# 8M x 8 HyperRAM™

SEPTEMBER 2018

### Overview

The IS66/67WVH8M8ALL/BLL are integrated memory device containing 64Mbit Pseudo Static Random Access Memory using a self-refresh DRAM array organized as 8M words by 8 bits. The device supports a HyperBus interface, Very Low Signal Count (Address, Command and data through 8 DQ pins), Hidden Refresh Operation, and Automotive Temperature Operation, designed specially for Mobile and Automotive applications.

## Distinctive Characteristics

### HyperBus™ Low Signal Count Interface

- 3.0V I/O, 11 bus signals
  - Single ended clock (CK)
- 1.8V I/O, 12 bus signals
  - Differential clock (CK, CK#)
- Chip Select (CS#)
- 8-bit data bus (DQ[7:0])
- Read-Write Data Strobe (RWDS)
  - Bidirectional Data Strobe / Mask
  - Output at the start of all transactions to indicate refresh latency
  - Output during read transactions as Read Data Strobe
  - Input during write transactions as Write Data Mask
- RWDS DCARS Timing
  - During read transactions RWDS is offset by a second clock, phase shifted from CK
  - The Phase Shifted Clock is used to move the RWDS transition edge within the read data eye

### High Performance

- Up to 333MB/s
- Double-Data Rate (DDR) - two data transfers per clock
- 166-MHz clock rate (333 MB/s) at 1.8V $V_{CC}$
- 100-MHz clock rate (200 MB/s) at 3.0V $V_{CC}$
- Sequential burst transactions
- Configurable Burst Characteristics
  - Wrapped burst lengths:
    - 16 bytes (8 clocks)
    - 32 bytes (16 clocks)
    - 64 bytes (32 clocks)
    - 128 bytes (64 clocks)
  - Linear burst
  - Hybrid option - one wrapped burst followed by linear burst
  - Wrapped or linear burst type selected in each transaction
  - Configurable output drive strength
- Package
  - 24-ball FBGA

## Performance Summary

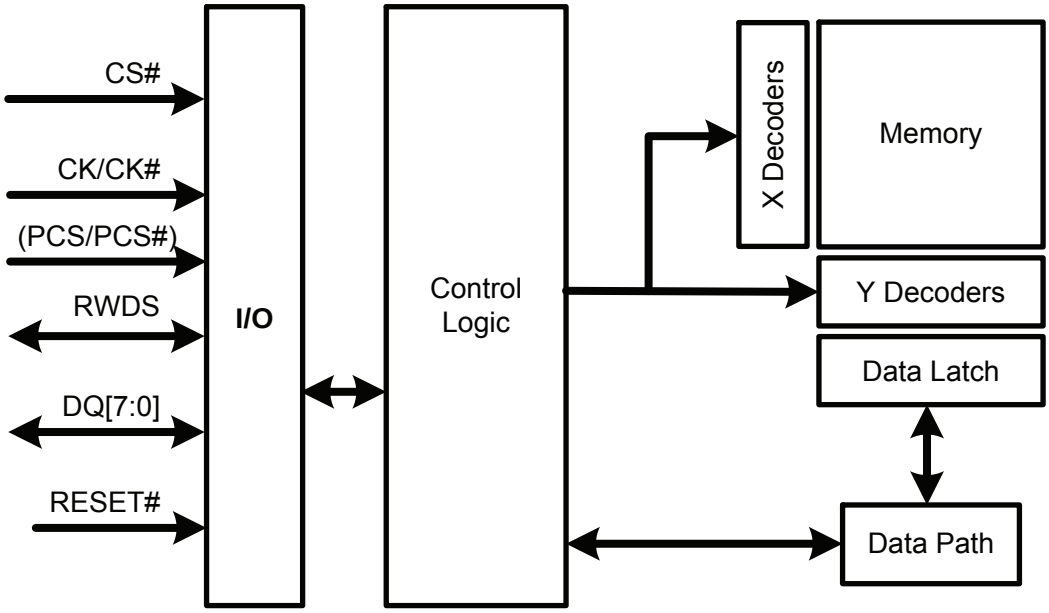
| Read Transaction Timings | |
|---|---|
| Maximum Clock Rate at 1.8V $V_{CC}/V_{CC}Q$ | 166 MHz |
| Maximum Clock Rate at 3.0V $V_{CC}/V_{CC}Q$ | 100 MHz |
| Maximum Access Time, ($t_{ACC}$ at 166 MHz) | 36 ns |
| Maximum CS# Access Time to first word at 166 MHz (excluding refresh latency) | 56 ns |

| Maximum Current Consumption | |
|---|---|
| Burst Read or Write (linear burst at 166 MHz, 1.8V) | 60 mA |
| Power On Reset | 50 mA |
| Standby (CS# = High, 3V, 105 °C) | 300 µA |
| Deep Power Down (CS# = High, 3V, 105 °C) | 40 µA |
| Standby (CS# = High, 1.8V, 105 °C) | 300 µA |
| Deep Power Down (CS# = High, 1.8V, 105 °C) | 20 µA |

Copyright © 2016 Integrated Silicon Solution, Inc. All rights reserved. ISSI reserves the right to make changes to this specification and its products at any time without notice. ISSI assumes no liability arising out of the application or use of any information, products or services described herein. Customers are advised to obtain the latest version of this device specification before relying on any published information and before placing orders for products.

Integrated Silicon Solution, Inc. does not recommend the use of any of its products in life support applications where the failure or malfunction of the product can reasonably be expected to cause failure of the life support system or to significantly affect its safety or effectiveness. Products are not authorized for use in such applications unless Integrated Silicon Solution, Inc. receives written assurance to its satisfaction, that:
a.) the risk of injury or damage has been minimized;
b.) the user assume all such risks; and
c.) potential liability of Integrated Silicon Solution, Inc is adequately protected under the circumstances

## Logic Block Diagram



# Contents

**1. General Description** ..... 4
**2. HyperRAM Product Overview** ..... 7
**3. HyperRAM Signal Descriptions** ..... 8
3.1 Input/Output Summary ..... 8
3.2 Command/Address Bit Assignments ..... 9
3.3 Read Transactions ..... 13
3.4 Write Transactions with Initial Latency (Memory Core Write) ..... 14
3.5 Write Transactions without Initial Latency (Register Write) ..... 16
**4. Memory Space** ..... 17
**5. Register Space** ..... 17
5.1 Device Identification Registers ..... 17
5.2 Register Space Access ..... 18

**HyperRAM Device Hardware Interface**

**6. Interface States** ..... 25
6.1 Power Conservation Modes ..... 25
**7. HyperRAM Connection Descriptions** ..... 27
7.1 Other Connectors Summary ..... 27
7.2 HyperRAM Block Diagram ..... 28
**8. Interface States** ..... 29
**9. Electrical Specifications** ..... 30
9.1 Absolute Maximum Ratings ..... 30
9.2 Latchup Characteristics ..... 31
9.3 Operating Ranges ..... 31
9.4 DC Characteristics ..... 32
9.5 Power-Up Initialization ..... 33
9.6 Power Down ..... 35
9.7 Hardware Reset ..... 36
**10. Timing Specifications** ..... 37
10.1 Key to Switching Waveforms ..... 37
10.2 AC Test Conditions ..... 37
10.3 AC Characteristics ..... 38
**11. Physical Interface** ..... 41
11.1 FBGA 24-Ball 5 x 5 Array Footprint ..... 41
**12. DDR Center Aligned Read Strobe Functionality** ..... 42
12.1 HyperRAM Products with DCARS Signal Descriptions ..... 42
12.2 HyperRAM Products with DCARS — FBGA 24-ball, 5x5 Array Footprint ..... 43
12.3 HyperRAM Memory with DCARS Timing ..... 43
**13. Ordering Rule Information** ..... 44
13.1 Ordering Part Number ..... 45
**14.** PACKAGE INFORMATION ..... 47

# 1. General Description

The ISSI 64-Mbit HyperRAM™ device is a high-speed CMOS, self-refresh Dynamic RAM (DRAM), with a HyperBus interface.

The Random Access Memory (RAM) array uses dynamic cells that require periodic refresh. Refresh control logic within the device manages the refresh operations on the RAM array when the memory is not being actively read or written by the HyperBus interface master (host). Since the host is not required to manage any refresh operations, the DRAM array appears to the host as though the memory uses static cells that retain data without refresh. Hence, the memory can also be described as Pseudo Static RAM (PSRAM).

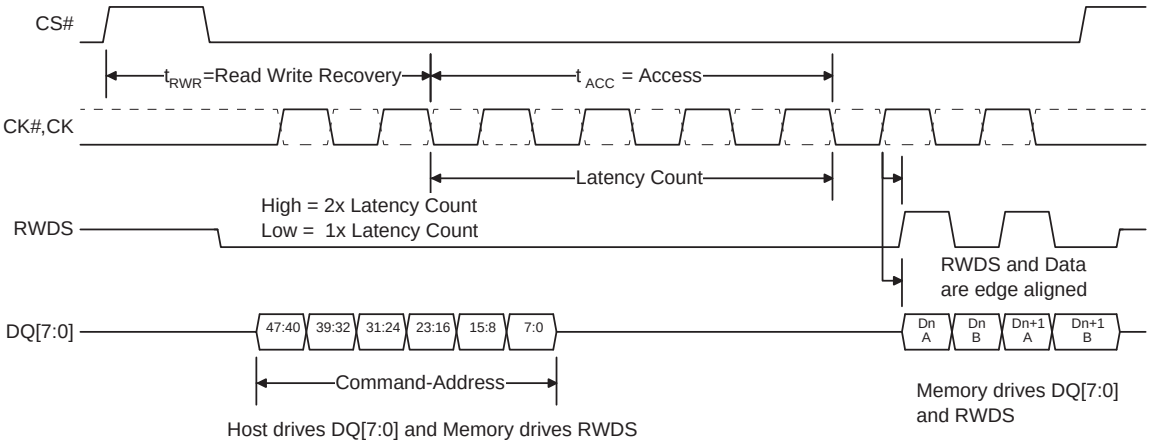
Because the DRAM cells cannot be refreshed during a read or write transaction, there is a requirement that the host not perform read or write burst transfers that are long enough to block the necessary internal logic refresh operations when they are needed. The host is required to limit the duration of transactions and allow additional initial access latency, at the beginning of a new transaction, if the memory indicates a refresh operation is needed.

HyperBus is a low signal count, Double Data Rate (DDR) interface, that achieves high speed read and write throughput. The DDR protocol transfers two data bytes per clock cycle on the DQ input/output signals. A read or write transaction on HyperBus consists of a series of 16-bit wide, one clock cycle data transfers at the internal HyperRAM core with two corresponding 8-bit wide, one-half-clock-cycle data transfers on the DQ signals. All inputs and outputs are LV-CMOS compatible. Ordering Part Number (OPN) device versions are available for core ($V_{CC}$) and IO buffer ($V_{CC}Q$) supplies of either 1.8V or 3.0V (nominal).

Command, address, and data information is transferred over the eight HyperBus DQ[7:0] signals. The clock is used for information capture by a HyperBus slave device when receiving command, address, or data on the DQ signals. Command or Address values are center aligned with clock transitions.

Every transaction begins with the assertion of CS# and Command-Address (CA) signals, followed by the start of clock transitions to transfer six CA bytes, followed by initial access latency and either read or write data transfers, until CS# is deasserted.

**Figure 1.1** Read Transaction, Single Initial Latency Count



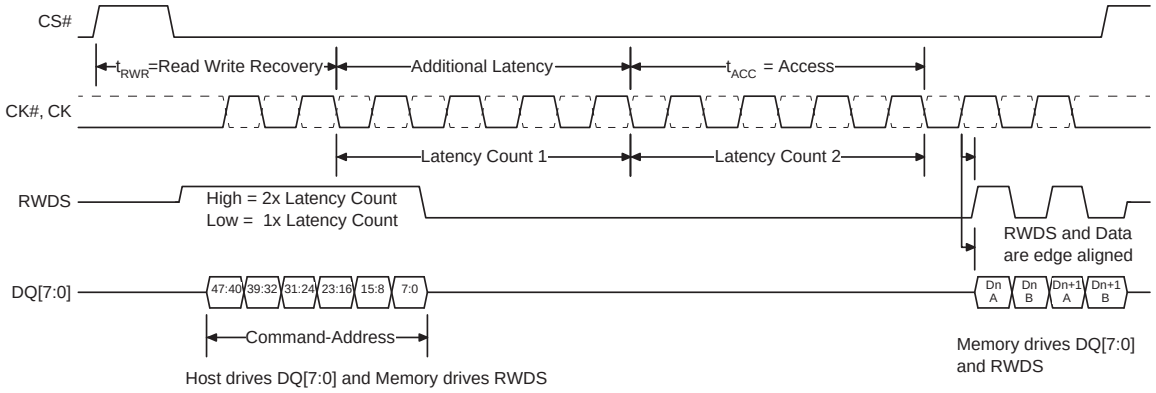

The Read/Write Data Strobe (RWDS) is a bidirectional signal that indicates:

- when data will start to transfer from a HyperRAM device to the master device in read transactions (initial read latency)
- when data is being transferred from a HyperRAM device to the master device during read transactions (as a source synchronous read data strobe)
- when data may start to transfer from the master device to a HyperRAM device in write transactions (initial write latency)
- data masking during write data transfers

During the CA transfer portion of a read or write transaction, RWDS acts as an output from a HyperRAM device to indicate whether additional initial access latency is needed in the transaction.

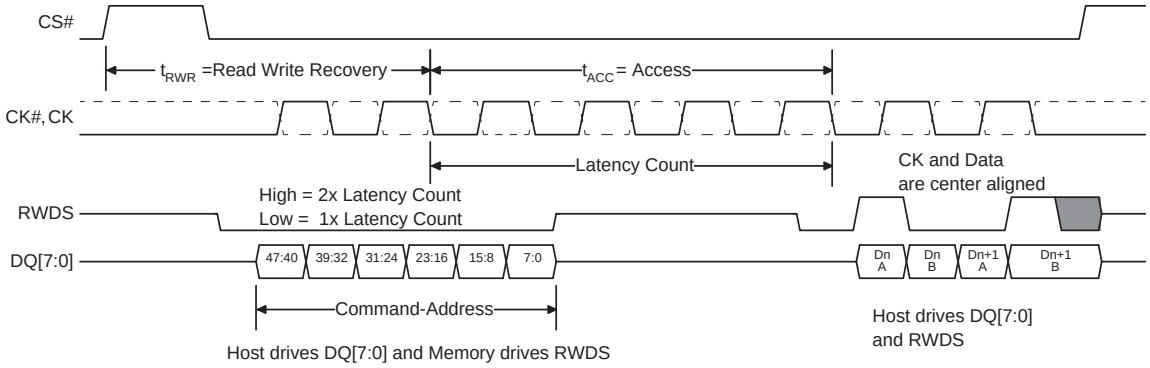
During read data transfers, RWDS is a read data strobe with data values edge aligned with the transitions of RWDS.

**Figure 1.2** Read Transaction, Additional Latency Count



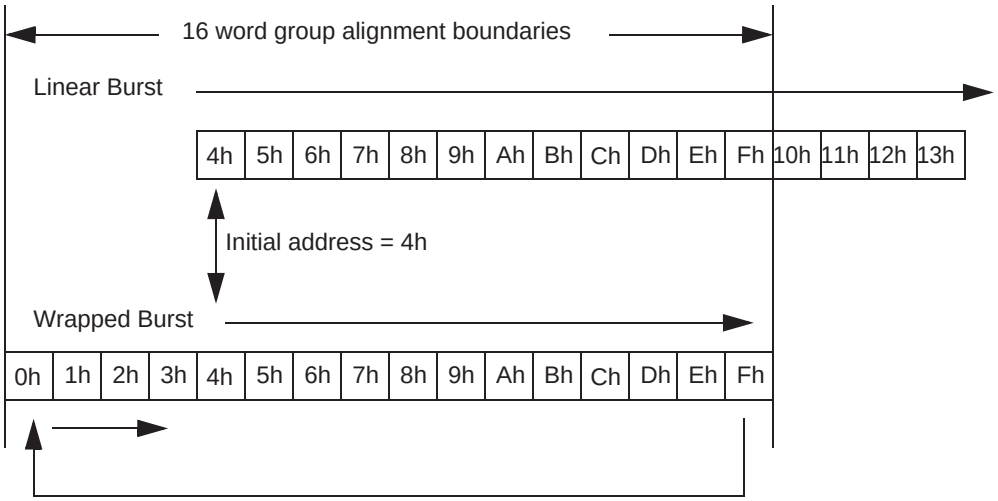

During write data transfers, RWDS indicates whether each data byte transfer is masked with RWDS High (invalid and prevented from changing the byte location in a memory) or not masked with RWDS Low (valid and written to a memory). Data masking may be used by the host to byte align write data within a memory or to enable merging of multiple non-word aligned writes in a single burst write. During write transactions, data is center aligned with clock transitions.

**Figure 1.3** Write Transaction, Single Initial Latency Count




Read and write transactions are burst oriented, transferring the next sequential word during each clock cycle. Each individual read or write transaction can use either a wrapped or linear burst sequence.

**Figure 1.4** Linear Versus Wrapped Burst Sequence

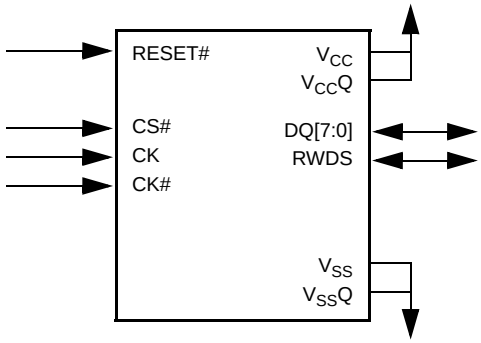



During wrapped transactions, accesses start at a selected location and continue to the end of a configured word group aligned boundary, then wrap to the beginning location in the group, then continue back to the starting location. Wrapped bursts are generally used for critical word first cache line fill read transactions. During linear transactions, accesses start at a selected location and continue in a sequential manner until the transaction is terminated when CS# returns High. Linear transactions are generally used for large contiguous data transfers such as graphic images. Since each transaction command selects the type of burst sequence for that transaction, wrapped and linear bursts transactions can be dynamically intermixed as needed.

## 2. HyperRAM Product Overview

The 64-Mbit HyperRAM device is a 1.8V or 3.0V core and I/O, synchronous self-refresh Dynamic RAM (DRAM). The HyperRAM device provides a HyperBus slave interface to the host system. HyperBus has an 8-bit (1 byte) wide DDR data bus and uses only word-wide (16-bit data) address boundaries. Read transactions provide 16 bits of data during each clock cycle (8 bits on both clock edges). Write transactions take 16 bits of data from each clock cycle (8 bits on each clock edge).

**Figure 2.1** HyperRAM Interface




Read and write transactions require two clock cycles to define the target row address and burst type, then an initial access latency of $t_{ACC}$. During the Command-Address (CA) part of a transaction, the memory will indicate whether an additional latency for a required refresh time ($t_{RFH}$) is added to the initial latency; by driving the RWDS signal to the High state. During the CA period the third clock cycle will specify the target word address within the target row. During a read (or write) transaction, after the initial data value has been output (or input), additional data can be read from (or written to) the row on subsequent clock cycles in either a wrapped or linear sequence. When configured in linear burst mode, the device will automatically fetch the next sequential row from the memory array to support a continuous linear burst. Simultaneously accessing the next row in the array while the read or write data transfer is in progress, allows for a linear sequential burst operation that can provide a sustained data rate of 333 MB/s (1 byte (8 bit data bus) * 2 *(data clock edges)* * 166 MHz = 333 MB/s).

# 3. HyperRAM Signal Descriptions

## 3.1 Input/Output Summary

HyperRAM signals are shown in Table 3.1. Active Low signal names have a hash symbol (#) suffix.

**Table 3.1** I/O Summary

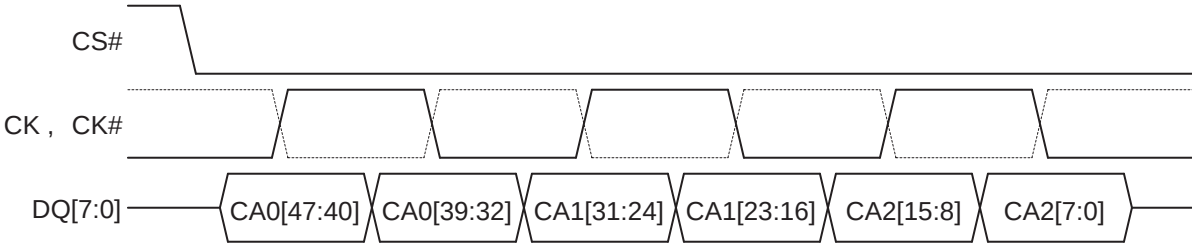
| Symbol | Type | Description |
|---|---|---|
| CS# | Master Output, Slave Input | **Chip Select.** Bus transactions are initiated with a High to Low transition. Bus transactions are terminated with a Low to High transition. The master device has a separate CS# for each slave. |
| CK, CK# | Master Output, Slave Input | **Differential Clock**. Command, address, and data information is output with respect to the crossing of the CK and CK# signals. Differential clock is used on 1.8V I/O devices.<br>**Single Ended Clock.** CK# is not used on 3.0V devices, only a single ended CK is used.<br>The clock is not required to be free-running. |
| DQ[7:0] | Input/Output | **Data Input/Output.** Command, Address, and Data information is transferred on these signals during Read and Write transactions. |
| RWDS | Input/Output | **Read Write Data Strobe.** During the Command/Address portion of all bus transactions RWDS is a slave output and indicates whether additional initial latency is required. Slave output during read data transfer, data is edge aligned with RWDS. Slave input during data transfer in write transactions to function as a data mask. (High = additional latency, Low = no additional latency). |
| RESET# | Master Output, Slave Input, Internal Pull-up | **Hardware RESET**. When Low the slave device will self initialize and return to the Standby state. RWDS and DQ[7:0] are placed into the High-Z state when RESET# is Low. The slave RESET# input includes a weak pull-up, if RESET# is left unconnected it will be pulled up to the High state. |
| $V_{CC}$ | Power Supply | **Core Power.** |
| $V_{CC}Q$ | Power Supply | **Core Input/Output Power.** |
| $V_{SS}$ | Power Supply | **Core Ground.** |
| $V_{SS}Q$ | Power Supply | **Core Input/Output Power.** |

## 3.2 Command/Address Bit Assignments

All HyperRAM bus transactions can be classified as either read or write. A bus transaction is started with CS# going Low with clock in idle state (CK=Low and CK#=High). The first three clock cycles transfer three words of Command/Address (CA0, CA1, CA2) information to define the transaction characteristics. The Command/Address words are presented with DDR timing, using the first six clock edges. The following characteristics are defined by the Command/Address information:

- Read or Write transaction
- Address Space: memory array space or register space
  - Register space is used to access Device Identification (ID) registers and Configuration Registers (CR) that identify the device characteristics and determine the slave specific behavior of read and write transfers on the HyperBus interface.
- Whether a transaction will use a linear or wrapped burst sequence.
- The target row (and half-page) address (upper order address)
- The target column (word within half-page) address (lower order address)

**Figure 3.1** Command-Address Sequence




***Notes:***

1. *Figure shows the initial three clock cycles of all transactions on the HyperBus.*
2. *CK# of differential clock is shown as dashed line waveform.*
3. *Command-Address information is "center aligned" with the clock during both Read and Write transactions.*

**Table 3.2** Command-Address Bit Assignment to DQ Signals

| Signal | CA0[47:40] | CA0[39:32] | CA1[31:24] | CA1[23:16] | CA2[15:8] | CA2[7:0] |
|---|---|---|---|---|---|---|
| **DQ[7]** | CA[47] | CA[39] | CA[31] | CA[23] | CA[15] | CA[7] |
| **DQ[6]** | CA[46] | CA[38] | CA[30] | CA[22] | CA[14] | CA[6] |
| **DQ[5]** | CA[45] | CA[37] | CA[29] | CA[21] | CA[13] | CA[5] |
| **DQ[4]** | CA[44] | CA[36] | CA[28] | CA[20] | CA[12] | CA[4] |
| **DQ[3]** | CA[43] | CA[35] | CA[27] | CA[19] | CA[11] | CA[3] |
| **DQ[2]** | CA[42] | CA[34] | CA[26] | CA[18] | CA[10] | CA[2] |
| **DQ[1]** | CA[41] | CA[33] | CA[25] | CA[17] | CA[9] | CA[1] |
| **DQ[0]** | CA[40] | CA[32] | CA[24] | CA[16] | CA[8] | CA[0] |

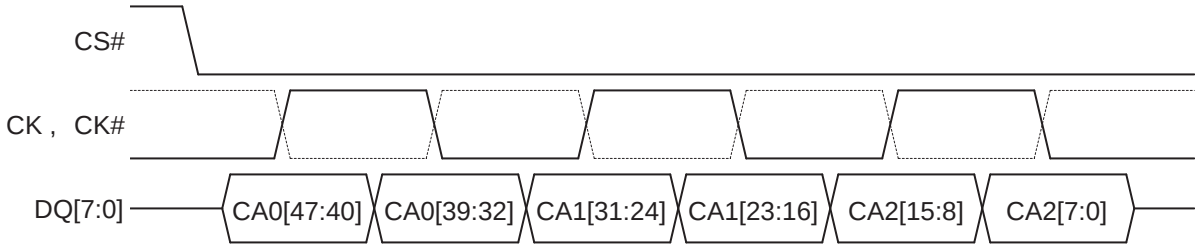
**Table 3.3** Command/Address Bit Assignments

| CA Bit# | Bit Name | Bit Function |
|---|---|---|
| 47 | R/W# | Identifies the transaction as a read or write.<br>R/W#=1 indicates a Read transaction<br>R/W#=0 indicates a Write transaction |
| 46 | Address Space (AS) | Indicates whether the read or write transaction accesses the memory or register space.<br>AS=0 indicates memory space<br>AS=1 indicates the register space<br>The register space is used to access device ID and Configuration registers. |
| 45 | Burst Type | Indicates whether the burst will be linear or wrapped.<br>Burst Type=0 indicates wrapped burst<br>Burst Type=1 indicates linear burst |
| 44-16 | Row & Upper Column Address | Row & Upper Column component of the target address: System word address bits A31-A3<br>Any upper Row address bits not used by a particular device density should be set to 0 by the host controller master interface. The size of Rows and therefore the address bit boundary between Row and Column address is slave device dependent. |
| 15-3 | Reserved | Reserved for future column address expansion.<br>Reserved bits are don't care in current HyperBus devices but should be set to 0 by the host controller master interface for future compatibility. |
| 2-0 | Lower Column Address | Lower Column component of the target address: System word address bits A2-0 selecting the starting word within a half-page. |

***Notes:***

1. *A Row is a group of words relevant to the internal memory array structure and additional latency may be inserted by RWDS when crossing Row boundaries - this is device dependent behavior, refer to each HyperBus device data sheet for additional information. Also, the number of Rows may be used in the calculation of a distributed refresh interval for HyperRAM memory.*
2. *A Page is a 16-word (32-byte) length and aligned unit of device internal read or write access and additional latency may be inserted by RWDS when crossing Page boundaries - this is device dependent behavior, refer to each HyperBus device data sheet for additional information.*
3. *The Column address selects the burst transaction starting word location within a Row. The Column address is split into an upper and lower portion. The upper portion selects an 8-word (16-byte) Half-page and the lower portion selects the word within a Half-page where a read or write transaction burst starts.*
4. *The initial read access time starts when the Row and Upper Column (Half-page) address bits are captured by a slave interface. Continuous linear read burst is enabled by memory devices internally interleaving access to 16 byte half-pages.*
5. *HyperBus protocol address space limit, assuming:*
   *29 Row &Upper Column address bits*
   *3 Lower Column address bits*
   *Each address selects a word wide (16 bit = 2 byte) data value*
   *29 + 3 = 32 address bits = 4G addresses supporting 8Gbyte (64Gbit) maximum address space*
   *Future expansion of the column address can allow for 29 Row &Upper Column + 16 Lower Column address bits = 35 Tera-word = 70 Tera-byte address space.*

**Figure 3.2** Data Placement During a Read Transaction




***Notes:***

1. *Figure shows a portion of a Read transaction on the HyperBus. CK# of differential clock is shown as dashed line waveform.*
2. *Data is "edge aligned" with the RWDS serving as a read data strobe during read transactions.*
3. *Data is always transferred in full word increments (word granularity transfers).*
4. *Word address increments in each clock cycle. Byte A is between RWDS rising and falling edges and is followed by byte B between RWDS falling and rising edges, of each word.*
5. *Data bits in each byte are always in high to low order with bit 7 on DQ7 and bit 0 on DQ0.*

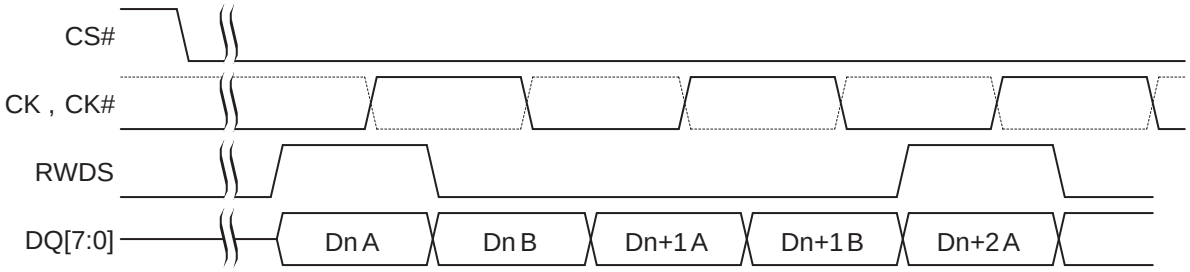
**Table 3.4** Data Bit Placement During Read or Write Transaction

| Address Space | Byte Order | Byte Position | Word Data Bit | DQ | Bit Order |
|---|---|---|---|---|---|
| Memory | Big-endian | A | 15 | 7 | When data is being accessed in memory space:<br>*The first byte of each word read or written is the "A" byte and the second is the "B" byte.*<br>*The bits of the word within the A and B bytes depend on how the data was written. If the word lower address bits 7-0 are written in the A byte position and bits 15-8 are written into the B byte position, or vice versa, they will be read back in the same order.*<br><br>So, memory space can be stored and read in either little-endian or big-endian order. |
| | | | 14 | 6 | |
| | | | 13 | 5 | |
| | | | 12 | 4 | |
| | | | 11 | 3 | |
| | | | 10 | 2 | |
| | | | 9 | 1 | |
| | | | 8 | 0 | |
| | | B | 7 | 7 | |
| | | | 6 | 6 | |
| | | | 5 | 5 | |
| | | | 4 | 4 | |
| | | | 3 | 3 | |
| | | | 2 | 2 | |
| | | | 1 | 1 | |
| | | | 0 | 0 | |
| | Little-endian | A | 7 | 7 | |
| | | | 6 | 6 | |
| | | | 5 | 5 | |
| | | | 4 | 4 | |
| | | | 3 | 3 | |
| | | | 2 | 2 | |
| | | | 1 | 1 | |
| | | | 0 | 0 | |
| | | B | 15 | 7 | |
| | | | 14 | 6 | |
| | | | 13 | 5 | |
| | | | 12 | 4 | |
| | | | 11 | 3 | |
| | | | 10 | 2 | |
| | | | 9 | 1 | |
| | | | 8 | 0 | |

**Table 3.4** Data Bit Placement During Read or Write Transaction (Continued)

| Address Space | Byte Order | Byte Position | Word Data Bit | DQ | Bit Order |
|---|---|---|---|---|---|
| Register | Big-endian | A | 15 | 7 | When data is being accessed in register space:<br>During a Read transaction on the HyperBus two bytes are transferred on each clock cycle. The upper order byte A (Word[15:8]) is transferred between the rising and falling edges of RWDS (edge aligned). The lower order byte B (Word[7:0]) is transferred between the falling and rising edges of RWDS.<br><br>During a write, the upper order byte A (Word[15:8]) is transferred on the CK rising edge and the lower order byte B (Word[7:0]) is transferred on the CK falling edge.<br>So, register space is always read and written in Big-endian order because registers have device dependent fixed bit location and meaning definitions. |
| | | | 14 | 6 | |
| | | | 13 | 5 | |
| | | | 12 | 4 | |
| | | | 11 | 3 | |
| | | | 10 | 2 | |
| | | | 9 | 1 | |
| | | | 8 | 0 | |
| | | B | 7 | 7 | |
| | | | 6 | 6 | |
| | | | 5 | 5 | |
| | | | 4 | 4 | |
| | | | 3 | 3 | |
| | | | 2 | 2 | |
| | | | 1 | 1 | |
| | | | 0 | 0 | |

**Figure 3.3** Data Placement During a Write Transaction




***Notes:***

1. *Figure shows a portion of a Write transaction on the HyperBus.*
2. *Data is "center aligned" with the clock during a Write transaction.*
3. *RWDS functions as a data mask during write data transfers with initial latency. Masking of the first and last byte is shown to illustrate an unaligned 3 byte write of data.*
4. *RWDS is not driven by the master during write data transfers with zero initial latency. Full data words are always written in this case. RWDS may be driven low or left High-Z by the slave in this case.*

## 3.3 Read Transactions

The HyperBus master begins a transaction by driving CS# Low while clock is idle. Then the clock begins toggling while Command-Address CA words are transfered.

In CA0, CA[47] = 1 indicates that a Read transaction is to be performed. CA[46] = 0 indicates the memory space is being read or CA[46] = 0 indicates the register space is being read. CA[45] indicates the burst type (wrapped or linear). Read transactions can begin the internal array access as soon as the row and upper column address has been presented in CA0 and CA1 (CA[47:16]). CA2 (CA(15:0]) identifies the target Word address within the chosen row. However, some HyperBus devices may require a minimum time between the end of a prior transaction and the start of a new access. This time is referred to as Read-Write-Recovery time ($t_{RWR}$). The master interface must start driving CS# Low only at a time when the CA1 transfer will complete after $t_{RWR}$ is satisfied.

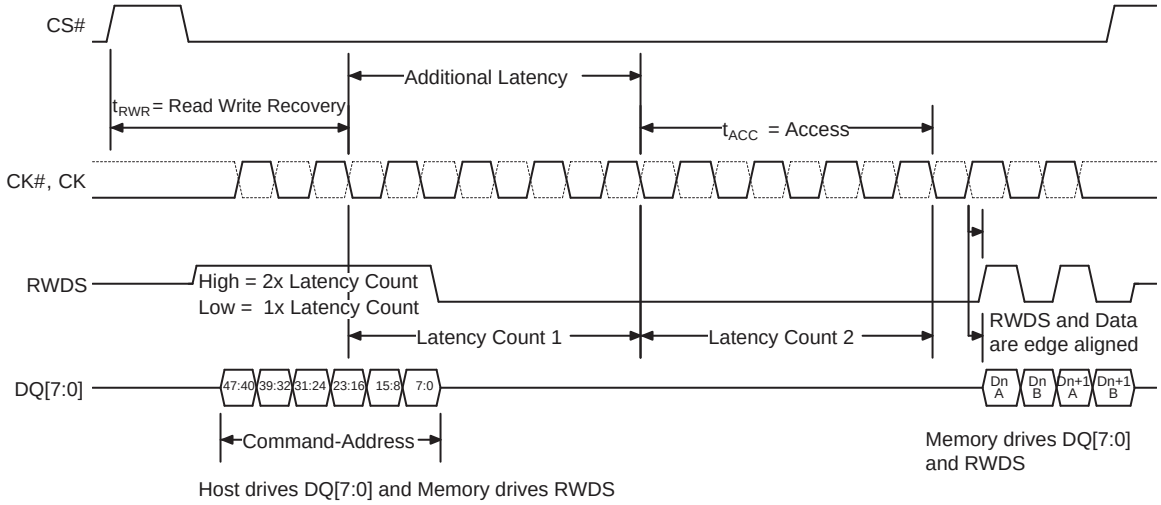
The HyperBus master then continues clocking for a number of cycles defined by the latency count setting in configuration register 0. The initial latency count required for a particular clock frequency is based on RWDS. If RWDS is Low during the CA cycles, one latency count is inserted. If RWDS is High during the CA cycles, an additional latency count is inserted. Once these latency clocks have been completed the memory starts to simultaneously transition the Read-Write Data Strobe (RWDS) and output the target data.

New data is output edge aligned with every transition of RWDS. Data will continue to be output as long as the host continues to transition the clock while CS# is Low. However, the HyperRAM device may stop RWDS transitions with RWDS Low, between the delivery of words, in order to insert latency between words when crossing memory array boundaries.

Wrapped bursts will continue to wrap within the burst length and linear burst will output data in a sequential manner across row boundaries. When a linear burst read reaches the last address in the array, continuing the burst beyond the last address will provide undefined data. Read transfers can be ended at any time by bringing CS# High when the clock is idle.

The clock is not required to be free-running. The clock may remain idle while CS# is high.

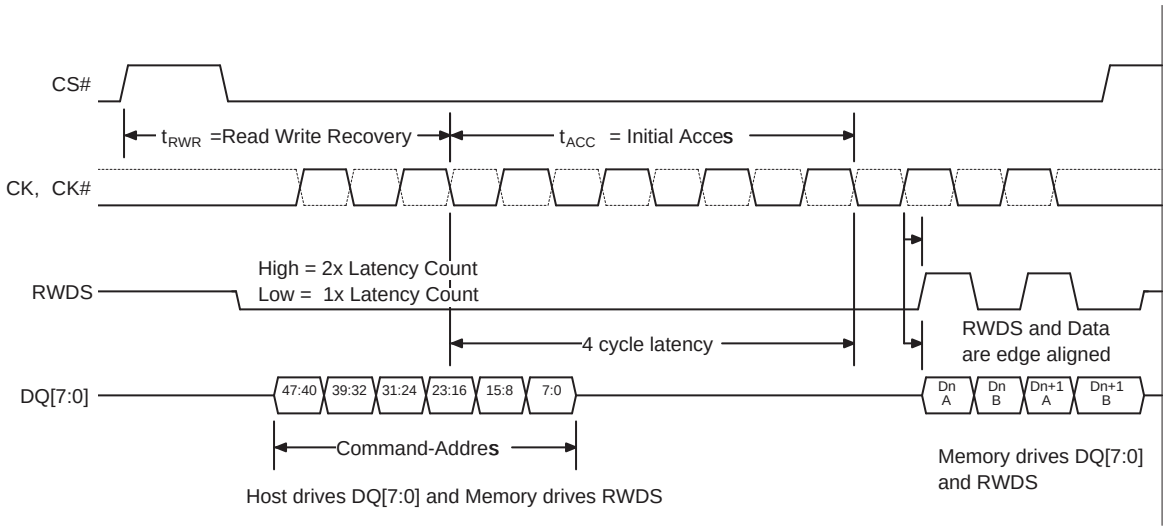
**Figure 3.4** Read Transaction with Additional Initial Latency




***Notes:***

1. *Transactions are initiated with CS# falling while CK=Low and CK#=High.*
2. *CS# must return High before a new transaction is initiated.*
3. *CK# is the complement of the CK signal. 3V devices use a single ended clock (CK only), CK# is used with CK on1.8V devices to provide a differential clock. CK# of a differential clock is shown as a dashed line waveform.*
4. *Read access array starts once CA[23:16] is captured.*
5. *The read latency is defined by the initial latency value in a configuration register.*
6. *In this read transaction example the initial latency count was set to four clocks.*
7. *In this read transaction a RWDS High indication during CA delays output of target data by an additional four clocks.*
8. *The memory device drives RWDS during read transactions.*

**Figure 3.5** Read Transaction Without Additional Initial Latency




***Notes:***

1. *RWDS is Low during the CA cycles. In this Read Transaction there is a single initial latency count for read data access because, this read transaction does not begin at a time when additional latency is required by the slave.*

## 3.4 Write Transactions with Initial Latency (Memory Core Write)

The HyperBus master begins a transaction by driving CS# Low while clock is idle. Then the clock begins toggling while Command-Address CA words are transfered.

In CA0, CA[47] = 0 indicates that a Write transaction is to be performed. CA[46] = 0 indicates the memory space is being written. CA[45] indicates the burst type (wrapped or linear). Write transactions can begin the internal array access as soon as the row and upper column address has been presented in CA0 and CA1 (CA[47:16]). CA2 (CA(15:0]) identifies the target word address within the chosen row. However, some HyperBus devices may require a minimum time between the end of a prior transaction and the start of a new access. This time is referred to as Read-Write-Recovery time ($t_{RWR}$). The master interface must start driving CS# Low only at a time when the CA1 transfer will complete after $t_{RWR}$ is satisfied.

The HyperBus master then continues clocking for a number of cycles defined by the latency count setting in configuration register 0. The initial latency count required for a particular clock frequency is based on RWDS. If RWDS is Low during the CA cycles, one latency count is inserted. If RWDS is High during the CA cycles, an additional latency count is inserted.

Once these latency clocks have been completed the HyperBus master starts to output the target data. Write data is center aligned with the clock edges. The first byte of data in each word is captured by the memory on the rising edge of CK and the second byte is captured on the falling edge of CK.

During the CA clock cycles, RWDS is driven by the memory.

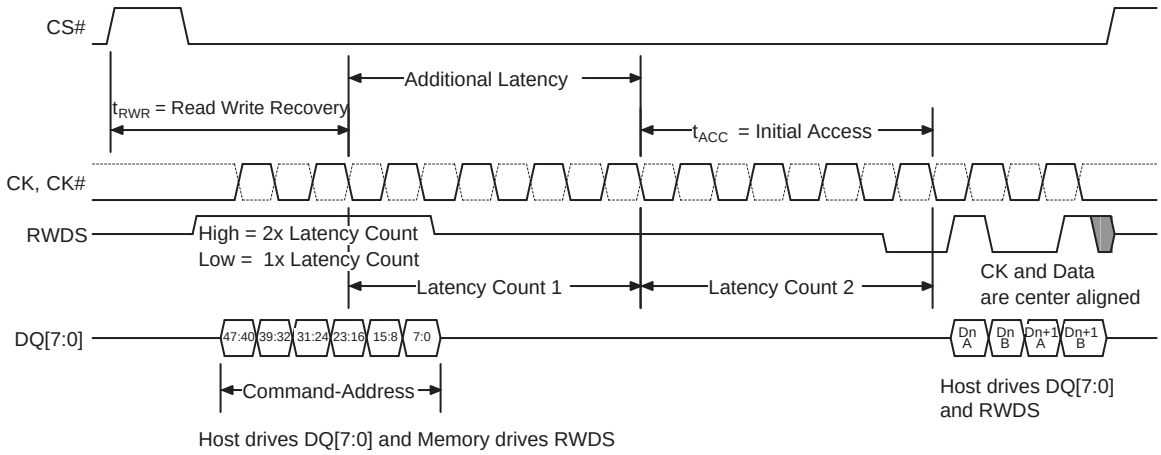
During the write data transfers, RWDS is driven by the host master interface as a data mask. When data is being written and RWDS is High the byte will be masked and the array will not be altered. When data is being written and RWDS is Low the data will be placed into the array. Because the master is driving RWDS during write data transfers, neither the master nor the HyperRAM device are able to indicate a need for latency within the data transfer portion of a write transaction. The acceptable write data burst length setting is also shown in configuration register 0.

Data will continue to be transferred as long as the HyperBus master continues to transition the clock while CS# is Low. Legacy format wrapped bursts will continue to wrap within the burst length. Hybrid wrap will wrap once then switch to linear burst starting at the next wrap boundary. Linear burst accepts data in a sequential manner across page boundaries. Write transfers can be ended at any time by bringing CS# High when the clock is idle.

When a linear burst write reaches the last address in the memory array space, continuing the burst will write to the beginning of the address range.

The clock is not required to be free-running. The clock may remain idle while CS# is high.

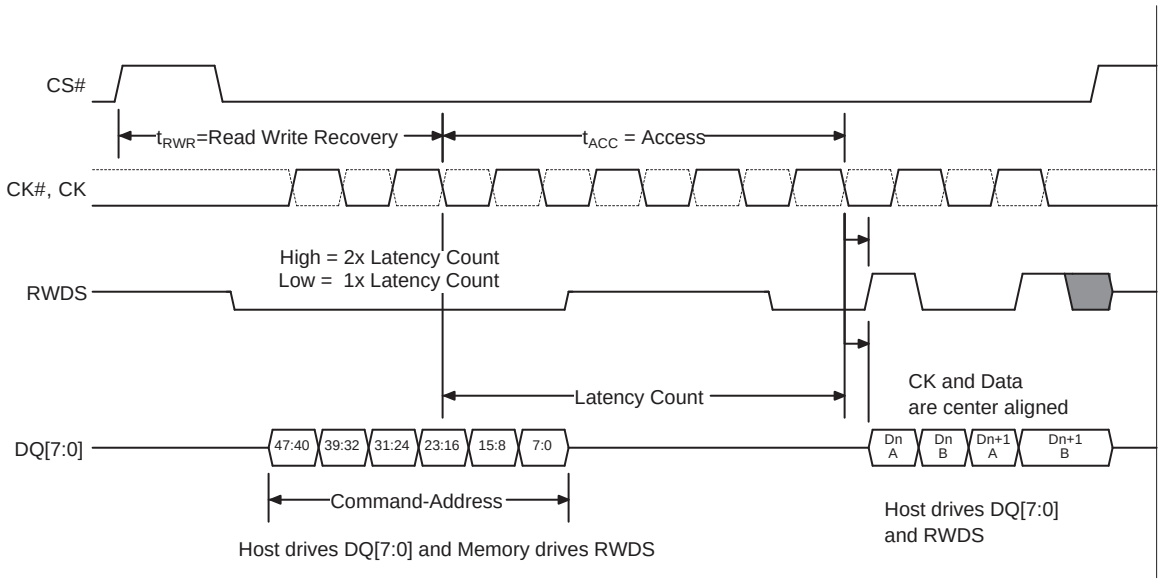
**Figure 3.6** Write Transaction with Additional Initial Latency




***Notes:***

1. *Transactions must be initiated with CK=Low and CK#=High.*
2. *CS# must return High before a new transaction is initiated.*
3. *During Command-Address, RWDS is driven by the memory and indicates whether additional latency cycles are required.*
4. *In this example, RWDS indicates that additional initial latency cycles are required.*
5. *At the end of Command-Address cycles the memory stops driving RWDS to allow the host HyperBus master to begin driving RWDS. The master must drive RWDS to a valid Low before the end of the initial latency to provide a data mask preamble period to the slave.*
6. *During data transfer, RWDS is driven by the host to indicate which bytes of data should be either masked or loaded into the array.*
7. *The figure shows RWDS masking byte A0 and byte B1 to perform an unaligned word write to bytes B0 and A1.*

**Figure 3.7** Write Transaction Without Additional Initial Latency




***Notes:***

1. *During Command-Address, RWDS is driven by the memory and indicates whether additional latency cycles are required.*
2. *In this example, RWDS indicates that there is no additional latency required.*
3. *At the end of Command-Address cycles the memory stops driving RWDS to allow the host HyperBus master to begin driving RWDS. The master must drive RWDS to a valid Low before the end of the initial latency to provide a data mask preamble period to the slave.*
4. *During data transfer, RWDS is driven by the host to indicate which bytes of data should be either masked or loaded into the array.*
5. *The figure shows RWDS masking byte A0 and byte B1 to perform an unaligned word write to bytes B0 and A1.*

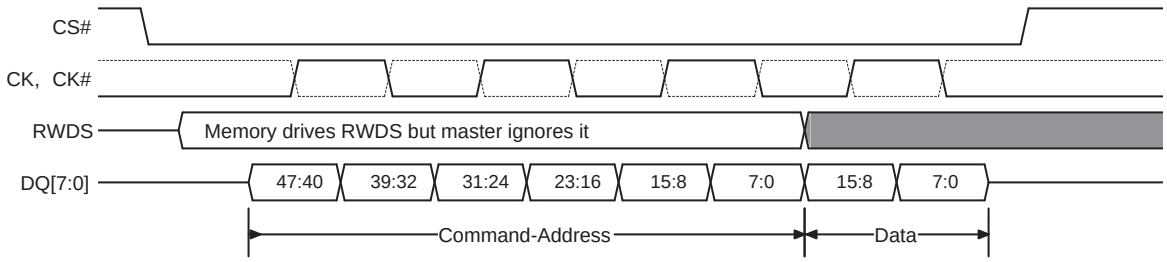
## 3.5 Write Transactions without Initial Latency (Register Write)

A Write transaction starts with the first three clock cycles providing the Command/Address information indicating the transaction characteristics. CA0 may indicate that a Write transaction is to be performed and also indicates the address space and burst type (wrapped or linear).

Writes without initial latency are used for register space writes. HyperRAM device write transactions with zero latency mean that the CA cycles are followed by write data transfers. Writes with zero initial latency, do not have a turn around period for RWDS. The HyperRAM device will always drive RWDS during the Command-Address period to indicate whether extended latency is required for a transaction that has initial latency. However, the RWDS is driven before the HyperRAM device has received the first byte of CA i.e. before the HyperRAM device knows whether the transaction is a read or write to register space. In the case of a write with zero latency, the RWDS state during the CA period does not affect the initial latency of zero. Since master write data immediately follows the Command-Address period in this case, the HyperRAM device may continue to drive RWDS Low or may take RWDS to High-Z during write data transfer. The master must not drive RWDS during Writes with zero latency. Writes with zero latency do not use RWDS as a data mask function. All bytes of write data are written (full word writes).

The first byte of data in each word is presented on the rising edge of CK and the second byte is presented on the falling edge of CK. Write data is center aligned with the clock inputs. Write transfers can be ended at any time by bringing CS# High when clock is idle. The clock is not required to be free-running.

**Figure 3.8** Write Operation without Initial Latency




Host drives DQ[7:0] with Command-Address and Write Data

# 4. Memory Space

When CA[46] is 0 a read or write transaction accesses the DRAM memory array.

**Table 4.1** Memory Space Address Map

| Unit Type | Count | System Word Address Bits | CA Bits | Notes |
|---|---|---|---|---|
| Rows within 64 Mb device | 8192 (Rows) | A21 - A9 | 34 - 22 | |
| Row | 1 (row) | A8 - A3 | 21 - 16 | 512 (word addresses)<br>1 kbytes |
| Half-Page | 8 (word addresses) | A2 - A0 | 2 - 0 | 16 bytes |

# 5. Register Space

When CA[46] is 1 a read or write transaction accesses the Register Space.

**Table 5.1** Register Space Address Map

| Register | System Address | — | — | — | 31-27 | 26-19 | 18-11 | 10-3 | — | 2-0 |
|---|---|---|---|---|---|---|---|---|---|---|
| | CA Bits | 47 | 46 | 45 | 44-40 | 39-32 | 31-24 | 23-16 | 15-8 | 7-0 |
| Identification Register 0 (read only) | | C0h or E0h | | | | 00h | 00h | 00h | 00h | 00h |
| Identification Register 1 (read only) | | C0h or E0h | | | | 00h | 00h | 00h | 00h | 01h |
| Configuration Register 0 Read | | C0h or E0h | | | | 00h | 01h | 00h | 00h | 00h |
| Configuration Register 0 Write | | 60h | | | | 00h | 01h | 00h | 00h | 00h |
| Configuration Register 1 Read | | C0h or E0h | | | | 00h | 01h | 00h | 00h | 01h |
| Configuration Register 1 Write | | 60h | | | | 00h | 01h | 00h | 00h | 01h |

***Note:***

*1. CA45 may be either 0 or 1 for either wrapped or linear read. CA45 must be 1 as only linear single word register writes are supported.*

## 5.1 Device Identification Registers

There are two read only, non-volatile, word registers, that provide information on the device selected when CS# is low. The device information fields identify:

- Manufacturer
- Type
- Density
  - Row address bit count
  - Column address bit count

**Table 5.2** ID Register 0 Bit Assignments

| Bits | Function | Settings (Binary) |
|---|---|---|
| 15-14 | Reserved | Reserved |
| 13 | Reserved | 0 - default |
| 12-8 | Row Address Bit Count | 00000 - One Row address bit<br>...<br>11111 - Thirty-two row address bits |

**Table 5.2** ID Register 0 Bit Assignments (Continued)

| Bits | Function | Settings (Binary) |
|---|---|---|
| 7-4 | Column Address Bit Count | 0000 - One column address bit<br>...<br>1111 - Sixteen column address bits |
| 3-0 | Manufacturer | 0000 - Reserved<br>0011 - ISSI<br>0010 to 1111 - Reserved |

**Table 5.3** ID Register 1 Bit Assignments

| Bits | Function | Settings (Binary) |
|---|---|---|
| 15-4 | Reserved | Reserved |
| 3-0 | Device Type | 0000 - HyperRAM<br>0001 to 1111 - Reserved |

### 5.1.1 Density and Row Boundaries

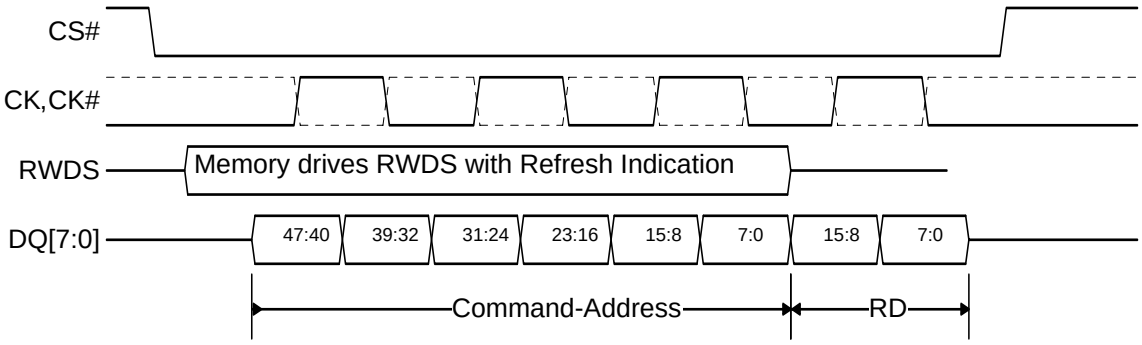
The DRAM array size (density) of the device can be determined from the total number of system address bits used for the row and column addresses as indicated by the Row Address Bit Count and Column Address Bit Count fields in the ID0 register. For example: a 64-Mbit HyperRAM device has 9 column address bits and 13 row address bits for a total of 22 word address bits = $2^{22}$ = 4 Mwords = 8 Mbytes. The 9 column address bits indicate that each row holds $2^9$ = 512 words = 1 kbytes. The row address bit count indicates there are 8196 rows to be refreshed within each array refresh interval. The row count is used in calculating the refresh interval.

## 5.2 Register Space Access

Register default values are loaded upon power-up or hardware reset. The registers can be altered at any time while the device is in the standby state.

Loading a register is accomplished with a single 16-bit word write transaction as shown in Figure 5.1. CA[47] is zero to indicate a write transaction, CA[46] is a one to indicate a register space write, CA[45] is a one to indicate a linear write, lower order bits in the CA field indicate the register address.

**Figure 5.1** Loading a Register




Host drives DQ[7:0] with Command-Address and Register Data

***Notes:***

1. *The host must not drive RWDS during a write to register space.*
2. *The RWDS signal is driven by the memory during the Command-Address period based on whether the memory array is being refreshed. This refresh indication does not affect the writing of register data. RWDS is driven immediately after CS# goes low, before CA[47:46] are received to indicate that the transaction is a write to register space, for which the RWDS refresh indication is not relevant.*
3. *The register value is always provided immediately after the CA value and is not delayed by a refresh latency.*
4. *The the RWDS signal returns to high impedance after the Command-Address period. Register data is never masked. Both data bytes of the register data are loaded into the selected register.*

Each register is written with a separate single word write transaction. Register write transactions have zero latency, the single word of data immediately follows the Command-Address. RWDS is not driven by the host during the write because RWDS is always driven by the memory during the CA cycles to indicate whether a memory array refresh is in progress. Because a register space write goes directly to a register, rather than the memory array, there is no initial write latency, related to an array refresh that may be in progress. In a register write, RWDS is also not used as a data mask because both bytes of a register are always written and never masked.

Reserved register fields must be written with their default value. Writing reserved fields with other than default values may produce undefined results.

Reading of a register is accomplished with a single 16 bit read transaction with CA[46]=1 to select register space. If more than one word is read, the same register value is repeated in each word read. The CA[45] burst type is “don’t care” because only a single register value is read. The contents of the register is returned in the same manner as reading array data, with one or two latency counts, based on the state of RWDS during the Command-Address period. The latency count is defined in the Configuration Register 0 Read Latency field (CR0[7:4]).

### 5.2.1 Configuration Register 0

Configuration Register 0 (CR0) is used to define the power mode and access protocol operating conditions for the HyperRAM device. Configurable characteristics include:

- Wrapped Burst Length (16, 32, 64, or 128-byte aligned and length data group)
- Wrapped Burst Type
  - Legacy wrap (sequential access with wrap around within a selected length and aligned group)
  - Hybrid wrap (Legacy wrap once then linear burst at start of the next sequential group)
- Initial Latency
- Variable Latency
  - Whether an array read or write transaction will use fixed or variable latency. If fixed latency is selected the memory will always indicate a refresh latency and delay the read data transfer accordingly. If variable latency is selected, latency for a refresh is only added when a refresh is required at the same time a new transaction is starting.
- Output Drive Strength
- Deep Power Down Mode

**Table 5.4** Configuration Register 0 Bit Assignments

| CR0 Bit | Function | Settings (Binary) |
|---|---|---|
| 15 | Deep Power Down Enable | 1 - Normal operation (default)<br>0 - Writing 0 to CR[15] causes the device to enter Deep Power Down |
| 14-12 | Drive Strength | 000 - 34 ohms (default)<br>001 - 115 ohms<br>010 - 67 ohms<br>011 - 46 ohms<br>100 - 34 ohms<br>101 - 27 ohms<br>110 - 22 ohms<br>111 - 19 ohms |
| 11-8 | Reserved | 1 - Reserved (default)<br>Reserved for Future Use. When writing this register, these bits should be set to 1 for future compatibility. |

**Table 5.4** Configuration Register 0 Bit Assignments (Continued)

| CR0 Bit | Function | Settings (Binary) |
|---|---|---|
| 7-4 | Initial Latency | 0000 - 5 Clock Latency<br>0001 - 6 Clock Latency (default)<br>0010 - Reserved<br>0011 - Reserved<br>0100 - Reserved<br>...<br>1101 - Reserved<br>1110 - 3 Clock Latency<br>1111 - 4 Clock Latency |
| 3 | Fixed Latency Enable | 0 - Variable Latency - 1 or 2 times Initial Latency depending on RWDS during CA cycles.<br>1 - Fixed 2 times Initial Latency (default) |
| 2 | Hybrid Burst Enable | 0: Wrapped burst sequences to follow hybrid burst sequencing<br>1: Wrapped burst sequences in legacy wrapped burst manner (default) |
| 1-0 | Burst Length | 00 - 128 bytes<br>01 - 64 bytes<br>10- 16 bytes<br>11 - 32 bytes (default) |

#### 5.2.1.1 Wrapped Burst

A wrapped burst transaction accesses memory within a group of words aligned on a word boundary matching the length of the configured group. Wrapped access groups can be configured as 16, 32, 64, or 128 bytes alignment and length. During wrapped transactions, access starts at the Command-Address selected location within the group, continues to the end of the configured word group aligned boundary, then wraps around to the beginning location in the group, then continues back to the starting location. Wrapped bursts are generally used for critical word first instruction or data cache line fill read accesses.

#### 5.2.1.2 Hybrid Burst

The beginning of a hybrid burst will wrap within the target address wrapped burst group length before continuing to the next half-page of data beyond the end of the wrap group. Continued access is in linear burst order until the transfer is ended by returning CS# high. This hybrid of a wrapped burst followed by a linear burst starting at the beginning of the next burst group, allows multiple sequential address cache lines to be filled in a single access. The first cache line is filled starting at the critical word. Then the next sequential line in memory can be read in to the cache while the first line is being processed.

**Table 5.5** CR0[2] Control of Wrapped Burst Sequence

| Bit | Default Value | Name |
|---|---|---|
| 2 | 1 | Hybrid Burst Enable<br>CR[2]= 0: Wrapped burst sequences to follow hybrid burst sequencing<br>CR[2]= 1: Wrapped burst sequences in legacy wrapped burst manner |

.

**Table 5.6** Example Wrapped Burst Sequences

| Burst Selection | | Burst Type | Wrap Boundary (bytes) | Start Address (Hex) | Address Sequence (Hex) (Words) |
|---|---|---|---|---|---|
| CA[45] | CR0[2:0] | | | | |
| 0 | 000 | Hybrid 128 | 128 Wrap once then Linear | XXXXXX03 | 03, 04, 05, 06, 07, 08, 09, 0A, 0B, 0C, 0D, 0E, 0F, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 1A, 1B, 1C, 1D, 1E, 1F, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 2A, 2B, 2C, 2D, 2E, 2F, 30, 31, 32, 33, 34, 35, 36, 37, 38, 39, 3A, 3B, 3C, 3D, 3E, 3F, 00, 01, 02<br>(wrap complete, now linear beyond the end of the initial 128 byte wrap group)<br>40, 41, 42, 43, 44, 45, 46, 47, 48, 49, 4A, 4B, 4C, 4D, 4E, 4F, 50, 51, ... |
| 0 | 001 | Hybrid 64 | 64 Wrap once then Linear | XXXXXX03 | 03, 04, 05, 06, 07, 08, 09, 0A, 0B, 0C, 0D, 0E, 0F, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 1A, 1B, 1C, 1D, 1E, 1F, 00, 01, 02,<br>(wrap complete, now linear beyond the end of the initial 64 byte wrap group)<br>20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 2A, 2B, 2C, 2D, 2E, 2F, 30, 31, ... |
| 0 | 001 | Hybrid 64 | 64 Wrap once then Linear | XXXXXX2E | 2E, 2F, 30, 31, 32, 33, 34, 35, 36, 37, 38, 39, 3A, 3B, 3C, 3D, 3E, 3F, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 2A, 2B, 2C, 2D,<br>(wrap complete, now linear beyond the end of the initial 64 byte wrap group)<br>40, 41, 42, 43, 44, 45, 46, 47, 48, 49, 4A, 4B, 4C, 4D, 4E, 4F, 50, 51, ... |
| 0 | 010 | Hybrid 16 | 16 Wrap once then Linear | XXXXXX02 | 02, 03, 04, 05, 06, 07, 00, 01,<br>(wrap complete, now linear beyond the end of the initial 16 byte wrap group)<br>08, 09, 0A, 0B, 0C, 0D, 0E, 0F, 10, 11, 12, ... |
| 0 | 010 | Hybrid 16 | 16 Wrap once then Linear | XXXXXX0C | 0C, 0D, 0E, 0F, 08, 09, 0A, 0B,<br>(wrap complete, now linear beyond the end of the initial 16 byte wrap group)<br>10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 1A, ... |
| 0 | 011 | Hybrid 32 | 32 Wrap once then Linear | XXXXXX0A | 0A, 0B, 0C, 0D, 0E, 0F, 00, 01, 02, 03, 04, 05, 06, 07, 08, 09, ... |
| 0 | 011 | Hybrid 32 | 32 Wrap once then Linear | XXXXXX1E | 1E, 1F, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 1A, 1B, 1C, 1D, ... |
| 0 | 100 | Wrap 128 | 128 | XXXXXX03 | 03, 04, 05, 06, 07, 08, 09, 0A, 0B, 0C, 0D, 0E, 0F, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 1A, 1B, 1C, 1D, 1E, 1F, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 2A, 2B, 2C, 2D, 2E, 2F, 30, 31, 32, 33, 34, 35, 36, 37, 38, 39, 3A, 3B, 3C, 3D, 3E, 3F, 00, 01, 02, ... |
| 0 | 101 | Wrap 64 | 64 | XXXXXX03 | 03, 04, 05, 06, 07, 08, 09, 0A, 0B, 0C, 0D, 0E, 0F, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 1A, 1B, 1C, 1D, 1E, 1F, 00, 01, 02, ... |
| 0 | 101 | Wrap 64 | 64 | XXXXXX2E | 2E, 2F, 30, 31, 32, 33, 34, 35, 36, 37, 38, 39, 3A, 3B, 3C, 3D, 3E, 3F, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 2A, 2B, 2C, 2D, ... |
| 0 | 110 | Wrap 16 | 16 | XXXXXX02 | 02, 03, 04, 05, 06, 07, 00, 01, ... |
| 0 | 110 | Wrap 16 | 16 | XXXXXX0C | 0C, 0D, 0E, 0F, 08, 09, 0A, 0B, ... |
| 0 | 111 | Wrap 32 | 32 | XXXXXX0A | 0A, 0B, 0C, 0D, 0E, 0F, 00, 01, 02, 03, 04, 05, 06, 07, 08, 09, ... |
| 0 | 111 | Wrap 32 | 32 | XXXXXX1E | 1E, 1F, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 1A, 1B, 1C, 1D, ... |
| 1 | XXX | Linear | Linear Burst | XXXXXX03 | 03, 04, 05, 06, 07, 08, 09, 0A, 0B, 0C, 0D, 0E, 0F, 10, 11, 12, 13, 14, 15, 16, 17, 18, ... |

#### 5.2.1.3 Initial Latency

Memory Space read and write transactions or Register Space read transactions require some initial latency to open the row selected by the Command-Address. This initial latency is $t_{ACC}$. The number of latency clocks needed to satisfy $t_{ACC}$ depends on the HyperBus frequency and can vary from 3 to 6 clocks. The value in CR0[7:4] selects the number of clocks for initial latency. The default value is 6 clocks, allowing for operation up to a maximum frequency of 166MHz prior to the host system setting a lower initial latency value that may be more optimal for the system.

In the event a distributed refresh is required at the time a Memory Space read or write transaction or Register Space read transaction begins, the RWDS signal goes high during the Command-Address to indicate that an additional initial latency is being inserted to allow a refresh operation to complete before opening the selected row.

Register Space write transactions always have zero initial latency. RWDS may be High or Low during the Command-Address period. The level of RWDS during the Command-Address period does not affect the placement of register data immediately after the Command-Address, as there is no initial latency needed to capture the register data. A refresh operation may be performed in the memory array in parallel with the capture of register data.

#### 5.2.1.4 Fixed Latency

A configuration register option bit CR0[3] is provided to make all Memory Space read and write transactions or Register Space read transactions require the same initial latency by always driving RWDS high during the Command-Address to indicate that two initial latency periods are required. This fixed initial latency is independent of any need for a distributed refresh, it simply provides a fixed (deterministic) initial latency for all of these transaction types. The fixed latency option may simplify the design of some HyperBus memory controllers or ensure deterministic transaction performance. Fixed latency is the default POR or reset configuration. The system may clear this configuration bit to disable fixed latency and allow variable initial latency with RWDS driven high only when additional latency for a refresh is required.

#### 5.2.1.5 Drive Strength

DQ signal line loading, length, and impedance vary depending on each system design. Configuration register bits CR0[14:12] provide a means to adjust the DQ[7:0] signal output impedance to customize the DQ signal impedance to the system conditions to minimize high speed signal behaviors such as overshoot, undershoot, and ringing. The default POR or reset configuration value is 000b to select the mid point of the available output impedance options.

The impedance values shown are typical for both pull-up and pull-down drivers at typical silicon process conditions, nominal operating voltage (1.8Vor 3V) and 50°C. The impedance values may vary by up to ±80% from the typical values depending on the Process, Voltage, and Temperature (PVT) conditions. Impedance will increase with slower process, lower voltage, or higher temperature. Impedance will decrease with faster process, higher voltage, or lower temperature.

Each system design should evaluate the data signal integrity across the operating voltage and temperature ranges to select the best drive strength settings for the operating conditions.

#### 5.2.1.6 Deep Power Down

When the HyperRAM device is not needed for system operation, it may be placed in a very low power consuming mode called Deep Power Down (DPD), by writing 0 to CR0[15]. When CR0[15] is cleared to 0, the device enters the DPD mode within $t_{DPDIN}$ time and all refresh operations stop. The data in RAM is lost, (becomes invalid without refresh) during DPD mode. The next access to the device driving CS# Low then High, POR, or a reset will cause the device to exit DPD mode. Returning to Standby mode requires $t_{DPDOUT}$ time. For additional details see Section 6.1.3, Deep Power Down on page 25.

### 5.2.2 Configuration Register 1

Configuration Register 1 (CR1) is used to define the distributed refresh interval for this HyperRAM device. The core DRAM array requires periodic refresh of all bits in the array. This can be done by the host system by reading or writing a location in each row within a specified time limit. The read or write access copies a row of bits to an internal buffer. At the end of the access the bits in the buffer are written back to the row in memory, thereby recharging (refreshing) the bits in the row of DRAM memory cells.

However, the host system generally has better things to do than to periodically read every row in memory and keep track that each row is visited within the required refresh interval for the entire memory array. HyperRAM devices include self-refresh logic that will refresh rows automatically so that the host system is relieved of the need to refresh the memory. The automatic refresh of a row can only be done when the memory is not being actively read or written by the host system. The refresh logic waits for the end of any active read or write before doing a refresh, if a refresh is needed at that time. If a new read or write begins before the refresh is completed, the memory will drive RWDS high during the Command-Address period to indicate that an additional initial latency time is required at the start of the new access in order to allow the refresh operation to complete before starting the new access.

The required refresh interval for the entire memory array varies with temperature as shown in Table 5.7, Array Refresh Interval per Temperature on page 23. This is the time within which all rows must be refreshed. Refresh of all rows could be done as a single batch of accesses at the beginning of each interval, in groups (burst refresh) of several rows at a time, spread throughout each interval, or as single row refreshes evenly distributed throughout the interval. The self-refresh logic distributes single row refresh operations throughout the interval so that the memory is not busy doing a burst of refresh operations for a long period, such that the burst refresh would delay host access for a long period.

**Table 5.7** Array Refresh Interval per Temperature

| Device Temperature (°C) | Array Refresh Interval (ms) | Array Rows | Recommended $t_{CMS}$ (µs) |
|---|---|---|---|
| 85 | 64 | 8192 | 4 |
| 105 | 16 | 8192 | 1 |

**Table 5.8** Configuration Register 1 Bit Assignments

| CR1 Bit | Function | Settings (Binary) |
|---|---|---|
| 15-2 | Reserved | 000000h — Reserved (default)<br>Reserved for Future Use. When writing this register, these bits should be cleared to 0 for future compatibility. |
| 1-0 | Distributed Refresh Interval | 10b — default<br>4 µs for Industrial temperature range devices<br>1 µs for Industrial Plus temperature range devices<br>11b — 1.5 times default<br>00b — 2 times default<br>01b — 4 times default |

The distributed refresh method requires that the host does not do burst transactions that are so long as to prevent the memory from doing the distributed refreshes when they are needed. This sets an upper limit on the length of read and write transactions so that the refresh logic can insert a refresh between transactions. This limit is called the CS# low maximum time ($t_{CMS}$). The $t_{CMS}$ value is determined by the array refresh interval divided by the number of rows in the array, then reducing this calculation by half to ensure that a distributed refresh interval cannot be entirely missed by a maximum length host access starting immediately before a distributed refresh is needed. Because $t_{CMS}$ is set to half the required distributed refresh interval, any series of maximum length host accesses that delay refresh operations will be catching up on refresh operations at twice the rate required by the refresh interval divided by the number of rows.

The host system is required to respect the $t_{CMS}$ value by ending each transaction before violating $t_{CMS}$. This can be done by host memory controller logic splitting long transactions when reaching the $t_{CMS}$ limit, or by host system hardware or software not performing a single read or write transaction that would be longer than $t_{CMS}$.

As noted in Table 5.7, Array Refresh Interval per Temperature on page 23 the array refresh interval is longer at lower temperatures such that $t_{CMS}$ could be increased to allow longer transactions. The host system can either use the $t_{CMS}$ value from the table for the maximum operating temperature or, may determine the current operating temperature from a temperature sensor in the system in order to set a longer distributed refresh interval.

The host system may also effectively increase the $t_{CMS}$ value by explicitly taking responsibility for performing all refresh and doing burst refresh reading of multiple sequential rows in order to catch up on distributed refreshes missed by longer transactions.

# HyperRAM Hardware Interface

For the general description of the HyperBus hardware interface of HyperFlash memories refer to the HyperBus Specification. The following section describes HyperRAM device dependent aspects of hardware interface.

## 6. Interface States

### 6.1 Power Conservation Modes

#### 6.1.1 Interface Standby

Standby is the default, low power, state for the interface while the device is not selected by the host for data transfer (CS#= High). All inputs, and outputs other than CS# and RESET# are ignored in this state.

#### 6.1.2 Active Clock Stop

The Active Clock Stop mode reduces device interface energy consumption to the $I_{CC6}$ level during the data transfer portion of a read or write operation. The device automatically enables this mode when clock remains stable for $t_{ACC}$ + 30 ns. While in Active Clock Stop mode, read data is latched and always driven onto the data bus. $I_{CC6}$ shown in Section 7.6, *DC Characteristics* on page 22.

Active Clock Stop mode helps reduce current consumption when the host system clock has stopped to pause the data transfer. Even though CS# may be Low throughout these extended data transfer cycles, the memory device host interface will go into the Active Clock Stop current level at $t_{ACC}$ + 30 ns. This allows the device to transition into a lower current mode if the data transfer is stalled. Active read or write current will resume once the data transfer is restarted with a toggling clock. The Active Clock Stop mode must not be used in violation of the $t_{CSM}$ limit. CS# must go high before $t_{CSM}$ is violated.

#### 6.1.3 Deep Power Down

In the Deep Power Down (DPD) mode, current consumption is driven to the lowest possible level ($i_{DPD}$). DPD mode is entered by writing a 0 to CR0[15]. The device reduces power within $t_{DPDIN}$ time and all refresh operations stop. The data in Memory Space is lost, (becomes invalid without refresh) during DPD mode. The next access to the device, driving CS# Low then High, will cause the device to exit DPD mode. A read or write transaction used to drive CS# Low then High to exit DPD mode is a dummy transaction that is ignored by the device. Also, POR, or a hardware reset will cause the device to exit DPD mode. Only the CS# and RESET# signals are monitored during DPD mode. Returning to Standby mode following a dummy transaction or reset requires $t_{DPDOUT}$ time. Returning to Standby mode following a POR requires $t_{VCS}$ time, as with any other POR. Following the exit from DPD due to any of these events, the device is in the same state as following POR.

**Table 6.1** Deep Power Down Timing Parameters

| Parameter | Description | Min | Max | Unit |
|---|---|---|---|---|
| $t_{DPDIN}$ | Deep Power Down CR0[15]=0 register write to DPD power level | 10 | - | µs |
| $t_{DPDCSL}$ | Length of CS# Low period to cause an exit from Deep Power Down | 200 | - | ns |
| $t_{DPDOUT}$ | CS# Low then High to Standby wakeup time | - | 150 | µs |

**Figure 6.1** Deep Power Down Entry Timing

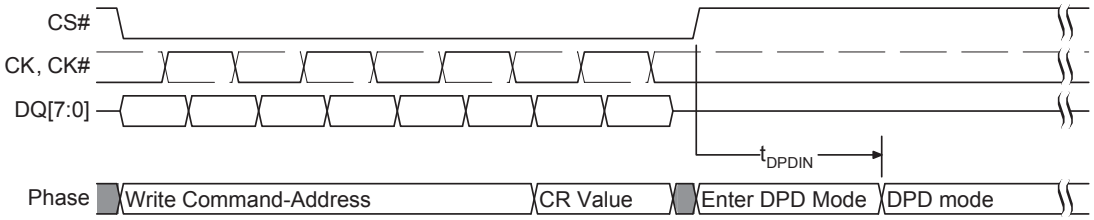
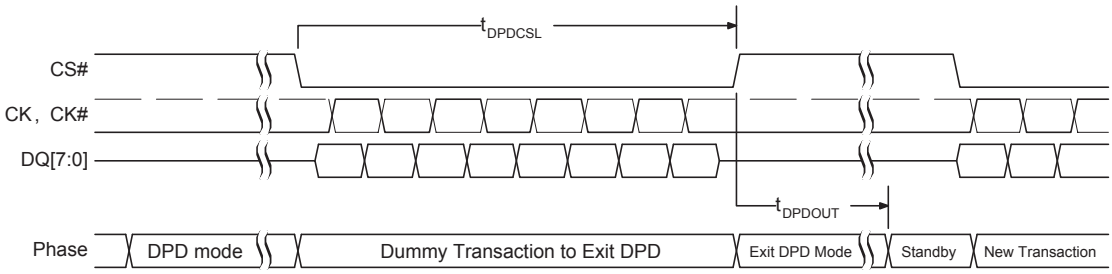



**Figure 6.2** Deep Power Down CS# Exit Timing



# 7. HyperRAM Connection Descriptions

Following is the description for other package connectors.

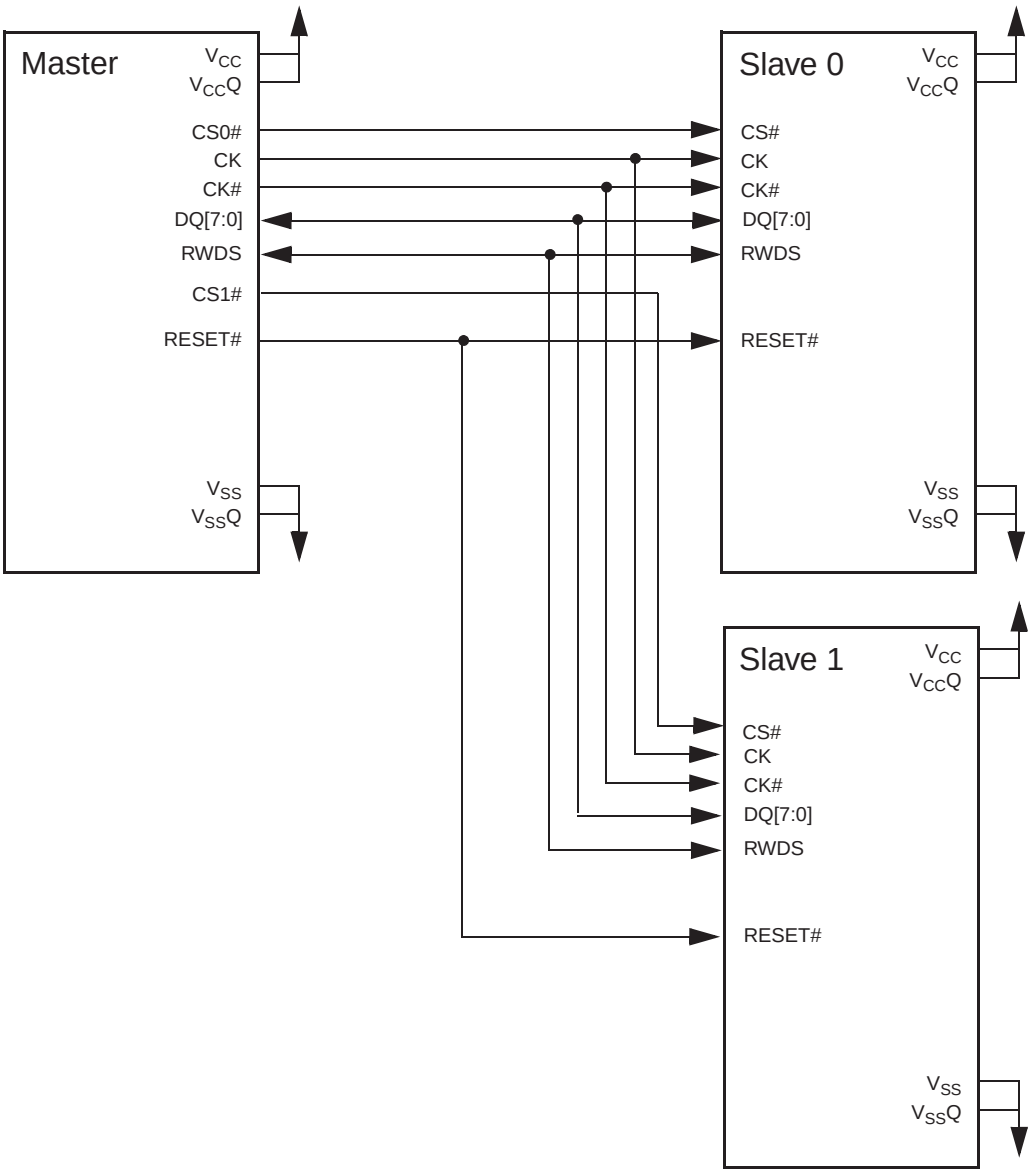
## 7.1 Other Connectors Summary

**Table 7.1** Other Connectors Summary

| Symbol | Type | Description |
|---|---|---|
| $V_{CC}$ | Power Supply | **Core Power.** |
| $V_{CC}Q$ | Power Supply | **Input/Output Power.** |
| $V_{SS}$ | Power Supply | **Core Ground.** |
| $V_{SS}Q$ | Power Supply | **Input/Output Ground.** |
| NC | No Connect | **Not Connected** internally. The signal/ball location may be used in Printed Circuit Board (PCB) as part of a routing channel. |
| RFU | No Connect | **Reserved for Future Use**. May or may not be connected internally, the signal/ball location should be left unconnected and unused by PCB routing channel for future compatibility. The signal/ball may be used by a signal in the future. |
| DNU | Reserved | **Do Not Use**. Reserved for use by ISSI. The signal/ball is connected internally. The signal/ball must be left open on the PCB. |

## 7.2 HyperRAM Block Diagram

**Figure 7.1** HyperRAM Connections, Including Optional Signals



# 8. Interface States

The Interface States table describes the required value of each signal for each interface state.

**Table 8.1** Interface States

| Interface State | $V_{CC}$ / $V_{CC}Q$ | CS# | CK, CK# | D7-D0 | RWDS | RESET# |
|---|---|---|---|---|---|---|
| Power-On (Cold) Reset | ≥ $V_{CC}$ / $V_{CC}Q$ min | X | X | High-Z | High-Z | X |
| Hardware (Warm) Reset | ≥ $V_{CC}$ / $V_{CC}Q$ min | X | X | High-Z | High-Z | L |
| Interface Standby | ≥ $V_{CC}$ / $V_{CC}Q$ min | H | X | High-Z | High-Z | H |
| Command-Address | ≥ $V_{CC}$ / $V_{CC}Q$ min | L | T | Master Output Valid | X | H |
| Read Initial Access Latency (data bus turn around period) | ≥ $V_{CC}$ / $V_{CC}Q$ min | L | T | High-Z | L | H |
| Write Initial Access Latency (RWDS turn around period) | ≥ $V_{CC}$ / $V_{CC}Q$ min | L | T | High-Z | High-Z | H |
| Read data transfer | ≥ $V_{CC}$ / $V_{CC}Q$ min | L | T | Slave Output Valid | Slave Output Valid X or T | H |
| Write data transfer with Initial Latency | ≥ $V_{CC}$ / $V_{CC}Q$ min | L | T | Master Output Valid | Master Output Valid X or T | H |
| Write data transfer without Initial Latency (2) | ≥ $V_{CC}$ / $V_{CC}Q$ min | L | T | Master Output Valid | Slave Output L or High-Z | H |
| Active Clock Stop (3) | ≥ $V_{CC}$ / $V_{CC}Q$ min | L | Idle | Master or Slave Output Valid or High-Z | X | H |
| Deep Power Down(3) | ≥ $V_{CC}$ / $V_{CC}Q$ min | H | X or T | Slave Output High-Z | High-Z | H |

***Legend***

*L = $V_{IL}$*
*H = $V_{IH}$*
*X = either $V_{IL}$ or $V_{IH}$*
*L/H = rising edge*
*H/L = falling edge*
*T = Toggling during information transfer*
*Idle = CK is low and CK# is high.*
*Valid = all bus signals have stable L or H level*

***Notes:***

1. *When the RSTO# or INT# open-drain outputs are High-Z the pull-up resistance provided by the master or an external resistor will pull the signal to High.*
2. *Writes without initial latency (with zero initial latency), do not have a turn around period for RWDS. The HyperRAM device will always drive RWDS during the Command-Address period to indicate whether extended latency is required. Since master write data immediately follows the Command-Address period the HyperRAM device may continue to drive RWDS Low or may take RWDS to High-Z. The master must not drive RWDS during Writes with zero latency. Writes with zero latency do not use RWDS as a data mask function. All bytes of write data are written (full word writes).*
3. *Active Clock Stop is described in Section 6.1.2, Active Clock Stop on page 25. DPD is described in Section 6.1.3, Deep Power Down on page 25.*

# 9. Electrical Specifications

## 9.1 Absolute Maximum Ratings

| | |
|---|---|
| Storage Temperature Plastic Packages | –65 °C to +150 °C |
| Ambient Temperature with Power Applied | –65°C to +115 °C |
| Voltage with Respect to Ground | |
| All signals (1) | –0.5V to +($V_{CC}$ + 0.5V) |
| Output Short Circuit Current (2) | 100 mA |
| $V_{CC}$ | –0.5V to +4.0V |

**Notes:**

1. Minimum DC voltage on input or I/O signal is –1.0V. During voltage transitions, input or I/O signals may undershoot $V_{SS}$ to -1.0V for periods of up to 20 ns. See Figure 9.1. Maximum DC voltage on input or I/O signals is $V_{CC}$ +1.0V. During voltage transitions, input or I/O signals may overshoot to $V_{CC}$ +1.0V for periods up to 20 ns. See Figure 9.2.
2. No more than one output may be shorted to ground at a time. Duration of the short circuit should not be greater than one second.
3. Stresses above those listed under Absolute Maximum Ratings may cause permanent damage to the device. This is a stress rating only; functional operation of the device at these or any other conditions above those indicated in the operational sections of this data sheet is not implied. Exposure of the device to absolute maximum rating conditions for extended periods may affect device reliability.

### 9.1.1 Input Signal Overshoot

During DC conditions, input or I/O signals should remain equal to or between $V_{SS}$ and $V_{DD}$. During voltage transitions, inputs or I/Os may overshoot $V_{SS}$ to –1.0V or overshoot to $V_{DD}$ +1.0V, for periods up to 20 ns.

**Figure 9.1** Maximum Negative Overshoot Waveform

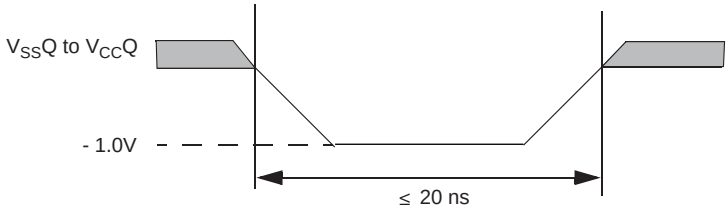
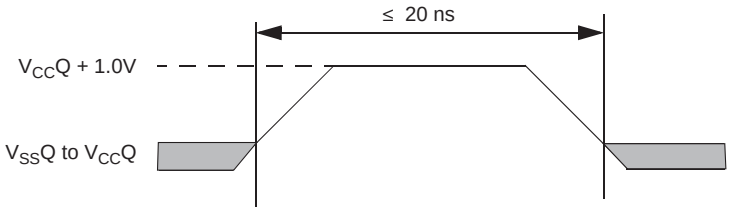



**Figure 9.2** Maximum Positive Overshoot Waveform



## 9.2 Latchup Characteristics

**Table 9.1** Latchup Specification

| Description | Min | Max | Unit |
|---|---|---|---|
| Input voltage with respect to $V_{SS}Q$ on all input only connections | – 1.0 | $V_{CCQ}$ + 1.0 | V |
| Input voltage with respect to $V_{SS}Q$ on all I/O connections | –1.0 | $V_{CCQ}$ + 1.0 | V |
| $V_{CCQ}$ Current | –100 | +100 | mA |

***Note:***

*1. Excludes power supplies $V_{CC}/V_{CCQ}$. Test conditions: $V_{CC} = V_{CCQ} = 1.8$ V, one connection at a time tested, connections not being tested are at $V_{SS}$.*

## 9.3 Operating Ranges

Operating ranges define those limits between which the functionality of the device is guaranteed.

### 9.3.1 Temperature Ranges

| Parameter | Symbol | Device | Spec | | Unit |
|---|---|---|---|---|---|
| | | | Min | Max | |
| Ambient Temperature | $T_A$ | Industrial (I) | –40 | 85 | °C |
| | | Extended (E) | –40 | 105 | |
| | | Automotive, Grade A1 | –40 | 85 | |
| | | Automotive, Grade A2 | –40 | 105 | |

### 9.3.2 Power Supply Voltages

| | |
|---|---|
| $V_{CC}$ and $V_{CC}Q$ | 1.7v to 1.95V |
| $V_{CC}$ and $V_{CC}Q$ | 2.7V to 3.6V |

## 9.4 DC Characteristics

**Table 9.2** DC Characteristics (CMOS Compatible)

| Parameter | Description | Test Conditions | Min | Typ (1) | Max | Unit |
|---|---|---|---|---|---|---|
| $I_{LI}$ | Input Leakage Current<br>3V Device Reset Signal High Only | $V_{IN} = V_{SS}$ to $V_{CC}$,<br>$V_{CC} = V_{CC}$ max | – | – | –0.1 | µA |
| $I_{LI}$ | Input Leakage Current<br>1.8V Device Reset Signal High Only | $V_{IN} = V_{SS}$ to $V_{CC}$,<br>$V_{CC} = V_{CC}$ max | – | – | –0.1 | µA |
| $I_{LI}$ | Input Leakage Current<br>3V Device Reset Signal Low Only (2) | $V_{IN} = V_{SS}$ to $V_{CC}$,<br>$V_{CC} = V_{CC}$ max | – | – | +20.0 | µA |
| $I_{LI}$ | Input Leakage Current<br>1.8V Device Reset Signal Low Only (2) | $V_{IN} = V_{SS}$ to $V_{CC}$,<br>$V_{CC} = V_{CC}$ max | – | – | +20.0 | µA |
| $I_{CC1}$ | $V_{CC}$ Active Read Current | CS# = $V_{IL}$, @166 MHz, $V_{CC}$ = 1.9V | –<br>– | 20 | 60 | mA |
| | | CS# = $V_{IL}$, @100 MHz, $V_{CC}$ = 3.6V | – | 20 | 35 | mA |
| $I_{CC2}$ | $V_{CC}$ Active Write Current | CS# = $V_{IL}$, @166 MHz, $V_{CC}$ = 1.9V | –<br>– | 15 | 60 | mA |
| | | CS# = $V_{IL}$, @100 MHz, $V_{CC}$ = 3.6V | – | 15 | 35 | mA |
| $I_{CC4I}$ | $V_{CC}$ Standby Current for Industrial<br>(–40 °C to +85 °C) | CS#, $V_{CC} = V_{CC}$ max, | – | 135 | 200 | µA |
| $I_{CC4IP}$ | $V_{CC}$ Standby Current for Industrial Plus<br>(–40 °C to +105 °C) | CS#, $V_{CC} = V_{CC}$ max | – | 135 | 300 | µA |
| $I_{CC5}$ | Reset Current | CS# = $V_{IH}$, RESET# = $V_{IL}$,<br>$V_{CC} = V_{CC}$ max | – | 10 | 20 | mA |
| $I_{CC6I}$ | Active Clock Stop Current for Industrial<br>(–40 °C to +85 °C) | CS# = $V_{IL}$, RESET# = $V_{IH}$,<br>$V_{CC} = V_{CC}$ max | – | 5.3 | 8 | mA |
| $I_{CC6IP}$ | Active Clock Stop Current for Industrial<br>Plus(–40 °C to +105 °C) | CS# = $V_{IL}$, RESET# = $V_{IH}$,<br>$V_{CC} = V_{CC}$ max | – | 5.3 | 12 | mA |
| $I_{CC7}$ | $V_{CC}$ Current during power up **(1)** | CS#, = H, $V_{CC} = V_{CC}$ max,<br>$V_{CC} = V_{CC}Q$ = *1.95V or 3.6V* (Note 9.4.1) | – | – | 35 | mA |
| $I_{DPD}$ | Deep Power Down Current 3V 85°C | CS#, $V_{CC}$ = 3.6V, $T_A$ = 85 °C | – | – | 20 | µA |
| $I_{DPD}$ | Deep Power Down Current 1.8V 85°C | CS#, $V_{CC}$ = 1.9V, $T_A$ = 85 °C | – | – | 10 | µA |
| $I_{DPD}$ | Deep Power Down Current 3V 105°C | CS#, $V_{CC}$ = 3.6V, $T_A$ = 105 °C | – | – | 40 | µA |
| $I_{DPD}$ | Deep Power Down Current 1.8V 105°C | CS#, $V_{CC}$ = 1.9V, $T_A$ = 105 °C | – | – | 20 | µA |
| $V_{IL}$ | Input Low Voltage | | -0.5 | - | 0.3 x $V_{CC}$ | V |
| $V_{IH}$ | Input High Voltage | | 0.7 x $V_{CC}$ | - | $V_{CC}$ + 0.3 | V |
| $V_{OL}$ | Output Low Voltage | $I_{OL}$ = 100 µA for DQ[7:0] | - | - | 0.15 x $V_{CC}$ | V |
| $V_{OH}$ | Output High Voltage | $I_{OH}$ = 100 µA for DQ[7:0] | - | 0.85 x $V_{CC}$ | - | V |

***Notes:***

1. *Not 100% tested.*
2. *RESET# low initiates exits from DPD mode and initiates the draw of $I_{CC5}$ reset current, making $I_{LI}$ during Reset# Low insignificant.*

### 9.4.1 Capacitance Characteristics

**Table 9.3** 1.8V Capacitive Characteristics

| Description | Parameter | Min | Max | Unit |
|---|---|---|---|---|
| Input Capacitance (CK, CK#, CS#) | CI | 3 | 4.5 | pF |
| Delta Input Capacitance (CK, CK#) | CID | – | 0.25 | pF |
| Output Capacitance (RWDS) | CO | 3 | 4 | pF |
| IO Capacitance (DQx) | CIO | 3 | 4 | pF |
| IO Capacitance Delta (DQx) | CIOD | – | 0.5 | pF |

***Notes:***

1. *These values are guaranteed by design and are tested on a sample basis only.*
2. *Contact capacitance is measured according to JEP147 procedure for measuring capacitance using a vector network analyzer. $V_{CC}$, $V_{CC}Q$ are applied and all other signals (except the signal under test) floating. DQ's should be in the high impedance state.*
3. *Note that the capacitance values for the CK, CK#, RWDS and DQx signals must have similar capacitance values to allow for signal propagation time matching in the system. The capacitance value for CS# is not as critical because there are no critical timings between CS# going active (Low) and data being presented on the DQs bus.*

**Table 9.4** 3.0V Capacitive Characteristics

| Description | Parameter | Min | Max | Unit |
|---|---|---|---|---|
| Input Capacitance (CK, CS#) | CI | 3 | 4.5 | pF |
| Output Capacitance (RWDS) | CO | 3 | 4 | pF |
| IO Capacitance (DQx) | CIO | 3 | 4 | pF |
| IO Capacitance Delta (DQx) | CIOD | – | 0.5 | pF |

***Notes:***

1. *These values are guaranteed by design and are tested on a sample basis only.*
2. *Contact capacitance is measured according to JEP147 procedure for measuring capacitance using a vector network analyzer. $V_{CC}$, $V_{CC}Q$ are applied and all other signals (except the signal under test) floating. DQ's should be in the high impedance state.*
3. *The capacitance values for the CK, RWDS and DQx signals must have similar capacitance values to allow for signal propagation time matching in the system. The capacitance value for CS# is not as critical because there are no critical timings between CS# going active (Low) and data being presented on the DQs bus.*

## 9.5 Power-Up Initialization

HyperRAM products include an on-chip voltage sensor used to launch the power-up initialization process. $V_{CC}$ and $V_{CC}Q$ must be applied simultaneously. When the power supply reaches a stable level at or above $V_{CC}$(min), the device will require $t_{VCS}$ time to complete its self-initialization process.

The device must not be selected during power-up. CS# must follow the voltage applied on $V_{CC}Q$ until $V_{CC}$ (min) is reached during power-up, and then CS# must remain high for a further delay of $t_{VCS}$. A simple pull-up resistor from $V_{CC}Q$ to Chip Select (CS#) can be used to insure safe and proper power-up.

If RESET# is Low during power up, the device delays start of the $t_{VCS}$ period until RESET# is High. The $t_{VCS}$ period is used primarily to perform refresh operations on the DRAM array to initialize it.

When initialization is complete, the device is ready for normal operation.

**Figure 9.3** Power-up with RESET# High

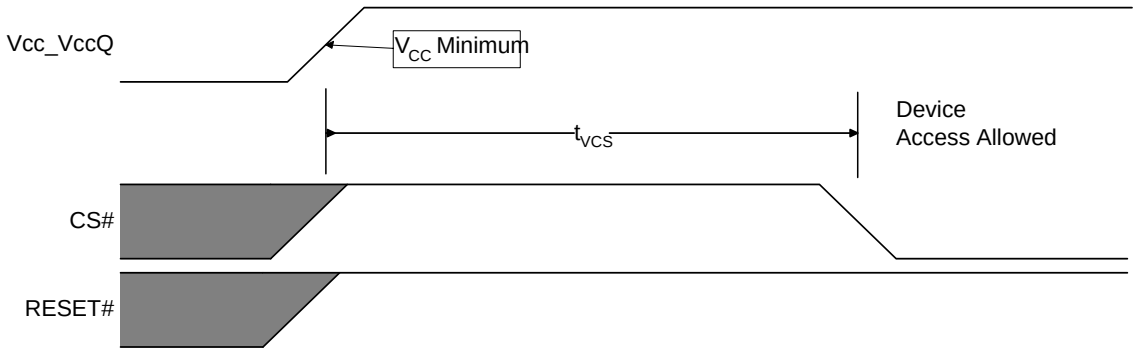



**Figure 9.4** Power-up with RESET# Low

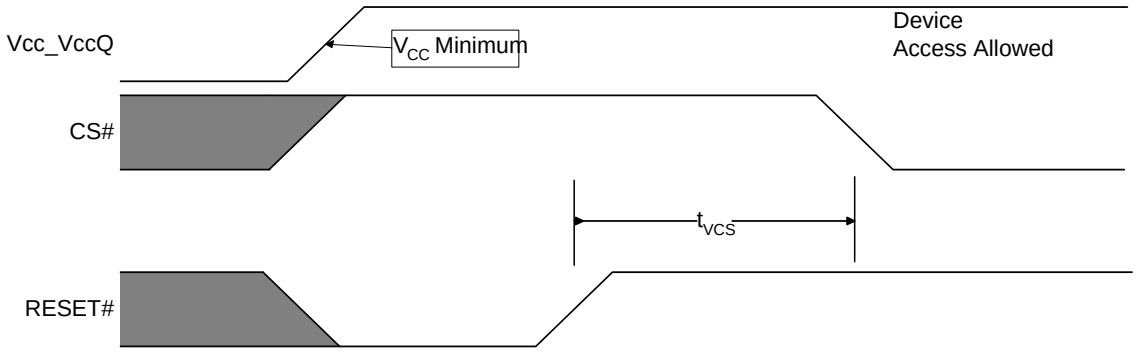



**Table 9.5** Power Up and Reset Parameters

| Parameter | Description | Min | Max | Unit |
|---|---|---|---|---|
| $V_{CC}$ | 1.8V $V_{CC}$ Power Supply | 1.7 | 1.95 | V |
| $V_{CC}$ | 3V $V_{CC}$ Power Supply | 2.7 | 3.6 | V |
| $t_{VCS}$ | $V_{CC}$ and $V_{CC}Q \geq$ minimum and RESET# High to first access | – | 150 | µs |

***Notes:***

1. *Bus transactions (read and write) are not allowed during the power-up reset time ($t_{VCS}$).*
2. *$V_{CC}Q$ must be the same voltage as $V_{CC}$.*
3. *$V_{CC}$ ramp rate may be non-linear.*

## 9.6 Power Down

HyperRAM devices are considered to be powered-off when the core power supply ($V_{CC}$) drops below the $V_{CC}$ Lock-Out voltage ($V_{LKO}$). During a power supply transition down to the VSS level, VCCQ should remain less than or equal to VCC.

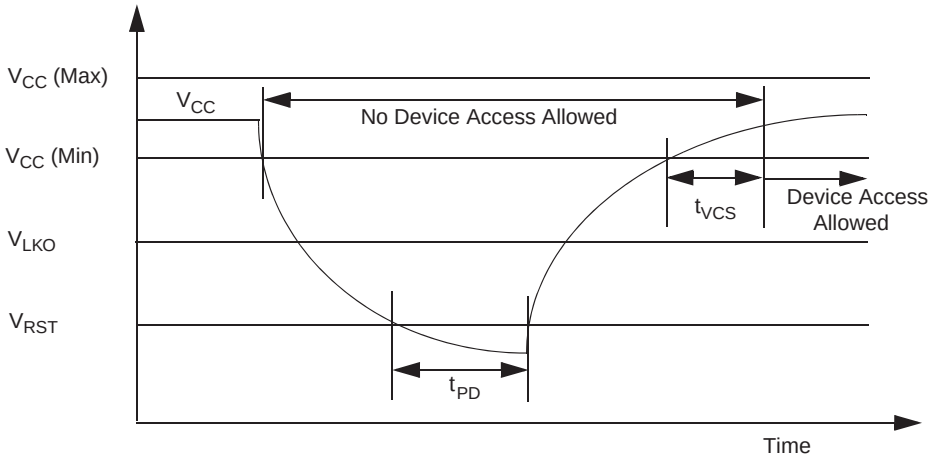
At the $V_{LKO}$ level, the HyperRAM device will have lost configuration or array data.

$V_{CC}$ must always be greater than or equal to $V_{CC}Q$ ($V_{CC} \geq V_{CC}Q$).

During Power-Down or voltage drops below $V_{LKO}$, the core power supply voltages must also drop below $V_{CC}$ Reset ($V_{RST}$) for a Power Down period ($t_{PD}$) for the part to initialize correctly when the power supply again rises to $V_{CC}$ minimum. See Figure 9.5.

If during a voltage drop the $V_{CC}$ stays above $V_{LKO}$ the part will stay initialized and will work correctly when $V_{CC}$ is again above $V_{CC}$ minimum. If $V_{CC}$ does not go below and remain below $V_{RST}$ for greater than $t_{PD}$, then there is no assurance that the POR process will be performed. In this case, a hardware reset will be required ensure the HyperBus device is properly initialized.

**Figure 9.5** Power Down or Voltage Drop




The following section describes HyperRAM device dependent aspects of power down specifications.

**Table 9.6** 1.8V Power-Down Voltage and Timing

| Symbol | Parameter | Min | Max | Unit |
|---|---|---|---|---|
| $V_{CC}$ | $V_{CC}$ Power Supply | 1.7 | 1.95 | V |
| $V_{LKO}$ | $V_{CC}$ Lock-out below which re-initialization is required | 1.7 | – | V |
| $V_{RST}$ | $V_{CC}$ Low Voltage needed to ensure initialization will occur | 0.8 | – | V |
| $t_{PD}$ | Duration of $V_{CC} \leq V_{RST}$ | 30 | – | µs |

***Note:***
*1. $V_{CC}$ ramp rate can be non-linear.*

**Table 9.7** 3.0V Power-Down Voltage and Timing

| Symbol | Parameter | Min | Max | Unit |
|---|---|---|---|---|
| $V_{CC}$ | $V_{CC}$ Power Supply | 2.7 | 3.6 | V |
| $V_{LKO}$ | $V_{CC}$ Lock-out below which re-initialization is required | 2.7 | – | V |
| $V_{RST}$ | $V_{CC}$ Low Voltage needed to ensure initialization will occur | 0.8 | – | V |
| $t_{PD}$ | Duration of $V_{CC} \leq V_{RST}$ | 50 | – | µs |

***Note:***
*1. $V_{CC}$ ramp rate can be non-linear.*

## 9.7 Hardware Reset

The RESET# input provides a hardware method of returning the device to the standby state.

During $t_{RPH}$ the device will draw $I_{CC5}$ current. If RESET# continues to be held Low beyond $t_{RPH}$, the device draws CMOS standby current ($I_{CC4}$). While RESET# is Low (during $t_{RP}$), and during $t_{RPH}$, bus transactions are not allowed.

A hardware reset will:

- cause the configuration registers to return to their default values,
- halt self-refresh operation while RESET# is low,
- and force the device to exit the Deep Power Down state.

After RESET# returns High, the self-refresh operation will resume. Because self-refresh operation is stopped during RESET# Low, and the self-refresh row counter is reset to its default value, some rows may not be refreshed within the required array refresh interval per Table 5.7, Array Refresh Interval per Temperature on page 23. This may result in the loss of DRAM array data during or immediately following a hardware reset. The host system should assume DRAM array data is lost after a hardware reset and reload any required data.

**Figure 9.6** Hardware Reset Timing Diagram

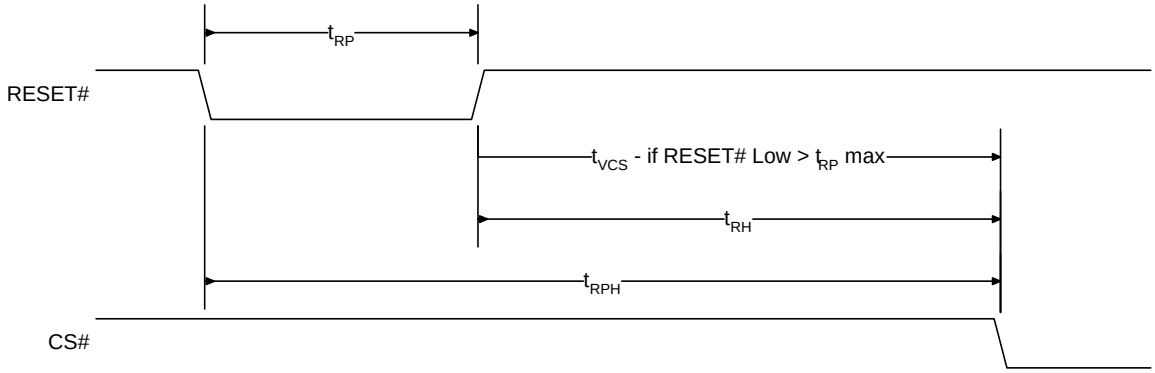


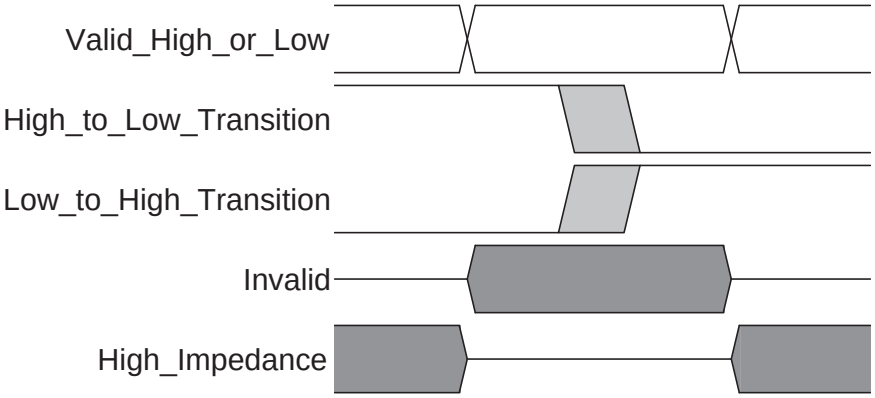

**Table 9.8** Power Up and Reset Parameters

| Parameter | Description | Min | Max | Unit |
|---|---|---|---|---|
| $t_{RP}$ | RESET# Pulse Width | 200 | – | ns |
| $t_{RH}$ | Time between RESET# (high) and CS# (low) | 200 | – | ns |
| $t_{RPH}$ | RESET# Low to CS# Low | 400 | – | ns |

# 10. Timing Specifications

The following section describes HyperRAM device dependent aspects of timing specifications.

## 10.1 Key to Switching Waveforms




## 10.2 AC Test Conditions

**Figure 10.1** Test Setup

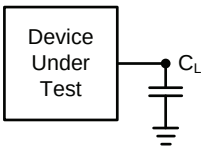


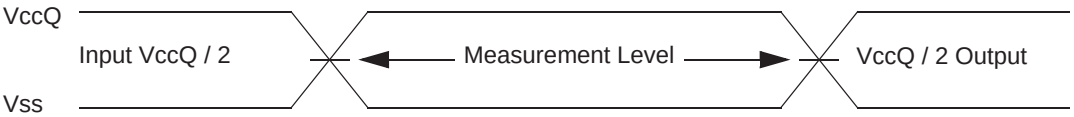

**Table 10.1** Test Specification

| Parameter | All Speeds | Units |
|---|---|---|
| Output Load Capacitance, $C_L$ | 20 | pF |
| Minimum Input Rise and Fall Slew Rates (Note 1) | 2.0 | V/ns |
| Input Pulse Levels | 0.0-$V_{CCQ}$ | V |
| Input timing measurement reference levels | $V_{CCQ}/2$ | V |
| Output timing measurement reference levels | $V_{CCQ}/2$ | V |

***Notes:***

1. *All AC timings assume an input slew rate of 2V/ns. CK/CK# differential slew rate of at least 4V/ns.*
2. *Input and output timing is referenced to $V_{CC}Q/2$ or to the crossing of CK/CK#.*

**Figure 10.2** Input Waveforms and Measurement Levels




***Note:***

1. *Input timings for the differential CK/CK# pair are measured from clock crossings.*

## 10.3 AC Characteristics

### 10.3.1 Read Transactions

**Table 10.2** HyperRAM Specific 1.8V Read Timing Parameters

| Parameter | Symbol | 166 MHz | | 133 MHz | | 100 MHz | | Unit |
|---|---|---|---|---|---|---|---|---|
| | | Min | Max | Min | | | Max | |
| Chip Select High Between Transactions | $t_{CSHI}$ | 6 | – | 7.5 | – | 10.0 | – | ns |
| HyperRAM Read-Write Recovery Time | $t_{RWR}$ | 36 | – | 37.5 | – | 40 | – | ns |
| Chip Select Setup to next CK Rising Edge | $t_{CSS}$ | 3 | – | | – | 3 | – | ns |
| Data Strobe Valid | $t_{DSV}$ | – | 12 | – | 12 | – | 12 | ns |
| Input Setup | $t_{IS}$ | 0.6 | – | 0.8 | – | 1.0 | – | ns |
| Input Hold | $t_{IH}$ | 0.6 | – | 0.8 | – | 1.0 | – | ns |
| HyperRAM Read Initial Access Time | $t_{ACC}$ | 36 | – | 37.5 | – | 40 | – | ns |
| Clock to DQs Low Z | $t_{DQLZ}$ | 0 | – | | – | 0 | – | ns |
| CK transition to DQ Valid | $t_{CKD}$ | 1 | 5.5 | 1 | 5.5 | 1 | 5.5 | ns |
| CK transition to DQ Invalid | $t_{CKDI}$ | 0 | 4.6 | 0 | 4.5 | 0 | 4.3 | ns |
| Data Valid ($t_{DV}$ *min = the lessor of:* $t_{CKHP}$ *min* - $t_{CKD}$ *max* + $t_{CKDI}$ *max*) or $t_{CKHP}$ *min* - $t_{CKD}$ *min* + $t_{CKDI}$ *min*) | $t_{DV}$ | 1.7 | – | 2.375 | – | 3.3 | – | ns |
| CK transition to RWDS valid | $t_{CKDS}$ | 1 | 5.5 | 1 | 5.5 | 1 | 5.5 | ns |
| RWDS transition to DQ Valid | $t_{DSS}$ | -0.45 | +0.45 | -0.6 | +0.6 | -0.8 | +0.8 | ns |
| RWDS transition to DQ Invalid | $t_{DSH}$ | -0.45 | +0.45 | -0.6 | +0.6 | -0.8 | +0.8 | ns |
| Chip Select Hold After CK Falling Edge | $t_{CSH}$ | 0 | – | | – | 0 | – | ns |
| Chip Select Inactive to RWDS High-Z | $t_{DSZ}$ | – | 6 | | 6 | – | 6 | ns |
| Chip Select Inactive to DQ High-Z | $t_{OZ}$ | – | 6 | | 6 | – | 6 | ns |
| HyperRAM Chip Select Maximum Low Time - Industrial Temperature | $t_{CSM}$ | - | 4.0 | - | 4.0 | - | 4.0 | us |
| HyperRAM Chip Select Maximum Low Time - Extended Temperature | | - | 1.0 | - | 1.0 | - | 1.0 | us |
| Refresh Time | $t_{RFH}$ | 36 | – | 37.5 | – | 40 | – | ns |

**Table 10.3** HyperRAM Specific 3.0V Read Timing Parameters

| Parameter | Symbol | 100 MHz | | Unit |
|---|---|---|---|---|
| | | Min | Max | |
| Chip Select High Between Transactions | $t_{CSHI}$ | 10.0 | – | ns |
| HyperRAM Read-Write Recovery Time | $t_{RWR}$ | 40 | - | ns |
| Chip Select Setup to next CK Rising Edge | $t_{CSS}$ | 3 | – | ns |
| Data Strobe Valid | $t_{DSV}$ | – | 12 | ns |
| Input Setup | $t_{IS}$ | 1.0 | – | ns |
| Input Hold | $t_{IH}$ | 1.0 | – | ns |
| HyperRAM Read Initial Access Time | $t_{ACC}$ | 40 | – | ns |
| Clock to DQs Low Z | $t_{DQLZ}$ | 0 | – | ns |
| HyperRAM CK transition to DQ Valid | $t_{CKD}$ | 1 | 7 | ns |
| HyperRAM CK transition to DQ Invalid | $t_{CKDI}$ | 0.5 | 5.2 | ns |
| Data Valid ($t_{DV}$ *min = the lessor of:* $t_{CKHP}$ *min* - $t_{CKD}$ *max* + $t_{CKDI}$ *max*) or $t_{CKHP}$ *min* - $t_{CKD}$ *min* + $t_{CKDI}$ *min*) | $t_{DV}$ | 2.7 | | ns |
| CK transition to RWDS valid | $t_{CKDS}$ | 1 | 7 | ns |
| RWDS transition to DQ Valid | $t_{DSS}$ | -0.8 | +0.8 | ns |
| RWDS transition to DQ Invalid | $t_{DSH}$ | -0.8 | +0.8 | ns |
| Chip Select Hold After CK Falling Edge | $t_{CSH}$ | 0 | – | ns |
| Chip Select Inactive to RWDS High-Z | $t_{DSZ}$ | – | 7 | ns |
| Chip Select Inactive to DQ High-Z | $t_{OZ}$ | – | 7 | ns |
| HyperRAM Chip Select Maximum Low Time - Industrial Temperature | $t_{CSM}$ | - | 4.0 | us |
| HyperRAM Chip Select Maximum Low Time - Extended Temperature | | - | 1.0 | us |
| Refresh Time | $t_{RFH}$ | 40 | – | ns |

### 10.3.2 Write Transactions

**Table 10.4** 1.8V Write Timing Parameters

| Parameter | Symbol | 166 MHz | | 133 MHz | | 100 MHz | | Unit |
|---|---|---|---|---|---|---|---|---|
| | | Min | Max | Min | | | Max | |
| Read-Write Recovery Time | $t_{RWR}$ | 36 | – | 37.5 | – | 40 | – | ns |
| Access Time | $t_{ACC}$ | 36 | – | 37.5 | – | 40 | – | ns |
| Refresh Time | $t_{RFH}$ | 36 | – | 37.5 | – | 40 | – | ns |
| Chip Select Maximum Low Time – Industrial Temperature | $t_{CSM}$ | – | 4.0 | – | 4.0 | – | 4.0 | µs |
| Chip Select Maximum Low Time – Industrial Plus Temperature | | – | 1.0 | – | 1.0 | – | 1.0 | µs |

**Table 10.5** 3.0V Write Timing Parameters

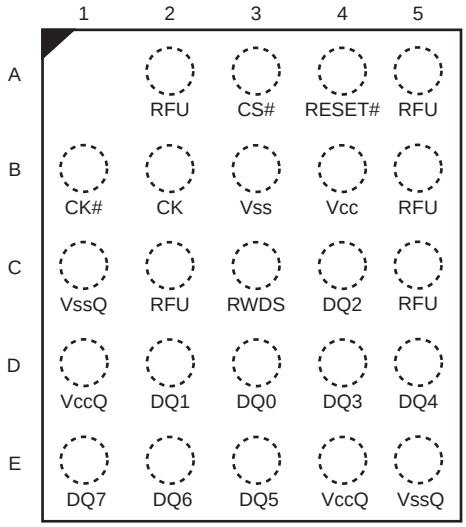
| Parameter | Symbol | 100 MHz | | Unit |
|---|---|---|---|---|
| | | Min | Max | |
| Read-Write Recovery Time | $t_{RWR}$ | 40 | – | ns |
| Access Time | $t_{ACC}$ | 40 | – | ns |
| Refresh Time | $t_{RFH}$ | 40 | – | ns |
| Chip Select Maximum Low Time – Industrial Temperature | $t_{CSM}$ | – | 4.0 | µs |
| Chip Select Maximum Low Time – Industrial Plus Temperature | | – | 1.0 | µs |

# 11. Physical Interface

## 11.1 FBGA 24-Ball 5 x 5 Array Footprint

HyperRAM devices are provided in Fortified Ball Grid Array (FBGA), 1 mm pitch, 24-ball, 5 x 5 ball array footprint, with 6mm x 8mm body.

**Figure 11.1** 24-Ball FBGA, 6 x 8 mm, 5x5 Ball Footprint, Top View




***Notes:***

1. *B1 is assigned to CK# on the 1.8V device.*
2. *B1 is a RFU on the 3.0V device.*

# 12. DDR Center Aligned Read Strobe Functionality

The HyperRAM device offers an optional feature that enables independent skewing (phase shifting) of the RWDS signal with respect to the read data outputs. This feature is provided in certain devices, based on the Ordering Part Number (OPN).

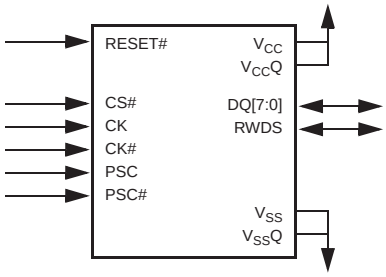
When the DDR Center Aligned Read Strobe (DCARS) feature is provided, a second differential Phase Shifted Clock input PSC/PSC# is used as the reference for RWDS edges instead of CK/CK#. The second clock is generally a copy of CK/CK# that is phase shifted 90 degrees to place the RWDS edges centered within the DQ signals valid data window. However, other degrees of phase shift between CK/CK# and PSC/PSC# may be used to optimize the position of RWDS edges within the DQ signals valid data window so that RWDS provides the desired amount of data setup and hold time in relation to RWDS edges.

PSC/PSC# is not used during a write transaction. PSC and PSC# may be driven Low and High respectively or, both may be driven Low during write transactions.

The PSC/PSC# differential clock is used only in HyperBus devices with 1.8 V nominal core and I/O voltage. HyperBus devices with 3 V nominal core and I/O voltage use only PSC as a single-ended clock.

## 12.1 HyperRAM Products with DCARS Signal Descriptions

**Figure 12.1** HyperBus Product with DCARS Signal Diagram

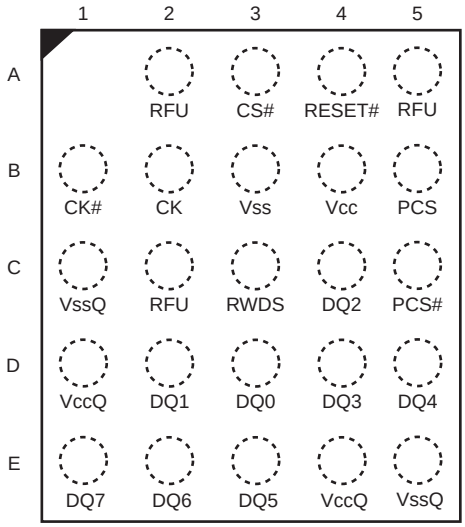



**Signal Descriptions**

| Symbol | Type | Description |
|---|---|---|
| CS# | Input | **Chip Select.** HyperBus transactions are initiated with a High to Low transition. HyperBus transactions are terminated with a Low to High transition. |
| CK, CK# | Input | **Differential Clock.** Command-Address/Data information is input or output with respect to the crossing of the CK and CK# signals. CK# is not used on the 3.0 V device, only a single ended CK is used. |
| PSC, PSC# | Input | **Phase Shifted Clock.** PSC/PSC# allows independent skewing of the RWDS signal with respect to the CK/CK# inputs. PSC# is only used on the 1.8 V device. PSC (and PSC#) may be driven High and Low respectively or both may be driven Low during write transactions. |
| RWDS | Output | **Read-Write Data Strobe.** Data bytes output during read transactions are aligned with RWDS based on the phase shift from CK, CK# to PSC, PSC#. PSC, PSC# cause the transitions of RWDS, thus the phase shift from CK, CK# to PSC, PSC# is used to place RWDS edges within the data valid window. RWDS is an input during write transactions to function as a data mask. At the beginning of all bus transactions RWDS is an output and indicates whether additional initial latency count is required (1 = additional latency count, 0 = no additional latency count). |
| DQ[7:0] | Input/Output | **Data Input/Output.** Command-Address/Data information is transferred on these DQs during Read and Write transactions. |
| RESET# | Input | **Hardware RESET**. When Low the device will self initialize and return to the idle state. RWDS and DQ[7:0] are placed into the High-Z state when RESET# is Low. RESET# includes a weak pull-up, if RESET# is left unconnected it will be pulled up to the High state. |
| $V_{CC}$ | Power Supply | **Core Power.** |
| $V_{CC}Q$ | Power Supply | **Input/Output Power.** |
| $V_{SS}$ | Power Supply | **Core Ground.** |
| $V_{SS}Q$ | Power Supply | **Input/Output Ground.** |

## 12.2 HyperRAM Products with DCARS — FBGA 24-ball, 5x5 Array Footprint

**Figure 12.2** 24-ball FBGA, 5x5 Ball Footprint, Top View



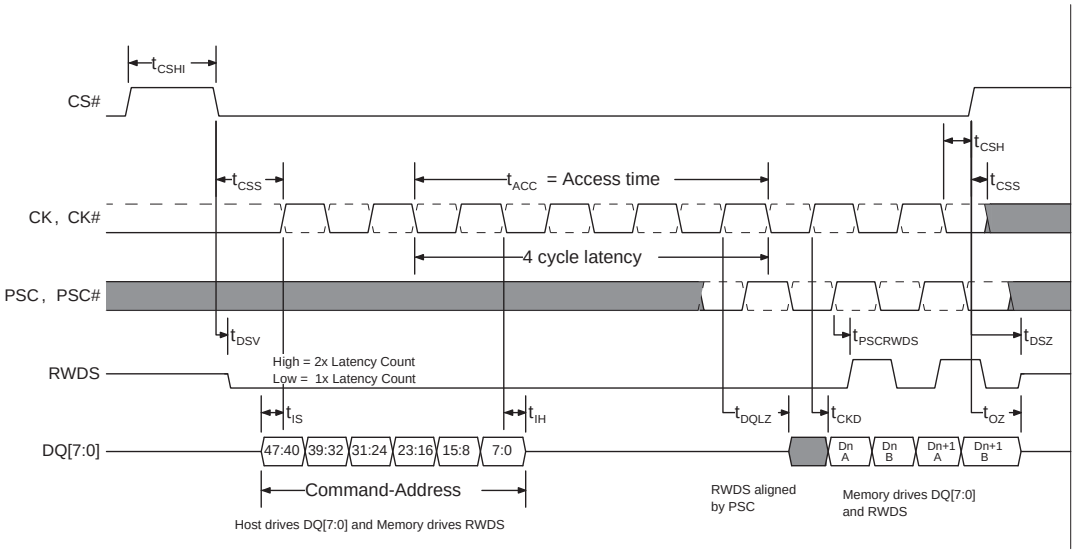

***Notes:***

1. *B1 is an RFU on the 3.0 V device and is assigned to CK# on the 1.8 V device.*
2. *C5 is an RFU on the 3.0 V device and is assigned to PSC# on the 1.8 V device.*

## 12.3 HyperRAM Memory with DCARS Timing

The HyperRAM device read transaction timing parameters vary by device, voltage range, and temperature range. Some timing parameters are not shown in this supplement documentation. Refer to the individual device data sheets for all current timing parameter information. The illustrations and parameters in this document are only those needed to define the DCARS feature and show the relationship between the Phase Shifted Clock, RWDS, and data.

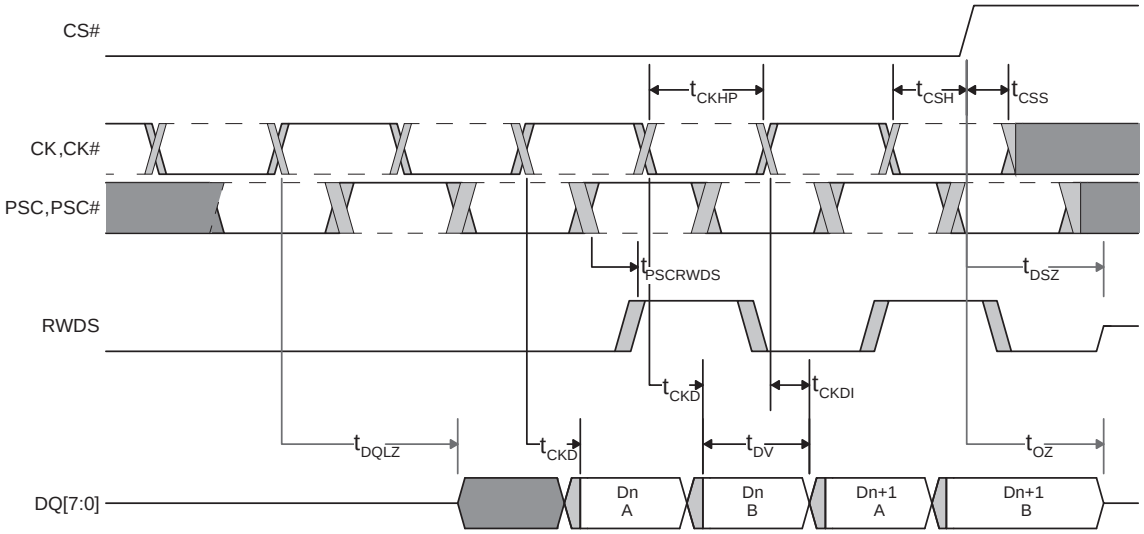
**Figure 12.3** HyperRAM Memory DCARS Timing Diagram




***Notes:***

1. *Transactions must be initiated with CK = Low and CK# = High. CS# must return High before a new transaction is initiated.*
2. *CK# and PSC# are only used on the 1.8 V device. The 3 V device uses a single ended CK and PSC input.*
3. *The memory drives RWDS during read transactions.*
4. *This example demonstrates a latency code setting of four clocks and no additional initial latency required.*

**Figure 12.4** DCARS Data Valid Timing




RWDS and Data are driven by the memory

***Notes:***

1. *This figure shows a closer view of the data transfer portion of Figure 12.1, HyperBus Product with DCARS Signal Diagram on page 43 in order to more clearly show the Data Valid period as affected by clock jitter and clock to output delay uncertainty.*
2. *CK# and PSC# are only used on the 1.8 V device. The 3 V device uses a single ended CK and PSC input.*
3. *The delay (phase shift) from CK to PSC is controlled by the HyperBus master interface (Host) and is generally between 40 and 140 degrees in order to place the RWDS edge within the data valid window with sufficient set-up and hold time of data to RWDS. The requirements for data set-up and hold time to RWDS are determined by the HyperBus master interface design and are not addressed by the HyperBus slave timing parameters.*
4. *The HyperBus timing parameters of $t_{CKD}$, and $t_{CKDI}$ define the beginning and end position of the data valid period. The $t_{CKD}$ and $t_{CKDI}$ values track together (vary by the same ratio) because RWDS and Data are outputs from the same device under the same voltage and temperature conditions.*

### DCARS Read Timings (@ 3.0 V)

| Parameter | Symbol | 100 MHz | | Unit |
|---|---|---|---|---|
| | | Min | Max | |
| HyperRAM PSC transition to RWDS transition | $t_{PSCRWDS}$ | 1 | 7 | ns |
| Time delta between CK to DQ valid and PSC to RWDS | $t_{PSCRWDS}$ - $t_{CKD}$ | -1.0 | +0.5 | ns |

***Note:***

1. *Sampled, not 100% tested.*

### DCARS Read Timings (@ 1.8 V)

| Parameter | Symbol | 133 MHz | | 100 MHz | | Unit |
|---|---|---|---|---|---|---|
| | | Min | Max | Min | Max | |
| HyperRAM PSC transition to RWDS transition | $t_{PSCRWDS}$ | 1 | 5.5 | 1 | 5.5 | ns |
| Time delta between CK to DQ valid and PSC to RWDS | $t_{PSCRWDS}$ - $t_{CKD}$ | -1.0 | +0.5 | -1.0 | +0.5 | ns |

***Note:***

1. *Sampled, not 100% tested.*

# 13. ORDERING RULE & INFORMATION

## 13.1 ORDERING RULE

**PACKAGED**

IS66/67 WVH 8M8 x BLL – 166 B1 L I

**66:** for PSRAM
**67:** for Automotive PSRAM

**Den/Org**
8M8 = 64Mb/8M x8
4M8 = 32Mb/4M x8
16M8 = 128Mb/16M x8
32M8 = 256Mb/32M x8

**PSRAM Product Type**
H – Hyper RAM

**Die Rev**
**Blank –** current die rev

**VDD**
ALL = 1.8V
BLL = 3.0V

**166:** 166 MHZ (1.8V)
**100:** 100 MHZ (3.0V)

**Solder Type**
L = Pb Free

**Package Code**
B1 = 24 TFBGA(1.2mm Height)
B2 = 24 TFBGA with PSC/PSC# (1.2mm Height)
B3 = 24 VFBGA (1.0mm Height)

**Temperature Range**
I = Industrial (-40 °C to 85 °C)
A1 = Automotive (-40 °C to 85 °C)
A2= Automotive (-40 °C to 105 °C)

**Note:**
**Call Factory for VFBGA (B3 Package)**

**KGD**

– XX X X X XX - XX

**Packing**
**WB** = Wafer Boat*

**Back Grinding Thickness**
31=31mil

**Inking**
**U** = Un-inked/ Electronic Mapping File*

**Test Flow**
**G** = Known Good Die (KGD) at Level 2 temp -40 to +85C
**D1** = Known Good Die (KGD) at Level 3 temp (auto) -40 to +85C
**D2** = Known Good Die (KGD) at Level 3 temp (auto) -40 to +105C
**I** = Known Tested Die (KTD)

**Die Type**
**X** = Die (Lead free)

**Speed Grade**
**6 ns** for KGD 1.8V
**10 ns** for KGD 3.0V
**'blank'** for KTD

## 13.2 ORDERING INFORMATION

Industrial Temperature Range: (-40ºC to +85ºC)

| Config. | Frequency (MHz) | Order Part No. | Package |
|---|---|---|---|
| 8Mx8 | 166 | IS66WVH8M8ALL-166B1LI | 24-ball TFBGA , Pb Free, 5x5 Array |
| | | IS66WVH8M8ALL-166B2LI | 24-ball TFBGA , Pb Free, 5x5 Array, PCS/PCS# |
| | 133 | IS66WVH8M8ALL-133B1LI | 24-ball TFBGA, Pb Free, 5x5 Array |
| | | IS66WVH8M8ALL-133B2LI | 24-ball TFBGA, Pb Free, 5x5 Array, PCS/PCS# |

Automotive A1 Temperature Range: (-40ºC to +85ºC)

| Config. | Frequency (MHz) | Order Part No. | Package |
|---|---|---|---|
| 8Mx8 | 166 | IS67WVH8M8ALL-166B1LA1 | 24-ball TFBGA , Pb Free, 5x5 Array |
| | | IS67WVH8M8ALL-166B2LA1 | 24-ball TFBGA , Pb Free, 5x5 Array, PCS/PCS# |
| | 133 | IS67WVH8M8ALL-133B1LA1 | 24-ball TFBGA, Pb Free, 5x5 Array |
| | | IS67WVH8M8ALL-133B2LA1 | 24-ball TFBGA, Pb Free, 5x5 Array, PCS/PCS# |

Automotive A2 Temperature Range: (-40ºC to +105ºC)

| Config. | Frequency (MHz) | Order Part No. | Package |
|---|---|---|---|
| 8Mx8 | 166 | IS67WVH8M8ALL-166B1LA2 | 24-ball TFBGA , Pb Free, 5x5 Array |
| | | IS67WVH8M8ALL-166B2LA2 | 24-ball TFBGA , Pb Free, 5x5 Array, PCS/PCS# |
| | 133 | IS67WVH8M8ALL-133B1LA2 | 24-ball TFBGA, Pb Free, 5x5 Array |
| | | IS67WVH8M8ALL-133B2LA2 | 24-ball TFBGA, Pb Free, 5x5 Array, PCS/PCS# |

**Note:**

**Call Factory for VFBGA (Height=1.0mm, B3 Package)**

**Ordering Information :**
**BLL : VDD 2.7V~3.6V, VDDQ 2.7V~3.6V**

**Industrial Temperature Range: (-40°C to +85°C)**

| Config. | Frequency (MHz) | Order Part No. | Package |
|---|---|---|---|
| 8Mx8 | 100 | IS66WVH8M8BLL-100B1LI | 24-ball TFBGA, Pb Free, 5x5 Array |
| | | IS66WVH8M8BLL-100B2LI | 24-ball TFBGA, Pb Free, 5x5 Array, PSC/PSC# |

**Automotive A1 Temperature Range: (-40°C to +85°C)**

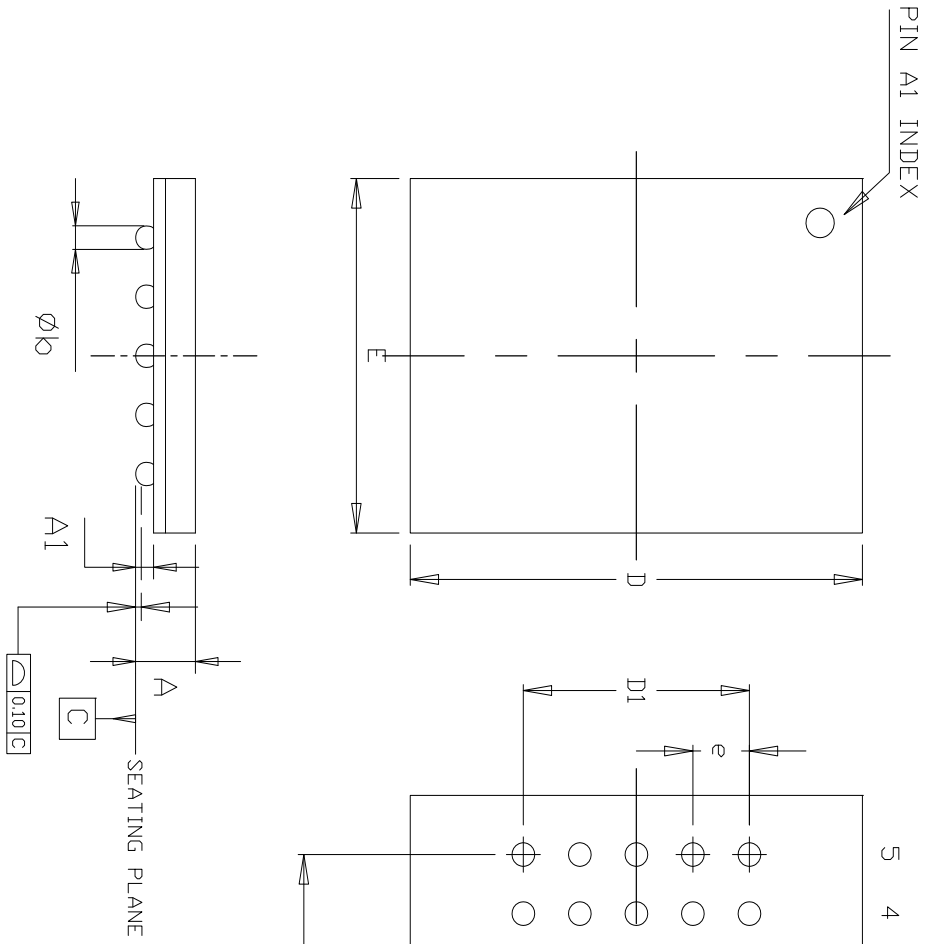
| Config. | Frequency (MHz) | Order Part No. | Package |
|---|---|---|---|
| 8Mx8 | 100 | IS67WVH8M8BLL-100B1LA1 | 24-ball TFBGA, Pb Free, 5x5 Array |
| | | IS67WVH8M8BLL-100B2LA1 | 24-ball TFBGA, Pb Free, 5x5 Array, PSC/PSC# |

**Automotive A2 Temperature Range: (-40°C to +105°C)**

| Config. | Frequency (MHz) | Order Part No. | Package |
|---|---|---|---|
| 8Mx8 | 100 | IS67WVH8M8BLL-100B1LA2 | 24-ball TFBGA, Pb Free, 5x5 Array |
| | | IS67WVH8M8BLL-100B2LA2 | 24-ball TFBGA, Pb Free, 5x5 Array, PSC/PSC# |

**Note:**
**Call Factory for VFBGA (Height=1.0mm, B3 Package)**

## 14. PACKAGE INFORMATION




| DIMENSION (mm) | | | |
|---|---|---|---|
| SYM. | MIN. | NOM. | MAX. |
| A | --- | --- | 1.20 |
| A1 | 0.26 | 0.31 | 0.36 |
| b | 0.35 | 0.40 | 0.45 |
| D | 7.90 | 8.00 | 8.10 |
| D1 | 4.00 BSC | | |
| E | 5.90 | 6.00 | 6.10 |
| E1 | 4.00 BSC | | |
| e | 1.00 BSC | | |

NOTE :

1. Controlling dimension : mm
2. Reference document : JEDEC MO-207

| TITLE | 24B 6x8x1.2mm TF-BGA Package Outline | REV. | A | DATE | 11/20/2014 |
|---|---|---|---|---|---|