Synchronous DRAM Controller

Purpose:
- This module covers the SDRAM controller specifically for providing the correct memory control signals.

Objectives:
- This module will provide you with a basic understanding of the SDRAM device within the memory controller and how it operates. You will also be provided an example of how this can be configured.

Contents:
- This section includes a basic block diagram and commands issued for the SDRAM device. It includes addressing overviews and an in-depth look at the memory controller registers. This section completes with a SDRAM programming procedure.

Learning Time:
- There are 13 pages and 8 reference pages in this module, which will take approximately 24 minutes to complete.

This module covers the SDRAM controller specifically for providing the correct memory control signals.

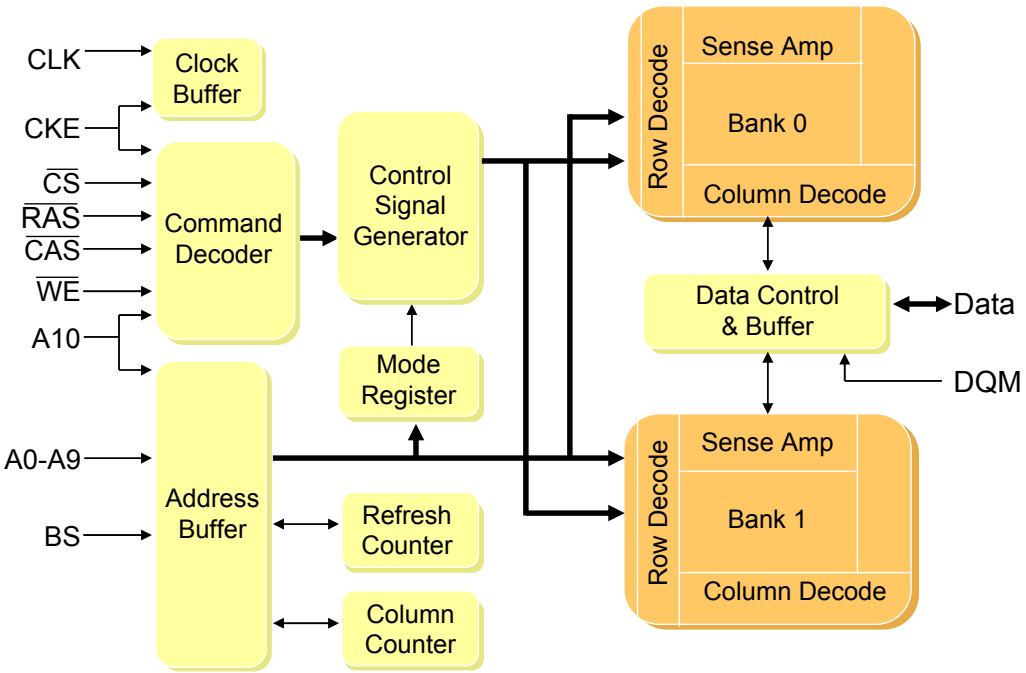
This module will provide you with a basic understanding of the SDRAM device within the memory controller and how it operates. You will also be provided an example of how this can be configured.

This section includes a basic block diagram and commands issued for the SDRAM device. It includes addressing overviews and in-depth look at the memory controller registers. This section completes with a SDRAM programming procedure.

There are 13 pages and 8 reference pages in this module, which will take approximately 24 minutes to complete.

SDRAM

SDRAMs contain the core of a standard DRAM plus a clock-gating pin that synchronizes all inputs and outputs to a single system clock.



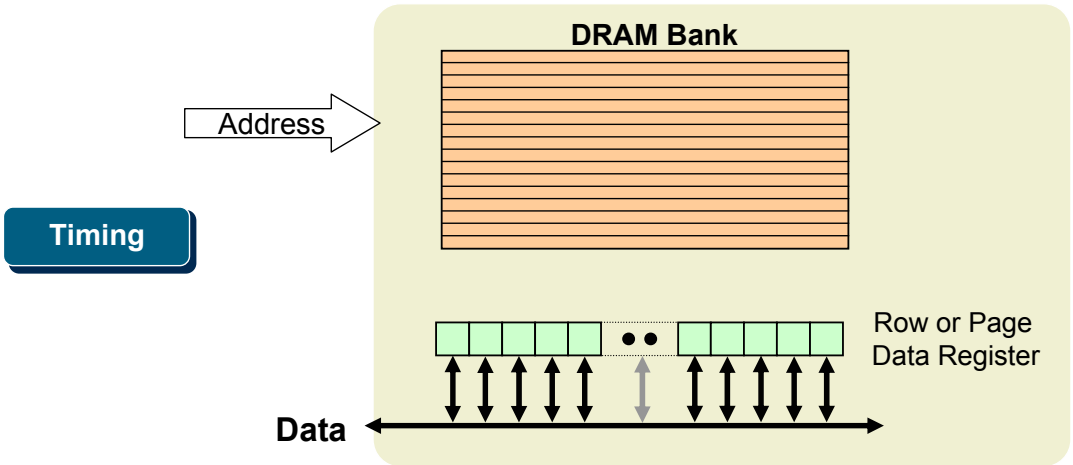
This diagram shows the basics of an SDRAM device. The main memory consists of a matrix of Dram like memory, and it's possible to have several of these, known as banks. The standard options are banks of 2, 4 or 8. Data is transferred via a buffer, so that when a memory location is accessed, a complete page of memory is loaded into the buffer, and then read or written via this buffer as required.

The basic accessing of the device is similar to that of DRAM using multiplexed addressing and it requires refreshing on a regular basis. The device is synchronous and so has a clock input which must be the same clock that controls the bus controller. A major difference from standard Dram is that to improve the speed and volume of this memory, the memory junctions are much weaker than standard DRAM, resulting in the junction losing the memory charge when it is read.

This is not a problem for reading the memory since the data is loaded into a buffer where it is accessed. However, if a new page is to be accessed, the data in the buffer must be written back into the matrix before it is lost. This results in some necessary commands during access known as activate and pre-charge. Activate is the act of transferring data from the matrix to the internal buffer. Pre-charge is the act of returning data from the buffer into the matrix.

The device also has a number of configuration options as well as static commands, and so there is a command decoder which responds to combinations of chip select, row address and column address strobes, write enable and address line ten. The SDRAM must be configured before use, at start-up. Data is transferred to and from the internal buffer using the qualifier signal known as DQM.

SDRAM Operation (1 of 2)



This diagram shows the basic operation of how SDRAM handles data. It only considers the steps necessary to reliably store and retrieve data, and focuses on the memory matrix and internal buffer.

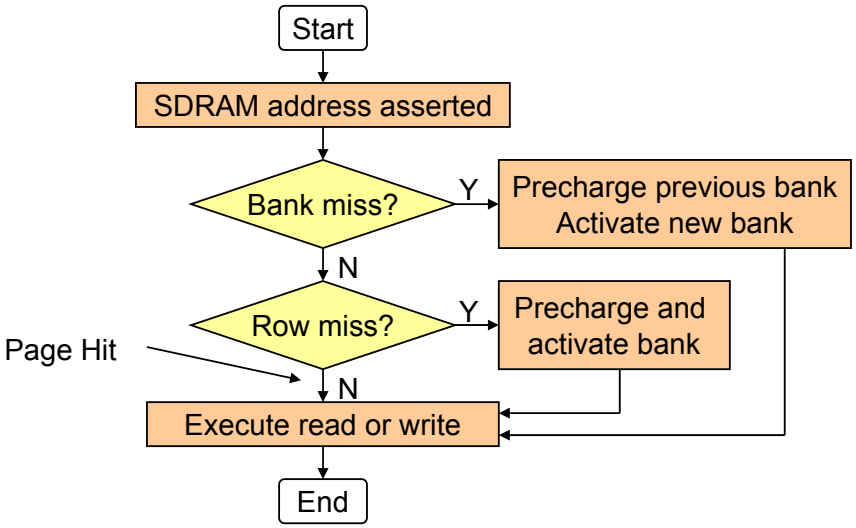
When the address to a new page arrives that data is transferred into the internal buffer.

As data is accessed within that page it is simply transferred to and from the buffer.

When a new page address is accessed, the data from the previous page must be written back to the original location from the page data register. This is known as pre-charge. To see an example timing diagram of various transfers, click on the timing button.

SDRAM Operation (2 of 2)

The flow diagram below shows the basic operation of an SDRAM when an address is asserted. It assumes the bank and row address registers are marked valid.



The flow chart shows the basic operation of an SDRAM when an address is asserted. It assumes the bank and row address registers are marked valid. When the address is asserted a check is made for the access being in the current bank. If not, then the data is written from the buffer into its original location and the new bank location is activated, loading the new data into the buffer.

If the access is in the current bank, then is it in the same row or page ? If not, then again the data is returned to it's original location and the new data loaded into the buffer. If the access is within the same bank and the same row, then the data is accessed via the buffer without affecting the bank.

SDRAM Commands

The MPC8260 SDRAM controller basically operates in two modes:
command mode and normal mode.

xSDMR[OP]

Command Mode	
001	CBR refresh
010	Self refresh
011	Mode register write
100	Precharge bank
101	Precharge all banks
110	Activate bank
111	Read/write

Normal Mode
xSDMR[OP] = **000**
Single-beat reads/writes up to 64 bits.
Bursts of two, three, and four 64-bit values.

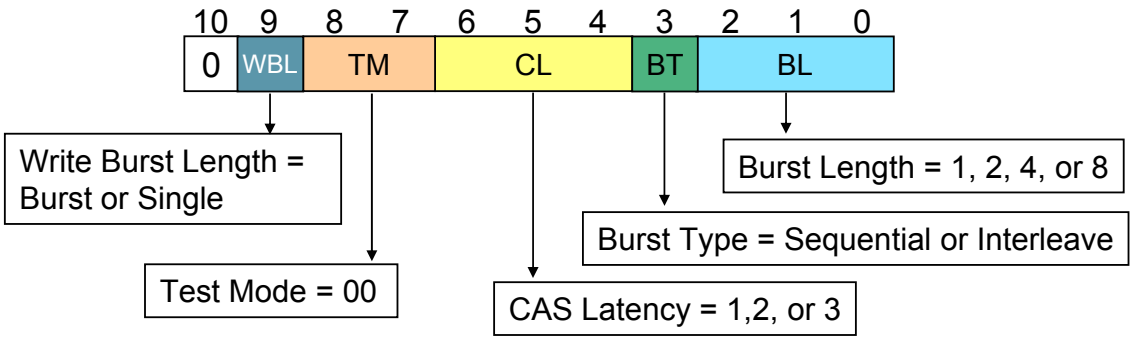
Some of the commands are implemented as functions which are part of read and write in the normal mode.

The MPC8260 SDRAM controller basically operates in two modes, command mode and normal mode. The mode of operation is controlled by a memory controller register called SDMR. Depending on the device, the normal operation can transfer data as single beats of up to 64-bits, or bursts of 2, 3, or 4 64-bit values.

There are some commands that are performed in normal operation, as seen earlier, the auto pre-charge and activate commands. There are seven commands that are used for initializing the device and for test purposes. The commands are provided as a code in the SDMR and then transferred to the device using a dummy memory access. For initializing the SDRAM, the mode register write command is sent and the command register data is transferred on the lower significant address bus.

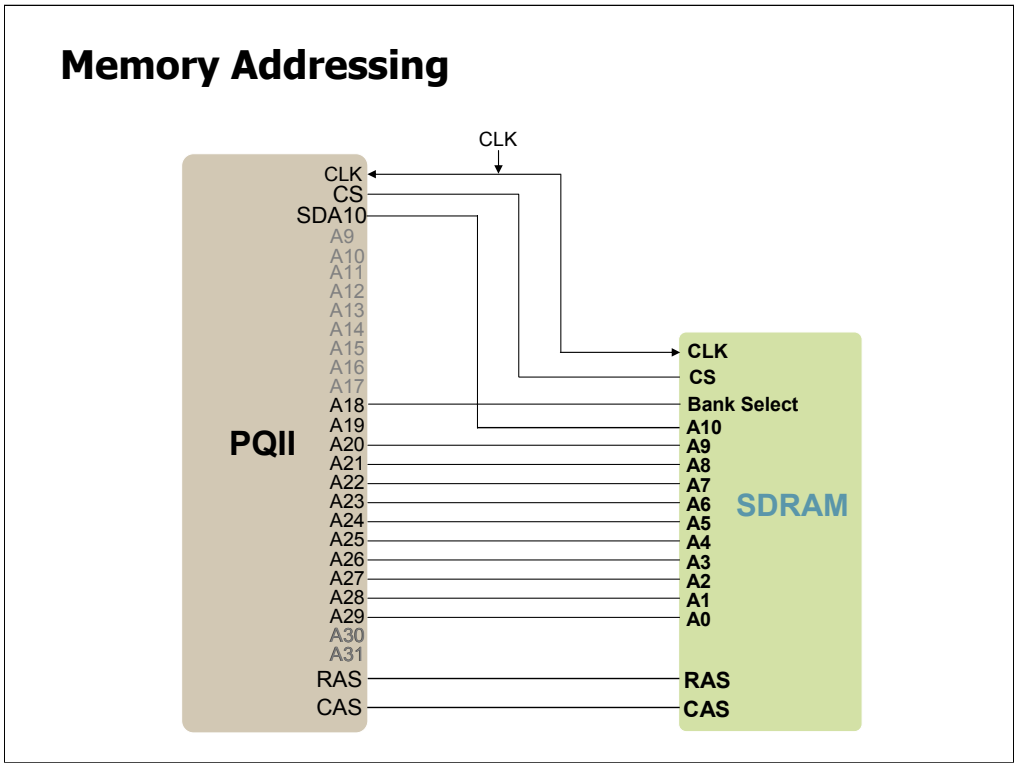
SDRAM Mode Register

- SDRAM devices contain a command register to enable certain options to be selected.
- The mode-set command initializes the mode register in the SDRAM.
- The value to go into the mode register is asserted on the address bus.



SDRAM devices contain a command register to enable various options to be selected. The mode-set command initializes the mode register in the SDRAM. The value to go into the mode register must be driven on low order address bits of the address bus. Any remaining address pins connected to the device should be driven low. This shows the standard format of an SDRAM command register.

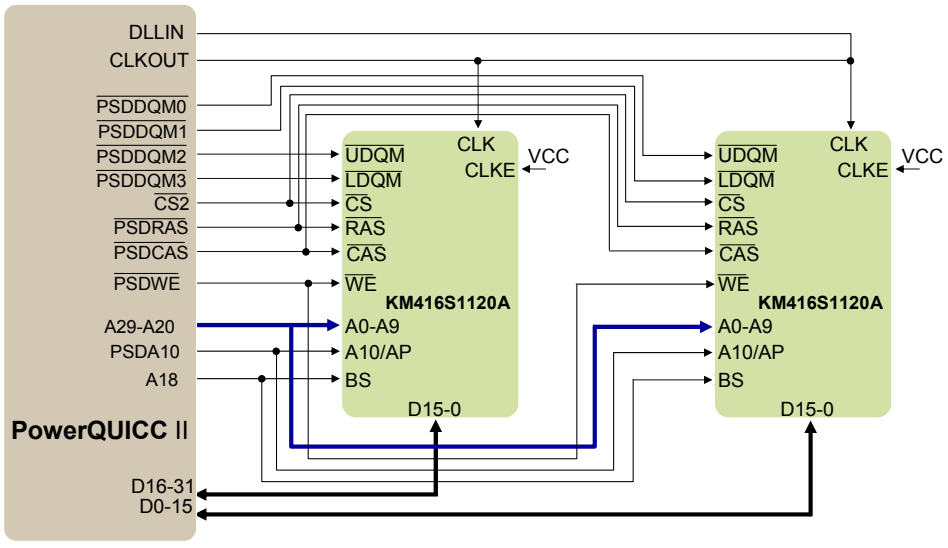
The lower order bits define the burst length expected on the bus. Burst type defines whether transfers are to be sequential or interleaved. The PowerQUICC II uses sequential transfers. CAS latency defines the number of clock pulses between assertion of CAS and data transfer. Test mode is specific to the device and should be set to zero. Write burst length defines if transfers are to be single beat or bursts. For write cycles, burst is normal.



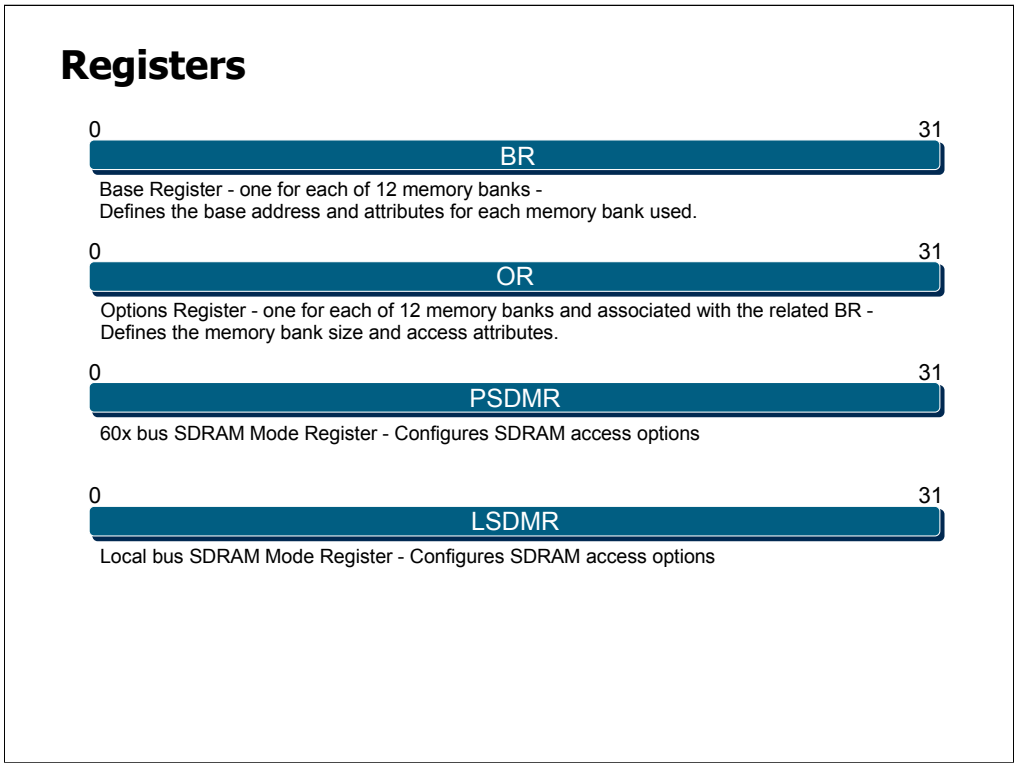
The Address connections between the PowerQUICC II and SDRAM are flexible and depend on the memory device. The basic connections will be the lower order address bits, depending on the data access size and memory range. At least one of the address lines is connected to bank select, and the SDA10 pin on the PowerQUICC II is connected to address pin ten on the SDRAM.

A chip select, clock, row address strobe and column address strobe are also required. A complete example connection diagram is shown later. For a complete explanation of addressing, and how the required parameters are decided, click on the SDRAM block.

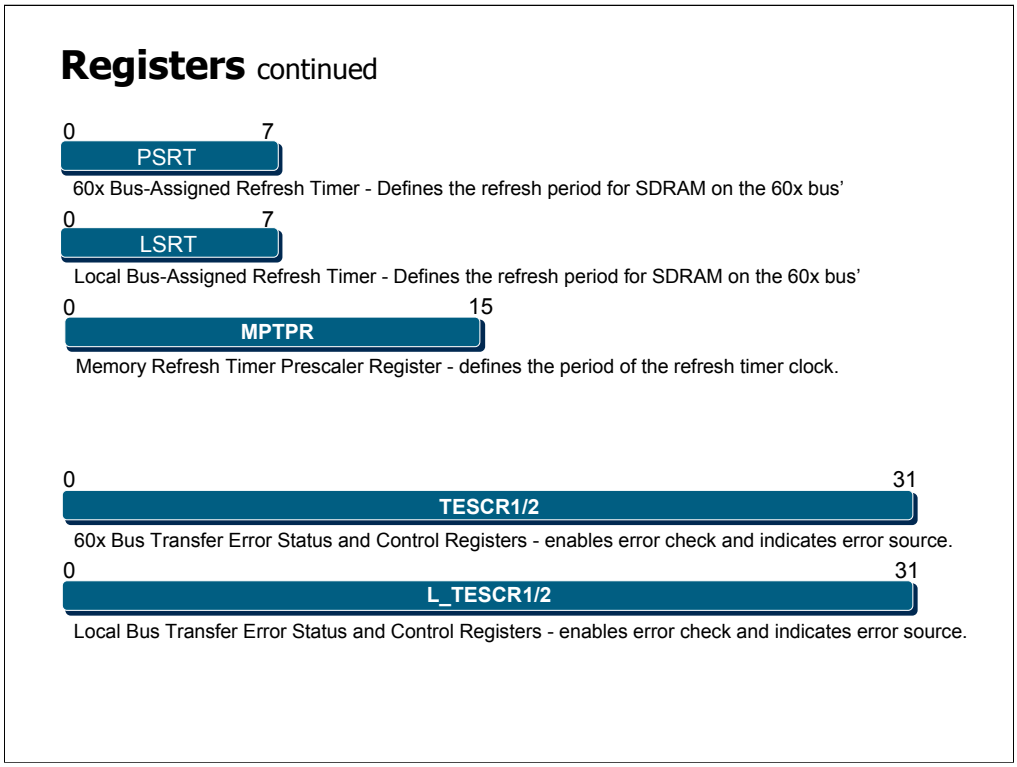
Example SDRAM connection



This diagram shows an example of the connection between the PowerQUICC II and SDRAM for eight megabytes by 32-bit data.



These are the major registers required to set up the memory controller. The base register is configured in the same way regardless of the controller it relates to. The option register takes a different format for the SDRAM controller and the PSDMR and LSDMR are specifically used for the SDRAM controller. To view the registers in detail, click on the one of interest.



These are the other registers involved in setting up the memory controller. PSRT, LSRT and MPTPR are used to define the refresh period for SDRAM on the 60x bus and local bus. PSRT and LSRT are simple 8-bit registers containing the refresh timer period required to generate the appropriate refresh cycle on the 60x bus and local bus.

TESCR1 and two and LTESCR1 and two are error status registers for both the 60x bus and local bus. To view the registers in detail, click on the one of interest.

xSDMR Timing Parameter Values

The PSDMR and LSDMR contain a number of timing parameters whose values must be determined from the SDRAM data sheet.

Data Sheet Parameters

xSDMR Parameter	IBM		Samsung		Etron Tech	
	Sym	Parameter	Sym	Parameter	Sym	Parameter
PRETOACT	t_{RP}	Precharge time	t_{RP}	Row pre-charge time	t_{RP}	Precharge to activate
ACTTORW	t_{RCD}	RAS* to CAS* delay	t_{RCD}	RAS* to CAS* delay	t_{RCD}	RAS* to CAS* delay
LDOTOPRE	t_{DPL}	Data in to Precharge	t_{RDL}	Last data in to precharge		
WRC	t_{WL}	Write latency	t_{CDL}	Last data in new col addr	t_{WR}	Write recov-ery time
CL	t_{AA}	CAS* latency	Variable from 2 to 3		Variable from 1 to 3	
RFRC	t_{RC}	Bank Cycle Time	t_{RC}	Row Cycle Time	t_{RC}	Row Cycle Time

PSDMR and LSDMR must be initialized with timing parameters related to the SDRAM in use. However, not all data sheets provided for memory devices quote these parameters in the same way or give them the same designation. This chart indicates how some of these parameters are indicated by some manufacturers.

Refresh Rate

Refresh Rate Calculation

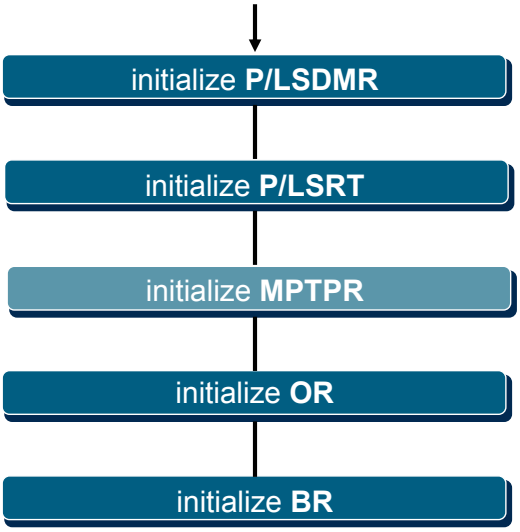
xSRT * MPTPR < System Clock Period * Refresh Rate

Given a system clock frequency of 100 MHz and a refresh rate requirement of 15.6 usec, determine the values for xSRT and MPTPR.

xSRT * MPTPR < Refresh Rate * System Clock Frequency
< 15.6 * 10 ^{6}* 100 * 10^{6}
< 1560
Square root of 1560 is 39.xxx
39 * 39 = 1521
Therefore,
xSRT = 39 and MPTPR = 39

The refresh rate is determined by the values in xSRT and MPTPR registers. Each time the timer expires, all blanks that qualify generate a bank staggering auto-refresh request using the SDRAM machine."

SDRAM Programming Procedure



SDRAM initialization

Wait 200 µSec after power is applied

Issue a precharge all banks command, **PSDMR[OP]=0b 101**

Issue 2 or more (typically 8) auto-refresh (CBR) command,

PSDMR[OP]=0b 001

Issue a mode register write command, **PSDMR[OP]=0b 011**

The memory control for the bank is enabled when the valid bit is set in the Base register

All that is necessary to program the memory controller for SDRAM is to initialize the registers. Given the required timing parameters the access timing information is defined in the SDMR, as well as the address routing needed. There is a register for each of the 60x bus and local bus. If only one of these has SDram connected to it, then only the relevant register needs programming.

The refresh timers and prescaler registers must be initialized with the appropriate values. Again, both buses have their own refresh timer register. The option register provides the memory bank size, as well as some additional signal timing options. Finally, the base register is programmed with the base address for the bank, data access size, the machine to use, in this case the SDRAM controller and some other bank specific options.

Since the error reporting registers TESCR are cleared at reset they should not require initialization, although it wouldn't hurt to clear them first. They are used to trace bus access errors. Before using SDRAM it must be initialized. First wait 200 microseconds after power is applied. Issue a precharge all banks command using the PSDMR parameter OP value 101.

Issue at least two, recommended eight, auto-refresh commands, CBR using the PSDMR parameter OP value 001. Issue a mode register write command, using the PSDMR parameter OP value 011. That completes the SDRAM module.