

Программная модель

Контроллер SDRAM ведёт себя как простая память, когда к нему обращаются по интерфейсу Avalon-MM. Ему не требуется программно конфигурируемых настроек и регистров с распределением в памяти. Не требуется программный драйвер для процессора, чтобы иметь доступ к контроллеру SDRAM.

Рассмотрение тактов, PLL и временных характеристик

В этой секции описывается вопрос, связанный с синхронизацией такта от ядра контроллера SDRAM, ведущего на чип SDRAM. Во время SDRAM передачи, сигналы адреса, данных и контроля на выводах SDRAM имеют правильное значение внутри временного окна, в течение которого такт должен защёлкнуть правильные значения. На низкой тактовой частоте, такт обычно спадает внутри правильного окна. На высоких частотах, вы должны компенсировать такт SDRAM, чтобы выровнять его с правильным окном.

Определение, когда появляется правильное окно, возможно с помощью подсчётов или при анализе выводов SDRAM осциллографом. Затем используется PLL для подстройки фазы SDRAM такта так, чтобы фронт приходился на середину правильного окна. Подстройка PLL ведётся методом проб и ошибок, чтобы получить фазовый сдвиг соответствующий вашей печатной плате.

Подробнее о схеме PLL в вашем чипе в настольной книге на выбранный чип. Подробнее о конфигурировании PLL в чипах Altera в [руководстве пользователя мегафункцией ALTPLL](#).

Факторы, влияющие на временные характеристики SDRAM

Положение и длительность окна зависит от нескольких факторов:

- Временные параметры чипа и I/O выводов SDRAM - I/O временные параметры варьируются, основываясь на семействе чипа и его градации по скорости.
- Размещение вывода чипа - I/O выводы, подключенные к строковой разводке, отличаются от подключенных к столбцовой разводке.
- Логические опции, использованные во время компиляции Quartus II – Логические опции, такие как логика **Fast Input Register** и **Fast Output Register** влияют на компоновку проекта. Размещение логики и регистров внутри чипа влияет на задержку распространения сигналов до I/O выводов.
- Задержка SDRAM CAS

В результате, правильное временное окно отличается для различной комбинации чипов FPGA и SDRAM. Окно зависит от результата компоновки проекта в программе Quartus II и от назначений выводов.

Симптомы ненастроенной PLL

Определить, что PLL настроена не корректно, бывает очень сложно. Передачи данных к и от SDRAM могут постоянно не пропадать. Например, отдельные передачи к контроллеру SDRAM могут быть успешными, тогда как передача пакетов теряться. Для процессорных систем, если программа выполняет операции чтения или записи с SDRAM, но не может запуститься, когда код размещён в SDRAM, то PLL настроена не корректно.

Оценка корректности сигнального окна

В этой секции описывается, как оценить положение и размер корректного сигнального окна, используя временные параметры, представленные в технической документации на SDRAM и в отчёте компилятора программы Quartus II. После того, как окно найдено, настройте PLL таким образом, чтобы тактовые фронты SDRAM появлялись точно на середине окна.

Расчёт окна – это двух шаговый процесс. Сначала определите, на сколько времени такт SDRAM может отставать от такта контроллера, а затем, на сколько он может опережать контроллер. После нахождения максимального значения отставания и опережения, рассчитайте среднюю точку между ними.

Такой расчёт может дать только оценочное значение. Следующие задержки могут оказать влияния на характеристики настройки PLL, но не могут быть учтены в подобных расчётах:

- Расфазировка сигнала из-за задержек на печатной плате – принимается нулевое значение расфазировки.
- Задержка от узла тактового выхода PLL до точки назначения – считается, что задержка от узла тактового выхода PLL SDRAM такта до вывода такая же, как и задержка от узла тактового выхода PLL такта контроллера до тактового входа SDRAM контроллера. Если эти тактовые задержки существенно различаются, вы должны учесть их при расчёте фазового сдвига для вашего окна.

На рис. 1-5 показано, как рассчитывать максимальную длину запаздывания SDRAM такта по отношению к такту контроллера, а на рис. 1-6 показано, как рассчитывать максимальное опережение. Задержка – это отрицательный фазовый сдвиг по отношению к такту контроллера, а опережение – положительный фазовый сдвиг. Такт SDRAM должен отставать от такта контроллера не менее максимальной задержки цикла чтения или цикла записи. Другими словами, *Максимальное запаздывание = минимуму (запаздывание чтения, запаздывание записи)*. Аналогично, такт SDRAM должен опережать такт контроллера не менее максимального опережения цикла чтения или цикла записи. Другими словами, *Максимальное опережение = минимуму (опережение чтения, опережение записи)*.

Figure 1–5. Calculating the Maximum SDRAM Clock Lag

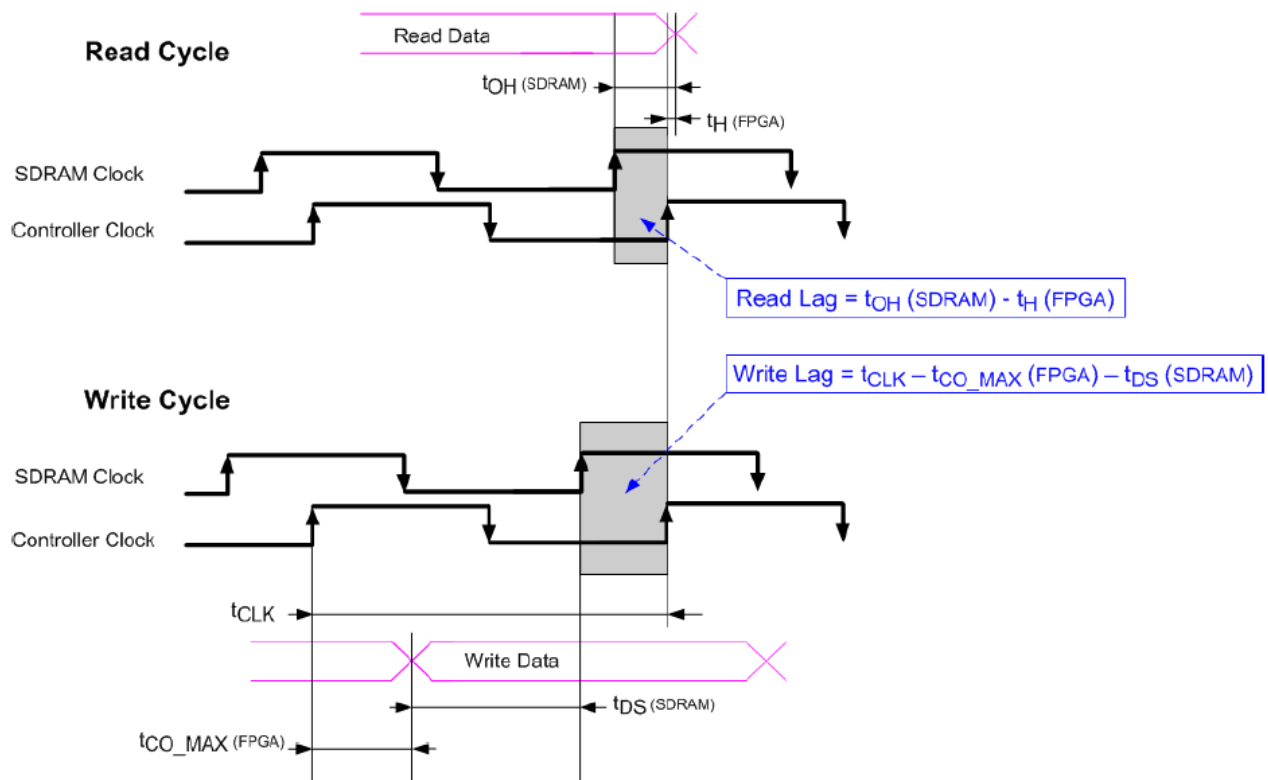
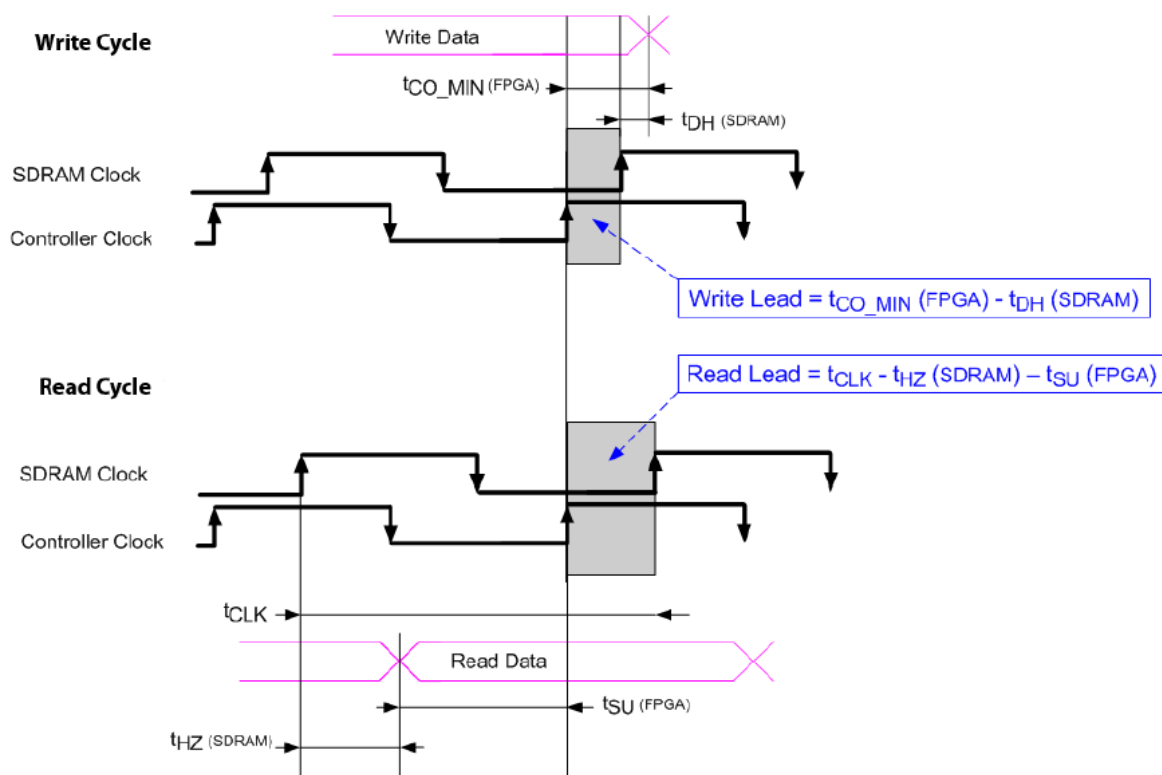


Figure 1–6. Calculating the Maximum SDRAM Clock Lead**Пример расчёта**

В этой секции приводится пример расчёта сигнального окна для чипа SDRAM Micron MT48LC4M32B2-7 и проекта, реализованного в чипе Stratix II EP2S60F672C5. В этом примере используется задержка CAS (CL) – 3 цикла и тактовая частота 50 МГц. Все сигналы SDRAM в чипе регистрируются в I/O ячейках с разрешёнными логическими опциями **Fast Input Register** и **Fast Output Register** в программе Quartus II. В табл. 1-3 показаны соответствующие временные параметры, взятые из технической документации на чип MT48LC4M32B2.

Table 1–3. Timing Parameters for Micron MT48LC4M32B2 SDRAM Device (Part 1 of 2)

Parameter		Symbol	Value (ns) in -7 Speed Grade	
			Min.	Max.
Access time from CLK (pos. edge)	CL = 3	$t_{AC(3)}$	—	5.5
	CL = 2	$t_{AC(2)}$	—	8
	CL = 1	$t_{AC(1)}$	—	17
Address hold time		t_{AH}	1	—
Address setup time		t_{AS}	2	—
CLK high-level width		t_{CH}	2.75	—
CLK low-level width		t_{CL}	2.75	—

Table 1–3. Timing Parameters for Micron MT48LC4M32B2 SDRAM Device (Part 2 of 2)

Parameter		Symbol	Value (ns) in -7 Speed Grade	
			Min.	Max.
Clock cycle time	CL = 3	$t_{CK(3)}$	7	—
	CL = 2	$t_{CK(2)}$	10	—
	CL = 1	$t_{CK(1)}$	20	—
CKE hold time		t_{CKH}	1	—
CKE setup time		t_{CKS}	2	—
CS#, RAS#, CAS#, WE#, DQM hold time		t_{CMH}	1	—
CS#, RAS#, CAS#, WE#, DQM setup time		t_{CMS}	2	—
Data-in hold time		t_{DH}	1	—
Data-in setup time		t_{DS}	2	—
Data-out high-impedance time	CL = 3	$t_{HZ(3)}$	—	5.5
	CL = 2	$t_{HZ(2)}$	—	8
	CL = 1	$t_{HZ(1)}$	—	17
Data-out low-impedance time		t_{LZ}	1	—
Data-out hold time		t_{OH}	2.5	—

В таблице 1-4 показана соответствующая информация по временным характеристикам, полученная из секции временного анализатора отчёта о компиляции Quartus II. Значения в таблице – это максимальное и минимальное значение из всех выводов чипа, подключенных к SDRAM. Разница во времени между SDRAM выводами чипа мала (менее 100 пс), поскольку регистры для этих сигналов размещены в I/O ячейках.

Table 1–4. FPGA I/O Timing Parameters

Parameter	Symbol	Value (ns)
Clock period	t_{CLK}	20
Minimum clock-to-output time	t_{CO_MIN}	2.399
Maximum clock-to-output time	t_{CO_MAX}	2.477
Maximum hold time after clock	t_{H_MAX}	–5.607
Maximum setup time before clock	t_{SU_MAX}	5.936

Вы должны скомпилировать проект в программе Quartus II, чтобы получить информацию об I/O временных характеристиках проекта. Вся техническая документация на семейство чипов Altera содержит общую информацию об I/O временных характеристиках для каждого чипа, но в отчёте компилятора Quartus II содержится более точная информация, конкретно для вашего проекта.

Значение временных характеристик в отчёте компилятора может отличаться, в зависимости от результатов компоновки, размещения выводов и прочих настроек логики Quartus II. Когда вы перекомпилируете проект в программе Quartus II, следите за тем, чтобы ваши временные I/O характеристики не слишком менялись.

В следующем примере показан расчёт для рис. 1-5 и рис. 1-6 с использованием значений из табл. 1-3 и табл. 1-4.

Такт SDRAM должен отставать от такта контроллера на наименьшее значение *отставания чтения* или *отставания записи*:

$$\text{Read Lag} = t_{OH}(\text{SDRAM}) - t_{H_MAX}(\text{FPGA})$$

$$= 2.5 \text{ ns} - (-5.607 \text{ ns}) = 8.107 \text{ ns}$$

или

$$\text{Write Lag} = t_{CLK} - t_{CO_MAX}(\text{FPGA}) - t_{DS}(\text{SDRAM})$$

$$= 20 \text{ ns} - 2.477 \text{ ns} - 2 \text{ ns} = 15.523 \text{ ns}$$

Такт SDRAM должен опережать такт контроллера на наименьшее значение *опережения чтения* или *опережения записи*:

$$\text{Read Lead} = t_{CO_MIN}(\text{FPGA}) - t_{DH}(\text{SDRAM})$$

$$= 2.399 \text{ ns} - 1.0 \text{ ns} = 1.399 \text{ ns}$$

или

$$\text{Write Lead} = t_{CLK} - t_{HZ(3)}(\text{SDRAM}) - t_{SU_MAX}(\text{FPGA})$$

$$= 20 \text{ ns} - 5.5 \text{ ns} - 5.936 \text{ ns} = 8.564 \text{ ns}$$

Таким образом, в этом примере вы можете сдвинуть фазу такта SDRAM от -8,107 нс до 1,399 нс по отношению к такту контроллера. Выберите фазовый сдвиг на средней точке результатов для этого окна $(-8.107 + 1.399)/2 = -3.35 \text{ ns}$.