unknown bus element delays add up to 1 clock cycle.
3. The latency from the SDRAM controller receiving a command to sending one (not necessarily the original command) is negligible.
    a. i.e., scheduler and arbiter take much less than 16 cycles to decide which command is getting sent to the SDRAM.
    b. This is reasonable since our MPU is clocked at 800MHz, which is 16x our FPGA's clock frequency. 16 clock cycles seems like enough time to do any scheduling/arbiting before sending the command out to SDRAM. It also seems possible that these scheduler and arbiter elements are implemented in combinational logic (requiring 0 clock cycles if fast enough). The Cyclone V manual does not give any information about the specific implementation or timing, however.
4. The latency from the SDRAM controller sending a command and receiving the data is 7 clock cycles at 400MHz.
    a. This takes almost 1 clock cycle equivalent on our 50MHz FPGA (400 MHz achieves 8 clock cycles per one 50 MHz clock cycle).
5. The latency from the SDRAM controller receiving data and sending it back out is negligible. This stems from the fact that the MPU is clocked at 800MHz. Definitely enough time to do any logic and reordering before sending the data back out on the bus.

With these assumptions, the calculation for worst-case latency is as follows:

- PPU logic will set addr and set read signal high on the Avalon-MM read-only master. 1 clock cycle later, this info is sent across the bus.
- Another clock cycle later, the address and read command is received by the SDRAM controller.
- 9 other command signals were received by the SDRAM at the same time (or are in the FIFOs). These 9 commands take 9 50MHz equivalent clock cycles before our command is sent to the SDRAM.
- Our read command is sent to the SDRAM. The SDRAM controller receives the data back one 50MHz-equivalent clock cycle later, and sends it across the bus.
- It takes one more clock cycle for read data to appear on the Avalon-MM Read-only master due to any bus delay element.

In total, there were 1+1+9+1+1 = 13 clock cycles in our worst case scenario.

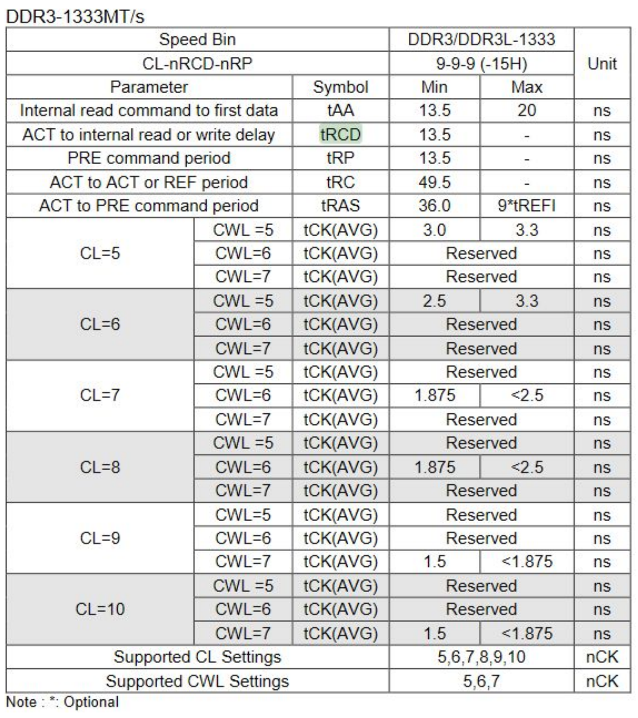
- Sounds reasonable given this source about the DE2 (Cyclone IV's) worst case read latency
- https://community.intel.com/t5/Intel-FPGA-University-Program/my-SDRAM-takes-14-clock-cycles-to-read-data/td-p/68175

Notes 3/13/21:

I forgot one aspect of the timing calculation. The actual worst case scenario is much worse, since CAS timings assume the same row of data is being accessed each read. This will likely not be true for non-burst reads.

This means the worst-case latency of 13 clock cycles is only achievable in burst-read scenarios where accesses have locality.

When the accesses are not in the same row, the actual command->data latency is RAS->CAS latency + CAS latency. RAS to CAS timing (also known as tRCD) of our chip can be found in the table below for our chip's speed bin:



DDR3-1333MT/s

Speed Bin			DDR3/DDR3L-1333		Unit
CL-nRCD-nRP			9-9-9 (-15H)		
Parameter		Symbol	Min	Max	
Internal read command to first data		tAA	13.5	20	ns
ACT to internal read or write delay		tRCD	13.5	-	ns
PRE command period		tRP	13.5	-	ns
ACT to ACT or REF period		tRC	49.5	-	ns
ACT to PRE command period		tRAS	36.0	9*tREFI	ns
CL=5	CWL =5	tCK(AVG)	3.0	3.3	ns
	CWL=6	tCK(AVG)	Reserved		ns
	CWL=7	tCK(AVG)	Reserved		ns
CL=6	CWL =5	tCK(AVG)	2.5	3.3	ns
	CWL=6	tCK(AVG)	Reserved		ns
	CWL=7	tCK(AVG)	Reserved		ns
CL=7	CWL =5	tCK(AVG)	Reserved		ns
	CWL=6	tCK(AVG)	1.875	<2.5	ns
	CWL=7	tCK(AVG)	Reserved		ns
CL=8	CWL =5	tCK(AVG)	Reserved		ns
	CWL=6	tCK(AVG)	1.875	<2.5	ns
	CWL=7	tCK(AVG)	Reserved		ns
CL=9	CWL=5	tCK(AVG)	Reserved		ns
	CWL=6	tCK(AVG)	Reserved		ns
	CWL=7	tCK(AVG)	1.5	<1.875	ns
CL=10	CWL =5	tCK(AVG)	Reserved		ns
	CWL=6	tCK(AVG)	Reserved		ns
	CWL=7	tCK(AVG)	1.5	<1.875	ns
Supported CL Settings			5,6,7,8,9,10		nCK
Supported CWL Settings			5,6,7		nCK

Note : *: Optional

Above: Table for our SDRAM chip from [1].

tRCD = 13.5 ns or just under 20ns (the length of one 50MHz clock cycle). If we round up to one 50MHz clock cycle, this changes our worst case calculation to:

- PPU->BUS: 1 clk
- BUS->SDRAM Controller: 1 clk
- 9 other read commands SDRAM Controller -> SDRAM: 2 clk each for a total of 18 clk
- Our read command SDRAM Controller -> SDRAM -> SDRAM Controller total of 2 clk
- BUS->PPU: 1 clk

Total: 1 + 1 + 18 + 2 + 1 = 23 50MHz-equivalent clock cycles.

Resources:

[0] https://en.wikipedia.org/wiki/CAS_latency
[1] DDR3 Chip the DE10-Nano is using
http://www.issi.com/ww/pdf/43-46tr16256a-85120al.pdf
[2] Cyclone V SOC pdf
https://www.intel.com/content/dam/www/programmable/us/en/pdfs/literature/hb/cyclone-v/cv_5v4.pdf
[3] Explanation of Memory Timing terms. Helpful when reading a datasheet for SDRAM.
https://en.wikipedia.org/wiki/Memory_timings