

Design and Verification of SDRAM and Bus Interface

Vijaykumar, Akshay, SID 010276253, SJSU, and K N, Hemanth, SID 010259249, SJSU

Abstract—SDRAM is a popular slave memory device owing to its cost effectiveness and ease of manufacturing as compared to other memory devices. This design example demonstrates a SDRAM slave memory interfaced to a bus master using a bus interface unit.

I. INTRODUCTION

Memory devices may have their own design protocol which may not be necessarily the same as the system bus protocol to which they are interfaced. A bus interface unit is required in order to translate signals from the local system bus, to the signals required by the memory slave device for read/write operations.

This example below explores designs of a simple SDRAM memory device and a bus interface unit compliant with the unidirectional bus protocol that was presented in an earlier project.

II. SYSTEM ARCHITECTURE

Fig.1 shows a high level block diagram of the BIU and SDRAM interfaced together. The BIU generates the required control and Address signals for the SDRAM device.

The input signals to the BIU, namely ‘Status’, ‘Write’, ‘Burst’ and ‘Size’ are provided by the bus master. The register file holds the timing configuration for the SDRAM. Each of the components above are explored in detail in the subsequent sections.

III. SDRAM

A. Architecture

SDRAM or Synchronous Dynamic Random Access Memory, utilizes capacitor as the fundamental storage element. In comparison to SRAM or Static Random Access Memory, which utilizes CMOS device as a storage unit, SDRAM is evidently more cost effective. SDRAM can also have a large packing density. But, because a capacitor continuously loses its charge, SDRAM needs a refreshing circuitry that periodically restores the charge to the capacitor. SRAM on the other hand needs no refreshing circuitry and have very high speed access rates as compared to SDRAMs.

A simple block diagram of a SDRAM device is shown in the Fig.2. The memory is organized into multiple banks where each bank has multiple rows and columns. Specific data bytes can be accessed by supplying a row and a column address to the device.

The memory module implemented in this design has the

following ports: CS, RAS, CAS, WE, Data, BS, AddrIn and Clk. All control signals are active low.

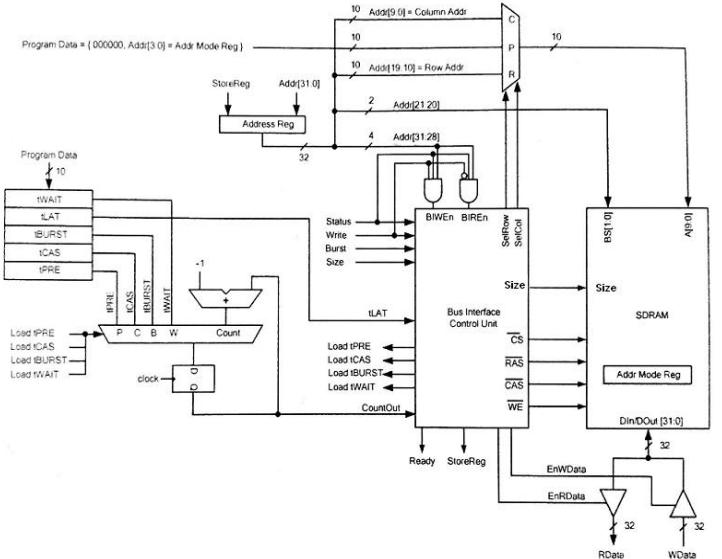


Fig 1. SDRAM and BIU; System Architecture

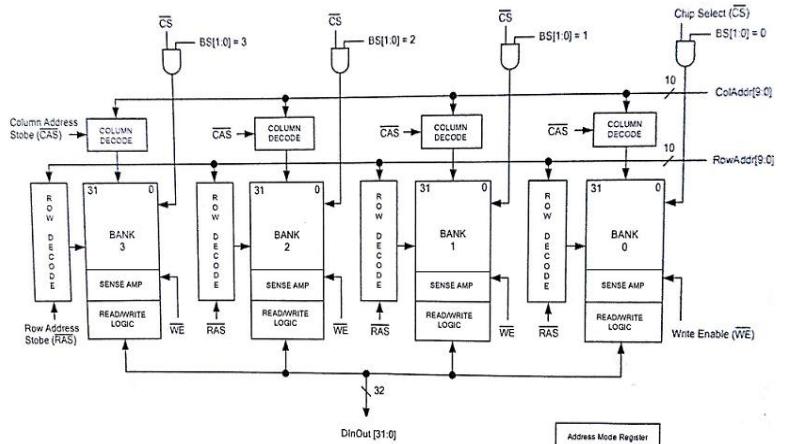


Fig 2. SDRAM Bank Organization

CS is Chip Select active low signal. RAS is the Row Address Strobe signal which is used to validate row address on the address line. CAS is Column Address Strobe signal used to validate the column address on the Address line. WE is Write Enable which signals whether a read or write operation should take place. Data is an inout port used for reading and writing data, from and to the memory device. BS is Bank Select, used to select a specific Bank from the device. AddrIn port is used to receive Row and Column addresses

from outside the module.

A read or a write transaction from the device requires a number of operations to be completed. The CS, RAS, CAS and WE signal combinations are also utilized for signaling the device to execute each of the required operations. The control flow and data flow for the device is discussed in the subsequent sections.

The device also has a Row Decoder and a Column Decoder. These decoders are used to select a specific row and a column from a specific bank.

Address mode register specifies whether the device should follow sequential or linear addressing modes. The module also has a programmable register file that holds the timing specifications for the device, which are pre-charge period(tPre), Activation period(tCas), Latency period(tLat), Burst period(tBurst) and Wait period(tWait).

B. Control and Data Flow

Table 1. BIU Command List

CS	RAS	CAS	WE	OPERATION
0	0	0	0	Program Addr. Mode Register
0	0	0	1	Self Refresh
0	0	1	0	Precharge a Bank with BS [1:0]
0	0	1	1	Activate a Bank with BS [1:0]
0	1	0	0	Write into a Bank with BS [1:0]
0	1	0	1	Read from a Bank with BS [1:0]
0	1	1	0	Burst Stop
0	1	1	1	Reserved
1	X	X	X	SDRAM Deselect

The control commands supported by the device are shown in the Table 1.

On system reset, the first command expected by the device is a “Program Addr. Mode Register” command. This command enables the register file for writing and programs the timing specifications.

For a read or write transaction to take place, the memory device should be pre-charged and activated. Pre-charging allows a specific bank to be prepared for a transaction. Any existing activated rows from the bank are closed and all the rows and columns of the bank are pre-charged to a certain voltage level. This operation completes after the tPre period. Pre-charge period is the number of clock periods between the beginning of pre-charge and issuing of activation command.

Activation command should be issued with a row address. This command will cause the selected bank to decode the row address using the row decoder and transfer the contents of the entire row into a row buffer. Any read or write transaction with this row will take place within the row buffer. Activation takes tCas periods to complete. Activation period is the number of clock periods between the start of activation and issuing of the read/write command.

A read or a write command should be issued next. These commands are issued along with the column address. The column decoder from the bank selects the required column from the row buffer. Read or write operation is performed from this buffer. Read operation has a configurable tLat period. Latency period is the number of clock periods between the issuing of the read command and the availability of the

first data packet from the device. Write operation will not incur any latency period. The SDRAM protocol is such that the data packets to be written to the bank must be supplied starting along with the write command.

Once the read or write command is issued, after the latency period is over if applicable, a tBurst number of data packets are read out from or written to the row buffer starting from the column address provided with the command.

After the read/write transaction the bank goes into the wait period and waits for a number of clock cycles defined by tWait before another pre-charge command can be issued. For a given bank, a pre-charge and activation is necessary for every transaction. Pre-charging a bank also closes the current active row, which means that, the contents of the row buffer are stored back into the memory core on every pre-charge operation. This ensures that before a new row is activated and the row buffer is overwritten with the contents of this new row, the existing contents of the buffer with any write modifications are written back to the core memory.

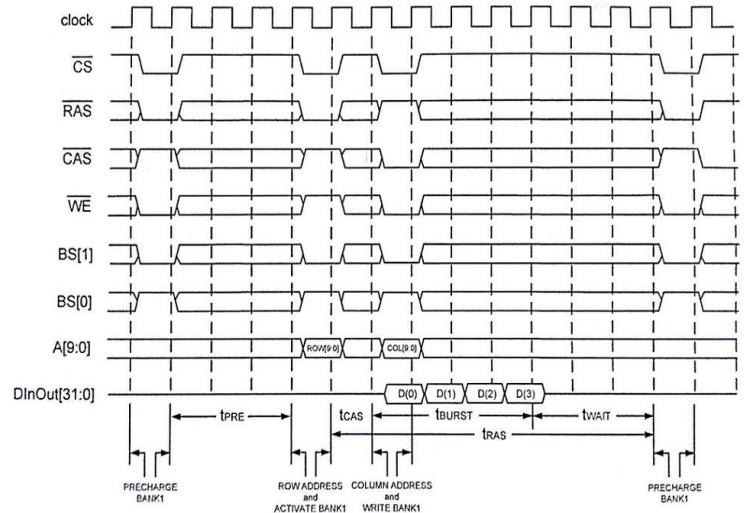


Fig 3. Single Write Cycle

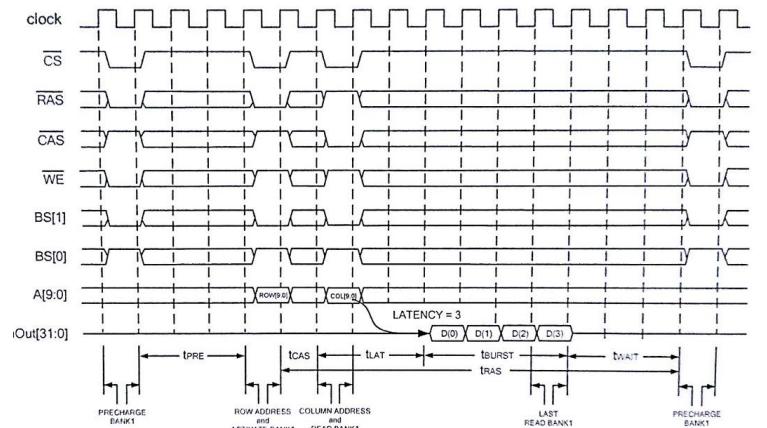


Fig 4. Single Read Cycle

Fig.3 and Fig.4 depict a single read and write cycle respectively. The tPRE, tCAS, tBURST, tLAT and tWAIT periods are all marked in the timing diagrams.

The test framework, verification table and waveforms are shown in V.11.

C. Components

The design of SDRAM in this implementation has 3 modules, viz; Register File, Bank Select Decoder and SDRAM Bank. A description of each of these modules and their sub-modules are provided in the sections below.

1. Register File

The register file module holds six registers namely tBurst, addr_mode, tLat, tPre, tWait and tCas. The inputs to the module are CS, RAS, CAS, WE and AddrIn. To program the address mode register CS, RAS, CAS and WE are pulled low. The data available from the AddrIn port are programmed into the registers.

On Boot-up all the registers are allowed to be programmed the first time. Any subsequent program commands will only affect addr_mode and tBurst. This ensures that the timing configuration of the RAM remains constant throughout its power cycle.

The outputs of these registers are always available from the module through the respective output ports.

Test framework, verification table and waveforms are as shown in Section V.1.

2. Bank Select Decoder

The Bank Select Logic has one output port and two input ports. ‘BSOut’ is a 4-bit wide output port through which the Bank Select Decoder sends the enable signals for each bank. ‘BSIn’ is a 2-bit input port that provides the ID of the Bank to be enabled. ‘CS’ is the chip select for the entire SDRAM. All the signals are active low.

In this design, we have four SDRAM Banks. The Bank Select Decoder enables a particular bank depending on the Bank Select and Chip Select input lines.

The verification tables and waveforms are as shown in Section V.2.

3. SDRAM Bank

In this design, each SDRAM Bank consists of 2 modules, the SDRAM Controller and SDRAM Memory Core. SDRAM Controller provides the necessary control signals and addresses for the operation of the SDRAM Memory Core. SDRAM Memory Core contains the memory array for each bank and carries out read/write operations from/to the memory array.

The test framework for this module involves a set of 14 transactions. The verification table and waveforms are as shown in Section V.7.

i. SDRAM Controller

The SDRAM Controller has 3 modules, viz; Timing Generation Unit, Enable Unit and Address Generation Unit. A description of each of these modules are provided below.

a. TimeGen

TimeGen is the Timing Generator unit for the SDRAM bank. It has two main functions. One is a state counter that counts through each operation of the SDRAM explained in section II.B. Another function is a down counter that tracks the pre-charge, activation, latency, burst and wait periods for the bank. Based on these counters the TimeGen generates relevant signals for the Enable unit to issue control signals to the memory core.

The state counter implementation in the TimeGen unit is a 3-bit counter. Each counter value and its interpretation is summarized in the Table 2 below.

Table 2. State Counter Information

State Counter Value	State	Description
000	Reset	Down counter reset in this state
001	Pre-charge	Down counter is loaded with tPre
010	Activation	Down counter is loaded with tCas
011	Read latency	Only during Read. Down counter loaded with tLat
100	SBurst period	Down counter loaded with tBurst for read and (tBurst -1) for write
101	Wait period	Down counter loaded with tWait

At every transition of the state counter a signal ‘TimerLd’ goes high which loads the relevant count-down value into the down counter. The loaded values are also summarized in the Table 2 above.

On system reset the state counter goes to reset state. When pre-charge command is issued (refer to Table 1 for a list of commands) TimerLd goes high and the state counter transitions to State 001. The down counter gets loaded with tPre. The state counter holds this state till the down-counter reaches the value 1 and waits for the activation command

When activation command is issued the state counter transitions to state 010 and loads the down-counter with the tCas and waits out the count down. Once count-down reaches 1, the TimeGen unit waits for the read or write command.

If read command is issued, the state counter proceeds to the read latency state (011) and waits out this period. If a write command is issued, the state counter jumps to burst state (100) directly from the activation state. The state counter will wait out the down-counter in each state as before.

Once the burst period expires the State counter progresses to wait state by itself and loads the down counter with the wait period.

The outputs of this module are the State counter value, Down Counter value and the TimerLd signal. Using these three outputs the Enable unit can generate the required control signals.

Verification table and waveforms of the TimeGen module is provided in the Section V.3.

b. Enable Unit

Enable unit of the SDRAM generates precharge, activate and Read/Write Enable signals for the memory core. Based on the outputs of the TimeGen, Enable unit issues a precharge signal to the memory core while transitioning to state 1. It issues activate signal to the memory core when the state counter output transitions to state 2. Similarly, in the burst

period the Enable unit issues Read/Write enable signals to the memory core.

The verification table and waveforms for the Enable unit is shown in the Section V.4.

c. Address Generation Unit

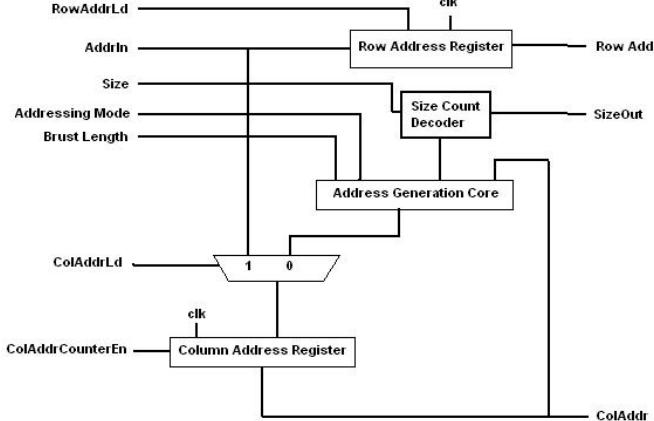


Fig 5. Address Generation Unit

This module generates the row and column addresses for the SDRAM Memory Core. It has three output ports and ten input ports. ‘RowAddr’ and ‘ColAddr’ are 8-bit wide output ports through which the Row and Column Addresses are sent out. ‘SizeOut’ is a 2-bit wide output port through which the size of data packet involved in the current transaction is sent out. ‘AddrIn’ is a 32-bit wide input port through which the row address and initial column address is provided to the Address Generation Unit. ‘RowAddrLd’, ‘ColAddrLd’, ‘ColCounterEn’, and ‘We’ are 1-bit input ports that provide control signals as required by the Address Generation Unit. ‘SizeIn’ is a 2-bit wide input port through which the size of the data packet is provided. ‘AddrMode’ is a 1-bit input port that specifies the addressing mode. ‘BurstLengthConfig’ is a 3-bit wide input port through which the Burst Length is provided to the Address Generation Unit. ‘Clk’ and ‘Rst’ are input signals that provide system clock and reset signals.

The block diagram for the Address Generation Unit is shown in Fig 5. The module consists of an Address Generator Core unit, a set of registers and a Size Count Decoder.

When ‘RowAddrLd’ goes high, the least significant byte of the ‘AddrIn’ port is latched into the ‘row’ register. Similarly, when the ‘ColAddrLd’ goes high, the least significant byte of the ‘AddrIn’ port is latched into the ‘column’ register. The registers now hold the Row Address and Initial Column Address.

Address Generator Core takes in the latched column address, Burst Length, Size and Address Mode information and generates the next set of addresses based on these inputs.

In this design, eight burst length configurations and two addressing modes are supported. Table 3 shows the burst length configurations and their burst lengths along with the addressing modes.

Table 3. Burst Length Configuration and Addressing Modes

A[2]	A[1]	A[0]	Burst Length	A[3]	Addressing Mode
0	0	0	1 Word	0	Sequential
0	0	1	2 Words	1	Linear
0	1	0	4 Words		
0	1	1	8 Words		
1	0	0	16 Words		
1	0	1	32 Words		
1	1	0	64 Words		
1	1	1	Full Page		

In linear addressing mode, the column address is incremented in a linear fashion. As shown in Fig 6, for the start address 13, the addresses generated are in continuous increments.

In sequential addressing mode, the set of addresses generated are within a range of values, depending on the burst length and packet size. Fig 7. and Fig 8. shows the set of addresses generated in the sequential addressing mode for a burst length of 4 and 8. Depending on the burst length configuration, the carry generated out of the bit location specified by this configuration information on every increment of the address is ignored. Assuming that the packet size is a byte, the set of addresses generated for burst length 4 are 13, 14, 15 and 12. The carry out of the 2nd bit is ignored. Similarly, the set of addresses generated for a burst length of 8 are 13, 14, 15, 8, 9, 10, 11 and 12, with the carry generated out of the 3rd bit location being ignored.

Starting Address = 13, Burst Length = 4, Mode = Linear

A[9]	A[8]	A[7]	A[6]	A[5]	A[4]	A[3]	A[2]	A[1]	A[0]	=
0	0	0	0	0	0	1	1	0	1	13
0	0	0	0	0	0	1	1	1	0	14
0	0	0	0	0	0	1	1	1	1	15
0	0	0	0	0	1	0	0	0	0	16

Fig 6. Addresses in Linear Addressing Mode

Starting Address = 13, Burst Length = 4, Mode = Sequential

A[9]	A[8]	A[7]	A[6]	A[5]	A[4]	A[3]	A[2]	A[1]	A[0]	=
0	0	0	0	0	0	1	1	0	1	13
								+ 1		
0	0	0	0	0	0	1	1	1	0	14
							+ 1			
0	0	0	0	0	0	1	1	1	1	15
							+ 1			
0	0	0	0	0	0	0	1	1	0	12
							delete the carry bit			

Fig 7. Addresses in Sequential Addressing Mode and Burst 4

Starting Address = 13, Burst Length = 8, Mode = Sequential										
A[9]	A[8]	A[7]	A[6]	A[5]	A[4]	A[3]	A[2]	A[1]	A[0]	
0	0	0	0	0	0	1	1	0	1	= 13
										+ 1
0	0	0	0	0	0	1	1	1	0	= 14
										+ 1
0	0	0	0	0	0	1	1	1	1	= 15
										+ 1
0	0	0	0	0	0	1	0	0	0	= 8
										+ 1
delete the carry bit										
0	0	0	0	0	0	1	0	0	1	= 9
										+ 1
0	0	0	0	0	0	1	0	1	0	= 10
										+ 1
0	0	0	0	0	0	1	0	1	1	= 11
										+ 1
0	0	0	0	0	0	1	1	0	0	= 12

Fig 8. Addresses in Sequential Addressing Mode and Burst 8

ii. SDRAM Memory Core

The SDRAM Memory Core has one output port and eleven input ports. ‘DataOut’ is a 32-bit wide output port over which the SDRAM Memory Core sends data. ‘DataIn’ is a 32-bit wide input port through which SDRAM Memory Core takes in new data to be written into the memory element. ‘RowAddr’ and ‘ColAddr’ are 8-bit wide input ports that provide the Row and Column Addresses to the SDRAM Memory Core. ‘Size’ is a 2-bit wide input port that provides the size of each data packet being written into or read from the SDRAM Memory Core. Other 1 bit input ports such as ‘Precharge’, ‘Activate’, ‘RE’ and ‘WE’ provide signals that control the operations being carried out by the SDRAM Memory Core. ‘BS’ is a 1 bit input port that provides an active low enable signal. ‘clk’ and ‘reset’ input ports provide the system clock and reset signals to the SDRAM Memory Core.

In this design, the memory array in the SDRAM Memory Core is organized as a 2D array of 256 rows and 64 columns. Each array element holds one word (4 Bytes) of data. Thus, the total storage capacity of each memory array is equal to 64 Kbytes ($256 \times 64 \times 4$ Bytes). To address into each row, an 8-bit wide address is used. To address into each column, a 6-bit wide address is used. A 2-bit wide offset is used to select a particular byte within each word.

Along with the memory array, a separate buffer called the ‘Row Buffer’ is present. This buffer is as big as one row of the memory array and hence, is named so. In this design, the row buffer will have 64 columns, with each element being able to hold one word of data. As explained before in Section III.A and III.B, every operation carried out by the SDRAM Memory Core such as Precharging a Bank, Activating a Row, Reading from a Column element or Writing to a Column element, is centered around the Row Buffer.

When the ‘Precharge’ input signal goes high, the SDRAM Memory Core will initiate the pre-charging of the current bank. Based on the Row Address provided to the SDRAM Memory Core, the entire content of the row buffer is copied to the row of the memory array.

When the ‘Activate’ input signal goes high, the SDRAM Memory Core would initiate the activation of a row. Based on the Row Address provided to the SDRAM Memory Core, the

entire row (with all its column elements) would be copied to the row buffer.

When the ‘RE’ input signal goes high, the SDRAM Memory Core would initiate a read transaction. Based on the Column Address provided to the SDRAM Memory Core, a particular column is selected from the row buffer. Based on the ‘Size’ input, either one byte or one half-word or the whole word is sent out through the ‘DataOut’ port. The first 2 bits from the 8-bit Column Address is used to decide as to which byte/half word is to be sent out.

When the ‘WE’ input signal goes high, the SDRAM Memory Core would initiate a write transaction. Based on the Column Address provided to the SDRAM Memory Core, a particular column is selected from the row buffer. The data input is given through the ‘DataIn’ port. Based on the ‘Size’ input, either one byte or one half-word or the whole word is written into the selected column. The first 2 bits from the 8-bit Column Address is used to decide as to which byte/half word is to be modified in the word.

As long as the BS input is low, the control signals ‘Precharge’ and ‘Activate’ will trigger the respective operations in the SDRAM Memory Core. ‘RE’ and ‘WE’ signals will trigger the read and write operations, whatever be the BS input.

The test framework, verification table and waveforms are as shown in Section V.6.

IV. BUS INTERFACE UNIT

A. Overview

The bus interface unit forms the bridge between the bus master and the SDRAM device protocols. The control and data signals issued by the master are as per the unidirectional bus protocol. The BIU emulates a unidirectional bus slave device while generating control signals for the SDRAM.

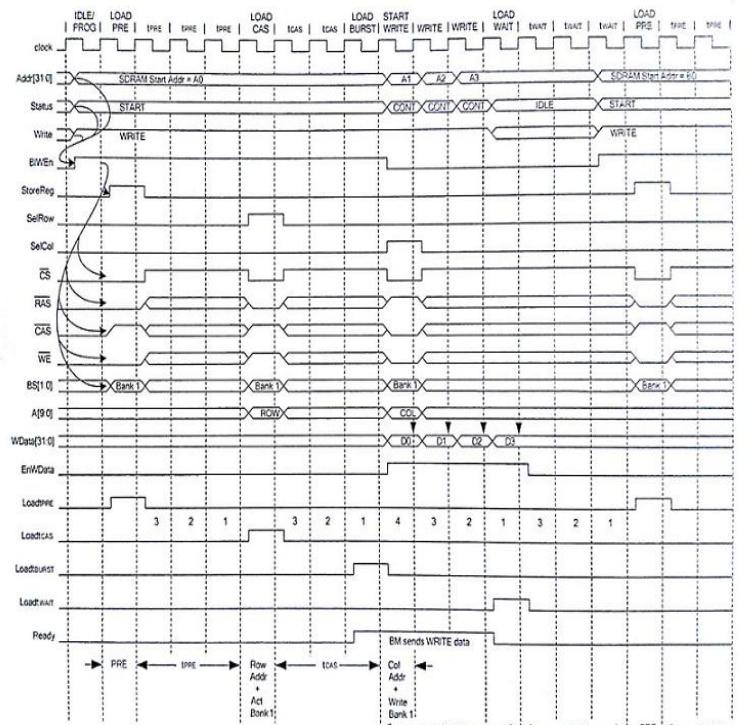


Fig 9. Single Write Cycle with BIU

For each transaction that the master requests, the BIU issues pre-charge, activate, read and write commands for the SDRAM device by generating the CS, RAS, CAS and WE signals. A single read and write transaction involving the BIU is shown in Fig 9. and Fig 10.

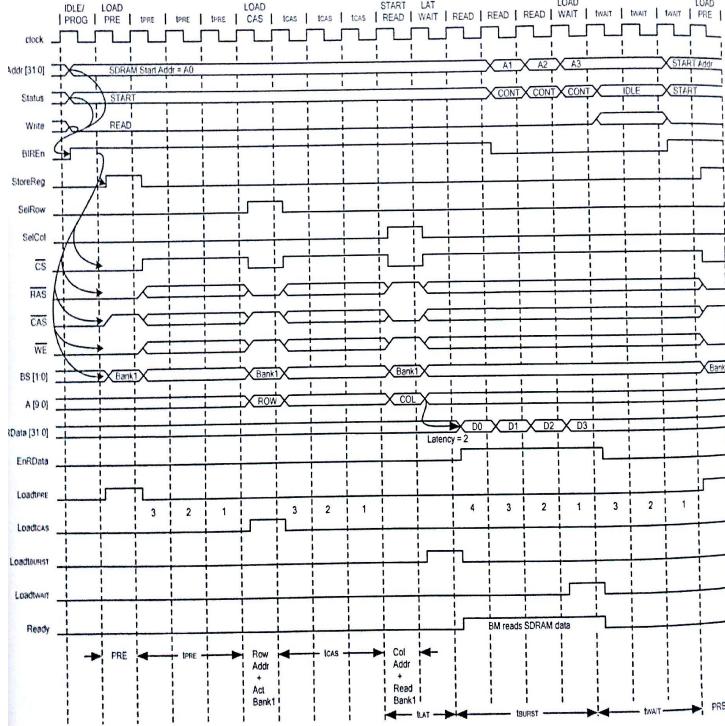


Fig 10. Single Read Cycle with BIU

B. BIU Controller

I. Register File

The BIU register file is exactly the same module as in the SDRAM register file in section III.C.1

The difference here is that the input control signal to the module is the master address line. When master issues an address 0xFFFFFFFF, the BIU expects the program register data to be available on the Data line in the next clock cycle. This is in compliance with the bus protocol. Along with this data the BIU controller issues a program command to the SDRAM device with the same timing configuration.

The verification table for this unit is provided in Section V.11.

2. Timing Generator

The timing generator module in the BIU is the same as the TimeGen in the SDRAM controller in section III.3.i.A. The state counter and the down counter follow the exact same design.

The important difference here is that the progression of the state counter is self-timed, i.e., it does not depend on any specific signals from the master. As soon as master issues a start command the Timing generator transitions to state 1 and prompts the BIU controller to issue the pre-charge signal to the SDRAM device. Similarly, it issues the activate, read/write commands following the timing specifications from the program file.

The test framework, verification table and waveform of the timing generator is shown in Section V.10.

The controller ensures that the Ready signal is high until a transaction is triggered by the master. On receiving a START signal from the master, the BIU puts its ready signal down and issues pre-charge and activate commands to the SDRAM. Once in the burst-phase the BIU sets the Ready signal high so that the master can issue or read the required data.

The glue logic within the BIU also latches the address and size issued by the master at the start of every transaction. The Row Address, Column Address, BS and Size signals are derived from these latched signals while transacting with the SDRAM.

V. WAVEFORMS AND VERIFICATION TABLES

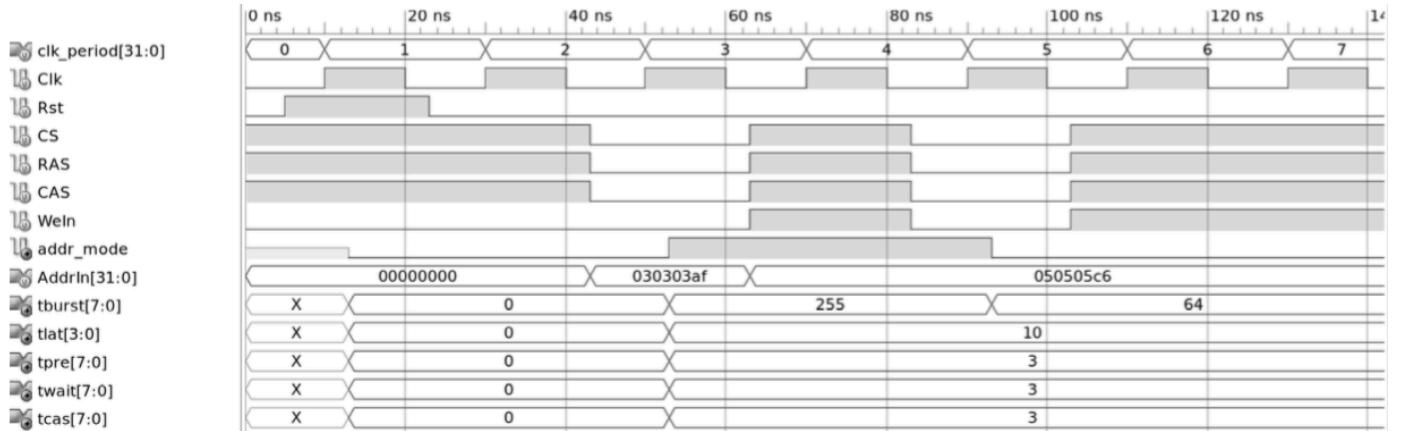
1. Register File

The verification table for the SDRAM register file module is shown below. Clock period 1 provides a reset signal causing the register file to reset to 0 in clock period 2.

First programming mode command is issued in clock period 3 (CS, RAS, CAS and WE are pulled low) with program data on AddrIn. This data latches in correctly as seen in clock period 4. Clock period 5 issues another programming command. This effects only the tburst and addr_mode values as expected from the design.

The waveform for the transactions are provided below.

Clk_period	Rst	addr_mode	tburst	tpe	tlat	tcas	twait	AddrIn	CS	RAS	CAS	WeIn
1	1	x	x	x	x	x	x	0x00000000	1	1	1	0
2	0	0	0	0	0	0	0	0x00000000	1	1	1	0
3	0	0	0	0	0	0	0	0x030303af	0	0	0	0
4	0	1	255	3	10	3	3	0x050505c6	1	1	1	1
5	0	1	255	3	10	3	3	0x050505c6	0	0	0	0
6	0	0	64	3	10	3	3	0x050505c6	1	1	1	1
7	0	0	64	3	10	3	3	0x050505c6	1	1	1	1



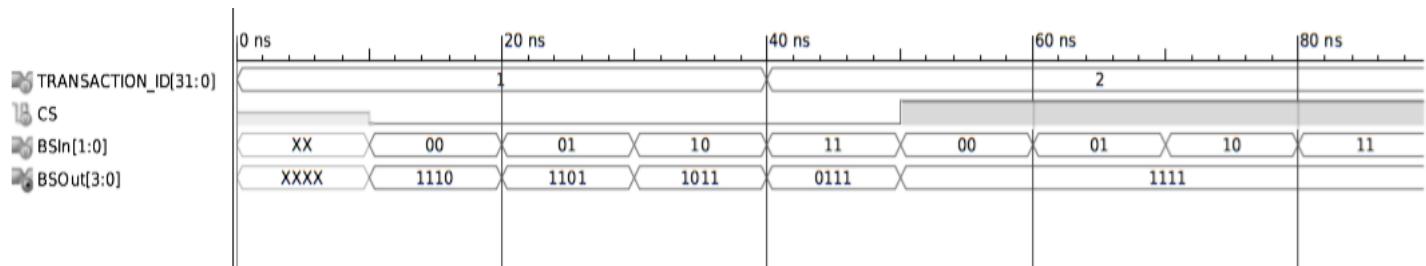
2. Bank Select Decoder

The verification table for Bank Select Decoder module and its waveforms are given below:

CS	BSIn	BSOut
0	2'b00	4'bxxxx
0	2'b01	4'b1110
0	2'b10	4'b1101
0	2'b11	4'b1011
1	2'b00	4'b0111
1	2'b01	4'b1111
1	2'b10	4'b1111
1	2'b11	4'b1111

The input values have been sampled when the inputs change. Due to propagation delay, the results appear in the next sample as seen in the verification table. When CS is low, one of the BSOut bits go low for different BSIn values.

When CS is high, none of the BSOut values are low. This unit enables a particular bank with active low enable input and hence, only one of the banks are selected when CS is held low.



3. SDRAM Timegen

Verification of the timing generator is provided in the table below.
System is reset in the first clock period.

The down counter and the state counter resets to state 0.

First transaction starts in clock period 2. As seen from the control signals, precharge signal is issued.

State counter proceeds to state 1 (pre-charge state), Down counter is loaded with tPre. Once the down-count completes activation command is issued in clock-period 8.

State counter transitions to state 2 (activation) and down counter is loaded with tCas in clock period 9.

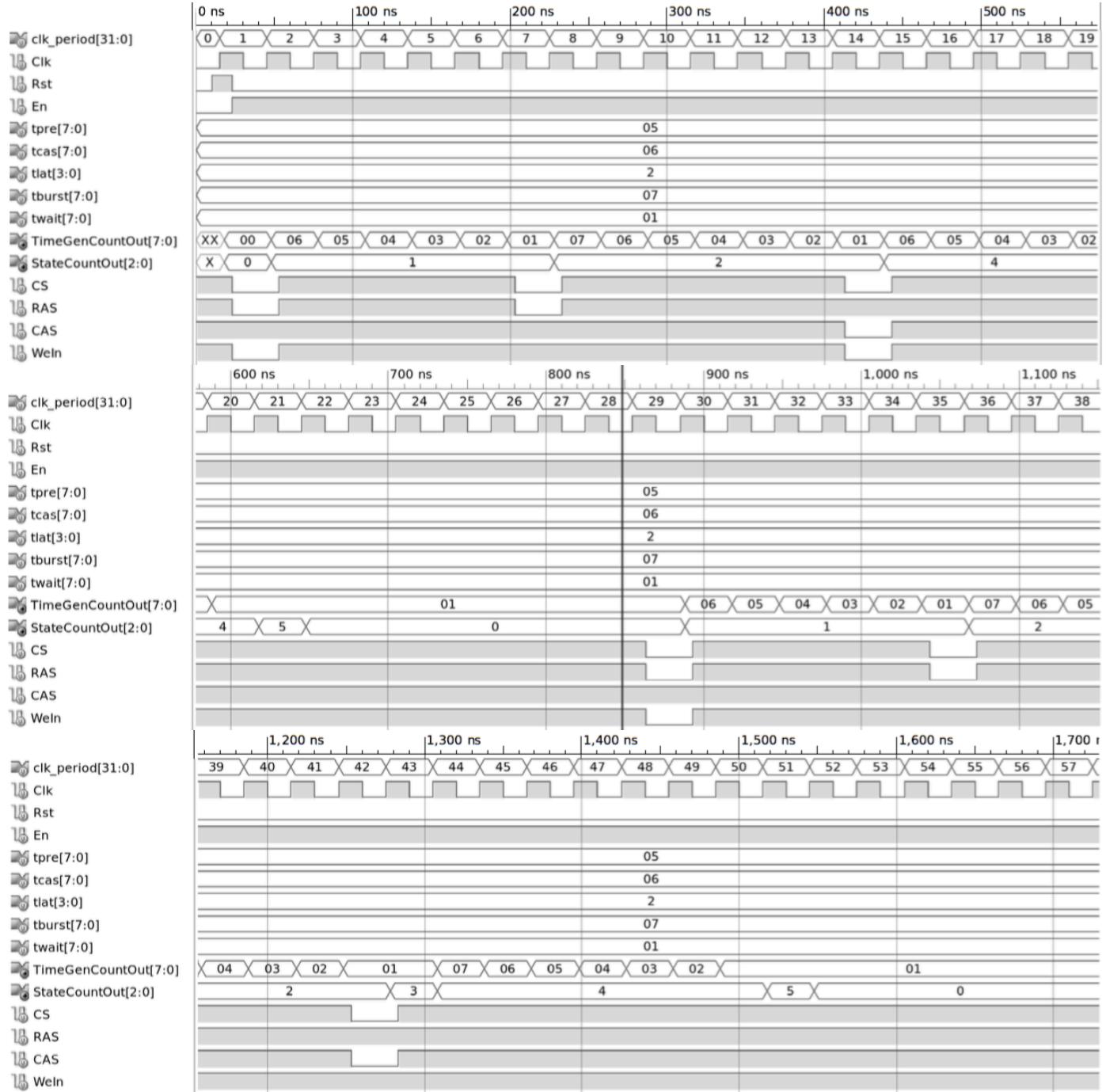
Once the down counter reaches 1 in clock period 15, a write command is issued. In the next clock period the state counter proceeds to state 4 (Burst period) and the down counter gets loaded with the burst value. Note that since this is a write transaction the burst count loaded into the down counter is one less than the tBurst value. This is because the first packet of data will be available with the command.

After the write burst is completed in clock period 21, the state counter proceeds to the wait state and down counts the wait period. Wait period gets completed in clock period 23.

Clock period 30 issues another pre-charge command beginning another transaction. The rest of the verification is available from the table and the waveforms below.

Fixed inputs: TPRE = 05;TCAS = 06;TBURST = 07;TWAIT = 01							
Clk_period = 1	Rst = 1	TimeGenCountOut = xx	StateCountOut= x	CS = 1	RAS = 1	CAS = 1	WeIn = 1
Clk_period = 2	Rst = 0	TimeGenCountOut = 00	StateCountOut= 0	CS = 0	RAS = 0	CAS = 1	WeIn = 0
Clk_period = 3	Rst = 0	TimeGenCountOut = 06	StateCountOut= 1	CS = 1	RAS = 1	CAS = 1	WeIn = 1
Clk_period = 4	Rst = 0	TimeGenCountOut = 05	StateCountOut= 1	CS = 1	RAS = 1	CAS = 1	WeIn = 1
Clk_period = 5	Rst = 0	TimeGenCountOut = 04	StateCountOut= 1	CS = 1	RAS = 1	CAS = 1	WeIn = 1
Clk_period = 6	Rst = 0	TimeGenCountOut = 03	StateCountOut= 1	CS = 1	RAS = 1	CAS = 1	WeIn = 1
Clk_period = 7	Rst = 0	TimeGenCountOut = 02	StateCountOut= 1	CS = 1	RAS = 1	CAS = 1	WeIn = 1
Clk_period = 8	Rst = 0	TimeGenCountOut = 01	StateCountOut= 1	CS = 0	RAS = 0	CAS = 1	WeIn = 1
Clk_period = 9	Rst = 0	TimeGenCountOut = 07	StateCountOut= 2	CS = 1	RAS = 1	CAS = 1	WeIn = 1
Clk_period = 10	Rst = 0	TimeGenCountOut = 06	StateCountOut= 2	CS = 1	RAS = 1	CAS = 1	WeIn = 1
Clk_period = 11	Rst = 0	TimeGenCountOut = 05	StateCountOut= 2	CS = 1	RAS = 1	CAS = 1	WeIn = 1
Clk_period = 12	Rst = 0	TimeGenCountOut = 04	StateCountOut= 2	CS = 1	RAS = 1	CAS = 1	WeIn = 1
Clk_period = 13	Rst = 0	TimeGenCountOut = 03	StateCountOut= 2	CS = 1	RAS = 1	CAS = 1	WeIn = 1
Clk_period = 14	Rst = 0	TimeGenCountOut = 02	StateCountOut= 2	CS = 1	RAS = 1	CAS = 1	WeIn = 1
Clk_period = 15	Rst = 0	TimeGenCountOut = 01	StateCountOut= 2	CS = 0	RAS = 1	CAS = 0	WeIn = 0
Clk_period = 16	Rst = 0	TimeGenCountOut = 06	StateCountOut= 4	CS = 1	RAS = 1	CAS = 1	WeIn = 1
Clk_period = 17	Rst = 0	TimeGenCountOut = 05	StateCountOut= 4	CS = 1	RAS = 1	CAS = 1	WeIn = 1
Clk_period = 18	Rst = 0	TimeGenCountOut = 04	StateCountOut= 4	CS = 1	RAS = 1	CAS = 1	WeIn = 1
Clk_period = 19	Rst = 0	TimeGenCountOut = 03	StateCountOut= 4	CS = 1	RAS = 1	CAS = 1	WeIn = 1
Clk_period = 20	Rst = 0	TimeGenCountOut = 02	StateCountOut= 4	CS = 1	RAS = 1	CAS = 1	WeIn = 1
Clk_period = 21	Rst = 0	TimeGenCountOut = 01	StateCountOut= 4	CS = 1	RAS = 1	CAS = 1	WeIn = 1
Clk_period = 22	Rst = 0	TimeGenCountOut = 01	StateCountOut= 5	CS = 1	RAS = 1	CAS = 1	WeIn = 1
Clk_period = 23	Rst = 0	TimeGenCountOut = 01	StateCountOut= 0	CS = 1	RAS = 1	CAS = 1	WeIn = 1
Clk_period = 24	Rst = 0	TimeGenCountOut = 01	StateCountOut= 0	CS = 1	RAS = 1	CAS = 1	WeIn = 1
Clk_period = 25	Rst = 0	TimeGenCountOut = 01	StateCountOut= 0	CS = 1	RAS = 1	CAS = 1	WeIn = 1
Clk_period = 26	Rst = 0	TimeGenCountOut = 01	StateCountOut= 0	CS = 1	RAS = 1	CAS = 1	WeIn = 1
Clk_period = 27	Rst = 0	TimeGenCountOut = 01	StateCountOut= 0	CS = 1	RAS = 1	CAS = 1	WeIn = 1
Clk_period = 28	Rst = 0	TimeGenCountOut = 01	StateCountOut= 0	CS = 1	RAS = 1	CAS = 1	WeIn = 1

Clk_period = 29	Rst = 0	TimeGenCountOut = 01	StateCountOut= 0	CS = 1	RAS = 1	CAS = 1	WeIn = 1
Clk_period = 30	Rst = 0	TimeGenCountOut = 01	StateCountOut= 0	CS = 0	RAS = 0	CAS = 1	WeIn = 0
Clk_period = 31	Rst = 0	TimeGenCountOut = 06	StateCountOut= 1	CS = 1	RAS = 1	CAS = 1	WeIn = 1
Clk_period = 32	Rst = 0	TimeGenCountOut = 05	StateCountOut= 1	CS = 1	RAS = 1	CAS = 1	WeIn = 1
Clk_period = 33	Rst = 0	TimeGenCountOut = 04	StateCountOut= 1	CS = 1	RAS = 1	CAS = 1	WeIn = 1
Clk_period = 34	Rst = 0	TimeGenCountOut = 03	StateCountOut= 1	CS = 1	RAS = 1	CAS = 1	WeIn = 1
Clk_period = 35	Rst = 0	TimeGenCountOut = 02	StateCountOut= 1	CS = 1	RAS = 1	CAS = 1	WeIn = 1



4. SDRAM Enable Unit

The verification for the core memory enable unit is provided below.

This module generates the signals Precharge, Activate, Re, We and AddrGenEn based on the output of the timing generator unit.

After reset in clock period 1 the first transaction starts from clock period 3. The LdState, which is the state counter output from the timing generator, is in reset state and the TimerLd goes high commencing precharge. The ‘Precharge’ output signal is issued in the same clock period.

While LdState proceeds to state 2 between clock period 7 and 8, the “Activate” signal goes high, signaling the core memory to activate a row.

Between clock period 13 and 14, the LdState transitions to burst mode. As latency is skipped, this is a write transaction. We can see that the We signal goes high in clock period 13, which is as expected and at the same time AddrGenEn goes high signaling the address generator to begin generating column addresses. A burst of 8 writes are carried out and We is pulled low in clock period 21. At this point the LdState goes into wait period.

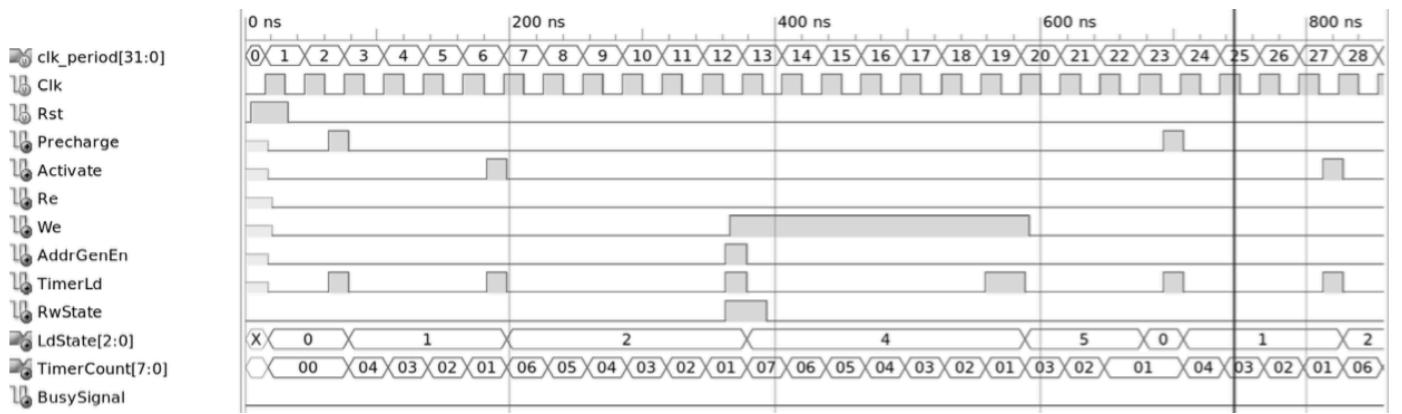
It can also be verified from the down counter output ‘TimerCount’ that the Precharge, Activate, Re, We and AddrGenEn signals are triggered at the correct time.

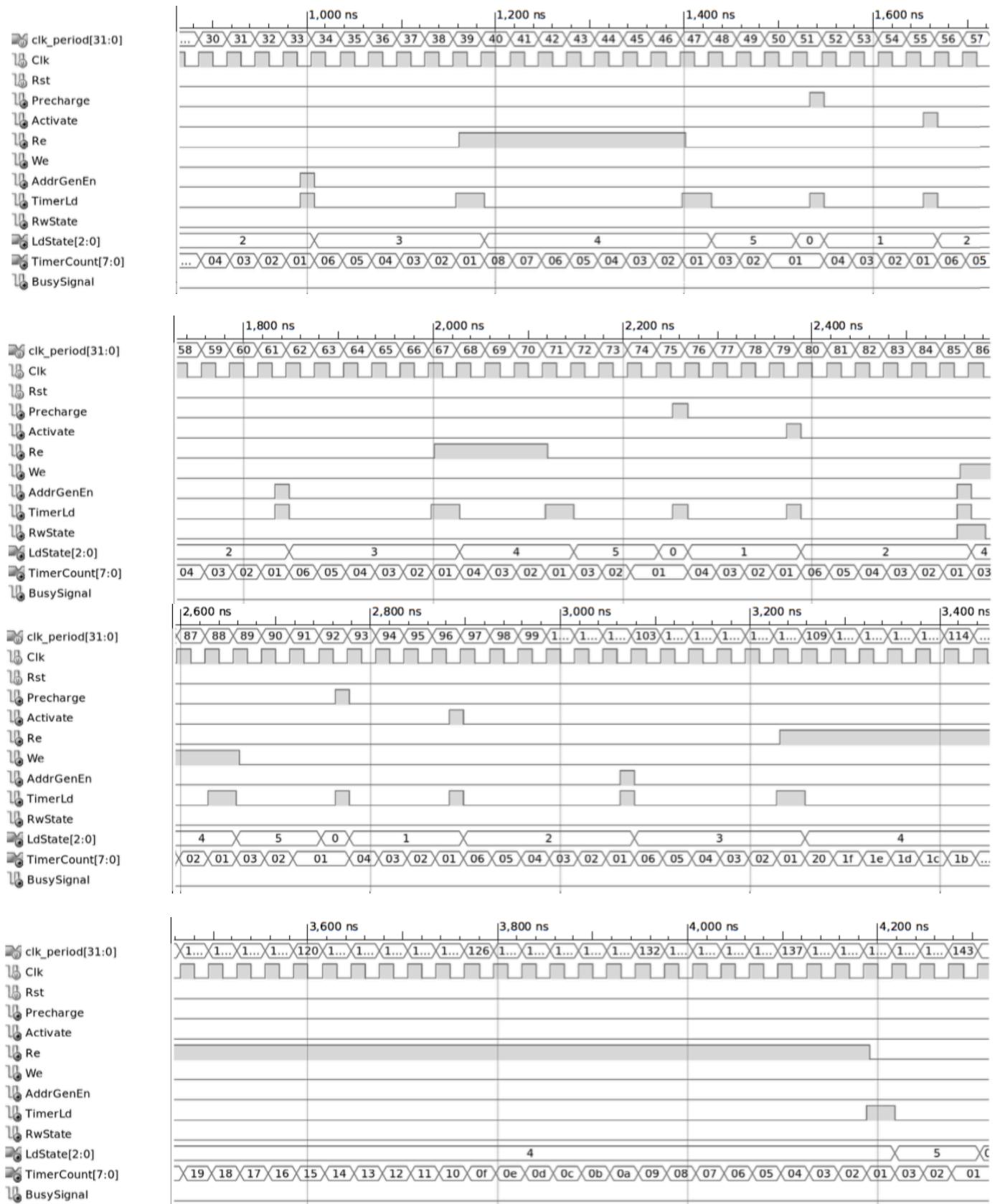
Clk Period	Reset	Precharge	Activate	Re	We	AddrGenEn	TimerLd	RwState	LdState	TimerCount	BusySignal
1	1	x	x	x	x	x	x	0	x	0xxx	0
2	0	0	0	0	0	0	0	0	0	0x00	0
3	0	1	0	0	0	0	1	0	0	0x00	0
4	0	0	0	0	0	0	0	0	1	0x04	0
5	0	0	0	0	0	0	0	0	1	0x03	0
6	0	0	0	0	0	0	0	0	1	0x02	0
7	0	0	1	0	0	0	1	0	1	0x01	0
8	0	0	0	0	0	0	0	0	2	0x06	0
9	0	0	0	0	0	0	0	0	2	0x05	0
10	0	0	0	0	0	0	0	0	2	0x04	0
11	0	0	0	0	0	0	0	0	2	0x03	0
12	0	0	0	0	0	0	0	0	2	0x02	0
13	0	0	0	0	1	1	1	1	2	0x01	0
14	0	0	0	0	1	0	0	0	4	0x07	0
15	0	0	0	0	1	0	0	0	4	0x06	0
16	0	0	0	0	1	0	0	0	4	0x05	0
17	0	0	0	0	1	0	0	0	4	0x04	0
18	0	0	0	0	1	0	0	0	4	0x03	0
19	0	0	0	0	1	0	0	0	4	0x02	0
20	0	0	0	0	1	0	1	0	4	0x01	0
21	0	0	0	0	0	0	0	0	5	0x03	0
22	0	0	0	0	0	0	0	0	5	0x02	0
23	0	0	0	0	0	0	0	0	5	0x01	0
24	0	1	0	0	0	0	1	0	0	0x01	0
25	0	0	0	0	0	0	0	0	1	0x04	0
26	0	0	0	0	0	0	0	0	1	0x03	0
27	0	0	0	0	0	0	0	0	1	0x02	0
28	0	0	1	0	0	0	1	0	1	0x01	0
29	0	0	0	0	0	0	0	0	2	0x06	0
30	n	n	n	n	n	n	n	n	?	nvn5	n

31	0	0	0	0	0	0	0	0	2	0x04	0
32	0	0	0	0	0	0	0	0	2	0x03	0
33	0	0	0	0	0	0	0	0	2	0x02	0
34	0	0	0	0	0	1	1	0	2	0x01	0
35	0	0	0	0	0	0	0	0	3	0x06	0
36	0	0	0	0	0	0	0	0	3	0x05	0
37	0	0	0	0	0	0	0	0	3	0x04	0
38	0	0	0	0	0	0	0	0	3	0x03	0
39	0	0	0	0	0	0	0	0	3	0x02	0
40	0	0	0	1	0	0	1	0	3	0x01	0
41	0	0	0	1	0	0	0	0	4	0x08	0
42	0	0	0	1	0	0	0	0	4	0x07	0
43	0	0	0	1	0	0	0	0	4	0x06	0
44	0	0	0	1	0	0	0	0	4	0x05	0
45	0	0	0	1	0	0	0	0	4	0x04	0
46	0	0	0	1	0	0	0	0	4	0x03	0
47	0	0	0	1	0	0	0	0	4	0x02	0
48	0	0	0	0	0	0	1	0	4	0x01	0
49	0	0	0	0	0	0	0	0	5	0x03	0
50	0	0	0	0	0	0	0	0	5	0x02	0
51	0	0	0	0	0	0	0	0	5	0x01	0
52	0	1	0	0	0	0	1	0	0	0x01	0
53	0	0	0	0	0	0	0	0	1	0x04	0
54	0	0	0	0	0	0	0	0	1	0x03	0
55	0	0	0	0	0	0	0	0	1	0x02	0
56	0	0	1	0	0	0	1	0	1	0x01	0
57	0	0	0	0	0	0	0	0	2	0x06	0
58	0	0	0	0	0	0	0	0	2	0x05	0
59	0	0	0	0	0	0	0	0	2	0x04	0
60	0	0	0	0	0	0	0	0	2	0x03	0
61	0	0	0	0	0	0	0	0	2	0x02	0
62	0	0	0	0	0	1	1	0	2	0x01	0
63	0	0	0	0	0	0	0	0	3	0x06	0
64	0	0	0	0	0	0	0	0	3	0x05	0
65	0	0	0	0	0	0	0	0	3	0x04	0
66	0	0	0	0	0	0	0	0	3	0x03	0
67	0	0	0	0	0	0	0	0	3	0x02	0
68	0	0	0	1	0	0	1	0	3	0x01	0
69	0	0	0	1	0	0	0	0	4	0x04	0
70	0	0	0	1	0	0	0	0	4	0x03	0
71	0	0	0	1	0	0	0	0	4	0x02	0
72	0	0	0	0	0	0	1	0	4	0x01	0

74	0	0	0	0	0	0	0	0	5	0x02	0
75	0	0	0	0	0	0	0	0	5	0x01	0
76	0	1	0	0	0	0	1	0	0	0x01	0
77	0	0	0	0	0	0	0	0	1	0x04	0
78	0	0	0	0	0	0	0	0	1	0x03	0
79	0	0	0	0	0	0	0	0	1	0x02	0
80	0	0	1	0	0	0	1	0	1	0x01	0
81	0	0	0	0	0	0	0	0	2	0x06	0
82	0	0	0	0	0	0	0	0	2	0x05	0
83	0	0	0	0	0	0	0	0	2	0x04	0
84	0	0	0	0	0	0	0	0	2	0x03	0
85	0	0	0	0	0	0	0	0	2	0x02	0
86	0	0	0	0	1	1	1	1	2	0x01	0
87	0	0	0	0	1	0	0	0	4	0x03	0
88	0	0	0	0	1	0	0	0	4	0x02	0
89	0	0	0	0	1	0	1	0	4	0x01	0
90	0	0	0	0	0	0	0	0	5	0x03	0
91	0	0	0	0	0	0	0	0	5	0x02	0
92	0	0	0	0	0	0	0	0	5	0x01	0
93	0	1	0	0	0	0	1	0	0	0x01	0
94	0	0	0	0	0	0	0	0	1	0x04	0
95	0	0	0	0	0	0	0	0	1	0x03	0
96	0	0	0	0	0	0	0	0	1	0x02	0
97	0	0	1	0	0	0	1	0	1	0x01	0
98	0	0	0	0	0	0	0	0	2	0x06	0
99	0	0	0	0	0	0	0	0	2	0x05	0
100	0	0	0	0	0	0	0	0	2	0x04	0
101	0	0	0	0	0	0	0	0	2	0x03	0
102	0	0	0	0	0	0	0	0	2	0x02	0
103	0	0	0	0	0	1	1	0	2	0x01	0
104	0	0	0	0	0	0	0	0	3	0x06	0
105	0	0	0	0	0	0	0	0	3	0x05	0
106	0	0	0	0	0	0	0	0	3	0x04	0
107	0	0	0	0	0	0	0	0	3	0x03	0
108	0	0	0	0	0	0	0	0	3	0x02	0
109	0	0	0	1	0	0	1	0	3	0x01	0
110	0	0	0	1	0	0	0	0	4	0x20	0
111	0	0	0	1	0	0	0	0	4	0x1f	0
112	0	0	0	1	0	0	0	0	4	0x1e	0
113	0	0	0	1	0	0	0	0	4	0x1d	0
114	0	0	0	1	0	0	0	0	4	0x1c	0
115	0	0	0	1	0	0	0	0	4	0x1b	0
116	0	0	0	1	0	0	0	0	4	0x1a	0

117	0	0	0	1	0	0	0	0	4	0x19	0
118	0	0	0	1	0	0	0	0	4	0x18	0
119	0	0	0	1	0	0	0	0	4	0x17	0
120	0	0	0	1	0	0	0	0	4	0x16	0
121	0	0	0	1	0	0	0	0	4	0x15	0
122	0	0	0	1	0	0	0	0	4	0x14	0
123	0	0	0	1	0	0	0	0	4	0x13	0
124	0	0	0	1	0	0	0	0	4	0x12	0
125	0	0	0	1	0	0	0	0	4	0x11	0
126	0	0	0	1	0	0	0	0	4	0x10	0
127	0	0	0	1	0	0	0	0	4	0x0f	0
128	0	0	0	1	0	0	0	0	4	0x0e	0
129	0	0	0	1	0	0	0	0	4	0x0d	0
130	0	0	0	1	0	0	0	0	4	0x0c	0
131	0	0	0	1	0	0	0	0	4	0x0b	0
132	0	0	0	1	0	0	0	0	4	0x0a	0
133	0	0	0	1	0	0	0	0	4	0x09	0
134	0	0	0	1	0	0	0	0	4	0x08	0
135	0	0	0	1	0	0	0	0	4	0x07	0
136	0	0	0	1	0	0	0	0	4	0x06	0
137	0	0	0	1	0	0	0	0	4	0x05	0
138	0	0	0	1	0	0	0	0	4	0x04	0
139	0	0	0	1	0	0	0	0	4	0x03	0
140	0	0	0	1	0	0	0	0	4	0x02	0
141	0	0	0	0	0	0	1	0	4	0x01	0
142	0	0	0	0	0	0	0	0	5	0x03	0
143	0	0	0	0	0	0	0	0	5	0x02	0
144	0	0	0	0	0	0	0	0	5	0x01	0





5. SDRAM Address Generation Unit

The verification table for SDRAM Address Generation Unit and its waveforms are given below:

TID	CK Period	Rst	RowAddr Ld	RowAddr	ColAddr Ld	ColAddr	SizeIn	Addr Mode	Burst Length Config	We	Size Out
0	0	1	0	xx	0	xx	0	0	0	0	x
0	1	0	0	0	0	0	0	0	0	0	0
1	2	0	0	0	0	0	0	0	0	0	0
2	3	0	1	ad	0	0	0	0	0	0	0
3	4	0	1	dd	0	0	0	0	0	0	0
4	5	0	0	dd	0	0	0	0	0	0	0
5	6	0	0	dd	1	a0	2	1	3	1	2
5	7	0	0	dd	0	a4	2	1	3	1	2
5	8	0	0	dd	0	a8	2	1	3	1	2
5	9	0	0	dd	0	ac	2	1	3	1	2
5	10	0	0	dd	0	b0	2	1	3	1	2
5	11	0	0	dd	0	b4	2	1	3	1	2
5	12	0	0	dd	0	b8	2	1	3	1	2
5	13	0	0	dd	0	bc	2	1	3	1	2
5	14	0	0	dd	0	bc	2	1	3	0	2
6	15	0	0	dd	1	a5	1	0	3	0	1
6	16	0	0	dd	0	a5	1	0	3	0	1
6	17	0	0	dd	0	a5	1	0	3	0	1
6	18	0	0	dd	0	a5	1	0	3	0	1
6	19	0	0	dd	0	a7	1	0	3	0	1
6	20	0	0	dd	0	a1	1	0	3	0	1
6	21	0	0	dd	0	a3	1	0	3	0	1
6	22	0	0	dd	0	a5	1	0	3	0	1
6	23	0	0	dd	0	a7	1	0	3	0	1
6	24	0	0	dd	0	a1	1	0	3	0	1
6	25	0	0	dd	0	a3	1	0	3	0	1
7	26	0	0	dd	1	a5	0	0	3	0	0
7	27	0	0	dd	0	a5	0	0	3	0	0
7	28	0	0	dd	0	a5	0	0	3	0	0
7	29	0	0	dd	0	a5	0	0	3	0	0
7	30	0	0	dd	0	a6	0	0	3	0	0
7	31	0	0	dd	0	a7	0	0	3	0	0
7	32	0	0	dd	0	a0	0	0	3	0	0
7	33	0	0	dd	0	a1	0	0	3	0	0
7	34	0	0	dd	0	a1	0	0	3	0	0
7	35	0	0	dd	0	a1	0	0	3	0	0
7	36	0	0	dd	0	a1	0	0	3	0	0
7	37	0	0	dd	0	a1	0	0	3	0	0
7	38	0	0	dd	0	a2	0	0	3	0	0
7	39	0	0	dd	0	a3	0	0	3	0	0
7	40	0	0	dd	0	a4	0	0	3	0	0
7	41	0	0	dd	0	a4	0	0	3	0	0

In Transaction with TID 0 a system reset is done.

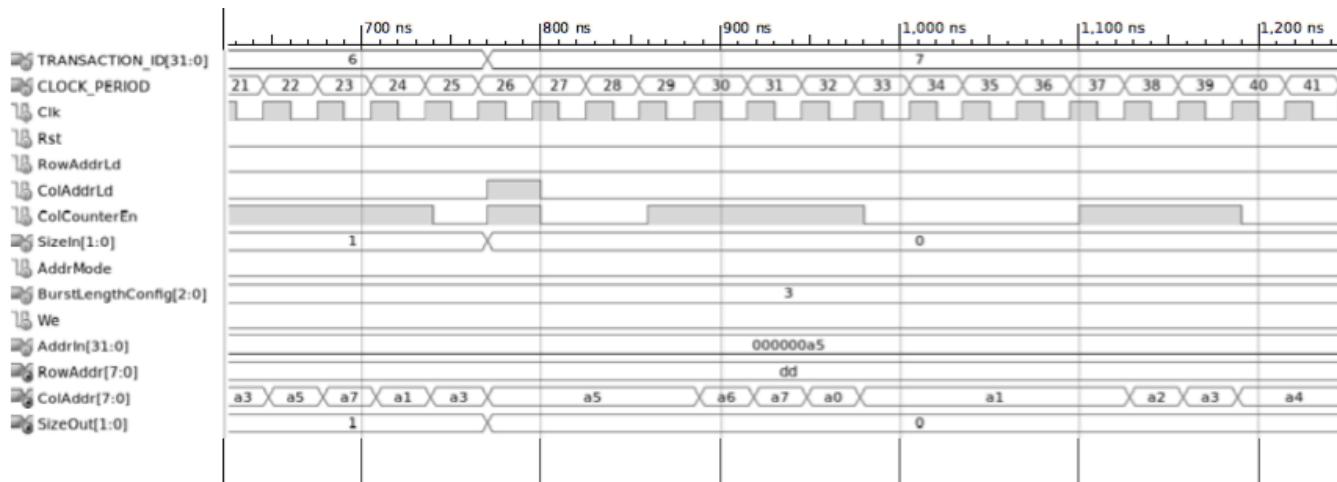
In Transaction with TID 1, a row address is provided through the AddrIn line with RowAddrLd signal kept low. The address should not get latched into the Row address register. This behavior is seen here as the RowAddr output port is still 0. In Transaction with TID 2, a row address is provided through AddrIn line with RowAddrLd signal kept high. The address should get latched into the Row address register. This behavior is observed here as the RowAddr output port is programmed with 0xAB. In Transaction with TID 3, a row address 0xABCDfedD is provided through AddrIn line with RowAddrLd signal kept high. Only the least significant byte should get latched into the Row address register. This behavior is observed here as the RowAddr output port is programmed with 0xDD.

In Transaction with TID 4, a column address is provided through the AddrIn line with ColAddrLd signal kept low. The address should not get latched into the Column address register. This behaviour is seen here as the ColAddr output port is not programmed with 0xA0. In the next clock period, a column address is provided through the AddrIn line with ColAddrLd signal kept high and the address should get latched into the Column address register. This behaviour is seen here as the ColAddr output port is programmed with 0xA0.

In Transaction with TID 5 onwards, as SizeIn is 2(Word), Burst Length Config is 3(8 Bursts) and address Mode is linear, the set addresses that should be generated are 0xa0, 0xa4, 0xa8, 0xaC, 0xb0, 0xb4, 0xb8, 0xbc. This behavior is observed here.

In Transaction with TID 6, the address generation is stalled for 2 clock cycles, simulating a ‘Master Busy’ period. The ColCounterEn is pulled low, thus halting the address generation. As the Size In is 1(Bytes), Burst length Config is 3(8 Bursts) and addressing mode is sequential, the set of addresses generated should loop through 0xa5, 0xa7, 0xa1, 0xa3 twice. This behavious is observed here.





6. SDRAM Memory Core

The verification tables and waveforms for unit testing of SDRAM Memory Core are shown below:

Clk Period	Reset	Precharge	Activate	RE	WE	BS	Size	Row Addr	Col Addr	DataIn	DataOut
0	1	z	z	z	z	1	z	zz	zz	0	zzzzzzzz
1	0	z	z	z	z	1	z	zz	zz	0	zzzzzzzz
2	0	1	0	0	0	0	z	zz	zz	0	zzzzzzzz
3	0	0	0	0	0	1	z	zz	zz	0	zzzzzzzz
4	0	0	0	0	0	1	z	zz	zz	0	zzzzzzzz
5	0	0	0	0	0	1	z	zz	zz	0	zzzzzzzz
6	0	0	1	0	0	0	z	0	zz	0	zzzzzzzz
7	0	0	0	0	0	1	z	0	zz	0	zzzzzzzz
8	0	0	0	0	0	1	z	0	zz	0	zzzzzzzz
9	0	0	0	0	0	1	z	0	zz	0	zzzzzzzz
10	0	0	0	0	1	1	0	0	0	11223344	zzzzzzzz
11	0	0	0	0	1	1	0	0	1	abababab	zzzzzzzz
12	0	0	0	0	0	1	0	0	1	zzzzzzzz	zzzzzzzz
13	0	0	0	0	0	1	0	0	1	zzzzzzzz	zzzzzzzz
14	0	0	0	0	0	1	0	0	1	zzzzzzzz	zzzzzzzz
15	0	0	0	0	0	1	0	0	1	zzzzzzzz	zzzzzzzz
16	0	1	0	0	0	0	z	0	1	zzzzzzzz	zzzzzzzz
17	0	0	0	0	0	1	z	0	1	zzzzzzzz	zzzzzzzz
18	0	0	0	0	0	1	z	0	1	zzzzzzzz	zzzzzzzz
19	0	0	0	0	0	1	z	0	1	zzzzzzzz	zzzzzzzz
20	0	0	1	0	0	0	z	0	1	zzzzzzzz	zzzzzzzz
21	0	0	0	0	0	1	z	0	1	zzzzzzzz	zzzzzzzz
22	0	0	0	0	0	1	z	0	1	zzzzzzzz	zzzzzzzz
23	0	0	0	0	0	1	z	0	1	zzzzzzzz	zzzzzzzz
24	0	0	0	0	1	1	0	2	deadbeef	zzzzzzzz	zzzzzzzz
25	0	0	0	0	0	1	1	0	2	zzzzzzzz	zzzzzzzz
26	0	0	0	0	0	1	1	0	2	zzzzzzzz	zzzzzzzz
27	0	0	0	0	0	1	1	0	2	zzzzzzzz	zzzzzzzz
28	0	0	0	0	0	1	1	0	2	zzzzzzzz	zzzzzzzz
29	0	1	0	0	0	0	z	0	2	zzzzzzzz	zzzzzzzz
30	0	0	0	0	0	1	z	0	2	zzzzzzzz	zzzzzzzz
31	0	0	0	0	0	1	z	0	2	zzzzzzzz	zzzzzzzz
32	0	0	0	0	0	1	z	0	2	zzzzzzzz	zzzzzzzz
33	0	0	1	0	0	0	z	0	2	zzzzzzzz	zzzzzzzz
34	0	0	0	0	0	1	z	0	2	zzzzzzzz	zzzzzzzz
35	0	0	0	0	0	1	z	0	2	zzzzzzzz	zzzzzzzz
36	0	0	0	0	0	1	z	0	2	zzzzzzzz	zzzzzzzz
37	0	0	0	0	1	1	2	0	8	eeffeef	zzzzzzzz
38	0	0	0	0	1	1	2	0	0c	aabbbaab	zzzzzzzz
39	0	0	0	0	0	1	2	0	0c	zzzzzzzz	zzzzzzzz
40	0	0	0	0	0	1	2	0	0c	zzzzzzzz	zzzzzzzz
41	0	0	0	0	0	1	2	0	0c	zzzzzzzz	zzzzzzzz
42	0	0	0	0	0	1	2	0	0c	zzzzzzzz	zzzzzzzz
43	0	1	0	0	0	1	2	0	0c	zzzzzzzz	zzzzzzzz
44	0	0	0	0	0	1	z	0	0c	zzzzzzzz	zzzzzzzz
45	0	0	0	0	0	1	z	0	0c	zzzzzzzz	zzzzzzzz
46	0	0	0	0	0	1	z	0	0c	zzzzzzzz	zzzzzzzz
47	0	0	1	0	0	1	z	0	0c	zzzzzzzz	zzzzzzzz
48	0	0	0	0	0	1	z	0	0c	zzzzzzzz	zzzzzzzz
49	0	0	0	0	0	1	z	0	0c	zzzzzzzz	zzzzzzzz
50	0	0	0	0	0	1	z	0	0c	zzzzzzzz	zzzzzzzz
51	0	0	0	0	0	1	z	0	0c	zzzzzzzz	zzzzzzzz

52	0	0	0	0	0	0	1	z	0	0c	zzzzzzzz	zzzzzzzz
53	0	0	0	0	0	0	1	z	0	0c	zzzzzzzz	zzzzzzzz
54	0	0	0	0	1	0	1	0	0	0	zzzzzzzz	zzzzzzzz
55	0	0	0	0	1	0	1	0	0	1	zzzzzzzz	44
56	0	0	0	0	0	0	1	0	0	1	zzzzzzzz	000000ab
57	0	0	0	0	0	0	1	0	0	1	zzzzzzzz	000000ab
58	0	0	0	0	0	0	1	0	0	1	zzzzzzzz	000000ab
59	0	0	0	0	0	0	1	0	0	1	zzzzzzzz	000000ab
60	0	1	0	0	0	0	1	0	0	1	zzzzzzzz	000000ab
61	0	0	0	0	0	0	1	z	0	1	zzzzzzzz	000000ab
62	0	0	0	0	0	0	1	z	0	1	zzzzzzzz	000000ab
63	0	0	0	0	0	0	1	z	0	1	zzzzzzzz	000000ab
64	0	0	0	1	0	0	1	z	0	1	zzzzzzzz	000000ab
65	0	0	0	0	0	0	1	z	0	1	zzzzzzzz	000000ab
66	0	0	0	0	0	0	1	z	0	1	zzzzzzzz	000000ab
67	0	0	0	0	0	0	1	z	0	1	zzzzzzzz	000000ab
68	0	0	0	0	0	0	1	z	0	1	zzzzzzzz	000000ab
69	0	0	0	0	0	0	1	z	0	1	zzzzzzzz	000000ab
70	0	0	0	0	0	0	1	z	0	1	zzzzzzzz	000000ab
71	0	0	0	0	1	0	1	1	0	0	zzzzzzzz	000000ab
72	0	0	0	0	1	0	1	1	0	2	zzzzzzzz	0000ab44
73	0	0	0	0	1	0	1	1	0	4	zzzzzzzz	0000beef
74	0	0	0	1	0	0	1	1	0	6	zzzzzzzz	0000ffff
75	0	0	0	0	0	1	1	0	0	6	zzzzzzzz	0000ffff
76	0	0	0	0	0	0	1	1	0	6	zzzzzzzz	0000ffff
77	0	0	0	0	0	0	1	1	0	6	zzzzzzzz	0000ffff
78	0	0	0	0	0	0	1	1	0	6	zzzzzzzz	0000ffff
79	0	1	0	0	0	0	1	1	0	6	zzzzzzzz	0000ffff
80	0	0	0	0	0	0	1	z	0	6	zzzzzzzz	0000ffff
81	0	0	0	0	0	0	1	z	0	6	zzzzzzzz	0000ffff
82	0	0	0	0	0	0	1	z	0	6	zzzzzzzz	0000ffff
83	0	0	1	0	0	0	1	z	0	6	zzzzzzzz	0000ffff
84	0	0	0	0	0	0	1	z	0	6	zzzzzzzz	0000ffff
85	0	0	0	0	0	0	1	z	0	6	zzzzzzzz	0000ffff
86	0	0	0	0	0	0	1	z	0	6	zzzzzzzz	0000ffff
87	0	0	0	0	0	0	1	z	0	6	zzzzzzzz	0000ffff
88	0	0	0	0	0	0	1	z	0	6	zzzzzzzz	0000ffff
89	0	0	0	0	0	0	1	z	0	6	zzzzzzzz	0000ffff
90	0	0	0	1	0	0	1	2	0	8	zzzzzzzz	0000ffff
91	0	0	0	0	0	0	1	2	0	8	zzzzzzzz	eefeff
92	0	0	0	0	0	0	1	2	0	8	zzzzzzzz	eefeff
93	0	0	0	0	0	0	1	2	0	8	zzzzzzzz	eefeff

Transaction 1 is a Write, packet size Bytes, Burst of 2 into Row Address 8'b00 and Column Address 8'b00 and 8'b01.

Transaction 2 is a Write, packet size Half Word, Burst 1 into Row Address 8'b00 and Column Address 8'b02.

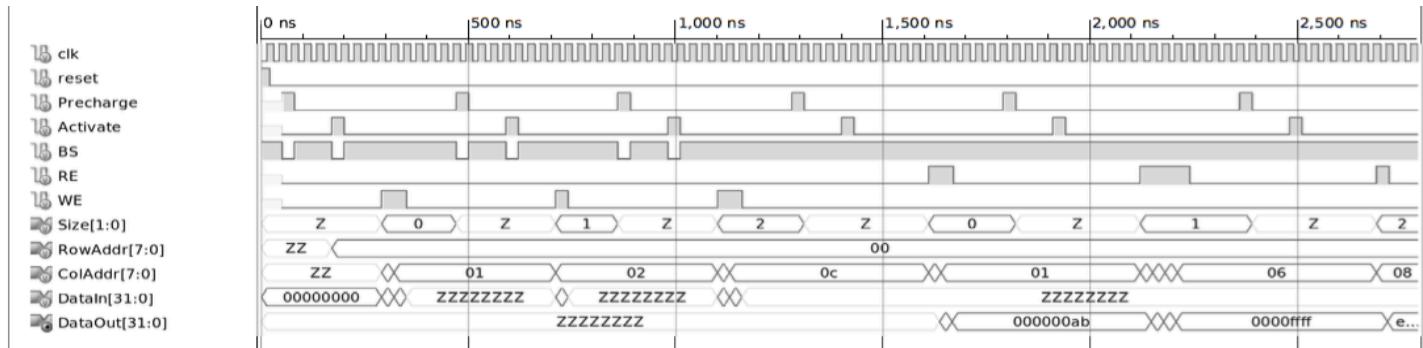
Transaction 3 is a Write, packet size Word, Burst 4 into Row Address 8'b00 and Column Address 8'b08, 8'b0C, 8'b10, 8'b14.

Transaction 4 is a Read, packet size Bytes, Burst 2 from Row Address 8'b00 and Column Address 8'b00, 8'b01.

Transaction 5 is a Read, packet size Half Words, Burst 4 from Row Address 8'b00 and Column Address 8'b00, 8'b02.

Transaction 6 is a Read, packet size Word, Burst 1 from Row Address 8'b00 and Column Address 8'b08 onwards.

The data written during all the write cycles are read back intact and 0x01 during clock cycles 10 and 11. When they are read during the read cycles. For example, bytes from back, we get 0x44 and 0xab in clock cycles 55 and 56. The data0x11223344 and 0xabababab are written into address 0x00 waveform is shown below.



7. SDRAM Bank

The verification tables and waveforms for unit testing of SDRAM Bank are shown below. A set of 14 transactions have been carried out to test the module as follows:

Transaction 1 is a Write, Size of Packet is Byte, Burst 1.

Transaction 2 is a Write, Half Words, Burst 2.

Transaction 3 is a Write, Words, Burst 4.

Transaction 4 is a Write, Bytes, Burst 8.

Transaction 5 is a Write, Half Words, Burst 16.

Transaction 6 is a Write, Words, Burst 32.

Transaction 7 is a Write, Words, Burst 64.

Transaction 8 is a Read, Bytes, Burst 1.

Transaction 9 is a Read, Half Words, Burst 2.

Transaction 10 is a Read, Words, Burst 4.

Transaction 11 is a Read, Words, Burst 8.

Transaction 12 is a Read, Words, Burst 16.

Transaction 13 is a Read, Words, Burst 32.

Transaction 14 is a Read, Words, Burst 64.

Page Burst read and writes have also been tested and are working as expected. Their results could not be placed into the report as they were too large.

First 7 transactions write data into particular addresses in the SDRAM. They are then read back in the next 7 transactions. Each of the operations tests the read and write functionalities for different bursts, sizes and addressing modes. The behaviours observed are as required.

TID	Clk Period	Reset	Tburst	TBurst Config	Addr Mode	TLat	TPre	TWait	TCas	CS	RAS	CAS	WE	AddrIn	SizeIn	DataIn	DataOut
0	0	1	0	0	1	4	3	3	3	1	1	1	1	0x00000000	0x0	0x00000000	0xxxxxxxxx
1	3	0	1	0	1	4	3	3	3	1	1	1	1	0x00000000	0x0	0x00000000	0xxxxxxxxx
1	4	0	1	0	1	4	3	3	3	1	1	1	1	0x00000000	0x0	0x00000000	0xxxxxxxxx
1	5	0	1	0	1	4	3	3	3	1	1	1	1	0x00000000	0x0	0x00000000	0xxxxxxxxx
1	6	0	1	0	1	4	3	3	3	0	0	1	1	0x00000000	0x0	0x00000000	0xxxxxxxxx
1	7	0	1	0	1	4	3	3	3	1	1	1	1	0x00000000	0x0	0x00000000	0xxxxxxxxx
1	8	0	1	0	1	4	3	3	3	1	1	1	1	0x00000000	0x0	0x00000000	0xxxxxxxxx
1	9	0	1	0	1	4	3	3	3	1	1	1	1	0x00000000	0x0	0x00000000	0xxxxxxxxx
1	10	0	1	0	1	4	3	3	3	0	1	0	0	0x00000000	0x0	0x110011ff	0xxxxxxxxx
1	11	0	1	0	1	4	3	3	3	1	1	1	1	0x00000000	0x0	0x00000000	0xxxxxxxxx
1	12	0	1	0	1	4	3	3	3	1	1	1	1	0x00000000	0x0	0x00000000	0xxxxxxxxx
1	13	0	1	0	1	4	3	3	3	1	1	1	1	0x00000000	0x0	0x00000000	0xxxxxxxxx
2	14	0	2	1	1	4	3	3	3	0	0	1	0	0x00000000	0x1	0x00000000	0xxxxxxxxx
2	15	0	2	1	1	4	3	3	3	1	1	1	1	0x00000000	0x1	0x00000000	0xxxxxxxxx
2	16	0	2	1	1	4	3	3	3	1	1	1	1	0x00000000	0x1	0x00000000	0xxxxxxxxx

2	17	0	2	1	1	4	3	3	3	1	1	1	1	0x00000000	0x1	0xxxxxxxxx	0xxxxxxxxx
2	18	0	2	1	1	4	3	3	3	0	0	1	1	0x00000002	0x1	0xxxxxxxxx	0xxxxxxxxx
2	19	0	2	1	1	4	3	3	3	1	1	1	1	0x00000002	0x1	0xxxxxxxxx	0xxxxxxxxx
2	20	0	2	1	1	4	3	3	3	1	1	1	1	0x00000002	0x1	0xxxxxxxxx	0xxxxxxxxx
2	21	0	2	1	1	4	3	3	3	1	1	1	1	0x00000002	0x1	0xxxxxxxxx	0xxxxxxxxx
2	22	0	2	1	1	4	3	3	3	0	1	0	0	0x00000002	0x1	0x00000001	0xxxxxxxxx
2	23	0	2	1	1	4	3	3	3	1	1	1	1	0x00000002	0x1	0x00000002	0xxxxxxxxx
2	24	0	2	1	1	4	3	3	3	1	1	1	1	0x00000002	0x1	0xxxxxxxxx	0xxxxxxxxx
2	25	0	2	1	1	4	3	3	3	1	1	1	1	0x00000002	0x1	0xxxxxxxxx	0xxxxxxxxx
2	26	0	2	1	1	4	3	3	3	1	1	1	1	0x00000002	0x1	0xxxxxxxxx	0xxxxxxxxx
3	27	0	4	2	1	4	3	3	3	0	0	1	0	0x00000002	0x2	0xxxxxxxxx	0xxxxxxxxx
3	28	0	4	2	1	4	3	3	3	1	1	1	1	0x00000002	0x2	0xxxxxxxxx	0xxxxxxxxx
3	29	0	4	2	1	4	3	3	3	1	1	1	1	0x00000002	0x2	0xxxxxxxxx	0xxxxxxxxx
3	30	0	4	2	1	4	3	3	3	1	1	1	1	0x00000002	0x2	0xxxxxxxxx	0xxxxxxxxx
3	31	0	4	2	1	4	3	3	3	0	0	1	1	0x00000000	0x2	0xxxxxxxxx	0xxxxxxxxx
3	32	0	4	2	1	4	3	3	3	1	1	1	1	0x00000000	0x2	0xxxxxxxxx	0xxxxxxxxx
3	33	0	4	2	1	4	3	3	3	1	1	1	1	0x00000000	0x2	0xxxxxxxxx	0xxxxxxxxx
3	34	0	4	2	1	4	3	3	3	1	1	1	1	0x00000000	0x2	0xxxxxxxxx	0xxxxxxxxx
3	35	0	4	2	1	4	3	3	3	0	1	0	0	0x00000009	0x2	0x000000a0	0xxxxxxxxx
3	36	0	4	2	1	4	3	3	3	1	1	1	1	0x00000009	0x2	0x000000a2	0xxxxxxxxx
3	37	0	4	2	1	4	3	3	3	1	1	1	1	0x00000009	0x2	0x000000a4	0xxxxxxxxx
3	38	0	4	2	1	4	3	3	3	1	1	1	1	0x00000009	0x2	0x000000a6	0xxxxxxxxx
3	39	0	4	2	1	4	3	3	3	1	1	1	1	0x00000009	0x2	0xxxxxxxxx	0xxxxxxxxx
3	40	0	4	2	1	4	3	3	3	1	1	1	1	0x00000009	0x2	0xxxxxxxxx	0xxxxxxxxx
3	41	0	4	2	1	4	3	3	3	1	1	1	1	0x00000009	0x2	0xxxxxxxxx	0xxxxxxxxx
4	42	0	8	3	0	4	3	3	3	0	0	1	0	0x00000009	0x0	0xxxxxxxxx	0xxxxxxxxx
4	43	0	8	3	0	4	3	3	3	1	1	1	1	0x00000009	0x0	0xxxxxxxxx	0xxxxxxxxx
4	44	0	8	3	0	4	3	3	3	1	1	1	1	0x00000009	0x0	0xxxxxxxxx	0xxxxxxxxx
4	45	0	8	3	0	4	3	3	3	1	1	1	1	0x00000009	0x0	0xxxxxxxxx	0xxxxxxxxx
4	46	0	8	3	0	4	3	3	3	0	0	1	1	0x00000000	0x0	0xxxxxxxxx	0xxxxxxxxx

4	47	0	8	3	0	4	3	3	3	1	1	1	1	0x00000000	0x0	0xxxxxxxx	0xxxxxxxx
4	48	0	8	3	0	4	3	3	3	1	1	1	1	0x00000000	0x0	0xxxxxxxx	0xxxxxxxx
4	49	0	8	3	0	4	3	3	3	1	1	1	1	0x00000000	0x0	0xxxxxxxx	0xxxxxxxx
4	50	0	8	3	0	4	3	3	3	0	1	0	0	0x000000d5	0x0	0x000000b0	0xxxxxxxx
4	51	0	8	3	0	4	3	3	3	1	1	1	1	0x000000d5	0x0	0x000000b0	0xxxxxxxx
4	52	0	8	3	0	4	3	3	3	1	1	1	1	0x000000d5	0x0	0x000000b0	0xxxxxxxx
4	53	0	8	3	0	4	3	3	3	1	1	1	1	0x000000d5	0x0	0x000000b0	0xxxxxxxx
4	54	0	8	3	0	4	3	3	3	1	1	1	1	0x000000d5	0x0	0x000000b0	0xxxxxxxx
4	55	0	8	3	0	4	3	3	3	1	1	1	1	0x000000d5	0x0	0x000000b0	0xxxxxxxx
4	56	0	8	3	0	4	3	3	3	1	1	1	1	0x000000d5	0x0	0x000000b0	0xxxxxxxx
4	57	0	8	3	0	4	3	3	3	1	1	1	1	0x000000d5	0x0	0x000000b0	0xxxxxxxx
4	58	0	8	3	0	4	3	3	3	1	1	1	1	0x000000d5	0x0	0xxxxxxxx	0xxxxxxxx
4	59	0	8	3	0	4	3	3	3	1	1	1	1	0x000000d5	0x0	0xxxxxxxx	0xxxxxxxx
4	60	0	8	3	0	4	3	3	3	1	1	1	1	0x000000d5	0x0	0xxxxxxxx	0xxxxxxxx
5	61	0	16	4	0	4	3	3	3	0	0	1	0	0x000000d5	0x1	0xxxxxxxx	0xxxxxxxx
5	62	0	16	4	0	4	3	3	3	1	1	1	1	0x000000d5	0x1	0xxxxxxxx	0xxxxxxxx
5	63	0	16	4	0	4	3	3	3	1	1	1	1	0x000000d5	0x1	0xxxxxxxx	0xxxxxxxx
5	64	0	16	4	0	4	3	3	3	1	1	1	1	0x000000d5	0x1	0xxxxxxxx	0xxxxxxxx
5	65	0	16	4	0	4	3	3	3	0	0	1	1	0x000000da	0x1	0xxxxxxxx	0xxxxxxxx
5	66	0	16	4	0	4	3	3	3	1	1	1	1	0x000000da	0x1	0xxxxxxxx	0xxxxxxxx
5	67	0	16	4	0	4	3	3	3	1	1	1	1	0x000000da	0x1	0xxxxxxxx	0xxxxxxxx
5	68	0	16	4	0	4	3	3	3	1	1	1	1	0x000000da	0x1	0xxxxxxxx	0xxxxxxxx
5	69	0	16	4	0	4	3	3	3	0	1	0	0	0x000000b6	0x1	0x000000b0	0xxxxxxxx
5	70	0	16	4	0	4	3	3	3	1	1	1	1	0x000000b6	0x1	0x000000b1	0xxxxxxxx
5	71	0	16	4	0	4	3	3	3	1	1	1	1	0x000000b6	0x1	0x000000b2	0xxxxxxxx
5	72	0	16	4	0	4	3	3	3	1	1	1	1	0x000000b6	0x1	0x000000b3	0xxxxxxxx
5	73	0	16	4	0	4	3	3	3	1	1	1	1	0x000000b6	0x1	0x000000b4	0xxxxxxxx
5	74	0	16	4	0	4	3	3	3	1	1	1	1	0x000000b6	0x1	0x000000b5	0xxxxxxxx
5	75	0	16	4	0	4	3	3	3	1	1	1	1	0x000000b6	0x1	0x000000b6	0xxxxxxxx
5	76	0	16	4	0	4	3	3	3	1	1	1	1	0x000000b6	0x1	0x000000b7	0xxxxxxxx

5	77	0	16	4	0	4	3	3	3	1	1	1	1	0x000000b6	0x1	0x000000b8	0xxxxxxxxx
5	78	0	16	4	0	4	3	3	3	1	1	1	1	0x000000b6	0x1	0x000000b9	0xxxxxxxxx
5	79	0	16	4	0	4	3	3	3	1	1	1	1	0x000000b6	0x1	0x000000ba	0xxxxxxxxx
5	80	0	16	4	0	4	3	3	3	1	1	1	1	0x000000b6	0x1	0x000000bb	0xxxxxxxxx
5	81	0	16	4	0	4	3	3	3	1	1	1	1	0x000000b6	0x1	0x000000bc	0xxxxxxxxx
5	82	0	16	4	0	4	3	3	3	1	1	1	1	0x000000b6	0x1	0x000000bd	0xxxxxxxxx
5	83	0	16	4	0	4	3	3	3	1	1	1	1	0x000000b6	0x1	0x000000be	0xxxxxxxxx
5	84	0	16	4	0	4	3	3	3	1	1	1	1	0x000000b6	0x1	0x000000bf	0xxxxxxxxx
5	85	0	16	4	0	4	3	3	3	1	1	1	1	0x000000b6	0x1	0x000000c0	0xxxxxxxxx
5	86	0	16	4	0	4	3	3	3	1	1	1	1	0x000000b6	0x1	0x000000c4	0xxxxxxxxx
5	87	0	16	4	0	4	3	3	3	1	1	1	1	0x000000b6	0x1	0x000000c8	0xxxxxxxxx
6	88	0	32	5	0	4	3	3	3	0	0	1	0	0x000000b6	0x2	0x000000c0	0xxxxxxxxx
6	89	0	32	5	0	4	3	3	3	1	1	1	1	0x000000b6	0x2	0x000000c4	0xxxxxxxxx
6	90	0	32	5	0	4	3	3	3	1	1	1	1	0x000000b6	0x2	0x000000c8	0xxxxxxxxx
6	91	0	32	5	0	4	3	3	3	1	1	1	1	0x000000b6	0x2	0x000000cc	0xxxxxxxxx
6	92	0	32	5	0	4	3	3	3	0	0	1	1	0x00000000	0x2	0x000000cd	0xxxxxxxxx
6	93	0	32	5	0	4	3	3	3	1	1	1	1	0x00000000	0x2	0x000000cf	0xxxxxxxxx
6	94	0	32	5	0	4	3	3	3	1	1	1	1	0x00000000	0x2	0x000000d0	0xxxxxxxxx
6	95	0	32	5	0	4	3	3	3	1	1	1	1	0x00000000	0x2	0x000000d4	0xxxxxxxxx
6	96	0	32	5	0	4	3	3	3	0	1	0	0	0x00000074	0x2	0x00000070	0xxxxxxxxx
6	97	0	32	5	0	4	3	3	3	1	1	1	1	0x00000074	0x2	0x00000072	0xxxxxxxxx
6	98	0	32	5	0	4	3	3	3	1	1	1	1	0x00000074	0x2	0x00000074	0xxxxxxxxx
6	99	0	32	5	0	4	3	3	3	1	1	1	1	0x00000074	0x2	0x00000076	0xxxxxxxxx
6	100	0	32	5	0	4	3	3	3	1	1	1	1	0x00000074	0x2	0x00000078	0xxxxxxxxx
6	101	0	32	5	0	4	3	3	3	1	1	1	1	0x00000074	0x2	0x0000007a	0xxxxxxxxx
6	102	0	32	5	0	4	3	3	3	1	1	1	1	0x00000074	0x2	0x0000007c	0xxxxxxxxx
6	103	0	32	5	0	4	3	3	3	1	1	1	1	0x00000074	0x2	0x0000007e	0xxxxxxxxx
6	104	0	32	5	0	4	3	3	3	1	1	1	1	0x00000074	0x2	0x00000080	0xxxxxxxxx
6	105	0	32	5	0	4	3	3	3	1	1	1	1	0x00000074	0x2	0x00000082	0xxxxxxxxx
6	106	0	32	5	0	4	3	3	3	1	1	1	1	0x00000074	0x2	0x00000084	0xxxxxxxxx

6	107	0	32	5	0	4	3	3	3	1	1	1	1	0x00000074	0x2	0x00000086	0xxxxxxxxx
6	108	0	32	5	0	4	3	3	3	1	1	1	1	0x00000074	0x2	0x00000088	0xxxxxxxxx
6	109	0	32	5	0	4	3	3	3	1	1	1	1	0x00000074	0x2	0x0000008a	0xxxxxxxxx
6	110	0	32	5	0	4	3	3	3	1	1	1	1	0x00000074	0x2	0x0000008c	0xxxxxxxxx
6	111	0	32	5	0	4	3	3	3	1	1	1	1	0x00000074	0x2	0x0000008e	0xxxxxxxxx
6	112	0	32	5	0	4	3	3	3	1	1	1	1	0x00000074	0x2	0x00000090	0xxxxxxxxx
6	113	0	32	5	0	4	3	3	3	1	1	1	1	0x00000074	0x2	0x00000092	0xxxxxxxxx
6	114	0	32	5	0	4	3	3	3	1	1	1	1	0x00000074	0x2	0x00000094	0xxxxxxxxx
6	115	0	32	5	0	4	3	3	3	1	1	1	1	0x00000074	0x2	0x00000096	0xxxxxxxxx
6	116	0	32	5	0	4	3	3	3	1	1	1	1	0x00000074	0x2	0x00000098	0xxxxxxxxx
6	117	0	32	5	0	4	3	3	3	1	1	1	1	0x00000074	0x2	0x0000009a	0xxxxxxxxx
6	118	0	32	5	0	4	3	3	3	1	1	1	1	0x00000074	0x2	0x0000009c	0xxxxxxxxx
6	119	0	32	5	0	4	3	3	3	1	1	1	1	0x00000074	0x2	0x0000009e	0xxxxxxxxx
6	120	0	32	5	0	4	3	3	3	1	1	1	1	0x00000074	0x2	0x000000a0	0xxxxxxxxx
6	121	0	32	5	0	4	3	3	3	1	1	1	1	0x00000074	0x2	0x000000a2	0xxxxxxxxx
6	122	0	32	5	0	4	3	3	3	1	1	1	1	0x00000074	0x2	0x000000a4	0xxxxxxxxx
6	123	0	32	5	0	4	3	3	3	1	1	1	1	0x00000074	0x2	0x000000a6	0xxxxxxxxx
6	124	0	32	5	0	4	3	3	3	1	1	1	1	0x00000074	0x2	0x000000a8	0xxxxxxxxx
6	125	0	32	5	0	4	3	3	3	1	1	1	1	0x00000074	0x2	0x000000aa	0xxxxxxxxx
6	126	0	32	5	0	4	3	3	3	1	1	1	1	0x00000074	0x2	0x000000ac	0xxxxxxxxx
6	127	0	32	5	0	4	3	3	3	1	1	1	1	0x00000074	0x2	0x000000ae	0xxxxxxxxx
6	128	0	32	5	0	4	3	3	3	1	1	1	1	0x00000074	0x2	0x000000a0	0xxxxxxxxx
6	129	0	32	5	0	4	3	3	3	1	1	1	1	0x00000074	0x2	0x000000a2	0xxxxxxxxx
6	130	0	32	5	0	4	3	3	3	1	1	1	1	0x00000074	0x2	0x000000a4	0xxxxxxxxx
7	131	0	64	6	0	4	3	3	3	0	0	1	0	0x00000074	0x0	0x000000a6	0xxxxxxxxx
7	132	0	64	6	0	4	3	3	3	1	1	1	1	0x00000074	0x0	0x000000a8	0xxxxxxxxx
7	133	0	64	6	0	4	3	3	3	1	1	1	1	0x00000074	0x0	0x000000aa	0xxxxxxxxx
7	134	0	64	6	0	4	3	3	3	1	1	1	1	0x00000074	0x0	0x000000ab	0xxxxxxxxx
7	135	0	64	6	0	4	3	3	3	0	0	1	1	0x00000000	0x0	0x000000a0	0xxxxxxxxx
7	136	0	64	6	0	4	3	3	3	1	1	1	1	0x00000000	0x0	0x000000a2	0xxxxxxxxx

7	197	0	64	6	0	4	3	3	3	1	1	1	1	0x00000030	0x0	0x00000000	0xxxxxxxx
7	198	0	64	6	0	4	3	3	3	1	1	1	1	0x00000030	0x0	0x00000000	0xxxxxxxx
7	199	0	64	6	0	4	3	3	3	1	1	1	1	0x00000030	0x0	0x00000000	0xxxxxxxx
7	200	0	64	6	0	4	3	3	3	1	1	1	1	0x00000030	0x0	0x00000000	0xxxxxxxx
7	201	0	64	6	0	4	3	3	3	1	1	1	1	0x00000030	0x0	0x00000000	0xxxxxxxx
7	202	0	64	6	0	4	3	3	3	1	1	1	1	0x00000030	0x0	0x00000000	0xxxxxxxx
7	203	0	64	6	0	4	3	3	3	1	1	1	1	0x00000030	0x0	0x00000000	0xxxxxxxx
7	204	0	64	6	0	4	3	3	3	1	1	1	1	0x00000030	0x0	0x00000000	0xxxxxxxx
7	205	0	64	6	0	4	3	3	3	1	1	1	1	0x00000030	0x0	0x00000000	0xxxxxxxx
8	206	0	1	0	1	4	3	3	3	0	0	1	0	0x00000030	0x0	0x00000000	0xxxxxxxx
8	207	0	1	0	1	4	3	3	3	1	1	1	1	0x00000030	0x0	0x00000000	0xxxxxxxx
8	208	0	1	0	1	4	3	3	3	1	1	1	1	0x00000030	0x0	0x00000000	0xxxxxxxx
8	209	0	1	0	1	4	3	3	3	1	1	1	1	0x00000030	0x0	0x00000000	0xxxxxxxx
8	210	0	1	0	1	4	3	3	3	0	0	1	1	0x00000000	0x0	0x00000000	0xxxxxxxx
8	211	0	1	0	1	4	3	3	3	1	1	1	1	0x00000000	0x0	0x00000000	0xxxxxxxx
8	212	0	1	0	1	4	3	3	3	1	1	1	1	0x00000000	0x0	0x00000000	0xxxxxxxx
8	213	0	1	0	1	4	3	3	3	1	1	1	1	0x00000000	0x0	0x00000000	0xxxxxxxx
8	214	0	1	0	1	4	3	3	3	0	1	0	1	0x00000000	0x0	0x00000000	0xxxxxxxx
8	215	0	1	0	1	4	3	3	3	1	1	1	1	0x00000000	0x0	0x00000000	0xxxxxxxx
8	216	0	1	0	1	4	3	3	3	1	1	1	1	0x00000000	0x0	0x00000000	0xxxxxxxx
8	217	0	1	0	1	4	3	3	3	1	1	1	1	0x00000000	0x0	0x00000000	0xxxxxxxx
8	218	0	1	0	1	4	3	3	3	1	1	1	1	0x00000000	0x0	0x00000000	0x00000000
8	219	0	1	0	1	4	3	3	3	1	1	1	1	0x00000000	0x0	0x00000000	0xxxxxxxx
8	220	0	1	0	1	4	3	3	3	1	1	1	1	0x00000000	0x0	0x00000000	0xxxxxxxx
8	221	0	1	0	1	4	3	3	3	1	1	1	1	0x00000000	0x0	0x00000000	0xxxxxxxx
9	222	0	2	1	1	4	3	3	3	0	0	1	0	0x00000000	0x1	0x00000000	0xxxxxxxx
9	223	0	2	1	1	4	3	3	3	1	1	1	1	0x00000000	0x1	0x00000000	0xxxxxxxx
9	224	0	2	1	1	4	3	3	3	1	1	1	1	0x00000000	0x1	0x00000000	0xxxxxxxx
9	225	0	2	1	1	4	3	3	3	1	1	1	1	0x00000000	0x1	0x00000000	0xxxxxxxx
9	226	0	2	1	1	4	3	3	3	0	0	1	1	0x00000002	0x1	0x00000000	0xxxxxxxx

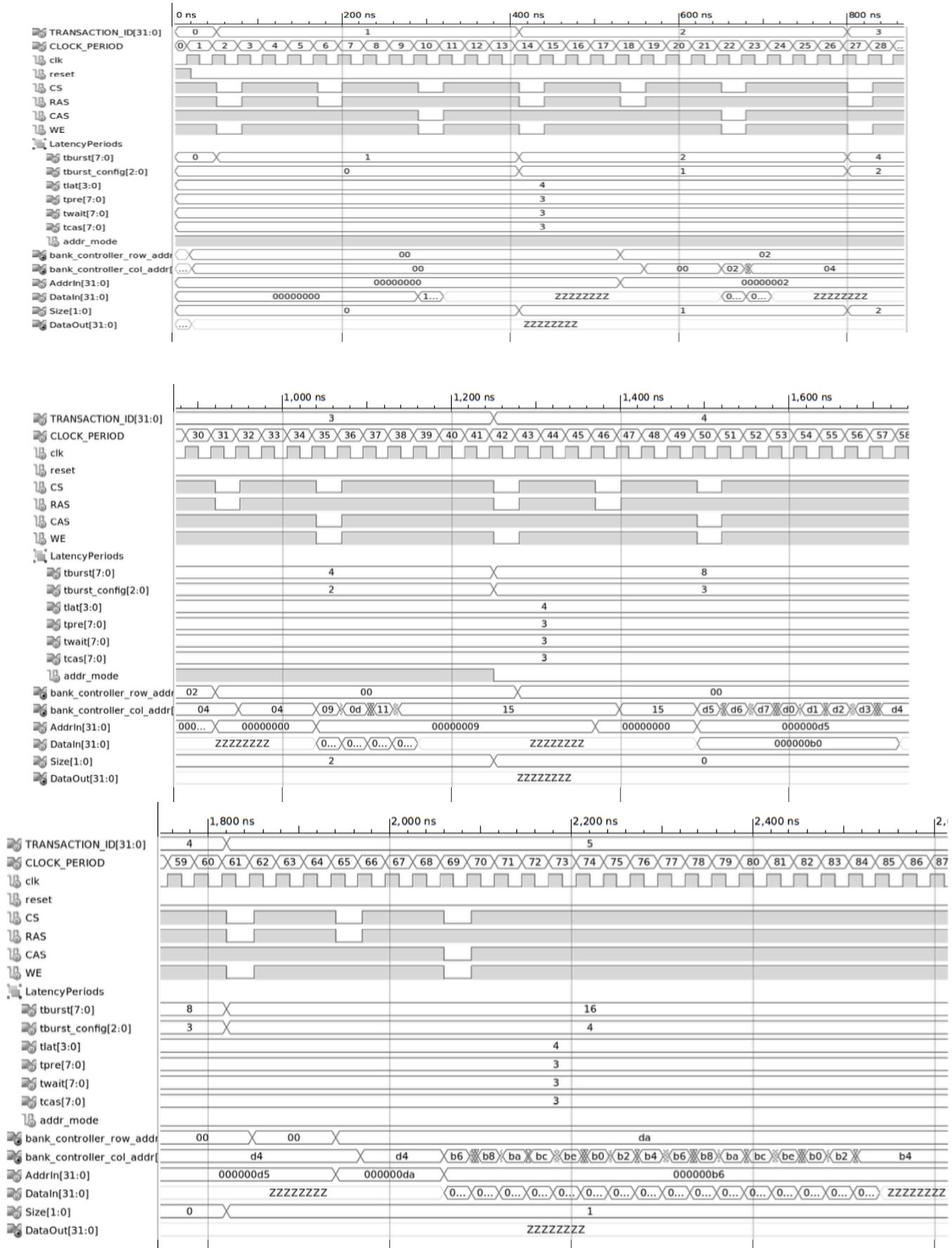
9	227	0	2	1	1	4	3	3	3	1	1	1	1	0x00000002	0x1	0xxxxxxxx	0xxxxxxxx
9	228	0	2	1	1	4	3	3	3	1	1	1	1	0x00000002	0x1	0xxxxxxxx	0xxxxxxxx
9	229	0	2	1	1	4	3	3	3	1	1	1	1	0x00000002	0x1	0xxxxxxxx	0xxxxxxxx
9	230	0	2	1	1	4	3	3	3	0	1	0	1	0x00000002	0x1	0xxxxxxxx	0xxxxxxxx
9	231	0	2	1	1	4	3	3	3	1	1	1	1	0x00000002	0x1	0xxxxxxxx	0xxxxxxxx
9	232	0	2	1	1	4	3	3	3	1	1	1	1	0x00000002	0x1	0xxxxxxxx	0xxxxxxxx
9	233	0	2	1	1	4	3	3	3	1	1	1	1	0x00000002	0x1	0xxxxxxxx	0xxxxxxxx
9	234	0	2	1	1	4	3	3	3	1	1	1	1	0x00000002	0x1	0xxxxxxxx	0x00000001
9	235	0	2	1	1	4	3	3	3	1	1	1	1	0x00000002	0x1	0xxxxxxxx	0x00000002
9	236	0	2	1	1	4	3	3	3	1	1	1	1	0x00000002	0x1	0xxxxxxxx	0xxxxxxxx
9	237	0	2	1	1	4	3	3	3	1	1	1	1	0x00000002	0x1	0xxxxxxxx	0xxxxxxxx
9	238	0	2	1	1	4	3	3	3	1	1	1	1	0x00000002	0x1	0xxxxxxxx	0xxxxxxxx
10	239	0	4	2	1	4	3	3	3	0	0	1	0	0x00000002	0x2	0xxxxxxxx	0xxxxxxxx
10	240	0	4	2	1	4	3	3	3	1	1	1	1	0x00000002	0x2	0xxxxxxxx	0xxxxxxxx
10	241	0	4	2	1	4	3	3	3	1	1	1	1	0x00000002	0x2	0xxxxxxxx	0xxxxxxxx
10	242	0	4	2	1	4	3	3	3	1	1	1	1	0x00000002	0x2	0xxxxxxxx	0xxxxxxxx
10	243	0	4	2	1	4	3	3	3	0	0	1	1	0x00000000	0x2	0xxxxxxxx	0xxxxxxxx
10	244	0	4	2	1	4	3	3	3	1	1	1	1	0x00000000	0x2	0xxxxxxxx	0xxxxxxxx
10	245	0	4	2	1	4	3	3	3	1	1	1	1	0x00000000	0x2	0xxxxxxxx	0xxxxxxxx
10	246	0	4	2	1	4	3	3	3	1	1	1	1	0x00000000	0x2	0xxxxxxxx	0xxxxxxxx
10	247	0	4	2	1	4	3	3	3	0	1	0	1	0x00000009	0x2	0xxxxxxxx	0xxxxxxxx
10	248	0	4	2	1	4	3	3	3	1	1	1	1	0x00000009	0x2	0xxxxxxxx	0xxxxxxxx
10	249	0	4	2	1	4	3	3	3	1	1	1	1	0x00000009	0x2	0xxxxxxxx	0xxxxxxxx
10	250	0	4	2	1	4	3	3	3	1	1	1	1	0x00000009	0x2	0xxxxxxxx	0xxxxxxxx
10	251	0	4	2	1	4	3	3	3	1	1	1	1	0x00000009	0x2	0xxxxxxxx	0x00000000
10	252	0	4	2	1	4	3	3	3	1	1	1	1	0x00000009	0x2	0xxxxxxxx	0x00000000
10	253	0	4	2	1	4	3	3	3	1	1	1	1	0x00000009	0x2	0xxxxxxxx	0x00000000
10	254	0	4	2	1	4	3	3	3	1	1	1	1	0x00000009	0x2	0xxxxxxxx	0x00000000
10	255	0	4	2	1	4	3	3	3	1	1	1	1	0x00000009	0x2	0xxxxxxxx	0xxxxxxxx
10	256	0	4	2	1	4	3	3	3	1	1	1	1	0x00000009	0x2	0xxxxxxxx	0xxxxxxxx

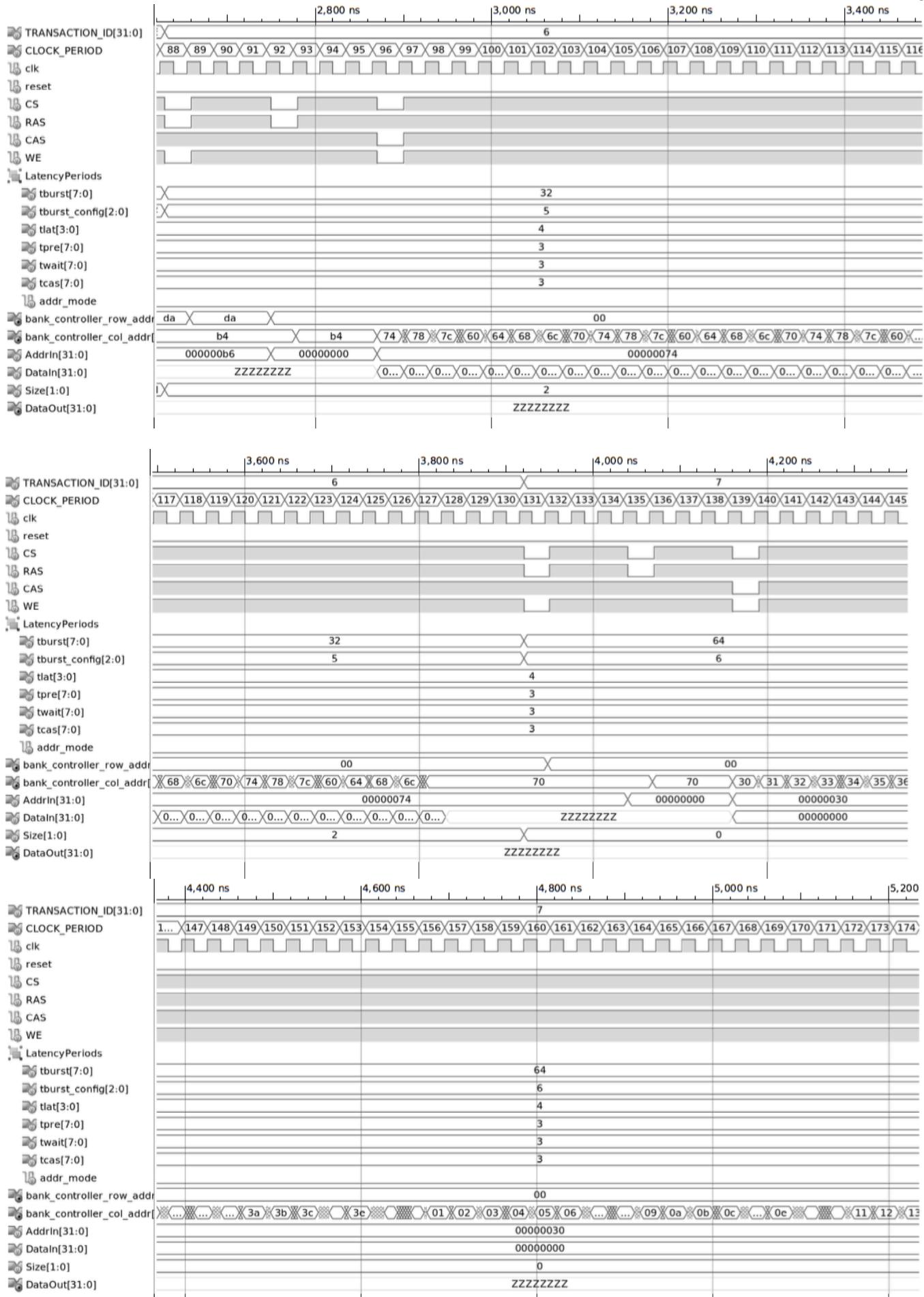
10	257	0	4	2	1	4	3	3	3	1	1	1	1	0x00000009	0x2	0xxxxxxxxx	0xxxxxxxxx
11	258	0	8	3	1	4	3	3	3	0	0	1	0	0x00000009	0x0	0xxxxxxxxx	0xxxxxxxxx
11	259	0	8	3	1	4	3	3	3	1	1	1	1	0x00000009	0x0	0xxxxxxxxx	0xxxxxxxxx
11	260	0	8	3	1	4	3	3	3	1	1	1	1	0x00000009	0x0	0xxxxxxxxx	0xxxxxxxxx
11	261	0	8	3	1	4	3	3	3	1	1	1	1	0x00000009	0x0	0xxxxxxxxx	0xxxxxxxxx
11	262	0	8	3	1	4	3	3	3	0	0	1	1	0x00000000	0x0	0xxxxxxxxx	0xxxxxxxxx
11	263	0	8	3	1	4	3	3	3	1	1	1	1	0x00000000	0x0	0xxxxxxxxx	0xxxxxxxxx
11	264	0	8	3	1	4	3	3	3	1	1	1	1	0x00000000	0x0	0xxxxxxxxx	0xxxxxxxxx
11	265	0	8	3	1	4	3	3	3	1	1	1	1	0x00000000	0x0	0xxxxxxxxx	0xxxxxxxxx
11	266	0	8	3	1	4	3	3	3	0	1	0	1	0x000000d0	0x0	0xxxxxxxxx	0xxxxxxxxx
11	267	0	8	3	1	4	3	3	3	1	1	1	1	0x000000d0	0x0	0xxxxxxxxx	0xxxxxxxxx
11	268	0	8	3	1	4	3	3	3	1	1	1	1	0x000000d0	0x0	0xxxxxxxxx	0xxxxxxxxx
11	269	0	8	3	1	4	3	3	3	1	1	1	1	0x000000d0	0x0	0xxxxxxxxx	0xxxxxxxxx
11	270	0	8	3	1	4	3	3	3	1	1	1	1	0x000000d0	0x0	0xxxxxxxxx	0x000000b0
11	271	0	8	3	1	4	3	3	3	1	1	1	1	0x000000d0	0x0	0xxxxxxxxx	0x000000b0
11	272	0	8	3	1	4	3	3	3	1	1	1	1	0x000000d0	0x0	0xxxxxxxxx	0x000000b0
11	273	0	8	3	1	4	3	3	3	1	1	1	1	0x000000d0	0x0	0xxxxxxxxx	0x000000b0
11	274	0	8	3	1	4	3	3	3	1	1	1	1	0x000000d0	0x0	0xxxxxxxxx	0x000000b0
11	275	0	8	3	1	4	3	3	3	1	1	1	1	0x000000d0	0x0	0xxxxxxxxx	0x000000b0
11	276	0	8	3	1	4	3	3	3	1	1	1	1	0x000000d0	0x0	0xxxxxxxxx	0x000000b0
11	277	0	8	3	1	4	3	3	3	1	1	1	1	0x000000d0	0x0	0xxxxxxxxx	0x000000b0
11	278	0	8	3	1	4	3	3	3	1	1	1	1	0x000000d0	0x0	0xxxxxxxxx	0xxxxxxxxx
11	279	0	8	3	1	4	3	3	3	1	1	1	1	0x000000d0	0x0	0xxxxxxxxx	0xxxxxxxxx
11	280	0	8	3	1	4	3	3	3	1	1	1	1	0x000000d0	0x0	0xxxxxxxxx	0xxxxxxxxx
12	281	0	16	4	0	4	3	3	3	0	0	1	0	0x000000d0	0x1	0xxxxxxxxx	0xxxxxxxxx
12	282	0	16	4	0	4	3	3	3	1	1	1	1	0x000000d0	0x1	0xxxxxxxxx	0xxxxxxxxx
12	283	0	16	4	0	4	3	3	3	1	1	1	1	0x000000d0	0x1	0xxxxxxxxx	0xxxxxxxxx
12	284	0	16	4	0	4	3	3	3	1	1	1	1	0x000000d0	0x1	0xxxxxxxxx	0xxxxxxxxx
12	285	0	16	4	0	4	3	3	3	0	0	1	1	0x000000da	0x1	0xxxxxxxxx	0xxxxxxxxx
12	286	0	16	4	0	4	3	3	3	1	1	1	1	0x000000da	0x1	0xxxxxxxxx	0xxxxxxxxx

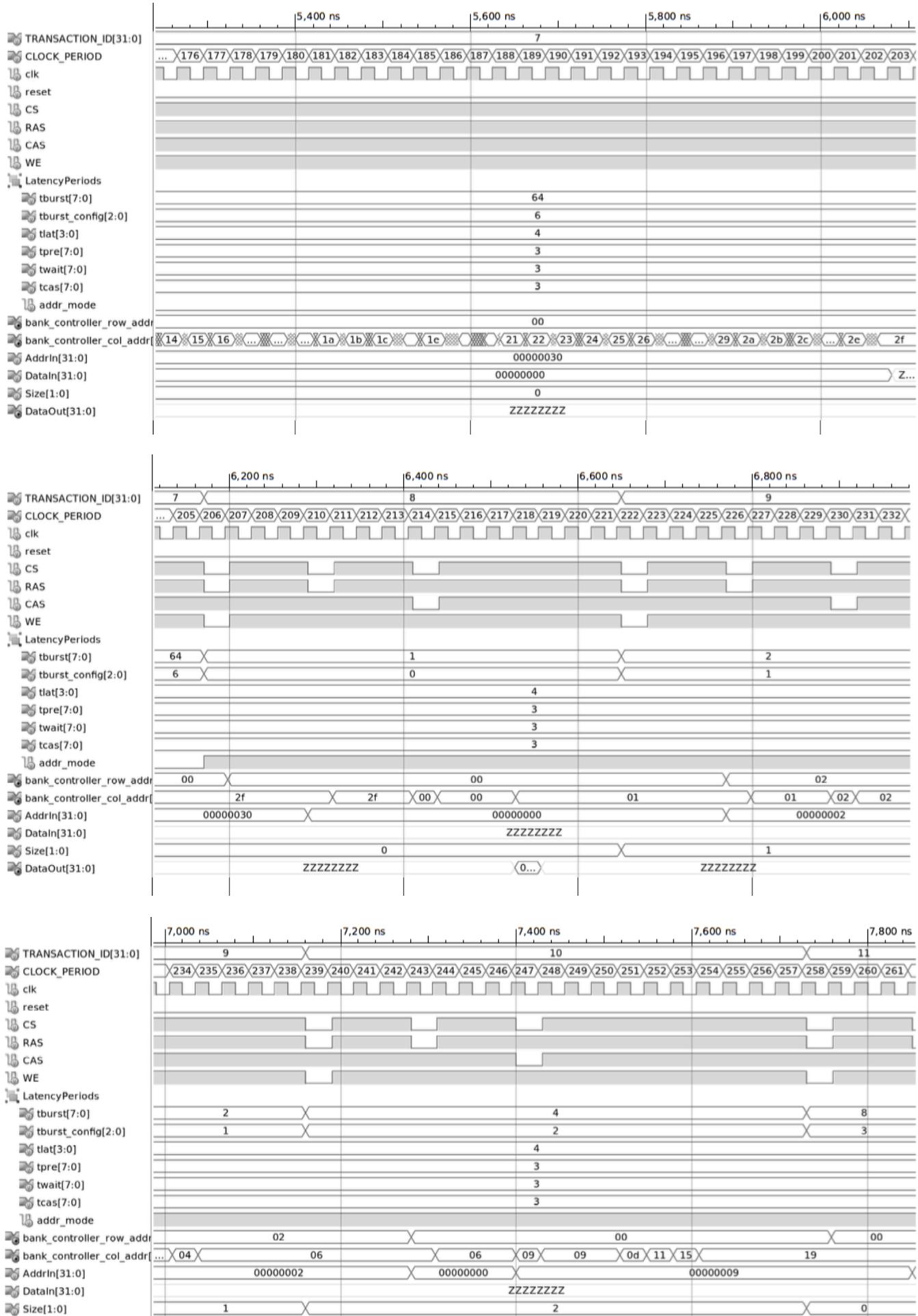
12	287	0	16	4	0	4	3	3	3	1	1	1	1	0x000000da	0x1	0xxxxxxxx	0xxxxxxxx
12	288	0	16	4	0	4	3	3	3	1	1	1	1	0x000000da	0x1	0xxxxxxxx	0xxxxxxxx
12	289	0	16	4	0	4	3	3	3	0	1	0	1	0x000000b6	0x1	0xxxxxxxx	0xxxxxxxx
12	290	0	16	4	0	4	3	3	3	1	1	1	1	0x000000b6	0x1	0xxxxxxxx	0xxxxxxxx
12	291	0	16	4	0	4	3	3	3	1	1	1	1	0x000000b6	0x1	0xxxxxxxx	0xxxxxxxx
12	292	0	16	4	0	4	3	3	3	1	1	1	1	0x000000b6	0x1	0xxxxxxxx	0xxxxxxxx
12	293	0	16	4	0	4	3	3	3	1	1	1	1	0x000000b6	0x1	0xxxxxxxx	0x000000b8
12	294	0	16	4	0	4	3	3	3	1	1	1	1	0x000000b6	0x1	0xxxxxxxx	0x000000b9
12	295	0	16	4	0	4	3	3	3	1	1	1	1	0x000000b6	0x1	0xxxxxxxx	0x000000ba
12	296	0	16	4	0	4	3	3	3	1	1	1	1	0x000000b6	0x1	0xxxxxxxx	0x000000bb
12	297	0	16	4	0	4	3	3	3	1	1	1	1	0x000000b6	0x1	0xxxxxxxx	0x000000bc
12	298	0	16	4	0	4	3	3	3	1	1	1	1	0x000000b6	0x1	0xxxxxxxx	0x000000bd
12	299	0	16	4	0	4	3	3	3	1	1	1	1	0x000000b6	0x1	0xxxxxxxx	0x000000be
12	300	0	16	4	0	4	3	3	3	1	1	1	1	0x000000b6	0x1	0xxxxxxxx	0x000000bf
12	301	0	16	4	0	4	3	3	3	1	1	1	1	0x000000b6	0x1	0xxxxxxxx	0x000000b8
12	302	0	16	4	0	4	3	3	3	1	1	1	1	0x000000b6	0x1	0xxxxxxxx	0x000000b9
12	303	0	16	4	0	4	3	3	3	1	1	1	1	0x000000b6	0x1	0xxxxxxxx	0x000000ba
12	304	0	16	4	0	4	3	3	3	1	1	1	1	0x000000b6	0x1	0xxxxxxxx	0x000000bb
12	305	0	16	4	0	4	3	3	3	1	1	1	1	0x000000b6	0x1	0xxxxxxxx	0x000000bc
12	306	0	16	4	0	4	3	3	3	1	1	1	1	0x000000b6	0x1	0xxxxxxxx	0x000000bd
12	307	0	16	4	0	4	3	3	3	1	1	1	1	0x000000b6	0x1	0xxxxxxxx	0x000000be
12	308	0	16	4	0	4	3	3	3	1	1	1	1	0x000000b6	0x1	0xxxxxxxx	0x000000bf
12	309	0	16	4	0	4	3	3	3	1	1	1	1	0x000000b6	0x1	0xxxxxxxx	0xxxxxxxx
12	310	0	16	4	0	4	3	3	3	1	1	1	1	0x000000b6	0x1	0xxxxxxxx	0xxxxxxxx
12	311	0	16	4	0	4	3	3	3	1	1	1	1	0x000000b6	0x1	0xxxxxxxx	0xxxxxxxx
13	312	0	32	5	0	4	3	3	3	0	0	1	0	0x000000b6	0x2	0xxxxxxxx	0xxxxxxxx
13	313	0	32	5	0	4	3	3	3	1	1	1	1	0x000000b6	0x2	0xxxxxxxx	0xxxxxxxx
13	314	0	32	5	0	4	3	3	3	1	1	1	1	0x000000b6	0x2	0xxxxxxxx	0xxxxxxxx
13	315	0	32	5	0	4	3	3	3	1	1	1	1	0x000000b6	0x2	0xxxxxxxx	0xxxxxxxx
13	316	0	32	5	0	4	3	3	3	0	0	1	1	0x00000000	0x2	0xxxxxxxx	0xxxxxxxx

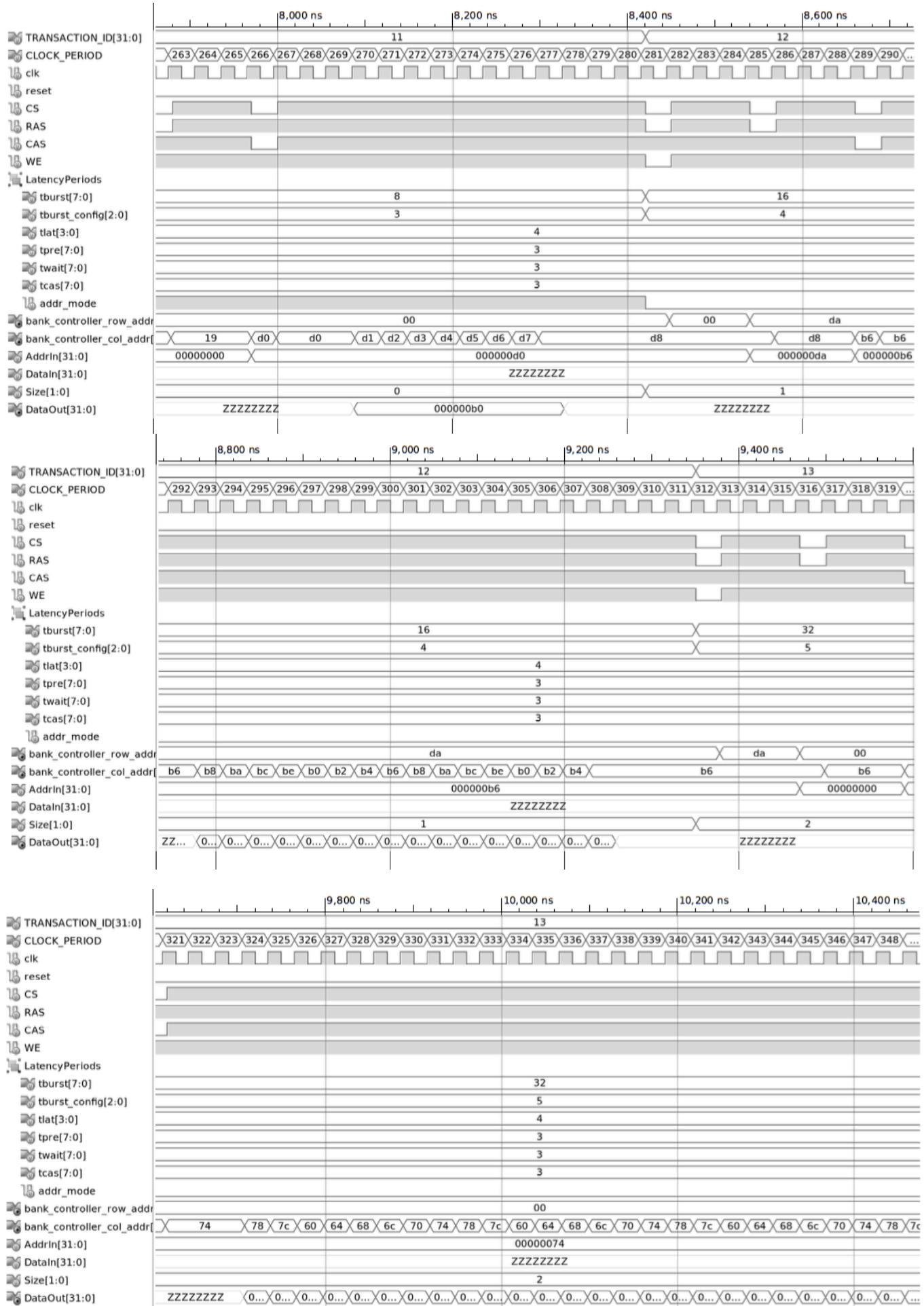
13	317	0	32	5	0	4	3	3	3	1	1	1	1	0x00000000	0x2	0xxxxxxxx	0xxxxxxxx
13	318	0	32	5	0	4	3	3	3	1	1	1	1	0x00000000	0x2	0xxxxxxxx	0xxxxxxxx
13	319	0	32	5	0	4	3	3	3	1	1	1	1	0x00000000	0x2	0xxxxxxxx	0xxxxxxxx
13	320	0	32	5	0	4	3	3	3	0	1	0	1	0x00000074	0x2	0xxxxxxxx	0xxxxxxxx
13	321	0	32	5	0	4	3	3	3	1	1	1	1	0x00000074	0x2	0xxxxxxxx	0xxxxxxxx
13	322	0	32	5	0	4	3	3	3	1	1	1	1	0x00000074	0x2	0xxxxxxxx	0xxxxxxxx
13	323	0	32	5	0	4	3	3	3	1	1	1	1	0x00000074	0x2	0xxxxxxxx	0xxxxxxxx
13	324	0	32	5	0	4	3	3	3	1	1	1	1	0x00000074	0x2	0xxxxxxxx	0x000000a0
13	325	0	32	5	0	4	3	3	3	1	1	1	1	0x00000074	0x2	0xxxxxxxx	0x000000a2
13	326	0	32	5	0	4	3	3	3	1	1	1	1	0x00000074	0x2	0xxxxxxxx	0x000000a4
13	327	0	32	5	0	4	3	3	3	1	1	1	1	0x00000074	0x2	0xxxxxxxx	0x000000a6
13	328	0	32	5	0	4	3	3	3	1	1	1	1	0x00000074	0x2	0xxxxxxxx	0x000000a8
13	329	0	32	5	0	4	3	3	3	1	1	1	1	0x00000074	0x2	0xxxxxxxx	0x000000aa
13	330	0	32	5	0	4	3	3	3	1	1	1	1	0x00000074	0x2	0xxxxxxxx	0x000000ac
13	331	0	32	5	0	4	3	3	3	1	1	1	1	0x00000074	0x2	0xxxxxxxx	0x000000ae
13	332	0	32	5	0	4	3	3	3	1	1	1	1	0x00000074	0x2	0xxxxxxxx	0x000000a0
13	333	0	32	5	0	4	3	3	3	1	1	1	1	0x00000074	0x2	0xxxxxxxx	0x000000a2
13	334	0	32	5	0	4	3	3	3	1	1	1	1	0x00000074	0x2	0xxxxxxxx	0x000000a4
13	335	0	32	5	0	4	3	3	3	1	1	1	1	0x00000074	0x2	0xxxxxxxx	0x000000a6
13	336	0	32	5	0	4	3	3	3	1	1	1	1	0x00000074	0x2	0xxxxxxxx	0x000000a8
13	337	0	32	5	0	4	3	3	3	1	1	1	1	0x00000074	0x2	0xxxxxxxx	0x000000aa
13	338	0	32	5	0	4	3	3	3	1	1	1	1	0x00000074	0x2	0xxxxxxxx	0x000000ac
13	339	0	32	5	0	4	3	3	3	1	1	1	1	0x00000074	0x2	0xxxxxxxx	0x000000ae
13	340	0	32	5	0	4	3	3	3	1	1	1	1	0x00000074	0x2	0xxxxxxxx	0x000000a0
13	341	0	32	5	0	4	3	3	3	1	1	1	1	0x00000074	0x2	0xxxxxxxx	0x000000a2
13	342	0	32	5	0	4	3	3	3	1	1	1	1	0x00000074	0x2	0xxxxxxxx	0x000000a4
13	343	0	32	5	0	4	3	3	3	1	1	1	1	0x00000074	0x2	0xxxxxxxx	0x000000a6
13	344	0	32	5	0	4	3	3	3	1	1	1	1	0x00000074	0x2	0xxxxxxxx	0x000000a8
13	345	0	32	5	0	4	3	3	3	1	1	1	1	0x00000074	0x2	0xxxxxxxx	0x000000aa
13	346	0	32	5	0	4	3	3	3	1	1	1	1	0x00000074	0x2	0xxxxxxxx	0x000000ac

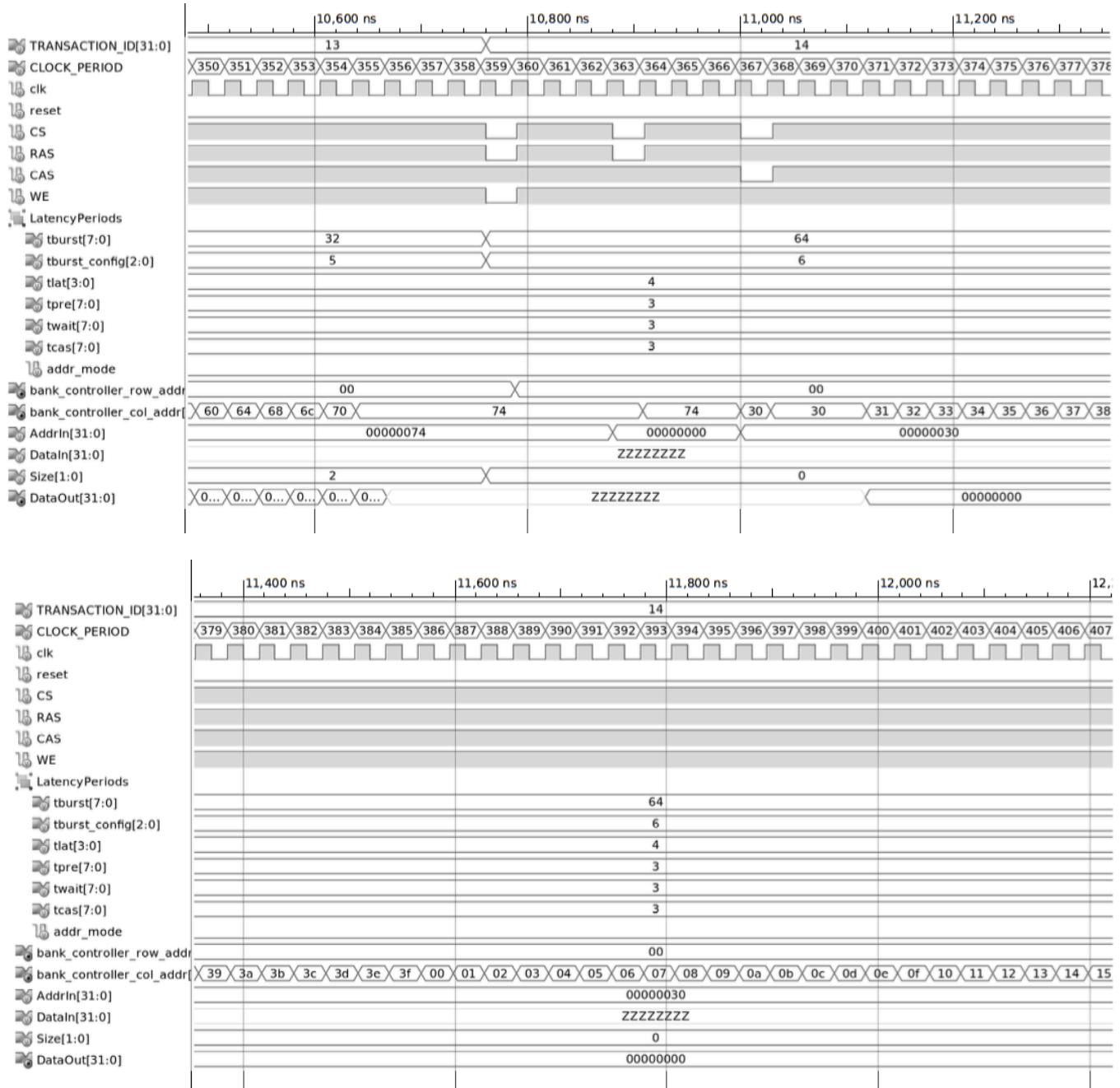
13	347	0	32	5	0	4	3	3	3	1	1	1	1	0x00000074	0x2	0xxxxxxxx	0x000000ae
13	348	0	32	5	0	4	3	3	3	1	1	1	1	0x00000074	0x2	0xxxxxxxx	0x000000a0
13	349	0	32	5	0	4	3	3	3	1	1	1	1	0x00000074	0x2	0xxxxxxxx	0x000000a2
13	350	0	32	5	0	4	3	3	3	1	1	1	1	0x00000074	0x2	0xxxxxxxx	0x000000a4
13	351	0	32	5	0	4	3	3	3	1	1	1	1	0x00000074	0x2	0xxxxxxxx	0x000000a6
13	352	0	32	5	0	4	3	3	3	1	1	1	1	0x00000074	0x2	0xxxxxxxx	0x000000a8
13	353	0	32	5	0	4	3	3	3	1	1	1	1	0x00000074	0x2	0xxxxxxxx	0x000000aa
13	354	0	32	5	0	4	3	3	3	1	1	1	1	0x00000074	0x2	0xxxxxxxx	0x000000ac
13	355	0	32	5	0	4	3	3	3	1	1	1	1	0x00000074	0x2	0xxxxxxxx	0x000000ae
13	356	0	32	5	0	4	3	3	3	1	1	1	1	0x00000074	0x2	0xxxxxxxx	0xxxxxxxx
13	357	0	32	5	0	4	3	3	3	1	1	1	1	0x00000074	0x2	0xxxxxxxx	0xxxxxxxx
13	358	0	32	5	0	4	3	3	3	1	1	1	1	0x00000074	0x2	0xxxxxxxx	0xxxxxxxx
14	359	0	64	6	0	4	3	3	3	0	0	1	0	0x00000074	0x0	0xxxxxxxx	0xxxxxxxx
14	360	0	64	6	0	4	3	3	3	1	1	1	1	0x00000074	0x0	0xxxxxxxx	0xxxxxxxx
14	361	0	64	6	0	4	3	3	3	1	1	1	1	0x00000074	0x0	0xxxxxxxx	0xxxxxxxx
14	362	0	64	6	0	4	3	3	3	1	1	1	1	0x00000074	0x0	0xxxxxxxx	0xxxxxxxx
14	363	0	64	6	0	4	3	3	3	0	0	1	1	0x00000000	0x0	0xxxxxxxx	0xxxxxxxx
14	364	0	64	6	0	4	3	3	3	1	1	1	1	0x00000000	0x0	0xxxxxxxx	0xxxxxxxx
14	365	0	64	6	0	4	3	3	3	1	1	1	1	0x00000000	0x0	0xxxxxxxx	0xxxxxxxx
14	366	0	64	6	0	4	3	3	3	1	1	1	1	0x00000000	0x0	0xxxxxxxx	0xxxxxxxx
14	367	0	64	6	0	4	3	3	3	0	1	0	1	0x00000030	0x0	0xxxxxxxx	0xxxxxxxx
14	368	0	64	6	0	4	3	3	3	1	1	1	1	0x00000030	0x0	0xxxxxxxx	0xxxxxxxx
14	369	0	64	6	0	4	3	3	3	1	1	1	1	0x00000030	0x0	0xxxxxxxx	0xxxxxxxx
14	370	0	64	6	0	4	3	3	3	1	1	1	1	0x00000030	0x0	0xxxxxxxx	0xxxxxxxx
14	371	0	64	6	0	4	3	3	3	1	1	1	1	0x00000030	0x0	0xxxxxxxx	0x00000000
14	372	0	64	6	0	4	3	3	3	1	1	1	1	0x00000030	0x0	0xxxxxxxx	0x00000000
14	373	0	64	6	0	4	3	3	3	1	1	1	1	0x00000030	0x0	0xxxxxxxx	0x00000000
14	374	0	64	6	0	4	3	3	3	1	1	1	1	0x00000030	0x0	0xxxxxxxx	0x00000000
14	375	0	64	6	0	4	3	3	3	1	1	1	1	0x00000030	0x0	0xxxxxxxx	0x00000000
14	376	0	64	6	0	4	3	3	3	1	1	1	1	0x00000030	0x0	0xxxxxxxx	0x00000000

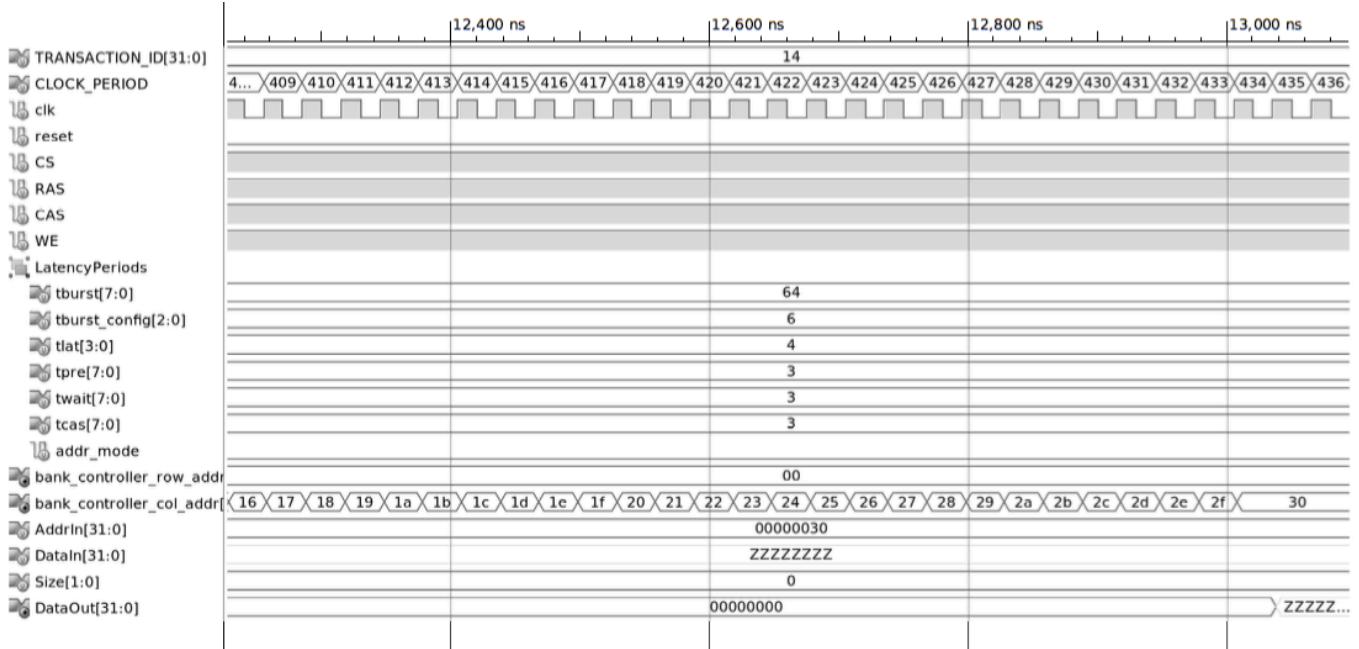












8. SDRAM (4 Banks)

The verification table, test framework and waveforms for the SDRAM in its entirety is show below. 4 SDRAM Bank cores have been used in this design and the same are tested. A set of 12 transactions have been performed as follows:

Transaction 1 Programs the register file and Address Mode Registers with tCAS = 3, tWait = 3, tPre = 3, TLat = 4, Burst 4 and Sequential Addressing Mode.

Transaction 2 is a write to Bank 0 with Burst 4, packet size in Bytes and a ‘Master Busy’ simulation turned off.

Transaction 3 is a read from Bank 0 with Burst 4, packet size in Bytes and ‘Master Busy’ simulation turned off.

Transaction 4 reprograms the register file and address mode register with Burst Configuration of 8 and linear addressing mode.

Transaction 5 is a write to Bank 1 with Burst 8, packet size in Half Words and a ‘Master Busy’ simulation turned on.

Transaction 6 is a read from Bank 1 with Burst 8, packet size in Half Words and a ‘Master Busy’ simulation turned on.

Transaction 7 reprograms the register file and address mode register with Burst Configuration of 16 and linear addressing mode.

Transaction 8 is a write to Bank 2 with Burst 16, packet size in Words and a ‘Master Busy’ simulation turned off.

Transaction 9 is a read from Bank 2 with Burst 16, packet size in Words and a ‘Master Busy’ simulation turned off.

Transaction 10 reprograms the register file and address mode register with Burst Configuration of 32 and linear addressing mode.

Transaction 11 is a write to Bank 3 with Burst 32, packet size in Words and a ‘Master Busy’ simulation turned off.

Transaction 12 is a read from Bank 3 with Burst 32, packet size in Words and a ‘Master Busy’ simulation turned off.

The testing of this module was done form the point of view of the BIU. Each read and write command carried a particular bank through all stages from Precharge, to activate and write/read. Transactions to all banks were tested. When the ‘Master Busy’ simulation is on, the ‘Busy’ signal is issued from the BIU for 2 cycles. When this happens, address generations should halt. The behavior of the module on testing was as expected.

TID	Clk Period	Reset	CS	BS	RAS	CAS	WE	Size	AddrIn	Data
0	0	1	1	3'b00	1	1	1	0	0xxxxxxxxx	0xxxxxxxxx
0	1	0	1	3'b00	1	1	1	0	0xxxxxxxxx	0xxxxxxxxx
1	2	0	0	3'b00	0	0	0	0	0x03030342	0xxxxxxxxx
1	3	0	1	3'b00	1	1	1	0	0x03030342	0xxxxxxxxx
2	4	0	0	3'b00	0	1	0	0	0x03030342	0xxxxxxxxx
2	5	0	1	3'b00	1	1	1	0	0x03030342	0xxxxxxxxx
2	6	0	1	3'b00	1	1	1	0	0x03030342	0xxxxxxxxx
2	7	0	1	3'b00	1	1	1	0	0x03030342	0xxxxxxxxx
2	8	0	0	3'b00	0	1	1	0	0x000000aa	0xxxxxxxxx
2	9	0	1	3'b00	1	1	1	0	0x000000aa	0xxxxxxxxx
2	10	0	1	3'b00	1	1	1	0	0x000000aa	0xxxxxxxxx
2	11	0	1	3'b00	1	1	1	0	0x000000aa	0xxxxxxxxx
2	12	0	0	3'b00	1	0	0	0	0x00000005	0xaabbccdd
2	13	0	1	3'b00	1	1	1	0	0x00000005	0xaabbccde
2	14	0	1	3'b00	1	1	1	0	0x00000005	0xaabbccdf
2	15	0	1	3'b00	1	1	1	0	0x00000005	0xaabbccce0
2	16	0	1	3'b00	1	1	1	0	0x00000005	0xxxxxxxxx

2	17	0	1	3'b00	1	1	1	0	0x00000005	0xxxxxxxx
2	18	0	1	3'b00	1	1	1	0	0x00000005	0xxxxxxxx
3	19	0	0	3'b00	0	1	0	0	0x00000005	0xxxxxxxx
3	20	0	1	3'b00	1	1	1	0	0x00000005	0xxxxxxxx
3	21	0	1	3'b00	1	1	1	0	0x00000005	0xxxxxxxx
3	22	0	1	3'b00	1	1	1	0	0x00000005	0xxxxxxxx
3	23	0	0	3'b00	0	1	1	0	0x000000aa	0xxxxxxxx
3	24	0	1	3'b00	1	1	1	0	0x000000aa	0xxxxxxxx
3	25	0	1	3'b00	1	1	1	0	0x000000aa	0xxxxxxxx
3	26	0	1	3'b00	1	1	1	0	0x000000aa	0xxxxxxxx
3	27	0	0	3'b00	1	0	1	0	0x00000005	0xxxxxxxx
3	28	0	1	3'b00	1	1	1	0	0x00000005	0xxxxxxxx
3	29	0	1	3'b00	1	1	1	0	0x00000005	0xxxxxxxx
3	30	0	1	3'b00	1	1	1	0	0x00000005	0xxxxxxxx
3	31	0	1	3'b00	1	1	1	0	0x00000005	0x000000dd
3	32	0	1	3'b00	1	1	1	0	0x00000005	0x000000de
3	33	0	1	3'b00	1	1	1	0	0x00000005	0x000000df
3	34	0	1	3'b00	1	1	1	0	0x00000005	0x000000e0
3	35	0	1	3'b00	1	1	1	0	0x00000005	0xxxxxxxx
3	36	0	1	3'b00	1	1	1	0	0x00000005	0xxxxxxxx
3	37	0	1	3'b00	1	1	1	0	0x00000005	0xxxxxxxx
4	38	0	0	3'b00	0	0	0	0	0x0303034b	0xxxxxxxx
4	39	0	1	3'b00	1	1	1	0	0x0303034b	0xxxxxxxx
5	40	0	0	3'b01	0	1	0	1	0x0303034b	0xxxxxxxx
5	41	0	1	3'b01	1	1	1	1	0x0303034b	0xxxxxxxx
5	42	0	1	3'b01	1	1	1	1	0x0303034b	0xxxxxxxx
5	43	0	1	3'b01	1	1	1	1	0x0303034b	0xxxxxxxx
5	44	0	0	3'b01	0	1	1	1	0x0000002d	0xxxxxxxx
5	45	0	1	3'b01	1	1	1	1	0x0000002d	0xxxxxxxx
5	46	0	1	3'b01	1	1	1	1	0x0000002d	0xxxxxxxx

5	47	0	1	3'b01	1	1	1	1	0x0000002d	0xxxxxxxx
5	48	0	0	3'b01	1	0	0	1	0x00000006	0xaabbccdd
5	49	0	1	3'b01	1	1	1	1	0x00000006	0xaabbccdf
5	50	0	1	3'b01	1	1	1	1	0x00000006	0xaabbcce1
5	51	0	1	3'b01	1	0	0	1	0x00000006	0xaabbcce1
5	52	0	1	3'b01	1	0	0	1	0x00000006	0xaabbcce1
5	53	0	1	3'b01	1	1	1	1	0x00000006	0xaabbcce3
5	54	0	1	3'b01	1	1	1	1	0x00000006	0xaabbcce5
5	55	0	1	3'b01	1	1	1	1	0x00000006	0xaabbcce7
5	56	0	1	3'b01	1	1	1	1	0x00000006	0xaabbcce9
5	57	0	1	3'b01	1	1	1	1	0x00000006	0xaabbcccb
5	58	0	1	3'b01	1	1	1	1	0x00000006	0xxxxxxxx
5	59	0	1	3'b01	1	1	1	1	0x00000006	0xxxxxxxx
5	60	0	1	3'b01	1	1	1	1	0x00000006	0xxxxxxxx
6	61	0	0	3'b01	0	1	0	1	0x00000006	0xxxxxxxx
6	62	0	1	3'b01	1	1	1	1	0x00000006	0xxxxxxxx
6	63	0	1	3'b01	1	1	1	1	0x00000006	0xxxxxxxx
6	64	0	1	3'b01	1	1	1	1	0x00000006	0xxxxxxxx
6	65	0	0	3'b01	0	1	1	1	0x0000002d	0xxxxxxxx
6	66	0	1	3'b01	1	1	1	1	0x0000002d	0xxxxxxxx
6	67	0	1	3'b01	1	1	1	1	0x0000002d	0xxxxxxxx
6	68	0	1	3'b01	1	1	1	1	0x0000002d	0xxxxxxxx
6	69	0	0	3'b01	1	0	1	1	0x00000006	0xxxxxxxx
6	70	0	1	3'b01	1	1	1	1	0x00000006	0xxxxxxxx
6	71	0	1	3'b01	1	1	1	1	0x00000006	0xxxxxxxx
6	72	0	1	3'b01	1	1	1	1	0x00000006	0xxxxxxxx
6	73	0	1	3'b01	1	1	1	1	0x00000006	0x0000ccdd
6	74	0	1	3'b01	1	1	1	1	0x00000006	0x0000ccdf
6	75	0	1	3'b01	1	1	1	1	0x00000006	0x0000cce1
6	76	0	1	3'b01	1	0	0	1	0x00000006	0x0000cce3

6	77	0	1	3'b01	1	0	0	1	0x00000006	0x0000cce5
6	78	0	1	3'b01	1	1	1	1	0x00000006	0x0000cce5
6	79	0	1	3'b01	1	1	1	1	0x00000006	0x0000cce5
6	80	0	1	3'b01	1	1	1	1	0x00000006	0x0000cce7
6	81	0	1	3'b01	1	1	1	1	0x00000006	0x0000cce9
6	82	0	1	3'b01	1	1	1	1	0x00000006	0x0000cccb
6	83	0	1	3'b01	1	1	1	1	0x00000006	0xzzzzzzz
6	84	0	1	3'b01	1	1	1	1	0x00000006	0xzzzzzzz
6	85	0	1	3'b01	1	1	1	1	0x00000006	0xzzzzzzz
7	86	0	0	3'b01	0	0	0	1	0x0303034c	0xzzzzzzz
7	87	0	1	3'b01	1	1	1	1	0x0303034c	0xzzzzzzz
8	88	0	0	3'b10	0	1	0	2	0x0303034c	0xzzzzzzz
8	89	0	1	3'b10	1	1	1	2	0x0303034c	0xzzzzzzz
8	90	0	1	3'b10	1	1	1	2	0x0303034c	0xzzzzzzz
8	91	0	1	3'b10	1	1	1	2	0x0303034c	0xzzzzzzz
8	92	0	0	3'b10	0	1	1	2	0x0000002d	0xzzzzzzz
8	93	0	1	3'b10	1	1	1	2	0x0000002d	0xzzzzzzz
8	94	0	1	3'b10	1	1	1	2	0x0000002d	0xzzzzzzz
8	95	0	1	3'b10	1	1	1	2	0x0000002d	0xzzzzzzz
8	96	0	0	3'b10	1	0	0	2	0x00000006	0xaabbccdd
8	97	0	1	3'b10	1	1	1	2	0x00000006	0xaabbccel
8	98	0	1	3'b10	1	1	1	2	0x00000006	0xaabbcce5
8	99	0	1	3'b10	1	1	1	2	0x00000006	0xaabbcce9
8	100	0	1	3'b10	1	1	1	2	0x00000006	0xaabbcced
8	101	0	1	3'b10	1	1	1	2	0x00000006	0xaabbccf1
8	102	0	1	3'b10	1	1	1	2	0x00000006	0xaabbccf5
8	103	0	1	3'b10	1	1	1	2	0x00000006	0xaabbccf9
8	104	0	1	3'b10	1	1	1	2	0x00000006	0xaabbccfd
8	105	0	1	3'b10	1	1	1	2	0x00000006	0xaabbcd01
8	106	0	1	3'b10	1	1	1	2	0x00000006	0xaabbcd05

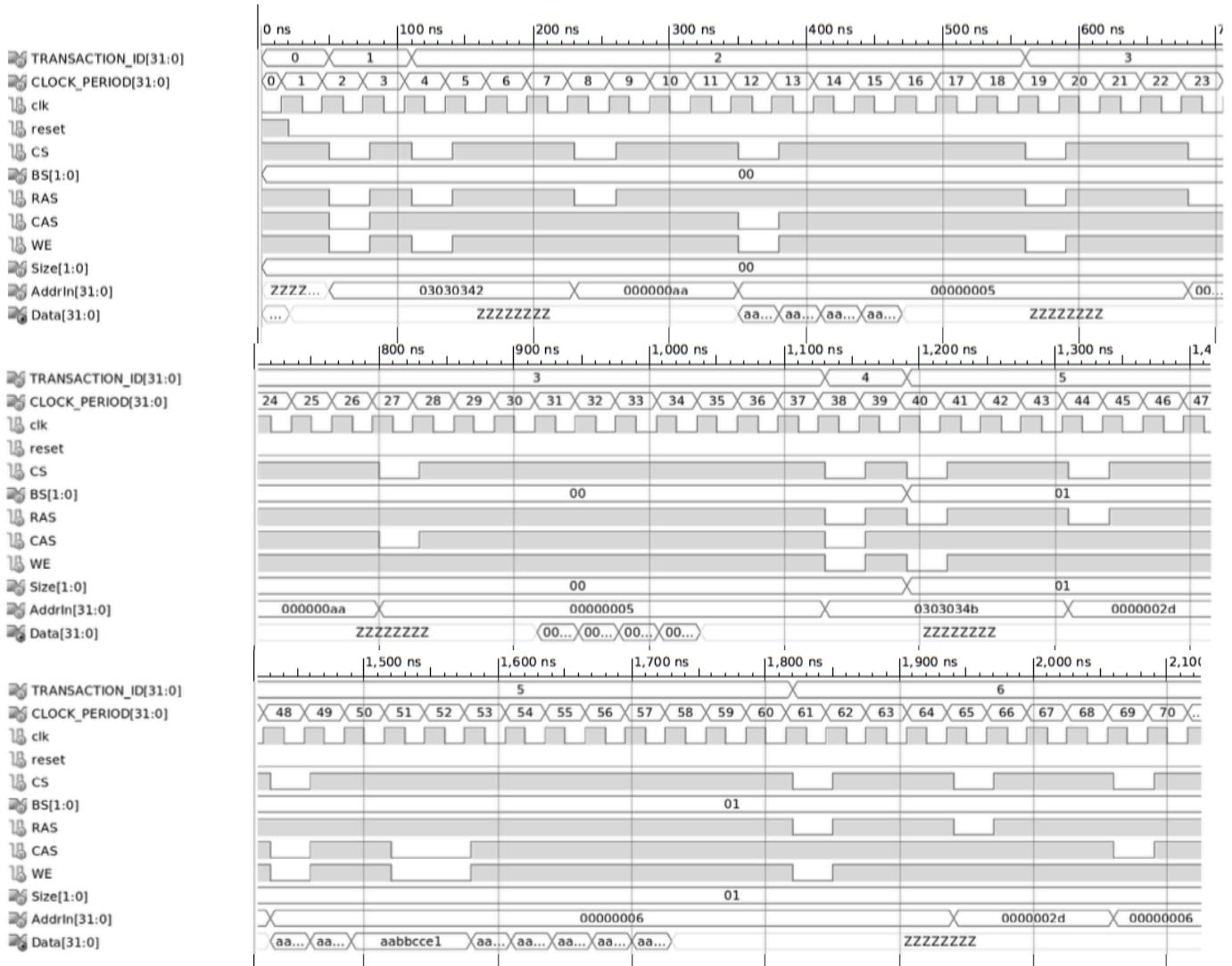
8	107	0	1	3'b10	1	1	1	2	0x00000006	0xaabbcd09
8	108	0	1	3'b10	1	1	1	2	0x00000006	0xaabbcd0d
8	109	0	1	3'b10	1	1	1	2	0x00000006	0xaabbcd11
8	110	0	1	3'b10	1	1	1	2	0x00000006	0xaabbcd15
8	111	0	1	3'b10	1	1	1	2	0x00000006	0xaabbcd19
8	112	0	1	3'b10	1	1	1	2	0x00000006	0xzzzzzzz
8	113	0	1	3'b10	1	1	1	2	0x00000006	0xzzzzzzz
8	114	0	1	3'b10	1	1	1	2	0x00000006	0xzzzzzzz
9	115	0	0	3'b10	0	1	0	2	0x00000006	0xzzzzzzz
9	116	0	1	3'b10	1	1	1	2	0x00000006	0xzzzzzzz
9	117	0	1	3'b10	1	1	1	2	0x00000006	0xzzzzzzz
9	118	0	1	3'b10	1	1	1	2	0x00000006	0xzzzzzzz
9	119	0	0	3'b10	0	1	1	2	0x0000002d	0xzzzzzzz
9	120	0	1	3'b10	1	1	1	2	0x0000002d	0xzzzzzzz
9	121	0	1	3'b10	1	1	1	2	0x0000002d	0xzzzzzzz
9	122	0	1	3'b10	1	1	1	2	0x0000002d	0xzzzzzzz
9	123	0	0	3'b10	1	0	1	2	0x00000006	0xzzzzzzz
9	124	0	1	3'b10	1	1	1	2	0x00000006	0xzzzzzzz
9	125	0	1	3'b10	1	1	1	2	0x00000006	0xzzzzzzz
9	126	0	1	3'b10	1	1	1	2	0x00000006	0xzzzzzzz
9	127	0	1	3'b10	1	1	1	2	0x00000006	0xaabbccdd
9	128	0	1	3'b10	1	1	1	2	0x00000006	0xaabbcce1
9	129	0	1	3'b10	1	1	1	2	0x00000006	0xaabbcce5
9	130	0	1	3'b10	1	1	1	2	0x00000006	0xaabbcce9
9	131	0	1	3'b10	1	1	1	2	0x00000006	0xaabbcced
9	132	0	1	3'b10	1	1	1	2	0x00000006	0xaabbccfl
9	133	0	1	3'b10	1	1	1	2	0x00000006	0xaabbccf5
9	134	0	1	3'b10	1	1	1	2	0x00000006	0xaabbccf9
9	135	0	1	3'b10	1	1	1	2	0x00000006	0xaabbccfd
9	136	0	1	3'b10	1	1	1	2	0x00000006	0xaabbcd01

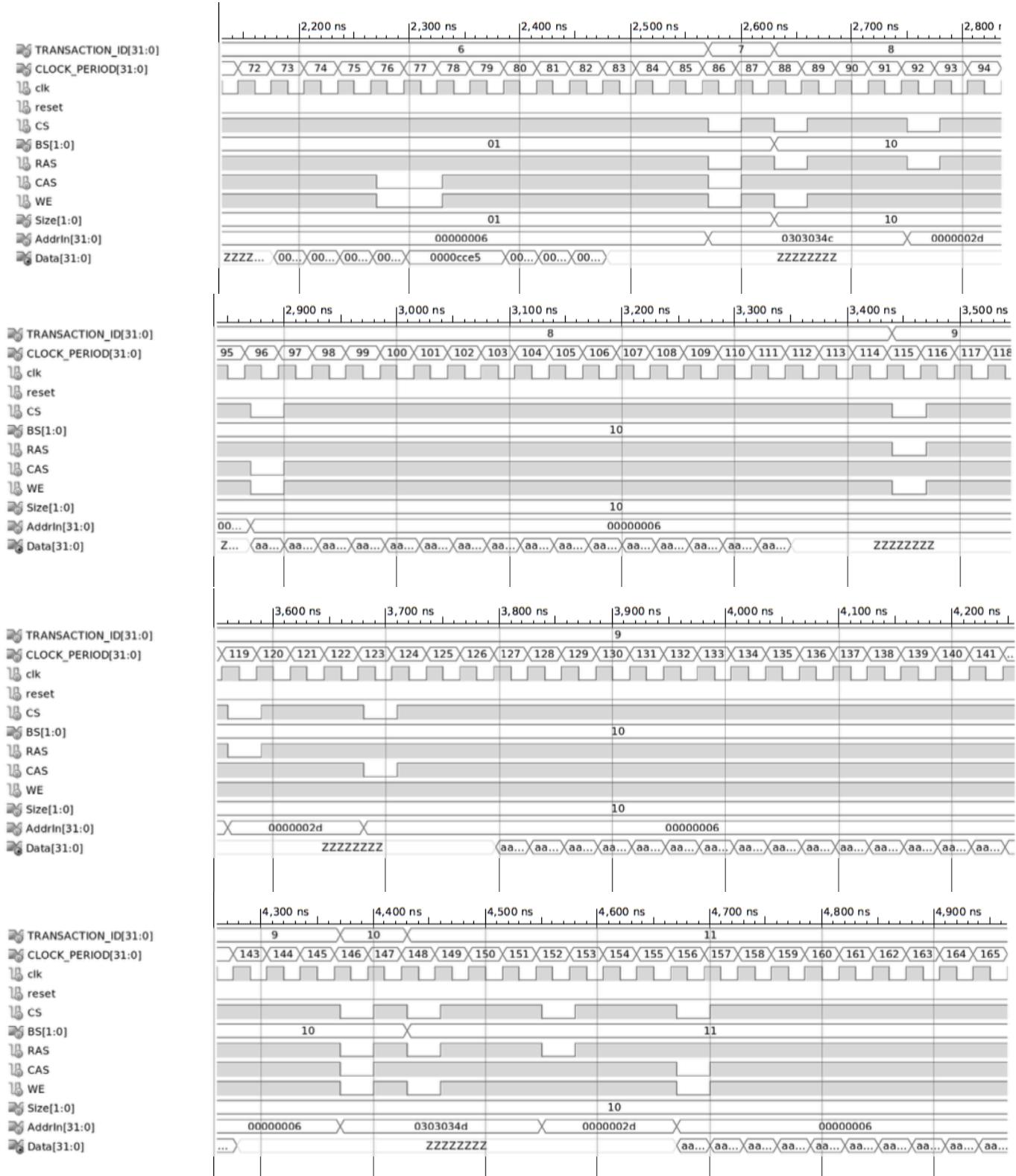
9	137	0	1	3'b10	1	1	1	2	0x00000006	0xaabbcd05
9	138	0	1	3'b10	1	1	1	2	0x00000006	0xaabbcd09
9	139	0	1	3'b10	1	1	1	2	0x00000006	0xaabbcd0d
9	140	0	1	3'b10	1	1	1	2	0x00000006	0xaabbcd11
9	141	0	1	3'b10	1	1	1	2	0x00000006	0xaabbcd15
9	142	0	1	3'b10	1	1	1	2	0x00000006	0xaabbcd19
9	143	0	1	3'b10	1	1	1	2	0x00000006	0xzzzzzzz
9	144	0	1	3'b10	1	1	1	2	0x00000006	0xzzzzzzz
9	145	0	1	3'b10	1	1	1	2	0x00000006	0xzzzzzzz
10	146	0	0	3'b10	0	0	0	2	0x0303034d	0xzzzzzzz
10	147	0	1	3'b10	1	1	1	2	0x0303034d	0xzzzzzzz
11	148	0	0	3'b11	0	1	0	2	0x0303034d	0xzzzzzzz
11	149	0	1	3'b11	1	1	1	2	0x0303034d	0xzzzzzzz
11	150	0	1	3'b11	1	1	1	2	0x0303034d	0xzzzzzzz
11	151	0	1	3'b11	1	1	1	2	0x0303034d	0xzzzzzzz
11	152	0	0	3'b11	0	1	1	2	0x0000002d	0xzzzzzzz
11	153	0	1	3'b11	1	1	1	2	0x0000002d	0xzzzzzzz
11	154	0	1	3'b11	1	1	1	2	0x0000002d	0xzzzzzzz
11	155	0	1	3'b11	1	1	1	2	0x0000002d	0xzzzzzzz
11	156	0	0	3'b11	1	0	0	2	0x00000006	0xaabbccdd
11	157	0	1	3'b11	1	1	1	2	0x00000006	0xaabbcce1
11	158	0	1	3'b11	1	1	1	2	0x00000006	0xaabbcce5
11	159	0	1	3'b11	1	1	1	2	0x00000006	0xaabbcce9
11	160	0	1	3'b11	1	1	1	2	0x00000006	0xaabbcced
11	161	0	1	3'b11	1	1	1	2	0x00000006	0xaabbccf1
11	162	0	1	3'b11	1	1	1	2	0x00000006	0xaabbccf5
11	163	0	1	3'b11	1	1	1	2	0x00000006	0xaabbccf9
11	164	0	1	3'b11	1	1	1	2	0x00000006	0xaabbccfd
11	165	0	1	3'b11	1	1	1	2	0x00000006	0xaabbcd01
11	166	0	1	3'b11	1	1	1	2	0x00000006	0xaabbcd05

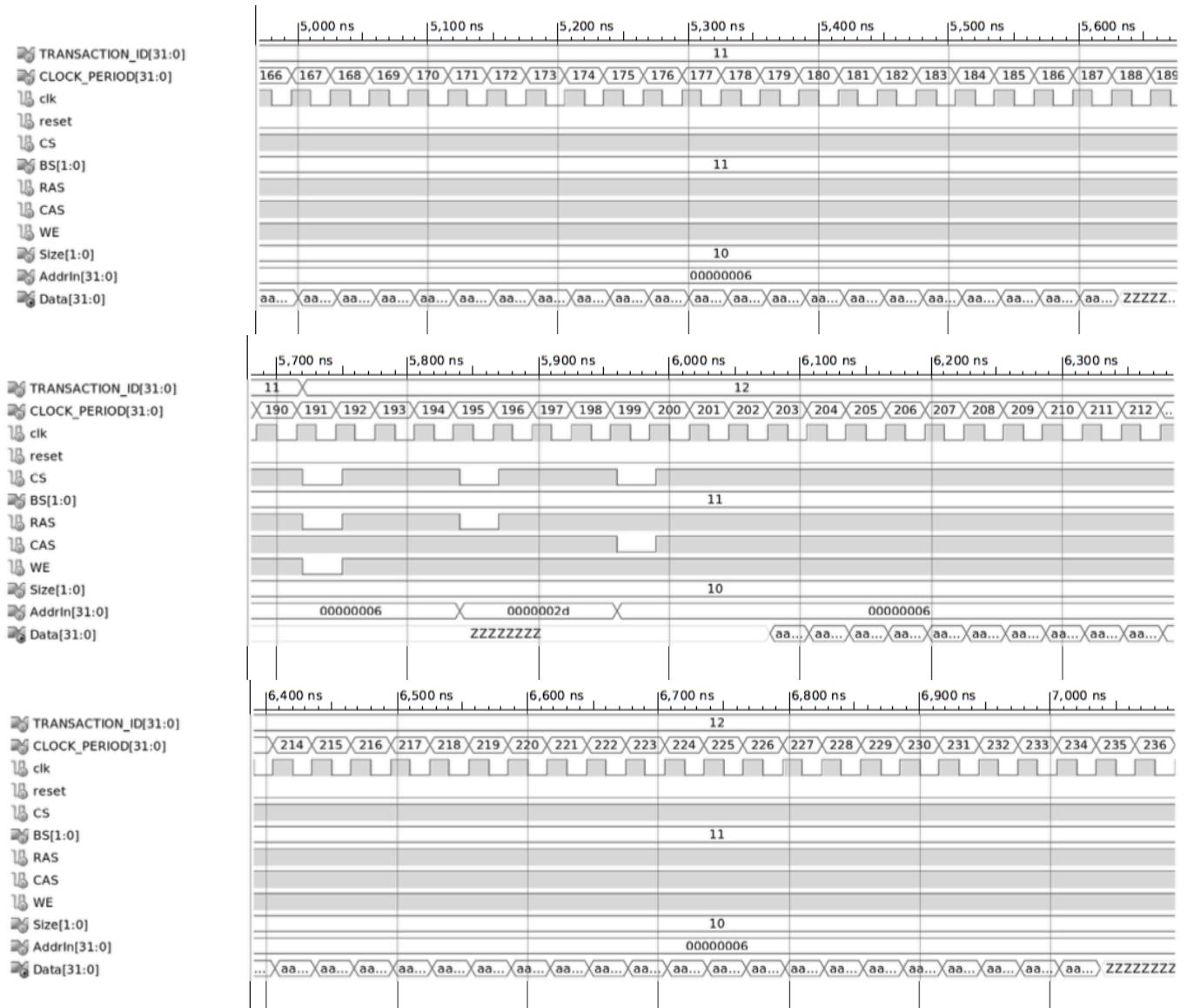
11	167	0	1	3'b11	1	1	1	2	0x00000006	0xaabbcd09
11	168	0	1	3'b11	1	1	1	2	0x00000006	0xaabbcd0d
11	169	0	1	3'b11	1	1	1	2	0x00000006	0xaabbcd11
11	170	0	1	3'b11	1	1	1	2	0x00000006	0xaabbcd15
11	171	0	1	3'b11	1	1	1	2	0x00000006	0xaabbcd19
11	172	0	1	3'b11	1	1	1	2	0x00000006	0xaabbcd1d
11	173	0	1	3'b11	1	1	1	2	0x00000006	0xaabbcd21
11	174	0	1	3'b11	1	1	1	2	0x00000006	0xaabbcd25
11	175	0	1	3'b11	1	1	1	2	0x00000006	0xaabbcd29
11	176	0	1	3'b11	1	1	1	2	0x00000006	0xaabbcd2d
11	177	0	1	3'b11	1	1	1	2	0x00000006	0xaabbcd31
11	178	0	1	3'b11	1	1	1	2	0x00000006	0xaabbcd35
11	179	0	1	3'b11	1	1	1	2	0x00000006	0xaabbcd39
11	180	0	1	3'b11	1	1	1	2	0x00000006	0xaabbcd3d
11	181	0	1	3'b11	1	1	1	2	0x00000006	0xaabbcd41
11	182	0	1	3'b11	1	1	1	2	0x00000006	0xaabbcd45
11	183	0	1	3'b11	1	1	1	2	0x00000006	0xaabbcd49
11	184	0	1	3'b11	1	1	1	2	0x00000006	0xaabbcd4d
11	185	0	1	3'b11	1	1	1	2	0x00000006	0xaabbcd51
11	186	0	1	3'b11	1	1	1	2	0x00000006	0xaabbcd55
11	187	0	1	3'b11	1	1	1	2	0x00000006	0xaabbcd59
11	188	0	1	3'b11	1	1	1	2	0x00000006	0xzzzzzzzz
11	189	0	1	3'b11	1	1	1	2	0x00000006	0xzzzzzzzz
11	190	0	1	3'b11	1	1	1	2	0x00000006	0xzzzzzzzz
12	191	0	0	3'b11	0	1	0	2	0x00000006	0xzzzzzzzz
12	192	0	1	3'b11	1	1	1	2	0x00000006	0xzzzzzzzz
12	193	0	1	3'b11	1	1	1	2	0x00000006	0xzzzzzzzz
12	194	0	1	3'b11	1	1	1	2	0x00000006	0xzzzzzzzz
12	195	0	0	3'b11	0	1	1	2	0x0000002d	0xzzzzzzzz
12	196	0	1	3'b11	1	1	1	2	0x0000002d	0xzzzzzzzz

12	197	0	1	3'b11	1	1	1	2	0x0000002d	0xxxxxxxx
12	198	0	1	3'b11	1	1	1	2	0x0000002d	0xxxxxxxx
12	199	0	0	3'b11	1	0	1	2	0x00000006	0xxxxxxxx
12	200	0	1	3'b11	1	1	1	2	0x00000006	0xxxxxxxx
12	201	0	1	3'b11	1	1	1	2	0x00000006	0xxxxxxxx
12	202	0	1	3'b11	1	1	1	2	0x00000006	0xxxxxxxx
12	203	0	1	3'b11	1	1	1	2	0x00000006	0xaabbccdd
12	204	0	1	3'b11	1	1	1	2	0x00000006	0xaabbce1
12	205	0	1	3'b11	1	1	1	2	0x00000006	0xaabbce5
12	206	0	1	3'b11	1	1	1	2	0x00000006	0xaabbce9
12	207	0	1	3'b11	1	1	1	2	0x00000006	0xaabbcced
12	208	0	1	3'b11	1	1	1	2	0x00000006	0xaabbccf1
12	209	0	1	3'b11	1	1	1	2	0x00000006	0xaabbccf5
12	210	0	1	3'b11	1	1	1	2	0x00000006	0xaabbccf9
12	211	0	1	3'b11	1	1	1	2	0x00000006	0xaabbccfd
12	212	0	1	3'b11	1	1	1	2	0x00000006	0xaabbc01
12	213	0	1	3'b11	1	1	1	2	0x00000006	0xaabbc05
12	214	0	1	3'b11	1	1	1	2	0x00000006	0xaabbc09
12	215	0	1	3'b11	1	1	1	2	0x00000006	0xaabbc0d
12	216	0	1	3'b11	1	1	1	2	0x00000006	0xaabbc11
12	217	0	1	3'b11	1	1	1	2	0x00000006	0xaabbc15
12	218	0	1	3'b11	1	1	1	2	0x00000006	0xaabbc19
12	219	0	1	3'b11	1	1	1	2	0x00000006	0xaabbc1d
12	220	0	1	3'b11	1	1	1	2	0x00000006	0xaabbc21
12	221	0	1	3'b11	1	1	1	2	0x00000006	0xaabbc25
12	222	0	1	3'b11	1	1	1	2	0x00000006	0xaabbc29
12	223	0	1	3'b11	1	1	1	2	0x00000006	0xaabbc2d
12	224	0	1	3'b11	1	1	1	2	0x00000006	0xaabbc31
12	225	0	1	3'b11	1	1	1	2	0x00000006	0xaabbc35
12	226	0	1	3'b11	1	1	1	2	0x00000006	0xaabbc39

12	227	0	1	3'b11	1	1	1	2	0x00000006	0xaabbcd3d
12	228	0	1	3'b11	1	1	1	2	0x00000006	0xaabbcd41
12	229	0	1	3'b11	1	1	1	2	0x00000006	0xaabbcd45
12	230	0	1	3'b11	1	1	1	2	0x00000006	0xaabbcd49
12	231	0	1	3'b11	1	1	1	2	0x00000006	0xaabbcd4d
12	232	0	1	3'b11	1	1	1	2	0x00000006	0xaabbcd51
12	233	0	1	3'b11	1	1	1	2	0x00000006	0xaabbcd55
12	234	0	1	3'b11	1	1	1	2	0x00000006	0xaabbcd59
12	235	0	1	3'b11	1	1	1	2	0x00000006	0xzzzzzzz
12	236	0	1	3'b11	1	1	1	2	0x00000006	0xzzzzzzz







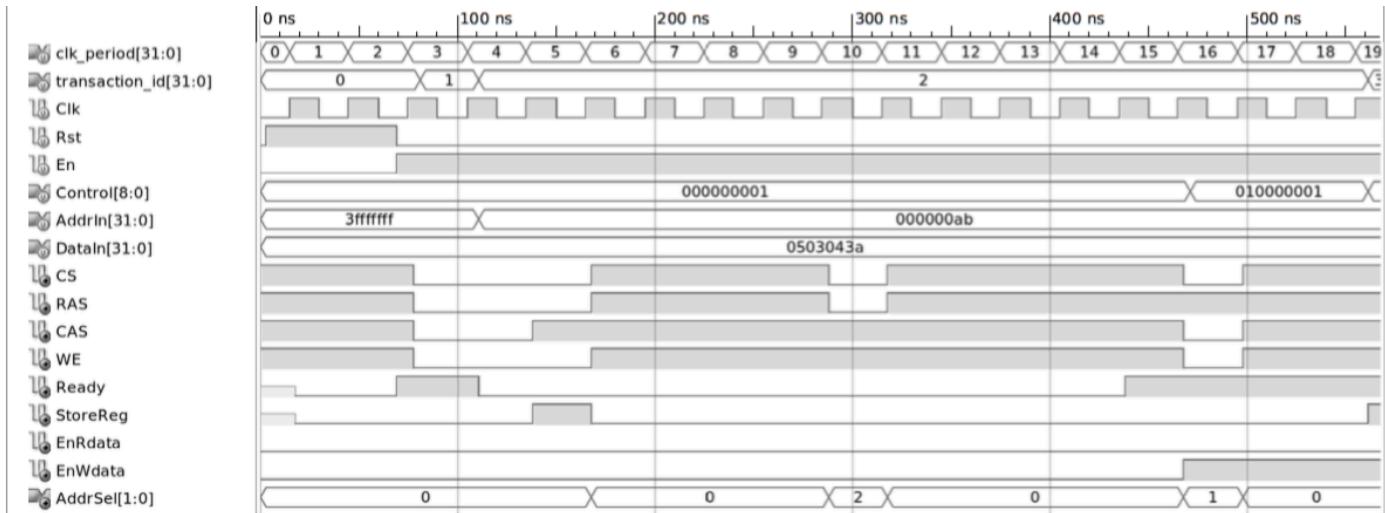
9. BIU Register File

This register file module here is the same as the one in the SDRAM module.

The input signals are given by the bus master. In this implementation address 3FFFFFFF signifies programming mode. The data dispatched by the master in the next cycle is programmed in.

Clock period 3 triggers a program mode. The data is provided in clock period 4 by the master. This data is stored in the register file and prog_mode goes high. This signal gets picked up by the controller which issues a program command to the SDRAM with the same data.

Clock period	Reset	MasterBusy	AddrIn	DataIn	En	tburst	addr_mode	tlat	tpre	twait	tcas	prog_mode
1	1	0	0	0	1	x	x	x	xx	xx	xx	x
2	0	0	0	0	1	0	0	0	0	0	0	0
3	0	0	3fffffff	0	1	0	0	0	0	0	0	0
4	0	0	1234567	060708af	1	0	0	0	0	0	0	1
5	0	0	3fffffff	0	1	7	1	a	8	7	6	0
6	0	1	3fffffff	abcdef12	1	7	1	a	8	7	6	1
7	0	0	3fffffff	abcdef12	1	7	1	a	8	7	6	1
8	0	0	3fffffff	abcdef12	0	2	0	a	8	7	6	1
9	0	0	3fffffff	abcdef12	0	2	0	a	8	7	6	0



10. BIU Timing Generators

The verification of the timing generator of the BIU is automated. The logs are captured from the verifier as shown below. There are a number of transactions and all have been verified successfully. Refer to the waveform below.

Clk period : 1, beginning write

*Timer load verified; Clk period: 2
Checking for precharge ---> PASSED
Count-down counter output: 7; Clk period: 3
Count-down counter output: 6; Clk period: 4
Count-down counter output: 5; Clk period: 5
Count-down counter output: 4; Clk period: 6
Count-down counter output: 3; Clk period: 7
Count-down counter output: 2; Clk period: 8
Count-down counter output: 1; Clk period: 9
Precharge phase SUCCESSFUL*

*Timer load verified; Clk period: 9
Checking for activation ---> PASSED
Count-down counter output: 8; Clk period: 10
Count-down counter output: 7; Clk period: 11
Count-down counter output: 6; Clk period: 12
Count-down counter output: 5; Clk period: 13
Count-down counter output: 4; Clk period: 14
Count-down counter output: 3; Clk period: 15
Count-down counter output: 2; Clk period: 16
Count-down counter output: 1; Clk period: 17
Activation phase SUCCESSFUL*

*Timer load verified; Clk period: 17
Checking for write burst ---> PASSED
Count-down counter output: 7; Clk period: 18
Count-down counter output: 6; Clk period: 19
Count-down counter output: 5; Clk period: 20
Count-down counter output: 4; Clk period: 21
Count-down counter output: 3; Clk period: 22
Count-down counter output: 2; Clk period: 23
No busy cycles requested
Count-down counter output: 1, Clk period: 24
Write burst phase SUCCESSFUL*

*Timer load verified; Clk period: 24
Checking for wait period ---> PASSED
Count-down counter output: 10; Clk period: 25
Count-down counter output: 9; Clk period: 26
Count-down counter output: 8; Clk period: 27
Count-down counter output: 7; Clk period: 28
Count-down counter output: 6; Clk period: 29
Count-down counter output: 5; Clk period: 30
Count-down counter output: 4; Clk period: 31
Count-down counter output: 3; Clk period: 32
Count-down counter output: 2; Clk period: 33
Count-down counter output: 1; Clk period: 34
Wait phase SUCCESSFUL*

*Completed WRITE transaction
xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx*

Changed burst length to 2

Clk period : 35, beginning write

*Timer load verified; Clk period: 36
Checking for precharge ---> PASSED
Count-down counter output: 7; Clk period: 37
Count-down counter output: 6; Clk period: 38
Count-down counter output: 5; Clk period: 39
Count-down counter output: 4; Clk period: 40
Count-down counter output: 3; Clk period: 41
Count-down counter output: 2; Clk period: 42
Count-down counter output: 1; Clk period: 43
Precharge phase SUCCESSFUL*

*Timer load verified; Clk period: 43
Checking for activation ---> PASSED
Count-down counter output: 8; Clk period: 44
Count-down counter output: 7; Clk period: 45
Count-down counter output: 6; Clk period: 46
Count-down counter output: 5; Clk period: 47
Count-down counter output: 4; Clk period: 48
Count-down counter output: 3; Clk period: 49
Count-down counter output: 2; Clk period: 50
Count-down counter output: 1; Clk period: 51
Activation phase SUCCESSFUL*

*Timer load verified; Clk period: 51
Checking for write burst ---> PASSED
Count-down counter output: 1; Clk period: 52
No busy cycles requested
Count-down counter output: 1, Clk period: 52
Write burst phase SUCCESSFUL*

*Timer load verified; Clk period: 52
Checking for wait period ---> PASSED
Count-down counter output: 10; Clk period: 53
Count-down counter output: 9; Clk period: 54
Count-down counter output: 8; Clk period: 55
Count-down counter output: 7; Clk period: 56
Count-down counter output: 6; Clk period: 57
Count-down counter output: 5; Clk period: 58
Count-down counter output: 4; Clk period: 59
Count-down counter output: 3; Clk period: 60
Count-down counter output: 2; Clk period: 61
Count-down counter output: 1; Clk period: 62
Wait phase SUCCESSFUL*

*Completed WRITE transaction
xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx*

Changed burst length to 4

=====

Clk period : 63, beginning read

Timer load verified; Clk period: 64
 Checking for precharge ---> PASSED
 Count-down counter output: 7; Clk period: 65
 Count-down counter output: 6; Clk period: 66
 Count-down counter output: 5; Clk period: 67
 Count-down counter output: 4; Clk period: 68
 Count-down counter output: 3; Clk period: 69
 Count-down counter output: 2; Clk period: 70
 Count-down counter output: 1; Clk period: 71
 Precharge phase SUCCESSFUL

Timer load verified; Clk period: 71
 Checking for activation ---> PASSED
 Count-down counter output: 8; Clk period: 72
 Count-down counter output: 7; Clk period: 73
 Count-down counter output: 6; Clk period: 74
 Count-down counter output: 5; Clk period: 75
 Count-down counter output: 4; Clk period: 76
 Count-down counter output: 3; Clk period: 77
 Count-down counter output: 2; Clk period: 78
 Count-down counter output: 1; Clk period: 79
 Activation phase SUCCESSFUL

Timer load verified; Clk period: 79
 Checking for read latency ---> PASSED
 Count-down counter output: 2; Clk period: 80
 Count-down counter output: 1; Clk period: 81
 Latency phase SUCCESSFUL

Timer load verified; Clk period: 81
 Checking for read burst ---> PASSED
 Count-down counter output: 4; Clk period: 82
 Count-down counter output: 3; Clk period: 83
 Count-down counter output: 2; Clk period: 84
 No busy cycles requested
 Count-down counter output: 1, Clk period: 85
 Read burst phase SUCCESSFUL

Timer load verified; Clk period: 85
 Checking for wait period ---> PASSED
 Count-down counter output: 10; Clk period: 86
 Count-down counter output: 9; Clk period: 87
 Count-down counter output: 8; Clk period: 88
 Count-down counter output: 7; Clk period: 89
 Count-down counter output: 6; Clk period: 90
 Count-down counter output: 5; Clk period: 91
 Count-down counter output: 4; Clk period: 92
 Count-down counter output: 3; Clk period: 93
 Count-down counter output: 2; Clk period: 94
 Count-down counter output: 1; Clk period: 95
 Wait phase SUCCESSFUL

Completed READ transaction
 xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx

Changed burst length to 8

=====

Clk period : 96, beginning read

Timer load verified; Clk period: 97
 Checking for precharge ---> PASSED
 Count-down counter output: 7; Clk period: 98
 Count-down counter output: 6; Clk period: 99
 Count-down counter output: 5; Clk period: 100
 Count-down counter output: 4; Clk period: 101
 Count-down counter output: 3; Clk period: 102
 Count-down counter output: 2; Clk period: 103
 Count-down counter output: 1; Clk period: 104
 Precharge phase SUCCESSFUL

Timer load verified; Clk period: 104
 Checking for activation ---> PASSED
 Count-down counter output: 8; Clk period: 105
 Count-down counter output: 7; Clk period: 106
 Count-down counter output: 6; Clk period: 107
 Count-down counter output: 5; Clk period: 108
 Count-down counter output: 4; Clk period: 109
 Count-down counter output: 3; Clk period: 110
 Count-down counter output: 2; Clk period: 111
 Count-down counter output: 1; Clk period: 112
 Activation phase SUCCESSFUL

Timer load verified; Clk period: 112
 Checking for read latency ---> PASSED
 Count-down counter output: 2; Clk period: 113
 Count-down counter output: 1; Clk period: 114
 Latency phase SUCCESSFUL

Timer load verified; Clk period: 114
 Checking for read burst ---> PASSED
 Count-down counter output: 8; Clk period: 115
 Count-down counter output: 7; Clk period: 116
 Count-down counter output: 6; Clk period: 117
 Emulating master busy; Clk period: 118
 Emulating master busy; Clk period: 119
 Emulating master busy; Clk period: 120
 Emulating master busy; Clk period: 121
 Emulating master busy; Clk period: 122
 BUSY phase verified successfully
 Count-down counter output: 5; Clk period: 123
 Count-down counter output: 4; Clk period: 124
 Count-down counter output: 3; Clk period: 125
 Count-down counter output: 2; Clk period: 126
 Count-down counter output: 1, Clk period: 127
 Read burst phase SUCCESSFUL

Timer load verified; Clk period: 127
 Checking for wait period ---> PASSED
 Count-down counter output: 10; Clk period: 128
 Count-down counter output: 9; Clk period: 129
 Count-down counter output: 8; Clk period: 130
 Count-down counter output: 7; Clk period: 131
 Count-down counter output: 6; Clk period: 132
 Count-down counter output: 5; Clk period: 133

Count-down counter output: 4; Clk period: 134
 Count-down counter output: 3; Clk period: 135
 Count-down counter output: 2; Clk period: 136
 Count-down counter output: 1; Clk period: 137
 Wait phase **SUCCESSFUL**

Completed READ transaction
 xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx

Changed burst length to 16

Clk period : 138, beginning write

Timer load verified; Clk period: 139
 Checking for precharge ---> PASSED
 Count-down counter output: 7; Clk period: 140
 Count-down counter output: 6; Clk period: 141
 Count-down counter output: 5; Clk period: 142
 Count-down counter output: 4; Clk period: 143
 Count-down counter output: 3; Clk period: 144
 Count-down counter output: 2; Clk period: 145
 Count-down counter output: 1; Clk period: 146
 Precharge phase **SUCCESSFUL**

Timer load verified; Clk period: 146
 Checking for activation ---> PASSED
 Count-down counter output: 8; Clk period: 147
 Count-down counter output: 7; Clk period: 148
 Count-down counter output: 6; Clk period: 149
 Count-down counter output: 5; Clk period: 150
 Count-down counter output: 4; Clk period: 151
 Count-down counter output: 3; Clk period: 152
 Count-down counter output: 2; Clk period: 153
 Count-down counter output: 1; Clk period: 154
 Activation phase **SUCCESSFUL**

Timer load verified; Clk period: 154
 Checking for write burst ---> PASSED
 Count-down counter output: 15; Clk period: 155
 Count-down counter output: 14; Clk period: 156
 Count-down counter output: 13; Clk period: 157
 Count-down counter output: 12; Clk period: 158
 Count-down counter output: 11; Clk period: 159
 Emulating master busy; Clk period: 160
 Emulating master busy; Clk period: 161
 Emulating master busy; Clk period: 162
 Emulating master busy; Clk period: 163
 Emulating master busy; Clk period: 164
 Emulating master busy; Clk period: 165
 Emulating master busy; Clk period: 166
 Emulating master busy; Clk period: 167
 Emulating master busy; Clk period: 168
 Emulating master busy; Clk period: 169
 BUSY phase verified successfully
 Count-down counter output: 10; Clk period: 170
 Count-down counter output: 9; Clk period: 171
 Count-down counter output: 8; Clk period: 172
 Count-down counter output: 7; Clk period: 173

Count-down counter output: 6; Clk period: 174
 Count-down counter output: 5; Clk period: 175
 Count-down counter output: 4; Clk period: 176
 Count-down counter output: 3; Clk period: 177
 Count-down counter output: 2; Clk period: 178
 Count-down counter output: 1; Clk period: 179
 Write burst phase **SUCCESSFUL**

Timer load verified; Clk period: 179
 Checking for wait period ---> PASSED
 Count-down counter output: 10; Clk period: 180
 Count-down counter output: 9; Clk period: 181
 Count-down counter output: 8; Clk period: 182
 Count-down counter output: 7; Clk period: 183
 Count-down counter output: 6; Clk period: 184
 Count-down counter output: 5; Clk period: 185
 Count-down counter output: 4; Clk period: 186
 Count-down counter output: 3; Clk period: 187
 Count-down counter output: 2; Clk period: 188
 Count-down counter output: 1; Clk period: 189
 Wait phase **SUCCESSFUL**

Completed WRITE transaction
 xxx

Changed burst length to 32

Clk period : 190, beginning read

Timer load verified; Clk period: 191
 Checking for precharge ---> PASSED
 Count-down counter output: 7; Clk period: 192
 Count-down counter output: 6; Clk period: 193
 Count-down counter output: 5; Clk period: 194
 Count-down counter output: 4; Clk period: 195
 Count-down counter output: 3; Clk period: 196
 Count-down counter output: 2; Clk period: 197
 Count-down counter output: 1; Clk period: 198
 Precharge phase **SUCCESSFUL**

Timer load verified; Clk period: 198
 Checking for activation ---> PASSED
 Count-down counter output: 8; Clk period: 199
 Count-down counter output: 7; Clk period: 200
 Count-down counter output: 6; Clk period: 201
 Count-down counter output: 5; Clk period: 202
 Count-down counter output: 4; Clk period: 203
 Count-down counter output: 3; Clk period: 204
 Count-down counter output: 2; Clk period: 205
 Count-down counter output: 1; Clk period: 206
 Activation phase **SUCCESSFUL**

Timer load verified; Clk period: 206
 Checking for read latency ---> PASSED
 Count-down counter output: 2; Clk period: 207
 Count-down counter output: 1; Clk period: 208
 Latency phase **SUCCESSFUL**

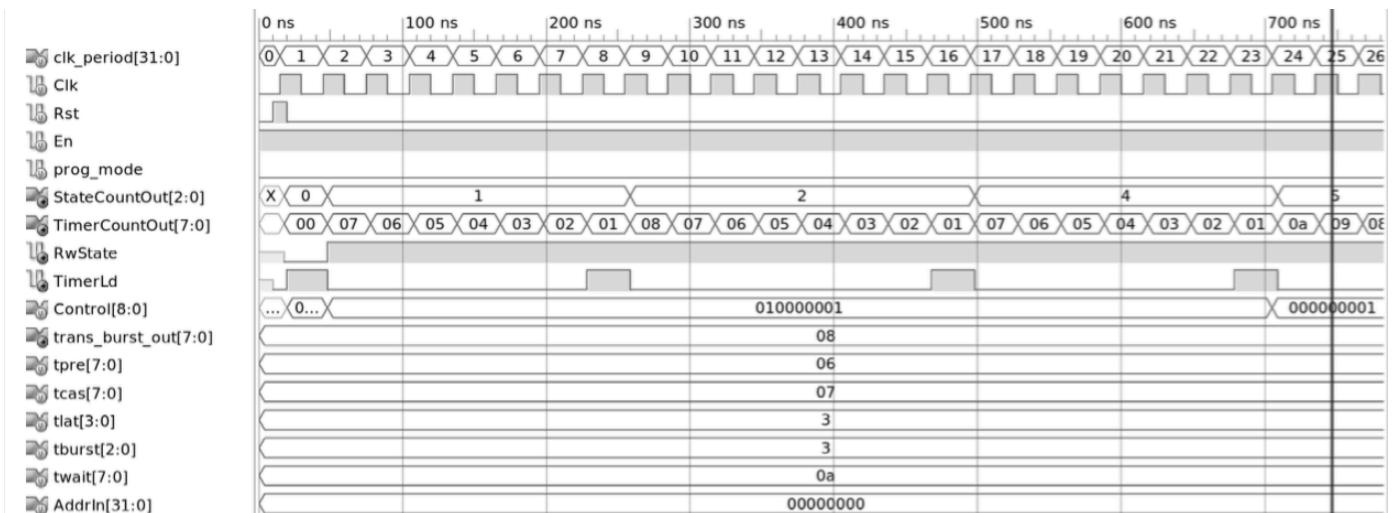
Timer load verified; Clk period: 208
 Checking for read burst ---> PASSED
 Count-down counter output: 32; Clk period: 209
 Count-down counter output: 31; Clk period: 210
 Count-down counter output: 30; Clk period: 211
 Count-down counter output: 29; Clk period: 212
 Count-down counter output: 28; Clk period: 213
 Count-down counter output: 27; Clk period: 214
 Count-down counter output: 26; Clk period: 215
 Count-down counter output: 25; Clk period: 216
 Count-down counter output: 24; Clk period: 217
 Count-down counter output: 23; Clk period: 218
 Count-down counter output: 22; Clk period: 219
 Count-down counter output: 21; Clk period: 220
 Count-down counter output: 20; Clk period: 221
 Count-down counter output: 19; Clk period: 222
 Count-down counter output: 18; Clk period: 223
 Count-down counter output: 17; Clk period: 224
 Count-down counter output: 16; Clk period: 225
 Count-down counter output: 15; Clk period: 226
 Count-down counter output: 14; Clk period: 227
 Count-down counter output: 13; Clk period: 228
 Count-down counter output: 12; Clk period: 229
 Count-down counter output: 11; Clk period: 230
 Count-down counter output: 10; Clk period: 231
 Count-down counter output: 9; Clk period: 232
 Count-down counter output: 8; Clk period: 233
 Count-down counter output: 7; Clk period: 234

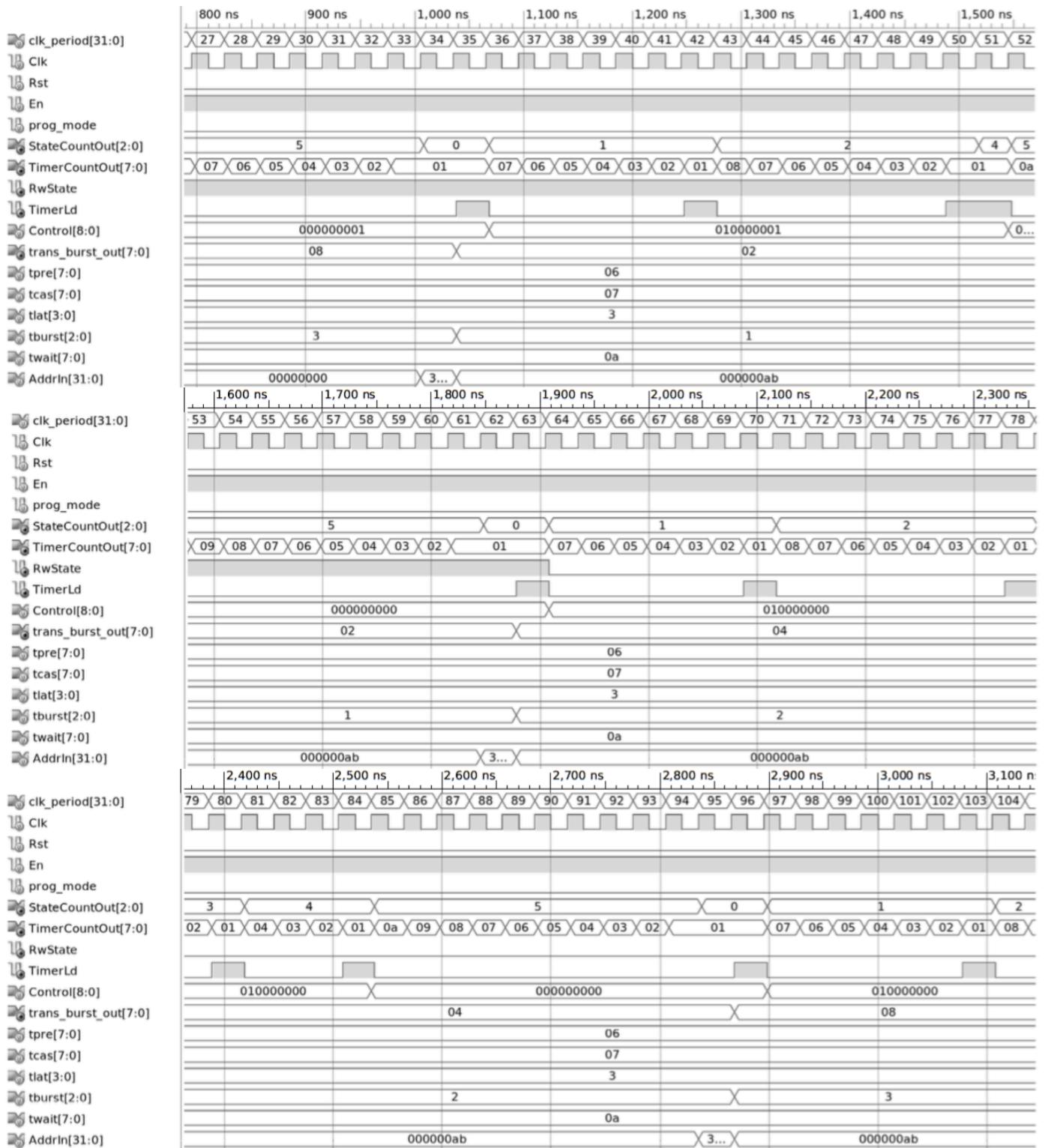
Count-down counter output: 6; Clk period: 235
 Count-down counter output: 5; Clk period: 236
 Count-down counter output: 4; Clk period: 237
 Emulating master busy; Clk period: 238
 Emulating master busy; Clk period: 239
 Emulating master busy; Clk period: 240
 BUSY phase verified successfully
 Count-down counter output: 3; Clk period: 241
 Count-down counter output: 2; Clk period: 242
 Count-down counter output: 1; Clk period: 243
 Read burst phase SUCCESSFUL

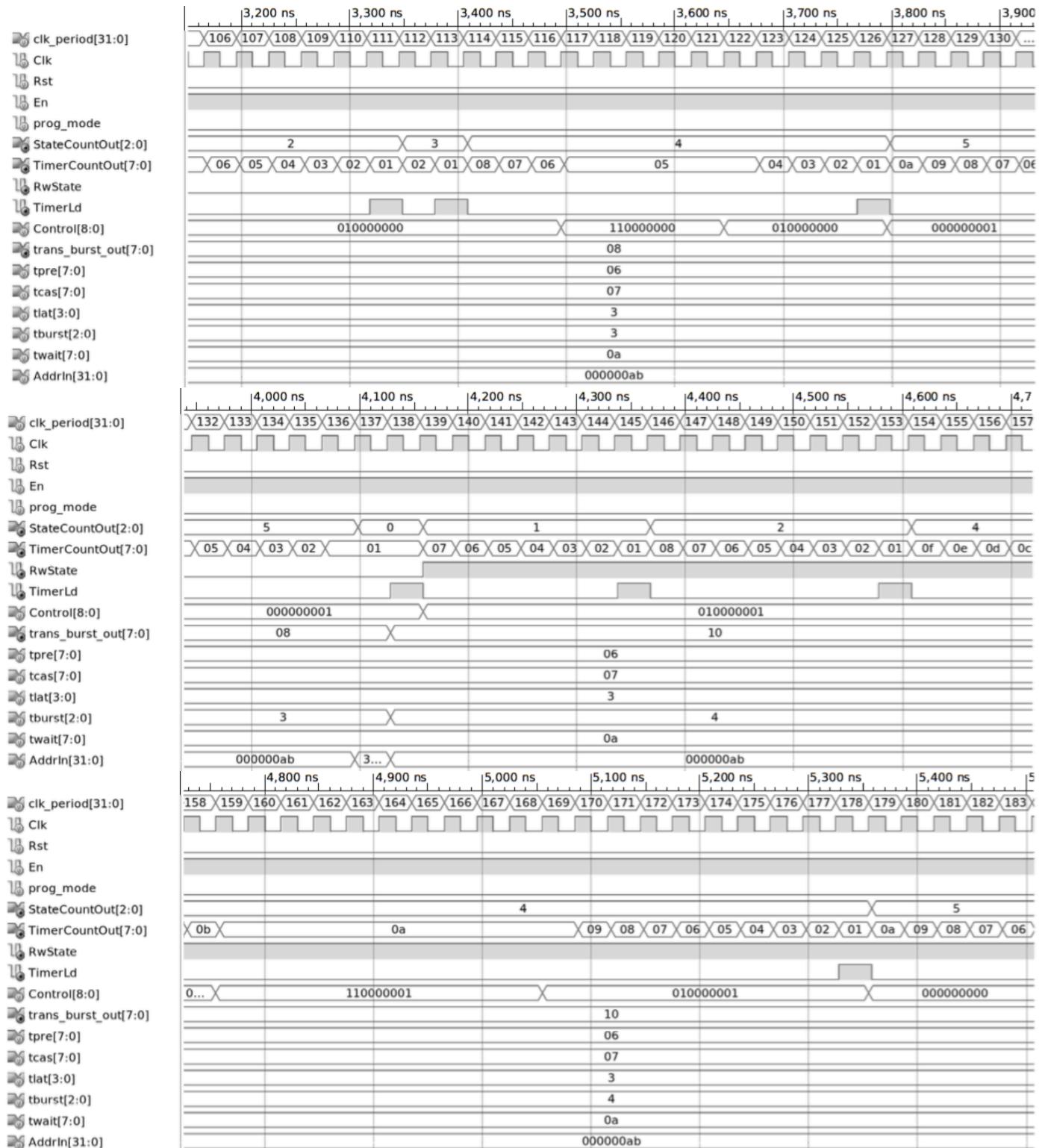
Timer load verified; Clk period: 243
 Checking for wait period ---> PASSED
 Count-down counter output: 10; Clk period: 244
 Count-down counter output: 9; Clk period: 245
 Count-down counter output: 8; Clk period: 246
 Count-down counter output: 7; Clk period: 247
 Count-down counter output: 6; Clk period: 248
 Count-down counter output: 5; Clk period: 249
 Count-down counter output: 4; Clk period: 250
 Count-down counter output: 3; Clk period: 251
 Count-down counter output: 2; Clk period: 252
 Count-down counter output: 1; Clk period: 253
 Wait phase SUCCESSFUL

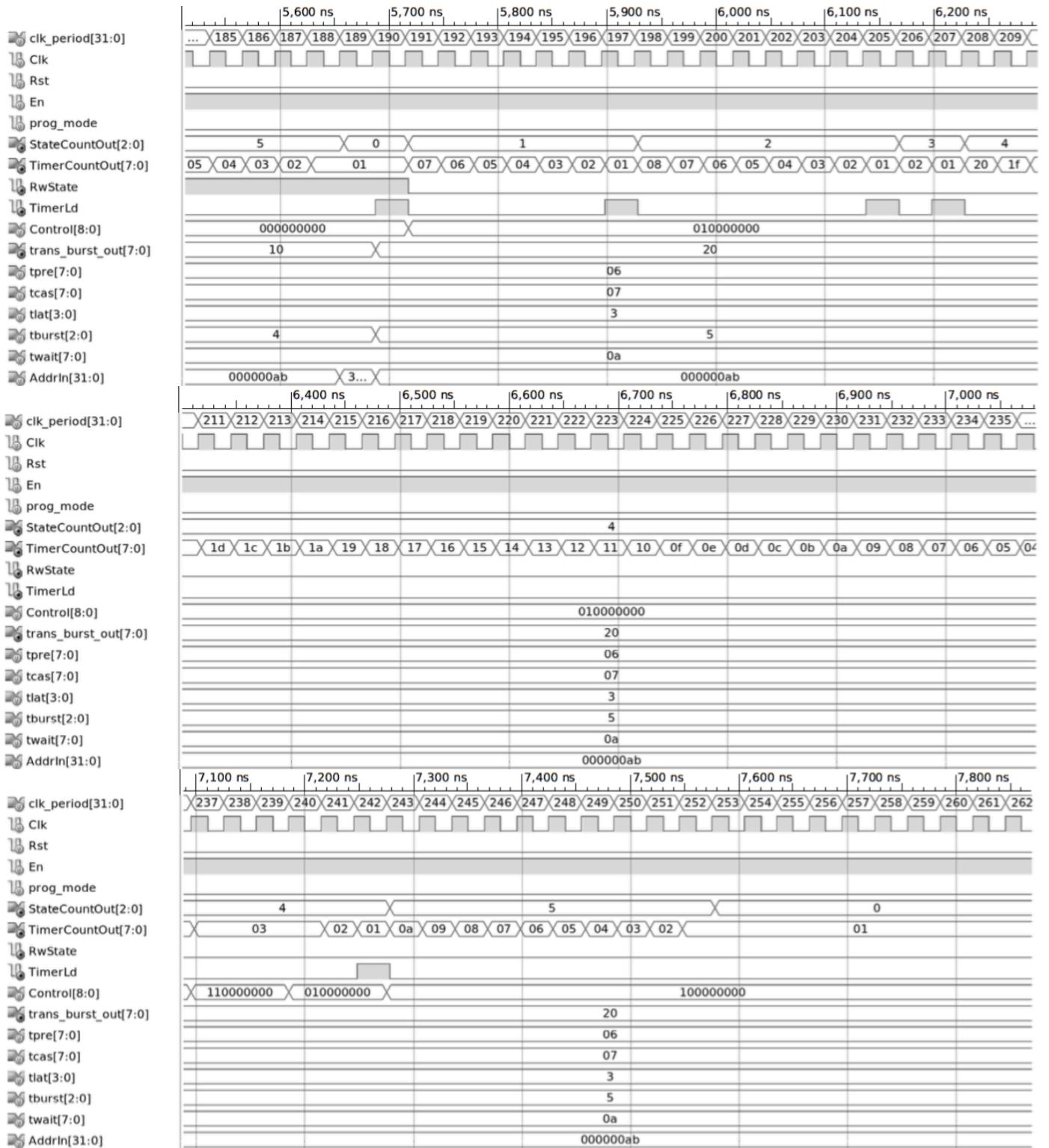
Completed READ transaction

xx









11. BIU Controller

The BIU controller generates the control signals for the SDRAM and the Ready signal for the bus master. A sample transaction can be followed as below.

In clock period 4 the master issues a program command by generating address 3FFFFFFF. The ready is high here. The master will proceed with the next address and controls while providing the program data in clock period 5.

In clock period 5, the ready signal goes low. The BIU generates the program command for the SDRAM interface.

In clock period 5, the controls from the master says START a write transaction (transaction 2). The BIU issues a precharge command for the BIU and triggers StoreReg to store the current address from the master. As verified before, the tPre period is counted down. In clock period 11, an activation signal issued by the BIU. Note that the tPre was programmed as 5 cycles.

After tCas period, a write command should be issued in clock period 17. The ready signal goes high in clock period 16 causing the master to proceed to continue state while issuing the first data packet in clock period 17. A burst of 4 packets are written between clock periods 17 and 20.

Note that EnWdata goes high during the write burst and the ready stays high.

During activation (clock period 11) AddrSel is 2, selecting the row address and during write command (clock period 17), AddrSel is 1 selecting the column address.

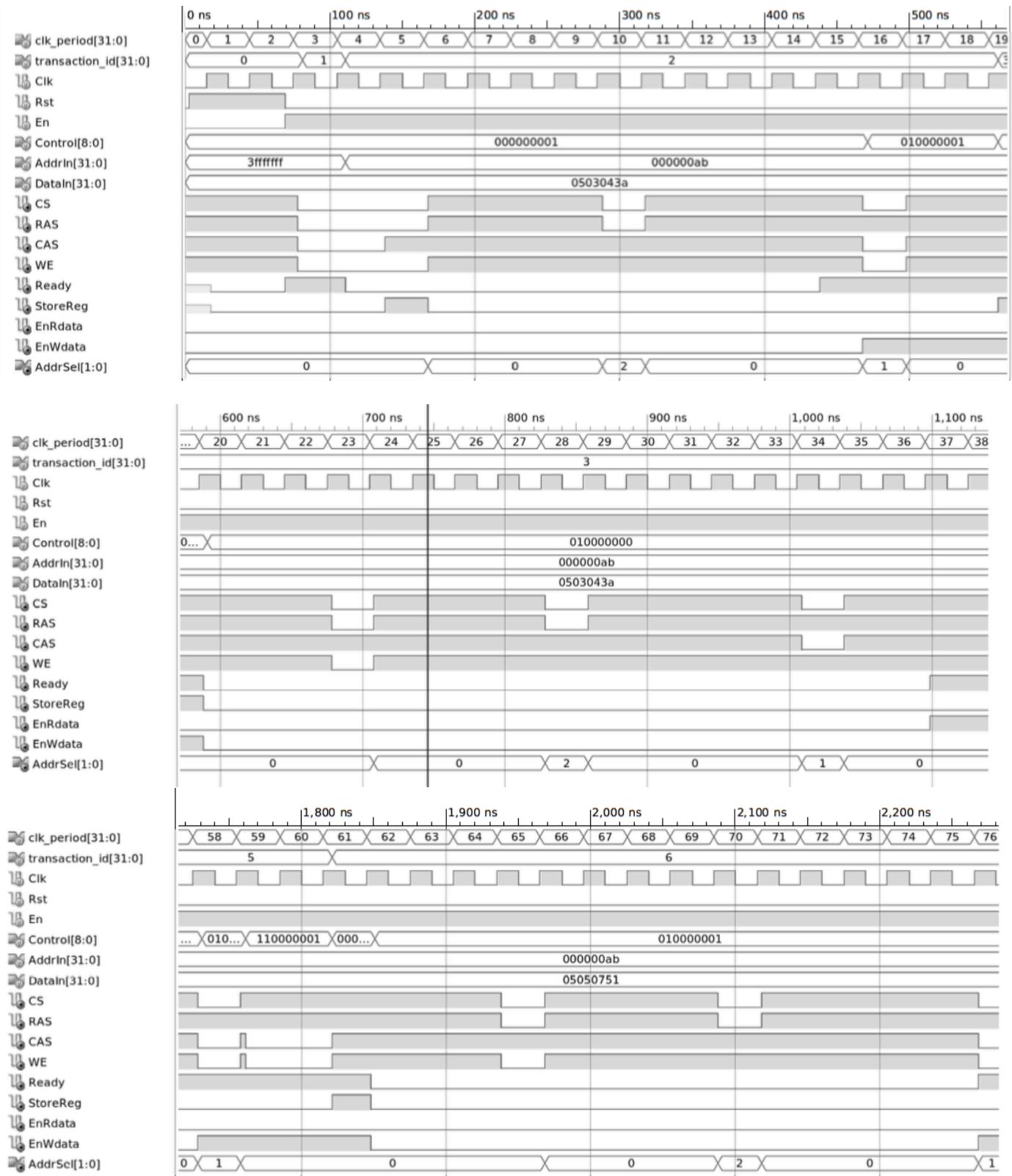
Similarly, all other transactions have been verified. Waveforms are provided below.

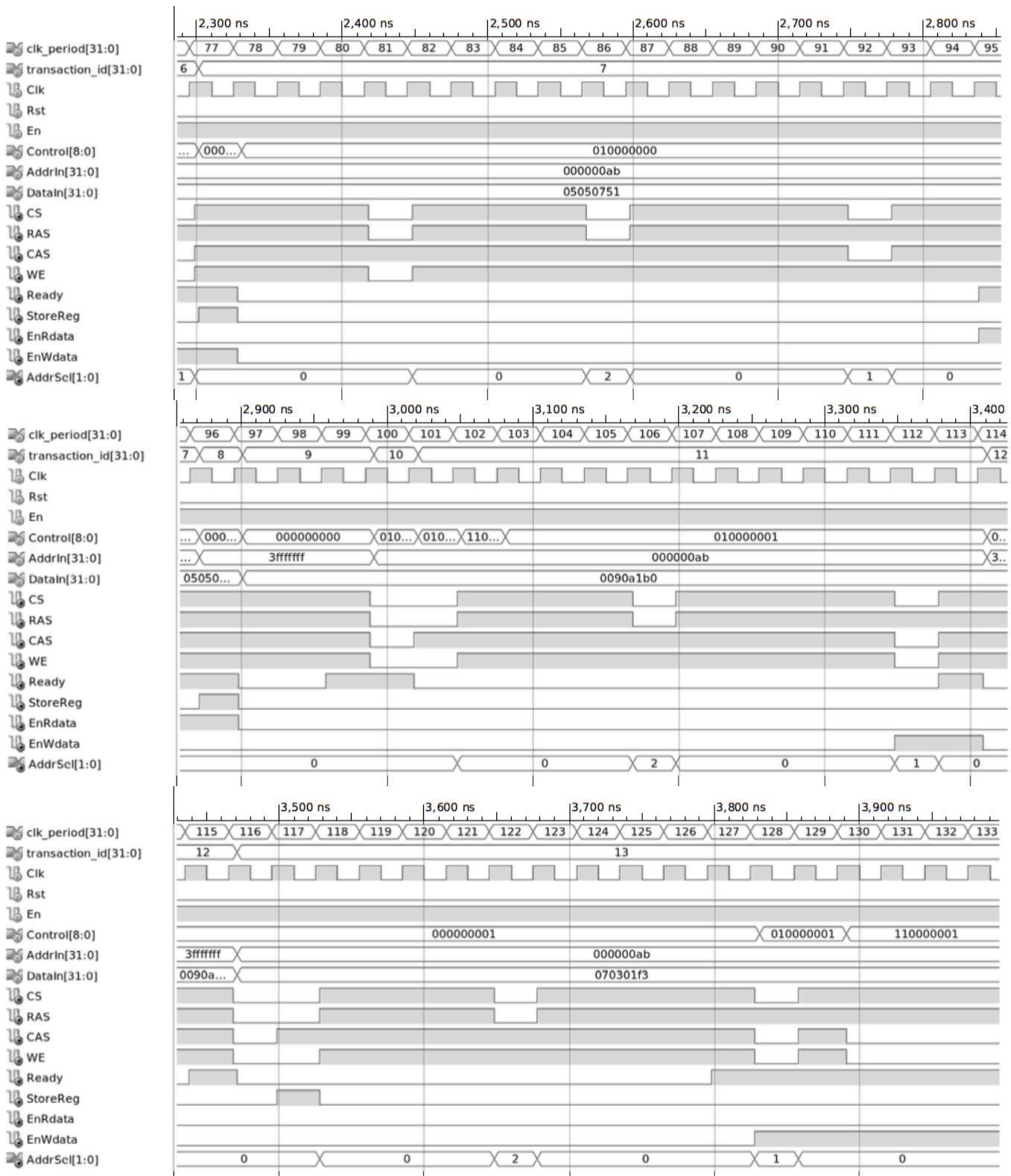
Clk	Transaction ID	Rst	En	Control	AddrIn	DataIn	CS	RAS	CAS	WE	Ready	StoreReg	EnRdata	EnWdata	AddrSel
1	0	1	0	1	0x3fffffff	0x0503043a	1	1	1	1	x	x	0	0	0
2	0	1	0	1	0x3fffffff	0x0503043a	1	1	1	1	0	0	0	0	0
3	0	0	1	1	0x3fffffff	0x0503043a	1	1	1	1	1	0	0	0	0
4	1	0	1	1	0x3fffffff	0x0503043a	0	0	0	0	1	0	0	0	0
5	2	0	1	1	0x000000ab	0x0503043a	0	0	0	0	0	0	0	0	0
6	2	0	1	1	0x000000ab	0x0503043a	0	0	1	0	0	1	0	0	0
7	2	0	1	1	0x000000ab	0x0503043a	1	1	1	1	0	0	0	0	0
8	2	0	1	1	0x000000ab	0x0503043a	1	1	1	1	0	0	0	0	0
9	2	0	1	1	0x000000ab	0x0503043a	1	1	1	1	0	0	0	0	0
10	2	0	1	1	0x000000ab	0x0503043a	1	1	1	1	0	0	0	0	0
11	2	0	1	1	0x000000ab	0x0503043a	0	0	1	1	0	0	0	0	2
12	2	0	1	1	0x000000ab	0x0503043a	1	1	1	1	0	0	0	0	0
13	2	0	1	1	0x000000ab	0x0503043a	1	1	1	1	0	0	0	0	0
14	2	0	1	1	0x000000ab	0x0503043a	1	1	1	1	0	0	0	0	0
15	2	0	1	1	0x000000ab	0x0503043a	1	1	1	1	0	0	0	0	0
16	2	0	1	1	0x000000ab	0x0503043a	1	1	1	1	1	0	0	0	0
17	2	0	1	10000001	0x000000ab	0x0503043a	0	1	0	0	1	0	0	1	1
18	2	0	1	10000001	0x000000ab	0x0503043a	1	1	1	1	1	0	0	1	0
19	2	0	1	10000001	0x000000ab	0x0503043a	1	1	1	1	1	0	0	1	0
20	3	0	1	0	0x000000ab	0x0503043a	1	1	1	1	1	1	0	1	0
21	3	0	1	10000000	0x000000ab	0x0503043a	1	1	1	1	0	0	0	0	0
22	3	0	1	10000000	0x000000ab	0x0503043a	1	1	1	1	0	0	0	0	0
23	3	0	1	10000000	0x000000ab	0x0503043a	1	1	1	1	0	0	0	0	0
24	3	0	1	10000000	0x000000ab	0x0503043a	0	0	1	0	0	0	0	0	0
25	3	0	1	10000000	0x000000ab	0x0503043a	1	1	1	1	0	0	0	0	0
26	3	0	1	10000000	0x000000ab	0x0503043a	1	1	1	1	0	0	0	0	0
27	3	0	1	10000000	0x000000ab	0x0503043a	1	1	1	1	0	0	0	0	0

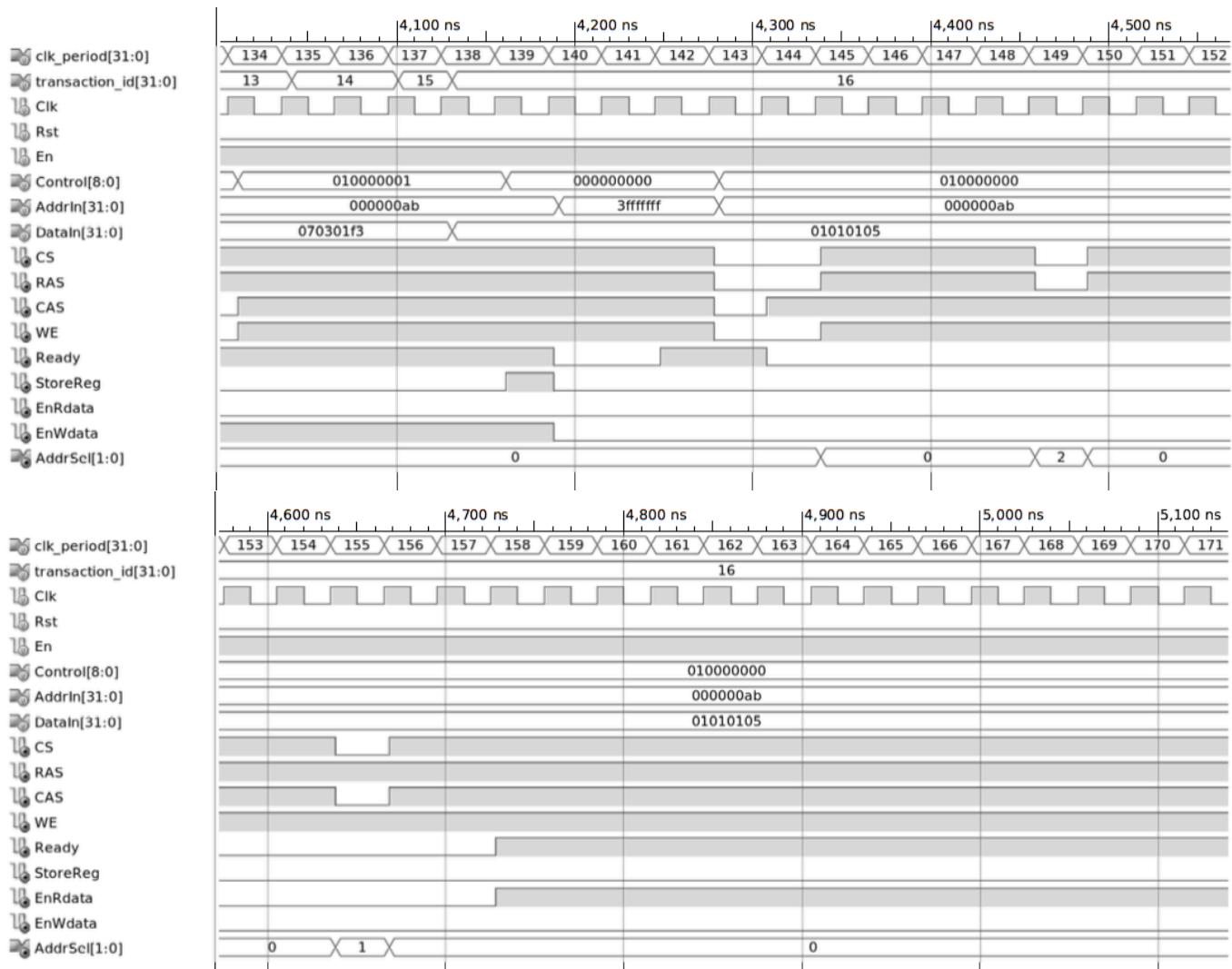
28	3	0	1	10000000	0x000000ab	0x0503043a	1	1	1	1	0	0	0	0	0
29	3	0	1	10000000	0x000000ab	0x0503043a	0	0	1	1	0	0	0	0	2
30	3	0	1	10000000	0x000000ab	0x0503043a	1	1	1	1	0	0	0	0	0
31	3	0	1	10000000	0x000000ab	0x0503043a	1	1	1	1	0	0	0	0	0
32	3	0	1	10000000	0x000000ab	0x0503043a	1	1	1	1	0	0	0	0	0
33	3	0	1	10000000	0x000000ab	0x0503043a	1	1	1	1	0	0	0	0	0
34	3	0	1	10000000	0x000000ab	0x0503043a	1	1	1	1	0	0	0	0	0
35	3	0	1	10000000	0x000000ab	0x0503043a	0	1	0	1	0	0	0	0	1
36	3	0	1	10000000	0x000000ab	0x0503043a	1	1	1	1	0	0	0	0	0
37	3	0	1	10000000	0x000000ab	0x0503043a	1	1	1	1	0	0	0	0	0
38	3	0	1	10000000	0x000000ab	0x0503043a	1	1	1	1	1	0	1	0	0
39	3	0	1	10000000	0x000000ab	0x0503043a	1	1	1	1	1	0	1	0	0
40	3	0	1	110000000	0x000000ab	0x0503043a	1	1	0	0	1	0	1	0	0
41	3	0	1	110000000	0x000000ab	0x0503043a	1	1	0	0	1	0	1	0	0
42	3	0	1	10000000	0x000000ab	0x0503043a	1	1	1	1	1	0	1	0	0
43	3	0	1	10000000	0x000000ab	0x0503043a	1	1	1	1	1	0	1	0	0
44	4	0	1	1	0x3fffffff	0x0503043a	1	1	1	1	0	0	0	0	0
45	4	0	1	1	0x3fffffff	0x0503043a	1	1	1	1	0	0	0	0	0
46	4	0	1	1	0x3fffffff	0x0503043a	1	1	1	1	1	0	0	0	0
47	5	0	1	1	0x000000ab	0x05050751	0	0	0	0	0	0	0	0	0
48	5	0	1	1	0x000000ab	0x05050751	0	0	1	0	0	1	0	0	0
49	5	0	1	1	0x000000ab	0x05050751	1	1	1	1	0	0	0	0	0
50	5	0	1	1	0x000000ab	0x05050751	1	1	1	1	0	0	0	0	0
51	5	0	1	1	0x000000ab	0x05050751	1	1	1	1	0	0	0	0	0
52	5	0	1	1	0x000000ab	0x05050751	1	1	1	1	0	0	0	0	0
53	5	0	1	1	0x000000ab	0x05050751	0	0	1	1	0	0	0	0	2
54	5	0	1	1	0x000000ab	0x05050751	1	1	1	1	0	0	0	0	0
55	5	0	1	1	0x000000ab	0x05050751	1	1	1	1	0	0	0	0	0
56	5	0	1	1	0x000000ab	0x05050751	1	1	1	1	0	0	0	0	0
57	5	0	1	1	0x000000ab	0x05050751	1	1	1	1	0	0	0	0	0
58	5	0	1	1	0x000000ab	0x05050751	1	1	1	1	1	0	0	0	0
59	5	0	1	10000001	0x000000ab	0x05050751	0	1	0	0	1	0	0	1	1
60	5	0	1	110000001	0x000000ab	0x05050751	1	1	0	0	1	0	0	1	0
61	5	0	1	110000001	0x000000ab	0x05050751	1	1	0	0	1	0	0	1	0
62	6	0	1	1	0x000000ab	0x05050751	1	1	1	1	1	1	0	1	0
63	6	0	1	10000001	0x000000ab	0x05050751	1	1	1	1	0	0	0	0	0
64	6	0	1	10000001	0x000000ab	0x05050751	1	1	1	1	0	0	0	0	0
65	6	0	1	10000001	0x000000ab	0x05050751	1	1	1	1	0	0	0	0	0
66	6	0	1	10000001	0x000000ab	0x05050751	0	0	1	0	0	0	0	0	0
67	6	0	1	10000001	0x000000ab	0x05050751	1	1	1	1	0	0	0	0	0
68	6	0	1	10000001	0x000000ab	0x05050751	1	1	1	1	0	0	0	0	0
69	6	0	1	10000001	0x000000ab	0x05050751	1	1	1	1	0	0	0	0	0
70	c	n	1	10000001	0x000000ab	0x05050751	1	1	1	1	n	n	n	n	n

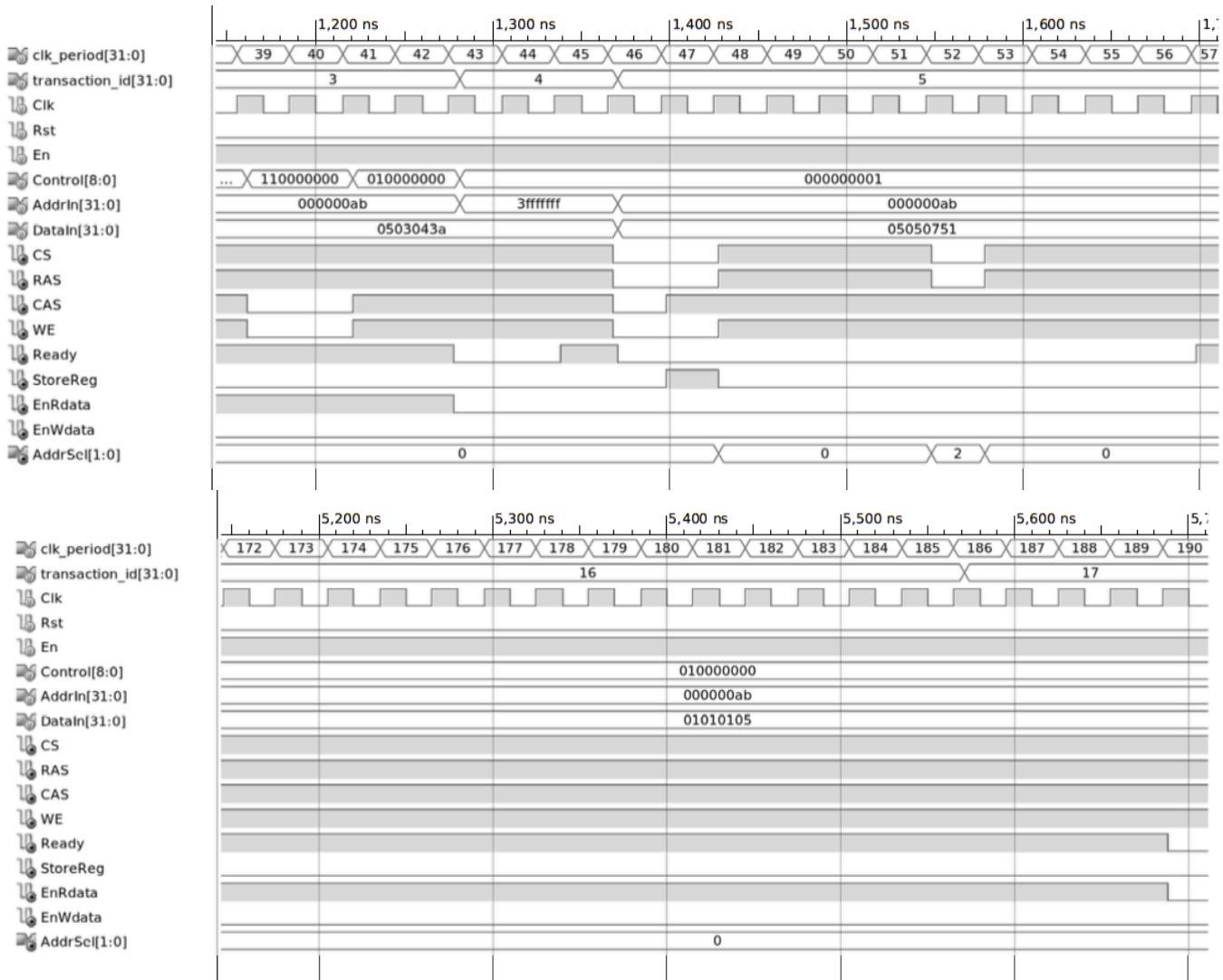
71	6	0	1	10000001	0x000000ab	0x05050751	0	0	1	1	0	0	0	0	2
72	6	0	1	10000001	0x000000ab	0x05050751	1	1	1	1	0	0	0	0	0
73	6	0	1	10000001	0x000000ab	0x05050751	1	1	1	1	0	0	0	0	0
74	6	0	1	10000001	0x000000ab	0x05050751	1	1	1	1	0	0	0	0	0
75	6	0	1	10000001	0x000000ab	0x05050751	1	1	1	1	0	0	0	0	0
76	6	0	1	10000001	0x000000ab	0x05050751	1	1	1	1	0	0	0	0	0
77	6	0	1	10000001	0x000000ab	0x05050751	0	1	0	0	1	0	0	1	1
78	7	0	1	0	0x000000ab	0x05050751	1	1	1	1	1	1	0	1	0
79	7	0	1	10000000	0x000000ab	0x05050751	1	1	1	1	0	0	0	0	0
80	7	0	1	10000000	0x000000ab	0x05050751	1	1	1	1	0	0	0	0	0
81	7	0	1	10000000	0x000000ab	0x05050751	1	1	1	1	0	0	0	0	0
82	7	0	1	10000000	0x000000ab	0x05050751	0	0	1	0	0	0	0	0	0
83	7	0	1	10000000	0x000000ab	0x05050751	1	1	1	1	0	0	0	0	0
84	7	0	1	10000000	0x000000ab	0x05050751	1	1	1	1	0	0	0	0	0
85	7	0	1	10000000	0x000000ab	0x05050751	1	1	1	1	0	0	0	0	0
86	7	0	1	10000000	0x000000ab	0x05050751	1	1	1	1	0	0	0	0	0
87	7	0	1	10000000	0x000000ab	0x05050751	0	0	1	1	0	0	0	0	2
88	7	0	1	10000000	0x000000ab	0x05050751	1	1	1	1	0	0	0	0	0
89	7	0	1	10000000	0x000000ab	0x05050751	1	1	1	1	0	0	0	0	0
90	7	0	1	10000000	0x000000ab	0x05050751	1	1	1	1	0	0	0	0	0
91	7	0	1	10000000	0x000000ab	0x05050751	1	1	1	1	0	0	0	0	0
92	7	0	1	10000000	0x000000ab	0x05050751	1	1	1	1	0	0	0	0	0
93	7	0	1	10000000	0x000000ab	0x05050751	0	1	0	1	0	0	0	0	1
94	7	0	1	10000000	0x000000ab	0x05050751	1	1	1	1	0	0	0	0	0
95	7	0	1	10000000	0x000000ab	0x05050751	1	1	1	1	0	0	0	0	0
96	7	0	1	10000000	0x000000ab	0x05050751	1	1	1	1	1	0	1	0	0
97	8	0	1	1	0x3fffffff	0x05050751	1	1	1	1	1	1	1	0	0
98	9	0	1	0	0x3fffffff	0x09090a1b0	1	1	1	1	0	0	0	0	0
99	9	0	1	0	0x3fffffff	0x09090a1b0	1	1	1	1	0	0	0	0	0
100	9	0	1	0	0x3fffffff	0x09090a1b0	1	1	1	1	1	0	0	0	0
101	10	0	1	10000000	0x000000ab	0x09090a1b0	0	0	0	0	1	0	0	0	0
102	11	0	1	10000001	0x000000ab	0x09090a1b0	0	0	1	0	0	0	0	0	0
103	11	0	1	110000001	0x000000ab	0x09090a1b0	1	1	1	1	0	0	0	0	0
104	11	0	1	10000001	0x000000ab	0x09090a1b0	1	1	1	1	0	0	0	0	0
105	11	0	1	10000001	0x000000ab	0x09090a1b0	1	1	1	1	0	0	0	0	0
106	11	0	1	10000001	0x000000ab	0x09090a1b0	1	1	1	1	0	0	0	0	0
107	11	0	1	10000001	0x000000ab	0x09090a1b0	0	0	1	1	0	0	0	0	2
108	11	0	1	10000001	0x000000ab	0x09090a1b0	1	1	1	1	0	0	0	0	0
109	11	0	1	10000001	0x000000ab	0x09090a1b0	1	1	1	1	0	0	0	0	0
110	11	0	1	10000001	0x000000ab	0x09090a1b0	1	1	1	1	0	0	0	0	0
111	11	0	1	10000001	0x000000ab	0x09090a1b0	1	1	1	1	0	0	0	0	0
112	11	0	1	10000001	0x000000ab	0x09090a1b0	1	1	1	1	0	0	0	0	0
113	11	0	1	10000001	0x000000ab	0x09090a1b0	1	1	1	1	0	0	1	1	1

114	11	0	1	10000001	0x000000ab	0x0090a1b0	1	1	1	1	1	0	0	1	0
115	12	0	1	1	0x3fffffff	0x0090a1b0	1	1	1	1	0	0	0	0	0
116	12	0	1	1	0x3fffffff	0x0090a1b0	1	1	1	1	1	0	0	0	0
117	13	0	1	1	0x000000ab	0x070301f3	0	0	0	0	0	0	0	0	0
118	13	0	1	1	0x000000ab	0x070301f3	0	0	1	0	0	1	0	0	0
119	13	0	1	1	0x000000ab	0x070301f3	1	1	1	1	0	0	0	0	0
120	13	0	1	1	0x000000ab	0x070301f3	1	1	1	1	0	0	0	0	0
121	13	0	1	1	0x000000ab	0x070301f3	1	1	1	1	0	0	0	0	0
122	13	0	1	1	0x000000ab	0x070301f3	1	1	1	1	0	0	0	0	0
123	13	0	1	1	0x000000ab	0x070301f3	0	0	1	1	0	0	0	0	2
124	13	0	1	1	0x000000ab	0x070301f3	1	1	1	1	0	0	0	0	0
125	13	0	1	1	0x000000ab	0x070301f3	1	1	1	1	0	0	0	0	0
126	13	0	1	1	0x000000ab	0x070301f3	1	1	1	1	0	0	0	0	0
127	13	0	1	1	0x000000ab	0x070301f3	1	1	1	1	0	0	0	0	0
128	13	0	1	1	0x000000ab	0x070301f3	1	1	1	1	1	0	0	0	0
129	13	0	1	10000001	0x000000ab	0x070301f3	0	1	0	0	1	0	0	1	1
130	13	0	1	10000001	0x000000ab	0x070301f3	1	1	1	1	1	0	0	1	0
131	13	0	1	110000001	0x000000ab	0x070301f3	1	1	0	0	1	0	0	1	0
132	13	0	1	110000001	0x000000ab	0x070301f3	1	1	0	0	1	0	0	1	0
133	13	0	1	110000001	0x000000ab	0x070301f3	1	1	0	0	1	0	0	1	0
134	13	0	1	110000001	0x000000ab	0x070301f3	1	1	0	0	1	0	0	1	0
135	13	0	1	10000001	0x000000ab	0x070301f3	1	1	1	1	1	0	0	1	0
136	14	0	1	10000001	0x000000ab	0x070301f3	1	1	1	1	1	0	0	1	0
137	14	0	1	10000001	0x000000ab	0x070301f3	1	1	1	1	1	0	0	1	0
138	15	0	1	10000001	0x000000ab	0x070301f3	1	1	1	1	1	0	0	1	0
139	16	0	1	10000001	0x000000ab	0x01010105	1	1	1	1	1	0	0	1	0
140	16	0	1	0	0x000000ab	0x01010105	1	1	1	1	1	1	0	1	0
141	16	0	1	0	0x3fffffff	0x01010105	1	1	1	1	0	0	0	0	0
142	16	0	1	0	0x3fffffff	0x01010105	1	1	1	1	0	0	0	0	0
143	16	0	1	0	0x3fffffff	0x01010105	1	1	1	1	1	0	0	0	0
144	16	0	1	10000000	0x000000ab	0x01010105	0	0	0	0	1	0	0	0	0
145	16	0	1	10000000	0x000000ab	0x01010105	0	0	1	0	0	0	0	0	0
146	16	0	1	10000000	0x000000ab	0x01010105	1	1	1	1	0	0	0	0	0
147	16	0	1	10000000	0x000000ab	0x01010105	1	1	1	1	0	0	0	0	0
148	16	0	1	10000000	0x000000ab	0x01010105	1	1	1	1	0	0	0	0	0
149	16	0	1	10000000	0x000000ab	0x01010105	1	1	1	1	0	0	0	0	0
150	16	0	1	10000000	0x000000ab	0x01010105	0	0	1	1	0	0	0	0	2
151	16	0	1	10000000	0x000000ab	0x01010105	1	1	1	1	0	0	0	0	0
152	16	0	1	10000000	0x000000ab	0x01010105	1	1	1	1	0	0	0	0	0
153	16	0	1	10000000	0x000000ab	0x01010105	1	1	1	1	0	0	0	0	0
154	16	0	1	10000000	0x000000ab	0x01010105	1	1	1	1	0	0	0	0	0
155	16	0	1	10000000	0x000000ab	0x01010105	1	1	1	1	0	0	0	0	0
156	16	0	1	10000000	0x000000ab	0x01010105	1	1	1	1	0	0	0	0	1









12. Top Module with Master, BIU and SDRAM

This is the top module with a Master transacting with the SDRAM slave based on the unidirectional bus protocol. The test frame work, verification tables and waveforms are as shown below.

The transactions carried out as part of the test framework are shown below:

Transaction 0 involves programming the Master's Register File with necessary transaction information. On pulling its program pin low, the Master starts to transact based on its Register File contents.

Transaction 1 involved the master programming the SDRAM with the required time period values, burst length and addressing modes. Here, it programs for tCas = 5, tWait = 4, tPre = 5, tLat = 5, Burst = 4 and Sequential Addressing Mode.

Transaction 2 is a write with Burst 4, data packet size of Words onto Bank 0 of SDRAM.

Transaction 3 is a write with Burst 4, data packet size of Half Words onto Bank 1 of SDRAM.

Transaction 4 is a read with Burst 4, data packet size of Bytes from Bank 0 of SDRAM.

Transaction 5 is a read with Burst 4, data packet size of Words from Bank 1 of SDRAM.

Transaction 6 programs the Burst to 8 and Sequential Addressing Mode.

Transaction 7 is a write with Burst 8, data packet size of Half Words onto Bank 3 of SDRAM.

Transaction 8 is a read with Burst 8, data packet size of Bytes from Bank 3 of SDRAM.

Transaction 9 programs the Burst to 1 and Linear Addressing Mode.

Transaction 10 is a write with Burst 8, data packet size of Half Words onto Bank 0 of SDRAM.

Transaction 11 is a read with Burst 8, data packet size of Word from Bank 0 of SDRAM.

The behaviour of the top module is as expected. The data written into the address after the write transactions are read out correctly, based on the data packet size during each of the read transactions. The Master also goes busy in between transactions. The module handles the situation.

TID	Clk Period	Reset	Address	Control	WData	RData	Busy
0	0	1	0xffffffff	9'bxxxxxxxxx	0xffffffff	0xffffffff	x
0	1	0	0xffffffff	9'bxxxxxxxxx	0xffffffff	0xffffffff	x
0	2	0	0xffffffff	9'bxxxxxxxxx	0xffffffff	0xffffffff	x
1	3	0	0x3fffffff	9'b000000101	0xffffffff	0x00000000	1
1	4	0	0x00004ad0	9'b000010101	0x0504053a	0x00000000	0
1	5	0	0x00004ad0	9'b000010101	0x0504053a	0x00000000	0
1	6	0	0x00004ad0	9'b000010101	0x0504053a	0x00000000	0
1	7	0	0x00004ad0	9'b000010101	0x0504053a	0x00000000	0
1	8	0	0x00004ad0	9'b000010101	0x0504053a	0x00000000	0
1	9	0	0x00004ad0	9'b000010101	0x0504053a	0x00000000	0
1	10	0	0x00004ad0	9'b000010101	0x0504053a	0x00000000	0
1	11	0	0x00004ad0	9'b000010101	0x0504053a	0x00000000	0
1	12	0	0x00004ad0	9'b000010101	0x0504053a	0x00000000	0
1	13	0	0x00004ad0	9'b000010101	0x0504053a	0x00000000	0
1	14	0	0x00004ad0	9'b000010101	0x0504053a	0x00000000	0
1	15	0	0x00004ad0	9'b000010101	0x0504053a	0x00000000	0

2	16	0	0x00004ad0	9'b000010101	0x0504053a	0x00000000	1
2	17	0	0x00004ad4	9'b010010101	0x11223344	0x00000000	1
2	18	0	0x00004ad8	9'b010010101	0x22222222	0x00000000	1
2	19	0	0x00004adc	9'b010010101	0x33333333	0x00000000	1
3	20	0	0x00014ad0	9'b000010011	0x44444444	0x00000000	1
3	21	0	0x00014ad2	9'b010010011	0xaabbaabb	0x00000000	0
3	22	0	0x00014ad2	9'b010010011	0xaabbaabb	0x00000000	0
3	23	0	0x00014ad2	9'b010010011	0xaabbaabb	0x00000000	0
3	24	0	0x00014ad2	9'b010010011	0xaabbaabb	0x00000000	0
3	25	0	0x00014ad2	9'b010010011	0xaabbaabb	0x00000000	0
3	26	0	0x00014ad2	9'b010010011	0xaabbaabb	0x00000000	0
3	27	0	0x00014ad2	9'b010010011	0xaabbaabb	0x00000000	0
3	28	0	0x00014ad2	9'b010010011	0xaabbaabb	0x00000000	0
3	29	0	0x00014ad2	9'b010010011	0xaabbaabb	0x00000000	0
3	30	0	0x00014ad2	9'b010010011	0xaabbaabb	0x00000000	0
3	31	0	0x00014ad2	9'b010010011	0xaabbaabb	0x00000000	0
3	32	0	0x00014ad2	9'b010010011	0xaabbaabb	0x00000000	0
3	33	0	0x00014ad2	9'b010010011	0xaabbaabb	0x00000000	0
3	34	0	0x00014ad2	9'b010010011	0xaabbaabb	0x00000000	0
3	35	0	0x00014ad2	9'b010010011	0xaabbaabb	0x00000000	0
3	36	0	0x00014ad2	9'b010010011	0xaabbaabb	0x00000000	0
3	37	0	0x00014ad2	9'b010010011	0xaabbaabb	0x00000000	1
3	38	0	0x00014ad4	9'b010010011	0xccddccdd	0x00000000	1
3	39	0	0x00014ad6	9'b010010011	0xabababab	0x00000000	1
4	40	0	0x00004ad0	9'b000010000	0xaabbffdd	0x00000000	1
4	41	0	0x00004ad0	9'b110010000	0xaabbffdd	0x00000000	0
4	42	0	0x00004ad0	9'b110010000	0xaabbffdd	0x00000000	0
4	43	0	0x00004ad0	9'b110010000	0xaabbffdd	0x00000000	0
4	44	0	0x00004ad0	9'b110010000	0xaabbffdd	0x00000000	0
4	45	0	0x00004ad0	9'b110010000	0xaabbffdd	0x00000000	0

4	46	0	0x00004ad0	9'b110010000	0xaabbffdd	0x00000000	0
4	47	0	0x00004ad0	9'b110010000	0xaabbffdd	0x00000000	0
4	48	0	0x00004ad0	9'b110010000	0xaabbffdd	0x00000000	0
4	49	0	0x00004ad0	9'b110010000	0xaabbffdd	0x00000000	0
4	50	0	0x00004ad0	9'b110010000	0xaabbffdd	0x00000000	0
4	51	0	0x00004ad0	9'b110010000	0xaabbffdd	0x00000000	0
4	52	0	0x00004ad0	9'b110010000	0xaabbffdd	0x00000000	0
4	53	0	0x00004ad0	9'b110010000	0xaabbffdd	0x00000000	0
4	54	0	0x00004ad0	9'b110010000	0xaabbffdd	0x00000000	0
4	55	0	0x00004ad0	9'b110010000	0xaabbffdd	0x00000000	0
4	56	0	0x00004ad0	9'b110010000	0xaabbffdd	0x00000000	0
4	57	0	0x00004ad0	9'b110010000	0xaabbffdd	0x00000000	0
4	58	0	0x00004ad0	9'b110010000	0xaabbffdd	0x00000000	0
4	59	0	0x00004ad0	9'b110010000	0xaabbffdd	0x00000000	0
4	60	0	0x00004ad0	9'b110010000	0xaabbffdd	0x00000044	1
4	61	0	0x00004ad1	9'b010010000	0xaabbffdd	0x00000044	1
4	62	0	0x00004ad2	9'b010010000	0xaabbffdd	0x00000033	1
4	63	0	0x00004ad3	9'b010010000	0xaabbffdd	0x00000022	1
5	64	0	0x00014ad0	9'b000010100	0xaabbffdd	0x00000011	1
5	65	0	0x00014ad4	9'b010010100	0xaabbffdd	0x00000011	0
5	66	0	0x00014ad4	9'b010010100	0xaabbffdd	0x00000011	0
5	67	0	0x00014ad4	9'b010010100	0xaabbffdd	0x00000011	0
5	68	0	0x00014ad4	9'b010010100	0xaabbffdd	0x00000011	0
5	69	0	0x00014ad4	9'b010010100	0xaabbffdd	0x00000011	0
5	70	0	0x00014ad4	9'b010010100	0xaabbffdd	0x00000011	0
5	71	0	0x00014ad4	9'b010010100	0xaabbffdd	0x00000011	0
5	72	0	0x00014ad4	9'b010010100	0xaabbffdd	0x00000011	0
5	73	0	0x00014ad4	9'b010010100	0xaabbffdd	0x00000011	0
5	74	0	0x00014ad4	9'b010010100	0xaabbffdd	0x00000011	0
5	75	0	0x00014ad4	9'b010010100	0xaabbffdd	0x00000011	0

5	76	0	0x00014ad4	9'b010010100	0xaabbffdd	0x00000011	0
5	77	0	0x00014ad4	9'b010010100	0xaabbffdd	0x00000011	0
5	78	0	0x00014ad4	9'b010010100	0xaabbffdd	0x00000011	0
5	79	0	0x00014ad4	9'b010010100	0xaabbffdd	0x00000011	0
5	80	0	0x00014ad4	9'b010010100	0xaabbffdd	0x00000011	0
5	81	0	0x00014ad4	9'b010010100	0xaabbffdd	0x00000011	0
5	82	0	0x00014ad4	9'b010010100	0xaabbffdd	0x00000011	0
5	83	0	0x00014ad4	9'b010010100	0xaabbffdd	0x00000011	0
5	84	0	0x00014ad4	9'b010010100	0xaabbffdd	0xccddaaabb	1
5	85	0	0x00014ad8	9'b010010100	0xaabbffdd	0xffddabab	1
5	86	0	0x00014adc	9'b010010100	0xaabbffdd	0xffffffff	1
6	87	0	0x00004ad0	9'b000010010	0xaabbffdd	0xffffffff	1
6	88	0	0x00004ad2	9'b010010010	0xaabbffdd	0xffffffff	0
6	89	0	0x00004ad2	9'b010010010	0xaabbffdd	0xffffffff	0
6	90	0	0x00004ad2	9'b010010010	0xaabbffdd	0xffffffff	0
6	91	0	0x00004ad2	9'b010010010	0xaabbffdd	0xffffffff	0
6	92	0	0x00004ad2	9'b010010010	0xaabbffdd	0xffffffff	0
6	93	0	0x00004ad2	9'b010010010	0xaabbffdd	0xffffffff	0
6	94	0	0x00004ad2	9'b010010010	0xaabbffdd	0xffffffff	0
6	95	0	0x00004ad2	9'b010010010	0xaabbffdd	0xffffffff	0
6	96	0	0x00004ad2	9'b010010010	0xaabbffdd	0xffffffff	0
6	97	0	0x00004ad2	9'b010010010	0xaabbffdd	0xffffffff	0
6	98	0	0x00004ad2	9'b010010010	0xaabbffdd	0xffffffff	0
6	99	0	0x00004ad2	9'b010010010	0xaabbffdd	0xffffffff	0
6	100	0	0x00004ad2	9'b010010010	0xaabbffdd	0xffffffff	0
6	101	0	0x00004ad2	9'b010010010	0xaabbffdd	0xffffffff	0
6	102	0	0x00004ad2	9'b010010010	0xaabbffdd	0xffffffff	0
6	103	0	0x00004ad2	9'b010010010	0xaabbffdd	0xffffffff	0
6	104	0	0x00004ad2	9'b010010010	0xaabbffdd	0xffffffff	0
6	105	0	0x00004ad2	9'b010010010	0xaabbffdd	0xffffffff	0

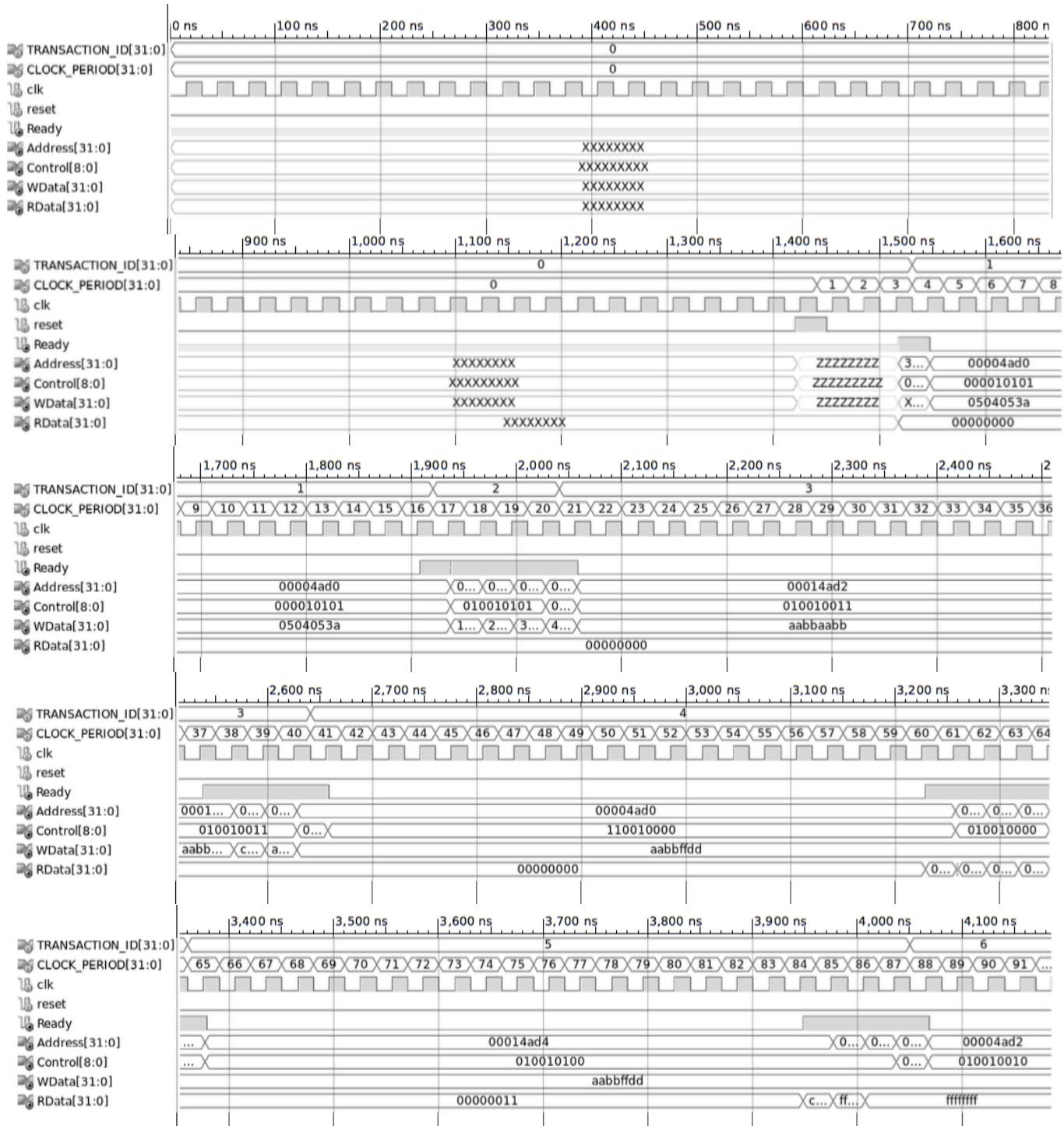
6	106	0	0x00004ad2	9'b010010010	0xaabbffdd	0xffffffff	0
6	107	0	0x00004ad2	9'b010010010	0xaabbffdd	0x00003344	1
6	108	0	0x00004ad4	9'b010010010	0xaabbffdd	0x00001122	1
6	109	0	0x00004ad6	9'b010010010	0xaabbffdd	0x00002222	1
7	110	0	0x3fffffff	9'b0000000101	0xaabbffdd	0x00002222	1
7	111	0	0x3fffffff	9'b0000000101	0x05040533	0x00002222	0
7	112	0	0x3fffffff	9'b0000000101	0x05040533	0x00002222	0
7	113	0	0x3fffffff	9'b0000000101	0x05040533	0x00002222	0
8	114	0	0x3fffffff	9'b0000000101	0x05040533	0x00002222	1
8	115	0	0x00024ad0	9'b000011011	0x05040533	0x00002222	0
8	116	0	0x00024ad0	9'b000011011	0x05040533	0x00002222	0
8	117	0	0x00024ad0	9'b000011011	0x05040533	0x00002222	0
8	118	0	0x00024ad0	9'b000011011	0x05040533	0x00002222	0
8	119	0	0x00024ad0	9'b000011011	0x05040533	0x00002222	0
8	120	0	0x00024ad0	9'b000011011	0x05040533	0x00002222	0
8	121	0	0x00024ad0	9'b000011011	0x05040533	0x00002222	0
8	122	0	0x00024ad0	9'b000011011	0x05040533	0x00002222	0
8	123	0	0x00024ad0	9'b000011011	0x05040533	0x00002222	0
8	124	0	0x00024ad0	9'b000011011	0x05040533	0x00002222	0
8	125	0	0x00024ad0	9'b000011011	0x05040533	0x00002222	0
8	126	0	0x00024ad0	9'b000011011	0x05040533	0x00002222	0
9	127	0	0x00024ad0	9'b000011011	0x05040533	0x00002222	1
9	128	0	0x00024ad2	9'b010011011	0x11112222	0x00002222	1
9	129	0	0x00024ad4	9'b010011011	0x33334444	0x00002222	1
9	130	0	0x00024ad6	9'b010011011	0x55556666	0x00002222	1
9	131	0	0x00024ad8	9'b010011011	0x77778888	0x00002222	1
9	132	0	0x00024ada	9'b010011011	0x11112222	0x00002222	1
9	133	0	0x00024adc	9'b010011011	0x33334444	0x00002222	1
9	134	0	0x00024ade	9'b010011011	0x55556666	0x00002222	1
10	135	0	0x00024ad0	9'b000011000	0x77778888	0x00002222	1

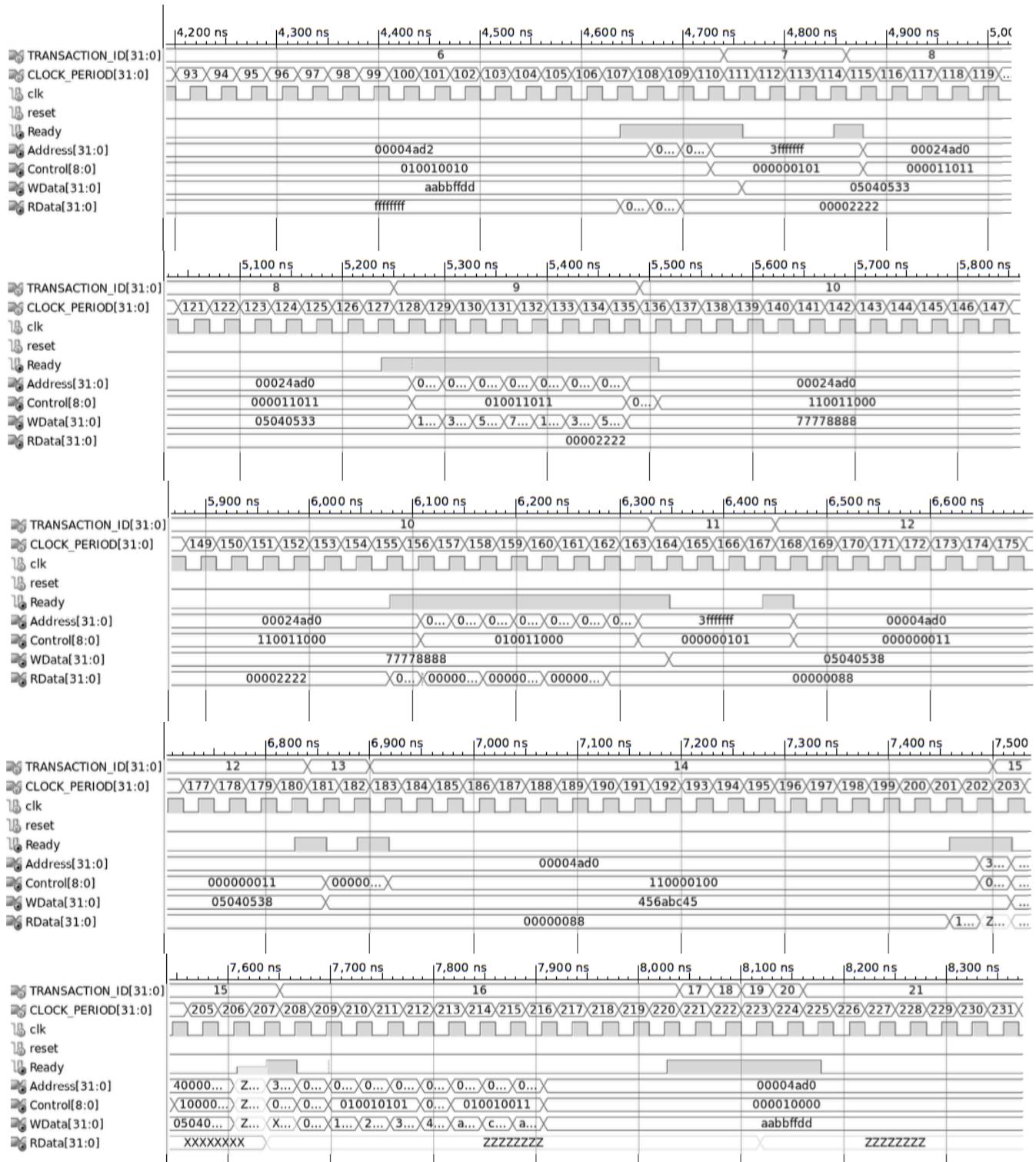
10	136	0	0x00024ad0	9'b110011000	0x77778888	0x00002222	0
10	137	0	0x00024ad0	9'b110011000	0x77778888	0x00002222	0
10	138	0	0x00024ad0	9'b110011000	0x77778888	0x00002222	0
10	139	0	0x00024ad0	9'b110011000	0x77778888	0x00002222	0
10	140	0	0x00024ad0	9'b110011000	0x77778888	0x00002222	0
10	141	0	0x00024ad0	9'b110011000	0x77778888	0x00002222	0
10	142	0	0x00024ad0	9'b110011000	0x77778888	0x00002222	0
10	143	0	0x00024ad0	9'b110011000	0x77778888	0x00002222	0
10	144	0	0x00024ad0	9'b110011000	0x77778888	0x00002222	0
10	145	0	0x00024ad0	9'b110011000	0x77778888	0x00002222	0
10	146	0	0x00024ad0	9'b110011000	0x77778888	0x00002222	0
10	147	0	0x00024ad0	9'b110011000	0x77778888	0x00002222	0
10	148	0	0x00024ad0	9'b110011000	0x77778888	0x00002222	0
10	149	0	0x00024ad0	9'b110011000	0x77778888	0x00002222	0
10	150	0	0x00024ad0	9'b110011000	0x77778888	0x00002222	0
10	151	0	0x00024ad0	9'b110011000	0x77778888	0x00002222	0
10	152	0	0x00024ad0	9'b110011000	0x77778888	0x00002222	0
10	153	0	0x00024ad0	9'b110011000	0x77778888	0x00002222	0
10	154	0	0x00024ad0	9'b110011000	0x77778888	0x00002222	0
10	155	0	0x00024ad0	9'b110011000	0x77778888	0x00000022	1
10	156	0	0x00024ad1	9'b010011000	0x77778888	0x00000022	1
10	157	0	0x00024ad2	9'b010011000	0x77778888	0x00000022	1
10	158	0	0x00024ad3	9'b010011000	0x77778888	0x00000044	1
10	159	0	0x00024ad4	9'b010011000	0x77778888	0x00000044	1
10	160	0	0x00024ad5	9'b010011000	0x77778888	0x00000066	1
10	161	0	0x00024ad6	9'b010011000	0x77778888	0x00000066	1
10	162	0	0x00024ad7	9'b010011000	0x77778888	0x00000088	1
11	163	0	0x3ffffffff	9'b0000000101	0x77778888	0x00000088	1
11	164	0	0x3ffffffff	9'b0000000101	0x05040538	0x00000088	0
11	165	0	0x3ffffffff	9'b0000000101	0x05040538	0x00000088	0

11	166	0	0x3fffffff	9'b000000101	0x05040538	0x00000088	0
12	167	0	0x3fffffff	9'b000000101	0x05040538	0x00000088	1
12	168	0	0x00004ad0	9'b000000011	0x05040538	0x00000088	0
12	169	0	0x00004ad0	9'b000000011	0x05040538	0x00000088	0
12	170	0	0x00004ad0	9'b000000011	0x05040538	0x00000088	0
12	171	0	0x00004ad0	9'b000000011	0x05040538	0x00000088	0
12	172	0	0x00004ad0	9'b000000011	0x05040538	0x00000088	0
12	173	0	0x00004ad0	9'b000000011	0x05040538	0x00000088	0
12	174	0	0x00004ad0	9'b000000011	0x05040538	0x00000088	0
12	175	0	0x00004ad0	9'b000000011	0x05040538	0x00000088	0
12	176	0	0x00004ad0	9'b000000011	0x05040538	0x00000088	0
12	177	0	0x00004ad0	9'b000000011	0x05040538	0x00000088	0
12	178	0	0x00004ad0	9'b000000011	0x05040538	0x00000088	0
12	179	0	0x00004ad0	9'b000000011	0x05040538	0x00000088	0
13	180	0	0x00004ad0	9'b000000011	0x05040538	0x00000088	1
13	181	0	0x00004ad0	9'b000000100	0x456abc45	0x00000088	0
14	182	0	0x00004ad0	9'b000000100	0x456abc45	0x00000088	1
14	183	0	0x00004ad0	9'b110000100	0x456abc45	0x00000088	0
14	184	0	0x00004ad0	9'b110000100	0x456abc45	0x00000088	0
14	185	0	0x00004ad0	9'b110000100	0x456abc45	0x00000088	0
14	186	0	0x00004ad0	9'b110000100	0x456abc45	0x00000088	0
14	187	0	0x00004ad0	9'b110000100	0x456abc45	0x00000088	0
14	188	0	0x00004ad0	9'b110000100	0x456abc45	0x00000088	0
14	189	0	0x00004ad0	9'b110000100	0x456abc45	0x00000088	0
14	190	0	0x00004ad0	9'b110000100	0x456abc45	0x00000088	0
14	191	0	0x00004ad0	9'b110000100	0x456abc45	0x00000088	0
14	192	0	0x00004ad0	9'b110000100	0x456abc45	0x00000088	0
14	193	0	0x00004ad0	9'b110000100	0x456abc45	0x00000088	0
14	194	0	0x00004ad0	9'b110000100	0x456abc45	0x00000088	0
14	195	0	0x00004ad0	9'b110000100	0x456abc45	0x00000088	0

14	196	0	0x00004ad0	9'b110000100	0x456abc45	0x00000088	0
14	197	0	0x00004ad0	9'b110000100	0x456abc45	0x00000088	0
14	198	0	0x00004ad0	9'b110000100	0x456abc45	0x00000088	0
14	199	0	0x00004ad0	9'b110000100	0x456abc45	0x00000088	0
14	200	0	0x00004ad0	9'b110000100	0x456abc45	0x00000088	0
14	201	0	0x00004ad0	9'b110000100	0x456abc45	0x1122bc45	1
15	202	0	0x3fffffff	9'b000001101	0x456abc45	0xxxxxxxxx	1
15	203	0	0x40000003	9'b010001101	0x05040538	0xxxxxxxxx	0
15	204	0	0x40000003	9'b100001101	0x05040538	0xxxxxxxxx	0
15	205	0	0x40000003	9'b100001101	0x05040538	0xxxxxxxxx	0
15	206	0	0xxxxxxxxx	9'bxxxxxxxxx	0xxxxxxxxx	0xxxxxxxxx	x
16	207	0	0x3fffffff	9'b000000101	0xxxxxxxxx	0xxxxxxxxx	1
16	208	0	0x00004ad0	9'b000010101	0x0504053a	0xxxxxxxxx	0
16	209	0	0x00004ad4	9'b010010101	0x11223344	0xxxxxxxxx	0
16	210	0	0x00004ad8	9'b010010101	0x22222222	0xxxxxxxxx	0
16	211	0	0x00004adc	9'b010010101	0x33333333	0xxxxxxxxx	0
16	212	0	0x00014ad0	9'b000010011	0x44444444	0xxxxxxxxx	0
16	213	0	0x00014ad2	9'b010010011	0xaabbaabb	0xxxxxxxxx	0
16	214	0	0x00014ad4	9'b010010011	0xccddccdd	0xxxxxxxxx	0
16	215	0	0x00014ad6	9'b010010011	0xabababab	0xxxxxxxxx	0
16	216	0	0x00004ad0	9'b000010000	0xaabbffdd	0xxxxxxxxx	0
16	217	0	0x00004ad0	9'b000010000	0xaabbffdd	0xxxxxxxxx	0
16	218	0	0x00004ad0	9'b000010000	0xaabbffdd	0xxxxxxxxx	0
16	219	0	0x00004ad0	9'b000010000	0xaabbffdd	0xxxxxxxxx	0
17	220	0	0x00004ad0	9'b000010000	0xaabbffdd	0xxxxxxxxx	1
18	221	0	0x00004ad0	9'b000010000	0xaabbffdd	0xxxxxxxxx	1
19	222	0	0x00004ad0	9'b000010000	0xaabbffdd	0xxxxxxxxx	1
20	223	0	0x00004ad0	9'b000010000	0xaabbffdd	0xxxxxxxxx	1
21	224	0	0x00004ad0	9'b000010000	0xaabbffdd	0xxxxxxxxx	1
21	225	0	0x00004ad0	9'b000010000	0xaabbffdd	0xxxxxxxxx	0

21	226	0	0x00004ad0	9'b000010000	0xaabbffdd	0xxxxxxxx	0
21	227	0	0x00004ad0	9'b000010000	0xaabbffdd	0xxxxxxxx	0
21	228	0	0x00004ad0	9'b000010000	0xaabbffdd	0xxxxxxxx	0
21	229	0	0x00004ad0	9'b000010000	0xaabbffdd	0xxxxxxxx	0
21	230	0	0x00004ad0	9'b000010000	0xaabbffdd	0xxxxxxxx	0
21	231	0	0x00004ad0	9'b000010000	0xaabbffdd	0xxxxxxxx	0





VI. CONCLUSION

All modules for the SDRAM and Bus Interface have been unit tested. The outputs are verified and are as expected from the design.

VII. REFERENCES

- [1] Dr. Ahmet Bindal, "Memory Circuits and Systems" in *Fundamentals of Computer Architecture and Design*, Maple Press Student ed. San Jose
- [2] Micron SDR SDRAM 512Mb: x8 Datasheet