20 YEARS of

ALTERA®

INNOVATION

# SDRAM PLL Tuning

# SDRAM PLL Tuning

- Objective: Find the Correct Skew Needed for the SDRAM Clock with Respect to the System Clock
- Two Methods:
  - "Scientific" vs. "Trial and Error"

# Design Recommendations

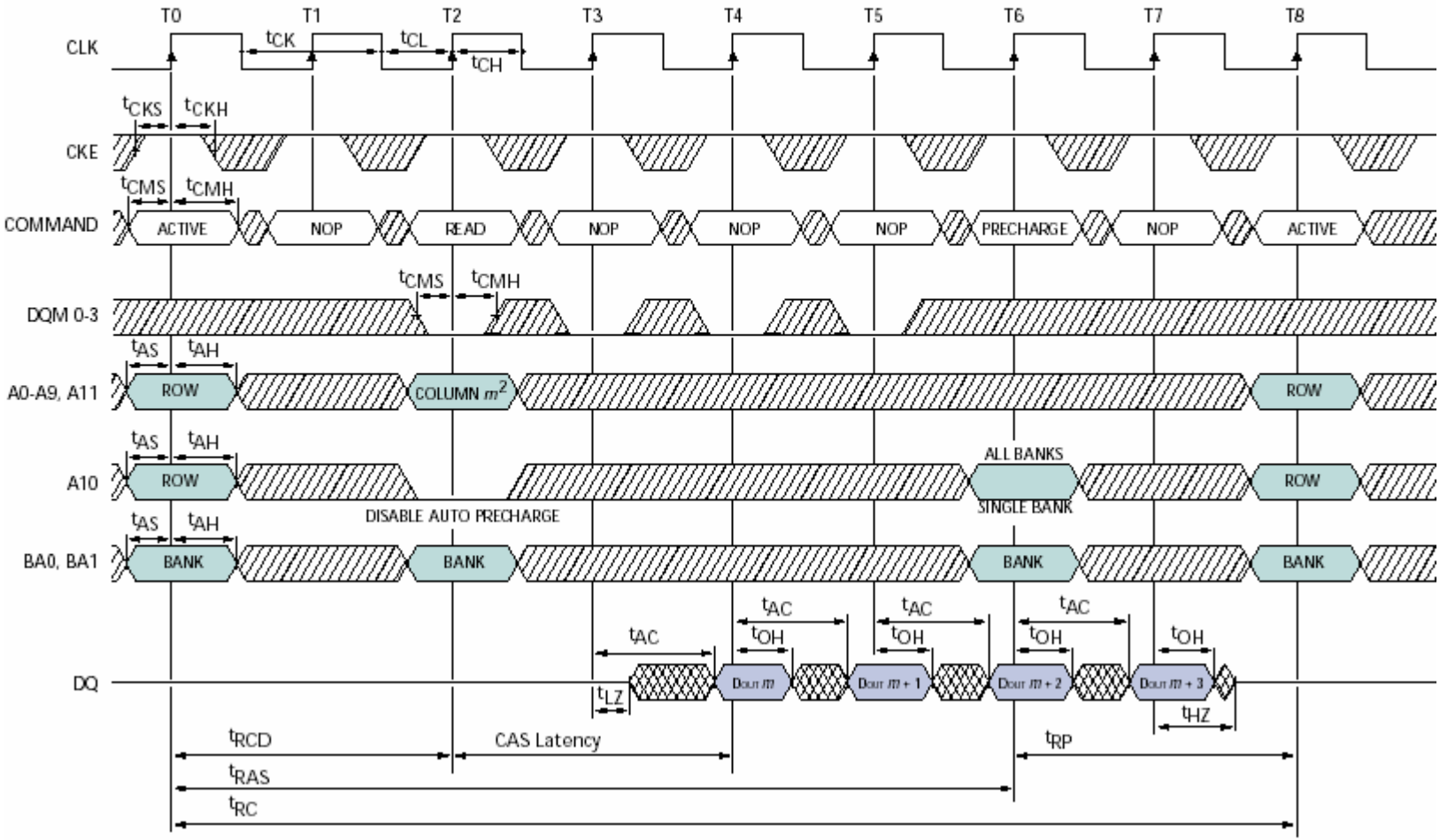
- Use “Zero Delay Buffer” Mode of the PLL
  - Allows Control of Phase of External Clock with Respect to the Input Clock
- SDRAM Clock Output of PLL (e0)
- SDRAM Pins:
  - Use Fast I/O’s Settings
  - Group Together to Minimize Skew

# First, Read the Datasheets

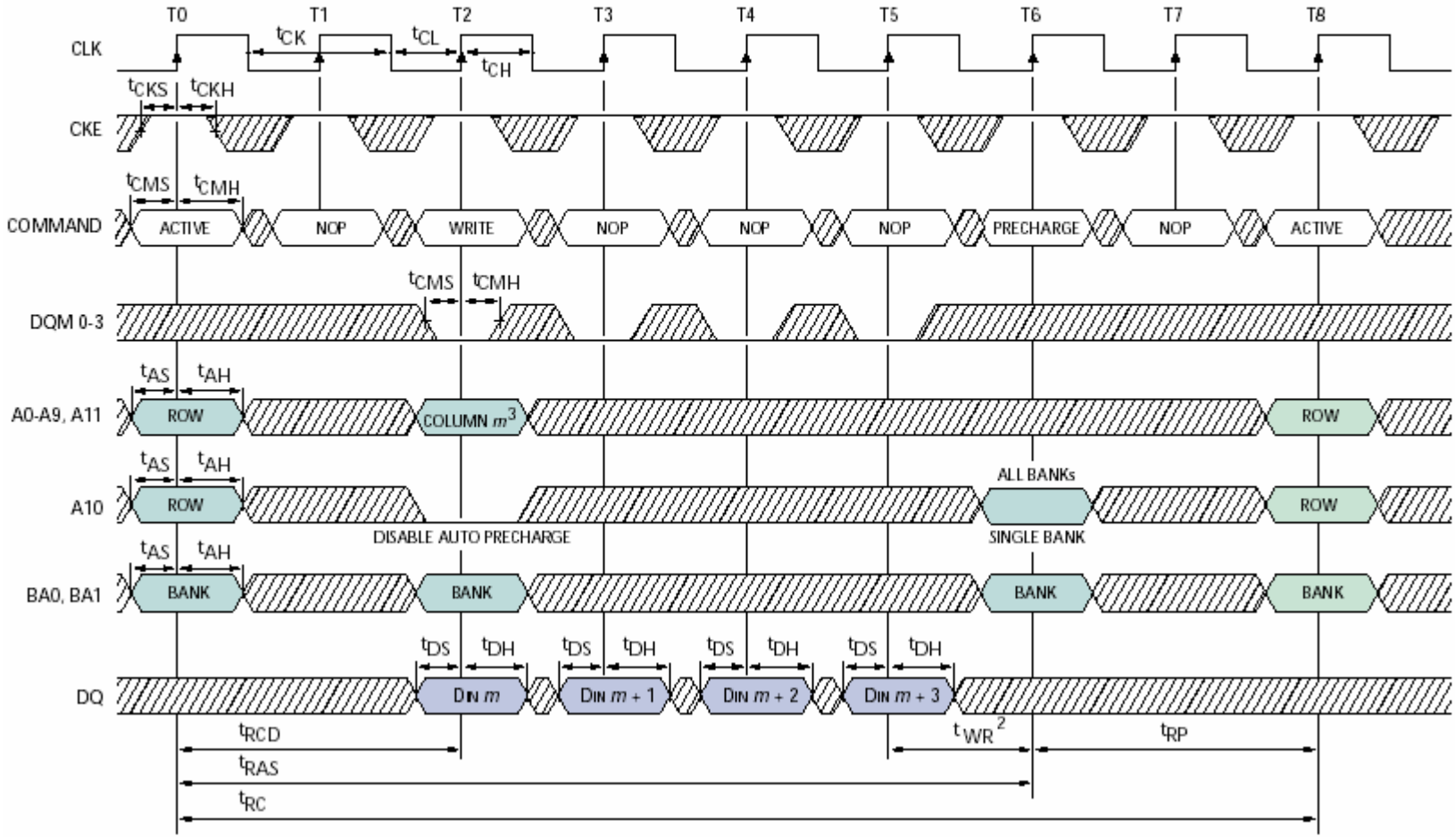
- SDRAM (MT48LC4M32B2-7)
  - Data In: $t_{su}$ = 2 ns, $t_h$ = 1 ns
  - Data Out: $t_{oh}$ = 2.5 ns, $t_{hz/tac}$ = 5.5 ns (CL=3)
    - 2.5 – 5.5 ns (Data Undefined)

| AC CHARACTERISTICS | | | -6 | | -7 | | | |
|---|---|---|---|---|---|---|---|---|
| PARAMETER | | SYMBOL | MIN | MAX | MIN | MAX | UNITS | NOTES |
| Access time from CLK (pos. edge) | CL = 3 | tAC (3) | | 5.5 | | 5.5 | ns | |
| | CL = 2 | tAC (2) | | 7.5 | | 8 | ns | |
| | CL = 1 | tAC (1) | | 17 | | 17 | ns | |
| Address hold time | | tAH | 1 | | 1 | | ns | |
| Address setup time | | tAS | 1.5 | | 2 | | ns | |
| CLK high-level width | | tCH | 2.5 | | 2.75 | | ns | |
| CLK low-level width | | tCL | 2.5 | | 2.75 | | ns | |
| Clock cycle time | CL = 3 | tCK (3) | 6 | | 7 | | ns | 23 |
| | CL = 2 | tCK (2) | 10 | | 10 | | ns | 23 |
| | CL = 1 | tCK (1) | 20 | | 20 | | ns | 23 |
| CKE hold time | | tCKH | 1 | | 1 | | ns | |
| CKE setup time | | tCKS | 1.5 | | 2 | | ns | |
| CS#, RAS#, CAS#, WE#, DQM hold time | | tCMH | 1 | | 1 | | ns | |
| CS#, RAS#, CAS#, WE#, DQM setup time | | tCMS | 1.5 | | 2 | | ns | |
| Data-in hold time | | tDH | 1 | | 1 | | ns | |
| Data-in setup time | | tDS | 1.5 | | 2 | | ns | |
| Data-out high-impedance time | CL = 3 | tHZ (3) | | 5.5 | | 5.5 | ns | 10 |
| | CL = 2 | tHZ (2) | | 7.5 | | 8 | ns | 10 |
| | CL = 1 | tHZ (1) | | 17 | | 17 | ns | 10 |
| Data-out low-impedance time | | tLZ | 1 | | 1 | | ns | |
| Data-out hold time | | tOH | 2 | | 2.5 | | ns | |

# READ – WITHOUT AUTO PRECHARGE[1]



## WRITE – WITHOUT AUTO PRECHARGE[1]



# Cyclone Parameters

- FPGA (Cyclone 1C20-7)
  - Column I/O's and Global Clock
  - Data In: $t_{su}$ = 2.4 ns, $t_h$ = 0 ns
  - Data Out: $t_{outco}$ = 2 ns (min) 4.4 ns (max)
    - 2.0 - 4.4 ns (Data Undefined)

*Table 4–36. EP1C20 Column Pin Global Clock External I/O Timing Parameters*

| Symbol | -6 Speed Grade | | -7 Speed Grade | | -8 Speed Grade | | Unit |
|---|---|---|---|---|---|---|---|
| | Min | Max | Min | Max | Min | Max | |
| $t_{INSU}$ | 2.226 | | 2.406 | | 2.585 | | ns |
| $t_{INH}$ | 0.000 | | 0.000 | | 0.000 | | ns |
| $t_{OUTCO}$ | 2.000 | 3.926 | 2.000 | 4.358 | 2.000 | 4.795 | ns |

# Calculate the Window - Cyclone

- SDRAM Clock Can Lead System Clock by:

  Minimum of:

  $t_{coutmin(FPGA)} - t_{h(SDRAM)} = 2\text{ ns} - 1\text{ ns} =$ **1 ns**

  $t_{clk} - t_{hz(SDRAM)} - t_{su(FPGA)} = 10\text{ ns} - 5.5\text{ ns} - 2.4\text{ ns} = 2.1\text{ ns}$

- SDRAM Clock Can Lag System Clock by:

  Minimum of:

  $t_{oh(SDRAM)} - t_{h(FPGA)} = 2.5\text{ ns} - 0\text{ ns} =$ **2.5 ns**

  $t_{clk} - t_{coutmax(FPGA)} - t_{su(SDRAM)} = 10\text{ ns} - 4.4\text{ ns} - 2\text{ ns} = 3.6\text{ ns}$

- Window Between +1 ns to – 2.5 ns

# Stratix Parameters

- FPGA (Stratix 1S10-6)
  - Column I/O's and Global Clock
  - Data In: $t_{su}$ = 1.75 ns, $t_h$ = 0 ns
  - Data Out: $t_{outco}$ = 2 ns(min) 5.5 ns(max)
    - 2.0 - 5.5 ns (Data Undefined)

*Table 4–54. EP1S10 Column Pin Global Clock External I/O Timing Parameters*

| Symbol | -5 Speed Grade | | -6 Speed Grade | | -7 Speed Grade | | Unit |
|---|---|---|---|---|---|---|---|
| | Min | Max | Min | Max | Min | Max | |
| $t_{INSU}$ | 1.699 | | 1.748 | | 1.993 | | ns |
| $t_{INH}$ | 0.000 | | 0.000 | | 0.000 | | ns |
| $t_{OUTCO}$ | 2.000 | 5.143 | 2.000 | 5.504 | 2.000 | 6.308 | ns |

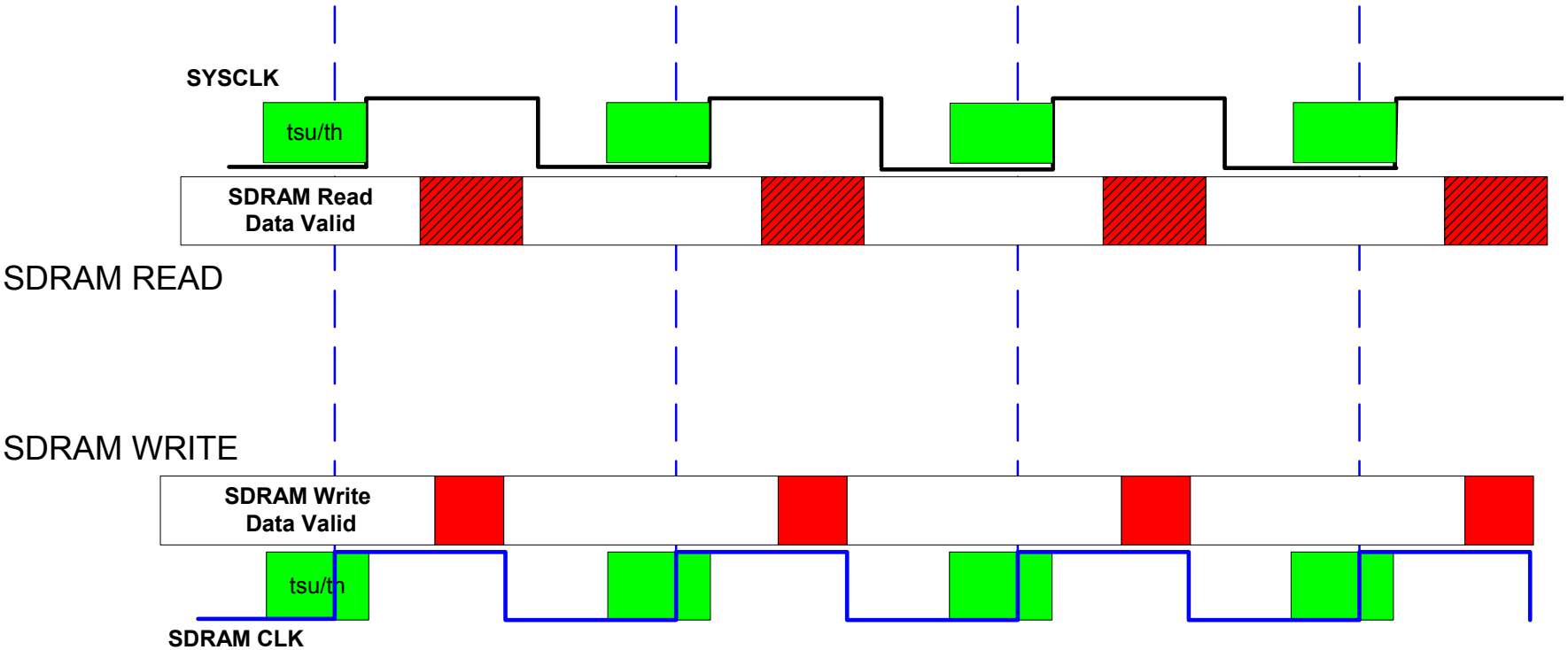
# Calculate the Window - Stratix

- SDRAM Clock Can Lead System Clock by:
  Minimum of:
  $t_{coutmin(FPGA)} - t_{h(SDRAM)} = 2\text{ ns} - 1\text{ ns} =$ **1 ns**
  $t_{clk} - t_{hz(SDRAM)} - t_{su(FPGA)} = 10\text{ns} - 5.5\text{ ns} - 1.75\text{ ns} = 2.75\text{ ns}$
- SDRAM Clock Can Lag System Clock by:
  Minimum of:
  $t_{oh(SDRAM)} - t_{h(FPGA)} = 2.5\text{ ns} - 0\text{ ns} =$ **2.5 ns**
  $t_{clk} - t_{coutmax(FPGA)} - t_{su(SDRAM)} = 10\text{ ns} - 5.5\text{ ns} - 2\text{ ns} = 2.5\text{ ns}$
- Window = +1 ns to – 2.5 ns
  - Same As Cyclone at 100 MHz
  - Windows Different at Higher Frequencies

# PLL Tuning

- Only a 3.5 ns Window (at 100 MHz)
  - +1 ns to -2.5 ns
- Center the Phase Shift in the Middle of Window
  - -0.75 ns Phase Shift
- Window Can Change, Dependent On:
  - SDRAM
  - SDRAM CAS Latency
  - FPGA Device:
    - Global Clock Versus Regional Clock
    - Column Versus Row I/O's
    - Speed Grade

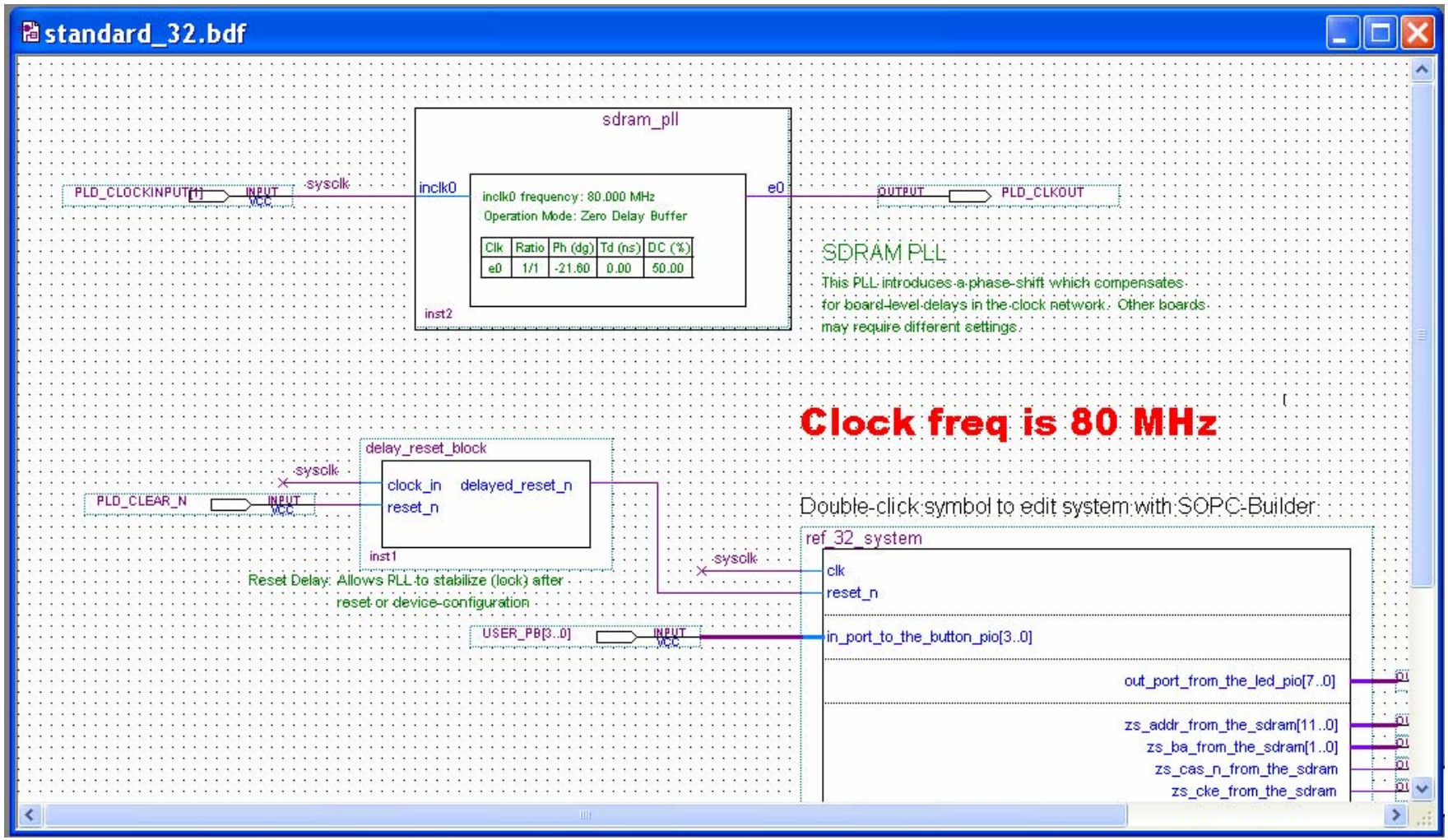
# PLL Tuning (100 MHz)



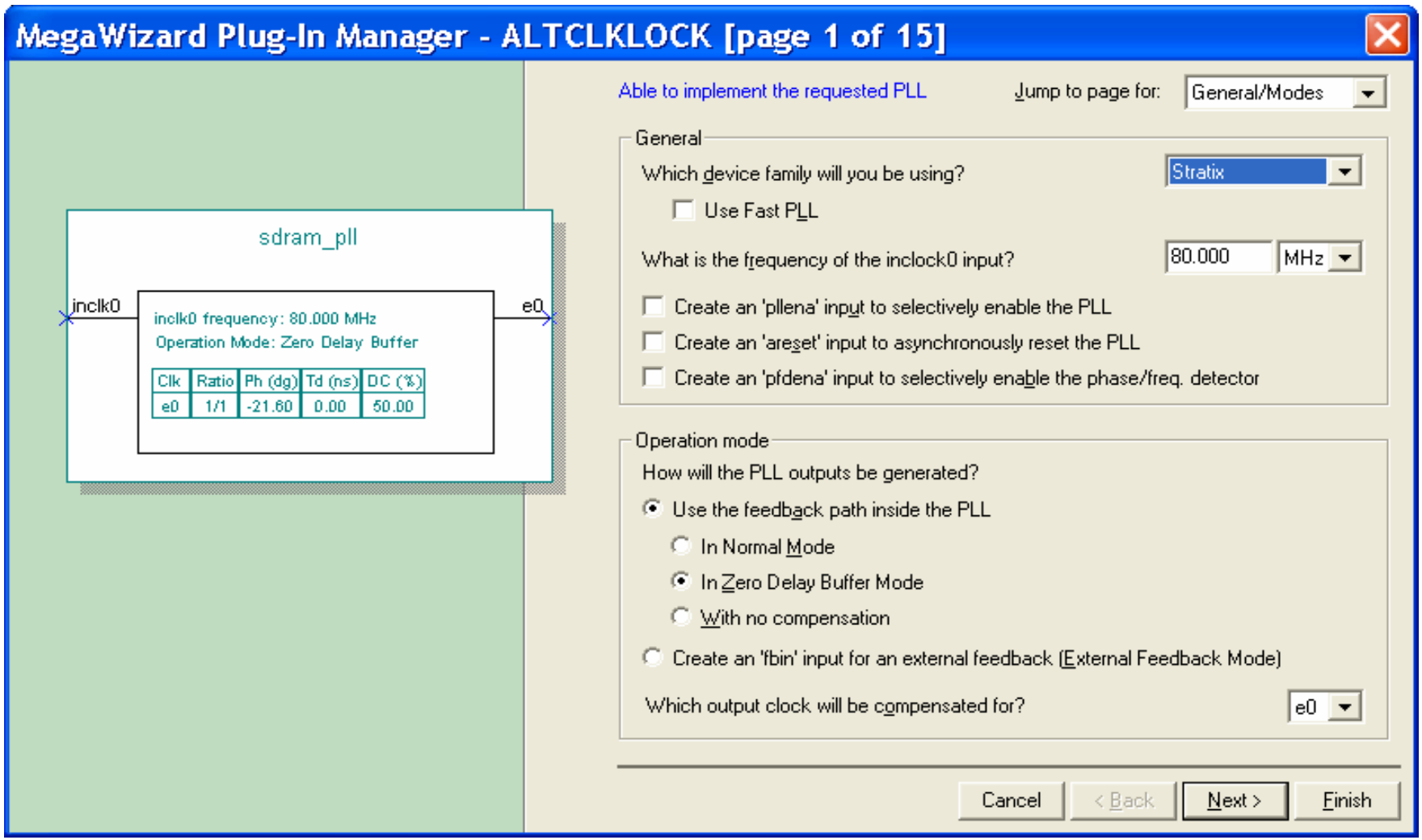
# Setting Up Clock Skew

- PLL Setup
  - Zero Delay Buffer
  - Use Input Clock for System Clock
  - Use e0 for SDRAM Clock
  - Add -0.75 ns Phase Shift

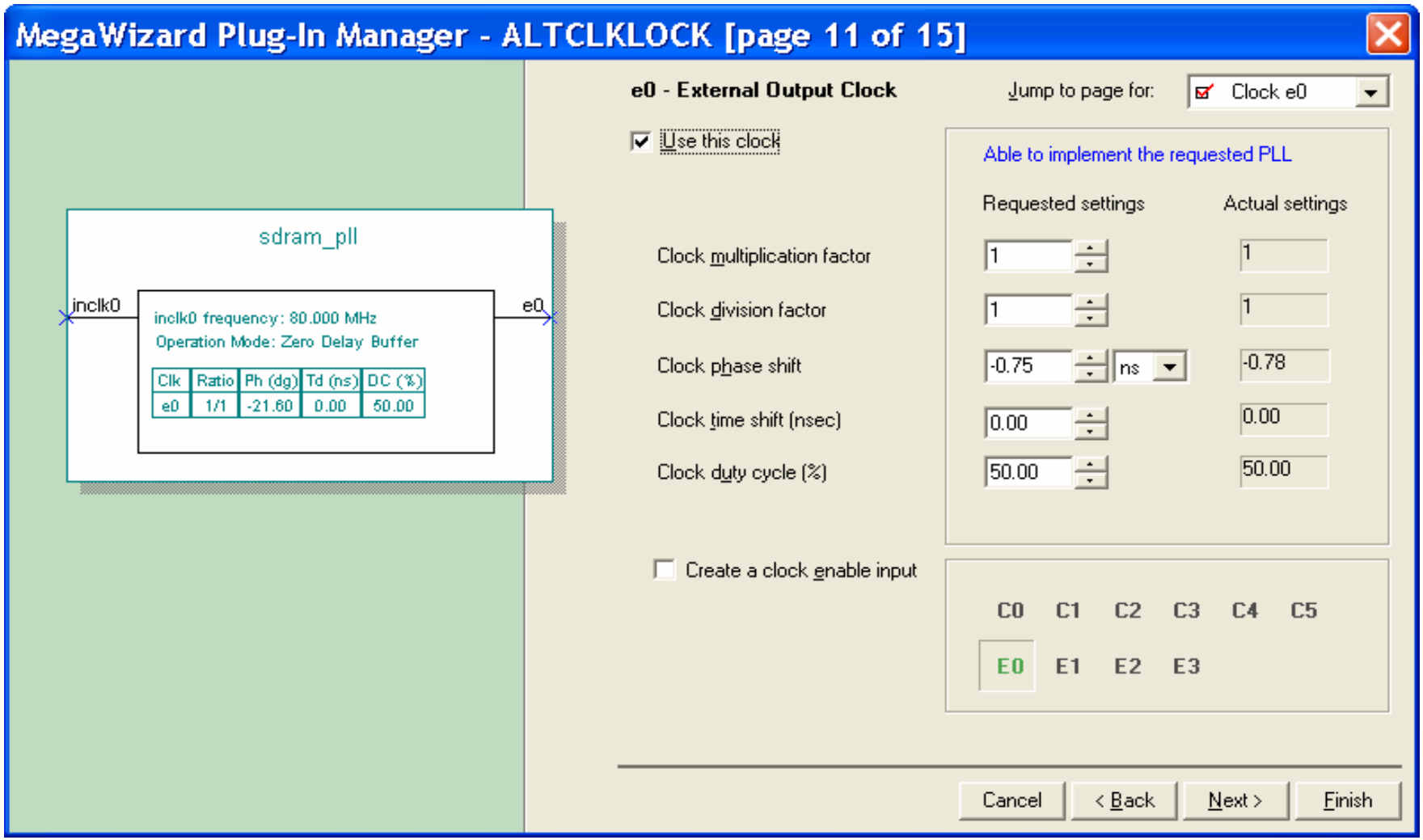
# Clock Setup



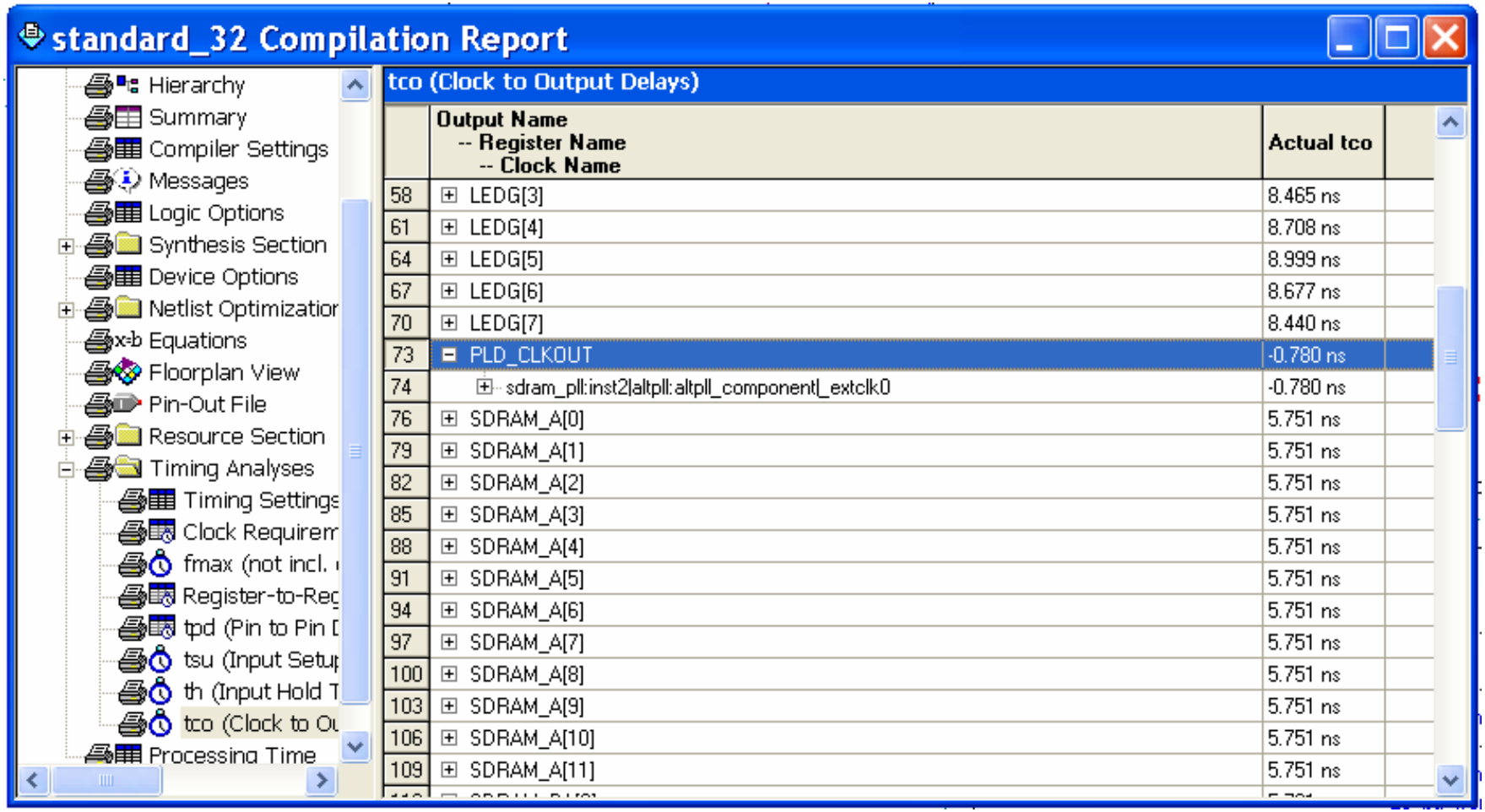
# Set Phase Shift



# Shift Clock e0



# Skew After Compile



20 YEARS *of*

ALTERA®

INNOVATION

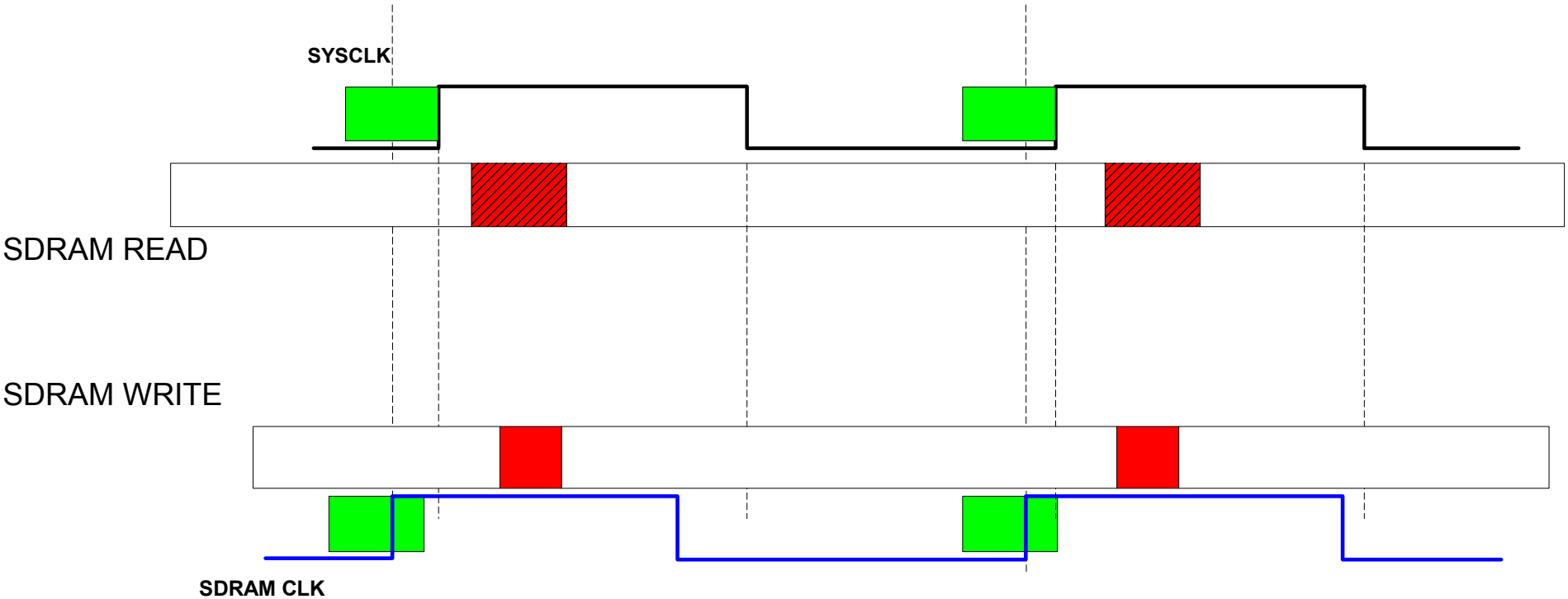
# Overtuning

# Overtuning Symptoms

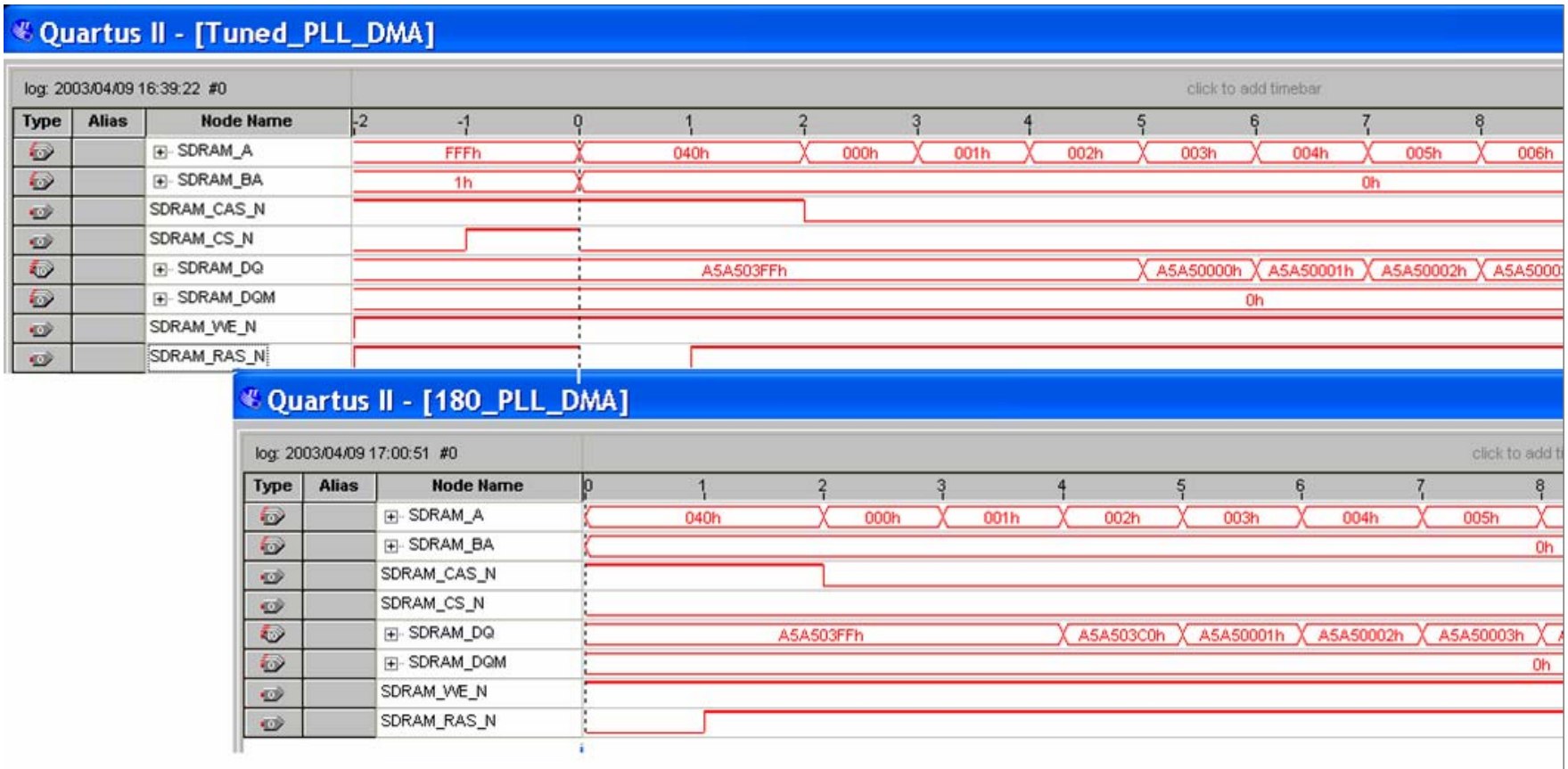
- Caused by “Trail and Error” Method
- False Tuning Window
- Single Read/Writes to SDRAM Work
- Running Code From SDRAM Fails
  - Best Way to Test Tuning
- DMA Transfer Provide Skewed Data

- This Causes the SDRAM to Provide the Data One Clock Cycle Early

# Overtuning (50 MHz)



# Overtuning: Data Received Early






20 YEARS of ALTERA INNOVATION

# Conclusions

- Tight Control is Needed for the SDRAM Clock and System Clock For System to Function Correctly
- Verify PLL's are Correct by Executing Software from SDRAM or Using DMA to Read/Write