CYNSE70128 Network Search Engine

Cypress Semiconductor Corporation • 3901 North First Street • San Jose, CA 95134 • 408-943-2600
Document #: 38-02040 Rev. *F Revised January 28, 2005

CONTENTS

1.0 OVERVIEW 9
2.0 FEATURES 9
3.0 FUNCTIONAL DESCRIPTION 10
3.1 Command Bus and DQ Bus 10
3.2 Database Entry (Data Array and Mask Array) 10
3.3 Arbitration Logic 11
3.4 Pipeline and SRAM Control 11
3.5 Full Logic 11
4.0 SIGNAL DESCRIPTIONS 11
5.0 CLOCKS 13
6.0 PHASE-LOCK LOOP USAGE 15
7.0 REGISTERS 15
7.1 Comparand Registers 15
7.2 Mask Registers 16
7.3 Search Successful Registers (SSR[0:7]) 16
7.4 Command Register 17
7.5 Information Register 18
7.6 Read Burst Address Register 18
7.7 Write Burst Address Register Description 19
7.8 NFA Register 19
8.0 NSE ARCHITECTURE AND OPERATION OVERVIEW 19
9.0 DATA AND MASK ADDRESSING 21
10.0 COMMANDS 21
10.1 Command Codes 21
10.2 Commands and Command Parameters 22
10.3 Read Command 22
10.4 Write Command 25
10.5 Parallel Write 27
10.6 Search Command 27
10.6.1 72-bit Search on Tables Configured as ×72 Using a Single CYNSE70128 Device 27
10.6.2 72-bit Search on Tables Configured as ×72 Using up to Eight CYNSE70128 Devices 30
10.6.3 72-bit Search on Tables Configured as x72 Using up to 31 CYNSE70128 Devices 36
10.6.4 144-bit Search on Tables Configured as x144 Using a Single CYNSE70128 Device 51
10.6.5 144-bit Search on Tables Configured as x144 Using up to Eight CYNSE70128 Devices 53
10.6.6 144-bit Search on Tables Configured as x144 Using up to 31 CYNSE70128 Devices 59
10.6.7 288-bit Search on Tables Configured as x288 Using a Single CYNSE70128 Device 74
10.6.8 288-bit Search on Tables Configured as x288 Using up to Eight CYNSE70128 Devices 76
10.6.9 288-bit Search on Tables Configured as x288 Using up to 31 CYNSE70128 Devices 82
10.6.10 Mixed-Size Searches on Tables Configured with Different Widths Using a CYNSE70128 with CFG_L LOW 97
10.6.11 Mixed-Size Searches on Tables Configured to Different Widths Using a CYNSE70128 with CFG_L HIGH 98
10.7 LRAM and LDEV Description 98
10.8 Learn Command 98

TABLE OF CONTENTS (continued)

11.0 DEPTH-CASCADING 102
11.1 Depth-Cascading up to Eight Devices (One Block) 102
11.2 Depth-Cascading up to 31 Devices (Four Blocks) 103
11.3 Depth-Cascading for a FULL Signal 104
12.0 SRAM ADDRESSING 105
12.1 Generating an SRAM BUS Address 106
12.2 SRAM PIO Access 106
12.3 SRAM Read with a Table of One Device 106
12.4 SRAM Read with a Table of up to Eight Devices 107
12.5 SRAM Read with a Table of up to 31 Devices 110
12.6 SRAM Write with a Table of One Device 113
12.7 SRAM Write with a Table of up to Eight Devices 114
12.8 SRAM Write with Table(s) of up to 31 Devices 117
12.9 Timing Sequences for Back-to-Back searches 120
13.0 POWER 122
13.1 Power-up Sequence 122
13.2 Power Consumption 123
14.0 APPLICATION 124
15.0 JTAG (1149.1) TESTING 124
16.0 ELECTRICAL SPECIFICATIONS 125
17.0 AC TIMING WAVEFORMS 126
17.1 Special Note for MULTI_HIT Function on the CYNSE70128 129
18.0 PINOUT DESCRIPTION 130
19.0 ORDERING INFORMATION 135
20.0 PACKAGE DIAGRAM 136

LIST OF FIGURES

Figure 5-1. CYNSE70128 Clocks (CLK2X and PHS_L) 13
Figure 5-2. CYNSE70128 Clocks (CLK1X) 13
Figure 5-3. CYNSE70128 Clocks for All Timing Diagrams 14
Figure 7-1. Comparand-Register Selection during Search and Learn Instructions 15
Figure 7-2. Addressing the Global Mask Register Array 16
Figure 8-1. CYNSE70128 Database Width Configuration 19
Figure 8-2. Multiwidth Database Configurations Example 21
Figure 9-1. Addressing the CYNSE70128 Data and Mask Arrays 21
Figure 10-1. Single-Location Read Cycle Timing 23
Figure 10-2. Burst Read of the Data and Mask Arrays (BLEN = 4) 24
Figure 10-3. Single Write Cycle Timing 25
Figure 10-4. Burst Write of the Data and Mask Arrays (BLEN = 4) 26
Figure 10-5. Timing Diagram for 72-bit Search in x72 Table (One Device) 28
Figure 10-6. Hardware Diagram for a Table with One Device 28
Figure 10-7. x72 Table with One Device 29
Figure 10-8. Hardware Diagram for a Table with Eight Devices 31
Figure 10-9. Timing Diagram for 72-bit Search Device Number 0 32
Figure 10-10. Timing Diagram for 72-bit Search Device Number 1 33
Figure 10-11. Timing Diagram for 72-bit Search Device Number 7 (Last Device) 34
Figure 10-12. x72 Table with Eight Devices 35
Figure 10-13. Hardware Diagram for a Table with 31 Devices 37
Figure 10-14. Hardware Diagram for a Block of up to Eight Devices 38
Figure 10-15. Timing Diagram for Each Device in Block Number 0 (Miss on Each Device) 39
Figure 10-16. Timing Diagram for Each Device Above the Winning Device in Block Number 1 40
Figure 10-17. Timing Diagram for Globally Winning Device in Block Number 1 41
Figure 10-18. Timing Diagram for Devices Below the Winning Device in Block Number 1 42
Figure 10-19. Timing Diagram for Devices Above the Winning Device in Block Number 2 43
Figure 10-20. Timing Diagram for Globally Winning Device in Block Number 2 44
Figure 10-21. Timing Diagram for Devices Below the Winning Device in Block Number 2 45
Figure 10-22. Timing Diagram for Devices Above the Winning Device in Block Number 3 46
Figure 10-23. Timing Diagram for Globally Winning Device in Block Number 3 47
Figure 10-24. Timing Diagram for Devices Below the Winning Device in Block Number 3 (Except the Last Device [Device 30]) 48
Figure 10-25. Timing Diagram for Device Number 6 in Block Number 3 (Device 30 in Depth-Cascaded Table) 49
Figure 10-26. x72 Table with 31 Devices 50
Figure 10-27. Timing Diagram for 144-bit Search (One Device) 51
Figure 10-28. Hardware Diagram for a Table With One Device 51
Figure 10-29. x144 Table with One Device 52
Figure 10-30. Hardware Diagram for a Table with Eight Devices 54
Figure 10-31. Timing Diagram for 144-bit Search Device Number 0 55
Figure 10-32. Timing Diagram for 144-bit Search Device Number 1 56
Figure 10-33. Timing Diagram for 144-bit Search Device Number 7 (Last Device) 57
Figure 10-34. x144 Table with Eight Devices 58
Figure 10-35. Hardware Diagram for a Table with 31 Devices 60
Figure 10-36. Hardware Diagram for a Block of up to Eight Devices 61
Figure 10-37. Timing Diagram for Each Device in Block Number 0 (Miss on Each Device) 62
Figure 10-38. Timing Diagram for Each Device Above the Winning Device in Block Number 1 63
Figure 10-39. Timing Diagram for Globally Winning Device in Block Number 1 64

LIST OF FIGURES (continued)

Figure 10-40. Timing Diagram for Devices Below the Winning Device in Block Number 1 65
Figure 10-41. Timing Diagram for Devices Above the Winning Device in Block Number 2 66
Figure 10-42. Timing Diagram for Globally Winning Device in Block Number 2 67
Figure 10-43. Timing Diagram for Devices Below the Winning Device in Block Number 2 68
Figure 10-44. Timing Diagram for Devices Above the Winning Device in Block Number 3 69
Figure 10-45. Timing Diagram for Globally Winning Device in Block Number 3 70
Figure 10-46. Timing Diagram for Devices Below the Winning Device in Block Number 3 Except Device 30 (the Last Device) 71
Figure 10-47. Timing Diagram for Device Number 6 in Block Number 3 (Device 30 in Depth-Cascaded Table) 72
Figure 10-48. x144 Table with 31 Devices 73
Figure 10-49. Timing Diagram for 288-bit Search (One Device) 74
Figure 10-50. Hardware Diagram for a Table with One Device 75
Figure 10-51. x288 Table with One Device 75
Figure 10-52. Hardware Diagram for a Table with Eight Devices 77
Figure 10-53. Timing Diagram for 288-bit Search Device Number 0 78
Figure 10-54. Timing Diagram for 288-bit Search Device Number 1 79
Figure 10-55. Timing Diagram for 288-bit Search Device Number 7 (Last Device) 80
Figure 10-56. x288 Table with Eight Devices 81
Figure 10-57. Hardware Diagram for a Table with 31 Devices 83
Figure 10-58. Hardware Diagram for a Block of up to Eight Devices 84
Figure 10-59. Timing Diagram for Each Device in Block Number 0 (Miss on Each Device) 85
Figure 10-60. Timing Diagram for Each Device Above the Winning Device in Block Number 1 86
Figure 10-61. Timing Diagram for Globally Winning Device in Block Number 1 87
Figure 10-62. Timing Diagram for Devices Below the Winning Device in Block Number 1 88
Figure 10-63. Timing Diagram for Devices Above the Winning Device in Block Number 2 89
Figure 10-64. Timing Diagram for Globally Winning Device in Block Number 2 90
Figure 10-65. Timing Diagram for Devices Below the Winning Device in Block Number 2 91
Figure 10-66. Timing Diagram for Devices Above the Winning Device in Block Number 3 92
Figure 10-67. Timing Diagram for Globally Winning Device in Block Number 3 93
Figure 10-68. Timing Diagram for Devices Below the Winning Device in Block Number 3 Except Device 30 (the Last Device) 94
Figure 10-69. Timing Diagram of the Last Device in Block Number 3 (Device 30 in the Table) 95
Figure 10-70. x288 Table with 31 Devices 96
Figure 10-71. Timing Diagram for Mixed Search (One Device) 97
Figure 10-72. Multiwidth Configurations Example 98
Figure 10-73. Timing Diagram of Learn (TLSZ = 00) 99
Figure 10-74. Timing Diagram of Learn (Except on the Last Device [TLSZ = 01]) 100
Figure 10-75. Timing Diagram of Learn on Device Number 7 (TLSZ = 01) 101
Figure 11-1. Depth-Cascading to Form a Single Block 103
Figure 11-2. Depth-Cascading Four Blocks 104
Figure 11-3. FULL Generation in a Cascaded Table 105
Figure 12-1. SRAM Read Access (TLSZ = 00, HLAT = 000, LRAM = 1, LDEV = 1) 107
Figure 12-2. Table of a Block of Eight Devices 108
Figure 12-3. SRAM Read Through Device Number 0 in a Block of Eight Devices 109
Figure 12-4. SRAM Read Timing for Device Number 7 in a Block of Eight Devices 110
Figure 12-5. Table of 31 Devices Made of Four Blocks 111
Figure 12-6. SRAM Read Through Device Number 0 in a Block of 31 Devices (Device Number 0 Timing) 112

LIST OF FIGURES (continued)

Figure 12-7. SRAM Read Through Device Number 0 in a Block of 31 Devices (Device Number 30 Timing) 113
Figure 12-8. SRAM Write Access (TLSZ = 00, HLAT = 000, LRAM = 1, LDEV = 1) 114
Figure 12-9. Table of a Block of Eight Devices 115
Figure 12-10. SRAM Write Through Device Number 0 in a Block of Eight Devices 116
Figure 12-11. SRAM Write Timing for Device Number 7 in a Block of Eight Devices 117
Figure 12-12. Table of 31 Devices (Four Blocks) 118
Figure 12-13. SRAM Write Through Device Number 0 in a Bank of 31 Devices (Device 0 Timing) 119
Figure 12-14. SRAM Write Through Device Number 0 in a Bank of 31 CYNSE70128 Devices (Device Number 30 Timing) 120
Figure 13-1. Power-up Sequence (CLK2x) 121
Figure 13-2. Power Consumption of CYNSE70128 122
Figure 14-1. Sample Switch/Router Using the CYNSE70128 Device 123
Figure 17-1. Input Wave Form for CYNSE70128 127
Figure 17-2. Output Load for CYNSE70128 127
Figure 17-3. I/O Output Load Equivalent for CYNSE70128 127
Figure 17-4. AC Timing Waveforms with CLK2X 128
Figure 17-5. AC Timing Waveforms with CLK1X 129
Figure 18-1. Pinout Diagram 130
Figure 20-1. 388-lead Ball Grid Array (35 x 35 x 2.33 mm) BG388 136

LIST OF TABLES

Table 4-1. CYNSE70128 Signal Description 11
Table 7-1. Register Overview 15
Table 7-2. Search Successful Register Description 16
Table 7-3. Command Register Description 17
Table 7-4. Information Register Description 18
Table 7-5. Read Burst Register Description 18
Table 7-6. Write Burst Register Description 19
Table 7-7. NFA Register 19
Table 8-1. Bit Position Match 20
Table 10-1. Command Codes 22
Table 10-2. Command Parameters 22
Table 10-3. Read Command Parameters 23
Table 10-4. Read Address Format for Data Array, Mask Array, or SRAM 23
Table 10-5. Read Address Format for Internal Registers 24
Table 10-6. Read Address Format for Data and Mask Arrays 25
Table 10-7. Write Address Format for Data Array, Mask Array or SRAM (Single Write) 26
Table 10-8. Write Address Format for Internal Registers 26
Table 10-9. Write Address Format for Data and Mask Array (Burst Write) 27
Table 10-10. The Latency of Search from Instruction to SRAM Access Cycle 29
Table 10-11. Shift OF SSF and SSV from SADR 29
Table 10-12. Hit/Miss Assumption 30
Table 10-13. The Latency of Search from Instruction to SRAM Access Cycle 35
Table 10-14. Shift OF SSF and SSV from SADR 35
Table 10-15. Hit/Miss Assumption 36
Table 10-16. The Latency of Search from Instruction to SRAM Access Cycle 50
Table 10-17. Shift of SSF and SSV from SADR 50
Table 10-18. The Latency of Search from Instruction to SRAM Access Cycle 52
Table 10-19. Shift OF SSF and SSV from SADR 52
Table 10-20. Hit/Miss Assumption 53
Table 10-21. Search Latency from Instruction to SRAM Access Cycle 58
Table 10-22. Shift OF SSF and SSV from SADR 58
Table 10-23. Hit/Miss Assumption 59
Table 10-24. The Latency of Search from Instruction to SRAM Access Cycle 73
Table 10-25. Shift OF SSF and SSV from SADR 73
Table 10-26. The Latency of Search from Cycles C and D to SRAM Access Cycle 76
Table 10-27. Shift OF SSF and SSV from SADR 76
Table 10-28. Hit/Miss Assumption 76
Table 10-29. The Latency of Search from Cycles C and D to SRAM Access Cycle 81
Table 10-30. Shift of SSF and SSV from SADR 81
Table 10-31. Hit/Miss Assumption 82
Table 10-32. The Latency of Search from Cycles C and D to SRAM Access Cycle 96
Table 10-33. Shift of SSF and SSV from SADR 96
Table 10-34. Searches with CFG_L Set High 98
Table 10-35. Latency of SRAM Write Cycle from Second Cycle of Learn Instruction 101
Table 12-1. SRAM Address 106
Table 12-2. Required Idle Cycles between Commands 121
Table 15-1. Supported Operations 124
Table 15-2. TAP Device ID Register 125
Table 16-1. DC Electrical Characteristics for CYNSE70128 125

LIST OF TABLES (continued)

Table 16-2. Operating Conditions for CYNSE70128 126
Table 17-1. AC Timing Parameters with CLK2X 126
Table 17-2. AC Timing Parameters with CLK1X 127
Table 17-3. 2.5V AC Table for Test Condition of CYNSE70128 127
Table 18-1. Pinout Descriptions for Pinout Diagram 132
Table 19-1. Ordering Information 135

1.0 Overview

Cypress Semiconductor Corporation's (Cypress's) CYNSE70128 network search engine (NSE) incorporates patent-pending Associative Processing Technology™ (APT) and is designed to be a high-performance, pipelined, synchronous, 64K-entry NSE. The CYNSE70128 database entry size can be 72 bits, 144 bits, or 288 bits. In the 72-bit entry mode, the size of the database is 64K entries. In the 144-bit mode, the size of the database is 32K entries, and in the 288-bit mode, the size of the database is 16K entries. The CYNSE70128 device is configurable to support multiple databases with different entry sizes. The 36-bit entry table can be implemented using the Global Mask Registers (GMRs) building-database size of 128K entries with a single device.

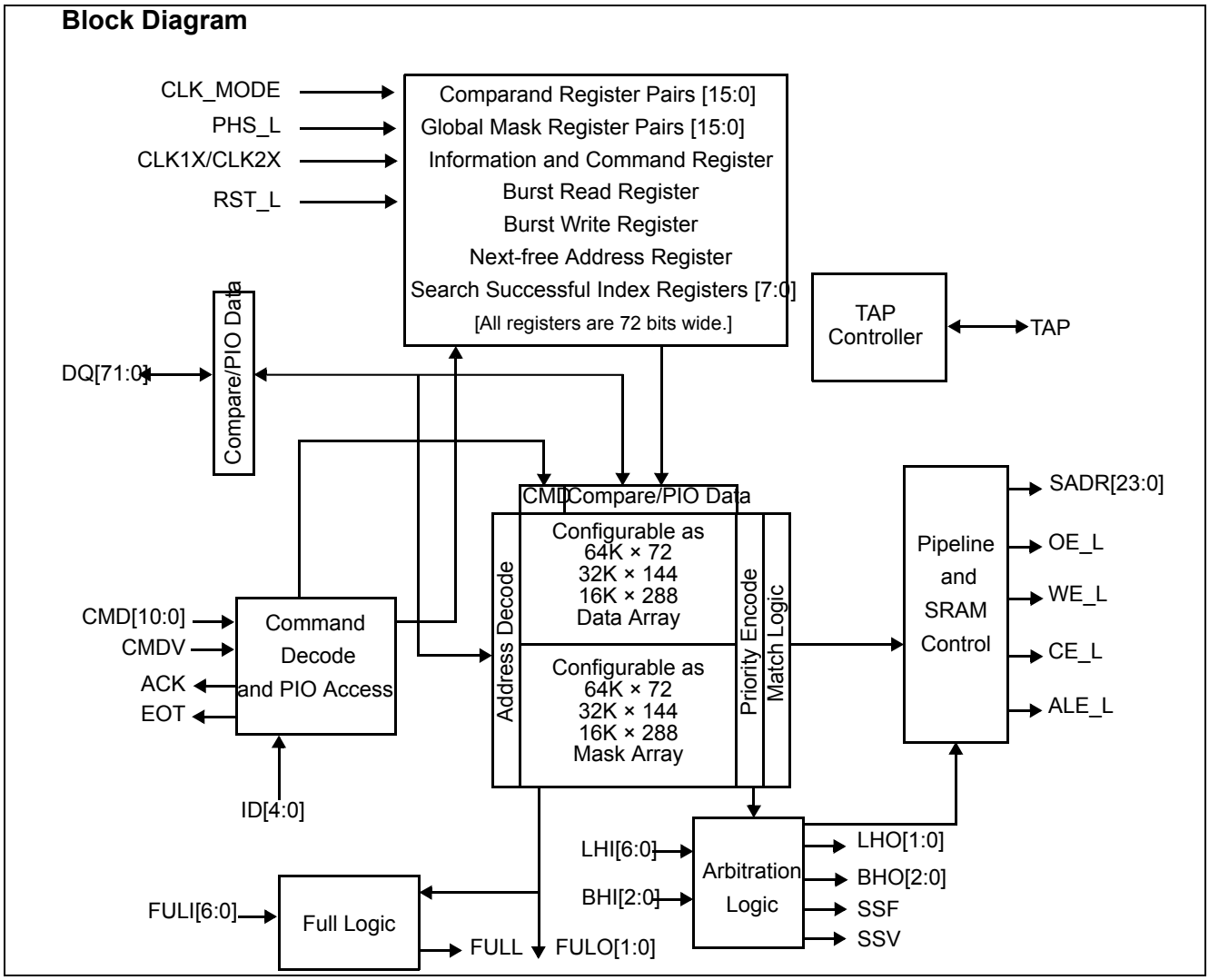
The NSE can sustain 100 million transactions per second when the database is programmed or configured as 72 or 144 bits. When the database is programmed to have an entry size of 36 or 288 bits, the NSE will perform at 50 million transactions per second. The CYNSE70128 can be used to accelerate network protocols such as Longest-prefix Match (CIDR), ARP, MPLS, and other layer 2, 3, and 4 protocols.

This high-speed, high-capacity NSE can be deployed in a variety of networking and communications applications. The performance and features of the CYNSE70128 make it attractive in applications such as Enterprise LAN switches and routers and broadband switching and/or routing equipment supporting multiple data rates at OC–48 and beyond. The NSE is designed to be scalable in order to support network database sizes to 3968K entries specifically for environments that require large network policy databases. The block diagram for the CYNSE70128 device is shown on page 10.

2.0 Features

- **128K 36-bit entries in a single device**
- **64K entries in 72-bit mode, 32K entries in 144-bit mode, 16K entries in 288-bit mode**
- **100 million transactions per second in 72- and 144-bit configurations**
- **50 million transactions in 36- and 288-bit configurations**
- **Searches any subfield in a single cycle**
- **Synchronous pipelined operation**
- **Up to 31 NSEs can be cascaded**
- **When cascaded, the database entries can range up to 3,968K 36-bit entries**
- **Multiple width tables in a single database bank**
- **Glueless interface to industry-standard SRAMs and/or SSRAMs**
- **Simple hardware instruction interface**
- **IEEE 1149.1 test access port**
- **1.5V core voltage supply up to 83 million searches per second (MSPS)/1.65V core voltage supply for search rates greater than 83 MSPS**
- **2.5/3.3V I/O voltage supply**
- **388-pin BGA package**.

Block Diagram



3.0 Functional Description

The following subsections contain command and DQ bus (command and databus), database entry, arbitration logic, pipeline and SRAM control, and full logic descriptions.

3.1 Command Bus and DQ Bus

CMD[10:0] carries the command and its associated parameter. DQ[71:0] is used for data transfer to and from the database entries, which comprise a data and a mask field that are organized as data and mask arrays. The DQ bus carries the search data during the Search command as well as the address and data during Read and/or Write operations. The DQ bus also carries the address information for the flow-through accesses to the external SRAMs and/or SSRAMs.

3.2 Database Entry (Data Array and Mask Array)

Each database entry comprises a data and a mask field. The resultant value of the entry is “1,” “0,” or “X (don’t care),” depending on the value in the data and mask bits. The on-chip priority encoder selects the first matching entry in the database that is nearest to location 0.

3.3 Arbitration Logic

When multiple NSEs are cascaded to create large databases, the data being searched is presented to all NSEs simultaneously in the cascaded system. If multiple matches occur within the cascaded devices, arbitration logic on the NSEs will enable the winning device (with a matching entry that is closest to address 0 of the cascaded database) to drive the SRAM bus.

3.4 Pipeline and SRAM Control

Pipeline latency is added to give enough time to a cascaded system's arbitration logic to determine the device that will drive the index of the matching entry on the SRAM bus. Pipeline logic adds latency to both the SRAM access cycles and the SSF and SSV signals to align them to the host ASIC receiving the associated data.

3.5 Full Logic

Bit[0] in each of the 72-bit entries has a special purpose for the Learn command (0 = empty, 1 = full). When all the data entries have bit[0] = 1, the database asserts the FULL flag, indicating that all the NSEs in the depth-cascaded array are full.

4.0 Signal Descriptions

Table 4-1 lists and describes all CYNSE70128 signals.

Table 4-1. CYNSE70128 Signal Description

Pin Name	Pin Type[1]	Pin Description
Clocks and Reset		
CLK_MODE	I	***Clock Mode:*** This signal allows the selection of clock (CLK[2]) input to the CLK1X/CLK2X pin. If the CLK_MODE pin is LOW, CLK2X must be supplied on that pin. PHS_L must also be supplied. If the CLK_MODE pin is HIGH, CLK1X must be supplied on the CLK2X/CLK1X pin, and the PHS_L signal is not required. When the CLK_mode is HIGH, PHS_L is unused and should be externally grounded.
CLK2X/CLK1X	I	***Master Clock:*** Depending on the CLK_MODE pin, either the CLK2X or the CLK1X must be supplied. CYNSE70128 samples control and data signals on both the edges of CLK1X (if CLK1X is supplied). CYNSE70128 samples all the data and control pins on the positive edge of CLK2X if the CLK2X and PHS_L signals are supplied. All signals are driven out of the device on the rising edge of CLK1X if CLK1X is supplied, and are driven on the rising edge of CLK2X (when PHS_L is low) if CLK2X is supplied.
PHS_L	I	***Phase:*** This signal runs at half the frequency of CLK2X and generates an internal clock from CLK2X. See "Clocks" on page 13.
RST_L	I	***Reset:*** Driving RST_L LOW initializes the device to a known state.
CFG_L	I	***Configuration:*** When CFG_L is LOW, CYNSE70128 will operate in backward compatibility mode with CYNSE70032 and CYNSE70064. When CFG_L is LOW, the CMD[10:9] should be externally grounded. With CFG_L LOW, the device will behave identically with CYNSE70032 and CYNSE70064, and the new feature added to CYNSE70128 will be disabled. When CFG_L is HIGH, the additional command CMD[10:9] can be used and the following additional features will be supported: 1. 16 pair of Global Masks are supported instead of eight; 2. Parallel Write to the data and mask arrays is supported (see "Parallel Write" on page 27); and 3. configuring tables of up to three different widths does not require table identification bits in the data array, thus saving two bits from each 72-bit entry.
Command and DQ Bus		
CMD[10:0]	I	***Command Bus:*** [1:0] specifies the command and [10:2] contains the command parameters. The descriptions of individual commands explains the details of the parameters. The encoding of commands based on the [1:0] field are: 00: PIO Read 01: PIO Write 10: Search 11: Learn.
CMDV	I	***Command Valid:*** This signal qualifies the CMD bus: 0: No command 1: Command.

Notes:
1. I = Input only, I/O = Input or Output, O = Output only, T = three-state output.
2. "CLK" is an internal clock signal.

Table 4-1. CYNSE70128 Signal Description (continued)

Pin Name	Pin Type[1]	Pin Description
DQ[71:0]	I/O	***Address/Data Bus:*** This signal carries the Read and Write address and data during register, data, and mask array operations. It carries the compare data during Search operations. It also carries the SRAM address during SRAM PIO accesses.
ACK[5]	T	***Read Acknowledge:*** This signal indicates that valid data is available on the DQ bus during register, data, and mask array Read operations, or that the data is available on the SRAM data bus during SRAM Read operations.
EOT[5]	T	***End of Transfer:*** This signal indicates the end of burst transfer to the data or mask array during Read or Write burst operations.
SSF	T	***Search Successful Flag:*** When asserted, this signal indicates that the device is the global winner in a Search operation.
SSV	T	***Search Successful Flag Valid:*** When asserted, this signal qualifies the SSF signal.
MULTI_HIT	O	***Multiple Hit Flag:*** When asserted, this signal indicates that there is more than one location having a match on this device.
HIGH_SPEED	I	***High Speed:*** When this signal is HIGH, the device will run up to 100 MHz and perform 100 MSPS. However, in this mode, a TLSZ value of 00 is not supported when only one device is used. The valid TLSZ values are shown in Command Register Description (*Table 7-3*). When the signal is LOW, the device will run up to 83 MHz and perform 83 MSPS.
SRAM Interface		
SADR[23:0]	T	***SRAM Address:*** This bus contains address lines to access off-chip SRAMs that contain associative data. See *Table 12-1* for the details of the generated SRAM address. In a database of multiple CYNSE70128s, each corresponding bit of SADR from all cascaded devices must be connected.
CE_L	T	***SRAM Chip Enable:*** This is the chip-enable control for external SRAMs. In a database of multiple CYNSE70128s, CE_L of all cascaded devices must be connected. This signal is then driven by only one of the devices.
WE_L	T	***SRAM Write Enable:*** This is the write-enable control for external SRAMs. In a database of multiple CYNSE70128s, WE_L of all cascaded devices must be connected together. This signal is then driven by only one of the devices.
OE_L	T	***SRAM Output Enable:*** This is the output-enable control for external SRAMs. Only the last device drives this signal (with the LRAM bit set).
ALE_L	T	***Address Latch Enable:*** When this signal is LOW, the addresses are valid on the SRAM address bus. In a database of multiple CYNSE70128s, the ALE_L of all cascaded devices must be connected. This signal is then driven by only one of the devices.
Cascade Interface		
LHI[6:0]	I	***Local Hit In:*** These pins depth-cascade the device to form a larger table. One signal of this bus is connected to the LHO[1] or LHO[0] of each of the upstream devices in a block. All unused LHI pins are connected to a logic 0. (For more information, see ”Depth-Cascading” on page 102.)
LHO[1:0]	O	***Local Hit Out:*** LHO[1] and LHO[0] are the same logical signal. Either the LHO[1] or the LHO[0] is connected to one input of the LHI bus of up to four downstream devices in a block of up to eight. (For more information see ”Depth-Cascading” on page 102.)
BHI[2:0]	I	***Block Hit In:*** Inputs from the previous block BHO[2:0] are tied to BHI[2:0] of the current device. In a four-block system, the last block can contain only seven devices because the identification code 11111 is used for broadcast access.
BHO[2:0]	O	***Block Hit Out:*** These outputs from the last device in a block are connected to the BHI[2:0] inputs of the devices in the downstream blocks.
FULI[6:0]	I	***Full In:*** Each signal in this bus is connected to FULO[0] or FULO[1] of an upstream device to generate the FULL flag for the depth-cascaded block.

Notes:
3. In the previous versions of this specification, this signal was called CLK_OUT.
4. In previous versions of this specification, this signal was called PLL_BYPASS.
5. ACK and EOT require a weak external pull-down such as 47 KΩ or 100 KΩ.

Table 4-1. CYNSE70128 Signal Description (continued)

Pin Name	Pin Type[1]	Pin Description
FULO[1:0]	O	***Full Out:*** FULO[1] and FULO[0] are the same logical signal. One of these two signals must be connected to the FULI of up to four downstream devices in a depth-cascaded table. Bit [0] in the data array indicates whether the entry is full (1) or empty (0).This signal is asserted if all bits in the data array are ones. (Refer to ”Depth-Cascading” on page 102 for information on how to generate the FULL flag.)
FULL	O	***Full Flag:*** When asserted, this signal indicates that the table of multiple depth-cascaded devices is full.
Device Identification		
ID[4:0]	I	***Device Identification:*** The binary-encoded device identification for a depth-cascaded system starts at 00000 and goes up to 11110. 11111 is reserved for a special broadcast address that selects all cascaded NSEs in the system. On a broadcast read-only, the device with the LDEV bit set to 1 responds.
Supplies		
V_{DD}	n/a	***Chip core supply:*** 1.5V. (1.65V for search rates greater than 83 msps.)
V_{DDQ}	n/a	***Chip I/O supply:*** 2.5V or 3.3V (CYNSE70128-XXX)
Test Access Port		
TDI	I	Test access port’s test data in.
TCK	I	Test access port’s test clock.
TDO	T	Test access port’s test data out.
TMS	I	Test access port’s test mode select.
TRST_L	I	Test access port’s reset.

5.0 Clocks[6]

If the CLK_MODE pin is LOW, CYNSE70128 receives the CLK2X and PHS_L signals. It uses the PHS_L signal to divide CLK2X and generate a CLK, as shown in *Figure 5-1*. The CYNSE70128 uses CLK2X and CLK for internal operations. Also noted on

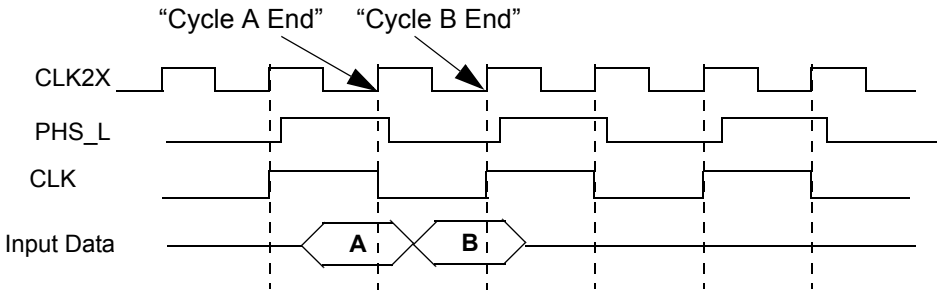


Figure 5-1. CYNSE70128 Clocks (CLK2X and PHS_L)

these figures are cycles A and B. Cycle A ends on the rising edge of CLK2X, when PHS_L is high. Cycle B ends on the falling edge of the CLK2X when PHS_L is low. Valid data for cycle A must be available for the NSE at the end of cycle A. Valid data for cycle B must be available for the NSE at the end of cycle B. PHS_L has setup and hold times requirements with respect to CLK2X. The setup and hold time requirements can be referred to in Sections 17.0 AC Timing Waveforms.

If the CLK_MODE pin is HIGH, CYNSE70128 receives the CLK1X only. CYNSE70128 uses an internal PLL to double the frequency of CLK1X and then divides that clock by two to generate a CLK for internal operations, as shown in *Figure 5-2*.[7]

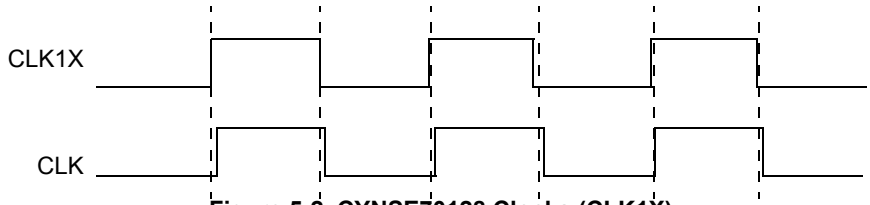


Figure 5-2. CYNSE70128 Clocks (CLK1X)

Notes:

6. Any reference to “CLK” cycles means one CLK cycle.
7. For the purpose of showing timing diagrams, all such diagrams in this document will be shown in CLK2X mode. For a timing diagram in CLK1X mode, the following substitution can be made (see *Figure 5-3*).

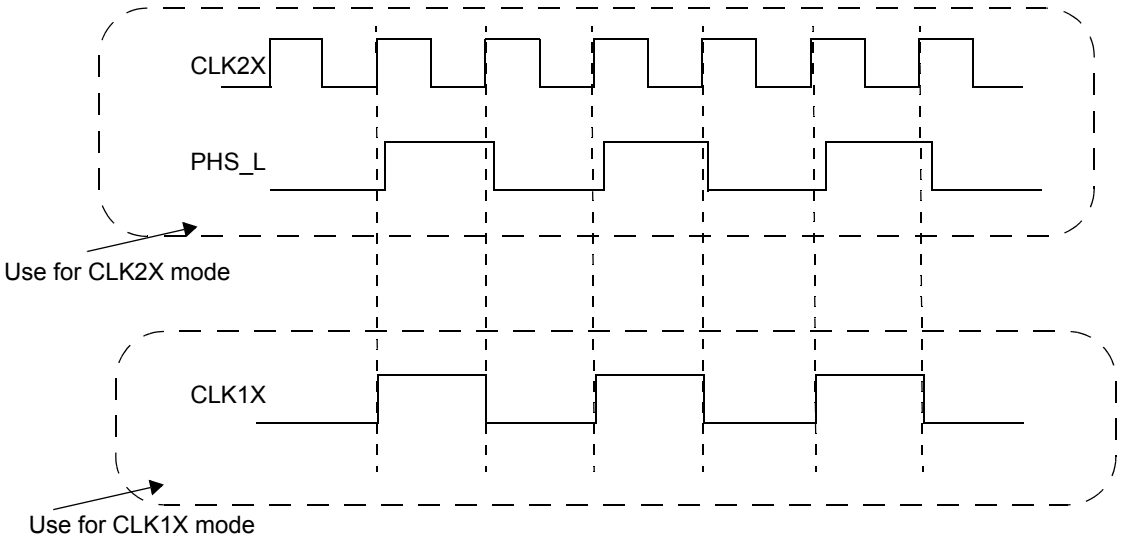


Figure 5-3. CYNSE70128 Clocks for All Timing Diagrams

6.0 Phase-Lock Loop Usage

When the device first powers up, it takes 0.5 ms to lock the internal phase-lock loop (PLL). During this locking of the PLL, in addition to 32 extra CLK1X cycles in CLK1X mode and 64 extra cycles in CLK2X mode, the RST_L must be held low for proper initialization of the device. set-up and hold requirements will change in CLK1X mode if the duty cycle of the CLK1X is varied. All signals into the device in CLK1X mode are sampled by a clock that is generated by multiplying CLK1X by two. Since PLL has a locking range, the device will only work between the range of frequencies specified in the timing specification section.

7.0 Registers

All registers in the CYNSE70128 are 72 bits wide. The CYNSE70128 contains 16 pairs of comparand storage registers, 16 pairs of global mask registers (GMRs), eight search successful index registers and one each of command, information, burst Read, burst Write, and next-free address registers. *Table 7-1* provides an overview of all the CYNSE70128 registers. The registers are ordered in ascending address order. Each register group is then described in the following subsections.

Table 7-1. Register Overview

Address	Abbreviation	Type	Name
0–31	COMP0–31	R	16 pairs of comparand registers that store comparands from the DQ bus for learning later.
32–47 96–111	MASKS	RW	16 global mask register pairs.
48–55	SSR0–7	R	Eight search successful index registers.
56	COMMAND	RW	Command register.
57	INFO	R	Information register.
58	RBURREG	RW	Burst Read register.
59	WBURREG	RW	Burst Write register.
60	NFA	R	Next-free address register.
61–63	–	–	Reserved.

7.1 Comparand Registers

The device contains 32 72-bit comparand registers (16 pairs) dynamically selected in every Search operation to store the comparand presented on the DQ bus. The Learn command will later use these registers when executed. The CYNSE70128 stores the Search command's cycle A comparand in the even-numbered register and the cycle B comparand in the odd-numbered register, as shown in *Figure 7-1*.

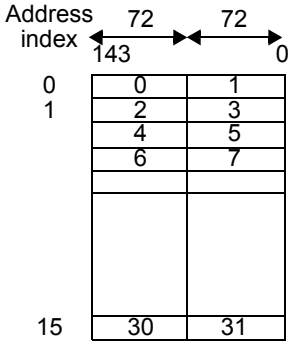


Figure 7-1. Comparand-Register Selection during Search and Learn Instructions

7.2 Mask Registers

The device contains 32 72-bit global mask registers (16 pairs) dynamically selected in every Search operation to select the search subfield. The addressing of these registers is explained in *Figure 7-2*. The four-bit GMR Index supplied on the command (CMD) bus can apply 16 pairs of global masks during the Search and Write operations, as shown below. ***Note***. In 72-bit Search and Write operations, the host ASIC must program both the even and odd mask registers with the same values.

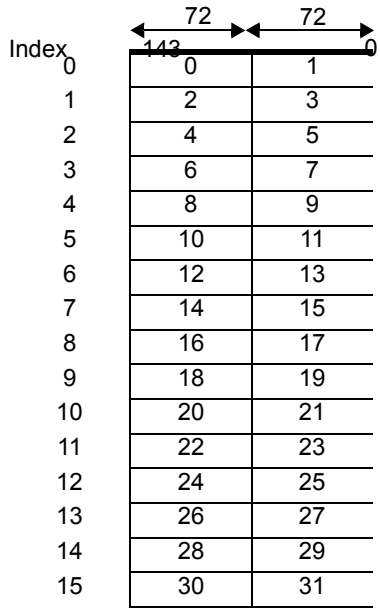


Figure 7-2. Addressing the Global Mask Register Array

Each mask bit in the GMRs is used during Search and Write operations. In Search operations, setting the mask bit to 1 enables compares; setting the mask bit to 0 disables compares (forced match) at the corresponding bit position. In Write operations to the data or mask array, setting the mask bit to 1 enables Writes; setting the mask bit to 0 disables Writes at the corresponding bit position.

7.3 Search Successful Registers (SSR[0:7])

The device contains eight search successful registers (SSRs) to hold the index of the location where a successful Search occurred. The format of each register is described in *Table 7-2*. The Search command specifies which SSR stores the index of a specific Search command in cycle B of the Search instruction. Subsequently, the host ASIC can use this register to access that data array, mask array, or external SRAM using the index as part of the indirect access address (see *Table 10-4* and *Table 7-2*).

The device with a valid bit set performs a Read or Write operation. All other devices suppress the operation.

Table 7-2. Search Successful Register Description

Field	Range	Initial Value	Description
INDEX	[15:0]	X	**Index**. This is the address of the 72-bit entry where a successful search occurs. The device updates this field only when the search is successful. If a hit occurs in a 144-bit entry-size quadrant, the LSB is 0. If a hit occurs in a 288-bit entry-size quadrant, the two LSBs are 00. This index updates if the device is either a local or global winner in a Search operation.
–	[30:16]	0	**Reserved**.
VALID	[31]	0	**Valid**. During Search operation in a depth-cascaded configuration, the device that is a global winner in a match sets this bit to 1. This bit updates only when the device is a global winner in a Search operation.
–	[71:32]	0	**Reserved**.

7.4 Command Register

Table 7-3 describes the command register fields.

Table 7-3. Command Register Description

Field	Range	Initial Value	Description
SRST	[0]	0	**Software Reset**. If 1, this bit resets the device with the same effect as a hardware reset. Internally, it generates a reset pulse lasting for eight CLK cycles. This bit automatically resets to 0 after the reset has completed.
DEVE	[1]	0	**Device Enable**. If 0, it keeps the SRAM bus (SADR, WE_L, CE_L, OE_L, and ALE_L), SSF, and SSV signals in three-state condition and forces the cascade interface output signals LHO[1:0] and BHO[2:0] to 0. It also keeps the DQ bus in input mode. The purpose of this bit is to make sure that there are no bus contentions when the devices power up in the system.
TLSZ	[3:2]	01	**Table Size**. The host ASIC must program this field to configure the chips into a table of a certain size. This field affects the pipeline latency of the Search and Learn operations as well as the Read and Write accesses to the SRAM (SADR[23:0], CE_L, OE_L, WE_L, ALE_L, SSV, SSF, and ACK). Once programmed, the search latency stays constant. Latency in number of CLK cycles with HIGH_SPEED LOW: 00: 1 device 4 01: Up to 8 devices 5 10: Up to 31 devices 6 11: Reserved. Latency number CLK cycles with HIGH_SPEED HIGH: 00: Not supported 01: 1 device 5 10: 2–31 devices 6 11: Reserved.
HLAT	[6:4]	000	**Latency of Hit Signals**. This field further adds latency to the SSF and SSV signals during Search, and ACK signal during SRAM Read access by the following number of CLK cycles. 000: 0 100: 4 001: 1 101: 5 010: 2 110: 6 011: 3 111: 7
LDEV	[7]	0	**Last Device in the Cascade**. When set, this is the last device in the depth-cascaded table and is the default driver for the SSF and SSV signals. In the event of a search failure, the device with this bit set drives the hit signals as follows: SSF = 0, SSV = 1. During nonsearch cycles, the device with this bit set drives the signals as follows: SSF = 0, SSV = 0.
LRAM	[8]	0	**Last Device on the SRAM Bus**. When set, this device is the last device on the SRAM bus in the depth-cascaded table and is the default driver for the SADR, CE_L, WE_L, and ALE_L signals. In cycles where no CYNSE70128 device in a depth cascaded table drives these signals, this devices drives the signals as follows: SADR = 24'hFFFFFF, CE_L = 1, WE_L = 1, and ALE_L = 1. OE_L is always driven by the device for which this bit is set.

Table 7-3. Command Register Description (continued)

Field	Range	Initial Value	Description
CFG	[24:9]	0000000000000000	**Database Configuration**. The device is divided internally into eight partitions of 8K × 72, each of which can be configured as 8K × 72, 4K × 144, or 2K × 288, as follows. 00: 8K × 72 01: 4K × 144 10: 2K × 288 11: low power, partition not used for Search. Bits [10:9] apply to configuring the first partition in the address space. Bits [12:11] apply to configuring the second partition in the address space. Bits [14:13] apply to configuring the third partition in the address space. Bits [16:15] apply to configuring the fourth partition in the address space. Bits [18:17] apply to configuring the fifth partition in the address space. Bits [20:19] apply to configuring the sixth partition in the address space. Bits [22:21] apply to configuring the seventh partition in the address space. Bits [24:23] apply to configuring the eighth partition in the address space.
	[71:25]	0	**Reserved**.

7.5 Information Register

Table 7-4 describes the information register fields.

Table 7-4. Information Register Description

Field	Range	Initial Value	Description
Revision	[3:0]	0001	**Revision Number**. This is the current device revision number. Numbers start at one and increment by one for each revision of the device.
Implementation	[6:4]	001	This is the CYNSE70128 implementation number.
Reserved	[7]	0	**Reserved**.
Device ID	[15:8]	00000100	This is the device identification number.
MFID	[31:16]	1101_1100_0111_1111	**Manufacturer ID**. This field is the same as the manufacturer identification number and continuation bits in the TAP controller.
Reserved	[71:32]		**Reserved**.

7.6 Read Burst Address Register

Table 7-5 shows the Read burst address register (RBURREG) fields which must be programmed before a burst Read.

Table 7-5. Read Burst Register Description

Field	Range	Initial Value	Description
ADR	[15:0]	0	**Address**. This is the starting address of the data or mask array during a burst Read operation. It automatically increments by one for each successive Read of the data or mask array. Once the operation is complete, the contents of this field must be reinitialized for the next operation.
	[18:16]		**Reserved**.
BLEN	[27:19]	0	**Length of Burst Access**. The device provides the capability to read from 4–511 locations in a single burst. The BLEN decrements automatically. Once the operation is complete, the contents of this field must be reinitialized for the next operation.
	[71:28]		**Reserved**.

7.7 Write Burst Address Register Description

Table 7-6 describes the Write burst address register (WBURREG) fields which must be programmed before a burst Write.

Table 7-6. Write Burst Register Description

Field	Range	Initial Value	Description
ADR	[15:0]	0	**Address**. This is the starting address of the data or mask array during a burst Write operation. It automatically increments by one for each successive Write of the data or mask array. Once the operation is complete, the contents of this field must be reinitialized for the next operation.
	[18:16]		**Reserved**.
BLEN	[27:19]	0	**Length of Burst Access**. The device provides the capability to write from 4–511 locations in a single burst. The BLEN decrements automatically. Once the operation is complete, the contents of this field must be reinitialized for the next operation.
	[71:28]		**Reserved**.

7.8 NFA Register

Bit [0] of each 72-bit data entry is specially designated for use in the operation of the Learn command. For 72-bit-configured quadrants, this bit indicates whether a location is full (bit set to 1) or empty (bit set to 0). Every Write and/or Learn command loads the address of the first 72-bit location that contains a 0 in the entry's bit[0]. This is stored in the NFA register (see *Table 7-7*). If all the bits[0] in a device are set to 1, the CYNSE70128 asserts FULO[1:0] to 1.

For a 144-bit-configured quadrants, the LSB of the NFA register is always set to 0. The host ASIC must set both bit[0] and bit[72] in a 144-bit word to either 0 or 1 to indicate full or empty status. Both bit[0] and bit[72] must be set to either 0 or 1, (that is, the 10 or 01 settings are invalid).

Table 7-7. NFA Register

Address	71 – 16	15 – 0
60	Reserved	Index

8.0 NSE Architecture and Operation Overview

The CYNSE70128 consists of 64K × 72-bit storage cells referred to as data bits. There is a mask cell corresponding to each data cell. *Figure 8-1* shows the three organizations of the device based on the value of the CFG bits in the command register.

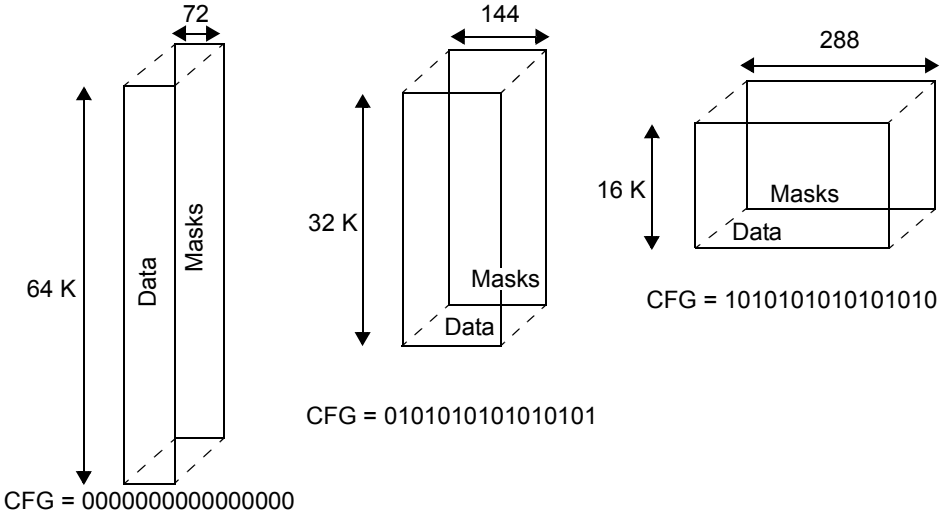


Figure 8-1. CYNSE70128 Database Width Configuration

During a Search operation, the search data bit (S), data array bit (D), mask array bit (M), and global mask bit (G) are used in the following manner to generate a match at that bit position (see *Table 8-1*). The entry with a match on every bit position results in a successful search during a Search operation.

Table 8-1. Bit Position Match

G	M	D	S	Match
0	X	X	X	1
1	0	X	X	1
1	1	0	0	1
1	1	1	0	0
1	1	0	1	0
1	1	1	1	1

In order for a successful search within a device to make the device the local winner in the Search operation, all 72-bit positions must generate a match for a 72-bit entry in 72-bit-configured quadrants, or all 144-bit positions must generate a match for two consecutive even and odd 72-bit entries in quadrants configured as 144 bits, or all 288-bit positions must generate a match for four consecutive entries aligned to four entry-page boundaries of 72-bit entries in quadrants configured as 288 bits.

An arbitration mechanism using a cascade bus determines the global winning device among the local winning devices in a Search cycle. The global winning device drives the SRAM bus, SSV, and the SSF signals. In case of a Search failure, the device(s) with the LDEV and LRAM bits set drive(s) the SRAM bus, SSF, and SSV signals.

The CYNSE70128 device can be configured to contain tables of different widths, even within the same chip. *Figure 8-2* shows a sample configuration of different widths.

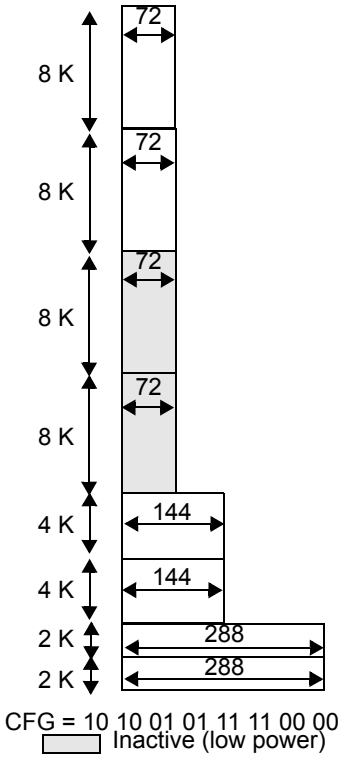


Figure 8-2. Multiwidth Database Configurations Example

9.0 Data and Mask Addressing

Figure 9-1 shows CYNSE70128 data and mask array addressing.

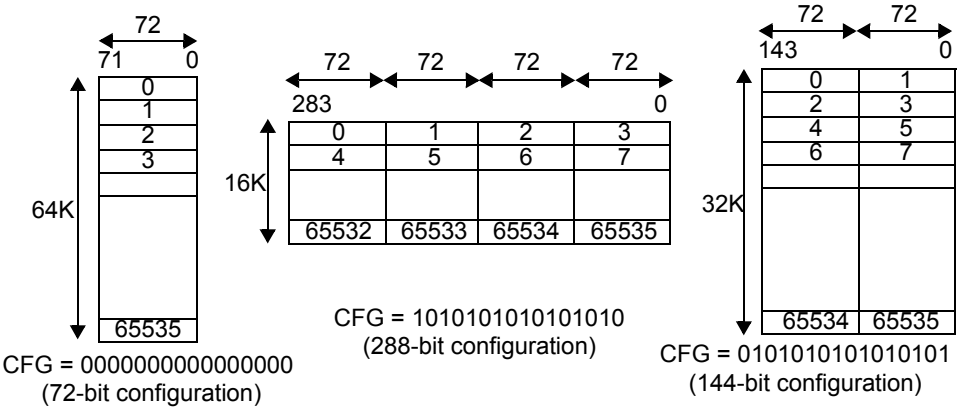


Figure 9-1. Addressing the CYNSE70128 Data and Mask Arrays

10.0 Commands

A master device such as an ASIC controller issues commands to the CYNSE70128 device using the command valid (CMDV) signal and the CMD bus. The following subsections describe the operation of the commands.

10.1 Command Codes

The CYNSE70128 implements four basic commands, shown in *Table 10-1*. The command code must be presented to CMD[1:0] while keeping the CMDV signal high for two CLK2X cycles (designated as cycles A and B) when the CLK_MODE pin is low. In CLK2X mode, the controller ASIC must align the instructions using the PHS_L signal. The command code must be presented to CMD[1:0] while keeping the CMDV signal high for one CLK1X cycle when the CLK_MODE pin is high. In CLK1X mode the high phase of the CLK1X is designated as cycle A and the low phase of the CLK1X is designated as cycle B. The CMD[10:2] field passes the parameters of the command in cycles A and B.

Table 10-1. Command Codes

Command Code	Command	Description
00	Read	Reads one of the following: data array, mask array, device registers, or external SRAM.
01	Write	Writes one of the following: data array, mask array, device registers, or external SRAM.
10	Search	Searches the data array for a desired pattern using the specified register from the GMR array and local mask associated with each data cell.
11	Learn[8]	The device has internal storage for up to 16 comparands that it can learn. The device controller can insert these entries at the next free address (as specified by the NFA register) using the Learn instruction.

10.2 Commands and Command Parameters

Table 10-2 lists the CMD bus fields that contain the CYNSE70128 command parameters and their respective cycles. Each command is described separately in the subsections that follow.

Table 10-2. Command Parameters

CMD[9, 10]	CYC	10	9	8	7	6	5	4	3	2	1	0
Read	A	X	X	SADR[23]	SADR[22]	SADR[21]	0	0	0	0 = Single 1 = Burst	0	0
	B	X	X	0	0	0	0	0	0	0 = Single 1 = Burst	0	0
Write	A	Global Mask Register Index[9]	0 Normal Write 1 Parallel Write	SADR[23]	SADR[22]	SADR[21]	Global Mask Register Index [2:0]			0 = Single 1 = Burst	0	1
	B	Global Mask Register Index[9]	0 Normal Write 1 Parallel Write	0	0	0	Global Mask Register Index [2:0]			0 = Single 1 = Burst	0	1
Search	A	Global Mask Register Index[10]	72 bit: 0 144-bit: 1 288 bit: X	SADR[23]	SADR[22]	SADR[21]	Global Mask Register Index [2:0]			72-bit or 144-bit: 0 288-bit: 1 in 1st cycle 0 in 2nd cycle	1	0
	B	X		Successful Search Register Index[2:0]			Comparand Register Index				1	0
Learn[8]	A	X	X	SADR[23]	SADR[22]	SADR[21]	Comparand Register Index				1	1
	B	X	X	0	0	Mode 0: 72-bit 1: 144-bit	Comparand Register Index				1	1

10.3 Read Command

The Read can be a single Read of a data array, a mask array, an SRAM, or a register location (CMD[2] = 0). It can be a burst Read of the data (CMD[2] = 1) or mask array locations using an internal auto-incrementing address register (RBURADR). A description of each type is provided in *Table 10-3*. A single-location Read operation lasts six cycles, as shown in *Figure 10-1*. The burst Read adds two cycles for each successive Read. The SADR[23:21] bits supplied in the Read instruction cycle A drives SADR[23:21] signals during the Read of an SRAM location.

Notes:

8. The 288-bit-configured devices or 288-bit-configured quadrants within devices do not support the Learn instruction. Also, CLK1X must be less than 8 3MHz.
9. Use only CMD[8:0] and connect the CMD[10:9] to ground with CFG_L LOW.
10. For a description of CMD[9] and CMD[2], see subsections on search 288-bit configured tables and mixed-size searches with CFG_L HIGH.

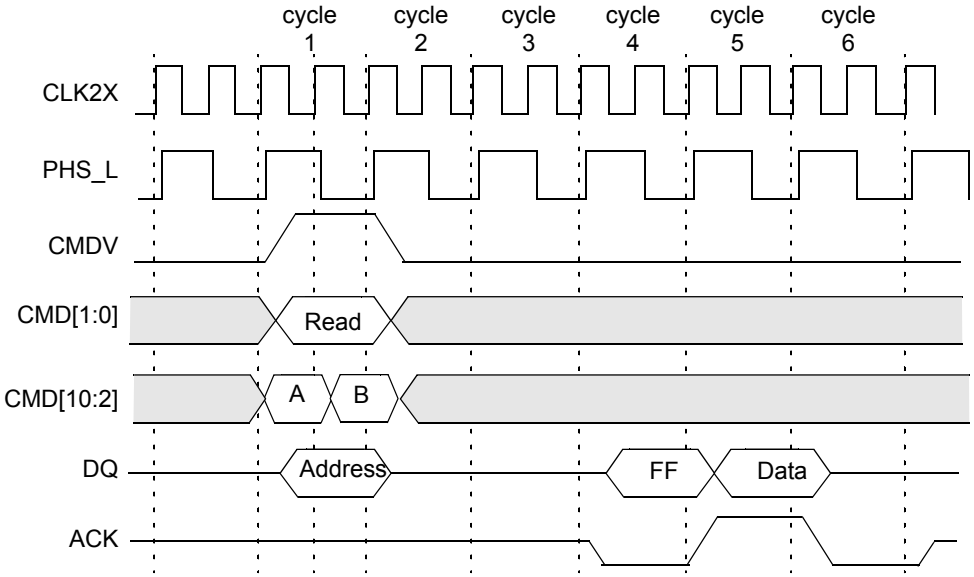


Figure 10-1. Single-Location Read Cycle Timing

Table 10-3. Read Command Parameters

CMD Parameter CMD[2]	Read Command	Description
0	Single Read	Reads a single location of the data array, mask array, external SRAM, or device registers. All access information is applied on the DQ bus.
1	Burst Read	Reads a block of locations from the data array, or mask array as a burst. The internal register (RBURADR) specifies the starting address and the length of the data transfer from the data or mask array, and it auto-increments the address for each access. All other access information is applied on the DQ bus. ***Note***. The device registers and external SRAM can only be read in single-Read mode.

The single Read operation takes six clock cycles, in the following sequence.

- **Cycle 1**: The host ASIC applies the Read instruction on the CMD[1:0] (CMD[2]= 0) using CMDV = 1 and the DQ bus supplies the address, as shown in *Table 10-4* and *Table 10-5*. The host ASIC selects the CYNSE70128 for which ID[4:0] matches the DQ[25:21] lines. If the DQ[25:21] = 11111, the host ASIC selects the CYNSE70128 with the LDEV bit set. The host ASIC also supplies SADR[23:21] on CMD[8:6] in cycle A of the Read instruction if the Read is directed to the external SRAM.
- **Cycle 2**: The host ASIC floats DQ[71:0] to three-state condition.
- **Cycle 3**: The host ASIC keeps DQ[71:0] in three-state condition.
- **Cycle 4**: The selected device starts to drive the DQ[71:0] bus, and drives the ACK signal from Z to LOW.
- **Cycle 5**: The selected device drives the read data from the addressed location on the DQ[71:0] bus, and drives the ACK signal HIGH.
- **Cycle 6**: The selected device floats the DQ[71:0] to three-state condition and drives the ACK signal LOW.

At the termination of cycle 6, the selected device releases the ACK line to three-state condition. The Read instruction is complete, and a new operation can begin. ***Note***. The latency of the SRAM Read will be different than the one described above (see “SRAM PIO Access” on page 106). *Table 10-4* lists and describes the format of the Read address for a data array, mask array, or SRAM.

Table 10-4. Read Address Format for Data Array, Mask Array, or SRAM

DQ[71:30]	DQ[29]	DQ[28:26]	DQ[25:21]	DQ[20:19]	DQ[18:16]	DQ[15:0]
Reserved	0: Direct 1: Indirect	Successful Search Register Index (appli-cable if DQ[29] is indirect)	ID	00: Data Array	Reserved	If DQ[29] is 0, this field carries the address of the data array location. If DQ[29] is 1, the SSRI specified on DQ[28:26] is used to generate the address of the data array location: {SSR[15:2], SSR[1] | DQ[1], SSR[0] | DQ[0]}.[11]

Note:
11. “|” stands for logical OR operation. “{}” stands for concatenation operator.

Table 10-4. Read Address Format for Data Array, Mask Array, or SRAM (continued)

DQ[71:30]	DQ[29]	DQ[28:26]	DQ[25:21]	DQ[20:19]	DQ[18:16]	DQ[15:0]
Reserved	0: Direct 1: Indirect	Successful Search Register Index (applicable if DQ[29] is indirect)	ID	01: Mask Array	Reserved	If DQ[29] is 0, this field carries the address of the mask array location. If DQ[29] is 1, the SSRI specified on DQ[28:26] is used to generate the address of the mask array location: {SSR[15:2], SSR[1] | DQ[1], SSR[0] | DQ[0]}.[11]
Reserved	0: Direct 1: Indirect	Successful Search Register Index (applicable if DQ[29] is indirect)	ID	10: External SRAM	Reserved	If DQ[29] is 0, this field carries the address of the SRAM location. If DQ[29] is 1, the SSRI specified on DQ[28:26] is used to generate the address of the SRAM location: {SSR[15:2], SSR[1] | DQ[1], SSR[0] | DQ[0]}.[11]

Table 10-5 describes the Read address format for the internal registers. *Figure 10-2* illustrates the timing diagram for the burst Read of the data or mask array.

Table 10-5. Read Address Format for Internal Registers

DQ[71:26]	DQ[25:21]	DQ[20:19]	DQ[18:7]	DQ[6:0]
Reserved	ID	11: Register	Reserved	Register Address

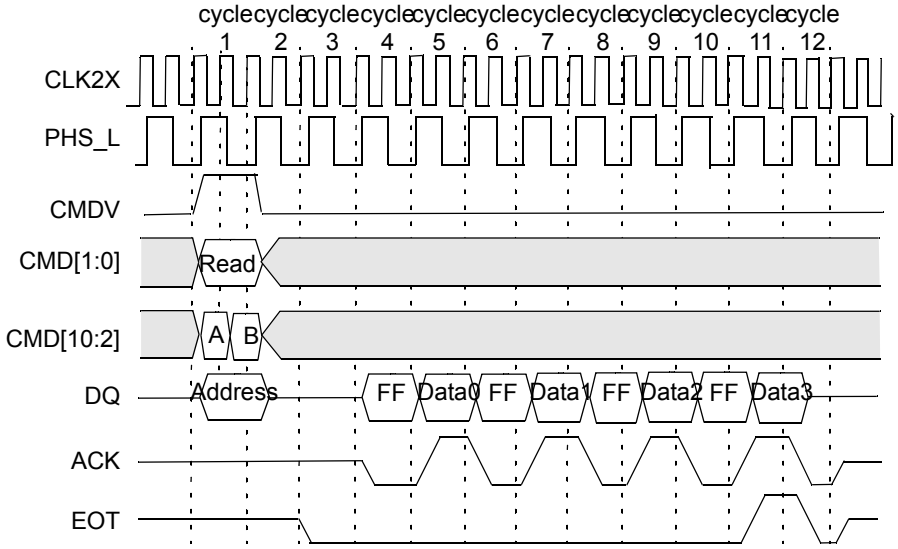


Figure 10-2. Burst Read of the Data and Mask Arrays (BLEN = 4)

The Read operation lasts 4 + 2n CLK cycles (where n is the number of accesses in the burst specified by the BLEN field of the RBURREG) in the sequence shown below. This operation assumes that the host ASIC has programmed the RBURREG with the starting address (ADR) and the length of the transfer (BLEN) before initiating the burst Read command.

- **Cycle 1**: The host ASIC applies the Read instruction on CMD[1:0] (CMD[2] = 1) using CMDV = 1 and the address supplied on the DQ bus, as shown in *Table 10-6*. The host ASIC selects the CYNSE70128 where ID[4:0] matches the DQ[25:21] lines. If the DQ[25:21] = 11111, the host ASIC selects the CYNSE70128 with the LDEV bit set.
- **Cycle 2**: The host ASIC floats DQ[71:0] to the three-state condition.
- **Cycle 3**: The host ASIC keeps DQ[71:0] in the three-state condition.
- **Cycle 4**: The selected device starts to drive the DQ[71:0] bus and drives ACK and EOT from Z to LOW.
- **Cycle 5**: The selected device drives the Read data from the addressed location on the DQ[71:0] bus, and drives the ACK signal HIGH.

Cycles 4 and 5 repeat for each additional access until all the accesses specified in the burst length (BLEN) field of RBURREG are complete. On the last transfer, the CYNSE70128 drives the EOT signal high.

- **Cycle (4 + 2n)**: The selected device drives the DQ[71:0] to the three-state condition, and drives the ACK and EOT signals LOW.

At the termination of cycle (4 + 2n), the selected device floats the ACK line to the three-state condition. The burst Read instruction is complete, and a new operation can begin. *Table 10-6* describes the Read address format for data and mask arrays for burst Read operations.

Table 10-6. Read Address Format for Data and Mask Arrays

DQ[71:26]	DQ[25:21]	DQ[20:19]	DQ[18:16]	DQ[15:0]
Reserved	ID	00: Data Array	Reserved	***Do not care***. These 16 bits come from the internal register (RBURADR) which increments for each access.
Reserved	ID	01: Mask Array	Reserved	***Do not care***. These 16 bits come from the internal register (RBURADR) which increments for each access.

10.4 Write Command

The Write can be a single Write of a data array, mask array, register, or external SRAM location (CMD[2] = 0). It can be a burst Write (CMD[2] = 1) using an internal auto-incrementing address register (WBURADR) of the data or mask array locations. A single-location Write is a three-cycle operation, as shown in *Figure 10-3*. The burst Write adds one extra cycle for each successive location Write.

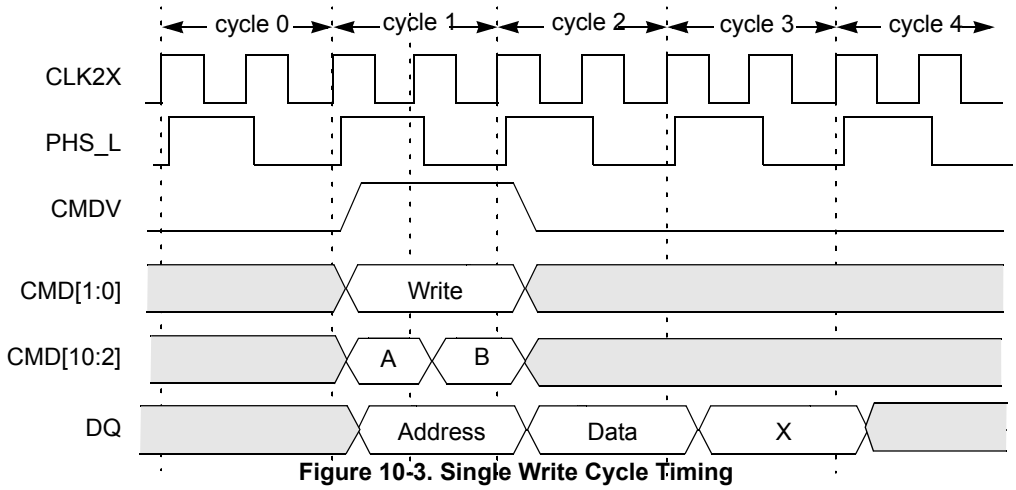


Figure 10-3. Single Write Cycle Timing

The following is the Write operation sequence, and *Table* shows the Write address format for the data array, the mask array, or the single-Write SRAM. *Table 10-8* shows the Write address format for the internal registers.

- **Cycle 1A**: The host ASIC applies the Write instruction to the CMD[1:0] (CMD[2] = 0), using CMDV = 1 and the address supplied on the DQ bus. The host ASIC also supplies the GMR Index to mask the Write to the data or mask array location on {CMD[10],CMD[5:3]}. For SRAM Writes, the host ASIC must supply the SADR[23:21] on CMD[8:6]. The host ASIC sets CMD[9] to 0 for the normal Write.
- **Cycle 1B**:The host ASIC continues to apply the Write instruction to the CMD[1:0] (CMD[2] = 0), using CMDV = 1 and the address supplied on the DQ bus. The host ASIC continues to supply the GMR Index to mask the Write to the data or mask array locations in {CMD[10], CMD[5:3]}.The host ASIC selects the device where ID[4:0] matches the DQ[25:21] lines, or it selects all the devices when DQ[25:21] = 11111.
- **Cycle 2**: The host ASIC drives the DQ[71:0] with the data to be written to the data array, mask array, or register location of the selected device.
- **Cycle 3**: Idle cycle.

At the termination of cycle 3, another operation can begin. ***Note***. The latency of the SRAM Write will be different than the one described above (see “SRAM PIO Access” on page 106).

Table 10-7. Write Address Format for Data Array, Mask Array or SRAM (Single Write)

DQ[71:30]	DQ[29]	DQ[28:26]	DQ[25:21]	DQ[20:19]	DQ[18:16]	DQ[15:0]
Reserved	0: Direct 1: Indirect	SSR (applicable if DQ[29] is indirect)	ID	00: Data Array	Reserved	If DQ[29] is 0, this field carries the address of the data array location. If DQ[29] is 1, the SSR specified on DQ[28:26] is used to generate the address of data array location: {SSR[15:2], SSR[1] | DQ[1], SSR[0] | DQ[0]} [*Figure 10-4*].[12]
Reserved	0: Direct 1: Indirect	SSR (applicable if DQ[29] is indirect)	ID	01: Mask Array	Reserved	If DQ[29] is 0, this field carries the address of the mask array location. If DQ[29] is 1, the SSR specified on DQ[28:26] is used to generate the address of the mask array location: {SSR[15:2], SSR[1] | DQ[1], SSR[0] | DQ[0]} [*Figure 10-4*].
Reserved	0: Direct 1: Indirect	SSR (applicable if DQ[29] is indirect)	ID	10: External SRAM	Reserved	If DQ[29] is 0, this field carries the address of the SRAM location. If DQ[29] is 1, the SSR specified on DQ[28:26] is used to generate the address of SRAM location: {SSR[15:2], SSR[1] | DQ[1], SSR[0] | DQ[0]} [*Figure 10-4*].

Table 10-8. Write Address Format for Internal Registers

DQ[71:26]	DQ[25:21]	DQ[20:19]	DQ[18:7]	DQ[6:0]
Reserved	ID	11: Register	Reserved	Register address

Figure 10-4 shows the timing diagram of a burst Write operation of the data or mask array.

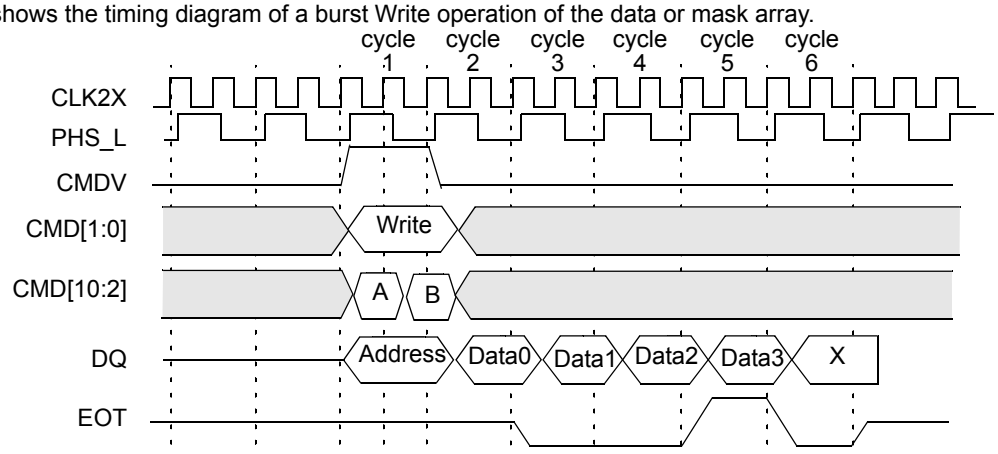


Figure 10-4. Burst Write of the Data and Mask Arrays (BLEN = 4)

The burst Write operation lasts for (n + 2) CLK cycles. n signifies the number of accesses in the burst as specified in the BLEN field of the WBURREG register. The following is the block Write operation sequence. This operation assumes that the host ASIC has programmed the WBURREG with the starting address (ADR) and the length of transfer (BLEN) before initiating a burst Write command.

- **Cycle 1A**: The host ASIC applies the Write instruction to the CMD[1:0] (CMD[2] = 1), using CMDV = 1 and the address supplied on the DQ bus, as shown in *Table 10-9*. The host ASIC also supplies the GMR Index to mask the Write to the data or mask array locations in {CMD[10], CMD[5:3]}. The host ASIC sets CMD[9] to 0 for the normal Write.
- **Cycle 1B**: The host ASIC continues to apply the Write instruction on the CMD[1:0] (CMD[2] = 1), using CMDV = 1 and the address supplied on the DQ bus. The host ASIC continues to supply the GMR Index to mask the Write to the data or mask array locations in {CMD[10], CMD[5:3]}. The host ASIC selects the device for which ID[4:0] matches the DQ[25:21] lines. It selects all the devices when DQ[25:21] = 11111.
- **Cycle 2**: The host ASIC drives the DQ[71:0] with the data to be written to the data or mask array location of the selected device. The CYNSE70128 writes the data from the DQ[71:0] bus only to the subfield that has the corresponding mask bit set to 1 in the GMR specified by the index {CMD[10],CMD[5:3]} supplied in cycle 1.

Note:
12. "|" stands for logical OR operation. "{}" stands for concatenation operator.

- **Cycles 3 to n + 1**: The host ASIC drives the DQ[71:0] with the data to be written to the next data or mask array location (addressed by the auto-increment ADR field of the WBURREG register) of the selected device.

The CYNSE70128 writes the data on the DQ[71:0] bus only to the subfield that has the corresponding mask bit set to 1 in the GMR specified by the index {CMD[10],CMD[5:3]} supplied in cycle 1. The CYNSE70128 drives the EOT signal low from cycle 3 to cycle n; the CYNSE70128 drives the EOT signal high in cycle n + 1 (n is specified in the BLEN field of the WBURREG).

- **Cycle n + 2**: TheCYNSE70128 drives the EOT signal LOW.

At the termination of cycle n + 2, the CYNSE70128 floats the EOT signal to a three-state operation, and a new instruction can begin.

Table 10-9. Write Address Format for Data and Mask Array (Burst Write)

DQ[71:26]	DQ[25:21]	DQ[20:19]	DQ[18:16]	DQ[15:0]
Reserved	ID	00: Data array	Reserved	***Do not care***. These 16 bits come from the internal register (WBURADR), which increments with each access.
Reserved	ID	01: Mask array	Reserved	***Do not care***. These 16 bits come from the internal register (WBURADR), which increments with each access.

10.5 Parallel Write

In order to write the data and mask arrays faster for initialization, testing, or diagnostics, many locations can be written simultaneously in the CYNSE70128 device using Parallel Write. Parallel Write allows the user to specify one address and write multiple locations in the data or mask array with the same data.In order to perform Parallel Write, CMD[9] should be set in cycles A and B of the Write command to the data or mask arrays. The address bits DQ[10:1] specify which location to perform parallel write to. DQ[15:11] defines a set of 32 partitions all of which write two 72 bit entries (DQ[0] is ignored). Thus 64 72-bit locations are simultaneously written in either the data or mask array during Parallel Write.

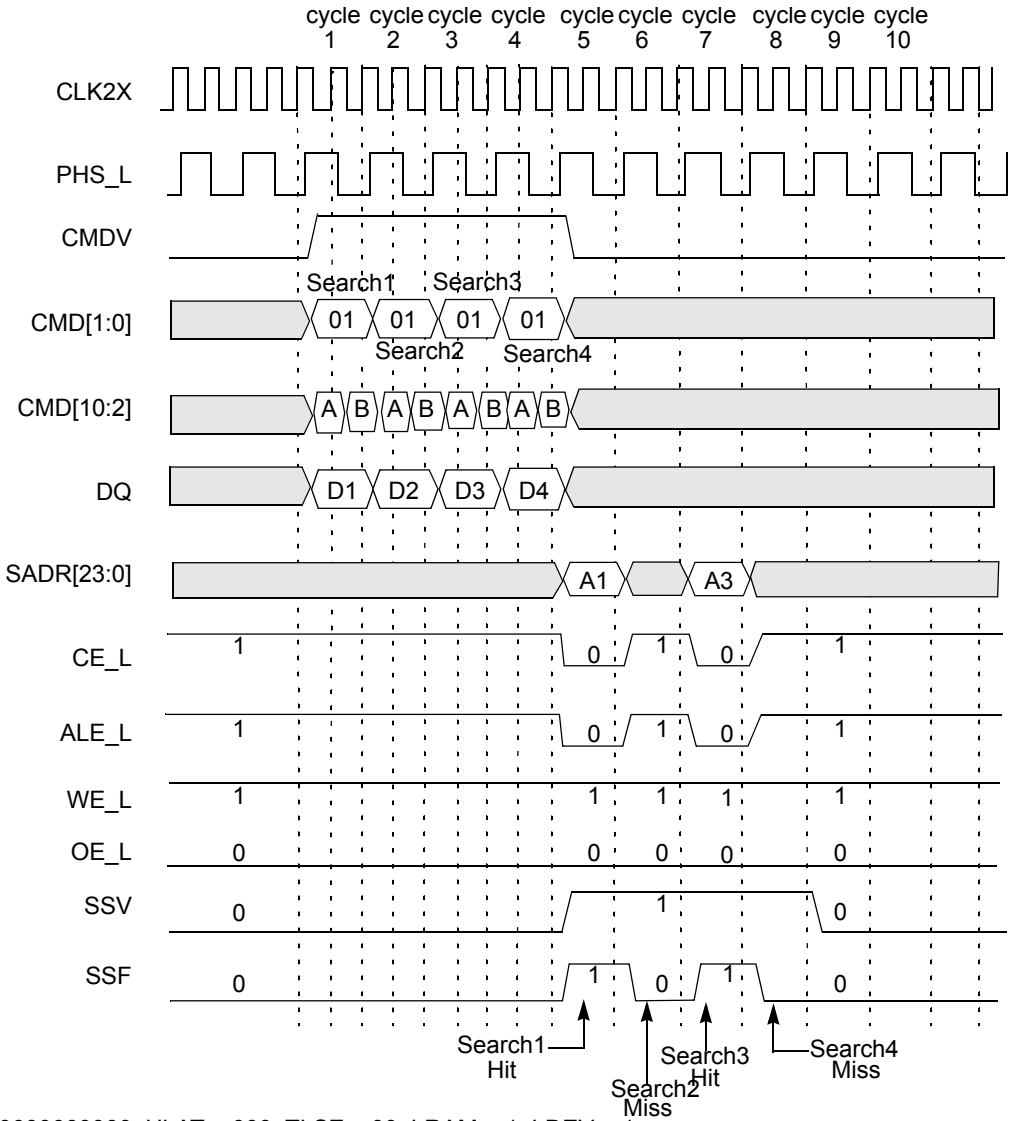
10.6 Search Command

This subsection describes the following:

- 72-bit search on tables configured as ×72 using one device
- 72-bit search on tables configured as ×72 using up to eight devices
- 72-bit search on tables configured as ×72 using up to 31 devices
- 144-bit search on tables configured as ×144 using one device
- 144-bit search on tables configured as ×144 using up to eight devices
- 144-bit search on tables configured as ×144 using up to 31 devices
- 288-bit search on tables configured as ×288 using one device
- 288-bit search on tables configured as ×288 using up to eight devices
- 288-bit search on tables configured as ×288 using up to 31 devices
- Mixed-size searches on tables configured with different widths using an CYNSE70128 with CFG_L low
- Mixed-size searches on tables configured with different widths using an CYNSE70128 with CFG_L high.

10.6.1 72-bit Search on Tables Configured as ×72 Using a Single CYNSE70128 Device

Figure 10-5 shows the timing diagram for a Search command in the 72-bit-configured table (CFG = 0000000000000000) consisting of a single device for one set of parameters: TLSZ = 00, HLAT = 000, LRAM = 1, LDEV = 1. The hardware diagram for this search subsystem is shown in *Figure 10-6*.



CFG = 0000000000000000, HLAT = 000, TLSZ = 00, LRAM = 1, LDEV = 1.

Figure 10-5. Timing Diagram for 72-bit Search in x72 Table (One Device)

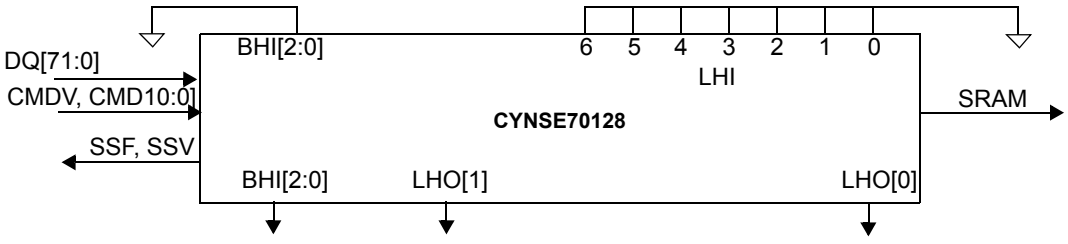


Figure 10-6. Hardware Diagram for a Table with One Device

The following is the sequence of operation for a single 72-bit search command (also refer to “Command and Command Parameters,” Subsection 10.2 on page 22).

- **Cycle A**: The host ASIC drives the CMDV high and applies Search command code (10) on CMD[1:0] signals. {CMD[10], CMD[5:3]} signals must be driven with the index to the GMR pair for use in this Search operation. CMD[8:6] signals must be driven with the same bits that will be driven on SADR[23:21] by this device if it has a hit. DQ[71:0] must be driven with the 72-bit data to be compared. The CMD[2] signal must be driven to logic 0.
- **Cycle B**: The host ASIC continues to drive the CMDV high and to apply Search command (10) on CMD[1:0]. CMD[5:2] must be driven by the index of the comparand register pair for storing the 144-bit word presented on the DQ bus during cycles A

and B. CMD[8:6] signals must be driven with the index of the SSR that will be used for storing the address of the matching entry and the hit flag (see page 8 for information on SSR[0:7]). The DQ[71:0] continues to carry the 72-bit data to be compared.

Note. For 72-bit searches, the host ASIC must supply the same 72-bit data on DQ[71:0] during both cycles A and B. The even and odd pair of GMRs selected for the compare must be programmed with the same value.

The logical 72-bit Search operation is shown in *Figure 10-7*. The entire table consisting of 72-bit entries is compared to a 72-bit word K (presented on the DQ bus in both cycles A and B of the command) using the GMR and the local mask bits. The effective GMR is the 72-bit word specified by the identical value in both even and odd GMR pairs selected by the GMR Index in the command's cycle A. The 72-bit word K (presented on the DQ bus in both cycles A and B of the command) is also stored in both even and odd comparand register pairs selected by the Comparand Register Index in the command's cycle B. In a ×72 configuration, only the even comparand register can be subsequently used by the Learn command. The word K (presented on the DQ bus in both cycles A and B of the command) is compared with each entry in the table starting at location 0. The first matching entry's location address L is the winning address that is driven as part of the SRAM address on the SADR[23:0] lines (see "SRAM Addressing" on page 105).

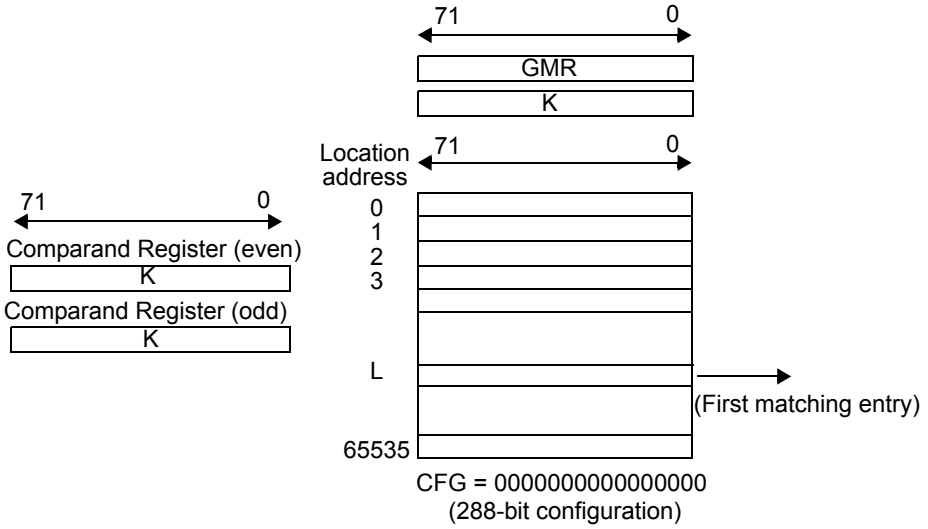


Figure 10-7. x72 Table with One Device

The Search command is a pipelined operation and executes a Search at half the rate of the frequency of CLK2X for 72-bit searches in ×72-configured tables. The latency of SADR, CE_L, ALE_L, WE_L, SSV, and SSF from the 72-bit Search command cycle (two CLK2X cycles) is shown in *Table 10-10*.

Table 10-10. The Latency of Search from Instruction to SRAM Access Cycle

Number of Devices	Max Table Size	Latency in CLK Cycles
1 (TLSZ = 00)	64K × 72 bits	4
1–8 (TLSZ = 01)	512K × 72 bits	5
1–31 (TLSZ = 10)	1984K × 72 bits	6

The latency of a Search from command to SRAM access cycle is 4 for a single device in the table and TLSZ = 00. In addition, SSV and SSF shift further to the right for different values of HLAT, as specified in *Table 10-11*.

Table 10-11. Shift OF SSF and SSV from SADR

HLAT	Number of CLK Cycles
000	0
001	1
010	2
011	3
100	4
101	5
110	6
111	7

10.6.2 72-bit Search on Tables Configured as ×72 Using up to Eight CYNSE70128 Devices

The hardware diagram of the search subsystem of eight devices is shown in *Figure 10-8*. The following are the parameters programmed into the eight devices.

- First seven devices (device 0–6): CFG = 0000000000000000, TLSZ = 01, HLAT = 010, LRAM = 0, and LDEV = 0.
- Eighth device (device 7): CFG = 0000000000000000, TLSZ = 01, HLAT = 010, LRAM = 1, and LDEV = 1.

Note. All eight devices must be programmed with the same values for TLSZ and HLAT. Only the last device in the table (device number 7 in this case) must be programmed with LRAM = 1 and LDEV = 1. All other upstream devices (devices 0 through 6 in this case) must be programmed with LRAM = 0 and LDEV = 0.

Figure 10-9 shows the timing diagram for a Search command in the 72-bit-configured table of eight devices for device number 0. *Figure 10-10* shows the timing diagram for a Search command in the 72-bit-configured table of eight devices for device number 1. *Figure 10-11* shows the timing diagram for a Search command in the 72-bit-configured table of eight devices for device number 7 (the last device in this specific table). For these timing diagrams four 72-bit searches are performed sequentially. Hit/Miss assumptions were made as shown below in *Table 10-12*.

Table 10-12. Hit/Miss Assumption

Search Number	1	2	3	4
Device 0	Hit	Miss	Hit	Miss
Device 1	Miss	Hit	Hit	Miss
Device 2–6	Miss	Miss	Miss	Miss
Device 7	Miss	Miss	Hit	Hit

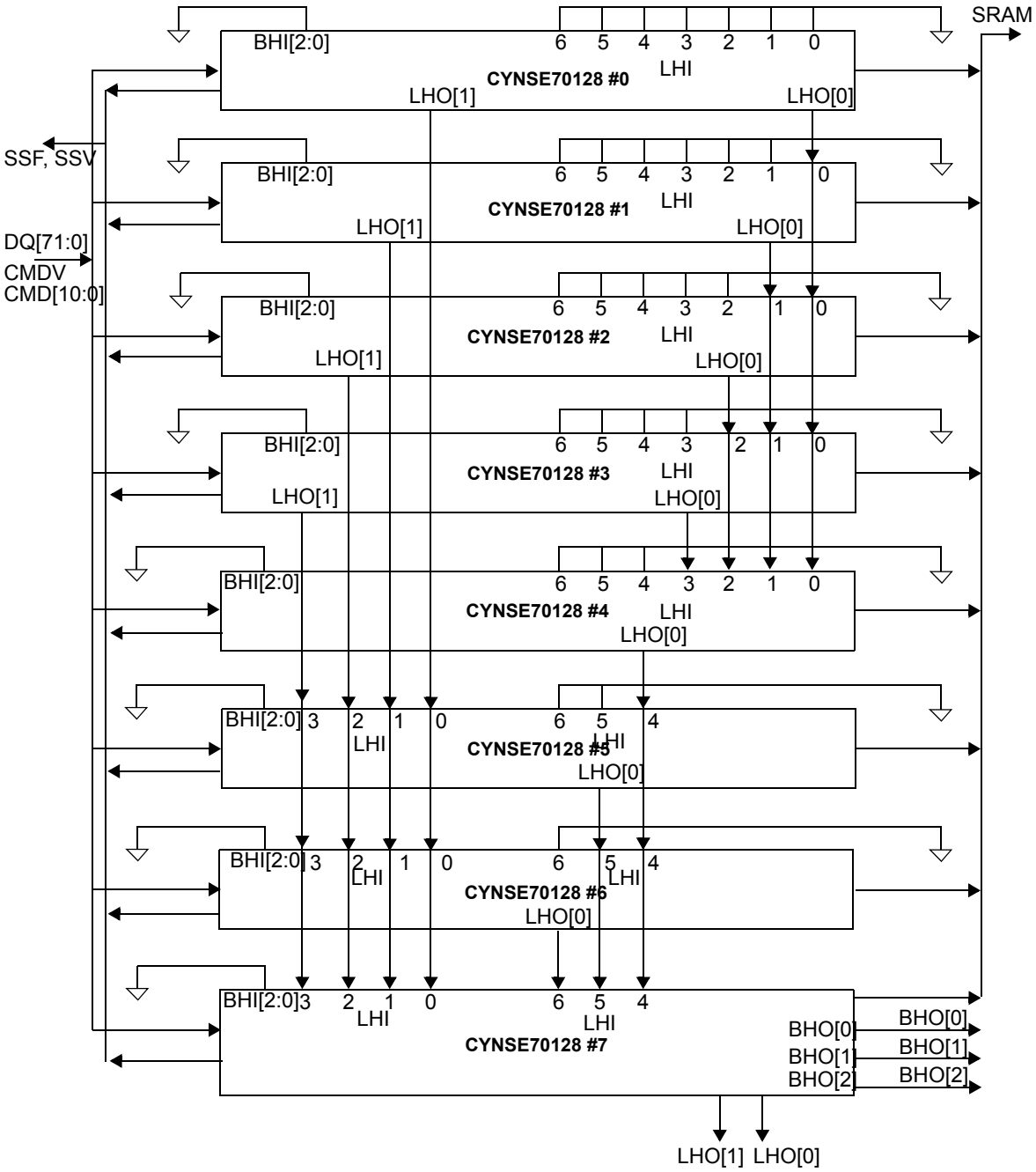
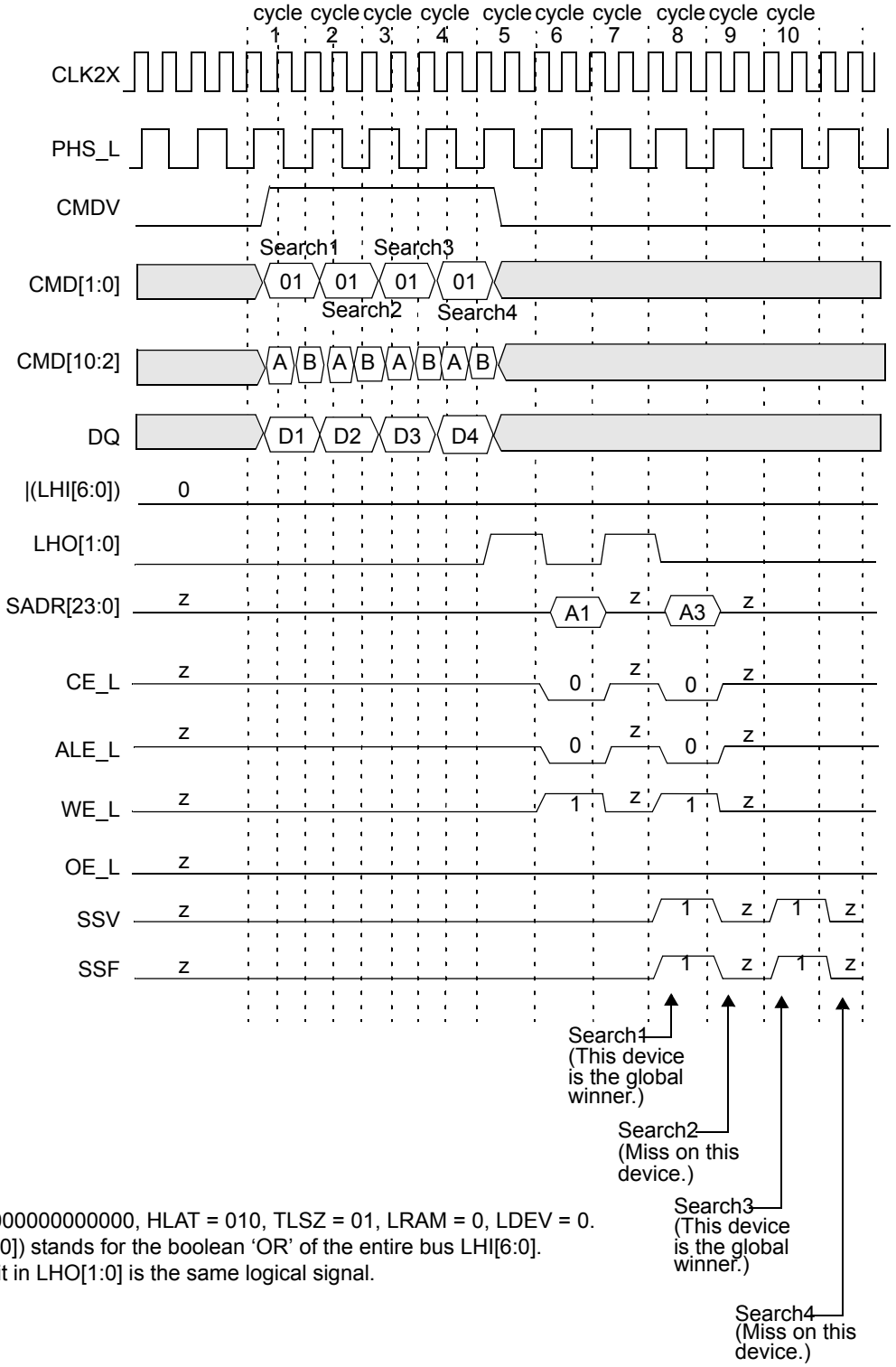
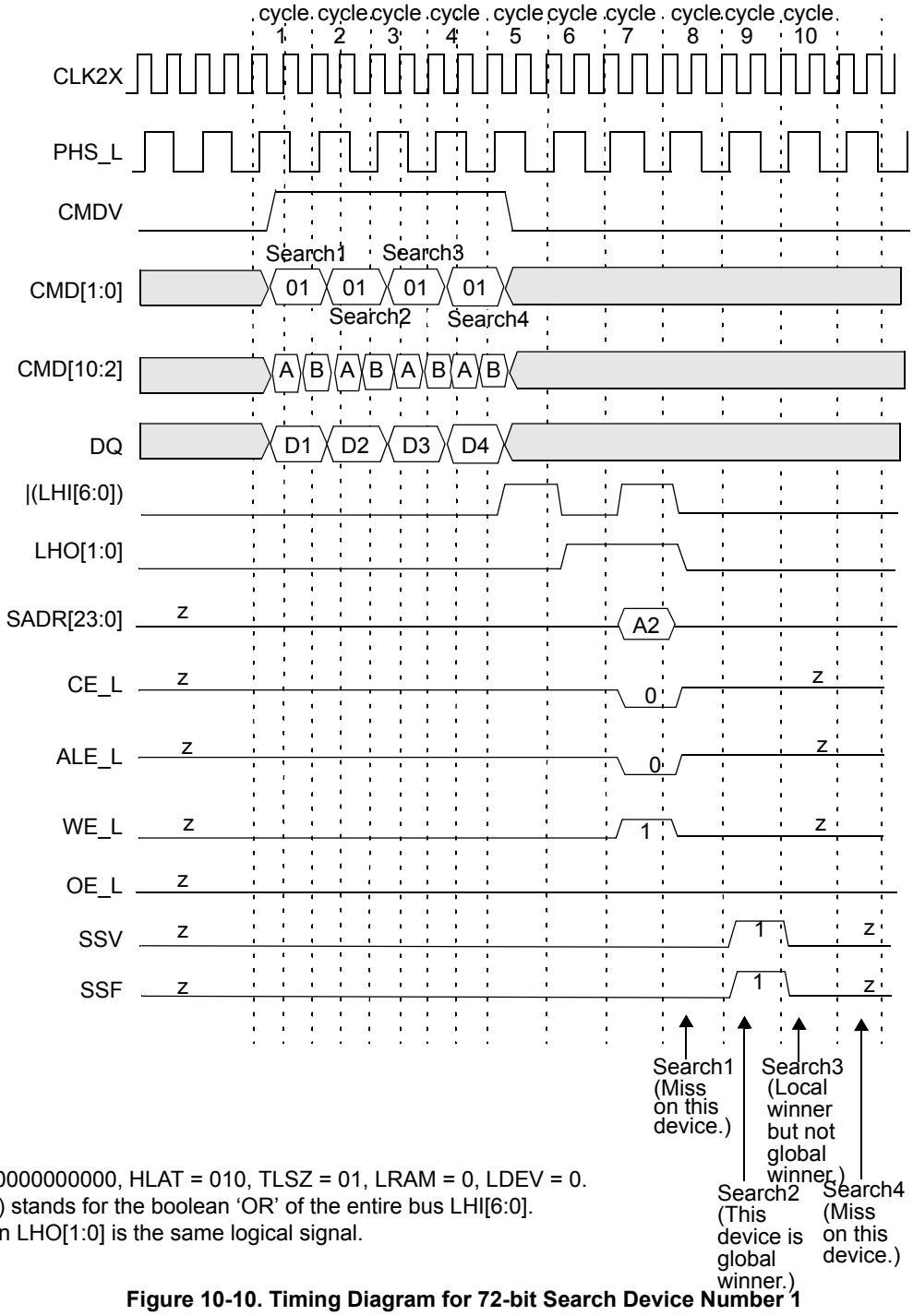


Figure 10-8. Hardware Diagram for a Table with Eight Devices



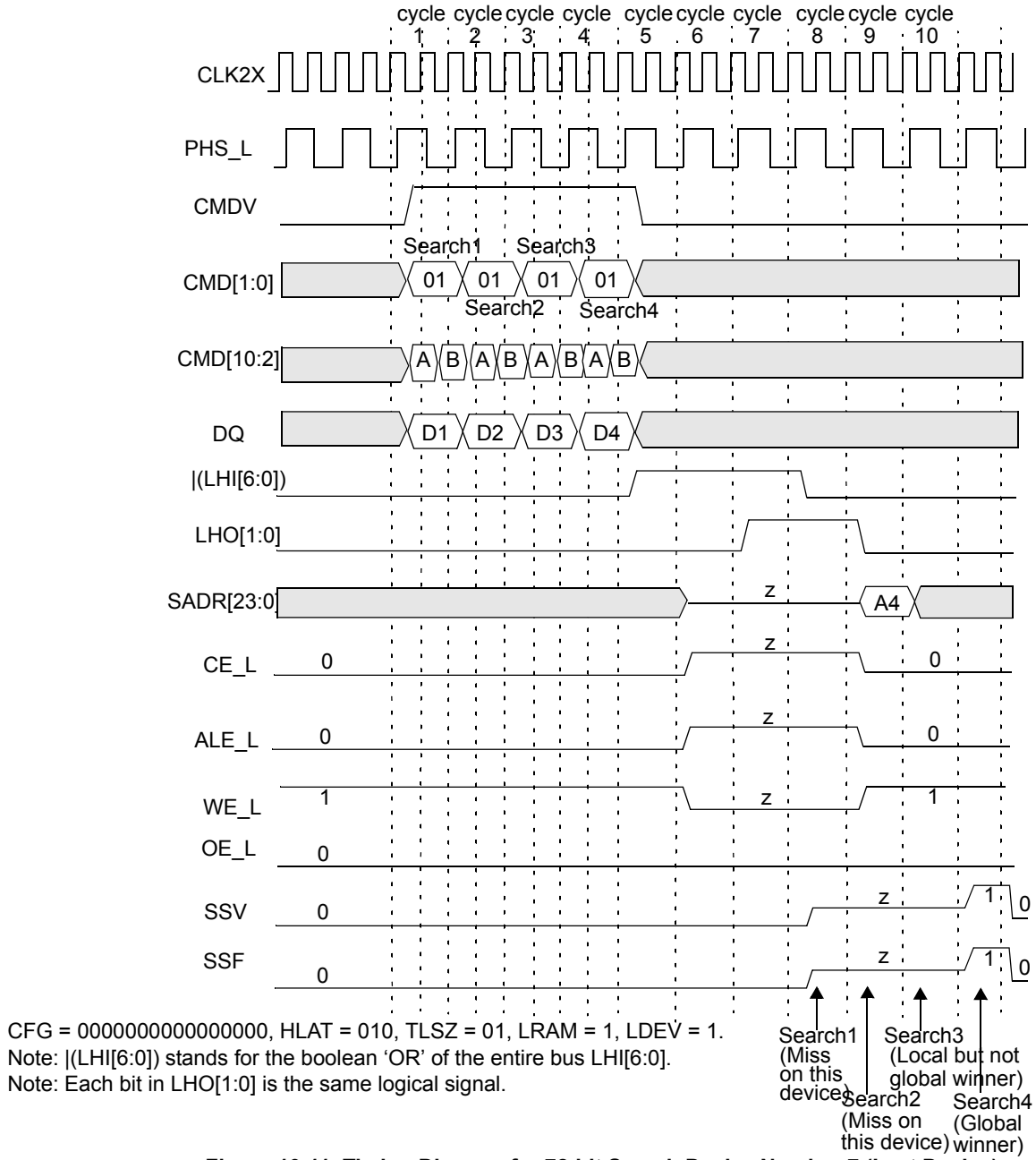
CFG = 0000000000000000, HLAT = 010, TLSZ = 01, LRAM = 0, LDEV = 0.
Note: |(LHI[6:0]) stands for the boolean 'OR' of the entire bus LHI[6:0].
Note: Each bit in LHO[1:0] is the same logical signal.

Figure 10-9. Timing Diagram for 72-bit Search Device Number 0



CFG = 0000000000000000, HLAT = 010, TLSZ = 01, LRAM = 0, LDEV = 0.
Note: |(LHI[6:0]) stands for the boolean 'OR' of the entire bus LHI[6:0].
Note: Each bit in LHO[1:0] is the same logical signal.

Figure 10-10. Timing Diagram for 72-bit Search Device Number 1



CFG = 0000000000000000, HLAT = 010, TLSZ = 01, LRAM = 1, LDEV = 1.
Note: |(LHI[6:0]) stands for the boolean 'OR' of the entire bus LHI[6:0].
Note: Each bit in LHO[1:0] is the same logical signal.

Figure 10-11. Timing Diagram for 72-bit Search Device Number 7 (Last Device)

The following is the sequence of operation for a single 72-bit Search command (also refer to "Command and Command Parameters," Subsection 10.2 on page 22).

- **Cycle A**: The host ASIC drives the CMDV high and applies Search command code (10) to CMD[1:0] signals. {CMD[10], CMD[5:3]} signals must be driven with the index to the GMR pair for use in this Search operation. CMD[8:6] signals must be driven with the same bits that will be driven on SADR[23:21] by this device if it has a hit. DQ[71:0] must be driven with the 72-bit data to be compared. The CMD[2] signal must be driven to logic 0.
- **Cycle B**: The host ASIC continues to drive the CMDV high and to apply Search command (10) on CMD[1:0]. CMD[5:2] must be driven by the index of the comparand register pair for storing the 144-bit word presented on the DQ bus during cycles A and B. CMD[8:6] signals must be driven with the index of the SSR that will be used for storing the address of the matching entry and hit flag (see page 8 for a description of SSR[0:7]). The DQ[71:0] continues to carry the 72-bit data to be compared.

Note. For 72-bit searches, the host ASIC must supply the same 72-bit data on DQ[71:0] during both cycles A and B, and the even and odd pairs of GMRs selected for the comparison must be programmed with the same value.

The logical 72-bit Search operation is shown in *Figure 10-12*. The entire table with eight devices of 72-bit entries is compared to a 72-bit word K (presented on the DQ bus in both cycles A and B of the command) using the GMR and the local mask bits. The effective GMR is the 72-bit word specified by the identical value in both even and odd GMR pairs in each of the eight devices and selected by the GMR Index in the command's cycle A. The 72-bit word K (presented on the DQ bus in both cycles A and B of the command) is also stored in both even and odd comparand register pairs (selected by the Comparand Register Index in command cycle B) in each of the eight devices. In the ×72 configuration, only the even comparand register can subsequently be used by the Learn command in one of the devices (only the first non-full device). The word K (presented on the DQ bus in both cycles A and B of the command) is compared with each entry in the table starting at location 0. The first matching entry's location address L is the winning address that is driven as part of the SRAM address on the SADR[23:0] lines (see "SRAM Addressing" on page 105). The global winning device will drive the bus in a specific cycle. On a global miss cycle the device with LRAM = 1 (default driving device for the SRAM bus) and LDEV = 1 (default driving device for SSF and SSV signals) will be the default driver for such missed cycles.

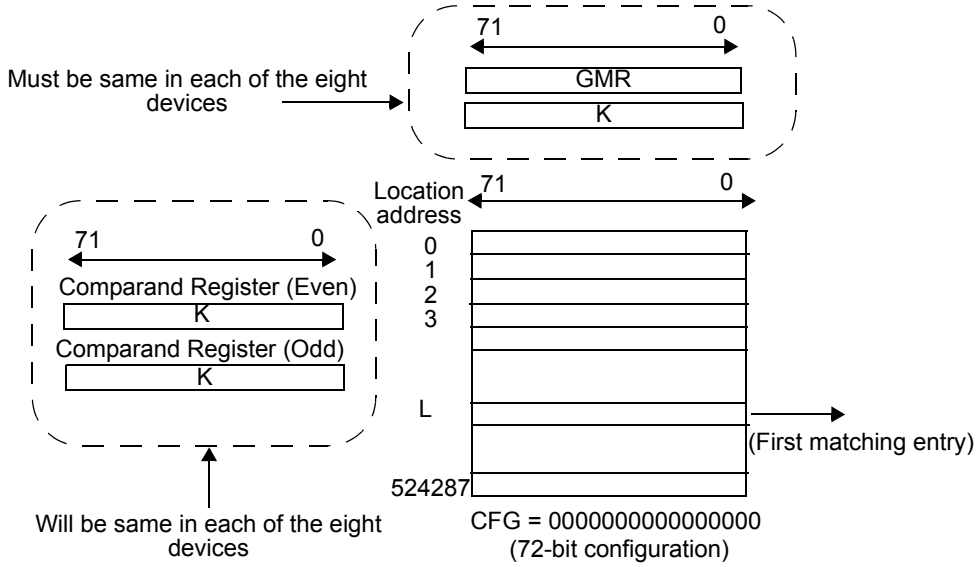


Figure 10-12. x72 Table with Eight Devices

The Search command is a pipelined operation and executes a search at half the rate of the frequency of CLK2X for 72-bit searches in x72-configured tables. The latency of SADR, CE_L, ALE_L, WE_L, SSV, and SSF from the 72-bit Search command cycle (two CLK2X cycles) is shown in *Table 10-13*.

Table 10-13. The Latency of Search from Instruction to SRAM Access Cycle

Number of Devices	Max Table Size	Latency in CLK Cycles
1 (TLSZ = 00)	64K × 72 bits	4
1–8 (TLSZ = 01)	512K × 72 bits	5
1–31 (TLSZ = 10)	1984K × 72 bits	6

The latency of the search from command to SRAM access cycle is 5 for up to eight devices in the table (TLSZ = 01). SSV and SSF also shift further to the right for different values of HLAT, as specified in *Table 10-14*.

Table 10-14. Shift OF SSF and SSV from SADR

HLAT	Number of CLK Cycles
000	0
001	1
010	2
011	3
100	4
101	5
110	6
111	7

10.6.3 72-bit Search on Tables Configured as x72 Using up to 31 CYNSE70128 Devices

The hardware diagram of the search subsystem of 31 devices is shown in *Figure 10-13*. Each of the four blocks in the diagram represents eight CYNSE70128 devices (except the last, which has seven devices). The diagram for a block of eight devices is shown in *Figure 10-14*. The following are the parameters programmed into the 31 devices.

- First thirty devices (devices 0–29): CFG = 0000000000000000, TLSZ = 10, HLAT = 001, LRAM = 0, and LDEV = 0.
- Thirty-first device (device 30): CFG = 0000000000000000, TLSZ = 10, HLAT = 001, LRAM = 1, and LDEV = 1.

Note. All 31 devices must be programmed with the same values for TLSZ and HLAT. Only the last device in the table must be programmed with LRAM = 1 and LDEV = 1 (device number 30 in this case). All other upstream devices must be programmed with LRAM = 0 and LDEV = 0 (devices 0 through 29 in this case).

The timing diagrams referred to in this paragraph reference the Hit/Miss assumptions defined in *Table 10-15*. For the purpose of illustrating the timings, it is further assumed that the there is only one device with a matching entry in each of the blocks. *Figure 10-15* shows the timing diagram for a Search command in the 72-bit-configured table of 31 devices for each of the eight devices in block 0. *Figure 10-16* shows a timing diagram for a Search command in the 72-bit-configured table of 31 devices for the all the devices in block number 1 (above the winning device in that block). *Figure 10-17* shows the timing diagram for the globally winning device (defined as the final winner within its own and all blocks) in block number 1. *Figure 10-18* shows the timing diagram for all the devices below the globally winning device in block number 1. *Figure 10-19*, *Figure 10-20*, and *Figure 10-21* show the timing diagrams of the devices above the globally winning device, the globally winning device, and the devices below the globally winning device, respectively, for block number 2. *Figure 10-22*, *Figure 10-23*,*Figure 10-24*, and *Figure 10-25* show the timing diagrams of the devices above globally winning device, the globally winning device, and the devices below the globally winning device except the last device (device 30), respectively, for block number 3.

The 72-bit Search operation is pipelined and executes as follows. Four cycles from the Search command, each of the devices knows the outcome internal to it for that operation. In the fifth cycle after the Search command, the devices in a block arbitrate for a winner amongst them (a “block” being defined as less than or equal to eight devices resolving the winner within them using the LHI[6:0] and LHO[1:0] signalling mechanism). In the sixth cycle after the Search command, the blocks (of devices) resolve the winning block through the BHI[2:0] and BHO[2:0] signalling mechanism. The winning device within the winning block is the global winning device for a Search operation.

Table 10-15. Hit/Miss Assumption

Search Number	1	2	3	4
Block 0	Miss	Miss	Miss	Miss
Block 1	Miss	Miss	Hit	Miss
Block 2	Miss	Hit	Hit	Miss
Block 3	Hit	Hit	Miss	Miss

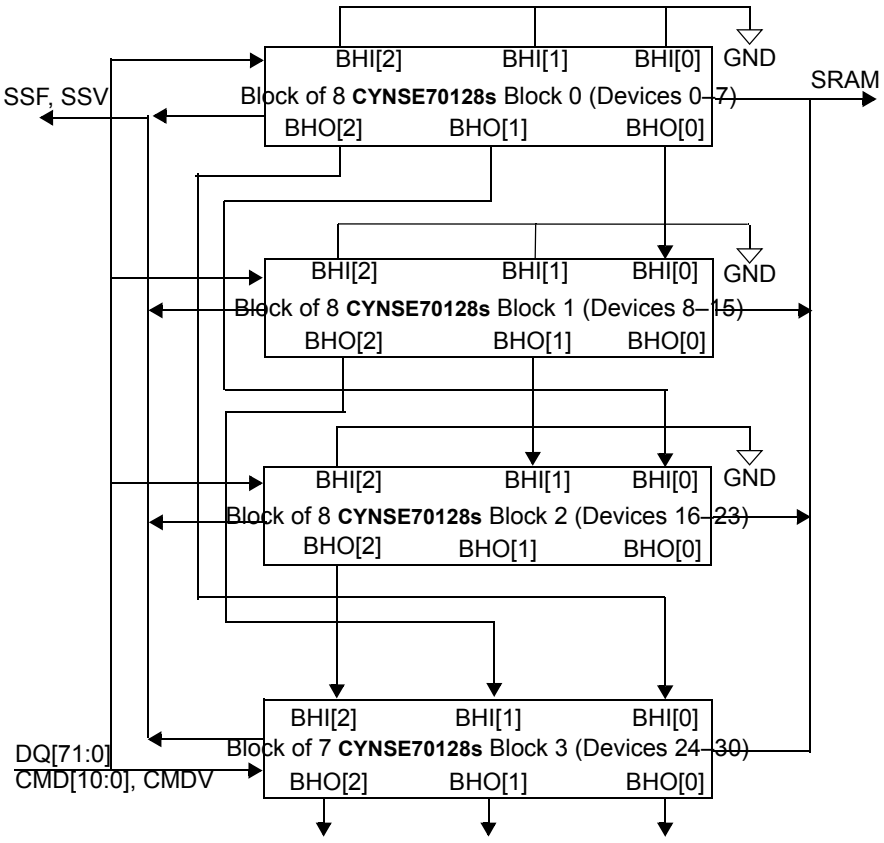


Figure 10-13. Hardware Diagram for a Table with 31 Devices

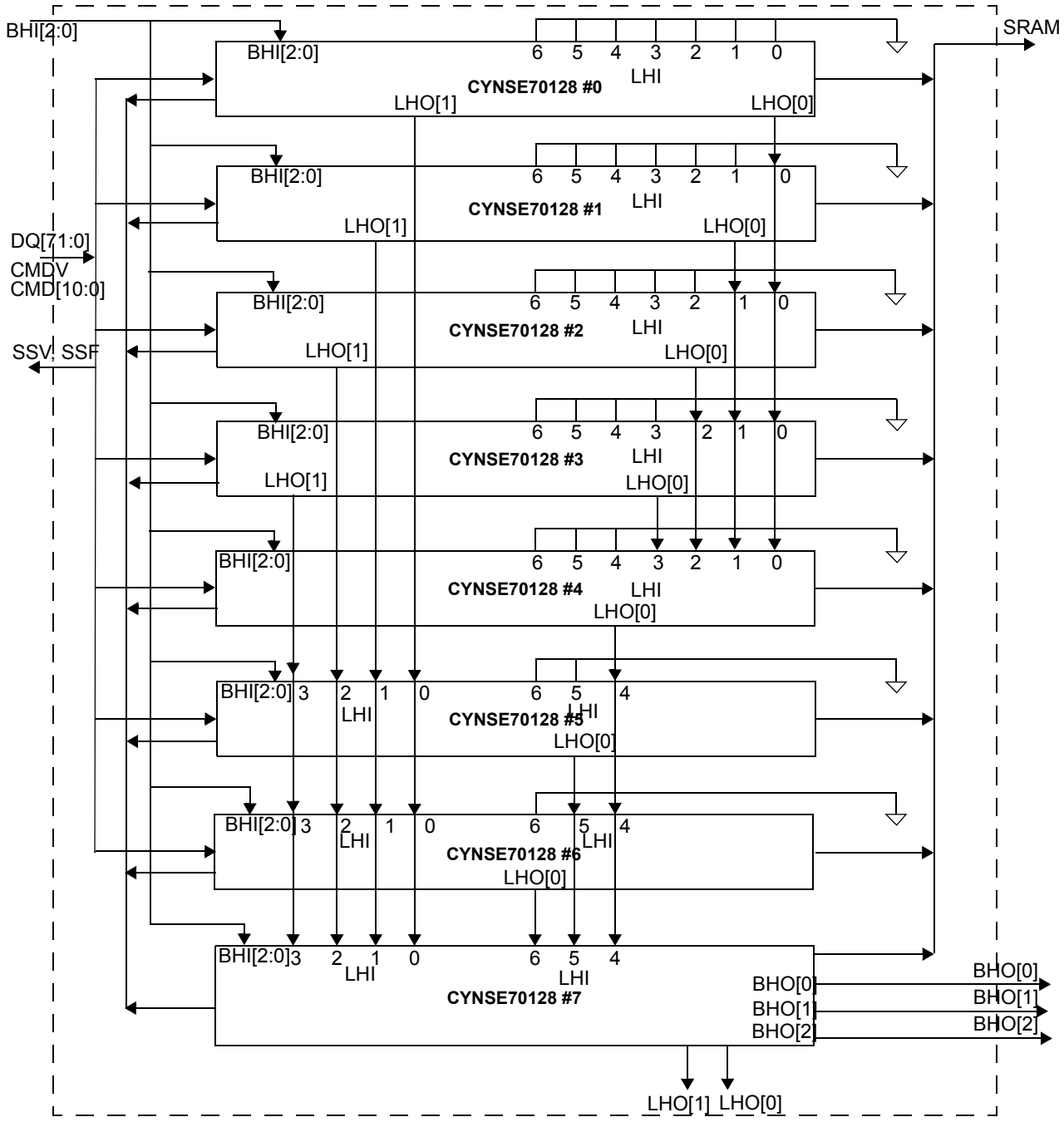
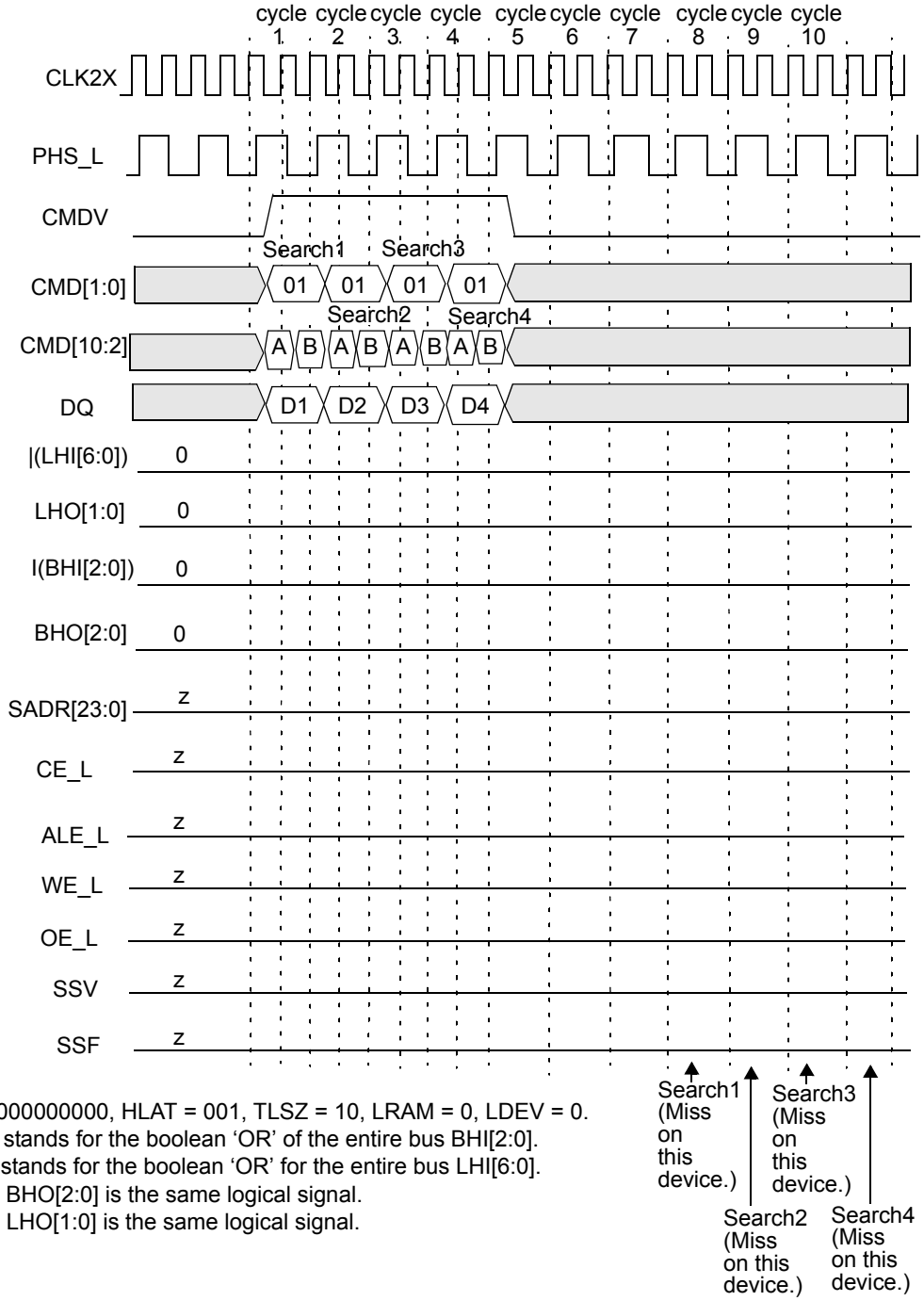
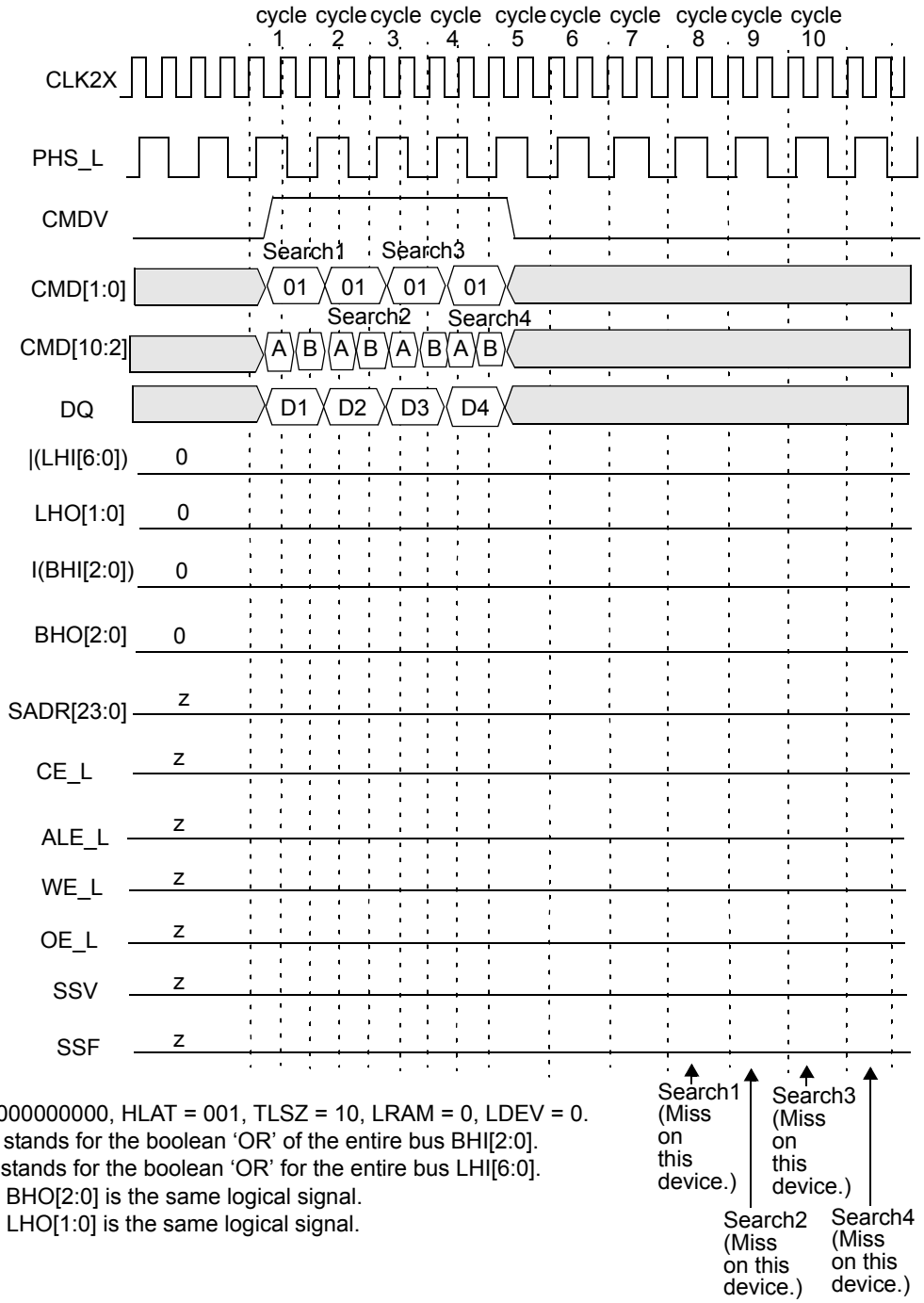


Figure 10-14. Hardware Diagram for a Block of up to Eight Devices



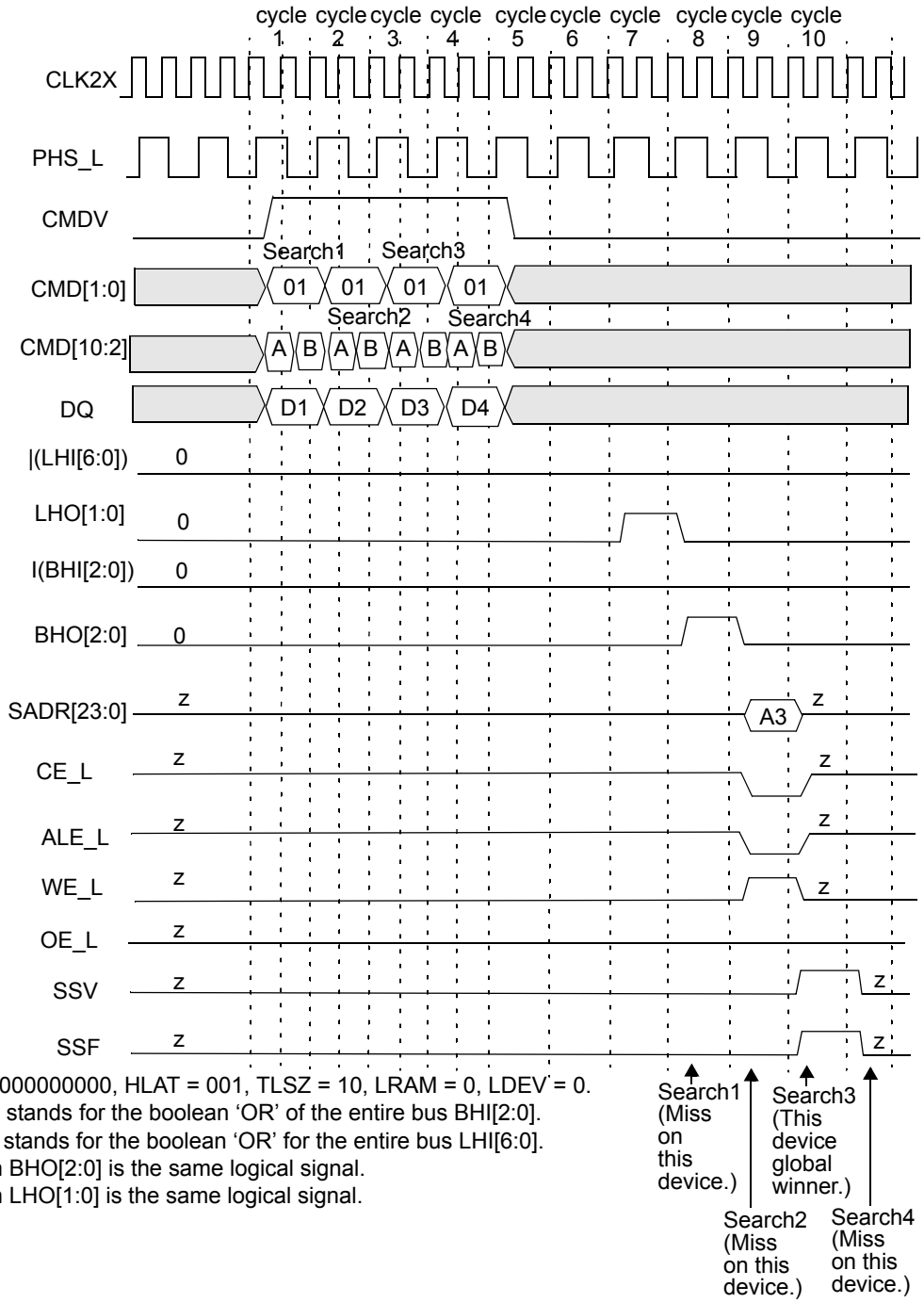
CFG = 0000000000000000, HLAT = 001, TLSZ = 10, LRAM = 0, LDEV = 0.
Note: |(BHI[2:0]) stands for the boolean 'OR' of the entire bus BHI[2:0].
Note: |(LHI[6:0]) stands for the boolean 'OR' for the entire bus LHI[6:0].
Note: Each bit in BHO[2:0] is the same logical signal.
Note: Each bit in LHO[1:0] is the same logical signal.

Figure 10-15. Timing Diagram for Each Device in Block Number 0 (Miss on Each Device)



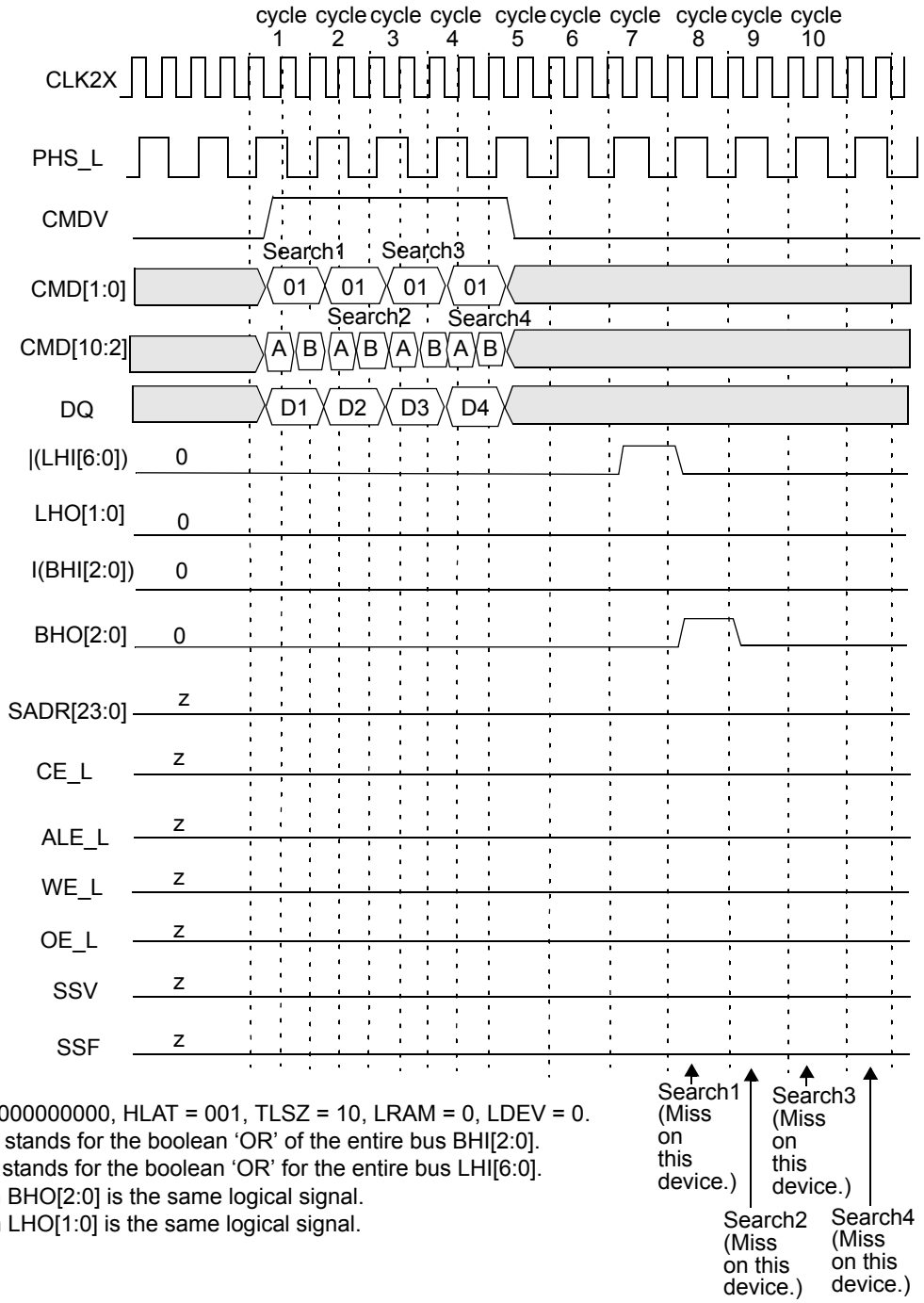
CFG = 0000000000000000, HLAT = 001, TLSZ = 10, LRAM = 0, LDEV = 0.
Note: |(BHI[2:0)] stands for the boolean 'OR' of the entire bus BHI[2:0].
Note: |(LHI[6:0]) stands for the boolean 'OR' for the entire bus LHI[6:0].
Note: Each bit in BHO[2:0] is the same logical signal.
Note: Each bit in LHO[1:0] is the same logical signal.

Figure 10-16. Timing Diagram for Each Device Above the Winning Device in Block Number 1



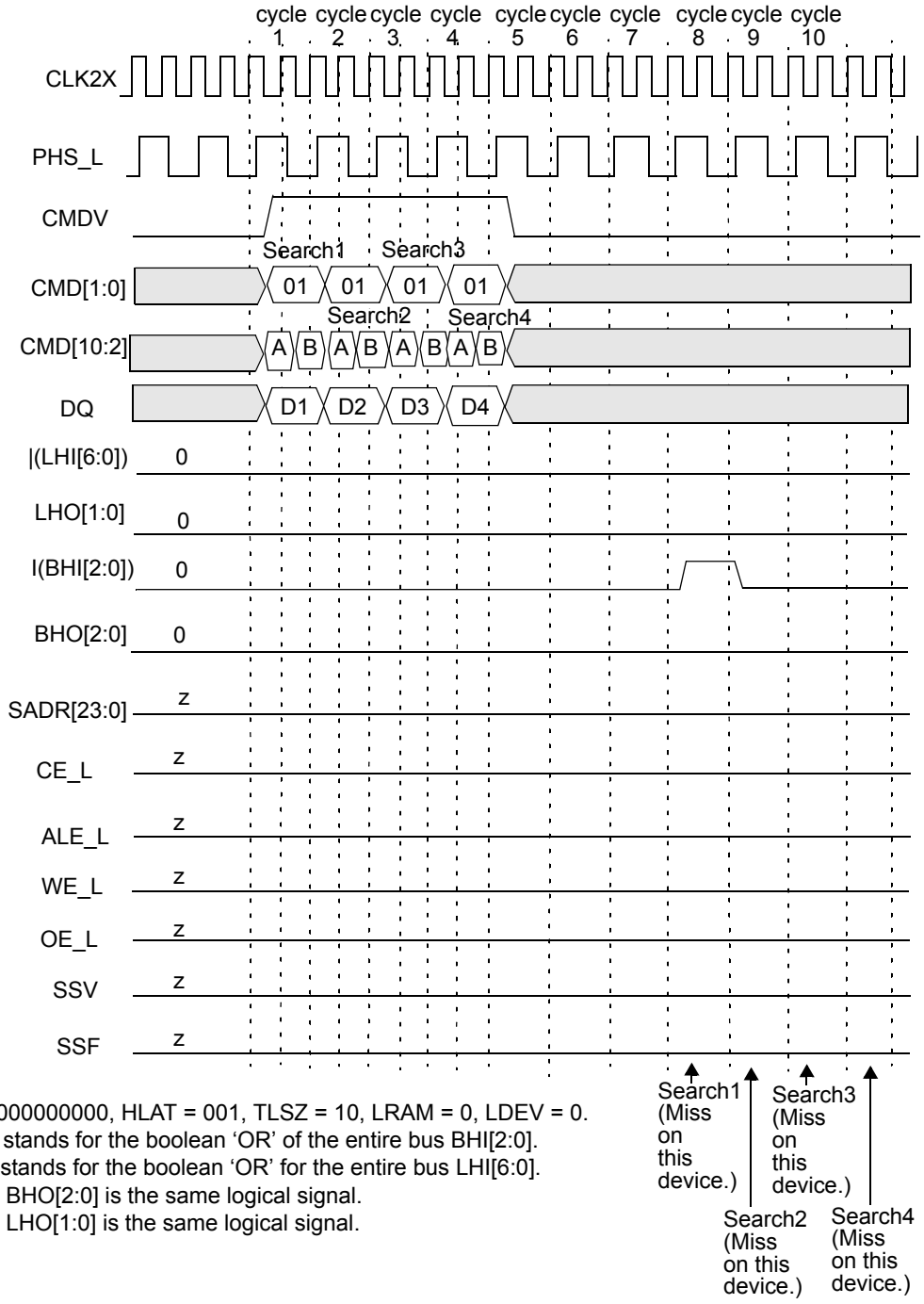
CFG = 0000000000000000, HLAT = 001, TLSZ = 10, LRAM = 0, LDEV = 0.
Note: |(BHI[2:0]) stands for the boolean 'OR' of the entire bus BHI[2:0].
Note: |(LHI[6:0]) stands for the boolean 'OR' for the entire bus LHI[6:0].
Note: Each bit in BHO[2:0] is the same logical signal.
Note: Each bit in LHO[1:0] is the same logical signal.

Figure 10-17. Timing Diagram for Globally Winning Device in Block Number 1



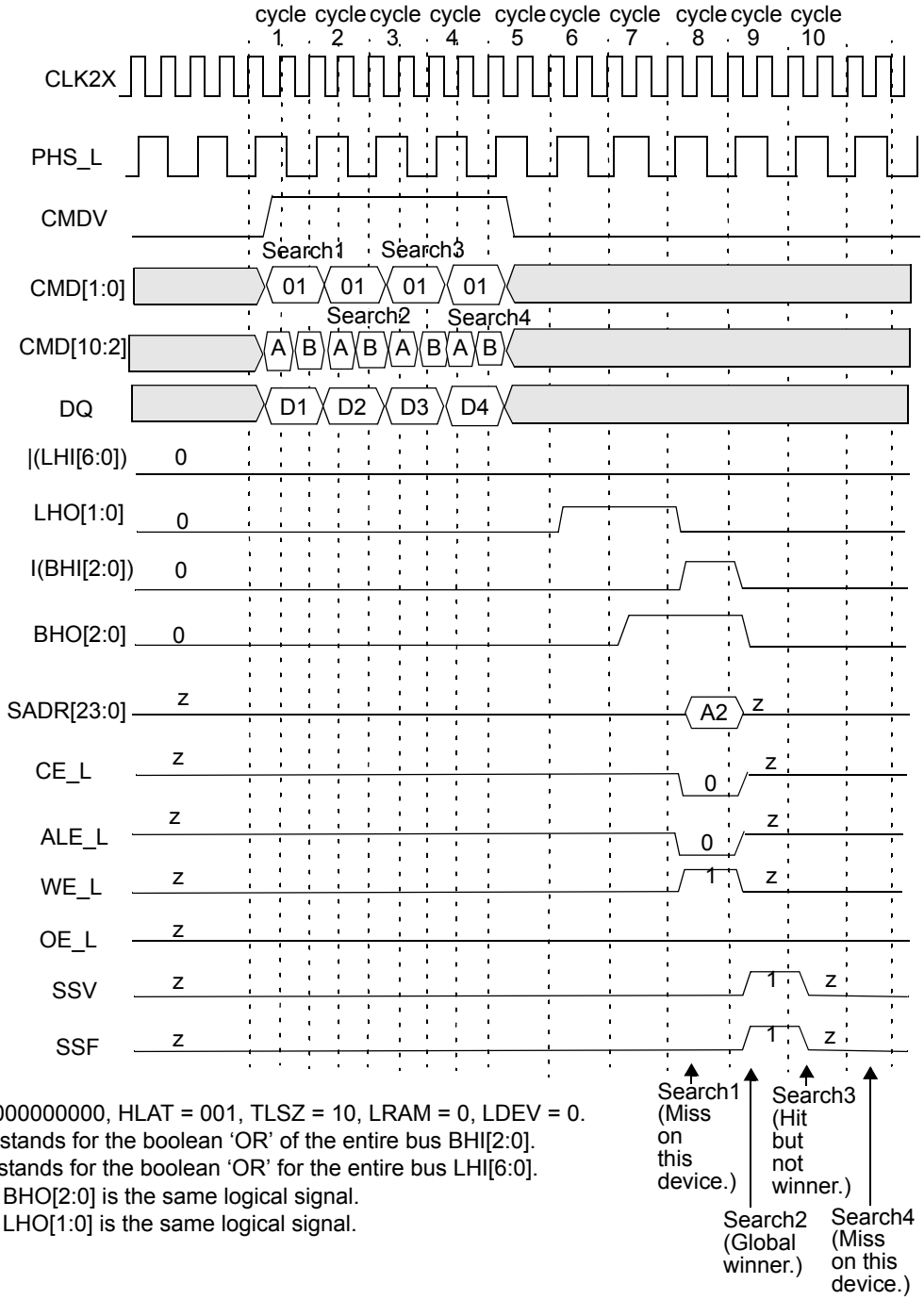
CFG = 0000000000000000, HLAT = 001, TLSZ = 10, LRAM = 0, LDEV = 0.
Note: |(BHI[2:0]) stands for the boolean 'OR' of the entire bus BHI[2:0].
Note: |(LHI[6:0]) stands for the boolean 'OR' for the entire bus LHI[6:0].
Note: Each bit in BHO[2:0] is the same logical signal.
Note: Each bit in LHO[1:0] is the same logical signal.

Figure 10-18. Timing Diagram for Devices Below the Winning Device in Block Number 1



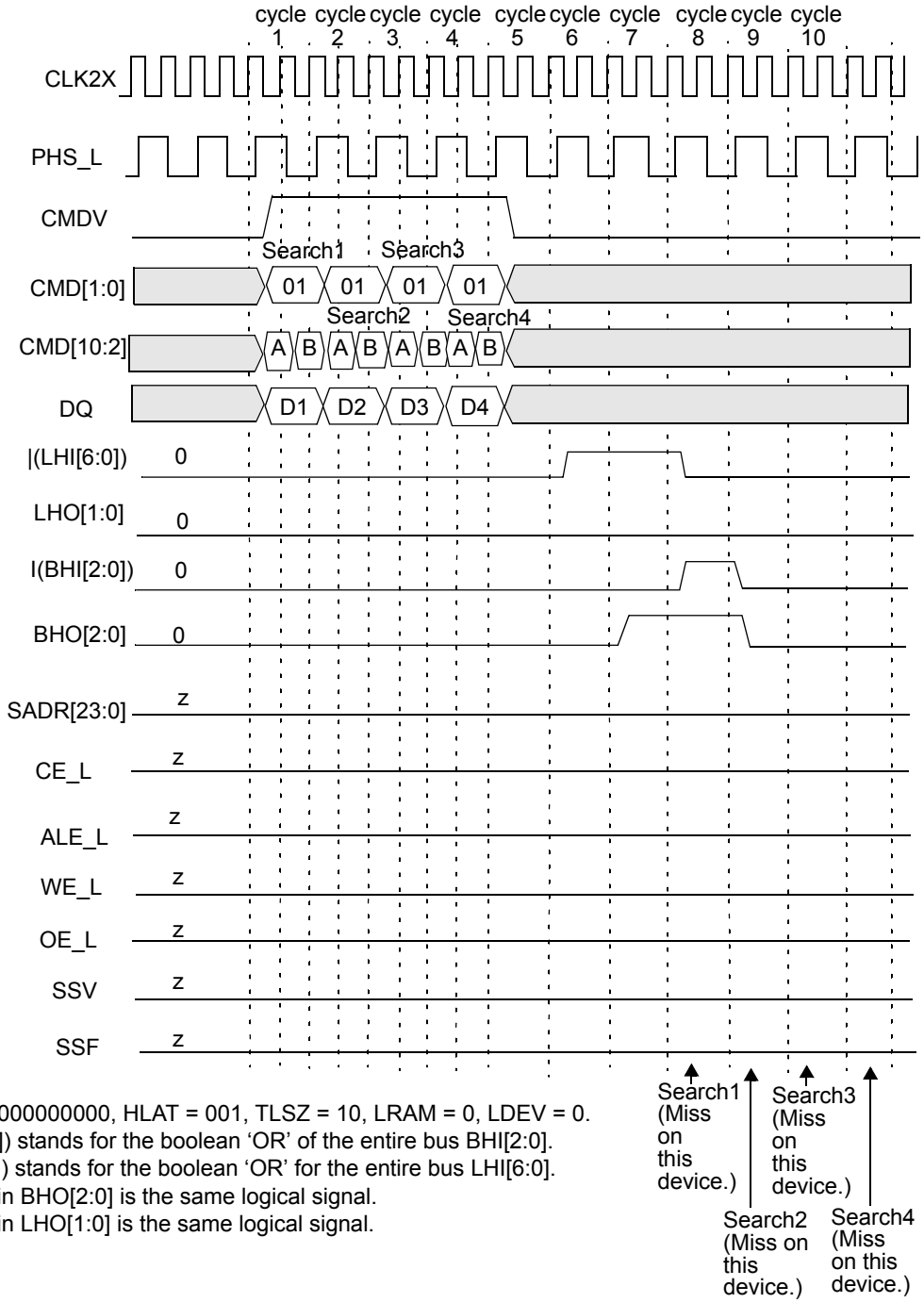
CFG = 0000000000000000, HLAT = 001, TLSZ = 10, LRAM = 0, LDEV = 0.
Note: |(BHI[2:0]) stands for the boolean 'OR' of the entire bus BHI[2:0].
Note: |(LHI[6:0]) stands for the boolean 'OR' for the entire bus LHI[6:0].
Note: Each bit in BHO[2:0] is the same logical signal.
Note: Each bit in LHO[1:0] is the same logical signal.

Figure 10-19. Timing Diagram for Devices Above the Winning Device in Block Number 2



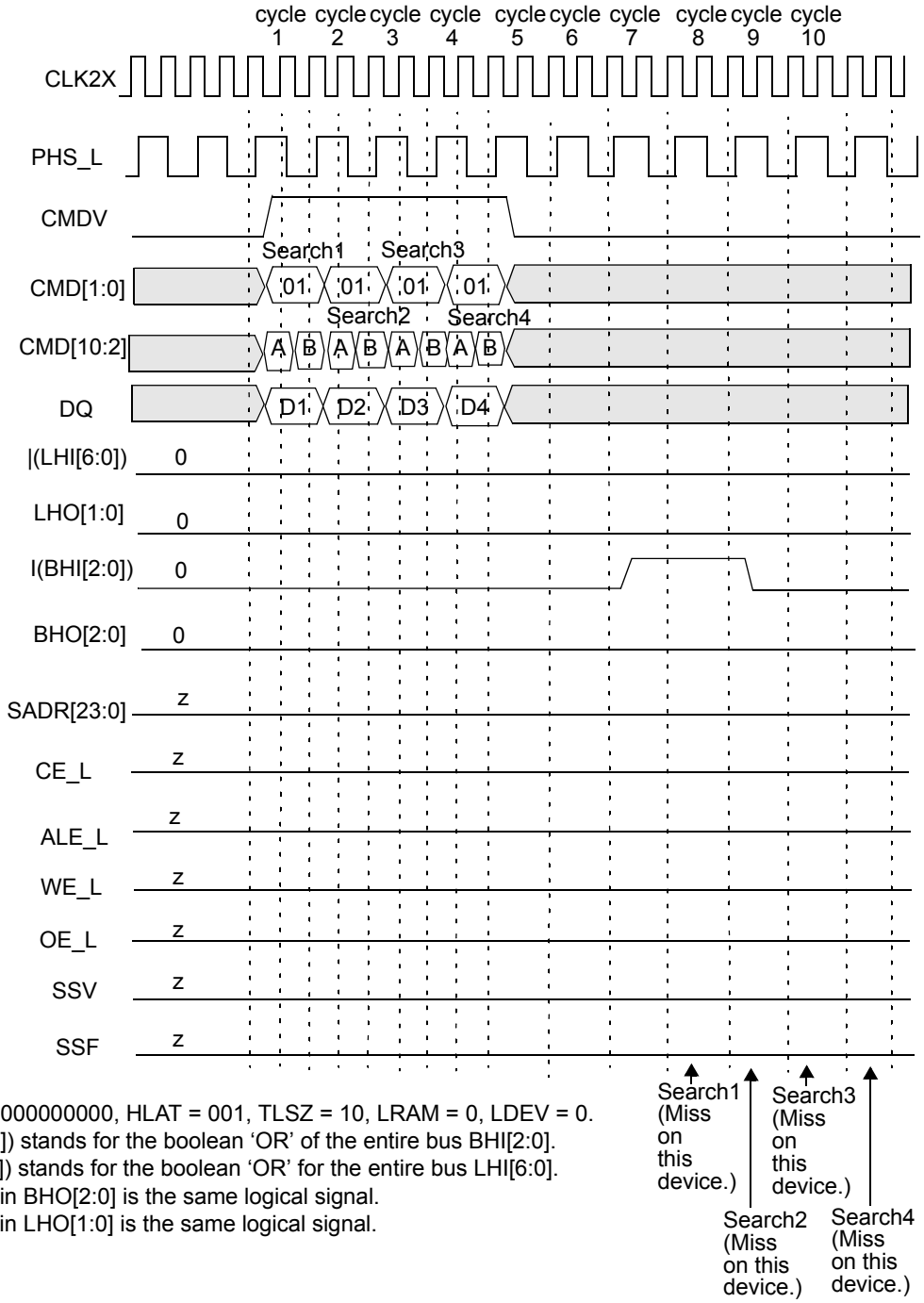
CFG = 0000000000000000, HLAT = 001, TLSZ = 10, LRAM = 0, LDEV = 0.
Note: |(BHI[2:0]) stands for the boolean 'OR' of the entire bus BHI[2:0].
Note: |(LHI[6:0]) stands for the boolean 'OR' for the entire bus LHI[6:0].
Note: Each bit in BHO[2:0] is the same logical signal.
Note: Each bit in LHO[1:0] is the same logical signal.

Figure 10-20. Timing Diagram for Globally Winning Device in Block Number 2



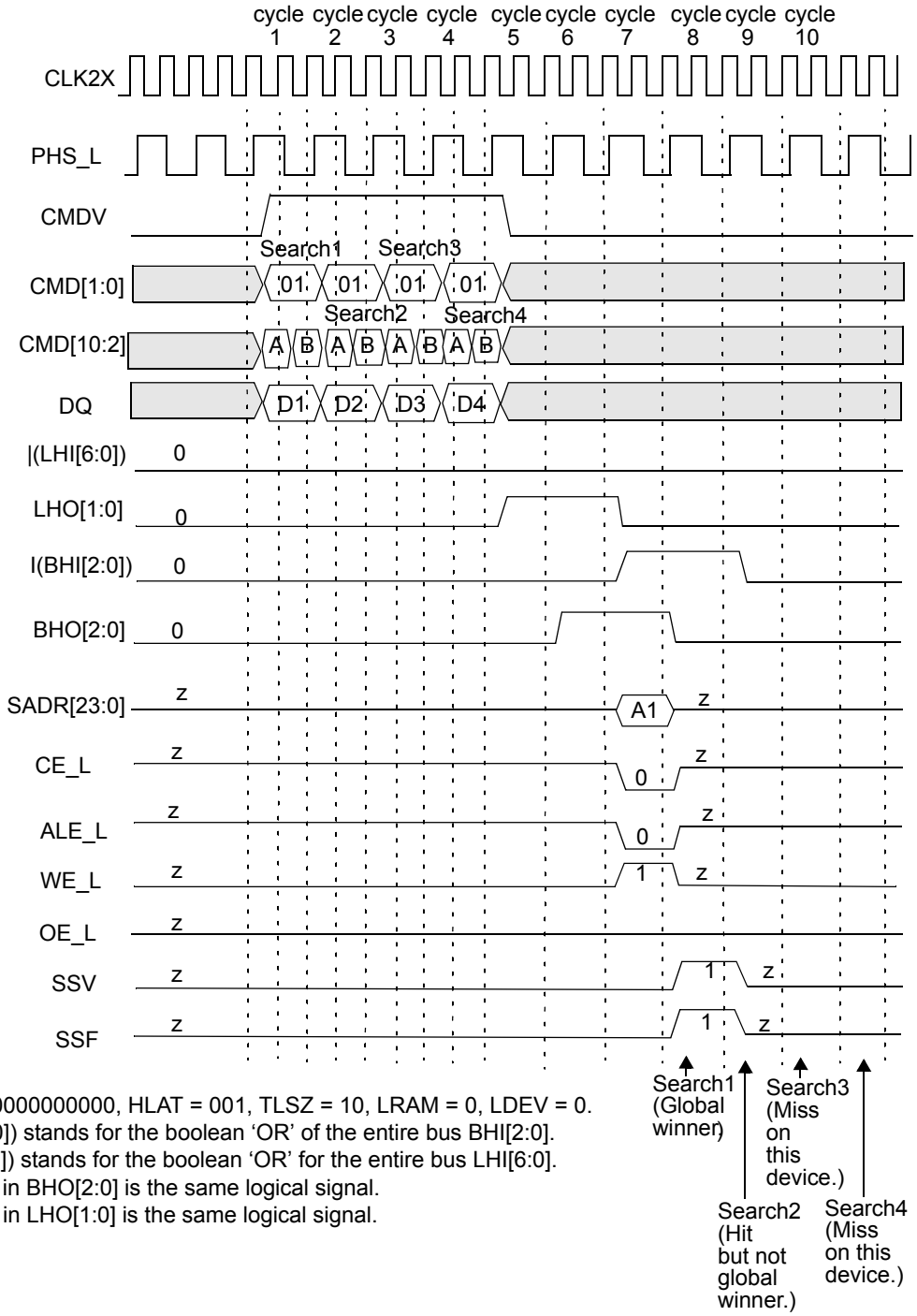
CFG = 0000000000000000, HLAT = 001, TLSZ = 10, LRAM = 0, LDEV = 0.
Note: |(BHI[2:0]) stands for the boolean 'OR' of the entire bus BHI[2:0].
Note: |(LHI[6:0]) stands for the boolean 'OR' for the entire bus LHI[6:0].
Note: Each bit in BHO[2:0] is the same logical signal.
Note: Each bit in LHO[1:0] is the same logical signal.

Figure 10-21. Timing Diagram for Devices Below the Winning Device in Block Number 2



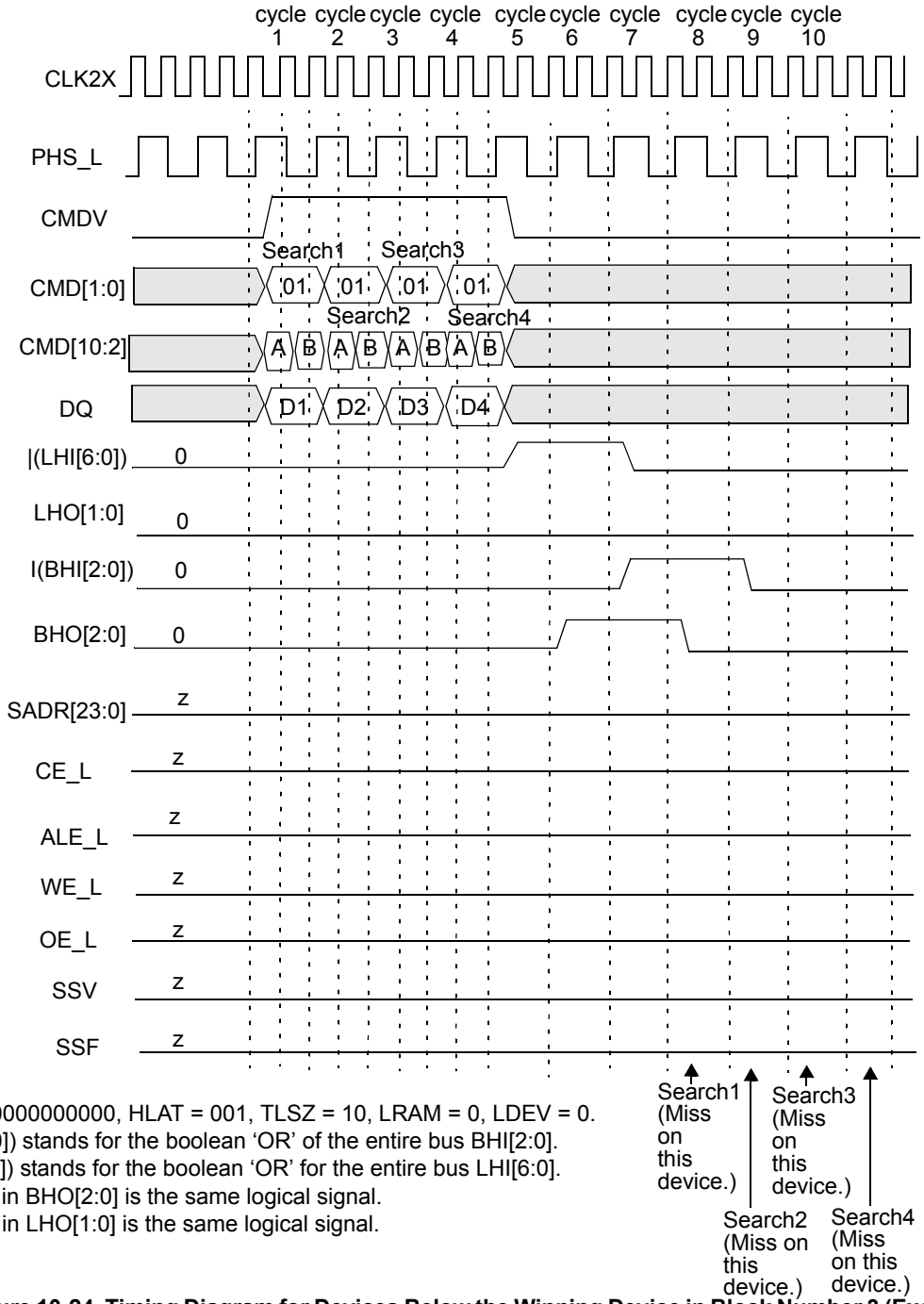
CFG = 0000000000000000, HLAT = 001, TLSZ = 10, LRAM = 0, LDEV = 0.
Note: |(BHI[2:0]) stands for the boolean 'OR' of the entire bus BHI[2:0].
Note: |(LHI[6:0]) stands for the boolean 'OR' for the entire bus LHI[6:0].
Note: Each bit in BHO[2:0] is the same logical signal.
Note: Each bit in LHO[1:0] is the same logical signal.

Figure 10-22. Timing Diagram for Devices Above the Winning Device in Block Number 3



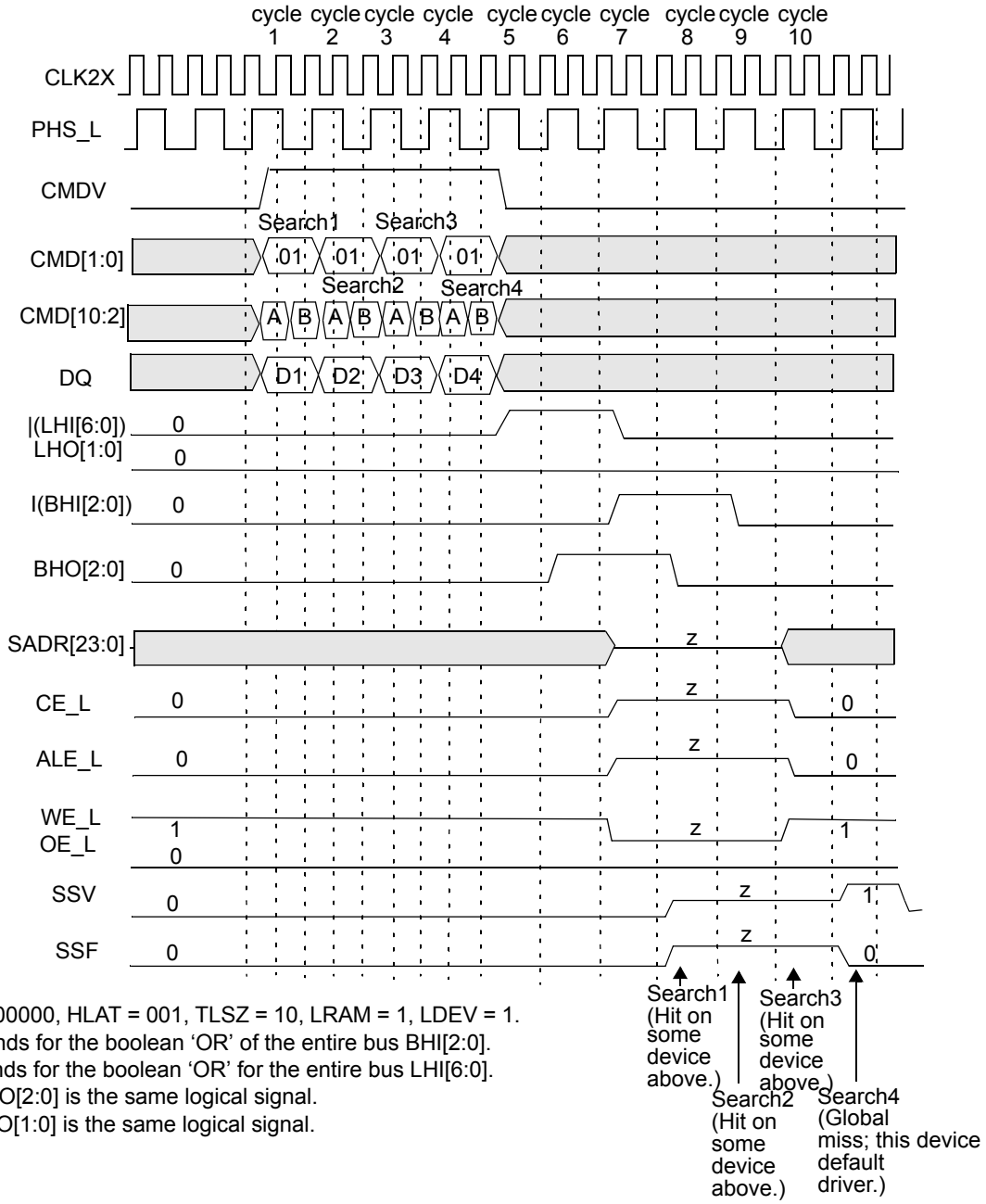
CFG = 0000000000000000, HLAT = 001, TLSZ = 10, LRAM = 0, LDEV = 0.
Note: |(BHI[2:0]) stands for the boolean 'OR' of the entire bus BHI[2:0].
Note: |(LHI[6:0]) stands for the boolean 'OR' for the entire bus LHI[6:0].
Note: Each bit in BHO[2:0] is the same logical signal.
Note: Each bit in LHO[1:0] is the same logical signal.

Figure 10-23. Timing Diagram for Globally Winning Device in Block Number 3



CFG = 0000000000000000, HLAT = 001, TLSZ = 10, LRAM = 0, LDEV = 0.
Note: |(BHI[2:0]) stands for the boolean 'OR' of the entire bus BHI[2:0].
Note: |(LHI[6:0]) stands for the boolean 'OR' for the entire bus LHI[6:0].
Note: Each bit in BHO[2:0] is the same logical signal.
Note: Each bit in LHO[1:0] is the same logical signal.

Figure 10-24. Timing Diagram for Devices Below the Winning Device in Block Number 3 (Except the Last Device [Device 30])



CFG = 0000000000000000, HLAT = 001, TLSZ = 10, LRAM = 1, LDEV = 1.
Note: |(BHI[2:0)] stands for the boolean 'OR' of the entire bus BHI[2:0].
Note: |(LHI[6:0]) stands for the boolean 'OR' for the entire bus LHI[6:0].
Note: Each bit in BHO[2:0] is the same logical signal.
Note: Each bit in LHO[1:0] is the same logical signal.

Figure 10-25. Timing Diagram for Device Number 6 in Block Number 3 (Device 30 in Depth-Cascaded Table)

The following is the sequence of operation for a single 72-bit Search command (also refer to the "Command and Command Parameters," on page 22).

- **Cycle A**: The host ASIC drives the CMDV high and applies Search command code (10) on CMD[1:0] signals. {CMD[10], CMD[5:3]} signals must be driven with the index to the GMR pair for use in this Search operation. CMD[8:6] signals must be driven with the same bits that will be driven on SADR[23:21] by this device if it has a hit. DQ[71:0] must be driven with the 72-bit data to be compared. The CMD[2] signal must be driven to a logic 0.
- **Cycle B**: The host ASIC continues to drive the CMDV high and applies Search command (10) on CMD[1:0]. CMD[5:2] must be driven by the index of the comparand register pair for storing the 144-bit word presented on the DQ bus during cycles A and B. CMD[8:6] signals must be driven with the index of the SSR that will be used for storing the address of the matching entry and the hit flag (see page 16 for the description of SSR[0:7]). The DQ[71:0] continues to carry the 72-bit data to be compared.

Note. For 72-bit searches, the host ASIC must supply the same 72-bit data on DQ[71:0] during both cycles A and B and the even and odd pair of GMRs selected for the compare must be programmed with the same value.

The logical 72-bit Search operation is shown in *Figure 10-26*. The entire table (31 devices of 72-bit entries) is compared to a 72-bit word K (presented on the DQ bus in both cycles A and B of the command) using the GMR and the local mask bits. The effective GMR is the 72-bit word specified by the identical value in both even and odd GMR pairs in each of the eight devices and selected by the GMR Index in the command's cycle A. The 72-bit word K (presented on the DQ bus in both cycles A and B of the command) is also stored in both even and odd comparand register pairs in each of the eight devices and selected by the Comparand Register Index in command's cycle B. In the x72 configuration, the even comparand register can be subsequently used by the Learn command only in the first non-full device. The word K (presented on the DQ bus in both cycles A and B of the command) is compared with each entry in the table starting at location 0. The first matching entry's location address L is the winning address that is driven as part of the SRAM address on the SADR[23:0] lines (see "SRAM Addressing" on page 105). The global winning device will drive the bus in a specific cycle. On global miss cycles the device with LRAM = 1 and LDEV = 1 will be the default driver for such missed cycles.

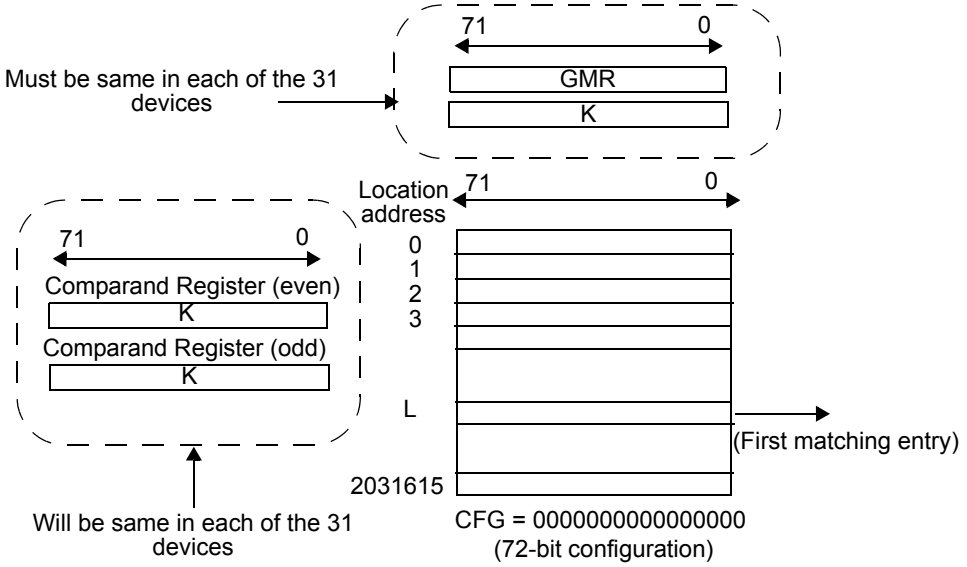


Figure 10-26. x72 Table with 31 Devices

The Search command is a pipelined operation and executes a search at half the rate of the frequency of CLK2X for 72-bit searches in x72-configured tables. The latency of SADR, CE_L, ALE_L, WE_L, SSV, and SSF from the 72-bit Search command cycle (two CLK2X cycles) is shown in *Table 10-16*.

Table 10-16. The Latency of Search from Instruction to SRAM Access Cycle

Number of Devices	Max Table Size	Latency in CLK Cycles
1 (TLSZ = 00)	64K × 72 bits	4
1–8 (TLSZ = 01)	512K × 72 bits	5
1–31 (TLSZ = 10)	1984K × 72 bits	6

For up to 31 devices in the table (TLSZ = 10), search latency from command to SRAM access cycle is 6. In addition, SSV and SSF shift further to the right for different values of HLAT, as specified in *Table 10-17*.

Table 10-17. Shift of SSF and SSV from SADR

HLAT	Number of CLK Cycles
000	0
001	1
010	2
011	3
100	4
101	5
110	6
111	7

10.6.4 144-bit Search on Tables Configured as x144 Using a Single CYNSE70128 Device

Figure 10-27 shows the timing diagram for a Search command in the 144-bit-configured table (CFG = 0101010101010101) consisting of a single device for one set of parameters: TLSZ = 00, HLAT = 001, LRAM = 1, and LDEV = 1. The hardware diagram for this search subsystem is shown in *Figure 10-28*.

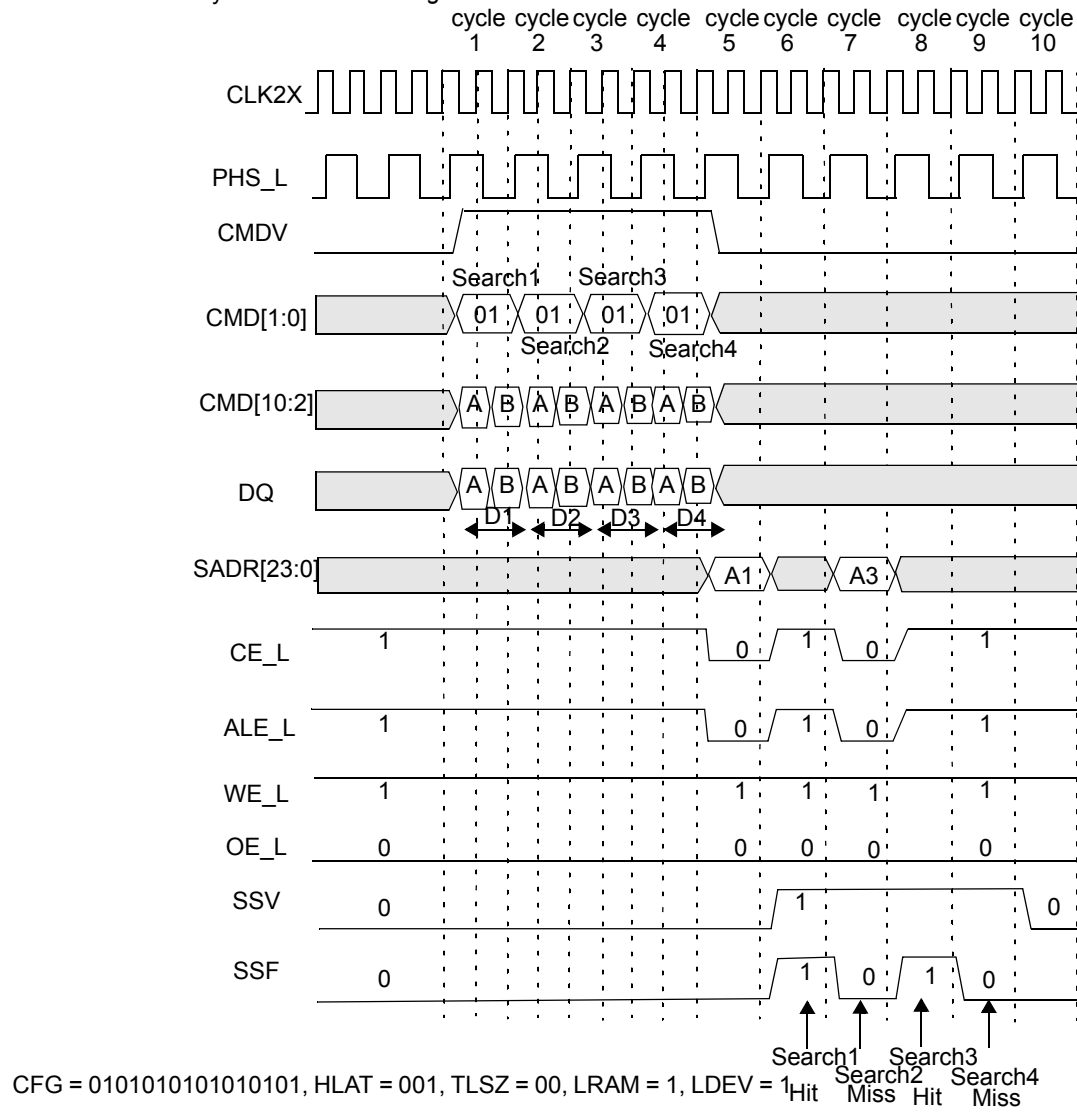


Figure 10-27. Timing Diagram for 144-bit Search (One Device)

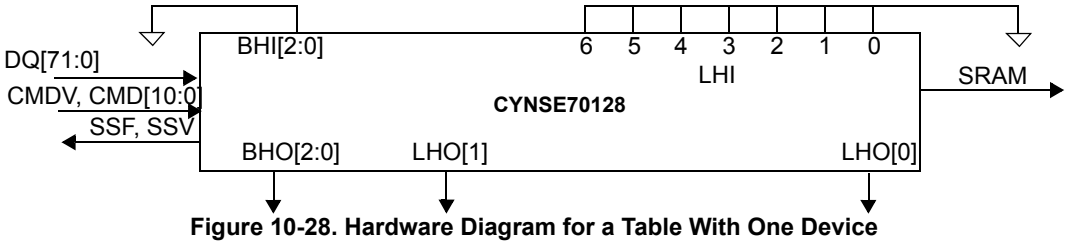


Figure 10-28. Hardware Diagram for a Table With One Device

The following is the operation sequence for a single 144-bit Search command (also refer to “Command and Command Parameters,” on page 22).

- **Cycle A**: The host ASIC drives the CMDV high and applies Search command code (10) to CMD[1:0] signals. {CMD[10],CMD[5:3]} signals must be driven with the index to the GMR pair for use in this Search operation. CMD[8:6] signals must be driven with the same bits that will be driven on SADR[23:21] by this device if it has a hit. DQ[71:0] must be driven with the 72-bit data ([143:72]) to be compared against all even locations. The CMD[2] signal must be driven to logic 0.

- **Cycle B**: The host ASIC continues to drive the CMDV high and applies the command code of Search command (10) on CMD[1:0]. CMD[5:2] must be driven by the index of the comparand register pair for storing the 144-bit word presented on the DQ bus during cycles A and B. CMD[8:6] signals must be driven with the index of the SSR that will be used for storing the address of the matching entry and hit flag (see page 8 for the description of SSR[0:7]). The DQ[71:0] is driven with 72-bit data ([71:0]), compared to all odd locations.

Note. For 144-bit searches, the host ASIC must supply two distinct 72-bit data words on DQ[71:0] during cycles A and B. The even-numbered GMR of the pair specified by the GMR Index is used for masking the word in cycle A. The odd-numbered GMR of the pair specified by the GMR Index is used for masking the word in cycle B.

The logical 144-bit search operation is shown in *Figure 10-29*. The entire table of 144-bit entries is compared to a 144-bit word K (presented on the DQ bus in cycles A and B of the command) using the GMR and the local mask bits. The GMR is the 144-bit word specified by the even and odd global mask pair selected by the GMR Index in the command's cycle A. The 144-bit word K (presented on the DQ bus in cycles A and B of the command) is also stored in both even and odd comparand register pairs selected by the Comparand Register Index in the command's cycle B. The two comparand registers can subsequently be used by the Learn command with the even comparand register stored in an even location, and the odd comparand register stored in an adjacent odd location. The word K (presented on the DQ bus in cycles A and B of the command) is compared with each entry in the table starting at location 0. The first matching entry's location address L is the winning address that is driven as part of the SRAM address on the SADR[23:0] lines (see "SRAM Addressing" on page 105). ***Note***. The matching address is always going to an even address for a 144-bit Search.

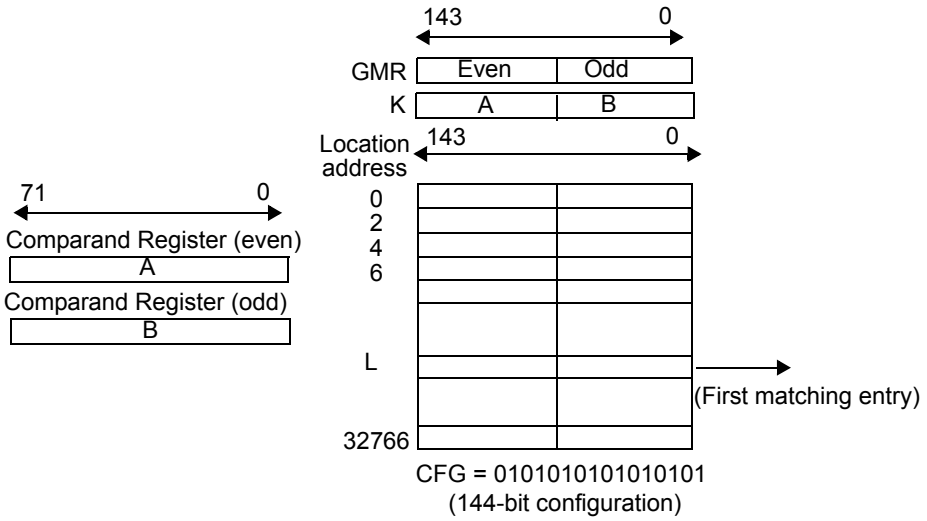


Figure 10-29. x144 Table with One Device

The Search command is a pipelined operation that executes searches at half the rate of the frequency of CLK2X for 144-bit searches in x144-configured tables. The latency of SADR, CE_L, ALE_L, WE_L, SSV, and SSF from the 144-bit Search command cycle (two CLK2X cycles) is shown in *Table 10-18*.

Table 10-18. The Latency of Search from Instruction to SRAM Access Cycle

Number of Devices	Max Table Size	Latency in CLK Cycles
1 (TLSZ = 00)	32K × 144 bits	4
1–8 (TLSZ = 01)	256K × 144 bits	5
1–31 (TLSZ = 10)	992K × 144 bits	6

For a single device in the table with TLSZ = 00, the latency of the Search from command to SRAM access cycle is 4. In addition, SSV and SSF shift further to the right for different values of HLAT, as specified in *Table 10-19*.

Table 10-19. Shift OF SSF and SSV from SADR

HLAT	Number of CLK Cycles
000	0
001	1
010	2
011	3

Table 10-19. Shift OF SSF and SSV from SADR (continued)

HLAT	Number of CLK Cycles
100	4
101	5
110	6
111	7

10.6.5 144-bit Search on Tables Configured as x144 Using up to Eight CYNSE70128 Devices

The hardware diagram of the search subsystem of eight devices is shown in *Figure 10-30*. The following are parameters programmed into the eight devices.

- First seven devices (devices 0–6): CFG = 0101010101010101, TLSZ = 01, HLAT = 010, LRAM = 0, and LDEV = 0.
- Eighth device (device 7): CFG = 0101010101010101, TLSZ = 01, HLAT = 010, LRAM = 1, and LDEV = 1.

Note. All eight devices must be programmed with the same value of TLSZ and HLAT. Only the last device in the table must be programmed with LRAM = 1 and LDEV = 1 (device number 7 in this case). All other upstream devices must be programmed with LRAM = 0 and LDEV = 0 (devices 0 through 6 in this case).

Figure 10-31 shows the timing diagram for a Search command in the 144-bit-configured table of eight devices for device 0. *Figure 10-32* shows the timing diagram for a Search command in the 144-bit-configured table consisting of eight devices for device number 1. *Figure 10-33* shows the timing diagram for a Search command in the 144-bit configured table consisting of eight devices for device number 7 (the last device in this specific table). For these timing diagrams, four 144-bit searches are performed sequentially, and the following Hit/Miss assumptions were made (see *Table 10-20*).

Table 10-20. Hit/Miss Assumption

Search Number	1	2	3	4
Device 0	Hit	Miss	Hit	Miss
Device 1	Miss	Hit	Hit	Miss
Device 2–6	Miss	Miss	Miss	Miss
Device 7	Miss	Miss	Hit	Hit

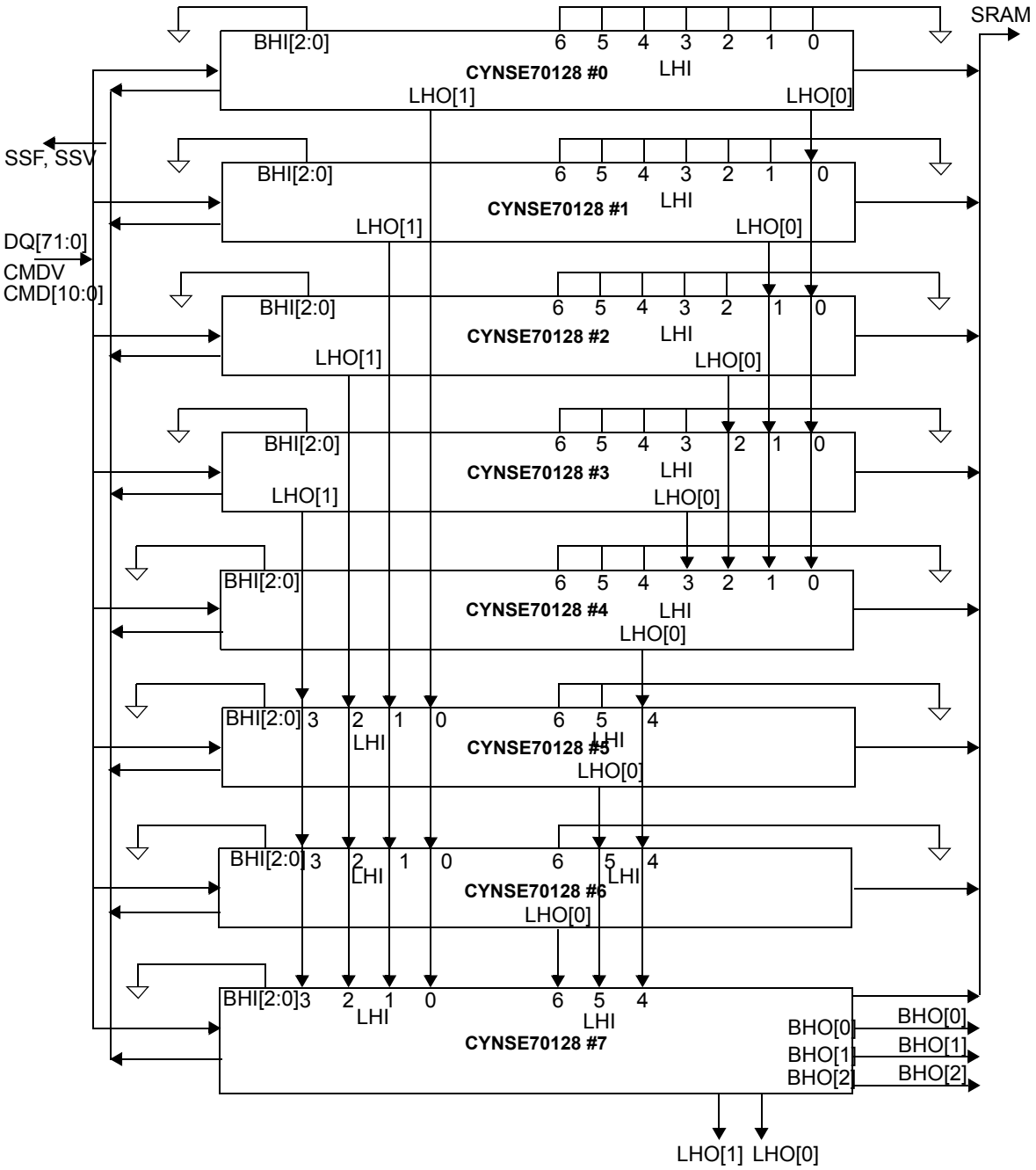
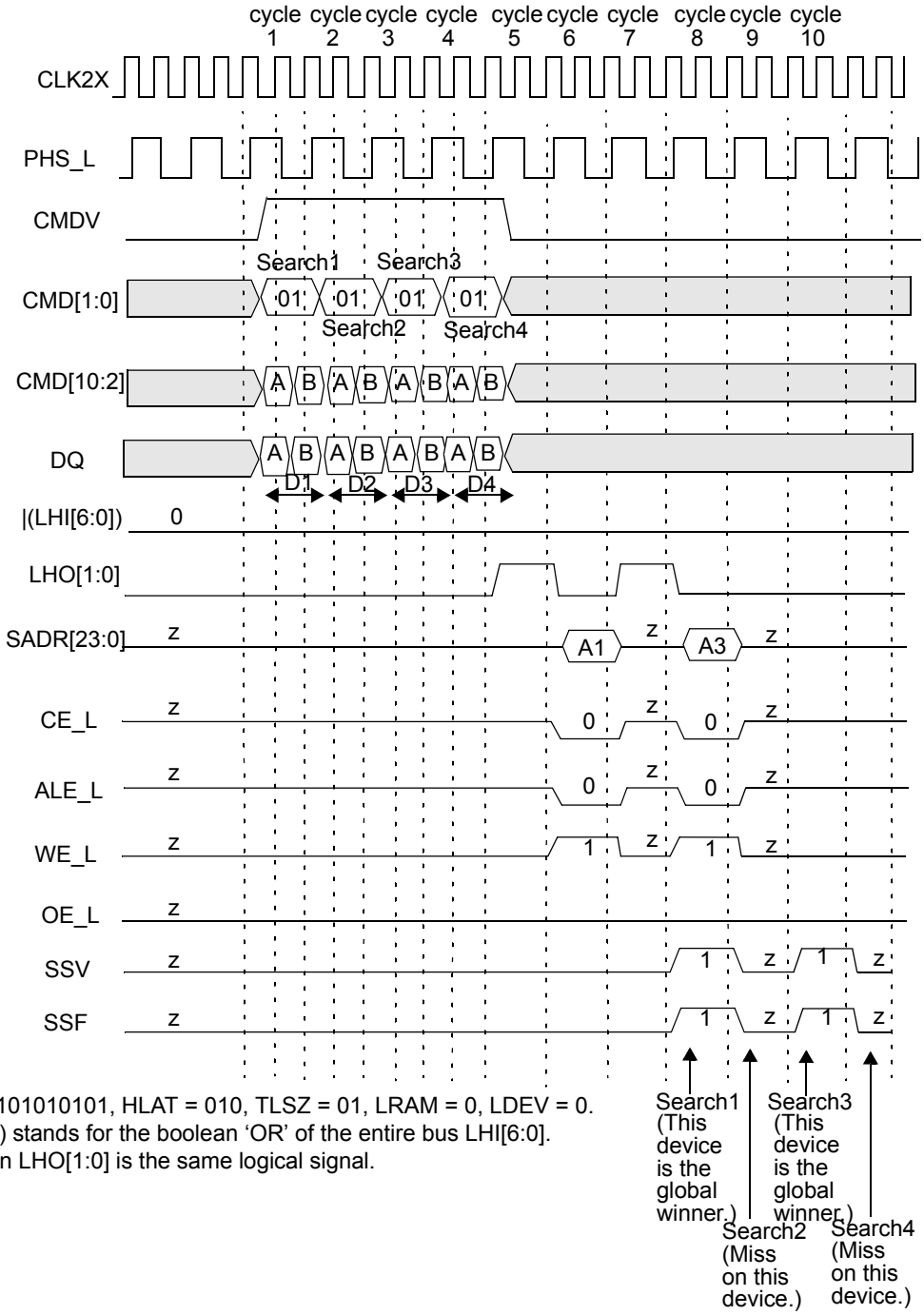
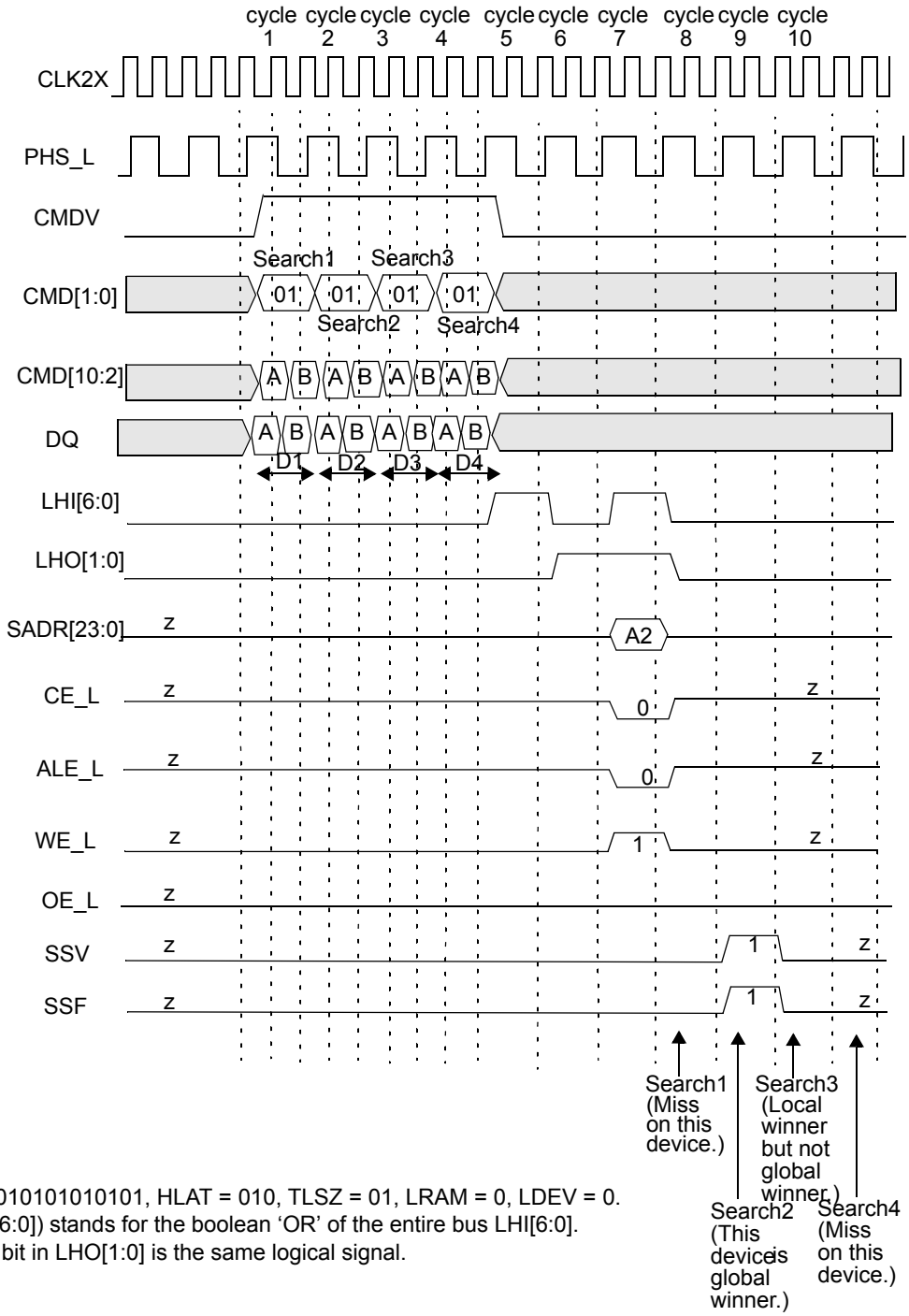


Figure 10-30. Hardware Diagram for a Table with Eight Devices



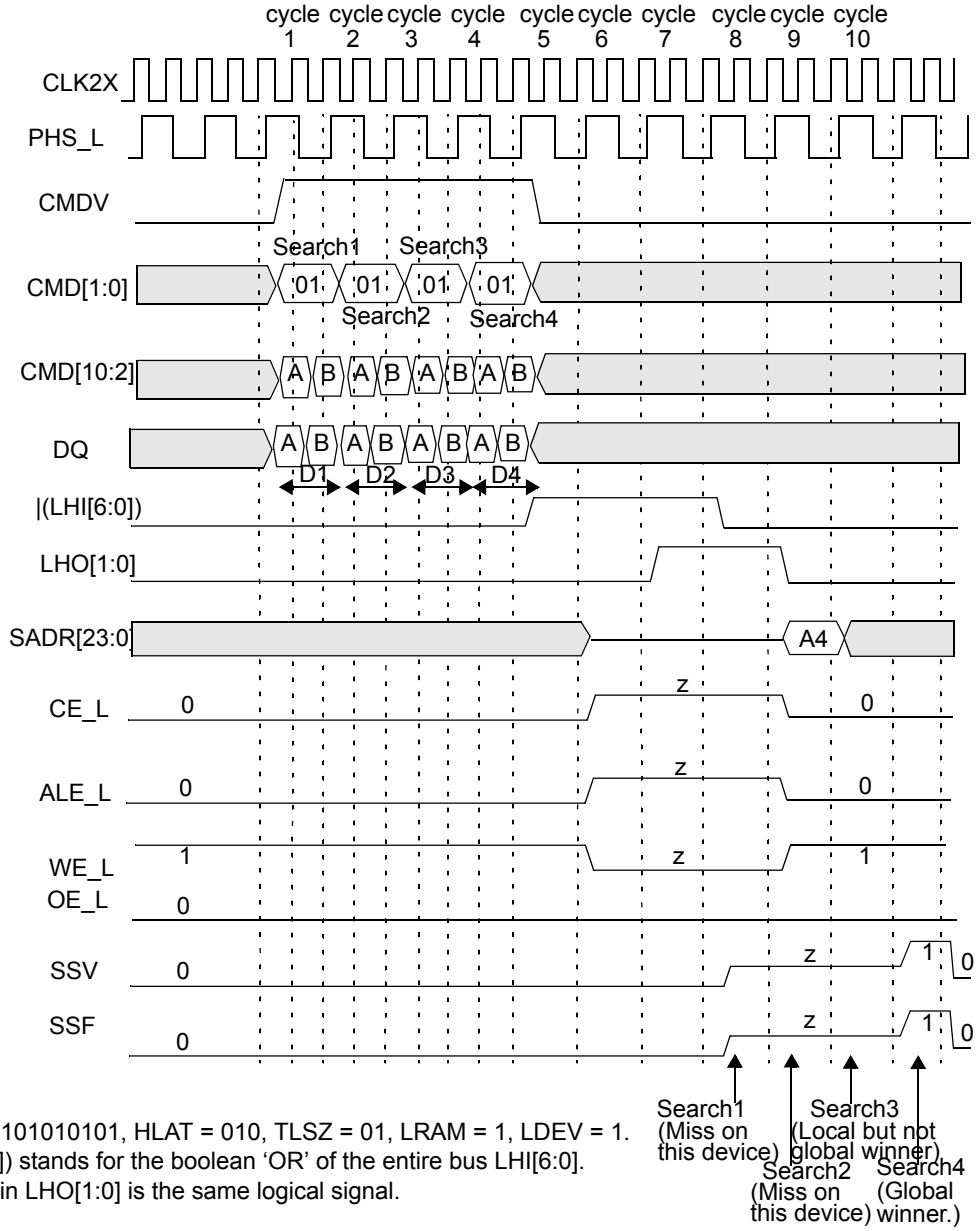
CFG = 0101010101010101, HLAT = 010, TLSZ = 01, LRAM = 0, LDEV = 0.
Note: |(LHI[6:0]) stands for the boolean 'OR' of the entire bus LHI[6:0].
Note: Each bit in LHO[1:0] is the same logical signal.

Figure 10-31. Timing Diagram for 144-bit Search Device Number 0



CFG = 0101010101010101, HLAT = 010, TLSZ = 01, LRAM = 0, LDEV = 0.
Note: |(LHI[6:0]) stands for the boolean 'OR' of the entire bus LHI[6:0].
Note: Each bit in LHO[1:0] is the same logical signal.

Figure 10-32. Timing Diagram for 144-bit Search Device Number 1



CFG = 0101010101010101, HLAT = 010, TLSZ = 01, LRAM = 1, LDEV = 1.
Note: |(LHI[6:0]) stands for the boolean 'OR' of the entire bus LHI[6:0].
Note: Each bit in LHO[1:0] is the same logical signal.

Figure 10-33. Timing Diagram for 144-bit Search Device Number 7 (Last Device)

The following is the sequence of operation for a single 144-bit Search command (also see "Commands and Command Parameters" on page 22).

- **Cycle A**: The host ASIC drives CMDV high and applies Search command code (10) on CMD[1:0] signals. {CMD[10],CMD[5:3]} signals must be driven with the index to the GMR pair for use in this Search operation. CMD[8:6] signals must be driven with the same bits that will be driven by this device on SADR[23:21] if it has a hit. DQ[71:0] must be driven with the 72-bit data ([143:72]) in order to be compared against all even locations. The CMD[2] signal must be driven to a logic 0.
- **Cycle B**: The host ASIC continues to drive CMDV high and to apply the command code for Search command (10) on CMD[1:0]. CMD[5:2] must be driven by the index of the comparand register pair for storing the 144-bit word presented on the DQ bus during cycles A and B. CMD[8:6] signals must be driven with the SSR index that will be used for storing the address of the matching entry and the hit flag (see page 8 for the description of SSR[0:7]). The DQ[71:0] is driven with 72-bit data ([71:0]) compared against all odd locations.

The logical 144-bit search operation is shown in *Figure 10-34*. The entire table (eight devices of 144-bit entries) is compared to a 144-bit word K (presented on the DQ bus in cycles A and B of the command) using the GMR and local mask bits. The GMR is the 144-bit word specified by the even and odd global mask pair selected by the GMR Index in the command's cycle A.

The 144-bit word K (presented on the DQ bus in cycles A and B of the command) is also stored in the even and odd comparand registers specified by the Comparand Register Index in the command's cycle B. In x144 configurations, the even and odd comparand registers can subsequently be used by the Learn command in only one of the devices (the first non-full device). The word K (presented on the DQ bus in cycles A and B of the command) is compared to each entry in the table starting at location 0. The first matching entry's location, address L, is the winning address that is driven as part of the SRAM address on the SADR[23:0] lines (see "SRAM Addressing" on page 105). The global winning device will drive the bus in a specific cycle. On global miss cycles the device with LRAM = 1 (the default driving device for the SRAM bus) and LDEV = 1 (the default driving device for SSF and SSV signals) will be the default driver for such missed cycles. ***Note***. During 144-bit searches of 144-bit-configured tables, the search hit will always be at an even address.

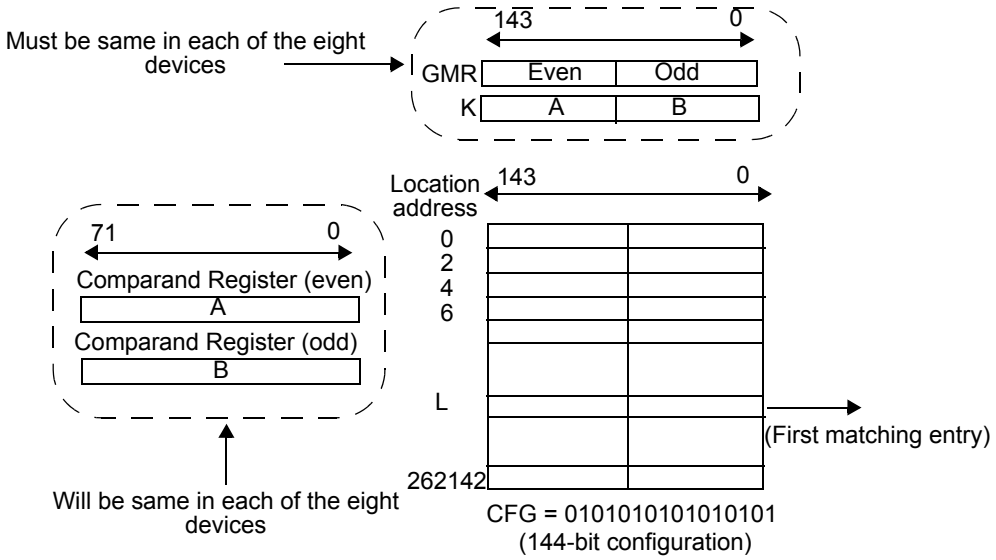


Figure 10-34. x144 Table with Eight Devices

The Search command is a pipelined operation and executes a Search at half the rate of the frequency of CLK2X for 144-bit searches in x144-configured tables. The latency of SADR, CE_L, ALE_L, WE_L, SSV, and SSF from the 144-bit Search command cycle (two CLK2X cycles) is shown in *Table 10-21*.

Table 10-21. Search Latency from Instruction to SRAM Access Cycle

Number of Devices	Max Table Size	Latency in CLK Cycles
1 (TLSZ = 00)	32K × 144 bits	4
1–8 (TLSZ = 01)	256K × 144 bits	5
1–31 (TLSZ = 10)	992K × 144 bits	6

For one to eight devices in the table and TLSZ = 01, the latency of a Search from command to SRAM access cycle is 5. In addition, SSV and SSF shift further to the right for different values of HLAT as specified in *Table 10-22*.

Table 10-22. Shift OF SSF and SSV from SADR

HLAT	Number of CLK Cycles
000	0
001	1
010	2
011	3
100	4
101	5
110	6
111	7

10.6.6 144-bit Search on Tables Configured as x144 Using up to 31 CYNSE70128 Devices

The hardware diagram of the search subsystem of 31 devices is shown in *Figure 10-35*. Each of the four blocks in the diagram represents a block of eight CYNSE70128 devices (except the last, which has seven devices). The diagram for a block of eight devices is shown in *Figure 10-36*. Following are the parameters programmed into the 31 devices.

- First thirty devices (devices 0–29): CFG = 0101010101010101, TLSZ = 10, HLAT = 001, LRAM = 0, and LDEV = 0.
- Thirty-first device (device 30): CFG = 0101010101010101, TLSZ = 10, HLAT = 001, LRAM = 1, and LDEV = 1.

Note. All 31 devices must be programmed with the same value of TLSZ and HLAT. Only the last device in the table must be programmed with LRAM = 1 and LDEV = 1 (device number 30 in this case). All other upstream devices must be programmed with LRAM = 0 and LDEV = 0 (devices 0 through 29 in this case).

The timing diagrams referred to in this paragraph reference the Hit/Miss assumptions defined in *Table 10-23*. For the purpose of illustrating timings, it is further assumed that the there is only one device with a matching entry in each of the blocks. *Figure 10-37* shows the timing diagram for a Search command in the 144-bit-configured table (31 devices) for each of the eight devices in block number 0. *Figure 10-38* shows the timing diagram for Search command in the 72-bit-configured table (31 devices) for all the devices in block number 1 above the winning device in that block. *Figure 10-39* shows the timing diagram for the globally winning device (the final winner within its own block and all blocks) in block number 1. *Figure 10-40* shows the timing diagram for all the devices below the globally winning device in block number 1. *Figure 10-41*, *Figure 10-42*, and *Figure 10-43* respectively show the timing diagrams of the devices above globally winning device, the globally winning device and devices below the globally winning device for block number 2. *Figure 10-44*, *Figure 10-45*, *Figure 10-46*, and *Figure 10-47* respectively show the timing diagrams of the devices above the globally winning device, the globally winning device, and devices below the globally winning device except the last device (device 30), and the last device (device 30) for block number 3.

The 144-bit Search operation is pipelined and executes as follows. Four cycles from the Search command, each of the devices knows the outcome internal to it for that operation. In the fifth cycle after the Search command, the devices in a block (being less than or equal to eight devices resolving the winner within them using the LHI[6:0] and LHO[1:0] signalling mechanism) arbitrate for a winner amongst them. In the sixth cycle after the Search command, the blocks (of devices) resolve the winning block through the BHI[2:0] and BHO[2:0] signalling mechanism. The winning device in the winning block is the global winning device for a Search operation.

Table 10-23. Hit/Miss Assumption

Search Number	1	2	3	4
Block 0	Miss	Miss	Miss	Miss
Block 1	Miss	Miss	Hit	Miss
Block 2	Miss	Hit	Hit	Miss
Block 3	Hit	Hit	Miss	Miss

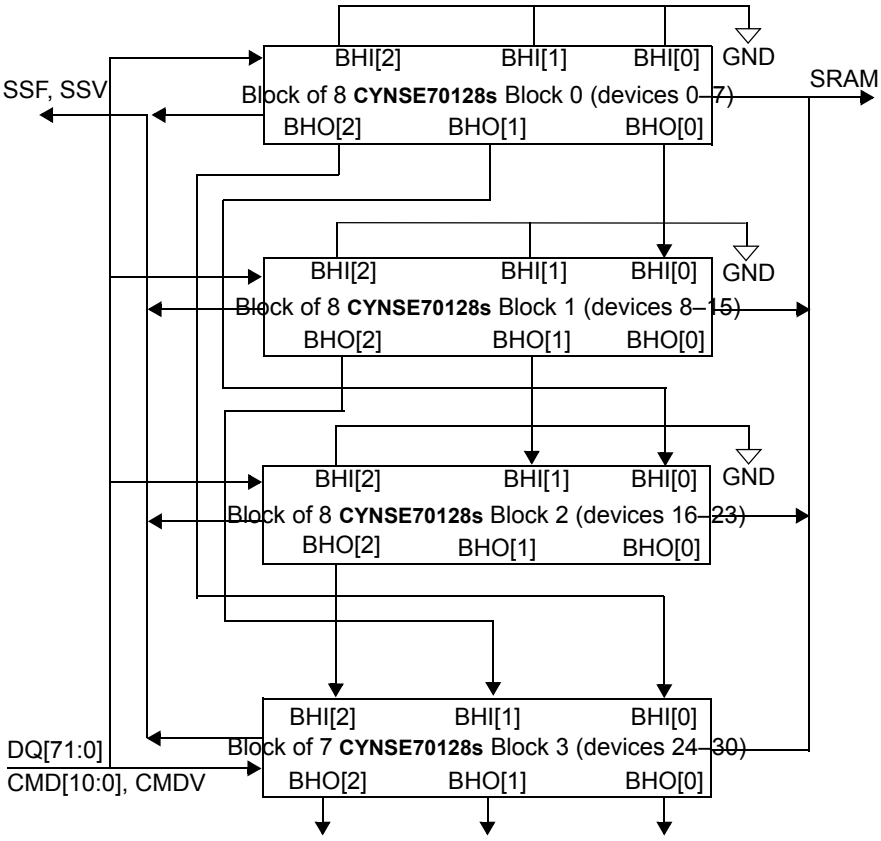


Figure 10-35. Hardware Diagram for a Table with 31 Devices

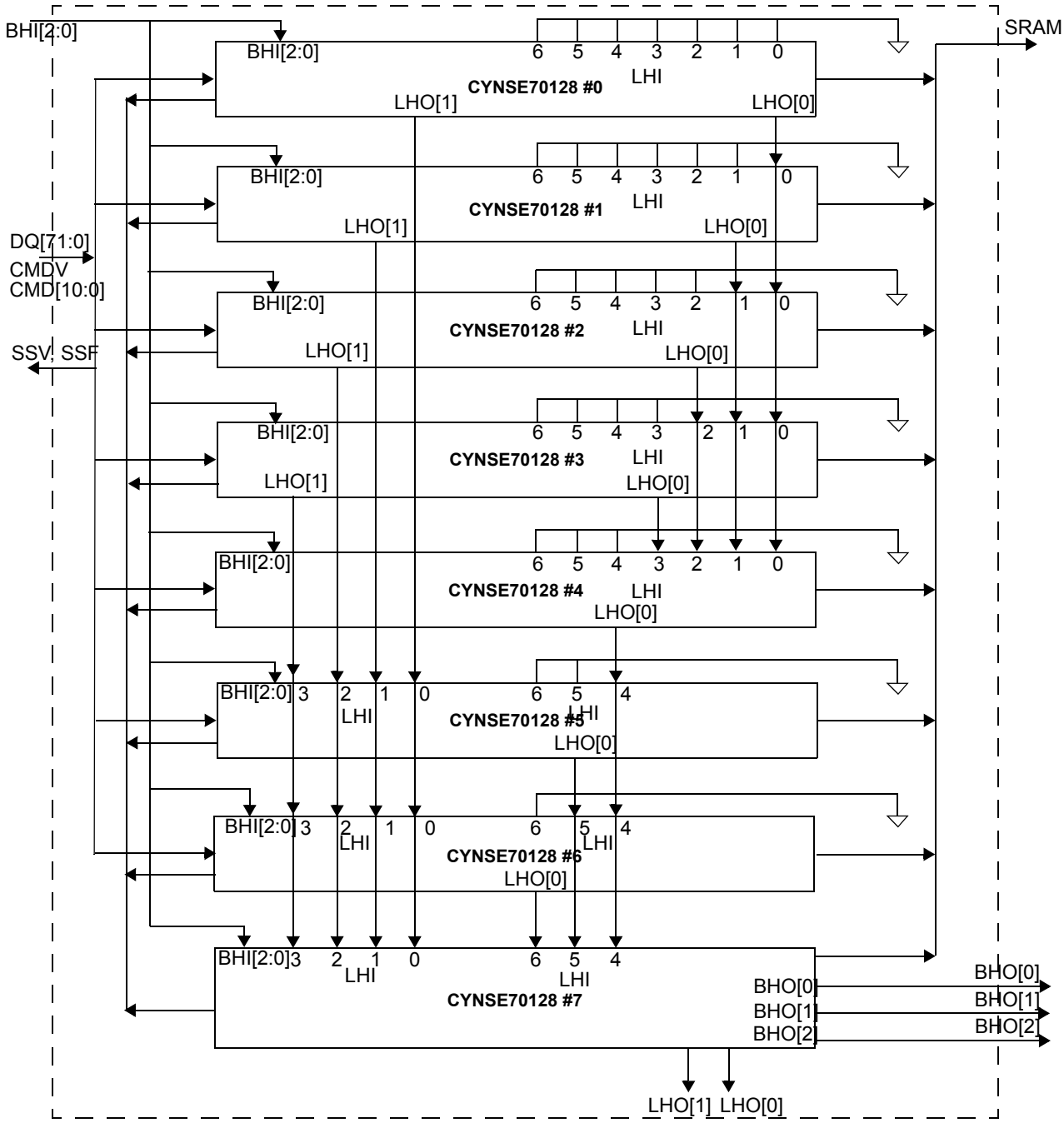
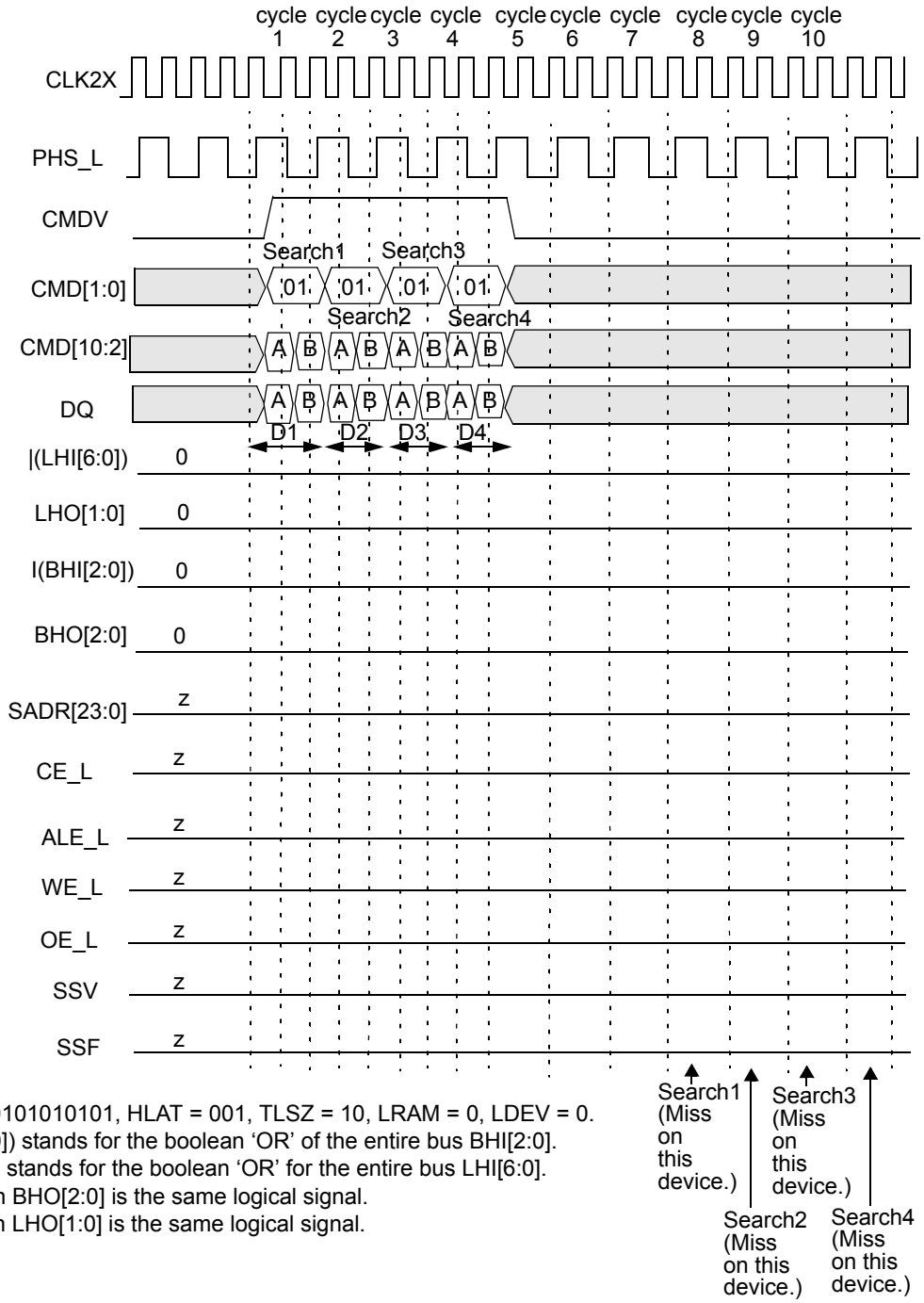
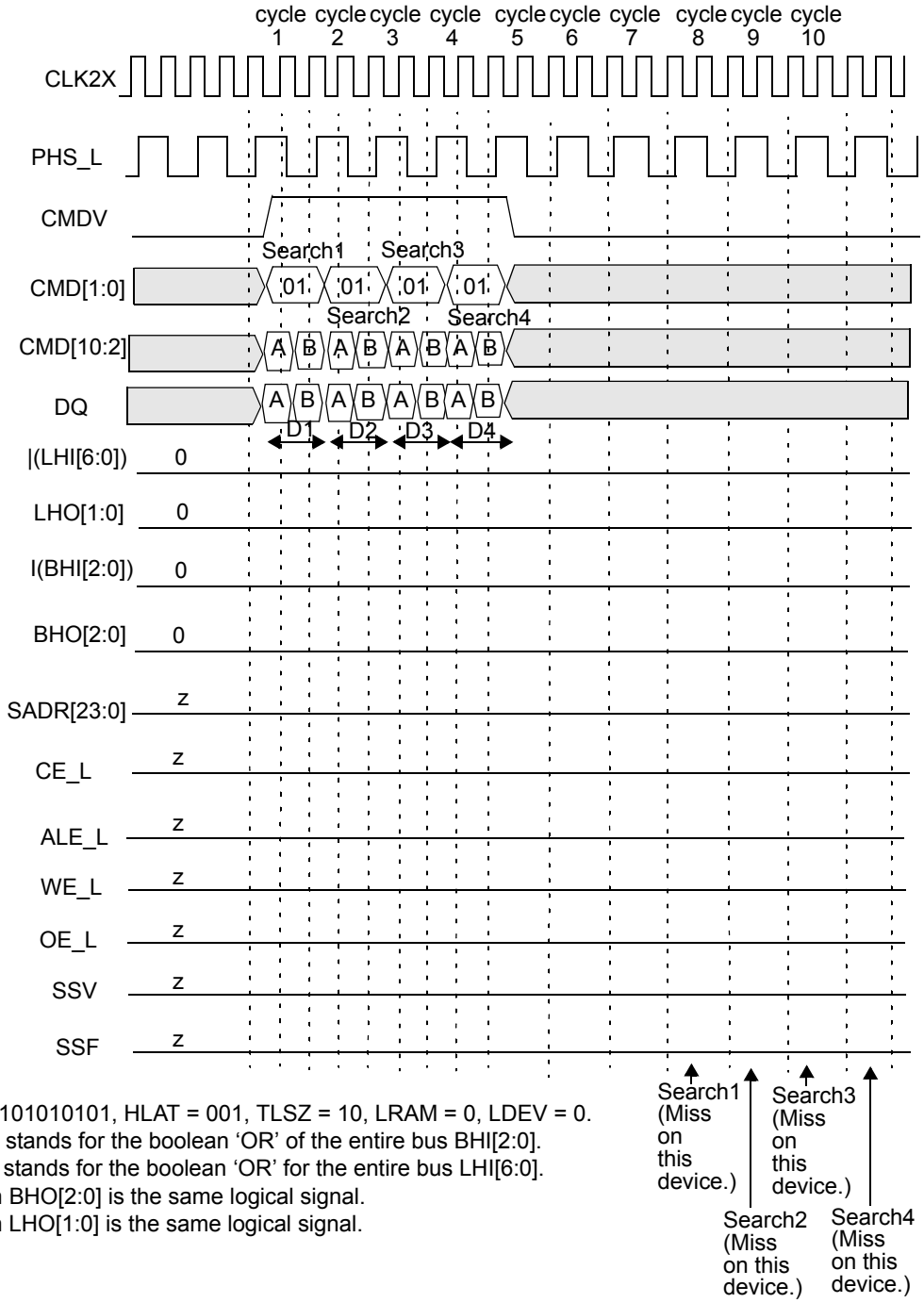


Figure 10-36. Hardware Diagram for a Block of up to Eight Devices



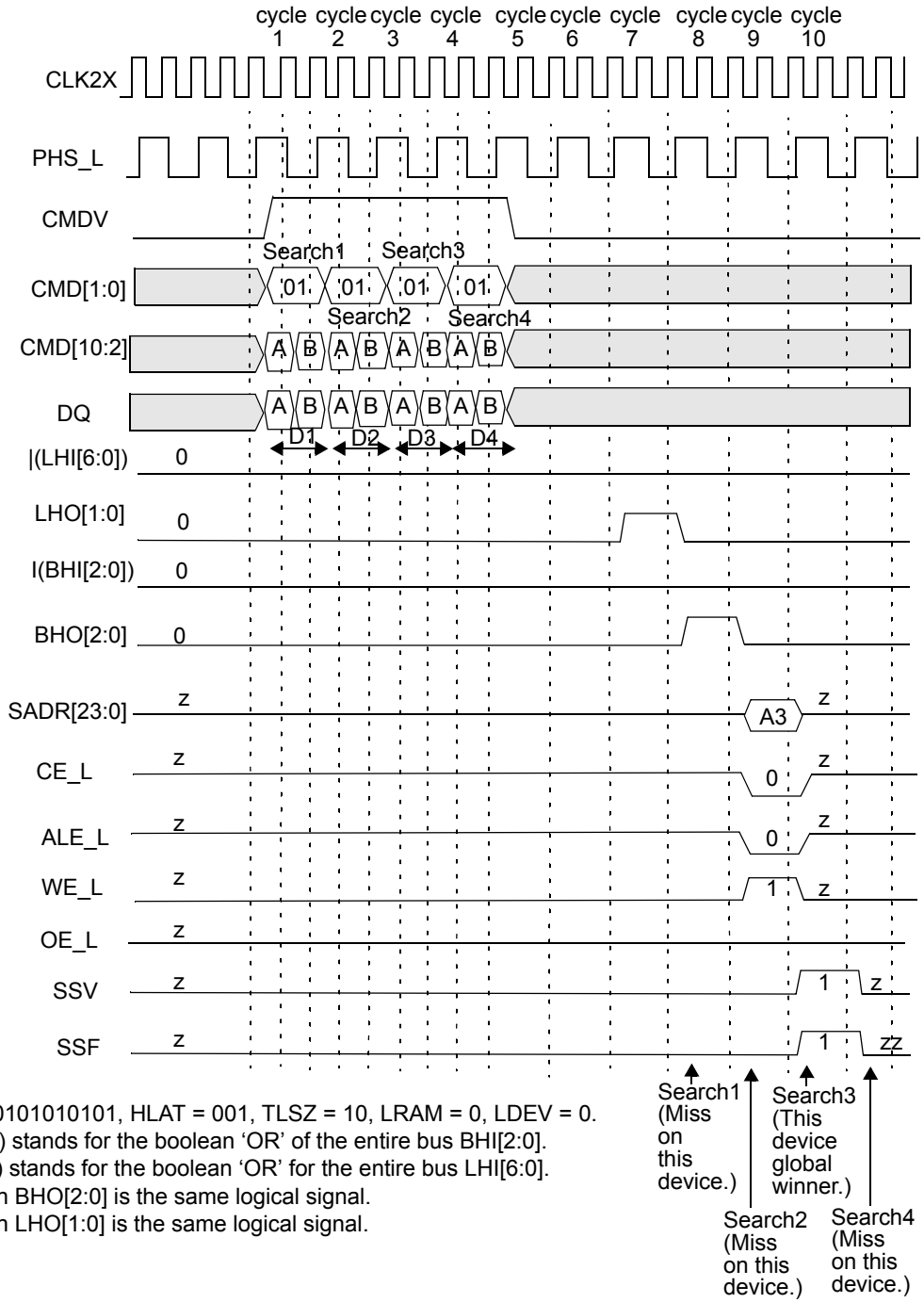
CFG = 0101010101010101, HLAT = 001, TLSZ = 10, LRAM = 0, LDEV = 0.
Note: |(BHI[2:0]) stands for the boolean 'OR' of the entire bus BHI[2:0].
Note: |(LHI[6:0]) stands for the boolean 'OR' for the entire bus LHI[6:0].
Note: Each bit in BHO[2:0] is the same logical signal.
Note: Each bit in LHO[1:0] is the same logical signal.

Figure 10-37. Timing Diagram for Each Device in Block Number 0 (Miss on Each Device)



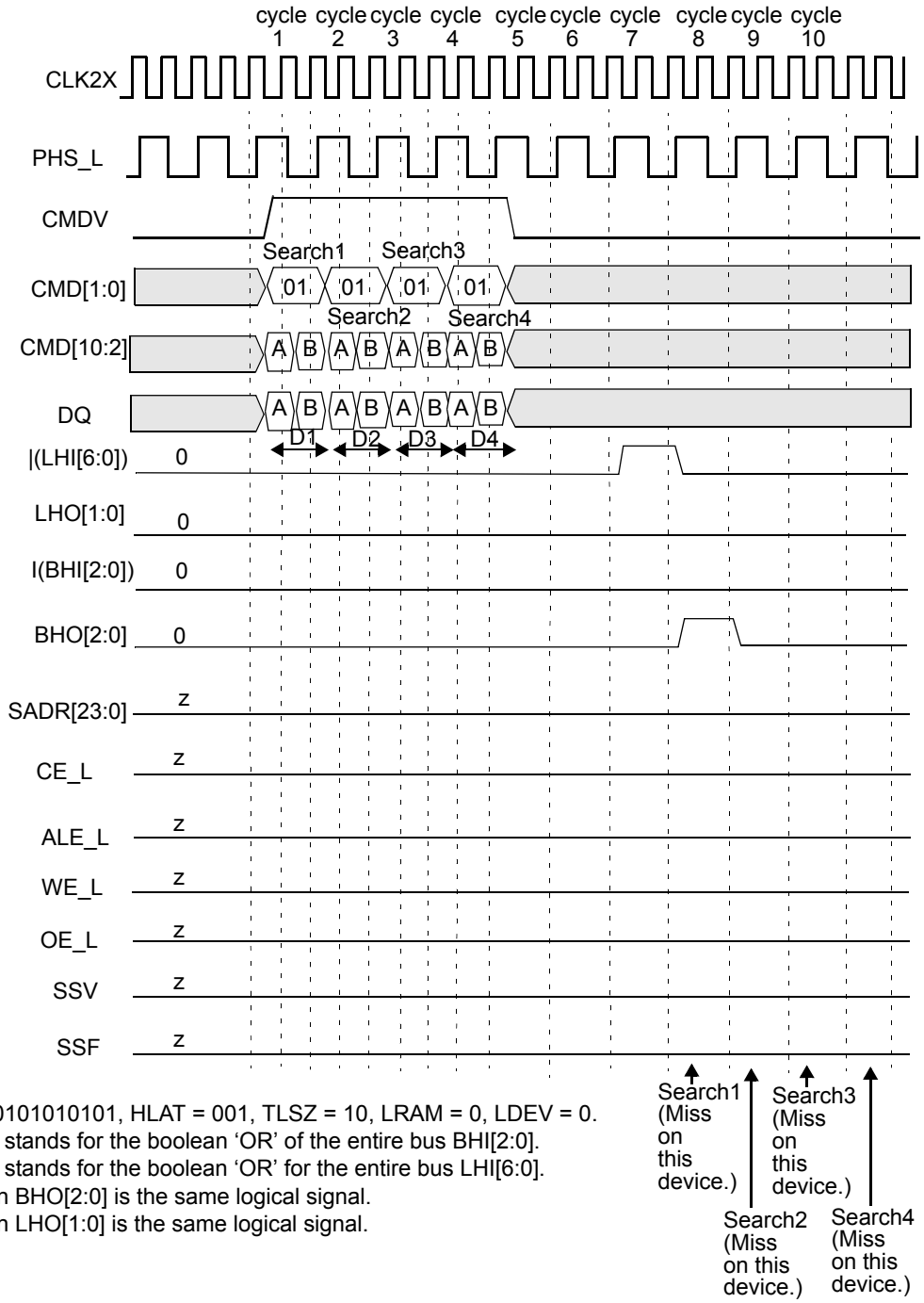
CFG = 0101010101010101, HLAT = 001, TLSZ = 10, LRAM = 0, LDEV = 0.
Note: |(BHI[2:0]) stands for the boolean 'OR' of the entire bus BHI[2:0].
Note: |(LHI[6:0]) stands for the boolean 'OR' for the entire bus LHI[6:0].
Note: Each bit in BHO[2:0] is the same logical signal.
Note: Each bit in LHO[1:0] is the same logical signal.

Figure 10-38. Timing Diagram for Each Device Above the Winning Device in Block Number 1



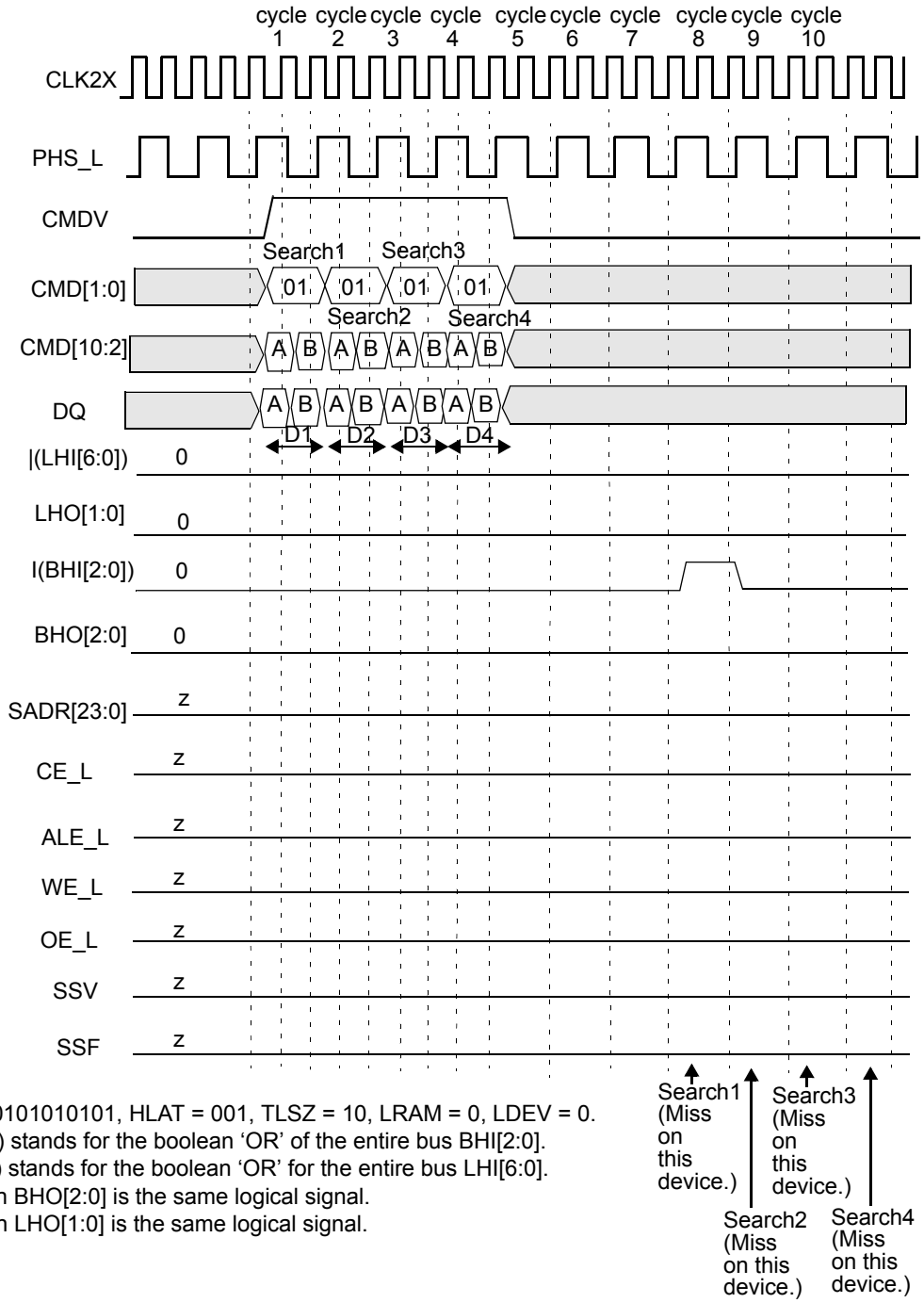
CFG = 0101010101010101, HLAT = 001, TLSZ = 10, LRAM = 0, LDEV = 0.
Note: |(BHI[2:0]) stands for the boolean 'OR' of the entire bus BHI[2:0].
Note: |(LHI[6:0]) stands for the boolean 'OR' for the entire bus LHI[6:0].
Note: Each bit in BHO[2:0] is the same logical signal.
Note: Each bit in LHO[1:0] is the same logical signal.

Figure 10-39. Timing Diagram for Globally Winning Device in Block Number 1



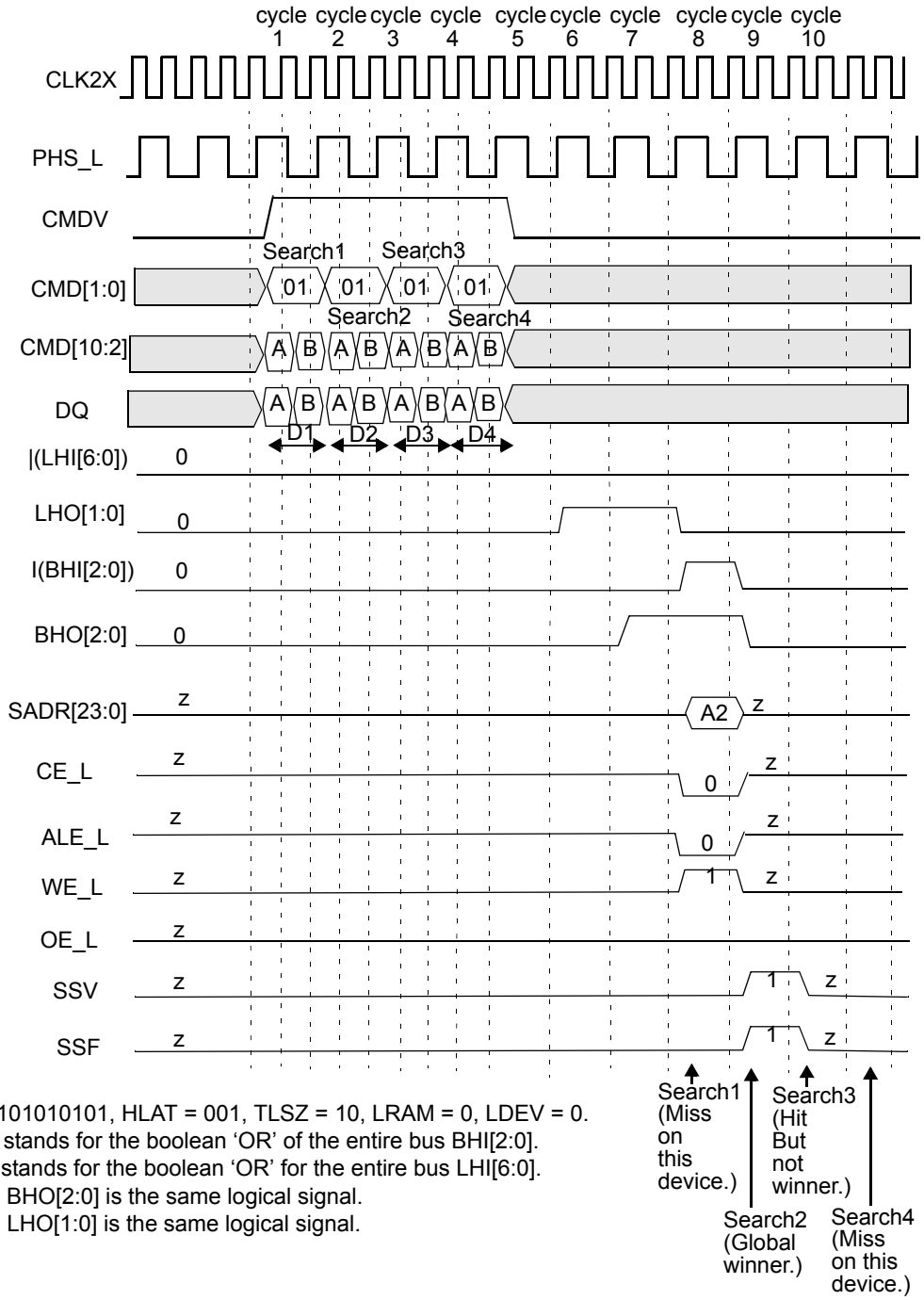
CFG = 0101010101010101, HLAT = 001, TLSZ = 10, LRAM = 0, LDEV = 0.
Note: |(BHI[2:0] stands for the boolean 'OR' of the entire bus BHI[2:0].
Note: |(LHI(6:0) stands for the boolean 'OR' for the entire bus LHI[6:0].
Note: Each bit in BHO[2:0] is the same logical signal.
Note: Each bit in LHO[1:0] is the same logical signal.

Figure 10-40. Timing Diagram for Devices Below the Winning Device in Block Number 1



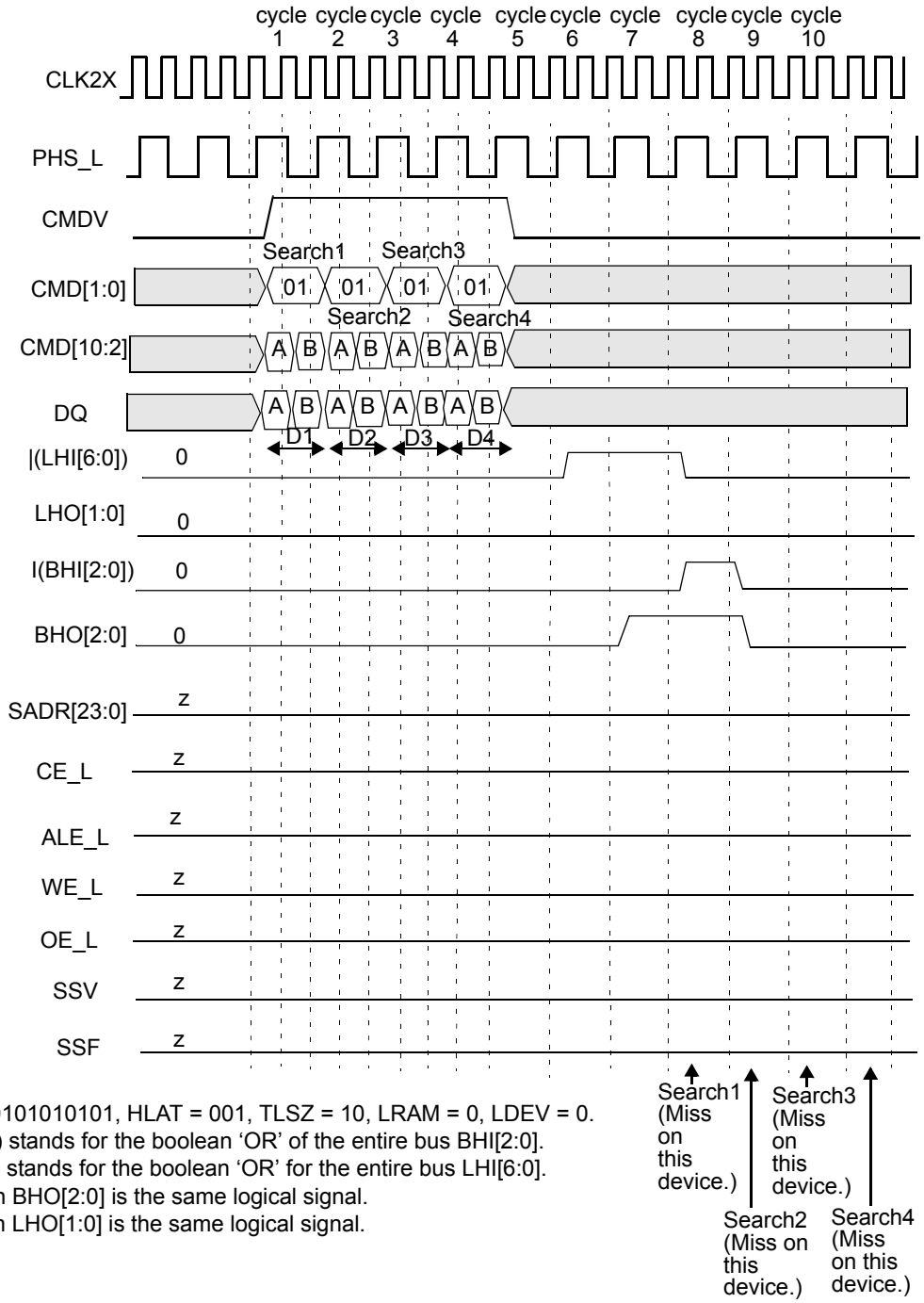
CFG = 0101010101010101, HLAT = 001, TLSZ = 10, LRAM = 0, LDEV = 0.
Note: |(BHI[2:0]) stands for the boolean 'OR' of the entire bus BHI[2:0].
Note: |(LHI[6:0]) stands for the boolean 'OR' for the entire bus LHI[6:0].
Note: Each bit in BHO[2:0] is the same logical signal.
Note: Each bit in LHO[1:0] is the same logical signal.

Figure 10-41. Timing Diagram for Devices Above the Winning Device in Block Number 2



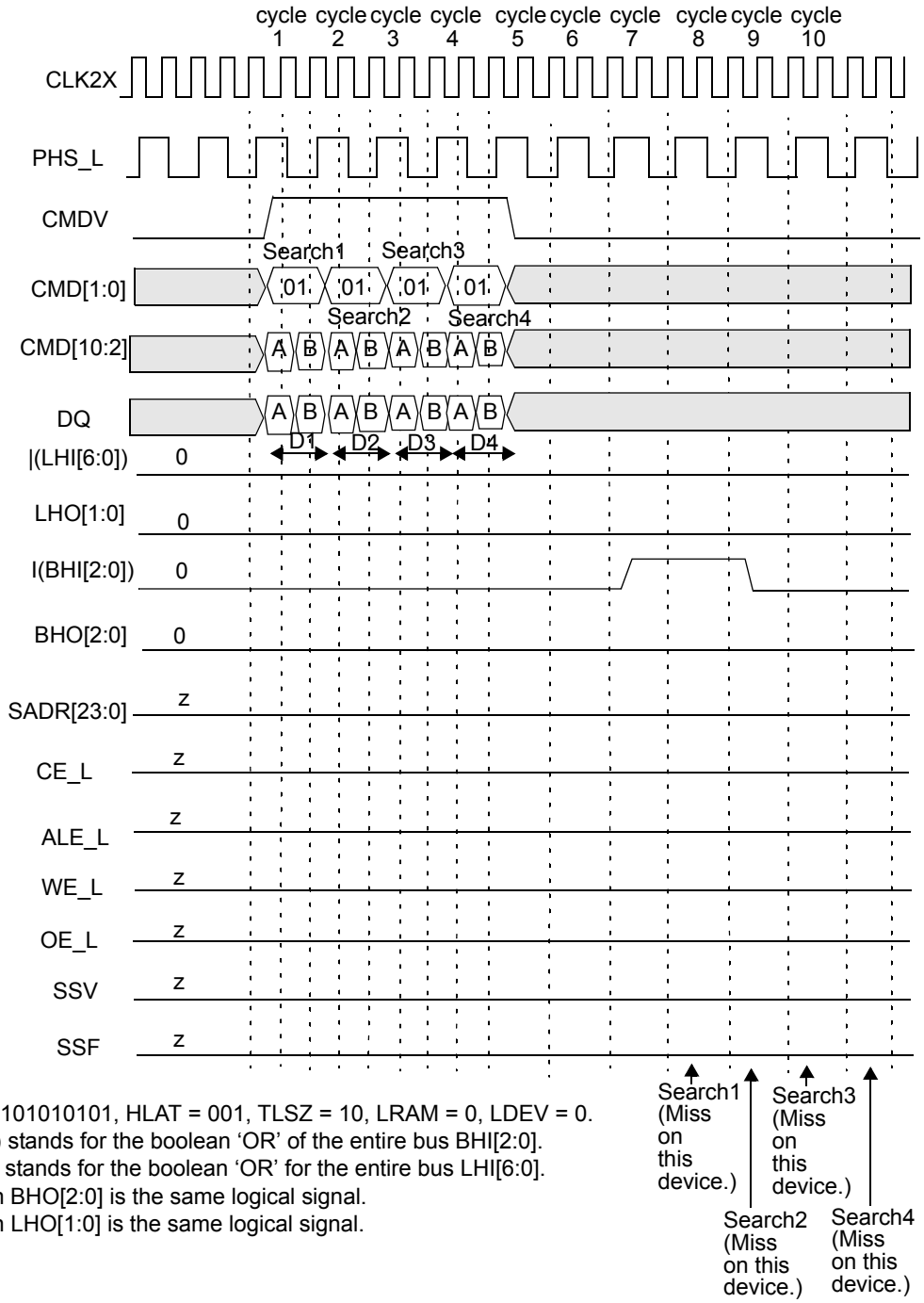
CFG = 0101010101010101, HLAT = 001, TLSZ = 10, LRAM = 0, LDEV = 0.
Note: |(BHI[2:0]) stands for the boolean 'OR' of the entire bus BHI[2:0].
Note: |(LHI[6:0]) stands for the boolean 'OR' for the entire bus LHI[6:0].
Note: Each bit in BHO[2:0] is the same logical signal.
Note: Each bit in LHO[1:0] is the same logical signal.

Figure 10-42. Timing Diagram for Globally Winning Device in Block Number 2



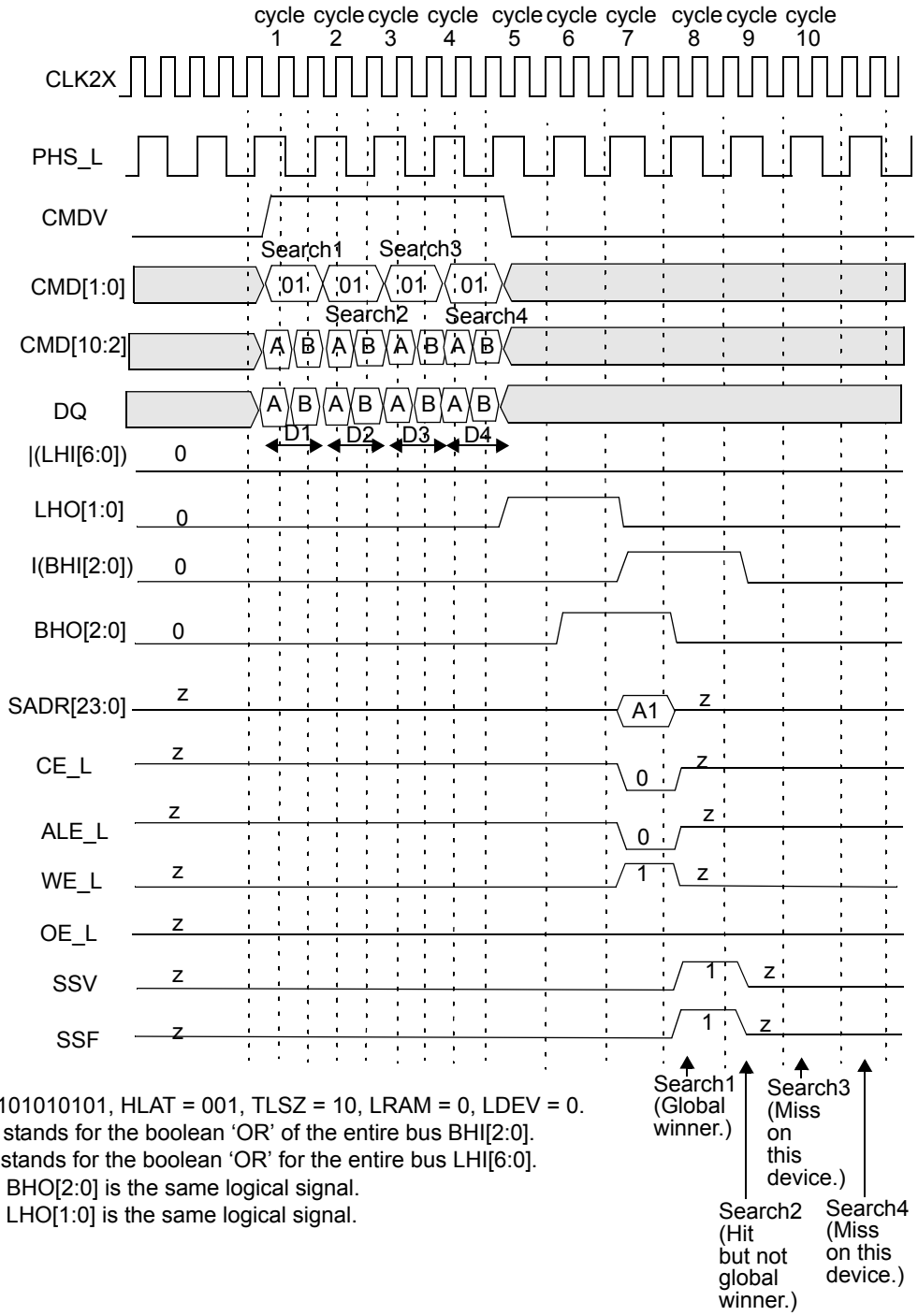
CFG = 0101010101010101, HLAT = 001, TLSZ = 10, LRAM = 0, LDEV = 0.
Note: |(BHI[2:0]) stands for the boolean 'OR' of the entire bus BHI[2:0].
Note: |(LHI[6:0]) stands for the boolean 'OR' for the entire bus LHI[6:0].
Note: Each bit in BHO[2:0] is the same logical signal.
Note: Each bit in LHO[1:0] is the same logical signal.

Figure 10-43. Timing Diagram for Devices Below the Winning Device in Block Number 2



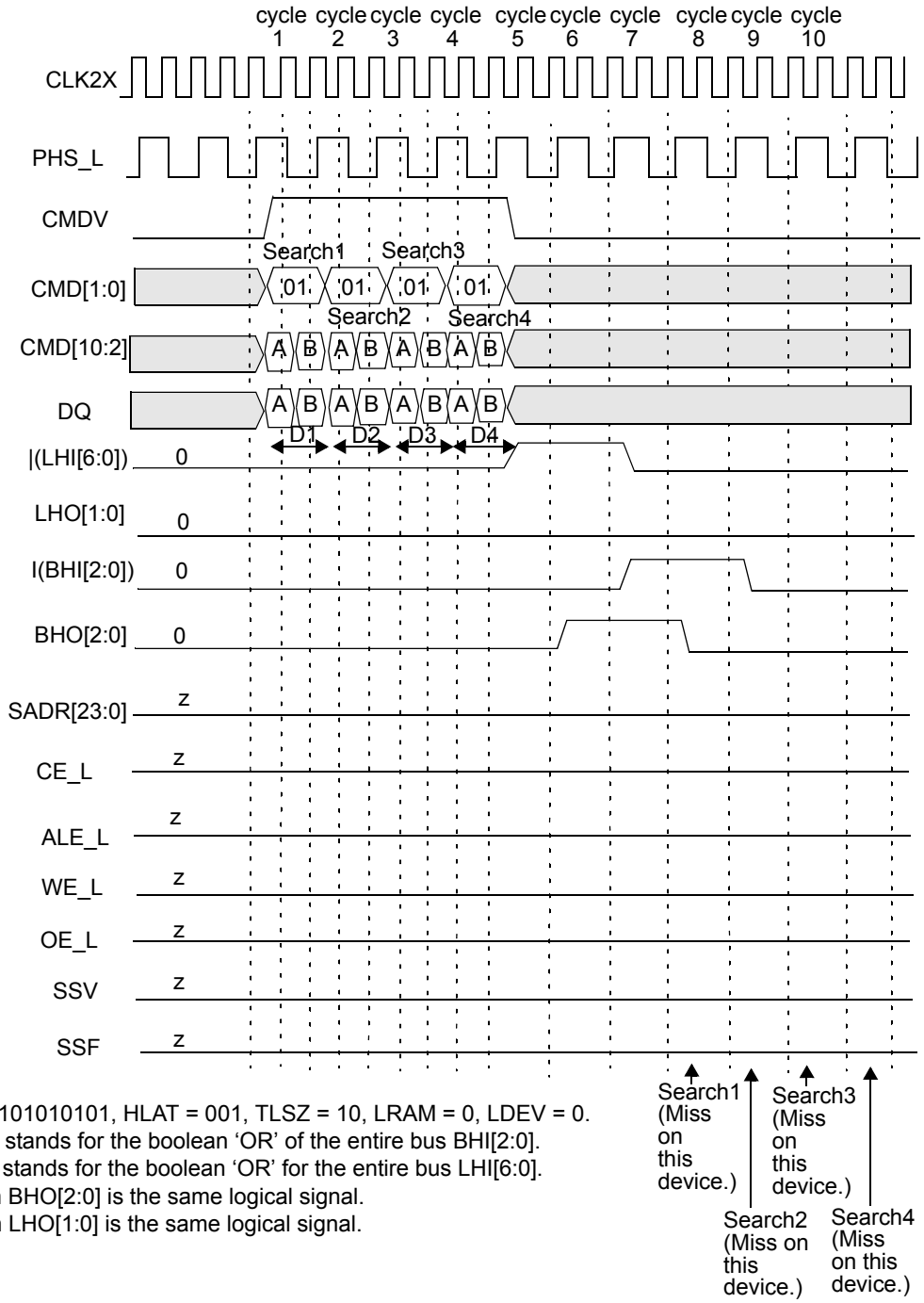
CFG = 0101010101010101, HLAT = 001, TLSZ = 10, LRAM = 0, LDEV = 0.
Note: |(BHI[2:0]) stands for the boolean 'OR' of the entire bus BHI[2:0].
Note: |(LHI[6:0]) stands for the boolean 'OR' for the entire bus LHI[6:0].
Note: Each bit in BHO[2:0] is the same logical signal.
Note: Each bit in LHO[1:0] is the same logical signal.

Figure 10-44. Timing Diagram for Devices Above the Winning Device in Block Number 3



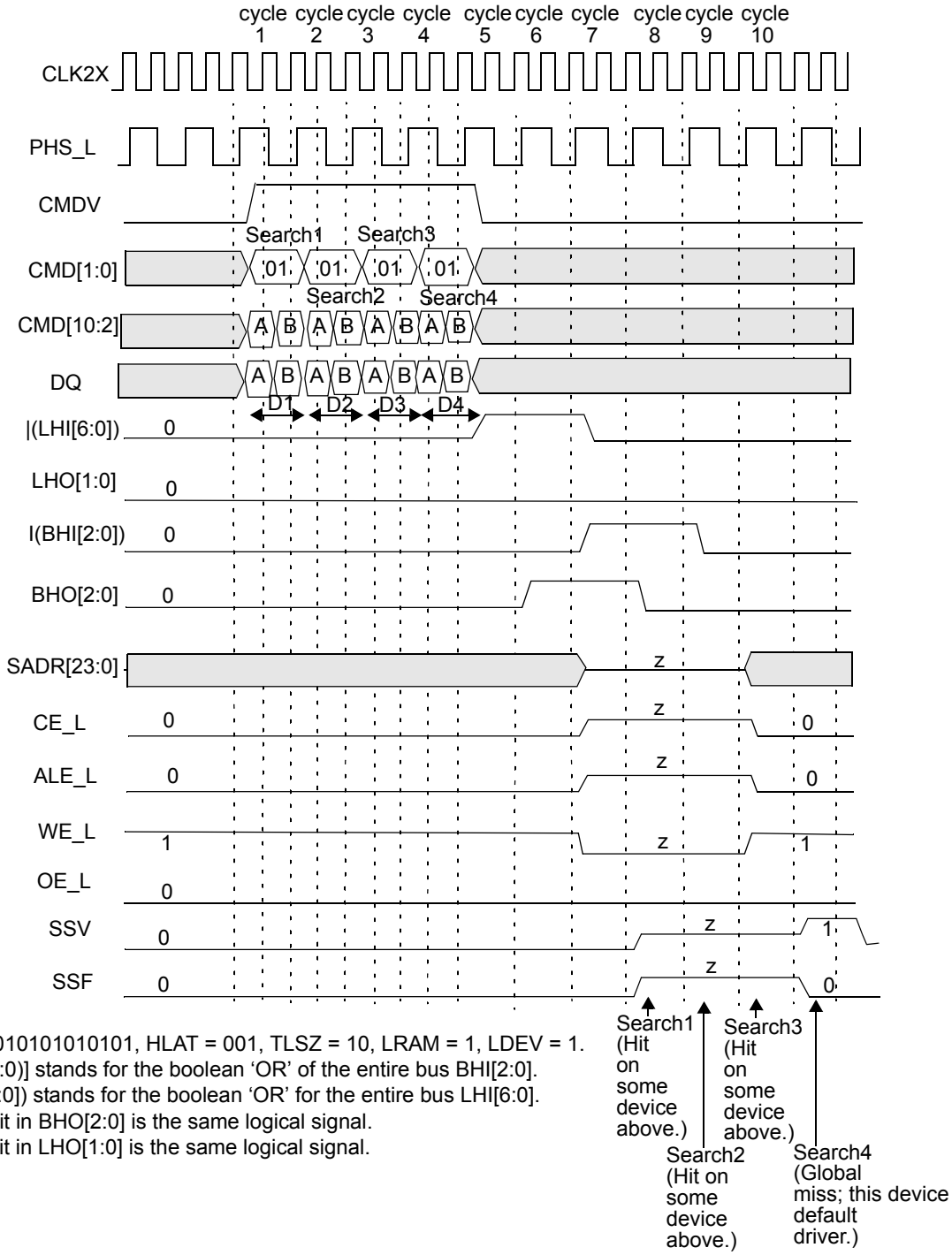
CFG = 0101010101010101, HLAT = 001, TLSZ = 10, LRAM = 0, LDEV = 0.
Note: |(BHI[2:0]) stands for the boolean 'OR' of the entire bus BHI[2:0].
Note: |(LHI[6:0]) stands for the boolean 'OR' for the entire bus LHI[6:0].
Note: Each bit in BHO[2:0] is the same logical signal.
Note: Each bit in LHO[1:0] is the same logical signal.

Figure 10-45. Timing Diagram for Globally Winning Device in Block Number 3



CFG = 0101010101010101, HLAT = 001, TLSZ = 10, LRAM = 0, LDEV = 0.
Note: |(BHI[2:0]) stands for the boolean 'OR' of the entire bus BHI[2:0].
Note: |(LHI[6:0]) stands for the boolean 'OR' for the entire bus LHI[6:0].
Note: Each bit in BHO[2:0] is the same logical signal.
Note: Each bit in LHO[1:0] is the same logical signal.

Figure 10-46. Timing Diagram for Devices Below the Winning Device in Block Number 3 Except Device 30 (the Last Device)



CFG = 0101010101010101, HLAT = 001, TLSZ = 10, LRAM = 1, LDEV = 1.
Note: |(BHI[2:0)] stands for the boolean 'OR' of the entire bus BHI[2:0].
Note: |(LHI[6:0]) stands for the boolean 'OR' for the entire bus LHI[6:0].
Note: Each bit in BHO[2:0] is the same logical signal.
Note: Each bit in LHO[1:0] is the same logical signal.

Figure 10-47. Timing Diagram for Device Number 6 in Block Number 3 (Device 30 in Depth-Cascaded Table)

The following is the sequence of operation for a single 144-bit Search command (also refer to "Command and Command Parameters," on page 22).

- **Cycle A**: The host ASIC drives the CMDV high and applies Search command code (10) on CMD[1:0] signals. {CMD[10], CMD[5:3]} signals must be driven with the index to the GMR pair for use in this Search operation. CMD[8:6] signals must be driven with the bits that will be driven on SADR[23:21] by this device if it has a hit. DQ[71:0] must be driven with the 72-bit data ([143:72]) in order to be compared against all even locations. The CMD[2] signal must be driven to logic 0.

- **Cycle B**: The host ASIC continues to drive the CMDV high and to apply Search command code (10) on CMD[1:0]. CMD[5:2] must be driven by the index of the comparand register pair for storing the 144-bit word presented on the DQ bus during cycles A and B. CMD[8:6] signals must be driven with the index of the SSR that will be used for storing the address of the matching entry and the hit flag (see page 16 for the description of SSR[0:7]). The DQ[71:0] is driven with 72-bit data ([71:0])to be compared against all odd locations.

The logical 144-bit search operation is as shown in *Figure 10-48*. The entire table of 31 devices (consisting of 144-bit entries) is compared against a 144-bit word K that is presented on the DQ bus in cycles A and B of the command using the GMR and local mask bits. The GMR is the 144-bit word specified by the even and odd global mask pair selected by the GMR Index in the command's cycle A.

The 144-bit word K that is presented on the DQ bus in cycles A and B of the command is also stored in the even and odd comparand registers specified by the Comparand Register Index in the command's cycle B. In x144 configurations, the even and odd comparand registers can subsequently be used by the Learn command in only the first non-full device. ***Note***. The Learn command is supported for only one of the blocks consisting of up to eight devices in a depth-cascaded table of more than one block. The word K that is presented on the DQ bus in cycles A and B of the command is compared with each entry in the table starting at location 0. The first matching entry's location address L is the winning address that is driven as part of the SRAM address on the SADR[23:0] lines (see "SRAM Addressing" on page 105). The global winning device will drive the bus in a specific cycle. On global miss cycles the device with LRAM = 1 (the default driving device for the SRAM bus) and LDEV = 1 (the default driving device for SSF and SSV signals) will be the default driver for such missed cycles. ***Note***. During 144-bit searches of 144-bit-configured tables, the search hit will always be at an even address.

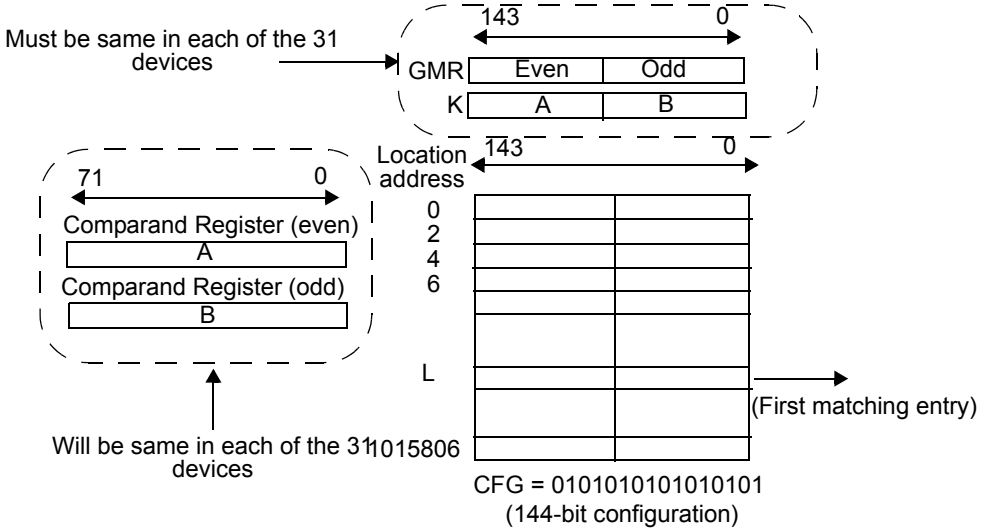


Figure 10-48. x144 Table with 31 Devices

The Search command is a pipelined operation. It executes a search at half the rate of the frequency of CLK2X for 144-bit searches in x144-configured tables. The latency of SADR, CE_L, ALE_L, WE_L, SSV, and SSF from the 144-bit Search command cycle (two CLK2X cycles) is shown in *Table 10-24*.

Table 10-24. The Latency of Search from Instruction to SRAM Access Cycle

Number of Devices	Max Table Size	Latency in CLK Cycles
1 (TLSZ = 00)	32K × 144 bits	4
1–8 (TLSZ = 01)	256K × 144 bits	5
1–31 (TLSZ = 10)	992K × 144 bits	6

The latency of a search from command to the SRAM access cycle is 6 for 1–31 devices in the table and where TLSZ = 10. In addition, SSV and SSF shift further to the right for different values of HLAT, as specified in *Table 10-25*.

Table 10-25. Shift OF SSF and SSV from SADR

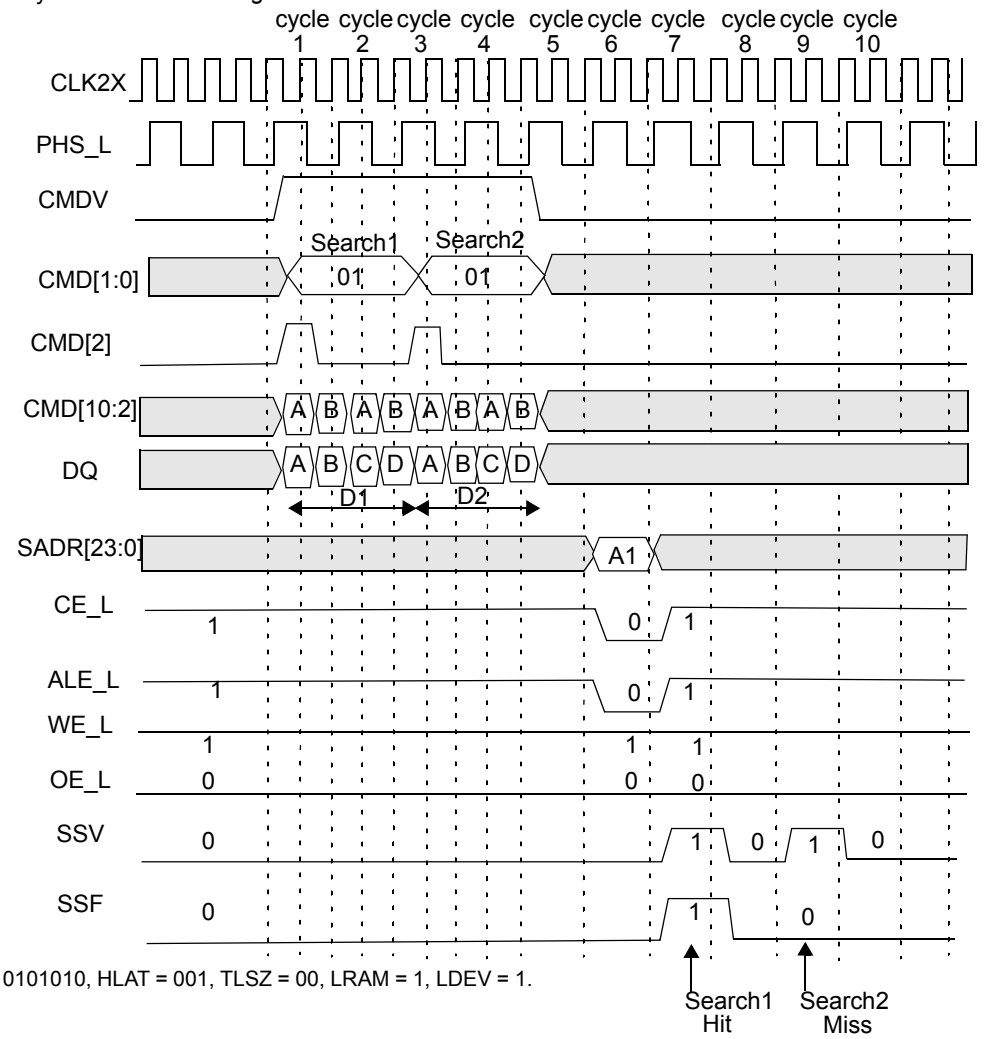
HLAT	Number of CLK Cycles
000	0
001	1
010	2
011	3

Table 10-25. Shift OF SSF and SSV from SADR (continued)

HLAT	Number of CLK Cycles
100	4
101	5
110	6
111	7

10.6.7 288-bit Search on Tables Configured as x288 Using a Single CYNSE70128 Device

Figure 10-49 shows the timing diagram for a Search command in the 288-bit-configured table (CFG = 1010101010101010) consisting of a single device for one set of parameters: TLSZ = 00, HLAT = 001, LRAM = 1, and LDEV = 1. The hardware diagram for this search subsystem is shown in *Figure 10-50*.



CFG = 1010101010101010, HLAT = 001, TLSZ = 00, LRAM = 1, LDEV = 1.

Figure 10-49. Timing Diagram for 288-bit Search (One Device)

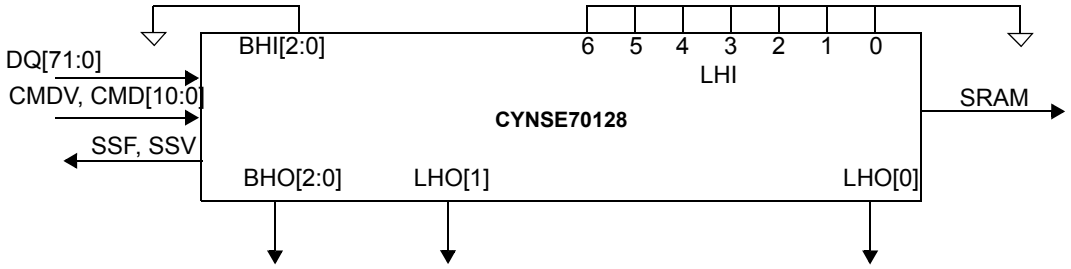


Figure 10-50. Hardware Diagram for a Table with One Device

The following is the sequence of operation for a single 144-bit Search command (also refer to “Commands and Command Parameters” on page 22).

- **Cycle A**: The host ASIC drives the CMDV high and applies Search command code (10) on CMD[1:0] signals. {CMD[10], CMD[5:3]} signals must be driven with the index to the GMR pair used for bits [287:144] of the data being searched. DQ[71:0] must be driven with the 72-bit data ([287:216]) to be compared to all locations 0 in the four 72-bits-word page. The CMD[2] signal must be driven to logic 1. ***Note***. CMD[2] = 1 signals that the Search is a x288-bit search. CMD[8:3] in this cycle is ignored.
- **Cycle B**: The host ASIC continues to drive the CMDV high and continues to apply the command code of Search command (10) on CMD[1:0]. The DQ[71:0] is driven with the 72-bit data ([215:144]) to be compared to all locations 1 in the four 72-bits-word page.
- **Cycle C**: The host ASIC drives the CMDV high and applies Search command code (10) on CMD[1:0] signals. {CMD[10],CMD[5:3]} signals must be driven with the index to the GMR pair used for bits [143:0] of the data being searched. CMD[8:6] signals must be driven with the bits that will be driven on SADR[23:21] by this device if it has a hit. DQ[71:0] must be driven with the 72-bit data ([143:72]) to be compared to all locations 2 in the four 72-bits-word page. The CMD[2] signal must be driven to logic 0.
- **Cycle D**: The host ASIC continues to drive the CMDV high and applies Search command code (10) on CMD[1:0]. CMD[8:6] signals must be driven with the index of the SSR that will be used for storing the address of the matching entry and the hit flag (see page 8 for the description of SSR[0:7]). The DQ[71:0] is driven with the 72-bit data ([71:0]) to be compared to all locations 3 in the four 72-bits-word page. CMD[5:2] is ignored because the Learn instruction is not supported for x288 tables.

Note. For 288-bit searches, the host ASIC must supply four distinct 72-bit data words on DQ[71:0] during cycles A, B, C, and D. The GMR index in cycle A selects a pair of GMRs that apply to DQ data in cycles A and B. The GMR index in cycle C selects a pair of GMRs that apply to DQ data in cycles C and D.

The logical 288-bit Search operation is shown in *Figure 10-51*. The entire table of 288-bit entries is compared to a 288-bit word K that is presented on the DQ bus in cycles A, B, C, and D of the command using the GMR and local mask bits. The GMR is the 288-bit word specified by the two pairs of GMRs selected by the GMR Indexes in the command’s cycles A and C. The 288-bit word K that is presented on the DQ bus in cycles A, B, C and D of the command is compared with each entry in the table starting at location 0. The first matching entry’s location address L is the winning address that is driven as part of the SRAM address on SADR[23:0] lines (see “SRAM Addressing” on page 105). ***Note***. The matching address is always going to be location 0 in a four-entry page for a 288-bit Search (two LSBs of the matching index will be 00).

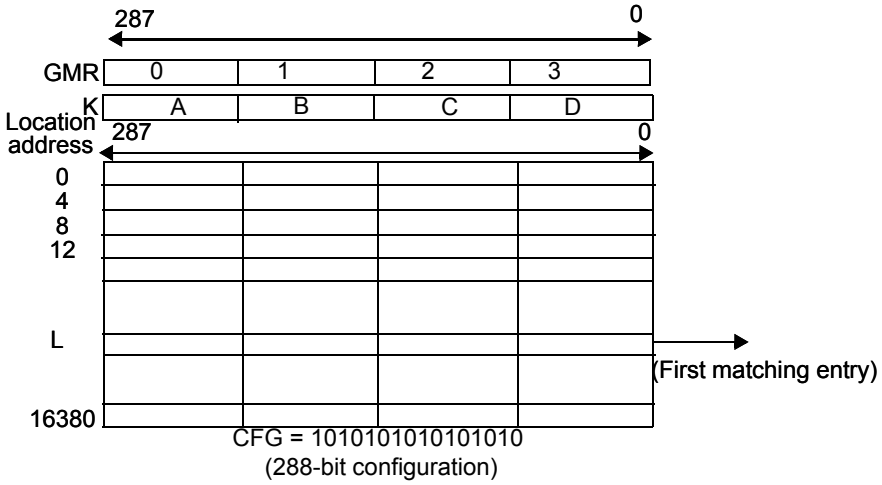


Figure 10-51. x288 Table with One Device

The Search command is a pipelined operation and executes at one-fourth the rate of the frequency of CLK2X for 288-bit searches in x288-configured tables. The latency of SADR, CE_L, ALE_L, WE_L, SSV, and SSF from the 288-bit Search command (measured in CLK cycles) from the CLK2X cycle that contains the C and D cycles is shown in *Table 10-26*.

Table 10-26. The Latency of Search from Cycles C and D to SRAM Access Cycle

Number of Devices	Max Table Size	Latency in CLK Cycles
1 (TLSZ = 00)	16K × 288 bits	4
1–8 (TLSZ = 01)	128K × 288 bits	5
1–31 (TLSZ = 10)	496K × 288 bits	6

The latency of a Search from command to SRAM access cycle is 4 for only a single device in the table and TLSZ = 00. In addition, SSV and SSF shift further to the right for different values of HLAT, as specified in *Table 10-27*.

Table 10-27. Shift OF SSF and SSV from SADR

HLAT	Number of CLK Cycles
000	0
001	1
010	2
011	3
100	4
101	5
110	6
111	7

10.6.8 288-bit Search on Tables Configured as x288 Using up to Eight CYNSE70128 Devices

The hardware diagram of the search subsystem of eight devices is shown in *Figure 10-52*. The following are the parameters programmed in the eight devices.

- First seven devices (devices 0–6): CFG = 1010101010101010, TLSZ = 01, HLAT = 000, LRAM = 0, and LDEV = 0.
- Eighth device (device 7): CFG = 1010101010101010, TLSZ = 01, HLAT = 000, LRAM = 1, and LDEV = 1.

Note. All eight devices must be programmed with the same value of TLSZ and HLAT. Only the last device in the table must be programmed with LRAM = 1 and LDEV = 1 (device number 7 in this case). All other upstream devices must be programmed with LRAM = 0 and LDEV = 0 (devices 0 through 6 in this case).

Figure 10-53 shows the timing diagram for a Search command in the 288-bit-configured table of eight devices for device number 0. *Figure 10-54* shows the timing diagram for a Search command in the 288-bit-configured table of eight devices for device number 1. *Figure 10-55* shows the timing diagram for a Search command in the 288-bit-configured table of eight devices for device number 7 (the last device in this specific table). For these timing diagrams three 288-bit searches are performed sequentially. The following Hit/Miss assumptions were made as shown in *Table 10-28*.

Table 10-28. Hit/Miss Assumption

Search Number	1	2	3
Device 0	Hit	Miss	Miss
Device 1	Miss	Hit	Miss
Device 2–6	Miss	Miss	Miss
Device 7	Miss	Miss	Miss

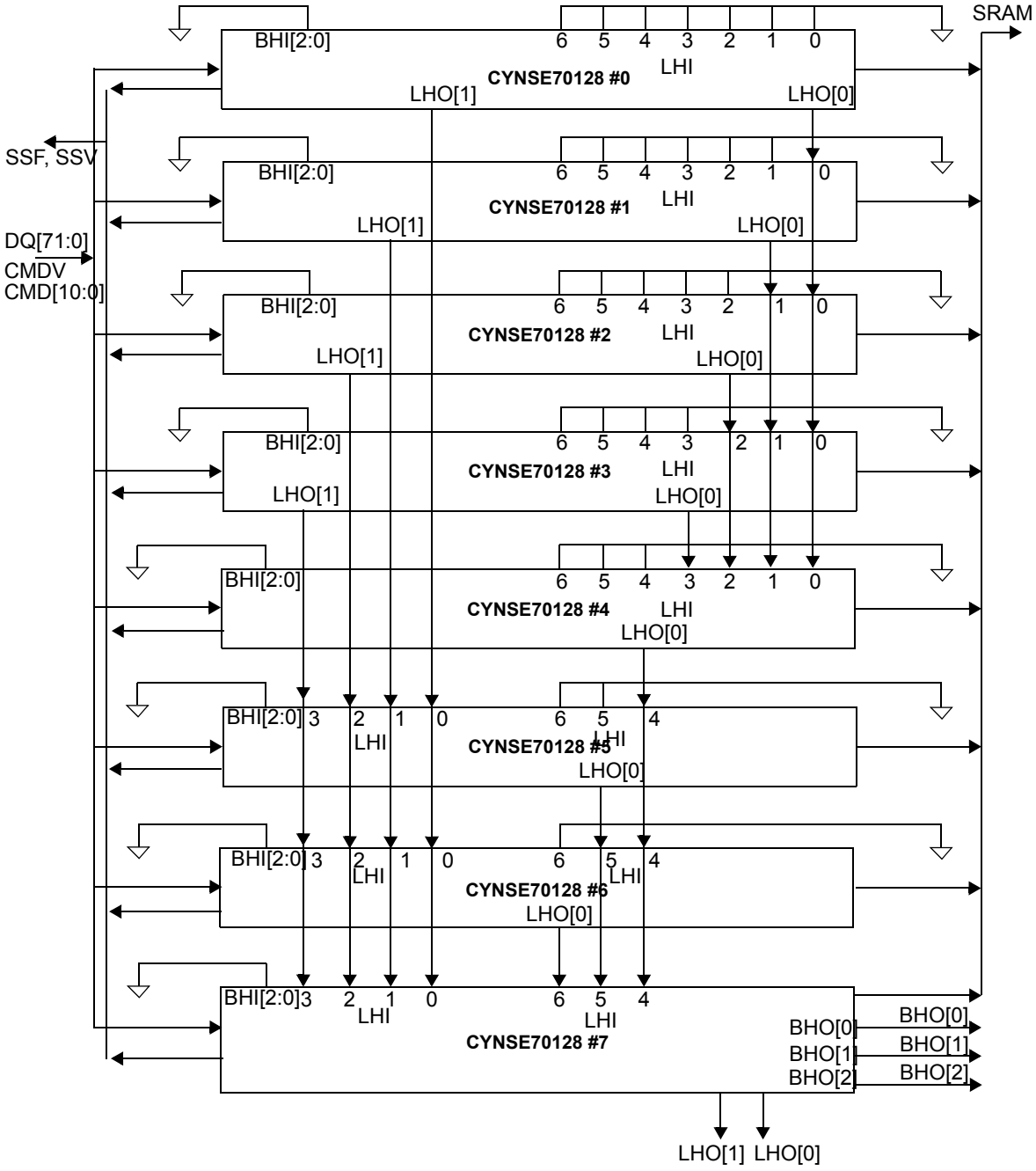
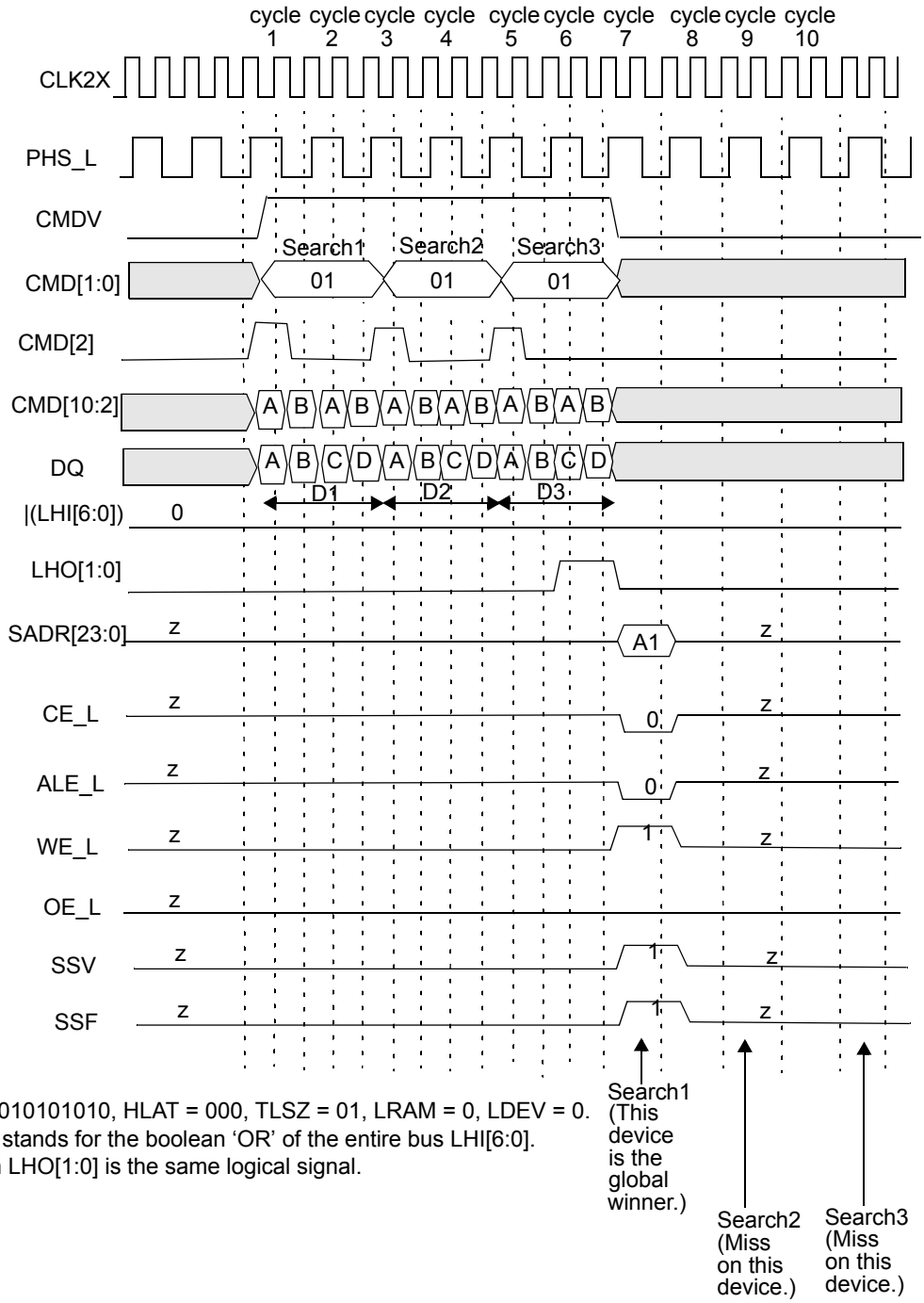
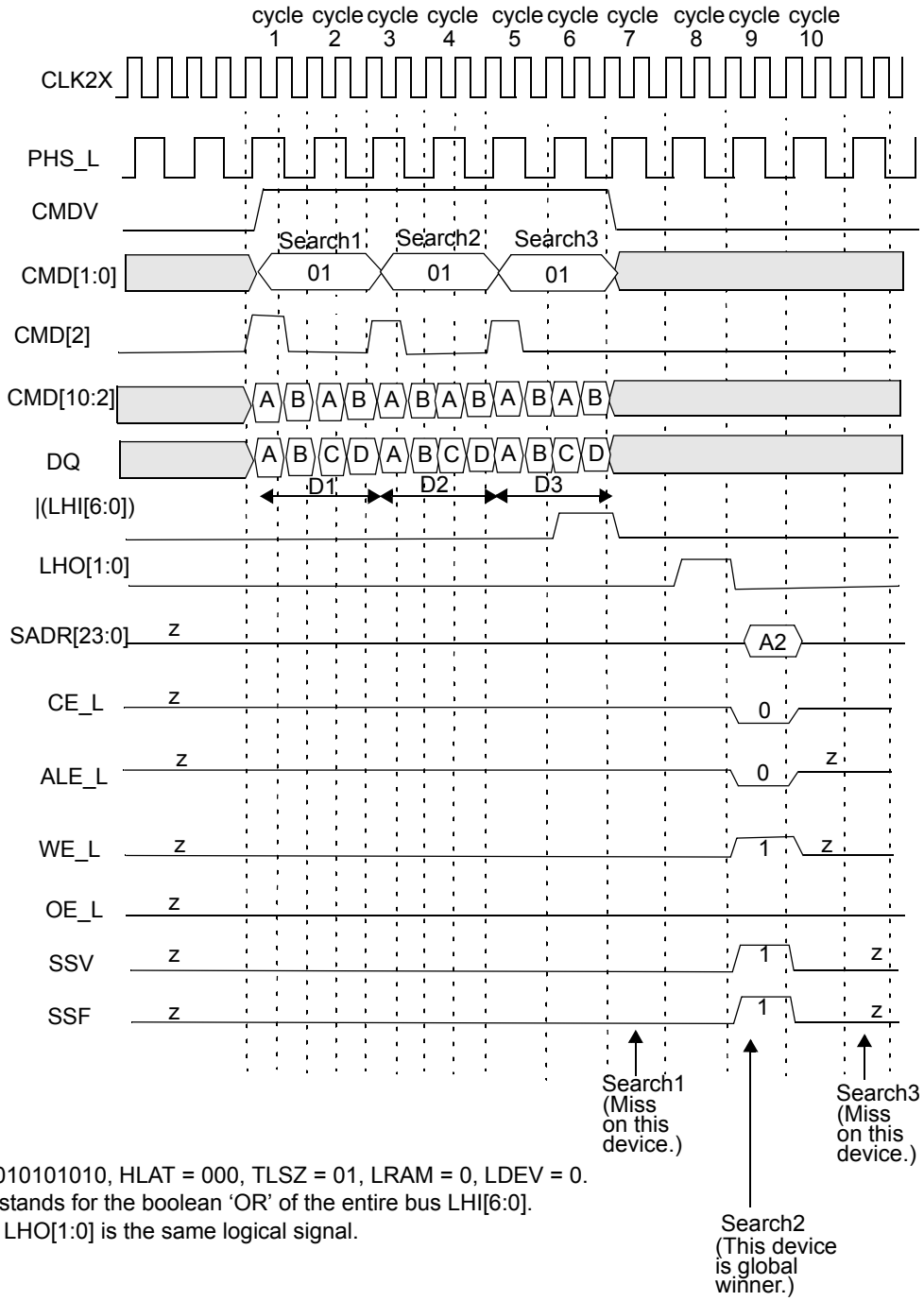


Figure 10-52. Hardware Diagram for a Table with Eight Devices



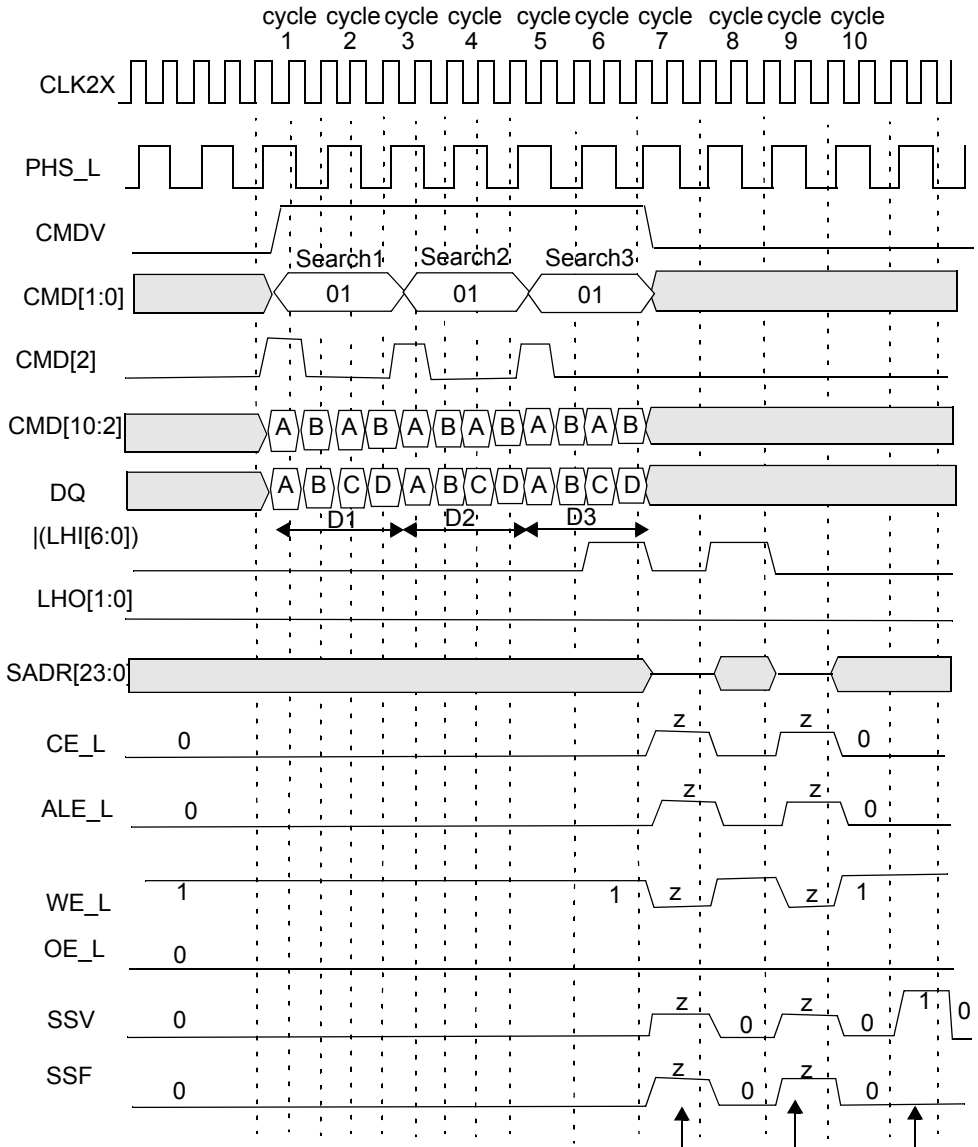
CFG = 1010101010101010, HLAT = 000, TLSZ = 01, LRAM = 0, LDEV = 0.
Note: |(LHI[6:0]) stands for the boolean 'OR' of the entire bus LHI[6:0].
Note: Each bit in LHO[1:0] is the same logical signal.

Figure 10-53. Timing Diagram for 288-bit Search Device Number 0



CFG = 1010101010101010, HLAT = 000, TLSZ = 01, LRAM = 0, LDEV = 0.
Note: |(LHI[6:0]) stands for the boolean 'OR' of the entire bus LHI[6:0].
Note: Each bit in LHO[1:0] is the same logical signal.

Figure 10-54. Timing Diagram for 288-bit Search Device Number 1



CFG = 1010101010101010, HLAT = 000, TLSZ = 01, LRAM = 1, LDEV = 1.
Note: |(LHI[6:0]) stands for the boolean 'OR' of the entire bus LHI[6:0].
Note: Each bit in LHO[1:0] is the same logical signal.

Figure 10-55. Timing Diagram for 288-bit Search Device Number 7 (Last Device)

The following is the sequence of operation for a single 288-bit Search command (also see "Commands and Command Parameters" on page 22).

- **Cycle A**: The host ASIC drives the CMDV high and applies Search command code (10) on CMD[1:0] signals. {CMD[10], CMD[5:3]} signals must be driven with the index to the GMR pair used for bits [287:144] of the data being searched in this operation. DQ[71:0] must be driven with the 72-bit data ([287:216]) to be compared against all locations 0 in the four-word 72-bit page. The CMD[2] signal must be driven to logic 1. ***Note***. CMD[2] = 1 signals that the search is a 288-bit search. CMD[8:3] in this cycle is ignored.
- **Cycle B**: The host ASIC continues to drive the CMDV high and applies Search command code (10) on CMD[1:0]. The DQ[71:0] is driven with the 72-bit data ([215:144]) to be compared against all locations 1 in the four 72-bits-word page.
- **Cycle C**: The host ASIC drives the CMDV high and applies Search command code (10) on CMD[1:0] signals. {CMD[10], CMD[5:3]} signals must be driven with the index to the GMR pair used for bits [143:0] of the data being searched. CMD[8:6] signals must be driven with the bits that will be driven on SADR[23:21] by this device if it has a hit. DQ[71:0] must be driven

with the 72-bit data ([143:72]) to be compared against all locations 2 in the four 72-bits-word page. The CMD[2] signal must be driven to logic 0.

- **Cycle D**: The host ASIC continues to drive the CMDV high and applies Search command code (10) on CMD[1:0]. CMD[8:6] signals must be driven with the index of the SSR that will be used for storing the address of the matching entry and the hit flag (see page 16 for the description of SSR[0:7]). The DQ[71:0] is driven with the 72-bit data ([71:0]) to be compared to all locations 3 in the four 72-bits-word page. CMD[5:2] is ignored because the Learn instruction is not supported for x288 tables.

Note. For 288-bit searches, the host ASIC must supply four distinct 72-bit data words on DQ[71:0] during cycles A, B, C, and D. The GMR index in cycle A selects a pair of GMRs in each of the eight devices that apply to DQ data in cycles A and B. The GMR index in cycle C selects a pair of GMRs in each of the eight devices that apply to DQ data in cycles C and D.

The logical 288-bit Search operation is shown in *Figure 10-56*. The entire table of 288-bit entries is compared to a 288-bit word K that is presented on the DQ bus in cycles A, B, C, and D of the command using the GMR and the local mask bits. The GMR is the 288-bit word specified by the two pairs of GMRs selected by the GMR Indexes in the command's cycles A and C in each of the eight devices. The 288-bit word K that is presented on the DQ bus in cycles A, B, C, and D of the command is compared to each entry in the table starting at location 0. The first matching entry's location address L is the winning address that is driven as part of the SRAM address on the SADR[23:0] lines (see "SRAM Addressing" on page 105). ***Note***. The matching address is always going to be a location 0 in a four-entry page for 288-bit Search (two LSBs of the matching index will be 00).

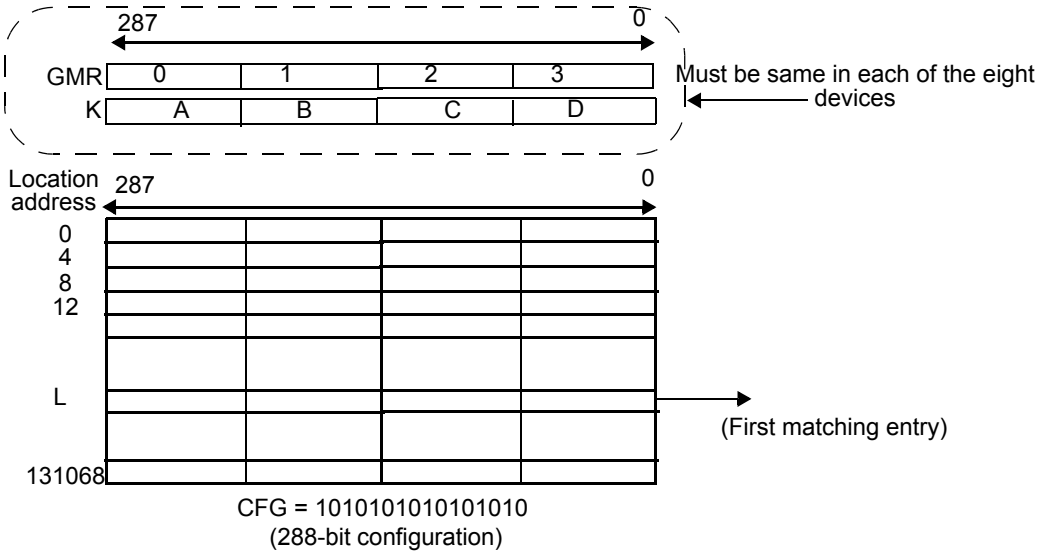


Figure 10-56. x288 Table with Eight Devices

The Search command is a pipelined operation and executes search at one-fourth the rate of the frequency of CLK2X for 288-bit searches in x288-configured tables. The latency of SADR, CE_L, ALE_L, WE_L, SSV, and SSF from the 288-bit Search command (measured in CLK cycles) from the CLK2X cycle that contains the C and D cycles is shown in *Table 10-29*.

Table 10-29. The Latency of Search from Cycles C and D to SRAM Access Cycle

Number of Devices	Max Table Size	Latency in CLK Cycles
1 (TLSZ = 00)	16K × 288 bits	4
1–8 (TLSZ = 01)	128K × 288 bits	5
1–31 (TLSZ = 10)	496K × 288 bits	6

The latency of search from command to SRAM access cycle is 5 for only a single device in the table and TLSZ = 01. In addition, SSV and SSF shift further to the right for different values of HLAT, as specified in *Table 10-30*.

Table 10-30. Shift of SSF and SSV from SADR

HLAT	Number of CLK Cycles
000	0
001	1
010	2
011	3
100	4

Table 10-30. Shift of SSF and SSV from SADR

HLAT	Number of CLK Cycles
101	5
110	6
111	7

10.6.9 288-bit Search on Tables Configured as x288 Using up to 31 CYNSE70128 Devices

The hardware diagram of the search subsystem of 31 devices is shown in *Figure 10-57*. Each of the four blocks in the diagram represents a block of eight CYNSE70128 devices, except the last which has seven devices. The diagram for a block of eight devices is shown in *Figure 10-58*. The following are the parameters programmed into the 31 devices.

- First thirty devices (devices 0–29): CFG = 1010101010101010, TLSZ = 10, HLAT = 000, LRAM = 0, and LDEV = 0.
- Thirty-first device (device 30): CFG = 1010101010101010, TLSZ = 10, HLAT = 000, LRAM = 1, and LDEV = 1.

Note. All 31 devices must be programmed with the same value of TLSZ and HLAT. Only the last device in the table must be programmed with LRAM = 1 and LDEV = 1 (device number 30 in this case). All other upstream devices must be programmed with LRAM = 0 and LDEV = 0 (devices 0 through 29 in this case).

The timing diagrams referred to in this paragraph reference the Hit/Miss assumptions defined in *Table 10-31*. For the purpose of illustrating the timings, it is further assumed that there is only one device with the matching entry in each block. *Figure 10-59* shows the timing diagram for a Search command in the 288-bit-configured table consisting of 31 devices for each of the eight devices in block number 0. *Figure 10-60* shows the timing diagram for a Search command in the 288-bit-configured table of 31 devices for all devices above the winning device in block number 1. *Figure 10-61* shows the timing diagram for the globally winning device (the final winner within its own and all blocks) in block number 1. *Figure 10-62* shows the timing diagram for all the devices below the globally winning device in block number 1. *Figure 10-63*, *Figure 10-64*, and *Figure 10-65*, respectively, show the timing diagrams of the devices above the globally winning device, the globally winning device, and the devices below the globally winning device for block number 2. *Figure 10-66*, *Figure 10-67*, *Figure 10-68*, and *Figure 10-69*, respectively, show the timing diagrams of the device above the globally winning device, the globally winning device, the devices below the globally winning device (except device 30), and last device (device 30) for block number 3.

The 288-bit Search operation is pipelined and executes as follows. Four cycles from the last cycle of the Search command each of the devices knows the outcome internal to it for that operation. In the fifth cycle from the Search command, the devices in a block (which is less than or equal to eight devices resolving the winner within them using an LHI[6:0] and LHO[1:0] signalling mechanism) arbitrate for a winner. In the sixth cycle after the Search command, the blocks of devices resolve the winning block through a BHI[2:0] and BHO[2:0] signalling mechanism. The winning device within the winning block is the global winning device for the Search operation.

Table 10-31. Hit/Miss Assumption

Search Number	1	2	3
Block 0	Miss	Miss	Miss
Block 1	Miss	Miss	Hit
Block 2	Miss	Hit	Hit
Block 3	Hit	Hit	Miss

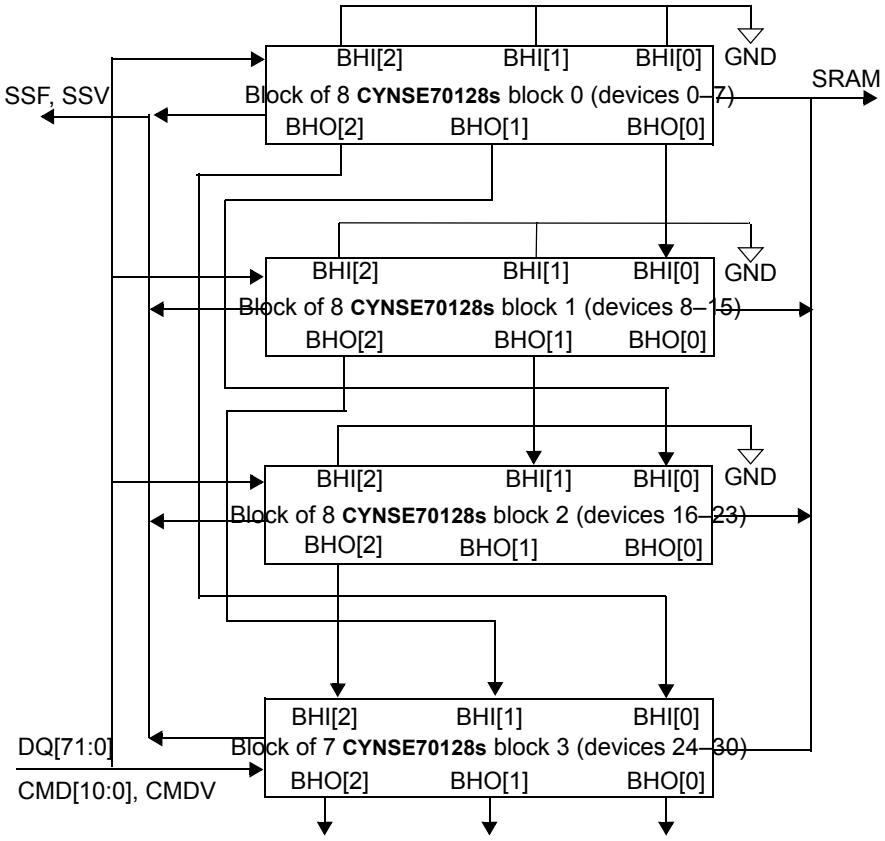


Figure 10-57. Hardware Diagram for a Table with 31 Devices

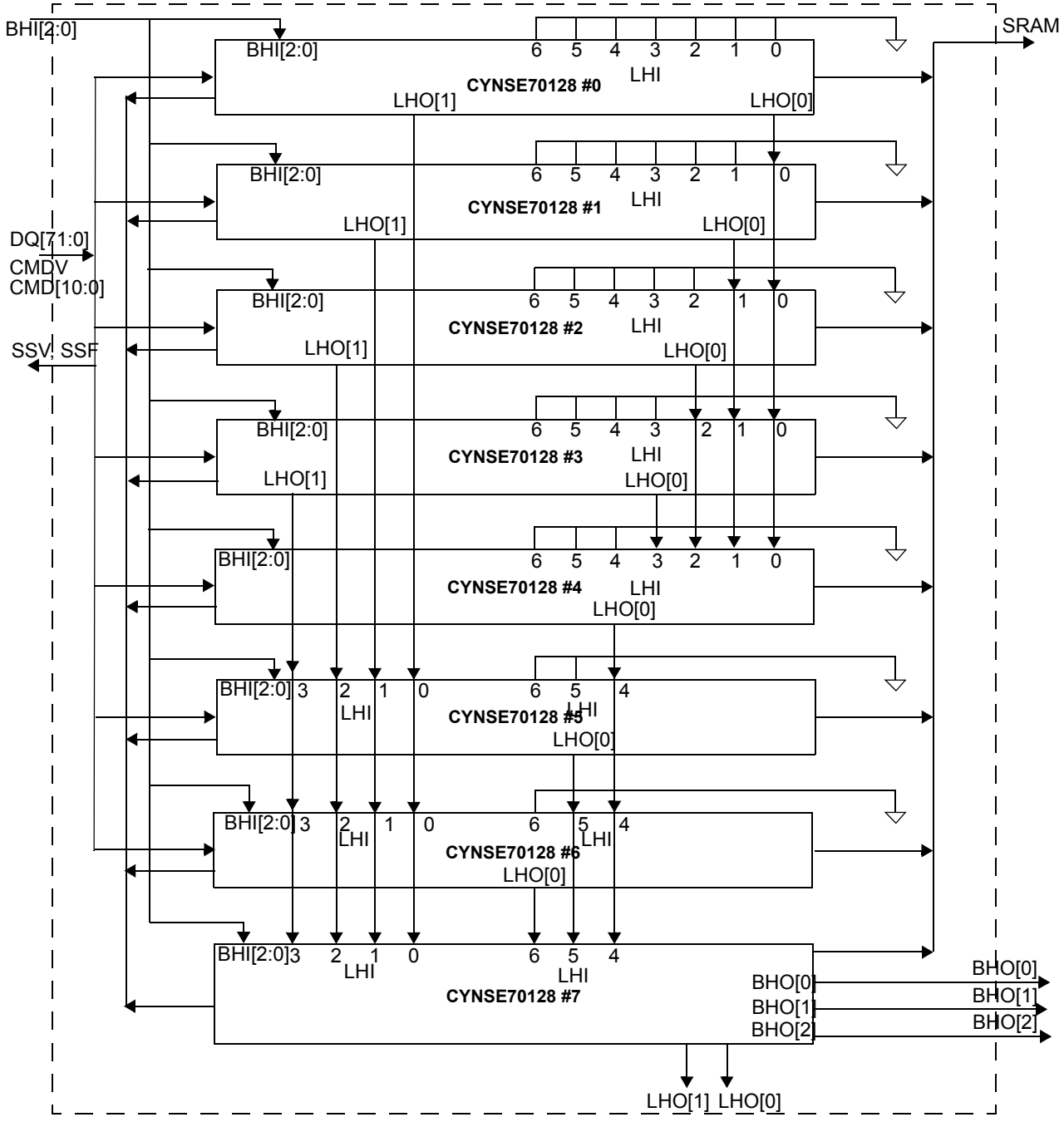
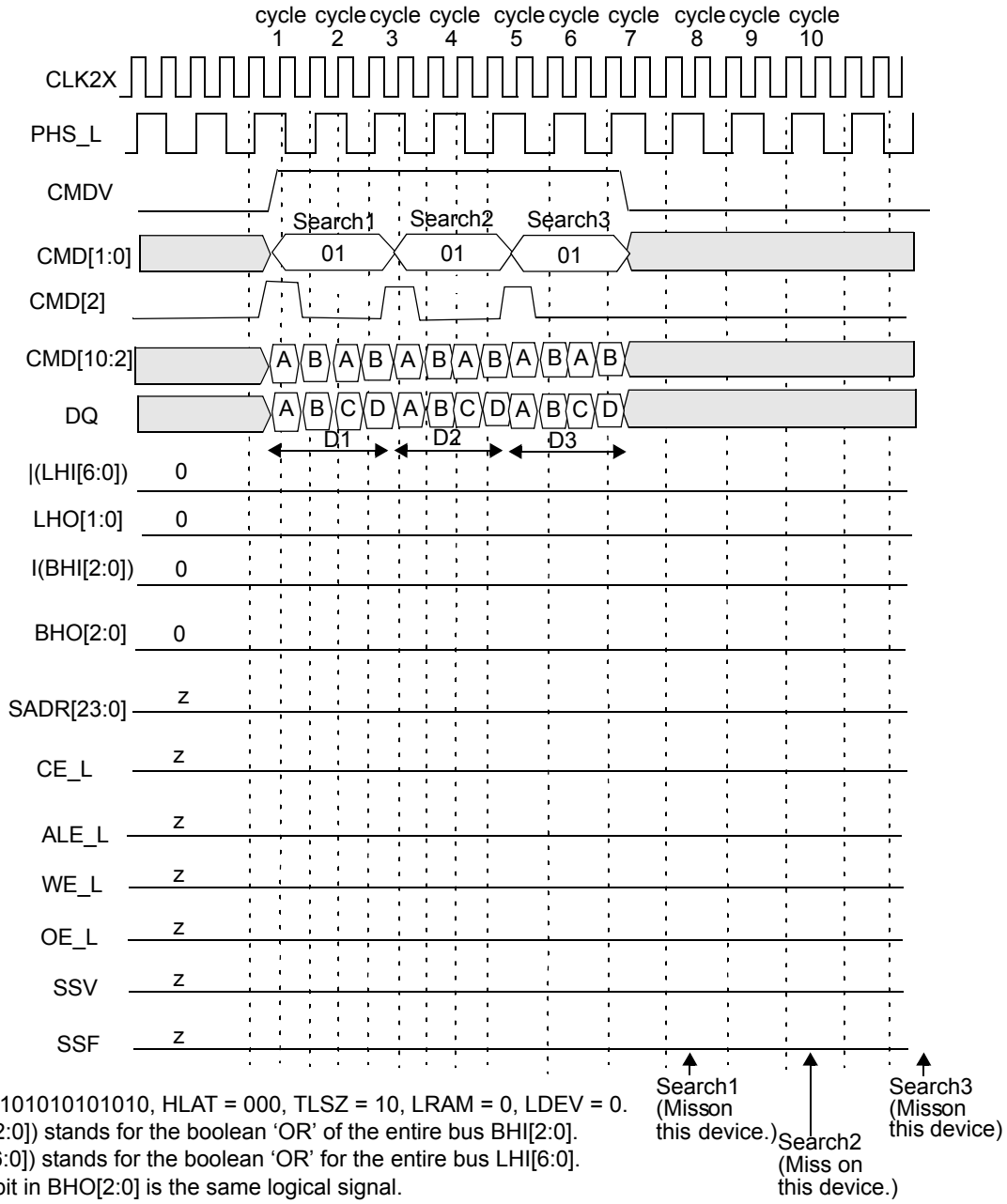
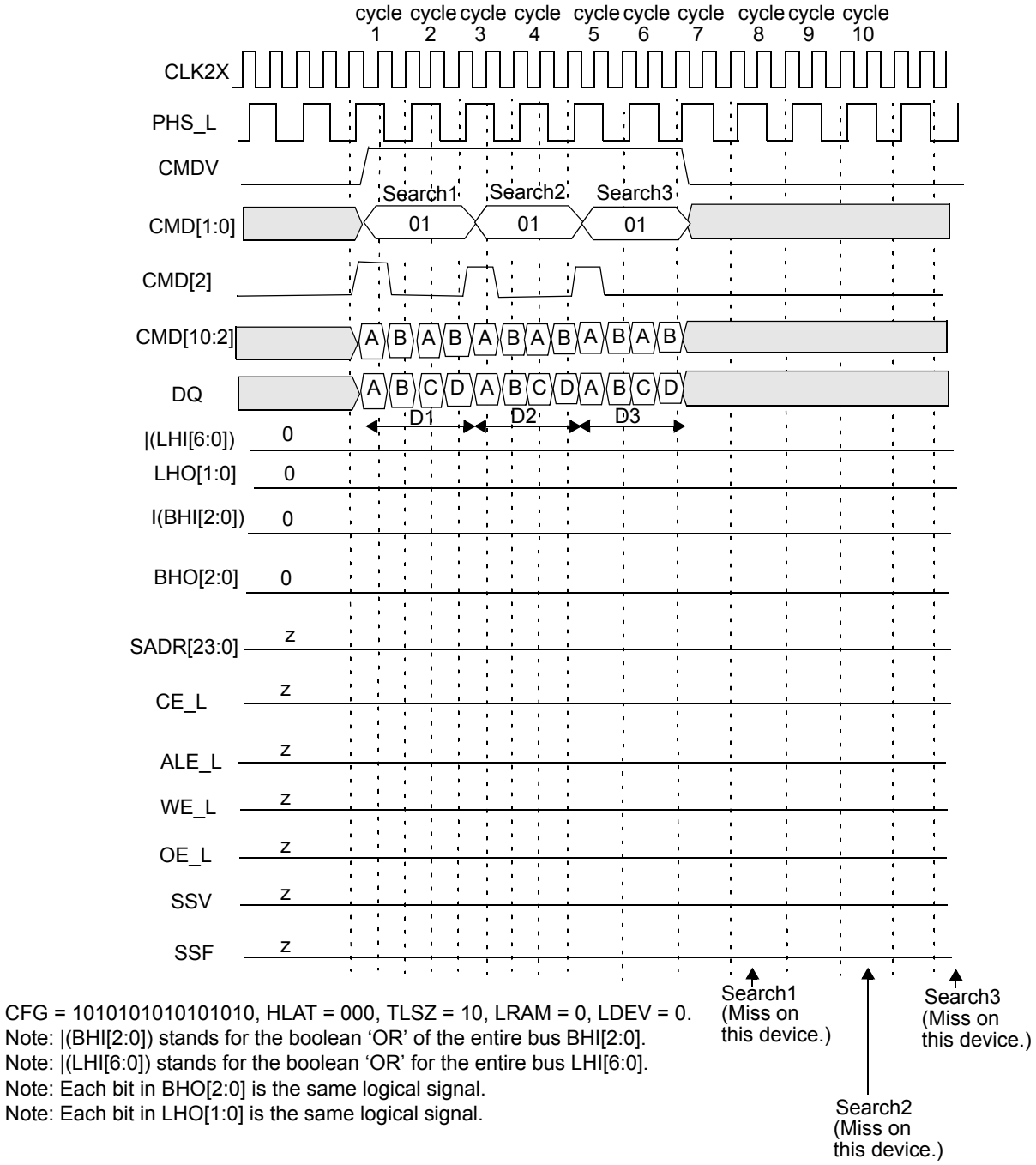


Figure 10-58. Hardware Diagram for a Block of up to Eight Devices



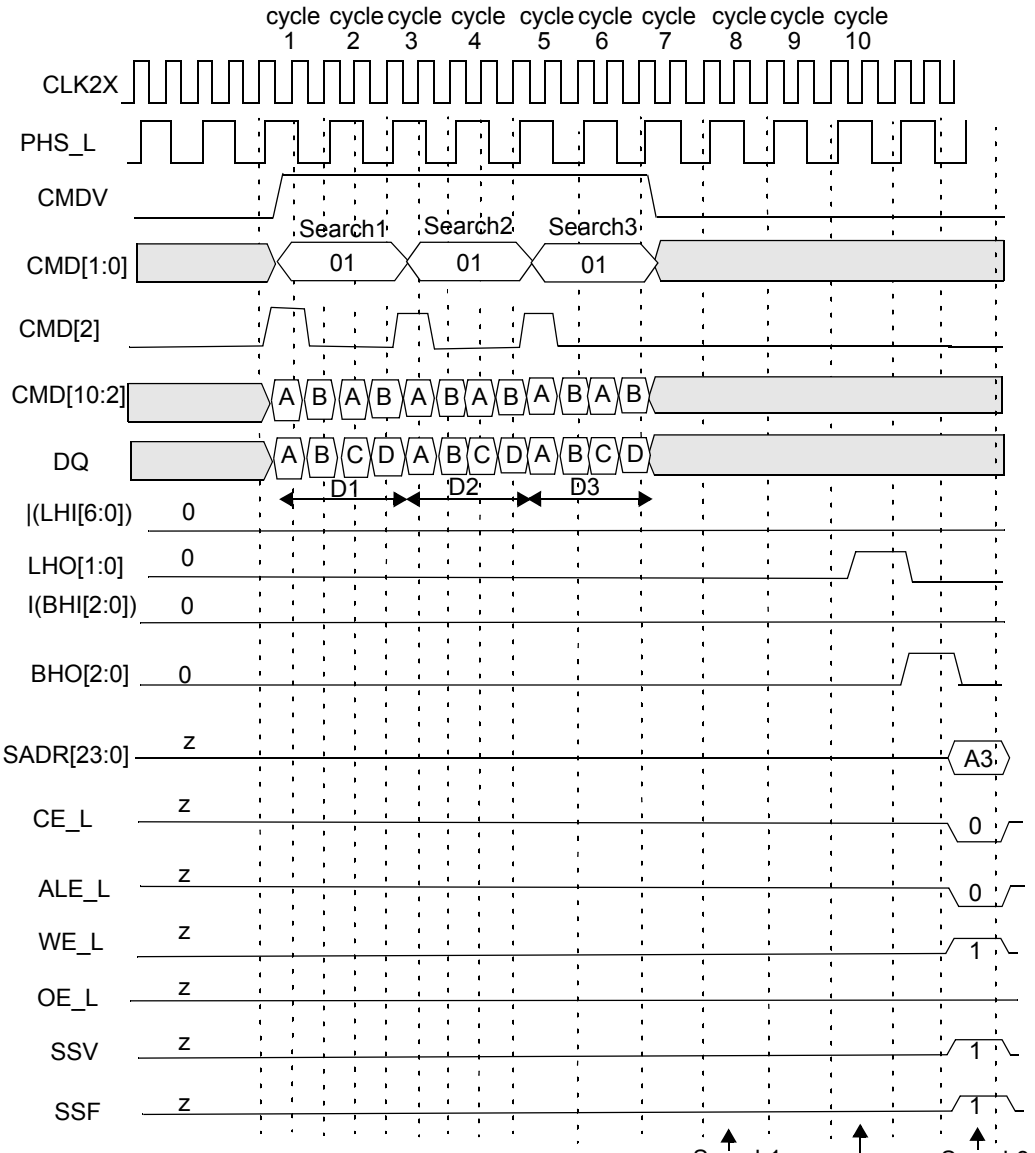
CFG = 1010101010101010, HLAT = 000, TLSZ = 10, LRAM = 0, LDEV = 0.
Note: |(BHI[2:0]) stands for the boolean 'OR' of the entire bus BHI[2:0].
Note: |(LHI[6:0]) stands for the boolean 'OR' for the entire bus LHI[6:0].
Note: Each bit in BHO[2:0] is the same logical signal.
Note: Each bit in LHO[1:0] is the same logical signal.

Figure 10-59. Timing Diagram for Each Device in Block Number 0 (Miss on Each Device)



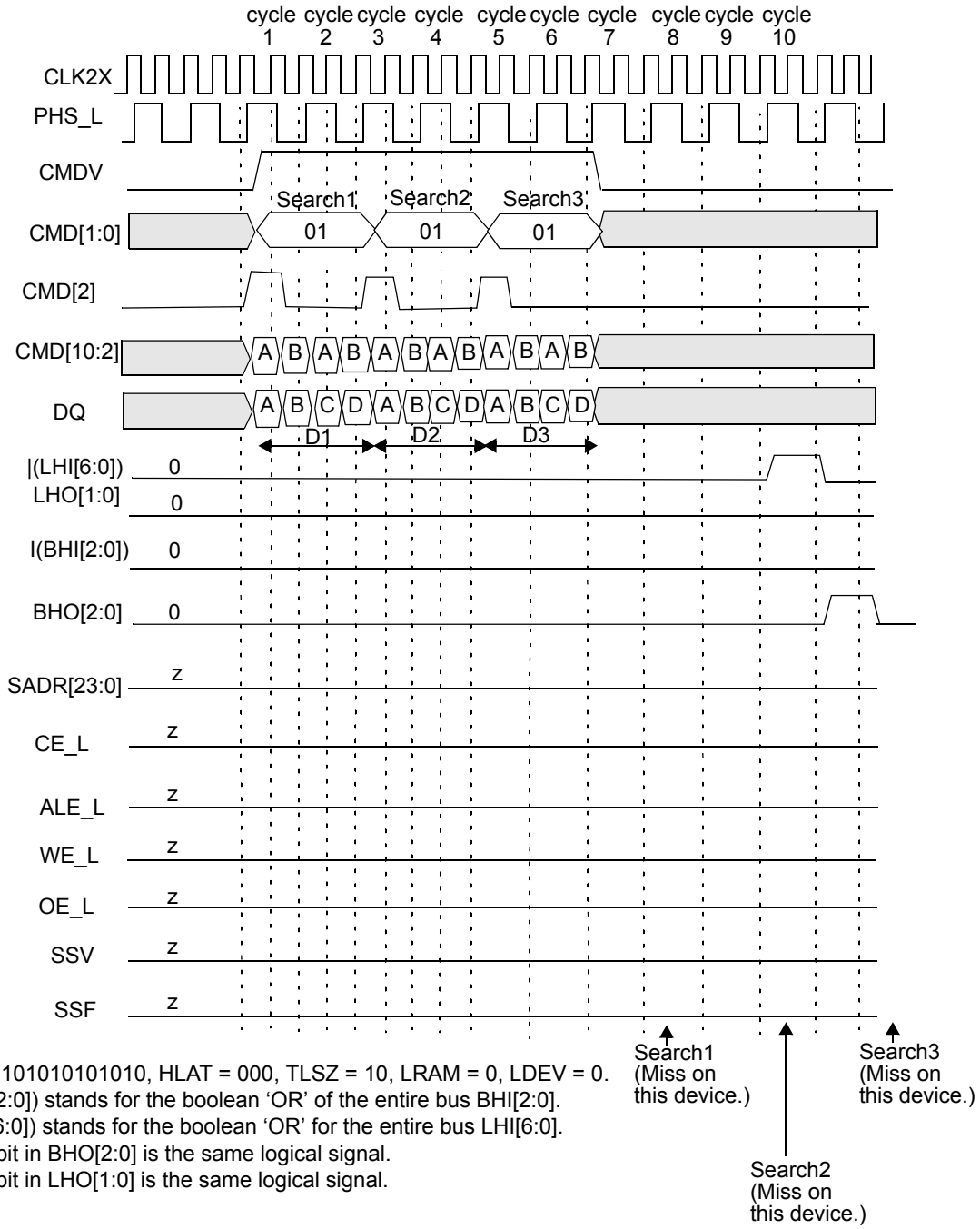
CFG = 1010101010101010, HLAT = 000, TLSZ = 10, LRAM = 0, LDEV = 0.
Note: |(BHI[2:0]) stands for the boolean 'OR' of the entire bus BHI[2:0].
Note: |(LHI[6:0]) stands for the boolean 'OR' for the entire bus LHI[6:0].
Note: Each bit in BHO[2:0] is the same logical signal.
Note: Each bit in LHO[1:0] is the same logical signal.

Figure 10-60. Timing Diagram for Each Device Above the Winning Device in Block Number 1



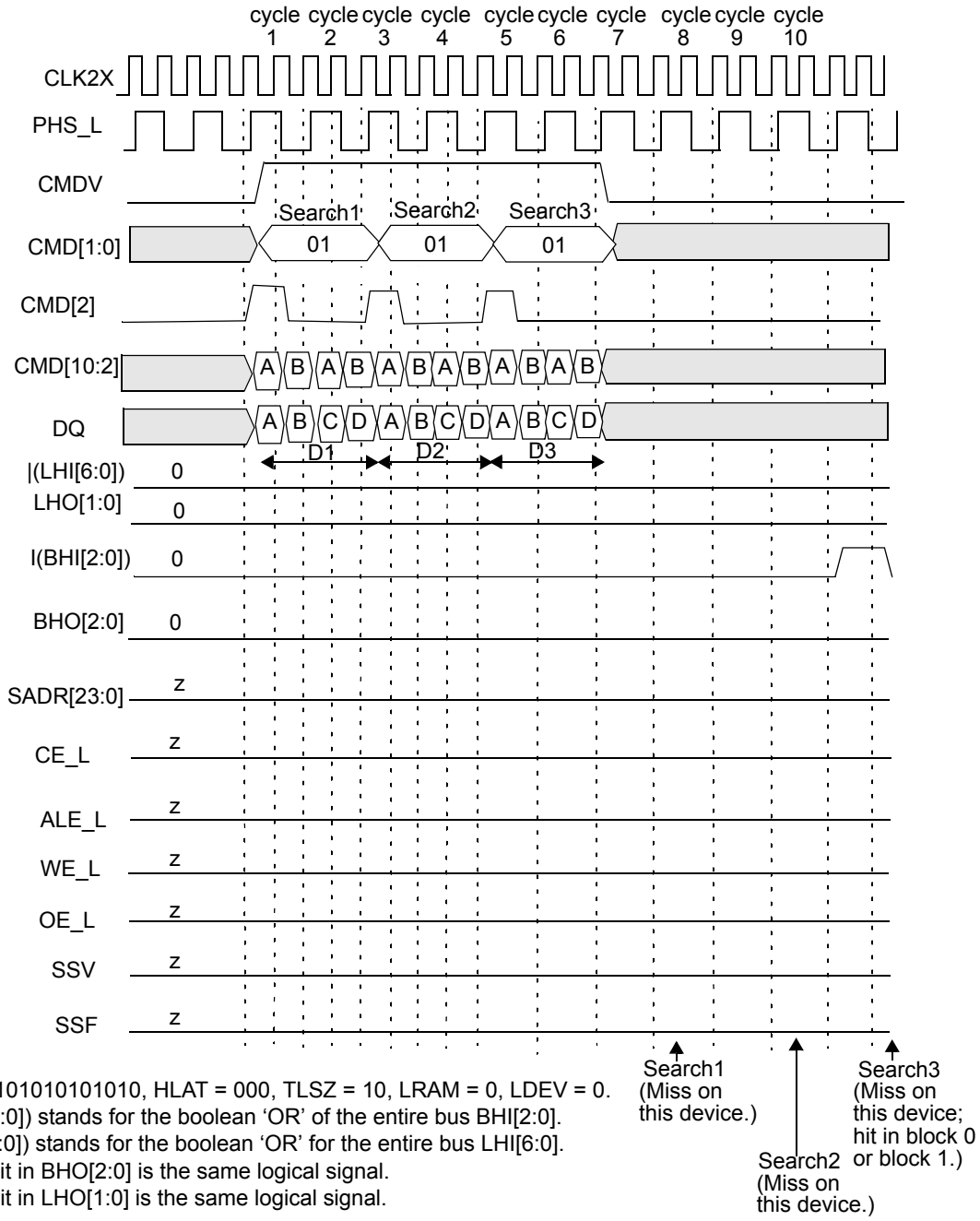
CFG = 1010101010101010, HLAT = 000, TLSZ = 10, LRAM = 0, LDEV = 0.
Note: |(BHI[2:0]) stands for the boolean 'OR' of the entire bus BHI[2:0].
Note: |(LHI[6:0]) stands for the boolean 'OR' for the entire bus LHI[6:0].
Note: Each bit in BHO[2:0] is the same logical signal.
Note: Each bit in LHO[1:0] is the same logical signal.

Figure 10-61. Timing Diagram for Globally Winning Device in Block Number 1



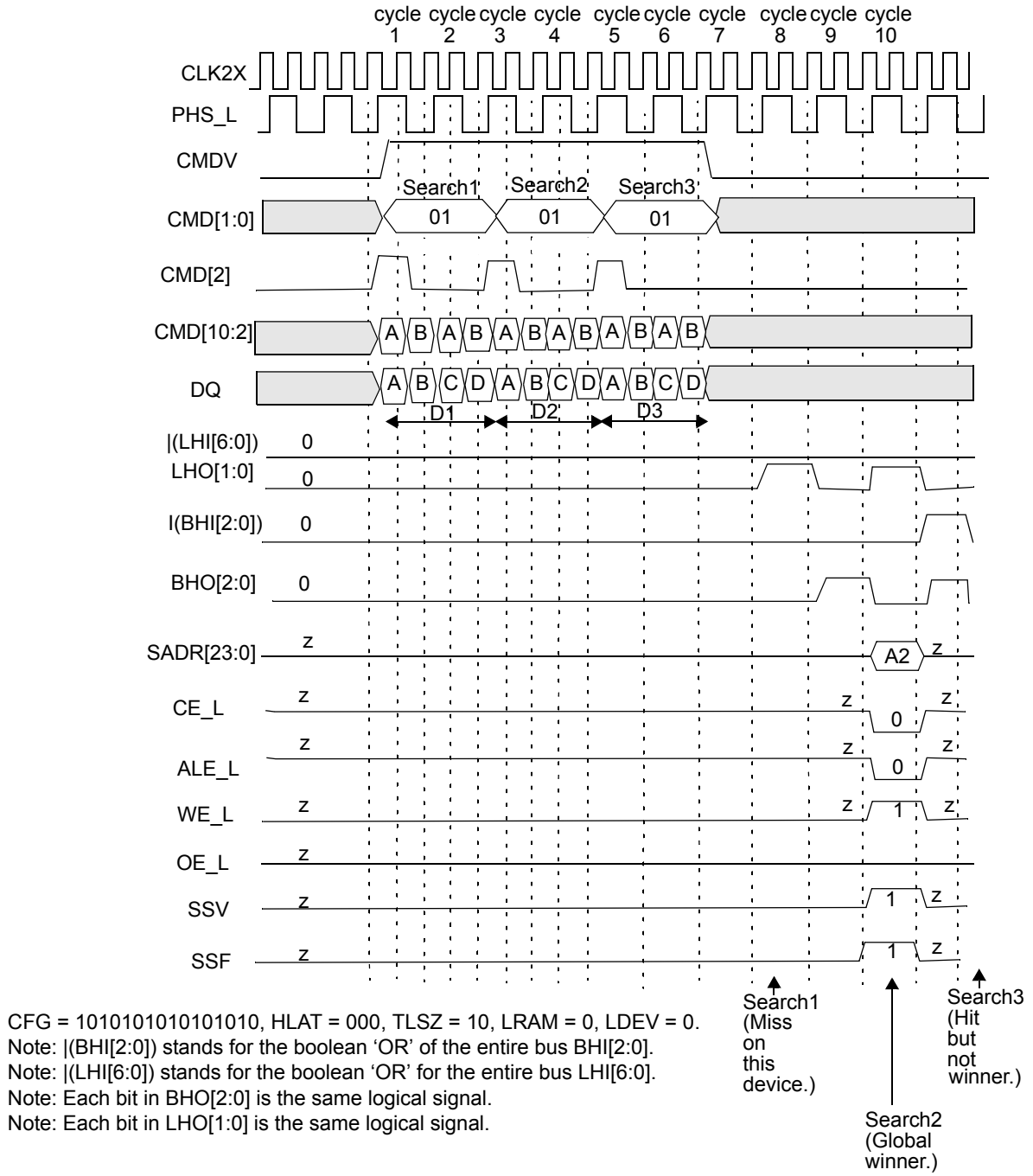
CFG = 1010101010101010, HLAT = 000, TLSZ = 10, LRAM = 0, LDEV = 0.
Note: |(BHI[2:0]) stands for the boolean 'OR' of the entire bus BHI[2:0].
Note: |(LHI[6:0]) stands for the boolean 'OR' for the entire bus LHI[6:0].
Note: Each bit in BHO[2:0] is the same logical signal.
Note: Each bit in LHO[1:0] is the same logical signal.

Figure 10-62. Timing Diagram for Devices Below the Winning Device in Block Number 1



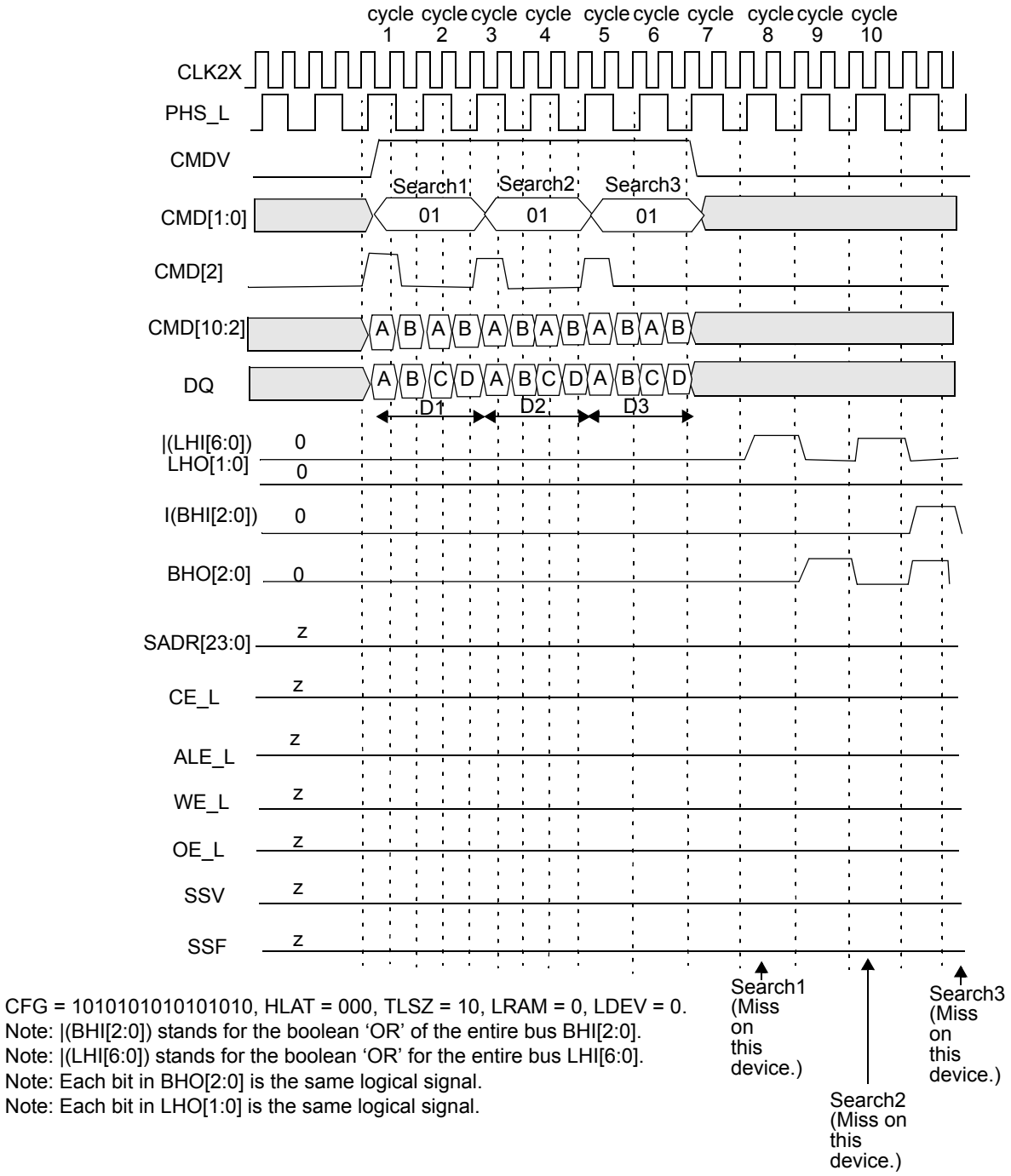
CFG = 1010101010101010, HLAT = 000, TLSZ = 10, LRAM = 0, LDEV = 0.
Note: |(BHI[2:0]) stands for the boolean 'OR' of the entire bus BHI[2:0].
Note: |(LHI[6:0]) stands for the boolean 'OR' for the entire bus LHI[6:0].
Note: Each bit in BHO[2:0] is the same logical signal.
Note: Each bit in LHO[1:0] is the same logical signal.

Figure 10-63. Timing Diagram for Devices Above the Winning Device in Block Number 2



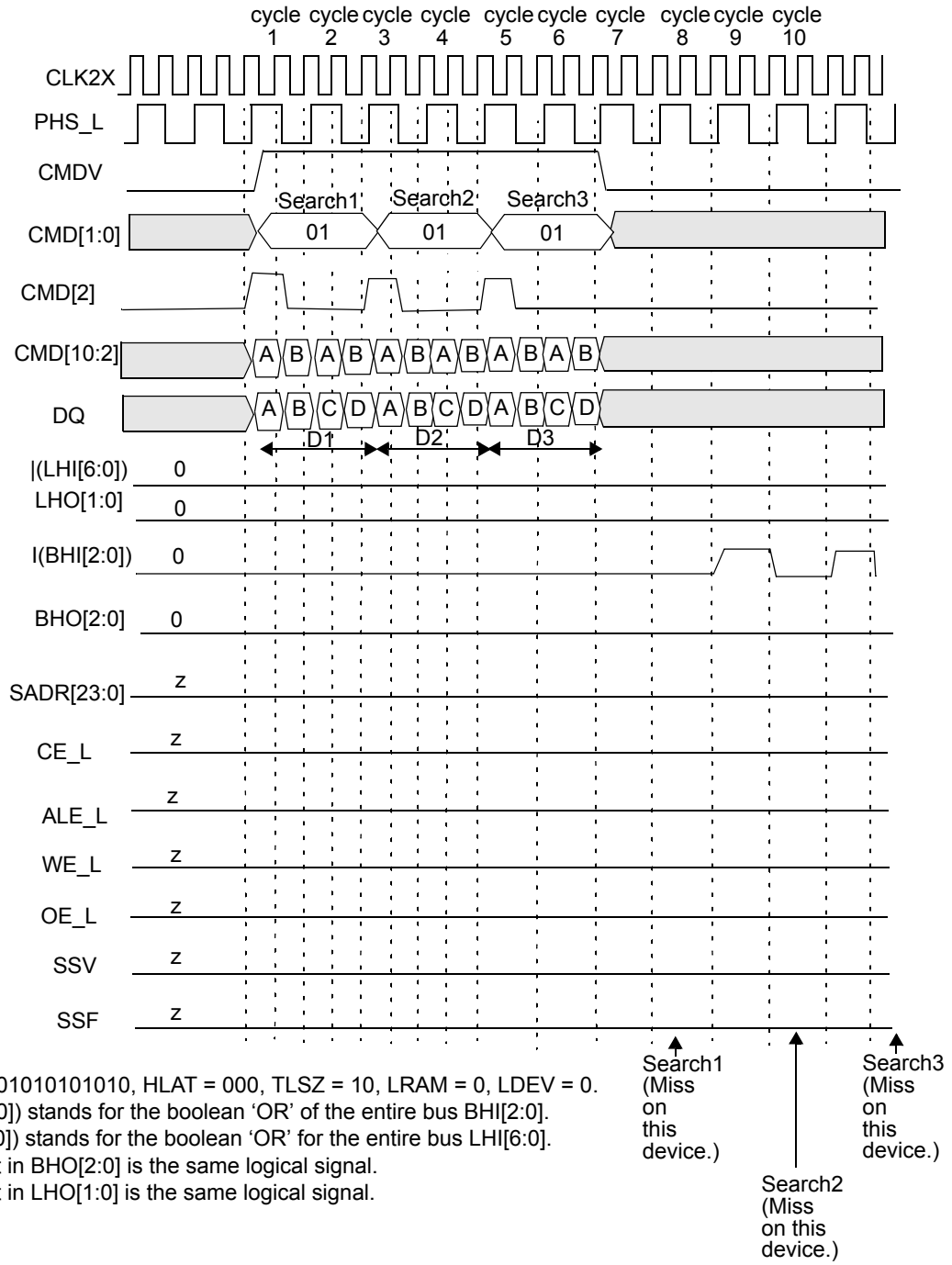
CFG = 1010101010101010, HLAT = 000, TLSZ = 10, LRAM = 0, LDEV = 0.
Note: |(BHI[2:0]) stands for the boolean 'OR' of the entire bus BHI[2:0].
Note: |(LHI[6:0]) stands for the boolean 'OR' for the entire bus LHI[6:0].
Note: Each bit in BHO[2:0] is the same logical signal.
Note: Each bit in LHO[1:0] is the same logical signal.

Figure 10-64. Timing Diagram for Globally Winning Device in Block Number 2



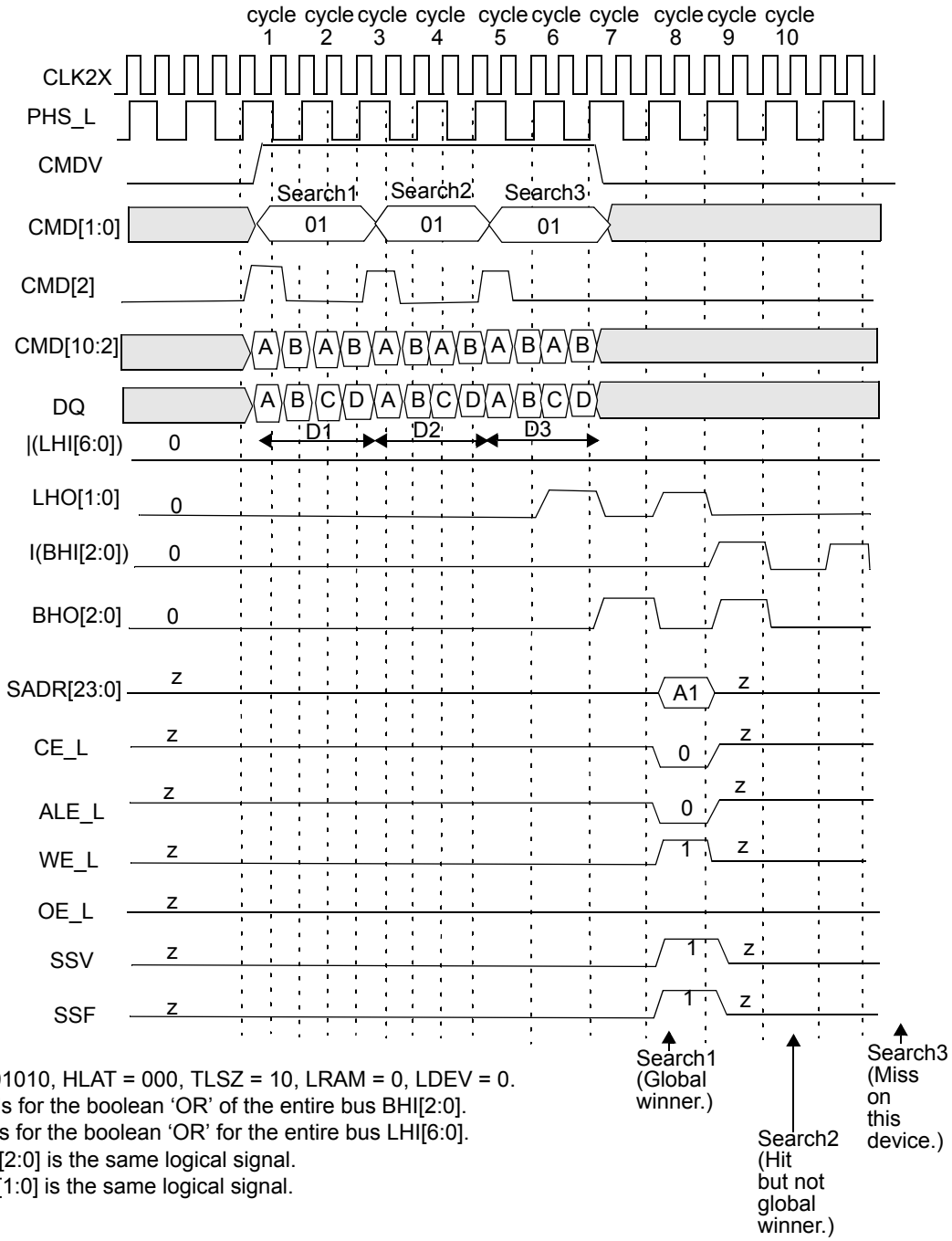
CFG = 1010101010101010, HLAT = 000, TLSZ = 10, LRAM = 0, LDEV = 0.
Note: |(BHI[2:0]) stands for the boolean 'OR' of the entire bus BHI[2:0].
Note: |(LHI[6:0]) stands for the boolean 'OR' for the entire bus LHI[6:0].
Note: Each bit in BHO[2:0] is the same logical signal.
Note: Each bit in LHO[1:0] is the same logical signal.

Figure 10-65. Timing Diagram for Devices Below the Winning Device in Block Number 2



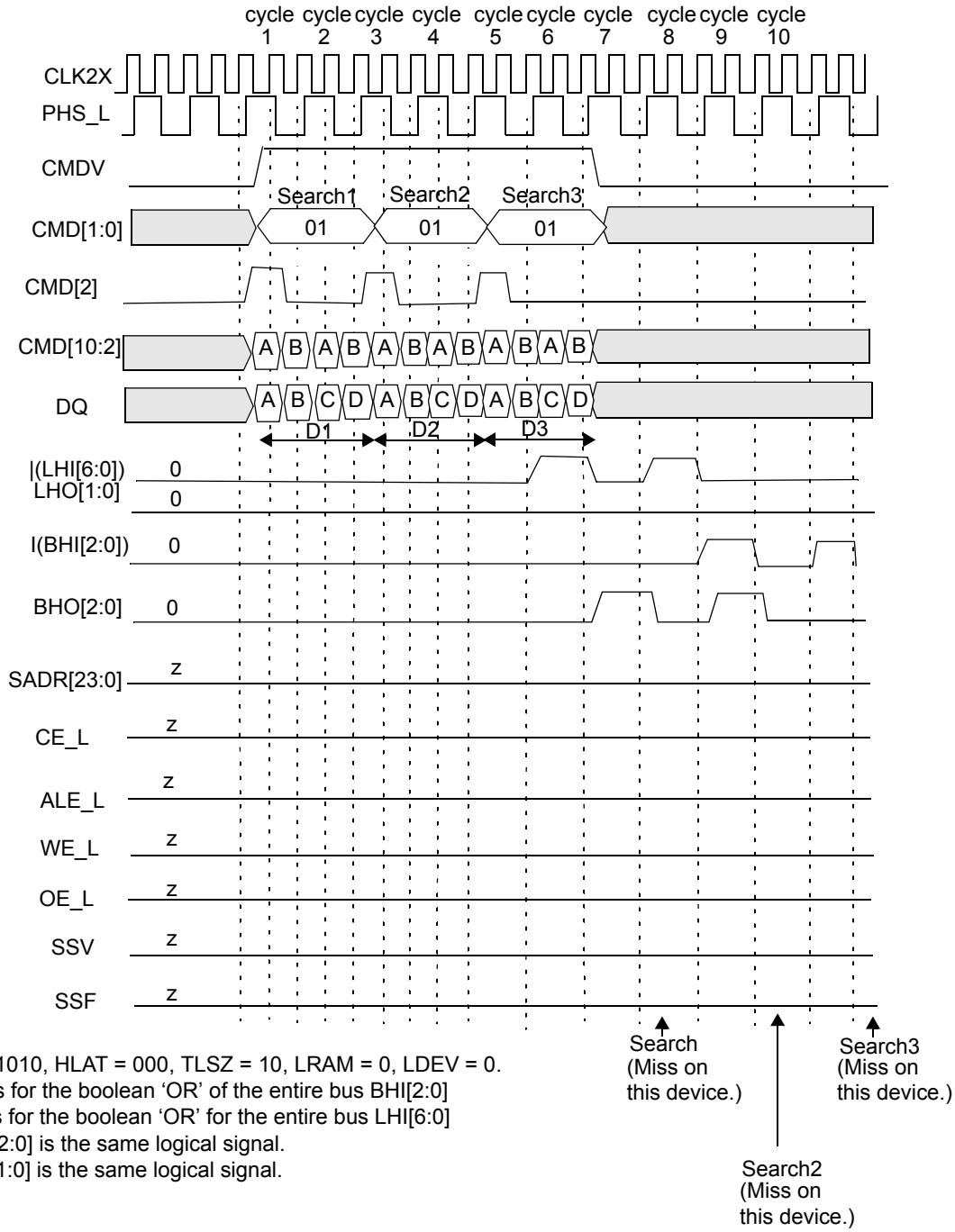
CFG = 1010101010101010, HLAT = 000, TLSZ = 10, LRAM = 0, LDEV = 0.
Note: |(BHI[2:0]) stands for the boolean 'OR' of the entire bus BHI[2:0].
Note: |(LHI[6:0]) stands for the boolean 'OR' for the entire bus LHI[6:0].
Note: Each bit in BHO[2:0] is the same logical signal.
Note: Each bit in LHO[1:0] is the same logical signal.

Figure 10-66. Timing Diagram for Devices Above the Winning Device in Block Number 3



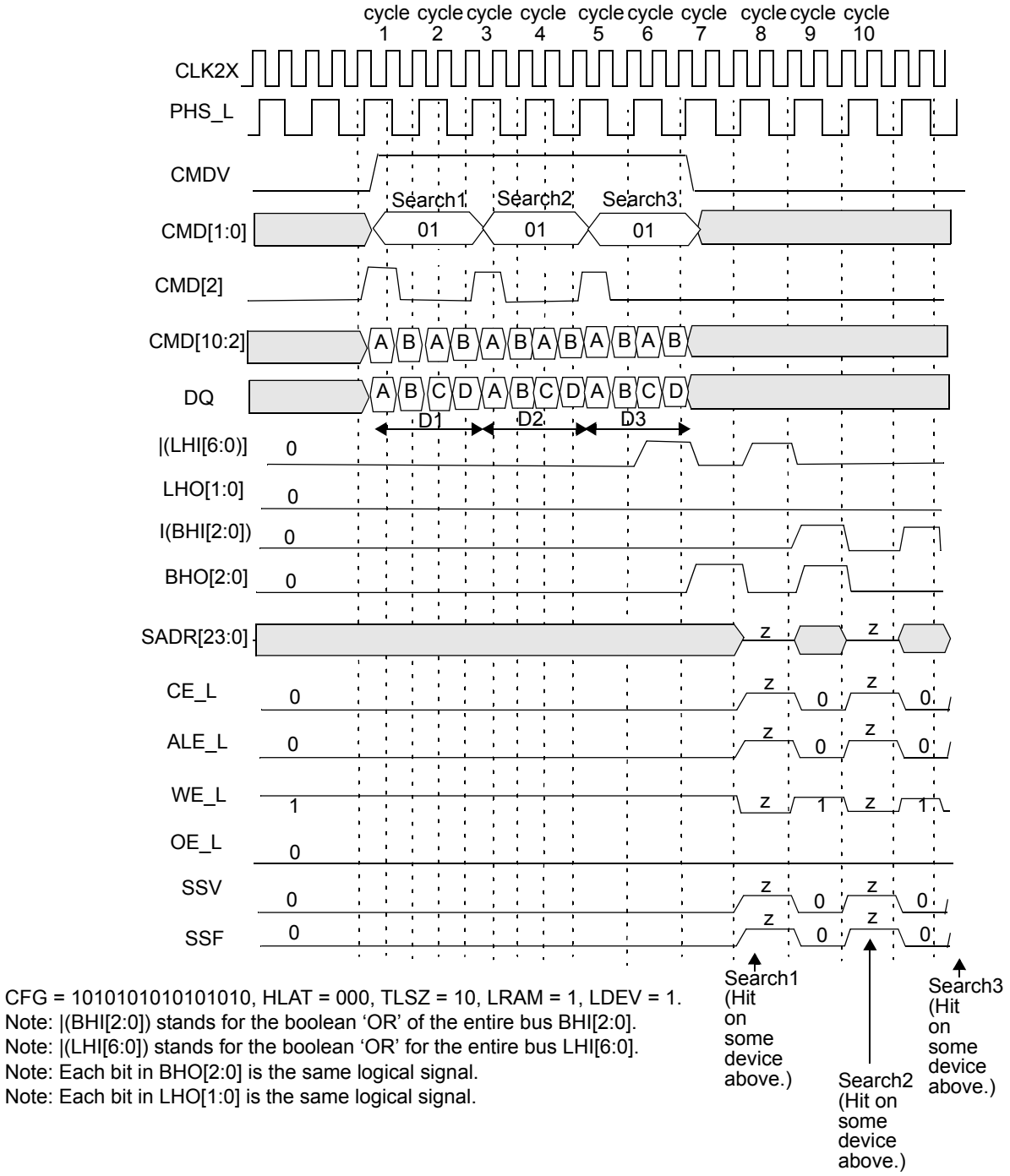
CFG = 1010101010101010, HLAT = 000, TLSZ = 10, LRAM = 0, LDEV = 0.
Note: |(BHI[2:0]) stands for the boolean 'OR' of the entire bus BHI[2:0].
Note: |(LHI[6:0]) stands for the boolean 'OR' for the entire bus LHI[6:0].
Note: Each bit in BHO[2:0] is the same logical signal.
Note: Each bit in LHO[1:0] is the same logical signal.

Figure 10-67. Timing Diagram for Globally Winning Device in Block Number 3



CFG = 1010101010101010, HLAT = 000, TLSZ = 10, LRAM = 0, LDEV = 0.
Note: |(BHI[2:0]) stands for the boolean 'OR' of the entire bus BHI[2:0]
Note: |(LHI[6:0]) stands for the boolean 'OR' for the entire bus LHI[6:0]
Note: Each bit in BHO[2:0] is the same logical signal.
Note: Each bit in LHO[1:0] is the same logical signal.

Figure 10-68. Timing Diagram for Devices Below the Winning Device in Block Number 3 Except Device 30 (the Last Device)



CFG = 1010101010101010, HLAT = 000, TLSZ = 10, LRAM = 1, LDEV = 1.
Note: |(BHI[2:0]) stands for the boolean 'OR' of the entire bus BHI[2:0].
Note: |(LHI[6:0]) stands for the boolean 'OR' for the entire bus LHI[6:0].
Note: Each bit in BHO[2:0] is the same logical signal.
Note: Each bit in LHO[1:0] is the same logical signal.

Figure 10-69. Timing Diagram of the Last Device in Block Number 3 (Device 30 in the Table)

The following is the sequence of operation for a single 288-bit Search command (also refer to "Commands and Command Parameters" on page 22).

- **Cycle A**: The host ASIC drives the CMDV high and applies Search command code (10) on CMD[1:0] signals. {CMD[10], CMD[5:3]} signals must be driven with the index to the GMR pair used for bits [287:144] of the data being searched. DQ[71:0] must be driven with the 72-bit data ([287:216])to be compared to all locations 0 in the four 72-bits-word page. The CMD[2] signal must be driven to logic 1. ***Note***. CMD[2] = 1 signals that the search is a x288-bit search. CMD[8:6] is ignored in this cycle.
- **Cycle B**: The host ASIC continues to drive the CMDV high and applies Search command (10) on CMD[1:0]. The DQ[71:0] is driven with the 72-bit data ([215:144]) to be compared to all locations 1 in the four 72-bits-word page.

- **Cycle C**: The host ASIC drives the CMDV high and applies Search command code (10) on CMD[1:0] signals. {CMD[10], CMD[5:3]} signals must be driven with the index to the GMR pair used for the bits [143:0] of the data being searched. CMD[8:6] signals must be driven with the bits that will be driven by this device on SADR[23:21] if it has a hit. DQ[71:0] must be driven with the 72-bit data ([143:72]) to be compared to all locations 2 in the four 72-bits-word page. The CMD[2] signal must be driven to logic 0.
- **Cycle D**: The host ASIC continues to drive the CMDV high and continues to apply Search command code (10) on CMD[1:0]. CMD[8:6] signals must be driven with the index of the SSR that will be used for storing the address of the matching entry and the hit flag (see page 16 for a description of SSR[0:7]). The DQ[71:0] is driven with the 72-bit data ([71:0]) to be compared to all locations 3 in the four 72-bits-word page. CMD[5:2] is ignored because the Learn instruction is not supported for x288 tables.

Note. For 288-bit searches, the host ASIC must supply four distinct 72-bit data words on DQ[71:0] during cycles A, B, C, and D. The GMR Index in cycle A selects a pair of GMRs in each of the 31 devices that apply to DQ data in cycles A and B. The GMR Index in cycle C selects a pair of GMRs in each of the 31 devices that apply to DQ data in cycles C and D.

The logical 288-bit Search operation is as shown in *Figure 10-70*. The entire table of 288-bit entries is compared to a 288-bit word K that is presented on the DQ bus in cycles A, B, C, and D of the command using the GMR and local mask bits. The GMR is the 288-bit word specified by the two pairs of GMRs selected by the GMR Indexes in the command's cycles A and C in each of the 31 devices. The 288-bit word K that is presented on the DQ bus in cycles A, B, C, and D of the command is compared to each entry in the table starting at location 0. The first matching entry's location address L is the winning address that is driven as part of the SRAM address on the SADR[23:0] lines (see see "SRAM Addressing" on page 105). ***Note***. The matching address is always going to be location 0 in a four-entry page for 288-bit search (two LSBs of the matching index will be 00).

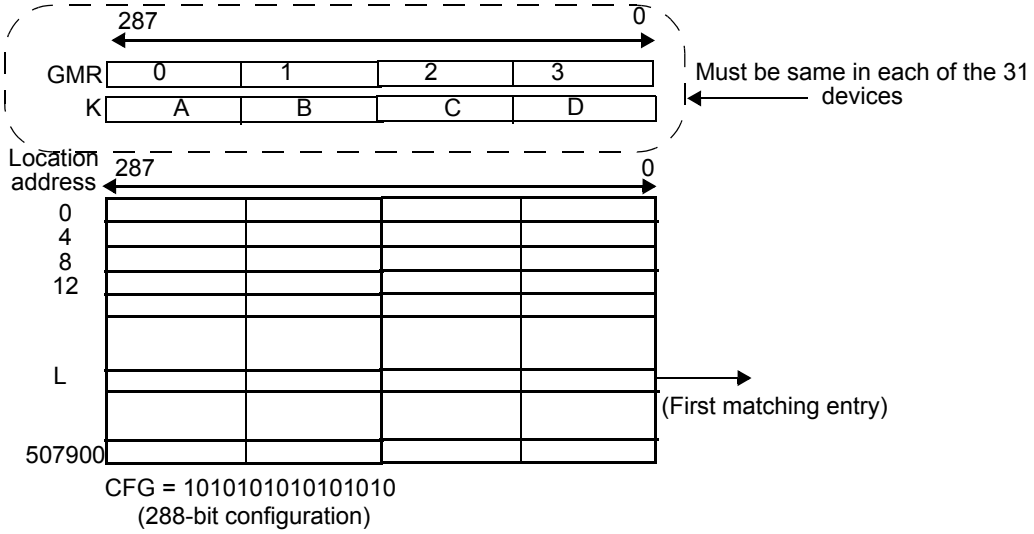


Figure 10-70. x288 Table with 31 Devices

The Search command is a pipelined operation and executes a search at one-fourth the rate of the frequency of CLK2X for 288-bit searches in x288-configured tables. The latency of SADR, CE_L, ALE_L, WE_L, SSV, and SSF from the 288-bit Search command (measured in CLK cycles) from the CLK2X cycle that contains the C and D cycles is shown in *Table 10-32*.

Table 10-32. The Latency of Search from Cycles C and D to SRAM Access Cycle

Number of Devices	Max Table Size	Latency in CLK Cycles
1 (TLSZ = 00)	16K × 288 bits	4
1–8 (TLSZ = 01)	128K × 288 bits	5
1–31 (TLSZ = 10)	496K × 288 bits	6

The latency of a Search from command to SRAM access cycle is 6 for only a single device in the table and TLSZ = 10. In addition, SSV and SSF shift further to the right for different values of HLAT, as specified in *Table 10-33*.

Table 10-33. Shift of SSF and SSV from SADR

HLAT	Number of CLK Cycles
000	0
001	1
010	2
011	3

Table 10-33. Shift of SSF and SSV from SADR (continued)

HLAT	Number of CLK Cycles
100	4
101	5
110	6
111	7

10.6.10 Mixed-Size Searches on Tables Configured with Different Widths Using a CYNSE70128 with CFG_L LOW

This subsection will cover mixed searches (×72, ×144, and ×288) with tables of different widths (×72, ×144, ×288). The sample operation shown is for a single device with CFG = 1010010100000000 containing three tables of ×72, ×144, and ×288 widths. The operation can be generalized to a block of 8–31 devices using four blocks; the timing and the pipeline operation is the same as described previously for fixed searches on a table of one-width-size.

Figure 10-71 shows three sequential searches: first, a 72-bit search on the table configured as ×72, then a 144-bit search on a table configured as ×144, and finally a 288-bit search on the table configured as ×288 bits that each results in a hit. ***Note***. The DQ[71:70] will be 00 in each of the two A and B cycles of the ×72-bit search (Search1). DQ[71:70] is 01 in each of the A and B cycles of the ×144-bit search (Search2). DQ[71:70] is 10 in each of the A, B, C, and D cycles of the ×288-bit search (Search3). By having table designation bits, the CYNSE70128 enables the creation of many tables in a bank of NSEs of different widths.

Figure 10-72 shows the sample table. Two bits in each 72-bit entry will need to designated as the table number bits. One example choice can be the 00 values for the table configured as ×72, 01 values for tables configured as ×144, and 10 values for tables configured as ×288. For the above explanation, it is further assumed that bits [71:70] for each entry will be designed as such table designation bits.

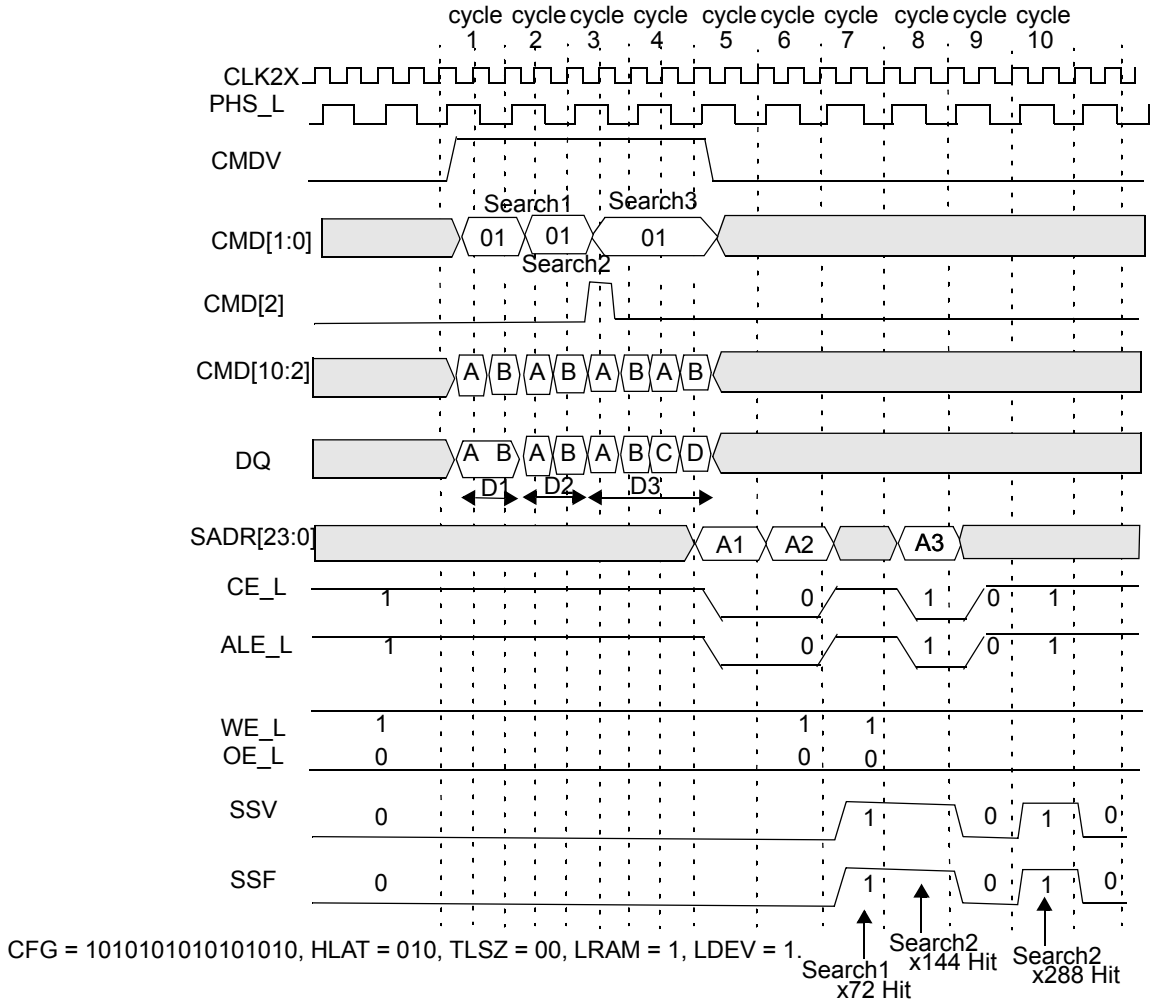
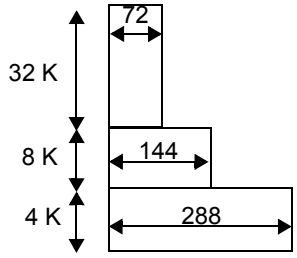


Figure 10-71. Timing Diagram for Mixed Search (One Device)



CFG = 10 10 01 01 00 00 00 00

Figure 10-72. Multiwidth Configurations Example

10.6.11 Mixed-Size Searches on Tables Configured to Different Widths Using a CYNSE70128 with CFG_L HIGH

This subsection will cover the mixed-size searches (×72, ×144, and ×288) with tables of different widths (×72, ×144, ×288) with CFG_L set high. The previous subsection described searches on tables of different widths using table designation bits in the data array. This can be wasteful of the bits in the data array. In order to avoid the waste of these bits and yet support up to three tables of ×72, ×144, and ×288, the CMD[2] and CMD[9] (in CFG_L high mode) in cycle A of the command can be used as shown in *Table 10-34*.

Table 10-34. Searches with CFG_L Set High

CMD[9]	CMD[2]	Search
0	0	Search 72-bit-configured partitions only.
1	0	Search 144-bit-configured partitions only.
X	1	Cycles A and B for searching 288-bit-configured partitions.
X	0	Cycles C and D for searching 288-bit-configured partitions.

10.7 LRAM and LDEV Description

When NSEs are cascaded using multiple CYNSE70128s, the SADR, CE_L, and WE_L (three-state signals) are all tied together. In order to eliminate external pull-up and pull downs, one device in a bank is designated as the default driver. For non-Search or non-Learn cycles (see “Learn Command” on page 98) or search cycles with a global miss, the SADR, CE_L, and WE_L signals are driven by the device with the LRAM bit set. It is important that only one device in a bank of NSEs that are cascaded have this bit set. Failure to do so will cause contention on SADR, CE_L, WE_L and can potentially cause damage to the device(s).

Similarly, when NSEs using multiple CYNSE70128s are cascaded, SSF and SSV (also three-state signals) are tied together. In order to eliminate external pull-up and pull downs, one device in a bank is designated as the default driver. For nonsearch cycles or Search cycles with a global miss the SSF and SSV signals are driven by the device with the LDEV bit set. It is important that only one device in a bank of NSEs that are cascaded together have this bit set. Failure to do so will cause contention on SSV and SSF and can potentially cause damage to the device(s).

10.8 Learn Command

Bit[0] of each 72-bit data location specifies whether an entry in the database is occupied. If all the entries in a device are occupied, the device asserts FULO signal to inform the downstream devices that it is full. The result of this communication between depth-cascaded devices determines the global FULL signal for the entire table. The FULL signal in the last device determines the fullness of the depth-cascaded table.

The device contains 16 pairs of internal, 72-bit-wide comparand registers that store the comparands as the device executes searches. On a miss by the Search signalled to ASIC through the SSV and SSF signals (SSV = 1, SSF = 0), the host ASIC can apply the Learn command to learn the entry from a comparand register to the next-free location (see “NFA Register” on page 19). The NFA updates to the next-free location following each Write or Learn command.

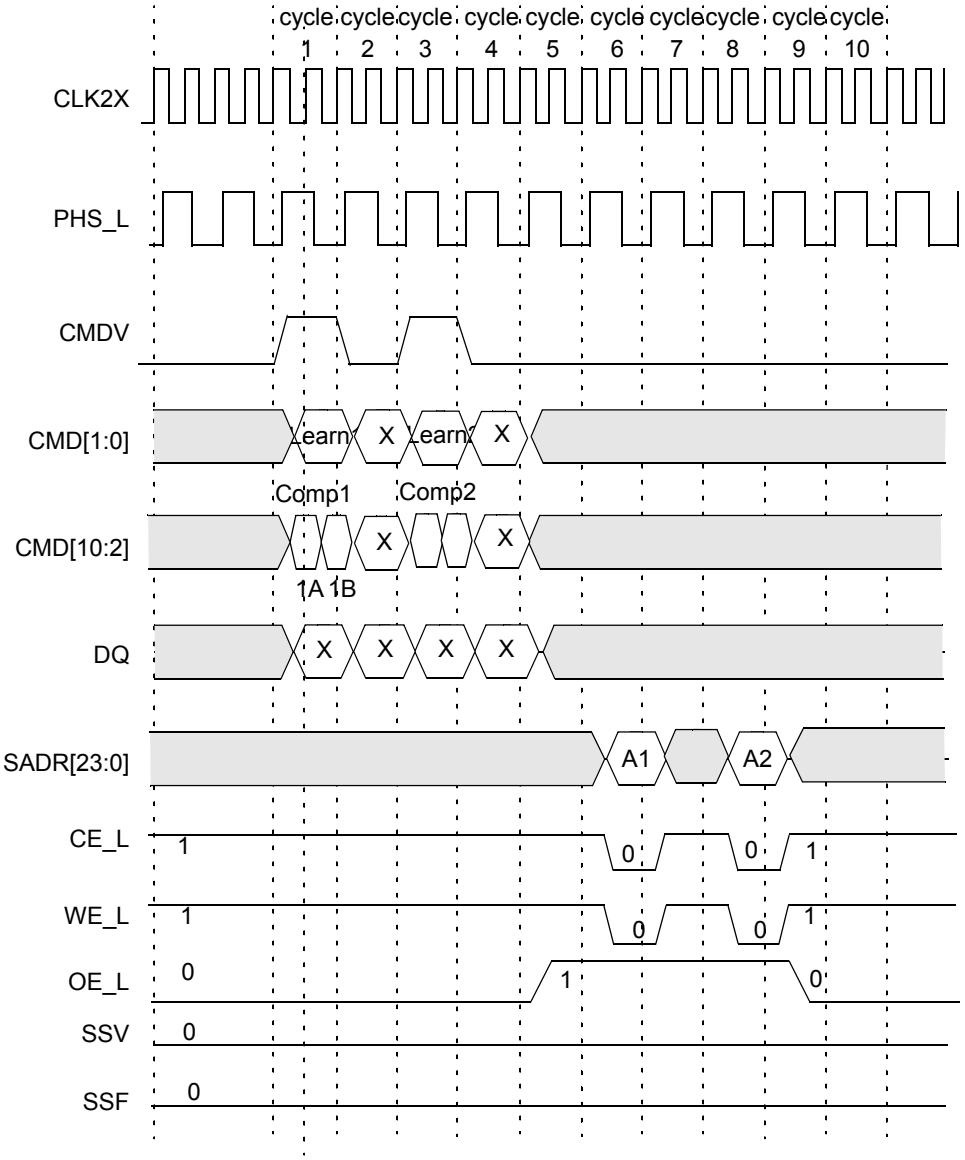
In a depth-cascaded table, only a single device will learn the entry through the application of a Learn instruction. The determination of which device is going to learn is based on the FULI and FULO signalling between the devices. The first non-full device learns the entry by storing the contents of the specified comparand registers to the location(s) pointed to by NFA.

In a ×72-configured table the Learn command writes a single 72-bit location. In a ×144-configured table the Learn command writes the next even and odd 72-bit locations. In 144-bit mode, bit[0] of the even and odd 72-bit locations is 0, which indicates that they are cascaded empty, or 1, which indicates that they are occupied.

The global FULL signal indicates to the table controller (the host ASIC) that all entries within a block are occupied and that no more entries can be learned. The CYNSE70128 updates the signal after each Write or Learn command to a data array. The Learn command generates a Write cycle to the external SRAM, also using the NFA register as part of the SRAM address (see ”SRAM Addressing” on page 105).

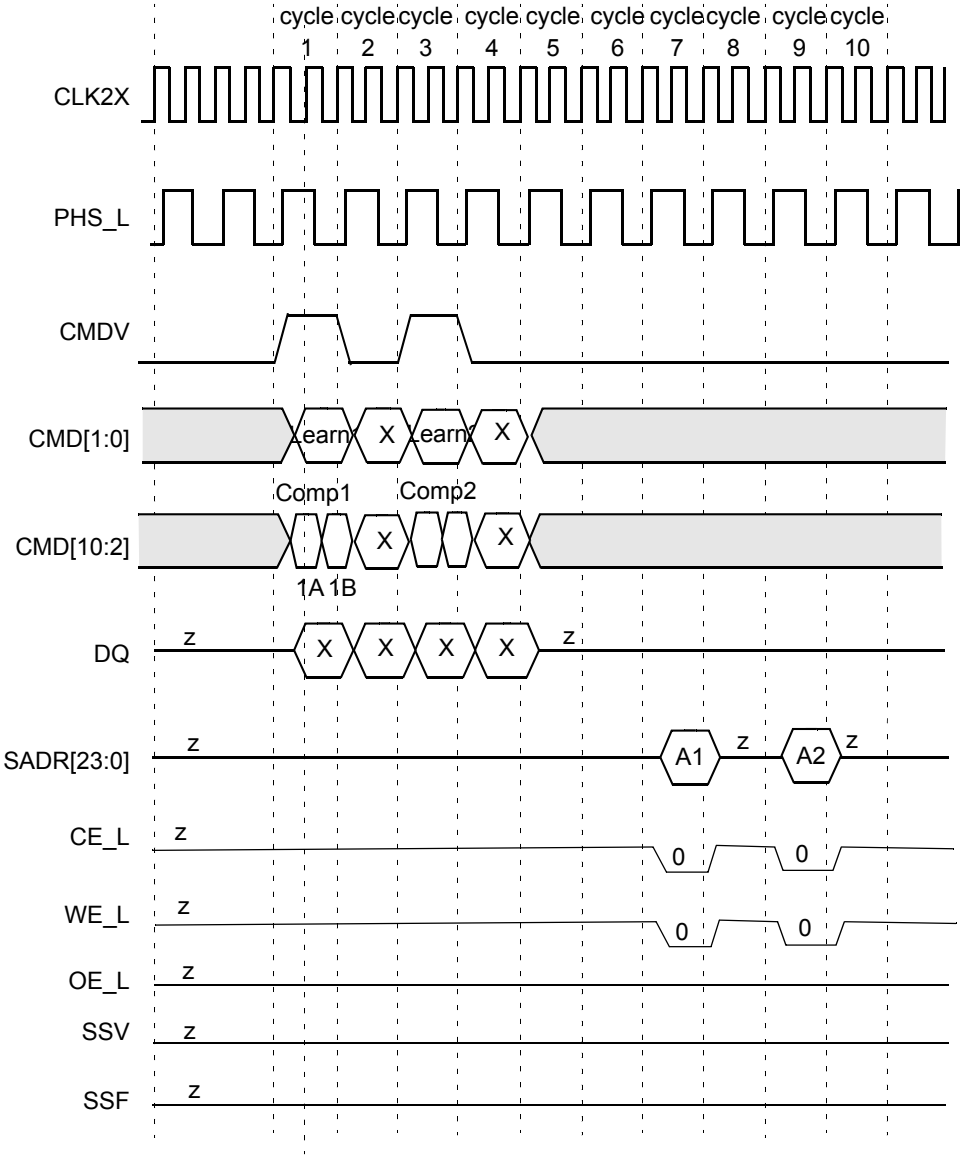
The Learn command is supported on a single block containing up to eight devices if the table is configured either as a ×72 or a ×144. The Learn command is not supported for x288-configured tables. Additionally, Learn is not supported when the device is operating at > 83-MHz CLK1X (166-MHz CLK2X).

Learn is a pipelined operation and lasts for two CLK cycles, as shown in *Figure 10-73* where TLSZ = 00, and *Figure 10-74* and *Figure 10-75* where TLSZ = 01. *Figure 10-74* and *Figure 10-75* assume that the device performing the Learn operation is not the last device in the table and has its LRAM bit set to 0. ***Note***. The OE_L for the device with the LRAM bit set goes high for two cycles for each Learn (one during the SRAM Write cycle, and one the cycle before). The latency of the SRAM Write cycle from the second cycle of the instruction is shown in *Table 10-35*.



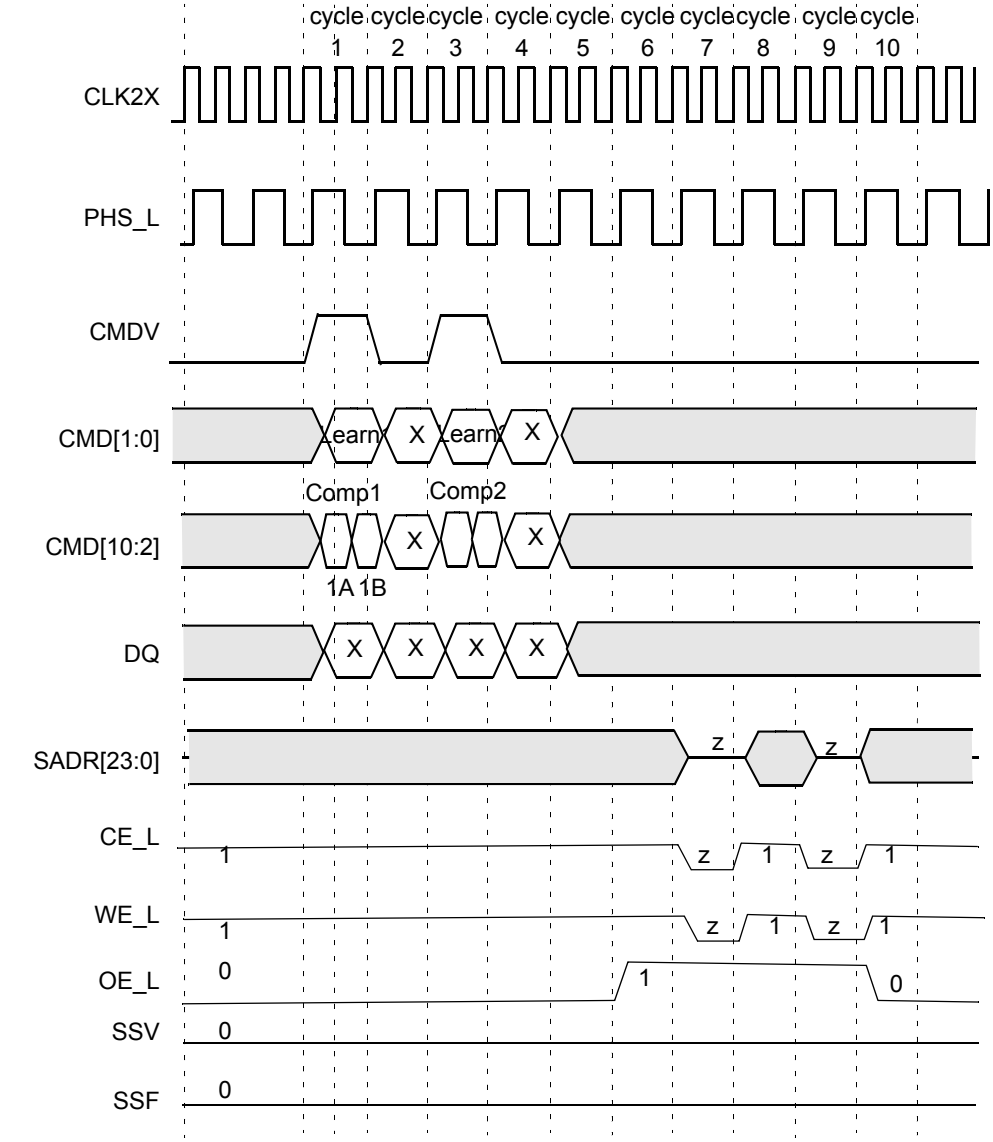
TLSZ = 00, LRAM = 1, LDEV = 1.

Figure 10-73. Timing Diagram of Learn (TLSZ = 00)



TLSZ = 01, LRAM = 0, LDEV = 0.

Figure 10-74. Timing Diagram of Learn (Except on the Last Device [TLSZ = 01])



TLSZ = 01, LRAM = 1, LDEV = 1.

Figure 10-75. Timing Diagram of Learn on Device Number 7 (TLSZ = 01)

Table 10-35. Latency of SRAM Write Cycle from Second Cycle of Learn Instruction

Number of Devices	Latency in CLK Cycles
1 (TLSZ = 00)	4
1–8 (TLSZ = 01)	5
1–31 (TLSZ = 10)	6

The Learn operation lasts two CLK cycles. The sequence of operation is as follows.

- **Cycle 1A**: The host ASIC applies the Learn instruction on CMD[1:0] using CMDV = 1. The CMD[5:2] field specifies the index of the comparand register pair that will be written in the data array in the 144-bit-configured table. For a Learn in a 72-bit-configured table, the even-numbered comparand specified by this index will be written. CMD[8:6] carries the bits that will be driven on SADR[23:21] in the SRAM Write cycle.
- **Cycle 1B**: The host ASIC continues to drive the CMDV to 1, the CMD[1:0] to 11, and the CMD[5:2] with the comparand pair index. CMD[6] must be set to 0 if the Learn is being performed on a 72-bit-configured table, and to 1 if the Learn is being performed on a 144-bit-configured table.

- **Cycle 2**: The host ASIC drives the CMDV to 0.

At the end of cycle 2, a new instruction can begin. The latency of the SRAM Write is the same as the search to the SRAM Read cycle. It is measured from the second cycle of the Learn instruction.

11.0 Depth-Cascading

The NSE application can depth-cascade the devices to various table sizes of different widths (72 bits, 144 bits, or 288 bits). The devices perform all the necessary arbitration to decide which device will drive the SRAM bus. The latency of the searches increases as the table size increases; the Search rate remains constant.

11.1 Depth-Cascading up to Eight Devices (One Block)

Figure 11-1 shows how up to eight devices can be cascaded to form 512K × 72, 256K × 144, or 128K × 288 tables. It also shows the interconnection between the devices for depth-cascading. Each NSE asserts the LHO[1] and LHO[0] signals to inform downstream devices of its result. The LHI[6:0] signals for a device are connected to LHO signals of the upstream devices. The host ASIC must program the TLSZ to 01 for each of up to eight devices in a block. Only a single device drives the SRAM bus in any single cycle.

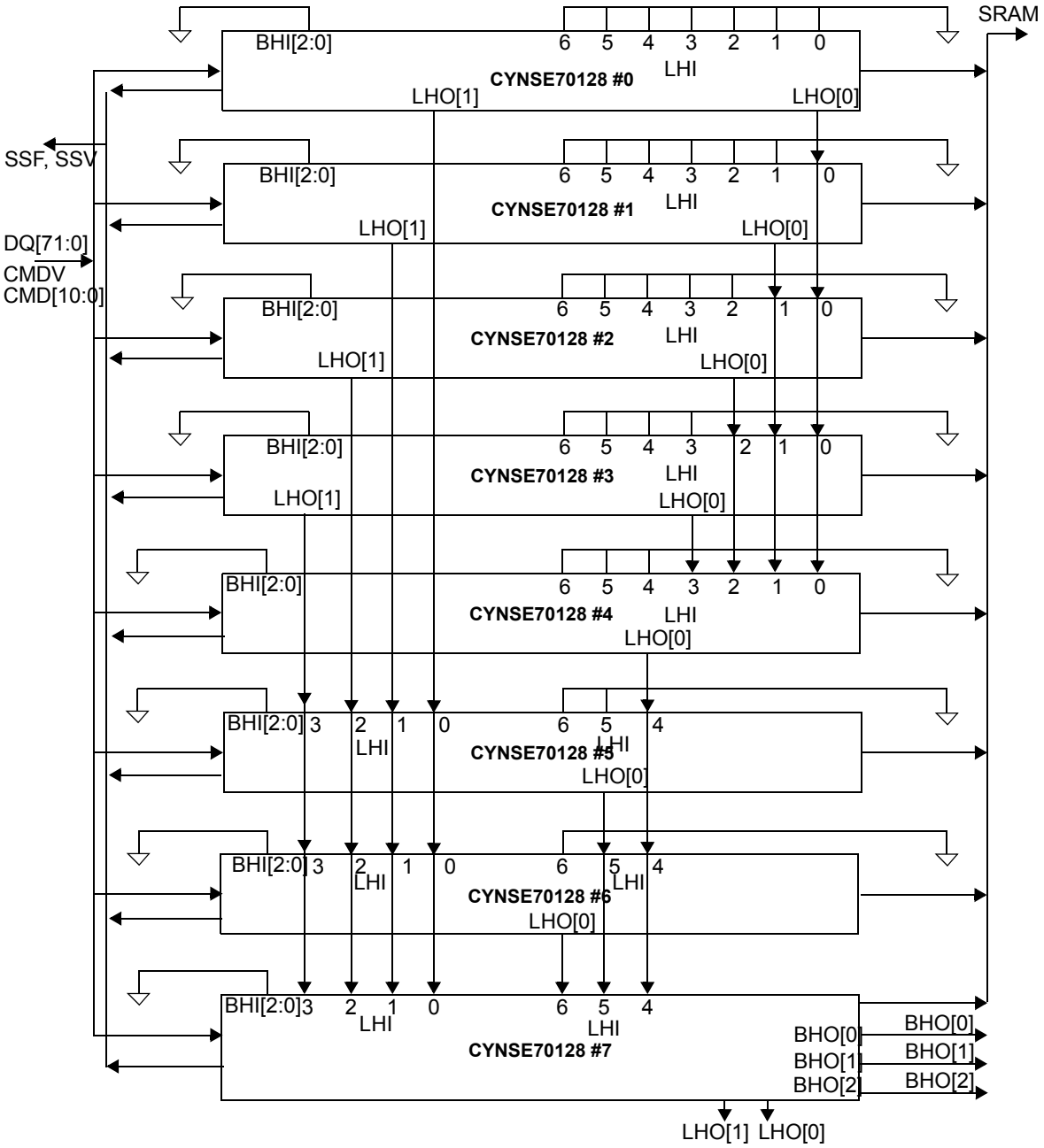


Figure 11-1. Depth-Cascading to Form a Single Block

11.2 Depth-Cascading up to 31 Devices (Four Blocks)

Figure 11-2 shows how to cascade up to four blocks. Each block contains up to eight CYNSE70128 devices except the last, and the interconnection within each was shown in the previous subsection with the cascading of up to eight devices in a block. ***Note***. The interconnection between blocks for depth-cascading is important. For each Search, a block asserts BHO[2], BHO[1], and BHO[0]. The BHO[2:0] signals for a block are the signals taken only from the last device in the block. For all other devices within that block, these signals stay open and floating. The host ASIC must program the table size (TLSZ) field to 10 in each of the devices for cascading up to 31 devices (in up to four blocks).

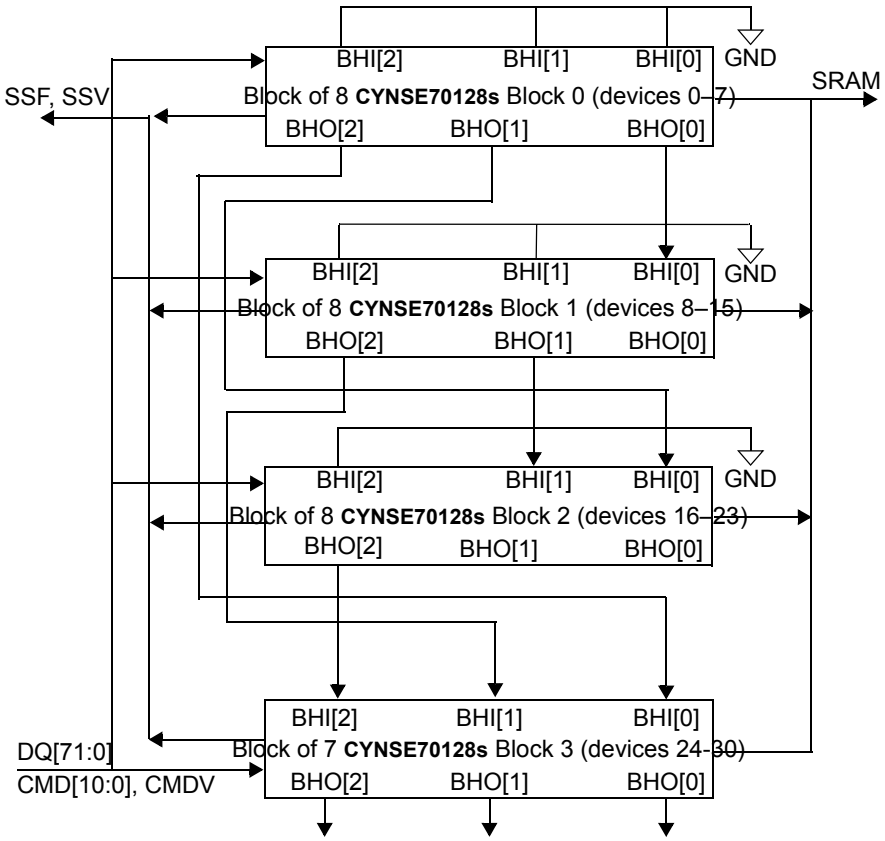


Figure 11-2. Depth-Cascading Four Blocks

11.3 Depth-Cascading for a FULL Signal

Bit[0] of each of the 72-bit entries is designated as a special bit (1 = occupied; 0 = empty). For each Learn or PIO Write to the data array, each device asserts FULO[1] and FULO[0] if it does not have any empty locations within it (see *Figure 11-3*). Each device combines the FULO signals from the devices above it with its own full status to generate a FULL signal that gives the full status of the table up to the device asserting the FULL signal. *Figure 11-3* shows the hardware connection diagram for generating the FULL signal that goes back to the ASIC. In a depth-cascaded block of up to eight devices, the FULL signal from the last device should be fed back to the ASIC controller to indicate the fullness of the table. The FULL signal of the other devices should be left open. ***Note***. The Learn instruction is supported for only up to eight devices, whereas FULL cascading is allowed only for one block in tables containing more than eight devices. In tables for which a Learn instruction is not going to be used, the bit[0] of each 72-bit entry should always be set to 1.

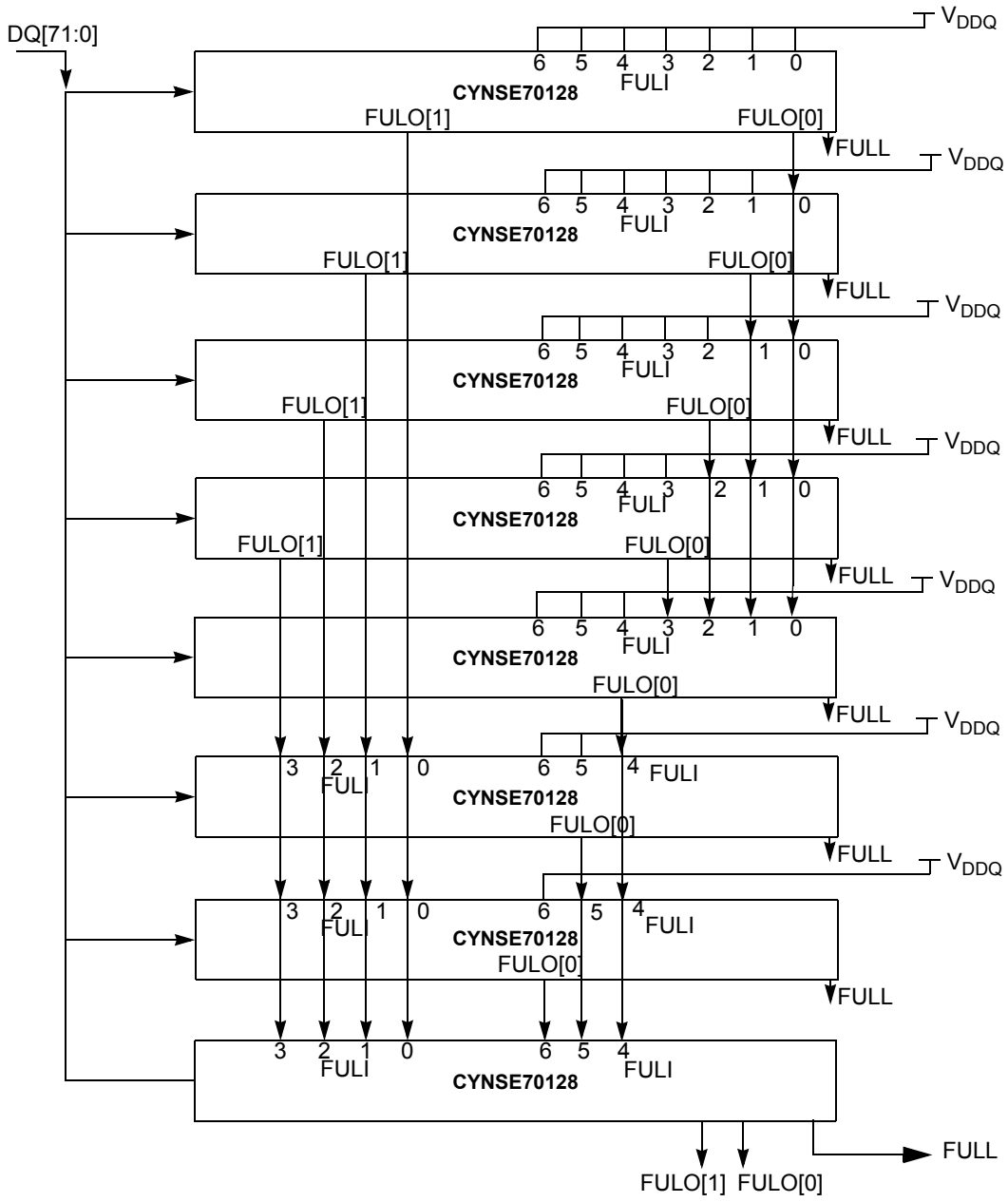


Figure 11-3. FULL Generation in a Cascaded Table

12.0 SRAM Addressing

Table 12-1 describes the commands used to generate addresses on the SRAM address bus. The index [15:0] field contains the address of a 72-bit entry that results in a hit in 72-bit-configured quadrant. It is the address of the 72-bit entry that lies at the 144-bit page, and the 288-bit page boundaries in 144-bit- and 288-bit-configured quadrants, respectively.

”Registers” on page 15 of this specification, describes the NFA and SSR registers. ADR[15:0] contains the address supplied on the DQ bus during PIO access to the CYNSE70128. Command bits 8, 7, and 6 {CMD[8:6]} are passed from the command to the SRAM address bus. See ”Commands” on page 21, for more information. ID[4:0] is the ID of the device driving the SRAM bus (see ”Pinout Description” on page 130 for more information).

12.1 Generating an SRAM BUS Address

Table 12-1. SRAM Address

Command	SRAM Operation	23	22	21	[20:16]	[15:0]
Search	Read	C8	C7	C6	ID[4:0]	Index[15:0]
Learn	Write	C8	C7	C6	ID[4:0]	NFA[15:0]
PIO Read	Read	C8	C7	C6	ID[4:0]	ADR15:0]
PIO Write	Write	C8	C7	C6	ID[4:0]	ADR[15:0]
Indirect Access	Write/Read	C8	C7	C6	ID[4:0]	SSR[15:0]

12.2 SRAM PIO Access

The remainder of this section describes SRAM Read and SRAM Write operations.

SRAM Read enables Read access to the off-chip SRAM containing associative data. The latency from the issuance of the Read instruction to the address appearing on the SRAM bus is the same as the latency of the Search instruction and will be depend on the value programmed for the TLSZ parameter in the device configuration register. The latency of the ACK from the Read instruction is the same as the latency of the Search instruction to the SRAM address plus the HLAT programmed in the configuration register. ***Note***. SRAM Read is a blocking operation—no new instruction can begin until the ACK is returned by the selected device performing the access.

SRAM Write enables Write access to the off-chip SRAM containing associative data. The latency from the second cycle of the Write instruction to the address appearing on the SRAM bus is the same as the latency of the Search instruction and will depend on the TLSZ value parameter programmed in the device configuration register. ***Note***. SRAM Write is a pipelined operation—new instruction can begin right after the previous command has ended.

12.3 SRAM Read with a Table of One Device

SRAM Read enables Read access to the off-chip SRAM containing associative data. The latency from the issuance of the Read instruction to the address appearing on the SRAM bus is the same as the latency of the Search instruction and will depend on the TLSZ value parameter programmed in the device configuration register. The latency of the ACK from the Read instruction is the same as the latency of the Search instruction to the SRAM address plus the HLAT programmed in the configuration register. The following explains the SRAM Read operation in a table with only one device that has the following parameters: TLSZ = 00, HLAT = 000, LRAM = 1, and LDEV = 1. *Figure 12-1* shows the associated timing diagram. For the following description, the selected device refers to the only device in the table because it is the only device to be accessed.

- **Cycle 1A**: The host ASIC applies the Read instruction on the CMD[1:0] using CMDV = 1. The DQ bus supplies the address, with DQ[20:19] set to 10, to select the SRAM address. The host ASIC selects the device for which the ID[4:0] matches the DQ[25:21] lines. During this cycle, the host ASIC also supplies SADR[23:21] on CMD[8:6].
- **Cycle 1B**: The host ASIC continues to apply the Read instruction on the CMD[1:0] using CMDV = 1. The DQ bus supplies the address with DQ[20:19] set to 10 to select the SRAM address.
- **Cycle 2**: The host ASIC floats DQ[71:0] to a three-state condition.
- **Cycle 3**: The host ASIC keeps DQ[71:0] in a three-state condition.
- **Cycle 4**: The selected device starts to drive DQ[71:0] and drives ACK from high-Z to low.
- **Cycle 5**: The selected device drives the Read address on SADR[23:0]; it also drives ACK high, CE_L low, and ALE_L low.
- **Cycle 6**: The selected device drives CE_L high, ALE_L high, the SADR bus, the DQ bus in a three-state condition, and ACK low.

At the end of cycle 6, the selected device floats ACK in a three-state condition, and a new command can begin.

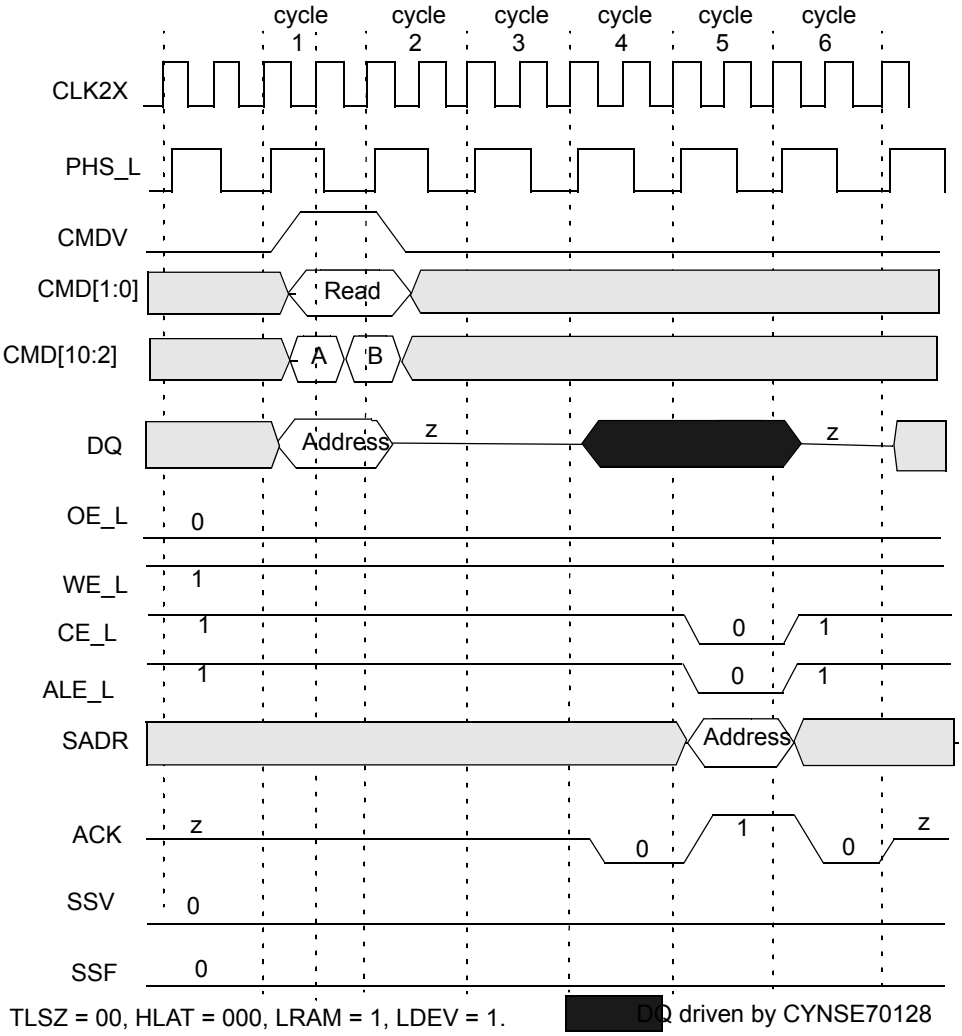


Figure 12-1. SRAM Read Access (TLSZ = 00, HLAT = 000, LRAM = 1, LDEV = 1)

12.4 SRAM Read with a Table of up to Eight Devices

The following explains the SRAM Read operation completed through a table of up to eight devices using the following parameters: TLSZ = 01. *Figure 12-2* diagrams a block of eight devices. The following assumes that SRAM access is successfully achieved through CYNSE70128 device number 0. *Figure 12-3* and *Figure 12-4* show timing diagrams for device number 0 and device number 7, respectively.

- **Cycle 1A**: The host ASIC applies the Read instruction on CMD[1:0] using CMDV = 1. The DQ bus supplies the address, with DQ[20:19] set to 10, to select the SRAM address. The host ASIC selects the device for which ID[4:0] matches the DQ[25:21] lines. During this cycle the host ASIC also supplies SADR[23:21] on CMD[8:6].
- **Cycle 1B**: The host ASIC continues to apply the Read instruction on CMD[1:0] using CMDV = 1. The DQ bus supplies the address, with DQ[20:19] set to 10 to select the SRAM address.
- **Cycle 2**: The host ASIC floats DQ[71:0] to a three-state condition.
- **Cycle 3**: The host ASIC keeps DQ[71:0] in a three-state condition.
- **Cycle 4**: The selected device starts to drive DQ[71:0].
- **Cycle 5**: The selected device continues to drive DQ[71:0] and drives ACK from high-Z to low.
- **Cycle 6**: The selected device drives the Read address on SADR[23:0]. It also drives ACK high, CE_L low, WE_L high, and ALE_L low.
- **Cycle 7**: The selected device drives CE_L, ALE_L, WE_L, and DQ bus in a three-state condition. It continues to drive ACK low.

At the end of cycle 7, the selected device floats ACK in three-state condition and a new command can begin.

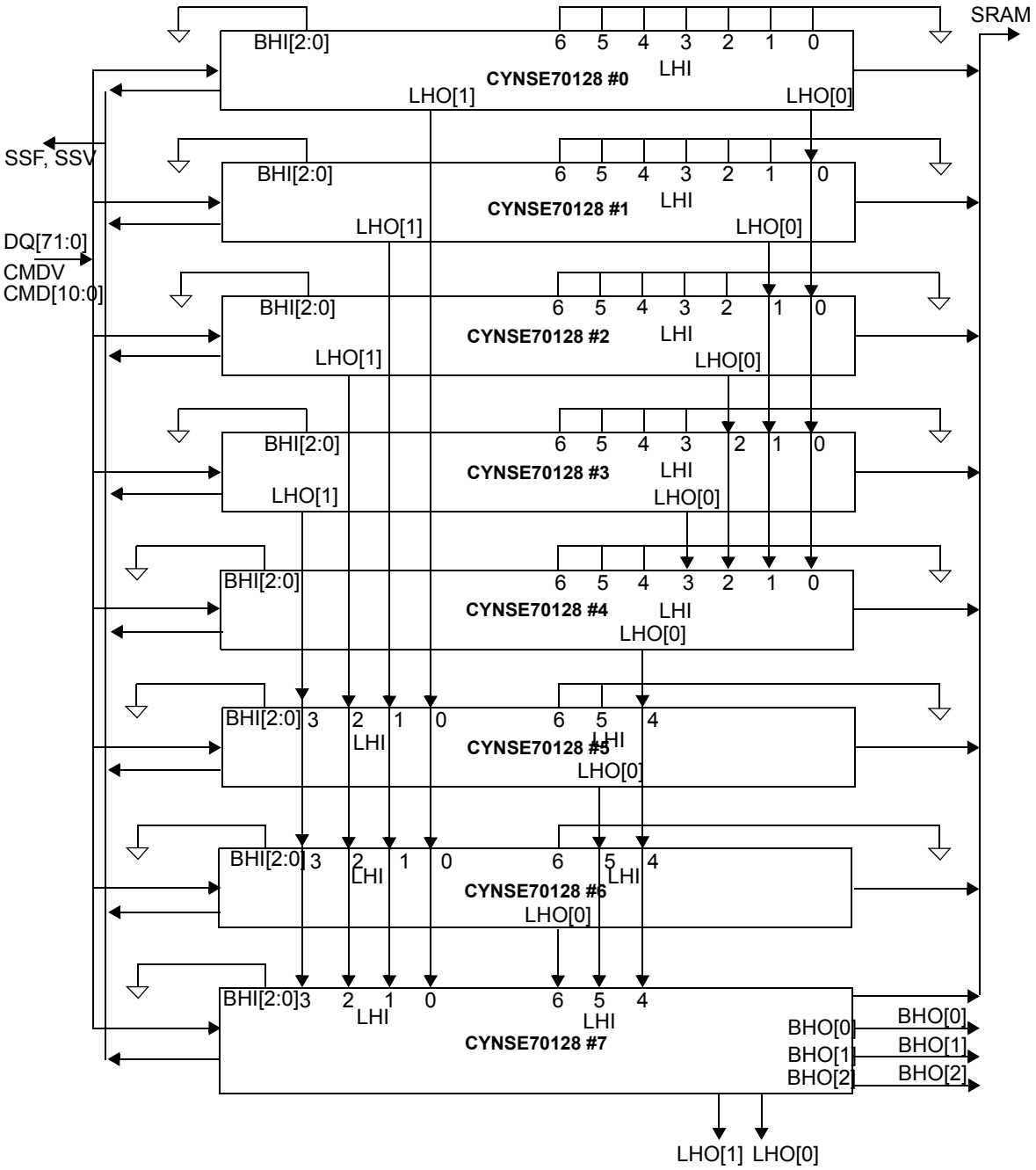


Figure 12-2. Table of a Block of Eight Devices

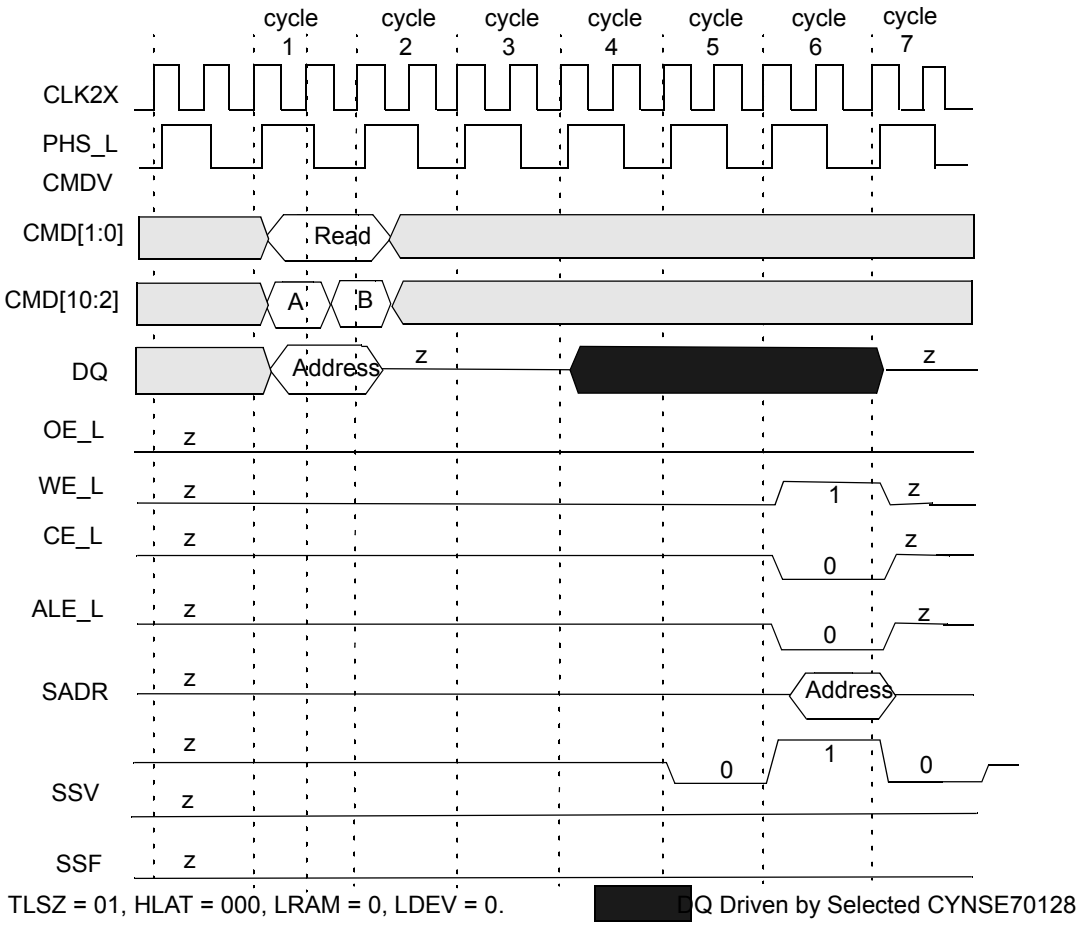
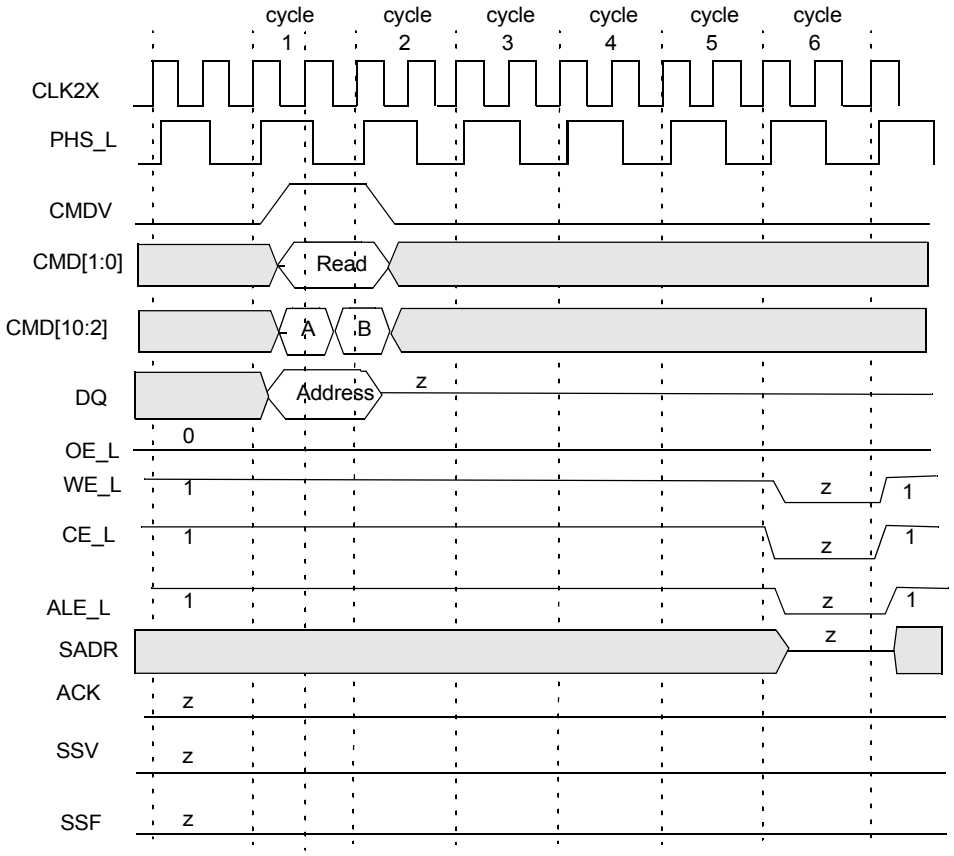


Figure 12-3. SRAM Read Through Device Number 0 in a Block of Eight Devices



TLSZ = 01, HLAT = 000, LRAM = 1, LDEV = 1.

Figure 12-4. SRAM Read Timing for Device Number 7 in a Block of Eight Devices

12.5 SRAM Read with a Table of up to 31 Devices

The following explains the SRAM Read operation accomplished through a table of up to 31 devices, using the following parameters: TLSZ = 10. The diagram of such a table is shown in *Figure 12-5*. The following assumes that SRAM access is being accomplished through CYNSE70128 device number 0, that device number 0 is the selected device. *Figure 12-6* and *Figure 12-7* show the timing diagrams for device number 0 and device number 30, respectively.

- **Cycle 1A**: The host ASIC applies the Read instruction to CMD[1:0] using CMDV = 1. The DQ bus supplies the address, with DQ[20:19] set to 10, to select the SRAM address. The host ASIC selects the device for which the ID[4:0] matches the DQ[25:21] lines. During this cycle, the host ASIC also supplies SADR[23:21] on CMD[8:6].
- **Cycle 1B**: The host ASIC continues to apply the Read instruction to CMD[1:0] using CMDV = 1. The DQ bus supplies the address, with DQ[20:19] set to 10, to select the SRAM address.
- **Cycle 2**: The host ASIC floats DQ[71:0] to a three-state condition.
- **Cycle 3**: The host ASIC keeps DQ[71:0] in a three-state condition.
- **Cycle 4**: The selected device starts to drive DQ[71:0].
- **Cycles 5 to 6**: The selected device continues to drive DQ[71:0].
- **Cycle 7**: The selected device continues to drive DQ[71:0] and drives an SRAM Read cycle.
- **Cycle 8**: The selected device drives ACK from Z to low.
- **Cycle 9**: The selected device drives ACK to high.
- **Cycle 10**: The selected device drives ACK from high to low.

At the end of cycle 10, the selected device floats ACK in a three-state condition.

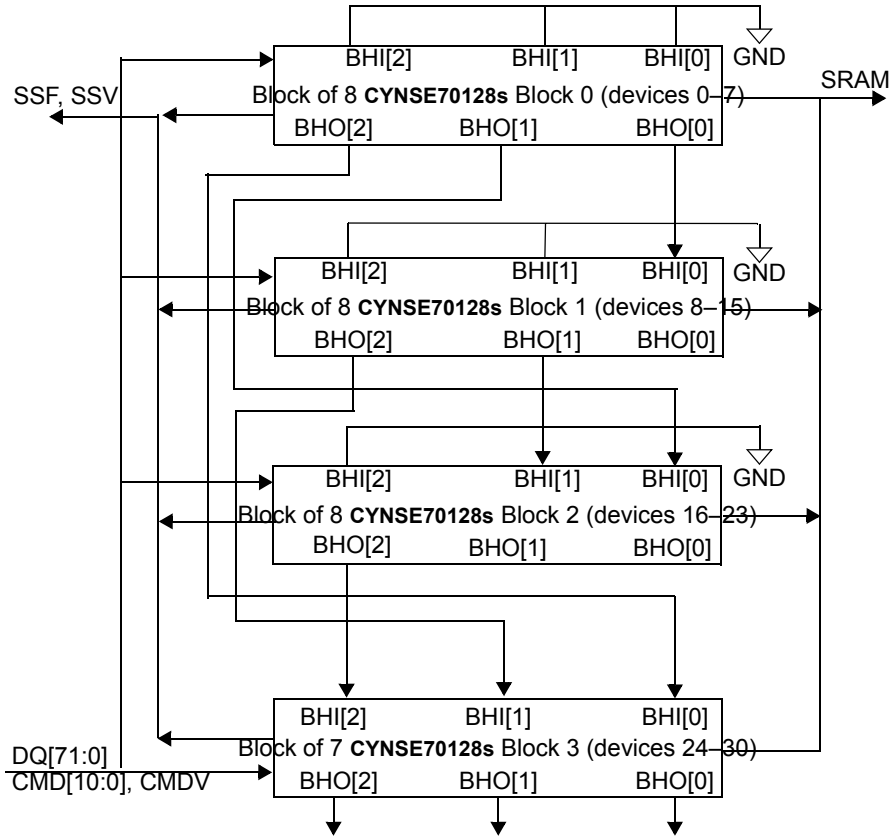


Figure 12-5. Table of 31 Devices Made of Four Blocks

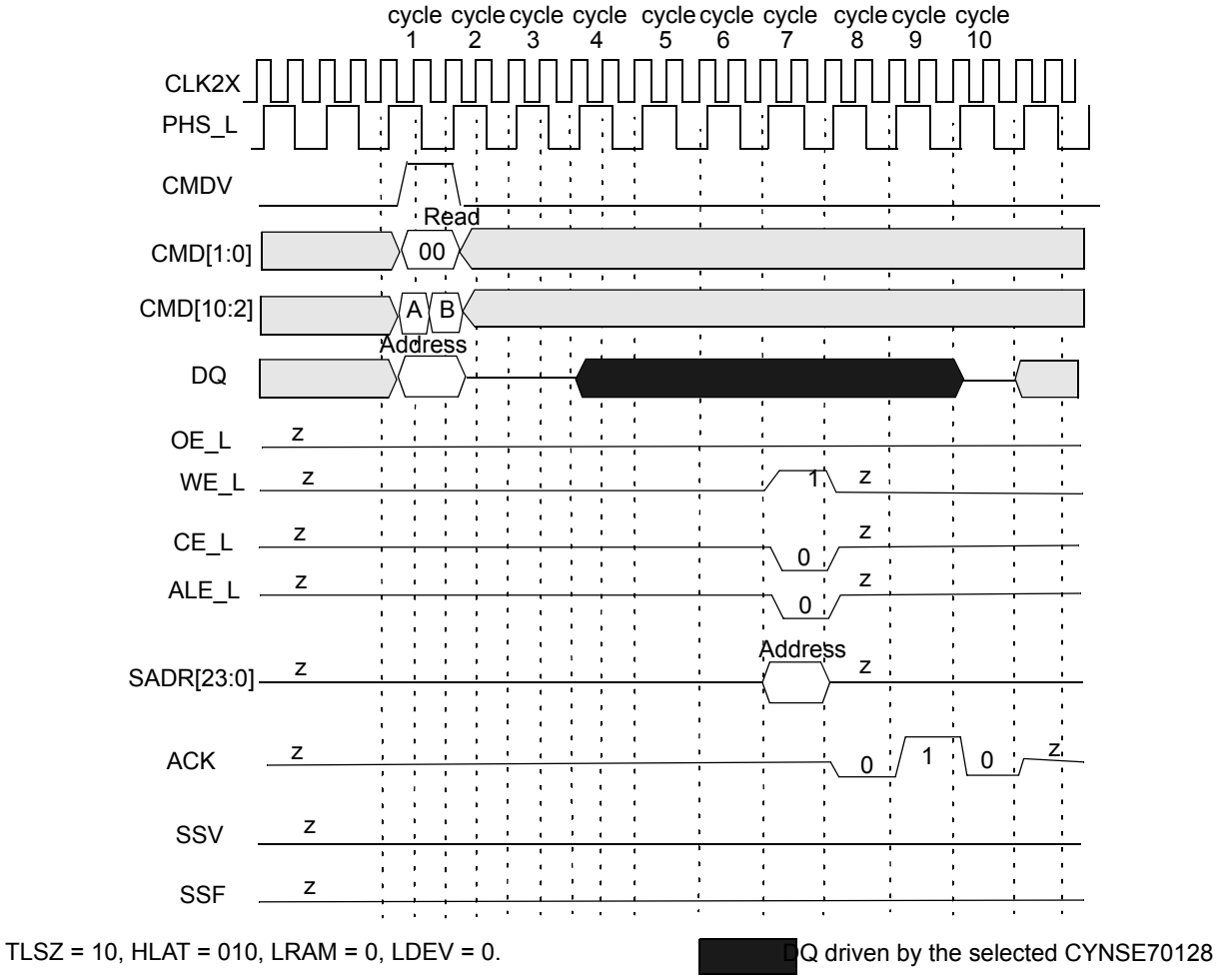
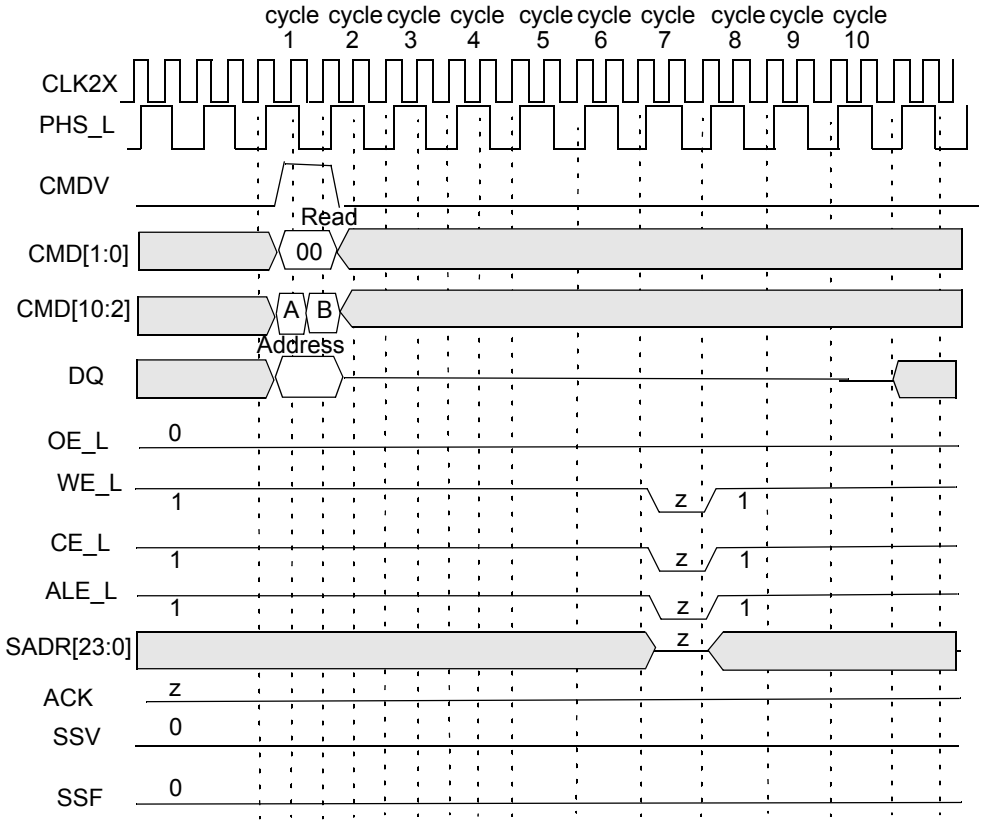


Figure 12-6. SRAM Read Through Device Number 0 in a Block of 31 Devices (Device Number 0 Timing)



TLSZ = 10, HLAT = 010, LRAM = 1, LDEV = 1.

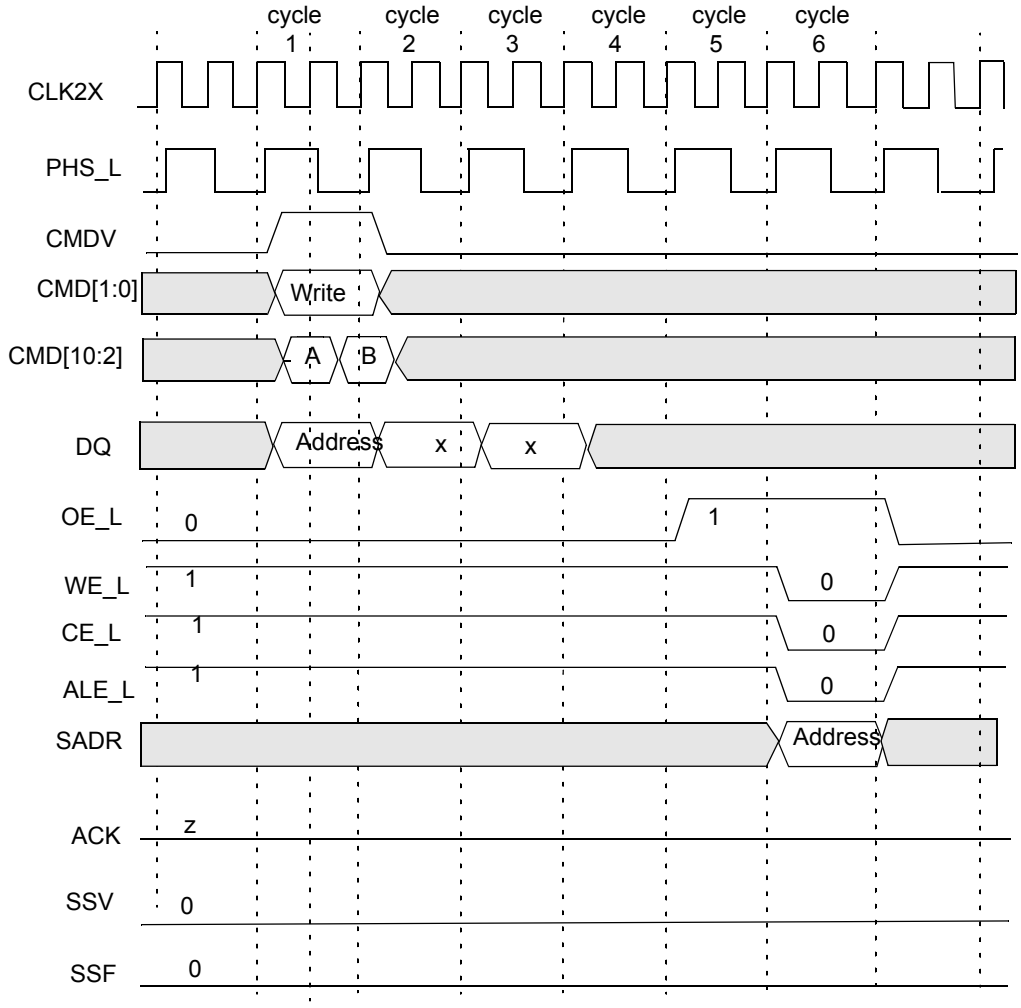
Figure 12-7. SRAM Read Through Device Number 0 in a Block of 31 Devices (Device Number 30 Timing)

12.6 SRAM Write with a Table of One Device

SRAM Write enables Write access to the off-chip SRAM that contains associative data. The latency from the second cycle of the Write instruction to the address appearing on the SRAM bus is the same as the latency of the Search instruction, and will depend on the TLSZ value parameter programmed in the device configuration register. The following explains the SRAM Write operation accomplished with a table of only one device of the following parameters: TLSZ = 00, HLAT = 000, LRAM = 1, and LDEV = 1. *Figure 12-8* shows the timing diagram. For the following description the selected device refers to the only device in the table as it is the only device that will be accessed.

- **Cycle 1A**: The host ASIC applies the Write instruction on CMD[1:0] using CMDV = 1. The DQ bus supplies the address with DQ[20:19] set to 10 to select the SRAM address. The host ASIC selects the device for which the ID[4:0] matches the DQ[25:21] lines. The host ASIC also supplies SADR[23:21] on CMD[8:6] in this cycle. ***Note***. CMD[2] must be set to 0 for SRAM Write because burst Writes into the SRAM are not supported.
- **Cycle 1B**: The host ASIC continues to apply the Write instruction on CMD[1:0], using CMDV = 1. The DQ bus supplies the address with DQ[20:19] set to 10 to select the SRAM address. ***Note***. CMD[2] must be set to 0 for SRAM Write because burst Writes into the SRAM are not supported.
- **Cycle 2**: The host ASIC continues to drive DQ[71:0]. The data in this cycle is not used by the CYNSE70128 device.
- **Cycle 3**: The host ASIC continues to drive DQ[71:0]. The data in this cycle is not used by the CYNSE70128 device.

At the end of cycle 3, a new command can begin. The Write is a pipelined operation. The Write cycle appears at the SRAM bus, however, with the same latency as that of a Search instruction, as measured from the second cycle of the Write command.



TLSZ = 00, HLAT = 000, LRAM = 1, LDEV = 1.

Figure 12-8. SRAM Write Access (TLSZ = 00, HLAT = 000, LRAM = 1, LDEV = 1)

12.7 SRAM Write with a Table of up to Eight Devices

The following explains the SRAM Write operation done through a table(s) of up to eight devices with the following parameters (TLSZ = 01). The diagram of such a table is shown in *Figure 12-9*. The following assumes that SRAM access is done through CYNSE70128 device number 0. *Figure 12-10* and *Figure 12-11* show the timing diagram for device number 0 and device number 7, respectively.

- **Cycle 1A**: The host ASIC applies the Write instruction on CMD[1:0] using CMDV = 1. The DQ bus supplies the address with DQ[20:19] set to 10 to select the SRAM address. The host ASIC selects the device for which the ID[4:0] matches the DQ[25:21] lines. The host ASIC also supplies SADR[23:21] on CMD[8:6] in this cycle. ***Note***. CMD[2] must be set to 0 for SRAM Write because burst Writes into the SRAM are not supported.
- **Cycle 1B**: The host ASIC continues to apply the Write instruction on CMD[1:0] using CMDV = 1. The DQ bus supplies the address with DQ[20:19] set to 10 to select the SRAM address. ***Note***. CMD[2] must be set to 0 for SRAM Write because burst Writes into the SRAM are not supported.
- **Cycle 2**: The host ASIC continues to drive DQ[71:0]. The data in this cycle is not used by the CYNSE70128 device.
- **Cycle 3**: The host ASIC continues to drive DQ[71:0]. The data in this cycle is not used by the CYNSE70128 device.

At the end of cycle 3, a new command can begin. The Write is a pipelined operation. The Write cycle appears at the SRAM bus, however, with the same latency as that of a Search instruction, as measured from the second cycle of the Write command.

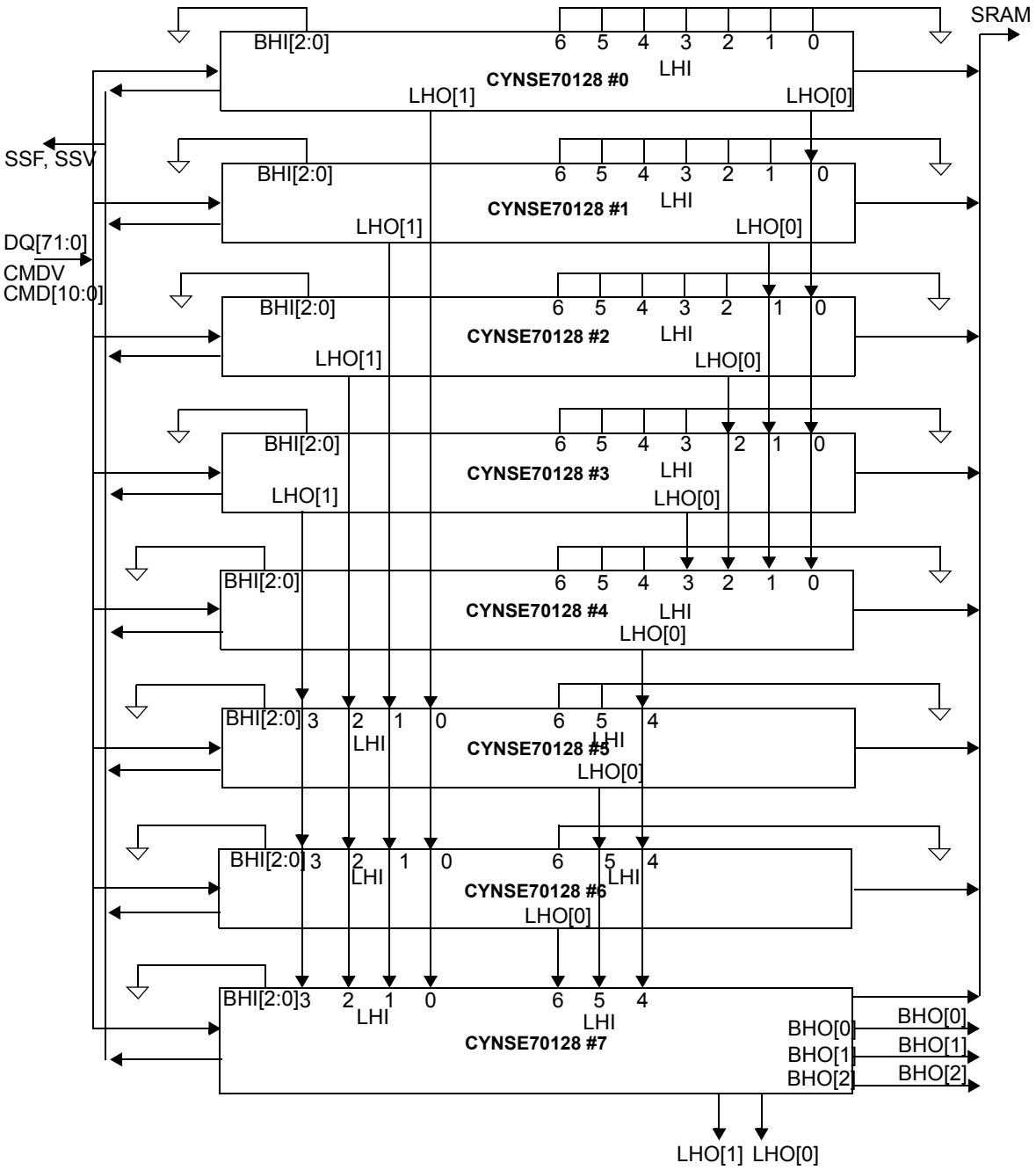
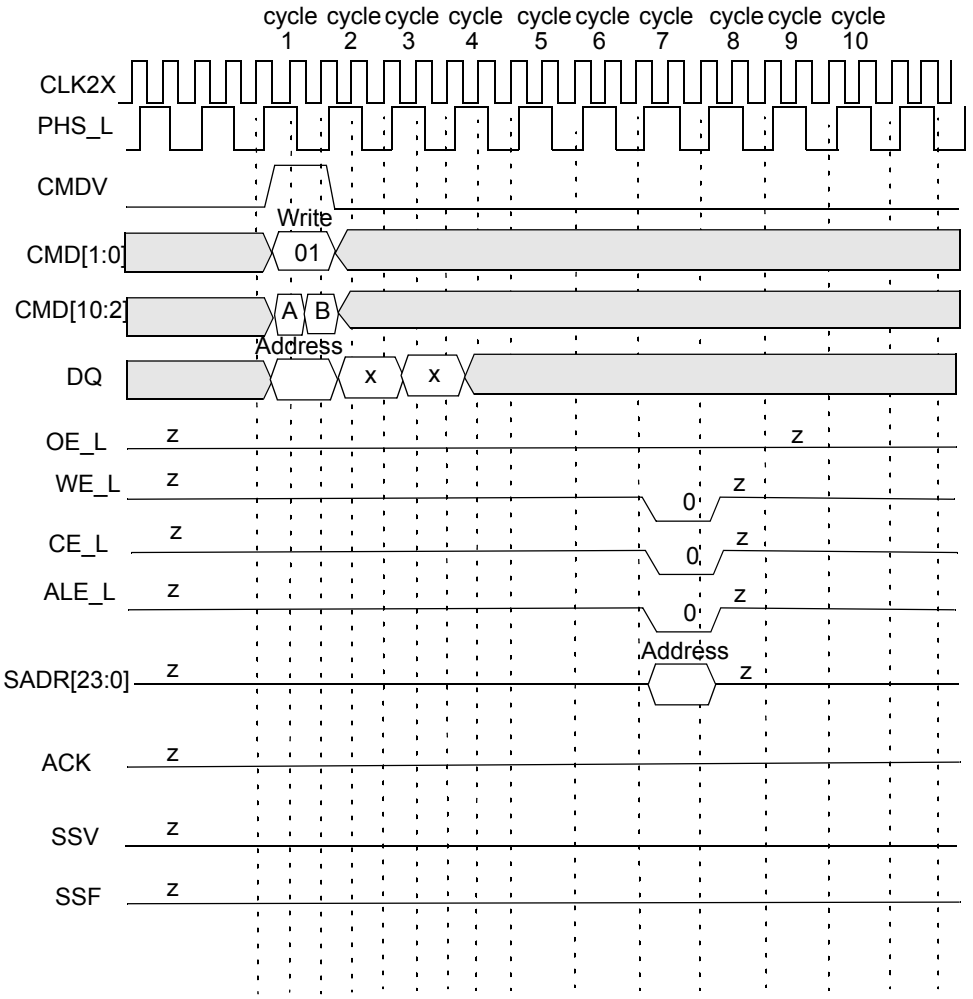
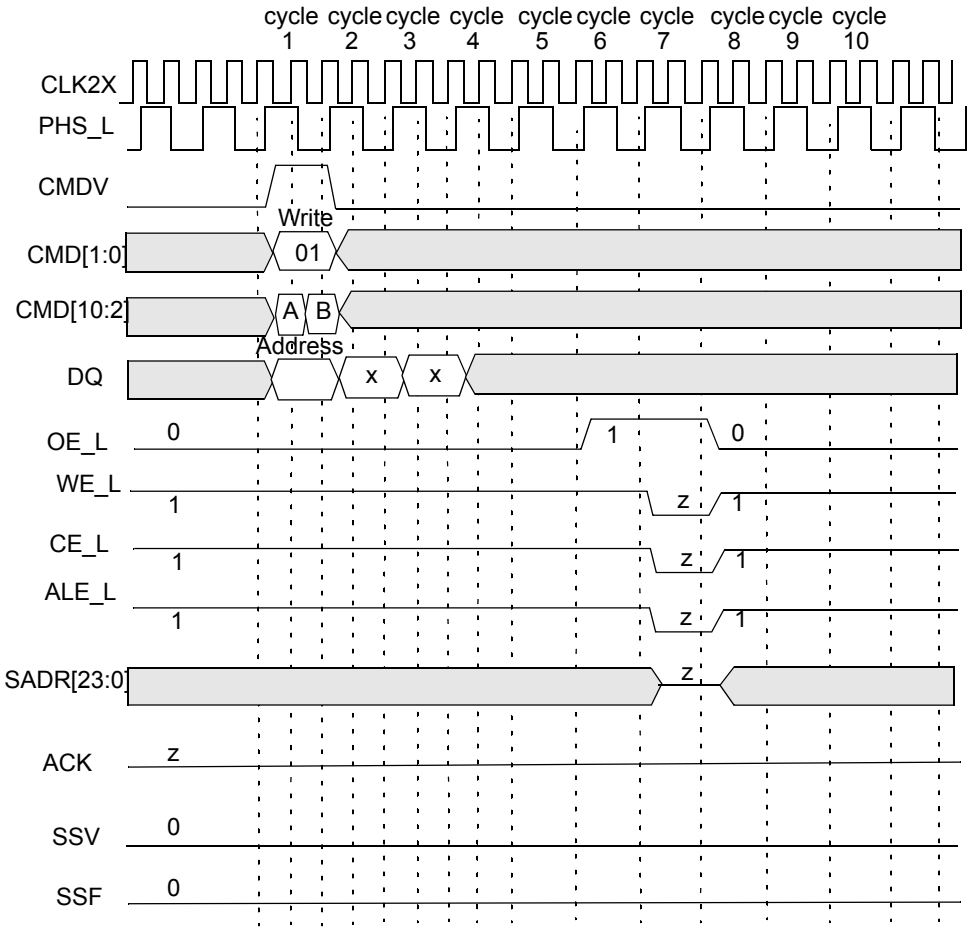


Figure 12-9. Table of a Block of Eight Devices



TLSZ = 01, HLAT = XXX, LRAM = 0, LDEV = 0.

Figure 12-10. SRAM Write Through Device Number 0 in a Block of Eight Devices



TLSZ = 01, HLAT = XXX, LRAM = 1, LDEV = 1.

Figure 12-11. SRAM Write Timing for Device Number 7 in a Block of Eight Devices

12.8 SRAM Write with Table(s) of up to 31 Devices

The following explains the SRAM Write operation done through a table(s) of up to 31 devices with the following parameters (TLSZ = 10). The diagram of such table(s) is shown in *Figure 12-12*. The following assumes that SRAM access is done through CYNSE70128 device number 0—device 0 is the selected device. *Figure 12-13* and *Figure 12-14* show the timing diagram for device number 0 and device number 30, respectively.

- **Cycle 1A**: The host ASIC applies the Write instruction on CMD[1:0] using CMDV = 1. The DQ bus supplies the address with DQ[20:19] set to 10 to select the SRAM address. The host ASIC selects the device for which the ID[4:0] matches the DQ[25:21] lines. The host ASIC also supplies SADR[23:21] on CMD[8:6] in this cycle. ***Note***. CMD[2] must be set to 0 for SRAM Write because burst Writes into the SRAM are not supported.
- **Cycle 1B**: The host ASIC continues to apply the Write instruction on CMD[1:0] using CMDV = 1. The DQ bus supplies the address with DQ[20:19] set to 10 to select the SRAM address. ***Note***. CMD[2] must be set to 0 for SRAM Write because burst Writes into the SRAM are not supported.
- **Cycle 2**: The host ASIC continues to drive DQ[71:0]. The data in this cycle is not used by the CYNSE70128 device.
- **Cycle 3**: The host ASIC continues to drive DQ[71:0]. The data in this cycle is not used by the CYNSE70128 device.

At the end of cycle 3, a new command can begin. The Write is a pipelined operation. The Write cycle appears at the SRAM bus, however, with the same latency as that of a Search instruction, as measured from the second cycle of the Write command.

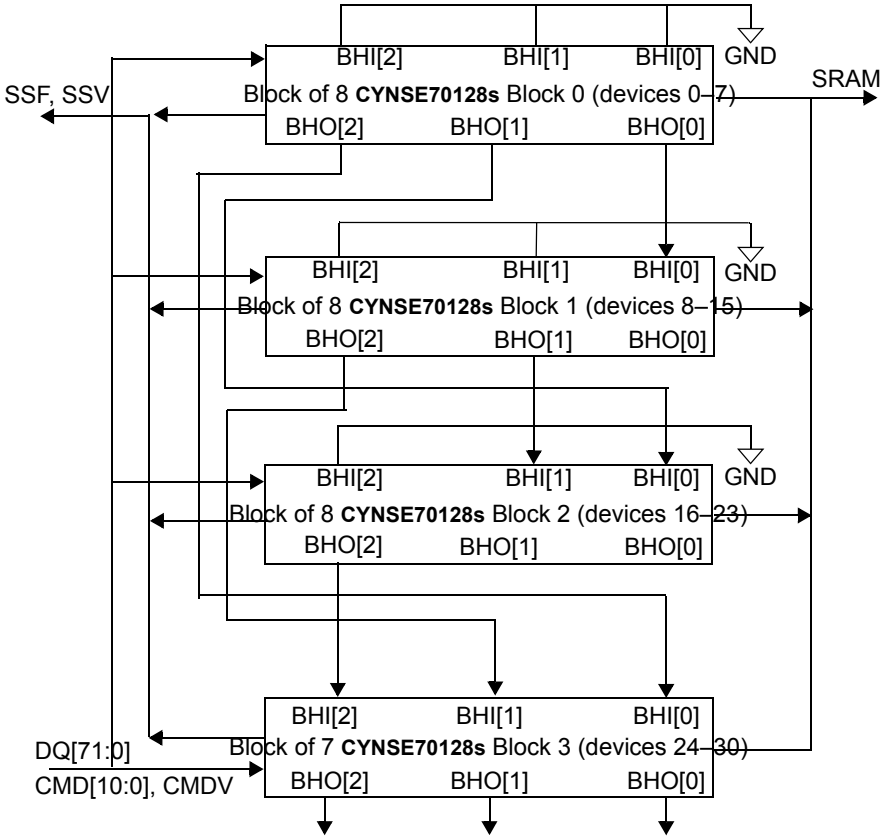
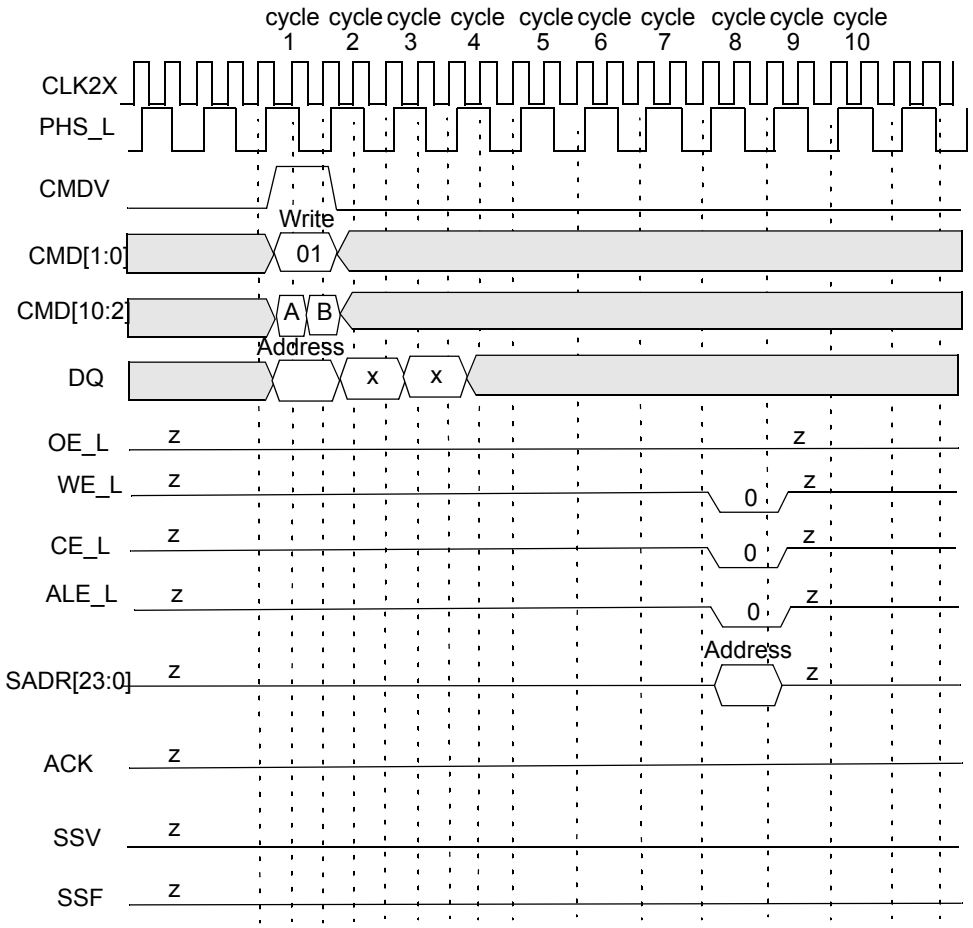
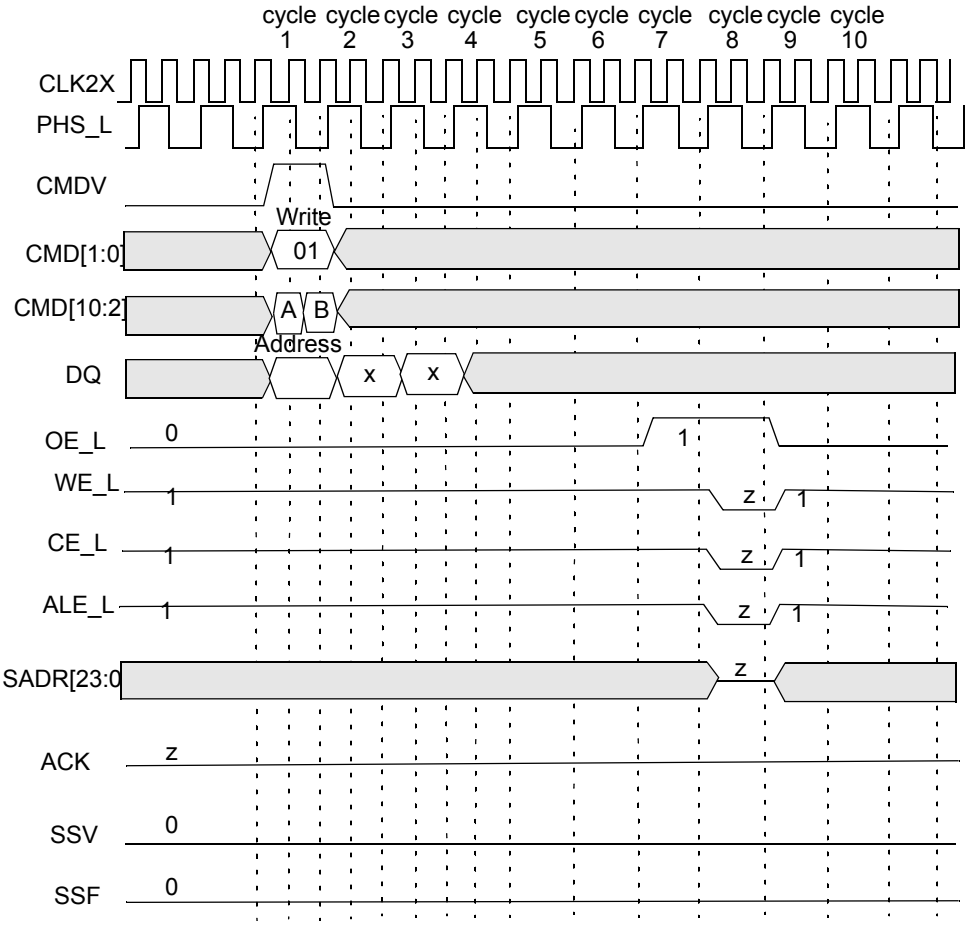


Figure 12-12. Table of 31 Devices (Four Blocks)



TLSZ = 10, HLAT = XXX, LRAM = 0, LDEV = 0.

Figure 12-13. SRAM Write Through Device Number 0 in a Bank of 31 Devices (Device 0 Timing)



TLSZ = 10, HLAT = XXX, LRAM = 1, LDEV = 1.

Figure 12-14. SRAM Write Through Device Number 0 in a Bank of 31 CYNSE70128 Devices (Device Number 30 Timing)

12.9 Timing Sequences for Back-to-Back Operations

Table 12-2 shows the idle cycle requirements between operations. The operations in the second column represent operations already performed, and the operations in the first row are those we would like to perform next.

Example calculations:

1. Read after Write: The Write takes two 2 cycles, and one 1 idle cycle is required. Thus if the Write is issued in cycle 1, the Read cannot be issued until cycle 4. Note, all cycles after an SRAM Read or an NSE Read (blocking) operation are considered blocked until the ACK signal is returned.
2. x72 Search after Read: The Read command can be issued in 1 cycle after which 5 idle cycles are required before issuing any other command. Thus if the Read is issued in cycle 1, the Search cannot be issued until cycle 7.

Table 12-2. Required Idle Cycles Between Commands

# of Cycles	OPERATIONS	SEARCH	READ	WRITE	LEARN	SRAM
x72/x144 = 1 Cycle x288 = 2 Cycles	SEARCH	No Wait	No Wait / (2 + TLSZ) [13]	No Wait	No Wait	TLSZ /(TLSZ + HLAT) [14]
1 Cycle	READ	5	5	5	5	5
2 Cycles	WRITE	1	1	1	1	1
x72/x144 = 1 Cycle	LEARN	1	1	1	1	1
1 Cycle	SRAM READ	5+TLSZ+HLAT	5+TLSZ+HLAT	5+TLSZ+HLAT	5+TLSZ+HLAT	5+TLSZ+HLAT
1 Cycles	SRAM WRITE	2	2	2	2	2

Notes:

13. When the register being read is SSR/SRR and it matches the target location of the previous search, a READ operation cannot be issued for 2+TLSZ idle cycles to avoid reading the old value. Otherwise there is no idle cycle requirement.
14. The SRAM operation needs to insert idle cycles to avoid SADR bus contention with previous SEARCH.

13.0 Power

CYNSE70128 has two separate power supplies, one for the core (V_{DD}) and another for the I/Os (V_{DDQ}).

13.1 Power-up Sequence

Proper power-up sequence is required to correctly initialize the Cypress NSEs before functional access to the device can begin. RST_L and TRST_L should be held low before the power supplies ramp-up. RST_L must be set low for a duration of time afterward and then set high. The following steps describe the proper power-up sequence.

1. Set RST_L and TRST_L low.
2. Power up V_{DD}, V_{DDQ} and start running CLK1X when operating in CLK1X mode or CLK2X and PHS_L when operating in CLK2X mode. The order in which these signals (including V_{DD} and V_{DDQ}) are applied is not critical.
3. RST_L should be held low for 0.5ms (PLL lock time requirement). In CLK1X mode, the counting starts on the first rising edge of CLK1X after both V_{DD} and V_{DDQ} have reached their steady state voltages. In CLK2X mode, the counting starts on the first rising edge of CLK2X when PHS_L is high, after both V_{DD} and V_{DDQ} have reached their steady state voltages.
4. Continue to hold RST_L low for a minimum of 32 CLK1X cycles (when operating in CLK1X mode) or 64 CLK2X cycles (when operating in CLK2X mode). Set RST_L to high afterward to complete the power-up sequence. For JTAG reset, TRST_L can be brought high after V_{DD} and V_{DDQ} have both reached their steady state voltages.
5. PHS_L does not need to be running during the reset. Also if JTAG is not used, TRST_L does not need to transition HIGH but instead can be held low. *Figure 13-1* and *Figure 13-2* illustrate the proper sequences of the power-up operation

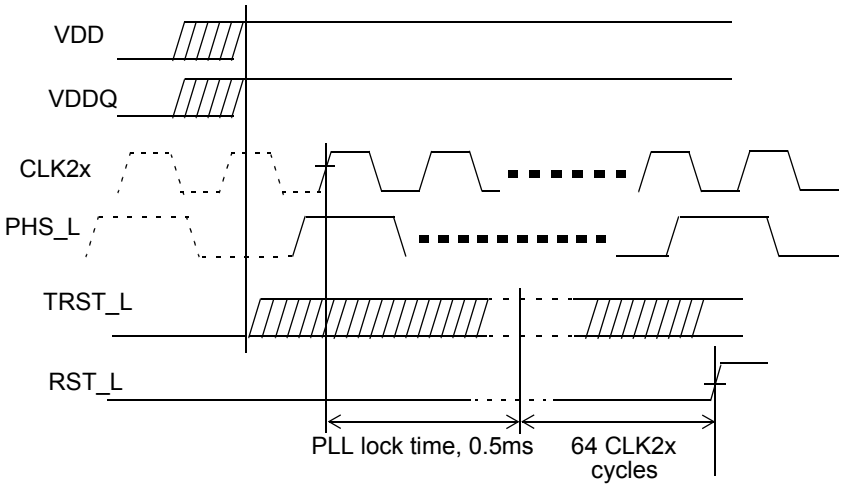


Figure 13-1. Power-up Sequence (CLK2x)

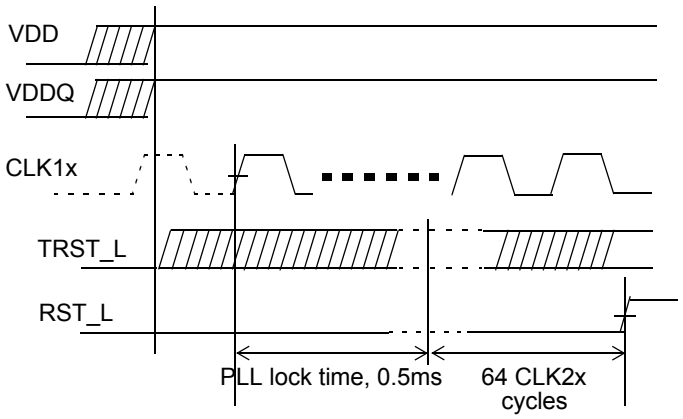


Figure 13-2. Power-up Sequence (CLK1x)

13.2 Power Consumption

Figure 13-3 depicts expected power consumption over a range of frequencies. The calculations assume 100% of the operations will be SEARCH operations. If an application includes other operations such as READ or WRITE, then power consumption will be lower. The worst case line indicates power consumption when the I/Os switch 100% of the time. The other lines (All Search Hit and All Search Miss) assume the I/Os switch 50% of the time.

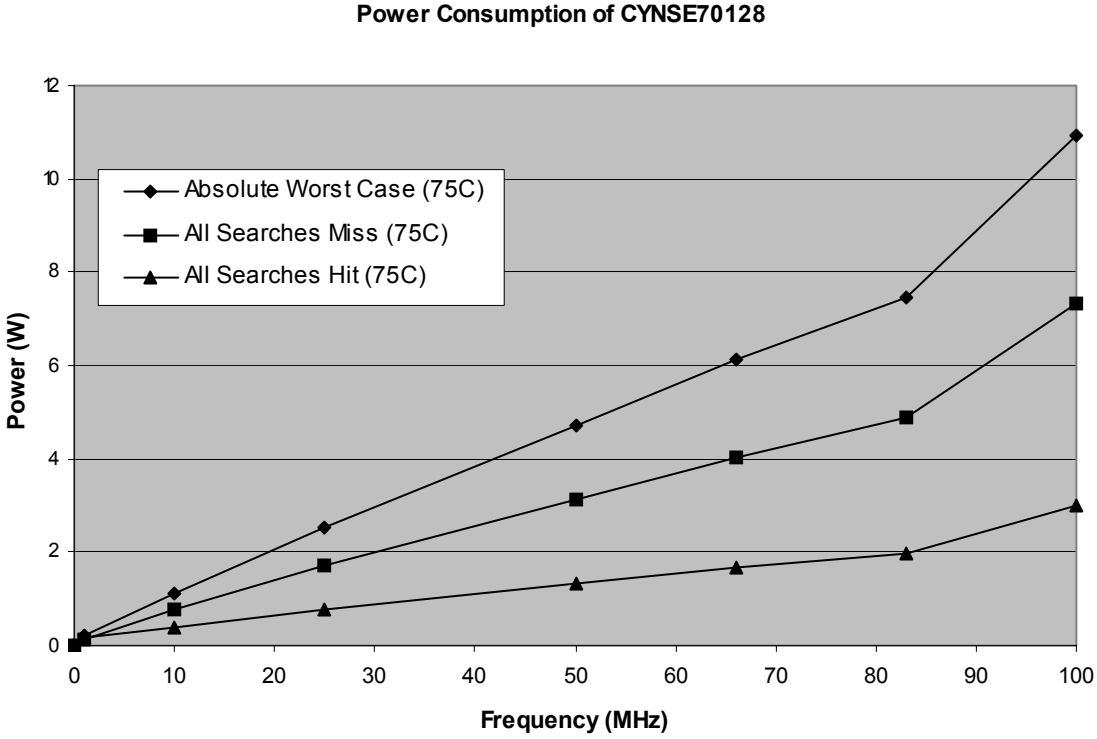


Figure 13-3. Power Consumption of CYNSE70128

14.0 Application

Figure 14-1 shows how an NSE subsystem can be formed using a host ASIC and an CYNSE70128 bank. It also shows how this NSE subsystem is integrated in a switch or router. The CYNSE70128 can access synchronous and asynchronous SRAMs by allowing the host ASIC to set the same HLAT parameter in all NSEs within a bank of NSEs.

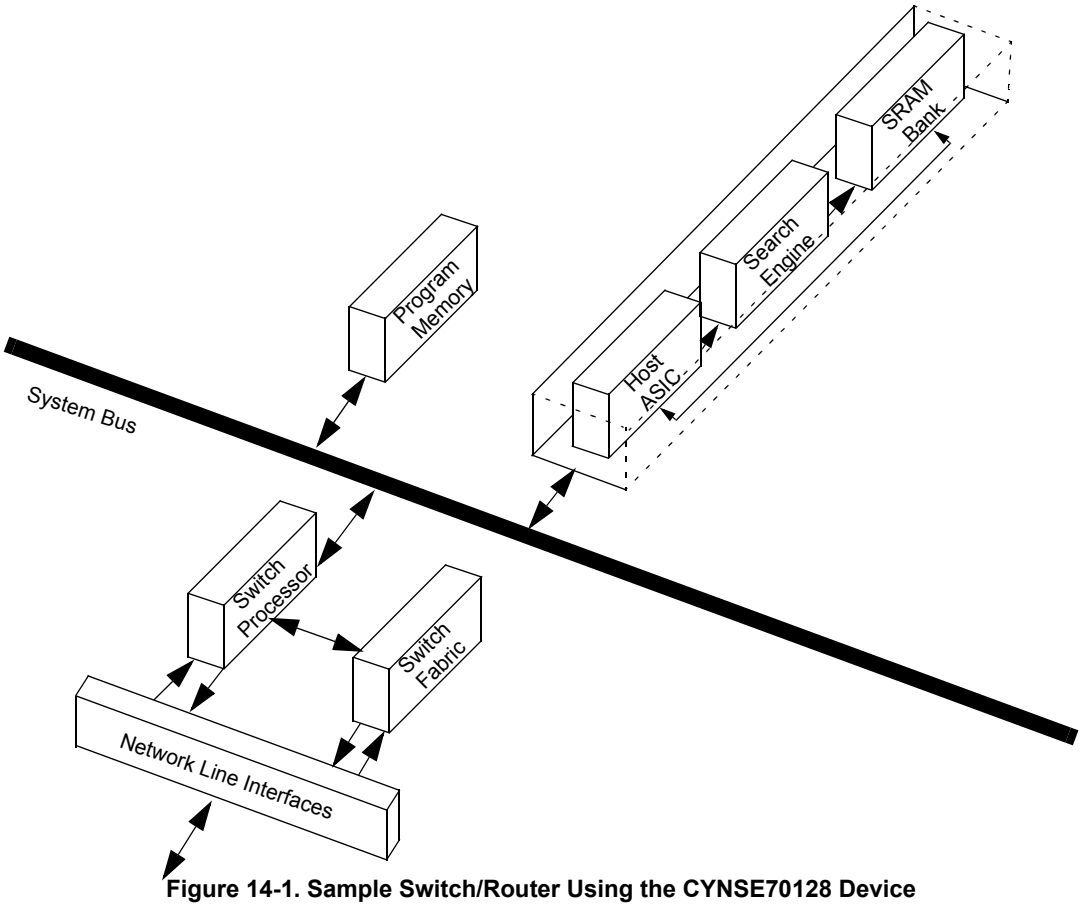


Figure 14-1. Sample Switch/Router Using the CYNSE70128 Device

15.0 JTAG (1149.1) Testing

The CYNSE70128 supports the Test Access Port and Boundary Scan Architecture as specified in the IEEE JTAG standard number 1149.1. The pin interface to the chip consists of five signals with the standard definitions: TCK, TMS, TDI, TDO, and TRST_L. *Table 15-1* describes the operations that the test access port controller supports, and *Table 15-2* describes the TAP Device ID Register. ***Note***. To disable JTAG functionality, connect the TCK, TMS and TDI pins to V_{DDQ} through a pull-up, and TRST_L to ground through a pull-down.

Table 15-1. Supported Operations

Instruction	Type	Description
SAMPLE/PRELOAD	Mandatory	This operation loads the values of signals going to and from I/O pins into the boundary scan shift register to provide a snapshot of the normal functional operation.
EXTEST	Mandatory	This operation uses boundary scan values shifted in from TAP to test connectivity external to the device.
BYPASS	Mandatory	This operation loads a single bit shift register between TDI and TDO and provides a minimum-length serial path when no test operation is required
IDCODE	Optional	This operation selects the Identification register between TDI and TDO and allows the “idcode” to be read serially through TDO.
CLAMP	Optional	This operation drives preset values onto the outputs of devices.
HighZ	Optional	This operation leaves the device output pins in a high-impedance state.

Table 15-2. TAP Device ID Register

Field	Range	Initial Value	Description
Revision	[31:28]	0001	Revision Number. This is the current device revision number. Numbers start from 1 and increment by 1 for each revision of the device.
Part Number	[27:12]	0000 0000 0000 0100	This is the part number for the device.
MFID	[11:1]	000_1101_1100	Manufacturer ID. This field is the same as the manufacturer ID used in the TAP controller.
LSB	[0]	1	Least significant bit.

16.0 Electrical Specifications

This section describes the electrical specifications, capacitance, operating conditions, DC characteristics, and AC timing parameters for the CYNSE70128, as shown in *Table 16-1* and *Table 16-2*.

Table 16-1. DC Electrical Characteristics for CYNSE70128

Parameter	Description	Test Condition	Min.	Max.	Unit
I_{LI}	Input leakage current	V_{DDQ} = V_{DDQ} Max, V_{IN}= 0 to V_{DDQ} Max	–10	10	mA
I_{LO}	Output leakage current	V_{DDQ} = V_{DDQ} Max, V_{IN}=0 to V_{DDQ} Max	–10	10	mA
V_{IL}	Input low voltage (V_{DDQ} = 3.3V)		–0.3	0.8	V
V_{IH}	Input high voltage (V_{DDQ} = 3.3V)		2.0	V_{DDQ} + 0.3	V
V_{IL}	Input low voltage (V_{DDQ} = 2.5V)		–0.3	0.7	V
V_{IH}	Input high voltage (V_{DDQ} = 2.5V)		1.7	V_{DDQ} + 0.3	V
V_{OL}	Output low voltage (V_{DDQ} = 3.3V)	V_{DDQ} = V_{DDQ} Min, I_{OL} = 16mA		0.4	V
V_{OH}	Output high voltage (V_{DDQ} = 3.3V)	V_{DDQ} = V_{DDQ} Min, I_{OH} = 8mA	2.4		V
V_{OL}	Output low voltage (V_{DDQ} = 2.5V)	V_{DDQ} = V_{DDQ} Min, I_{OL} = 8mA		0.4	V
V_{OH}	Output high voltage (V_{DDQ} = 2.5V)	V_{DDQ} = V_{DDQ} Min, I_{OH} = 8mA	2.0		V
I_{DD2}	3.3V supply current at V_{DD} Max	100 MHz search rate, I_{OUT} = 0mA		350	mA
I_{DD2}	3.3V supply current at V_{DD} Max	83 MHz search rate, I_{OUT} = 0mA		300	mA
I_{DD2}	3.3V supply current at V_{DD} Max	66 MHz search rate, I_{OUT} = 0mA		240	mA
I_{DD2}	2.5V supply current at V_{DD} Max	100 MHz search rate, I_{OUT} = 0mA		350	mA
I_{DD2}	2.5V supply current at V_{DD} Max	83 MHz search rate, I_{OUT} = 0mA		300	mA
I_{DD2}	2.5V supply current at V_{DD} Max	66 MHz search rate, I_{OUT} = 0mA		240	mA
I_{DDI}	1.65V supply current at V_{DD} Max	100 MHz search rate		6.0	A
I_{DDI}	1.5V supply current at V_{DD} Max	83 MHz search rate		5.0	A
I_{DDI}	1.5V supply current at V_{DD} Max	66 MHz search rate		4.0	A

Parameter	Description	Max.	Unit
C_{IN}	Input capacitance	6	pF[15]
C_{OUT}	Output capacitance	6	pF[16]

Notes:
15. f = 1 MHz, V_{IN} = 0 V.
16. f = 1 MHz, V_{OUT} = 0 V.

Table 16-2. Operating Conditions for CYNSE70128

Parameter	Description	Min.	Max.	Unit
V_{DDQ}	Operating voltage for I/O (3.3V)	3.1	3.5	V
V_{DDQ}	Operating voltage for I/O (2.5V)	2.375	2.625	V
V_{DD}	Operating supply voltage	1.425 (for CLK1X = 83 MHz) 1.568 (for CLK1X = 100 MHz)	1.575 (for CLK1X = 83 MHz) 1.733 (for CLK1X = 100 MHz)	V
V_{IH}	Input high voltage[17] (3.3V)	2.0	V_{DDQ} + 0.3	V
V_{IH}	Input high voltage[17] (2.5V)	1.7	V_{DDQ} + 0.3	V
V_{IL}	Input low voltage[18] (3.3V)	–0.3	0.8	V
V_{IL}	Input low voltage[18] (2.5)V	–0.3	0.7	V
t_A	Ambient operating temperature (Commercial)	0	+70	°C
t_A	Ambient operating temperature (Industrial)	–40	+85	°C
	Supply voltage tolerance	–5	+5	%

17.0 AC Timing Waveforms

Table 17-1 and *Table 17-2* show the AC timing parameters for the CYNSE70128 device; *Table 17-3* shows the same parameters but for 2.5V.

Table 17-1. AC Timing Parameters with CLK2X

Row	Parameter	Description		CYNSE70128-66 (V_{DDQ} = 3.3V, 2.5V) (V_{DD} = 1.5V)		CYNSE70128-83 (V_{DDQ} = 3.3V, 2.5V) (V_{DD} = 1.5V)		CYNSE70128-100 (V_{DDQ} = 3.3V, 2.5V) (V_{DD} = 1.65V)		Unit
				Min.	Max.	Min.	Max.	Min.	Max.	
1	f_{CLOCK}	CLK2X frequency.		40	133	40	166	40	200	MHz
2	t_{CLOK}	PLL lock time.			0.5		0.5		0.5	ms
3	t_{CKHI}	CLK2X high pulse.[19]		3.0		2.4		2.0		ns
4	t_{CKLO}	CLK2X low pulse.[19]		3.0		2.4		2.0		ns
5	t_{ISCH}	Input set-up time to CLK2X rising edge.[19]		2.5		1.8		1.5		ns
6	t_{IHCH}	Input hold time to CLK2X rising edge.[19]	Commercial	0.6		0.6		0.5		ns
			Industrial	0.8		0.8				ns
7	t_{ICSCH}	Cascaded input set-up time to CLK2X rising edge.[19]		4.2		3.5		3.0		ns
8	t_{ICHCH}	Cascaded input hold time to CLK2X rising edge.[19]		2.0		2.0		2.0		ns
9	t_{CKHOV}	Rising edge of CLK2X to LHO, FULO, BHO, FULL valid.[20]			8.5		7.0		6.5	ns
10	t_{CKHDV}	Rising edge of CLK2X to DQ valid.[20]			9.0		7.5		7.0	ns
11	t_{CKHDZ}	Rising edge of CLK2X to DQ high-Z.[21]		0.5	8.5	0.5	7.0	0.5	6.5	ns
12	t_{CKHSV}	Rising edge of CLK2X to SRAM bus valid.[20]			9.0		7.5		7.0	ns
13	t_{CKHSHZ}	Rising edge of CLK2X to SRAM bus high-Z.[21]		0.5	6.5	0.5	6.0	0.5	5.5	ns
14	t_{CKHSLZ}	Rising edge of CLK2X to SRAM bus low-Z.[21]		7.0		6.5		6.0		ns

Notes:
17. Maximum allowable applies to overshoot only (V_{DDQ} is 2.5 V supply).
18. Minimum allowable applies to undershoot only.
19. Values are based on 50% signal levels.
20. Based on an AC load of CL = 30 pF (see *Figure 17-1*, *Figure 17-2*, and *Figure 17-3*).
21. These parameters are sampled but not 100% tested, and are based on an AC load of 5 pF.

Table 17-2. AC Timing Parameters with CLK1X

Row	Parameter	Description		CYNSE70128-066 (V_{DDQ} = 3.3V, 2.5V) (V_{DD} = 1.5V) Min.	CYNSE70128-066 (V_{DDQ} = 3.3V, 2.5V) (V_{DD} = 1.5V) Max.	CYNSE70128-083 (V_{DDQ} = 3.3V, 2.5V) (V_{DD} = 1.5V) Min.	CYNSE70128-083 (V_{DDQ} = 3.3V, 2.5V) (V_{DD} = 1.5V) Max.	CYNSE70128-100 (V_{DDQ} = 3.3V, 2.5V) (V_{DD} = 1.65V) Min.	CYNSE70128-100 (V_{DDQ} = 3.3V, 2.5V) (V_{DD} = 1.65V) Max.	Unit
1	f_{CLOCK}	CLK1X frequency.		20	66	20	83	20	100	MHz
2	t_{CLOK}	PLL lock time.			0.5		0.5		0.5	ms
3	t_{CKHI}	CLK1X high pulse.[22]		6.75		5.4		4.5		ns
4	t_{CKLO}	CLK1X low pulse.[22]		6.75		5.4		4.5		ns
5	t_{ISCH}	Input set-up time to CLK1X edge.[22]		2.5		1.8		1.5		ns
6	t_{IHCH}	Input hold time to CLK1X edge.[22]	Commercial	0.6		0.6		0.5		ns
			Industrial	0.9		0.9				ns
7	t_{ICSCH}	Cascaded input set-up time to CLK1X rising edge.[22]		4.2		3.5		3.0		ns
8	t_{ICHCH}	Cascaded input hold time to CLK1X rising edge.[22]		2.0		2.0		2.0		ns
9	t_{CKHOV}	Rising edge of CLK1X to LHO, FULO, BHO, FULL valid.[23]			8.5		7.0		6.5	ns
10	t_{CKHDV}	Rising edge of CLK1X to DQ valid.[23]			9.0		7.5		7.0	ns
11	t_{CKHDZ}	Rising edge of CLK1X to DQ high-Z.[24]		0.5	8.5	0.5	7.0	0.5	6.5	ns
12	t_{CKHSV}	Rising edge of CLK1X to SRAM bus valid.[23]			9.0		7.5		7.0	ns
13	t_{CKHSHZ}	Rising edge of CLK1X to SRAM bus high-Z.[24]		0.5	6.5	0.5	6.0	0.5	5.5	ns
14	t_{CKHSLZ}	Rising edge of CLK1X to SRAM bus low-Z.[24]		7.0		6.5		6.0		ns

Table 17-3. 2.5V AC Table for Test Condition of CYNSE70128

Conditions	Results
Input pulse levels (V_{DDQ} = 3.3V)	GND to 3.0V
Input pulse levels (V_{DDQ} = 2.5V)	GND to 2.5V
Input rise and fall times measured at 0.3V and 2.7V (V_{DDQ} = 3.3V)	≤ 2 ns see *Figure 17-1*
Input rise and fall times measured at 0.25V and 2.25V (V_{DDQ} = 2.5V)	≤ 2 ns see *Figure 17-1*
Input timing reference levels (V_{DDQ} = 3.3V)	1.5V
Input timing reference levels (V_{DDQ} = 2.5V)	1.25
Output reference levels (V_{DDQ} = 3.3V)	1.5V
Output reference levels (V_{DDQ} = 2.5V)	1.25V
Output load	See *Figure 17-2* and *Figure 17-3*

Notes:
22. Values are based on 50% signal levels and a 50%/50% duty cycle of CLK1X.
23. Based on an AC load of CL = 30 pF (see *Figure 17-1*, *Figure 17-2*, and *Figure 17-3*).
24. These parameters are sampled but not 100% tested, and are based on an AC load of 5 pF.

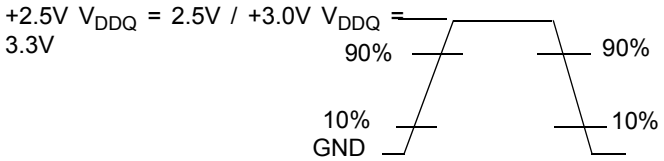


Figure 17-1. Input Wave Form for CYNSE70128

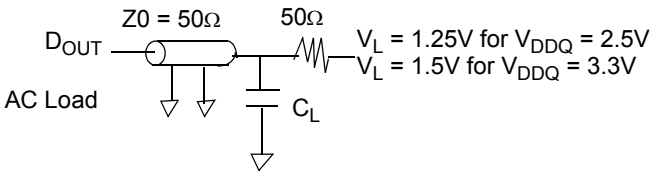


Figure 17-2. Output Load for CYNSE70128

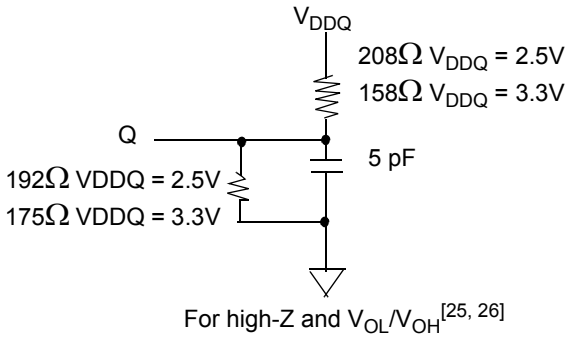
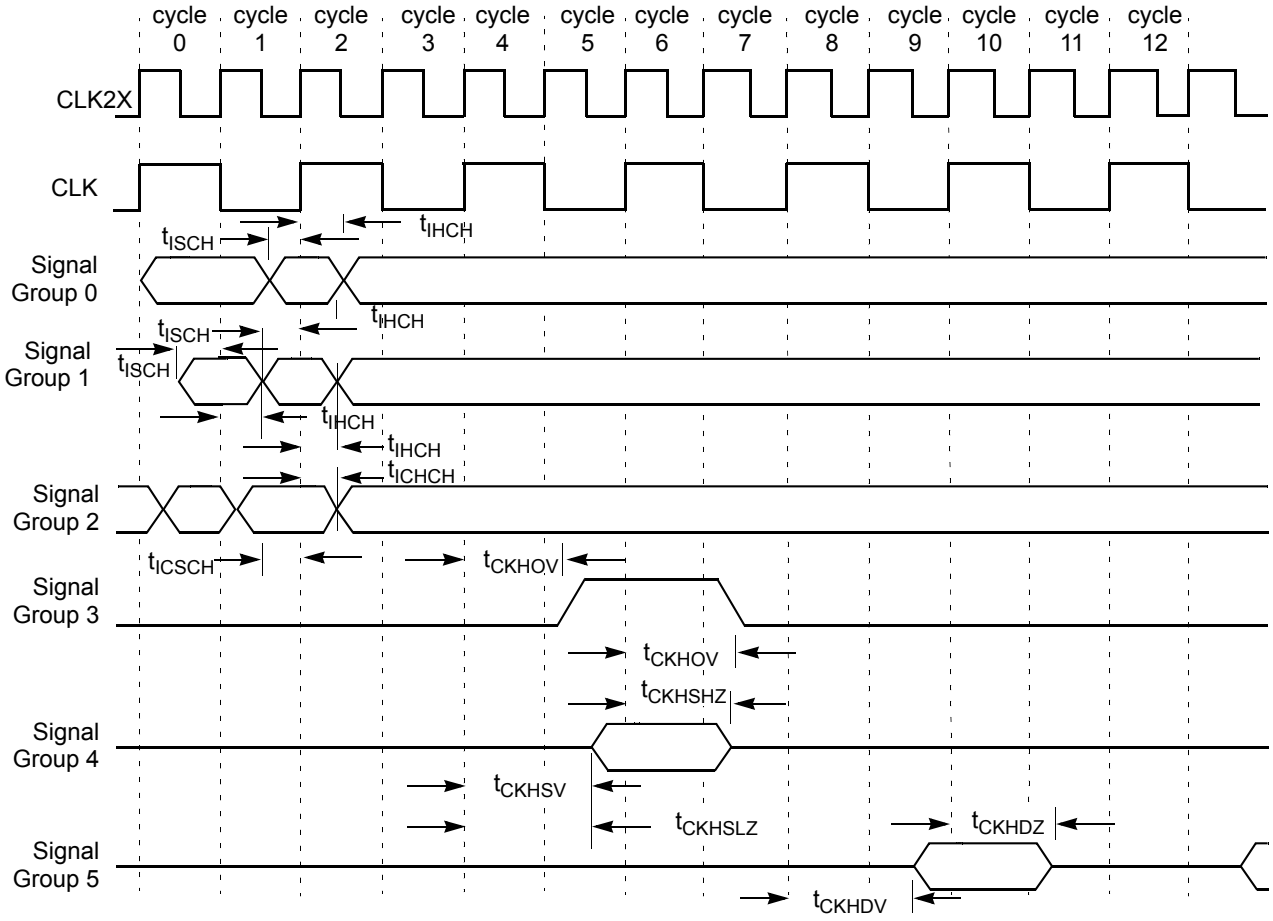


Figure 17-3. I/O Output Load Equivalent for CYNSE70128

Figure 17-4 shows timing waveform diagrams for CLK2X.

25. Output loading is specified with C_L = 5 pF, as in *Figure 17-3*. Transition is measured at ± 200 mV from steady-state voltage.
26. The load used for VOH, VOL testing is shown in *Figure 17-3*.



Signal Group 0: PHS_L, RST_L.
Signal Group 1: DQ, CMD, CMDV.
Signal Group 2: LHI, BHI, FULI.
Signal Group 3: LHO, BHO, FULO, FULL.
Signal Group 4: SADR, CE_L, OE_L, WE_L, ALE_L, SSF, SSV.
Signal Group 5: DQ, ACK, EOT.

Figure 17-4. AC Timing Waveforms with CLK2X

17.1 Special Note for MULTI_HIT Function on the CYNSE70128

General Description: The CYNSE70128 device provides a “MULTI_HIT” signal as an output. The purpose of this signal is to indicate the occurrence of a multiple hit on a Search in the search engine.

Correct Usage: In order to ensure correct function for MULTI_HIT on CYNSE70128:

- Backward compatibility mode (with CYNSE70032 and CYNSE70064): Data array bits need to be used as table ID bits.
- Non-backward compatibility mode: Either data array bits need to still be used as table ID bits, or the entire data array needs to be configured with one table size.
- In multiple NSE configuration (cascaded mode), the “OR” function for the MULTI_HITs from each NSE must be provided externally.

The MULTI_HIT signal should be sampled on four CLK1X cycles (or on eight CLK2X cycles) after the issue of the Search command, regardless of the TLSZ and HLAT parameters in the command register.

18.0 Pinout Description

In the following figure and table, the CYNSE70128 device pinout diagram and descriptions are shown (see *Figure 18-1* and *Table 18-1*).

	AF	AE	AD	AC	AB	AA	Y	W	V	U	T	R	P	N	M	L	K	J	H	G	F	E	D	C	B	A	
1	NC	V_{SS}	RST_L	V_{SS}	FULL	FULO1	FULI6	V_{DDQ}	FULI2	FULI0	BHO2	V_{DDQ}	BHO0	BHI1	V_{DDQ}	LHO0	LHI6	LHI2	LHI0	ID3	ID1	ID0	TRST_L	TCK	TDI	V_{DD}	1
2	V_{SS}	V_{SS}	V_{DDQ}	EOT	ACK	V_{DDQ}	FULO0	FULI5	FULI3	V_{DDQ}	V_{SS}	BHO1	MULTI_HIT	BHI2	BHI0	LHO1	LHI4	LHI3	LHI1	ID4	ID2	V_{DDQ}	TDO	TMS	V_{SS}	DQ71	2
3	DQ68	DQ70	V_{DD}	V_{DD}	V_{DD}	V_{DD}	V_{DD}	NC	FULI4	FULI1	V_{DD}	V_{DD}	V_{DD}	V_{DD}	V_{DD}	V_{DD}	LHI5	V_{DDQ}	NC	V_{DD}	V_{DD}	V_{DD}	V_{DD}	V_{DD}	DQ69	V_{DDQ}	3
4	DQ66	V_{DDQ}	V_{DD}	V_{SS}	V_{SS}	V_{SS}	V_{SS}	V_{SS}	V_{SS}	V_{SS}	V_{DD}	V_{DD}	V_{DD}	V_{DD}	V_{DD}	V_{DD}	V_{SS}	V_{SS}	V_{SS}	V_{SS}	V_{SS}	V_{SS}	V_{SS}	V_{DD}	DQ65	DQ67	4
5	DQ62	DQ64	V_{DD}	V_{SS}																			V_{SS}	V_{DD}	DQ61	DQ63	5
6	V_{DDQ}	DQ60	V_{DD}	V_{SS}																			V_{SS}	V_{DD}	DQ59	V_{DDQ}	6
7	DQ56	DQ58	V_{DD}	V_{SS}																			V_{SS}	V_{DD}	DQ55	DQ57	7
8	DQ52	DQ54	NC	V_{SS}																			V_{SS}	NC	V_{DDQ}	DQ53	8
9	DQ48	DQ50	V_{DDQ}	V_{SS}																			V_{SS}	DQ49	DQ47	DQ51	9
10	V_{DDQ}	DQ44	DQ46	V_{SS}																			V_{SS}	V_{DDQ}	DQ45	DQ43	10
11	DQ40	DQ42	V_{DD}	V_{DD}							V_{SS}	V_{SS}	V_{SS}	V_{SS}	V_{SS}	V_{SS}							V_{DD}	V_{DD}	DQ39	DQ41	11
12	DQ36	DQ38	V_{DD}	V_{DD}							V_{SS}	V_{SS}	V_{SS}	V_{SS}	V_{SS}	V_{SS}							V_{DD}	V_{DD}	V_{DDQ}	DQ37	12
13	DQ34	V_{DDQ}	V_{DD}	V_{DD}							V_{SS}	V_{SS}	V_{SS}	V_{SS}	V_{SS}	V_{SS}							V_{DD}	V_{DD}	DQ33	DQ35	13
14	DQ30	DQ32	V_{DD}	V_{DD}							V_{SS}	V_{SS}	V_{SS}	V_{SS}	V_{SS}	V_{SS}							V_{DD}	V_{DD}	DQ29	DQ31	14
15	V_{DDQ}	DQ28	V_{DD}	V_{DD}							V_{SS}	V_{SS}	V_{SS}	V_{SS}	V_{SS}	V_{SS}							V_{DD}	V_{DD}	DQ27	V_{DDQ}	15
16	DQ24	DQ26	V_{DD}	V_{DD}							V_{SS}	V_{SS}	V_{SS}	V_{SS}	V_{SS}	V_{SS}							V_{DD}	V_{DD}	DQ23	DQ25	16
17	DQ22	V_{DDQ}	DQ20	V_{SS}																			V_{SS}	DQ19	V_{DDQ}	DQ21	17
18	DQ14	DQ18	DQ16	V_{SS}																			V_{SS}	DQ13	DQ15	DQ17	18
19	V_{DDQ}	DQ12	NC	V_{SS}																			V_{SS}	NC	DQ11	V_{DDQ}	19
20	DQ08	DQ10	V_{DD}	V_{SS}																			V_{SS}	V_{DD}	DQ07	DQ09	20
21	DQ04	DQ06	V_{DD}	V_{SS}																			V_{SS}	V_{DD}	V_{DDQ}	DQ05	21
22	DQ02	V_{DDQ}	V_{DD}	V_{SS}																			V_{SS}	V_{DD}	DQ01	DQ03	22
23	SSV	DQ00	V_{DD}	V_{SS}	V_{SS}	V_{SS}	V_{SS}	V_{SS}	V_{SS}	V_{SS}	V_{DD}	V_{DD}	V_{DD}	V_{DD}	V_{DD}	V_{DD}	V_{SS}	V_{SS}	V_{SS}	V_{SS}	V_{SS}	V_{SS}	V_{SS}	V_{SS}	V_{DD}	V_{SS}	V_{SS}	23
24	SSF	V_{DDQ}	V_{DD}	V_{DD}	V_{DD}	V_{DD}	V_{DD}	NC	CE_L	OE_L	V_{DD}	V_{DD}	V_{DD}	V_{DD}	V_{DD}	V_{DD}	SADR13	SADR11	NC	V_{DD}	V_{DD}	V_{DD}	V_{DD}	V_{DD}	CFG_L	V_{DDQ}	24
25	CMD10	V_{SS}	CMD8	CMD6	CMD5	CMD3	CMD1	CMDV	V_{DDQ}	PHS_L	CLK_MODE	SADR22	SADR21	SADR19	V_{DDQ}	SADR15	V_{DDQ}	SADR12	V_{DDQ}	SADR08	SADR06	SADR05	SADR03	SADR01	V_{SS}	HIGH_SPEED	25
26	CMD9	V_{SS}	CMD7	V_{DDQ}	CMD4	CMD2	CMD0	ALE_L	WE_L	CLK1X/ CLK2X	SADR23	V_{DDQ}	SADR20	SADR18	SADR17	SADR16	SADR14	SADR10	SADR09	SADR07	V_{DDQ}	SADR04	SADR02	V_{DDQ}	SADR00	V_{DD}	26
	AF	AE	AD	AC	AB	AA	Y	W	V	U	T	R	P	N	M	L	K	J	H	G	F	E	D	C	B	A	

Figure 18-1. Pinout Diagram

Table 18-1. Pinout Descriptions for Pinout Diagram

Package Ball Number	Signal Name	Signal Type	Package Ball Number	Signal Name	Signal Type
A1	V_{DDQ}	2.5V/3.3V	AA26	CMD[2]	Input
A10	DQ[43]	I/O	AA3	V_{DD}	1.5V/1.65V
A11	DQ[41]	I/O	AA4	V_{SS}	Ground
A12	DQ[37]	I/O	AB1	FULL	Output-T
A13	DQ[35]	I/O	AB2	ACK	Output-T
A14	DQ[31]	I/O	AB23	VSS	Ground
A15	V_{DDQ}[27]	2.5V/3.3V	AB24	V_{DD}	1.5V/1.65V
A16	DQ[25]	I/O	AB25	CMD[5]	Input
A17	DQ[21]	I/O	AB26	CMD[4]	Input
A18	DQ[17]	I/O	AB3	V_{DD}	1.5V/1.65V
A19	V_{DDQ}	2.5V/3.3V	AB4	V_{SS}	Ground
A2	DQ[71]	I/O	AC1	V_{SS}	Ground
A20	DQ[09]	I/O	AC10	V_{SS}	Ground
A21	DQ[05]	I/O	AC11	V_{DD}	1.5V/1.65V
A22	DQ[03]	I/O	AC12	V_{DD}	1.5V/1.65V
A23	V_{SS}	Ground	AC13	V_{DD}	1.5V/1.65V
A24	V_{DDQ}	2.5V/3.3V	AC14	V_{DD}	1.5V/1.65V
A25	HIGH_SPEED	Input	AC15	V_{DD}	1.5V/1.65V
A26	V_{DDQ}	2.5V/3.3V	AC16	V_{DD}	1.5V/1.65V
A3	V_{DDQ}	2.5V/3.3V	AC17	V_{SS}	Ground
A4	DQ[67]	I/O	AC18	V_{SS}	Ground
A5	DQ[63]	I/O	AC19	V_{SS}	Ground
A6	V_{DDQ}	2.5V/3.3V	AC2	EOT	Output-T
A7	DQ[57]	I/O	AC20	V_{SS}	Ground
A8	DQ[53]	I/O	AC21	V_{SS}	Ground
A9	DQ[51]	I/O	AC22	V_{SS}	Ground
AA1	FULO[1]	Output-T	AC23	V_{SS}	Ground
AA2	V_{DDQ}	2.5V/3.3V	AC24	V_{DD}	1.5V/1.65V
AA23	V_{SS}	Ground	AC25	CMD[6]	Input
AA24	V_{DD}	1.5V/1.65V	AC26	V_{DDQ}	2.5V/3.3V
AA25	CMD[3]	Input	AC3	V_{DD}	1.5V/1.65V
AC4	V_{SS}	Ground	AE10	DQ[44]	I/O
AC5	V_{SS}	Ground	AE11	DQ[42]	I/O
AC6	V_{SS}	Ground	AE12	DQ[38]	I/O
AC7	V_{SS}	Ground	AE13	V_{DDQ}	2.5V/3.3V
AC8	V_{SS}	Ground	AE14	DQ[32]	I/O
AC9	V_{SS}	Ground	AE15	DQ[28]	I/O
AD1	RST_L	Input	AE16	DQ[26]	I/O
AD10	DQ[46]	I/O	AE17	V_{DDQ}	2.5V/3.3V
AD11	V_{DD}	1.5V/1.65V	AE18	DQ[18]	I/O
AD12	V_{DD}	1.5V/1.65V	AE19	DQ[12]	I/O
AD13	V_{DD}	1.5V/1.65V	AE2	V_{SS}	Ground
AD14	V_{DD}	1.5V/1.65V	AE20	DQ[10]	I/O

Table 18-1. Pinout Descriptions for Pinout Diagram (continued)

Package Ball Number	Signal Name	Signal Type	Package Ball Number	Signal Name	Signal Type
AD15	V_{DD}	1.5V/1.65V	AE21	DQ[06]	I/O
AD16	V_{DD}	1.5V/1.65V	AE22	V_{DDQ}	2.5V/3.3V
AD17	DQ[20]	I/O	AE23	DQ[00]	I/O
AD18	DQ[16]	I/O	AE24	V_{DDQ}	2.5V/3.3V
AD19	NC	No Connect	AE25	V_{SS}	Ground
AD2	V_{DDQ}	2.5V/3.3V	AE26	V_{SS}	Ground
AD20	V_{DD}	1.5V/1.65V	AE3	DQ[70]	I/O
AD21	V_{DD}	1.5V/1.65V	AE4	V_{DDQ}	2.5V/3.3V
AD22	V_{DD}	1.5V/1.65V	AE5	DQ[64]	I/O
AD23	V_{DD}	1.5V/1.65V	AE6	DQ[60]	I/O
AD24	V_{DD}	1.5V/1.65V	AE7	DQ[58]	I/O
AD25	CMD[8]	Input	AE8	DQ[54]	I/O
AD26	CMD[7]	Input	AE9	DQ[50]	I/O
AD3	V_{DD}	1.5V/1.65V	AF1	NC	No Connect
AD4	V_{DD}	1.5V/1.65V	AF10	V_{DDQ}	2.5V/3.3V
AD5	V_{DD}	1.5V/1.65V	AF11	DQ[40]	I/O
AD6	V_{DD}	1.5V/1.65V	AF12	DQ[36]	I/O
AD7	V_{DD}	1.5V/1.65V	AF13	DQ[34]	I/O
AD8	NC	No Connect	AF14	DQ[30]	I/O
AD9	V_{DDQ}	2.5V/3.3V	AF15	V_{DDQ}	2.5V/3.3V
AE1	V_{SS}	Ground	AF16	DQ[24]	I/O
AF17	DQ[22]	I/O	B23	V_{SS}	Ground
AF18	DQ[14]	I/O	B24	CFG_L	Input
AF19	V_{DDQ}	2.5V/3.3V	B25	V_{SS}	Ground
AF2	V_{SS}	Ground	B26	SADR[00]	Output
AF20	DQ[08]	I/O	B3	DQ[69]	I/O
AF21	DQ[04]	I/O	B4	DQ[65]	I/O
AF22	DQ[02]	I/O	B5	DQ[61]	I/O
AF23	SSV	Output-T	B6	DQ[59]	I/O
AF24	SSF	Output-T	B7	DQ[55]	I/O
AF25	CMD[10]	Input	B8	V_{DDQ}	2.5V/3.3V
AF26	CMD[9]	Input	B9	DQ[47]	I/O
AF3	DQ[68]	I/O	C1	TCK	Input
AF4	DQ[66]	I/O	C10	V_{DDQ}	2.5V/3.3V
AF5	DQ[62]	I/O	C11	V_{DD}	1.5V/1.65V
AF6	V_{DDQ}	2.5V/3.3V	C12	V_{DD}	1.5V/1.65V
AF7	DQ[56]	I/O	C13	V_{DD}	1.5V/1.65V
AF8	DQ[52]	I/O	C14	V_{DD}	1.5V/1.65V
AF9	DQ[48]	I/O	C15	V_{DD}	1.5V/1.65V
B1	TDI	Input	C16	V_{DD}	1.5V/1.65V
B10	DQ[45]	I/O	C17	DQ[19]	I/O
B11	DQ[39]	I/O	C18	DQ[13]	I/O
B12	V_{DDQ}	2.5V/3.3V	C19	NC	No Connect

Table 18-1. Pinout Descriptions for Pinout Diagram (continued)

Package Ball Number	Signal Name	Signal Type	Package Ball Number	Signal Name	Signal Type
B13	DQ[33]	I/O	C2	TMS	Input
B14	DQ[29]	I/O	C20	V_{DD}	1.5V/1.65V
B15	DQ[27]	I/O	C21	V_{DD}	1.5V/1.65V
B16	DQ[23]	I/O	C22	V_{DD}	1.5V/1.65V
B17	V_{DDQ}	2.5V/3.3V	C23	V_{DD}	1.5V/1.65V
B18	DQ[15]	I/O	C24	V_{DD}	1.5V/1.65V
B19	DQ[11]	I/O	C25	SADR[01]	Output
B2	V_{SS}	Ground	C26	V_{DDQ}	2.5V/3.3V
B20	DQ[07]	I/O	C3	V_{DD}	1.5V/1.65V
B21	V_{DDQ}	2.5V/3.3V	C4	V_{DD}	1.5V/1.65V
B22	DQ[01]	I/O	C5	V_{DD}	1.5V/1.65V
C6	V_{DD}	1.5V/1.65V	E24	V_{DD}	1.5V/1.65V
C7	V_{DD}	1.5V/1.65V	E25	SADR[05]	Output
C8	NC	No Connect	E26	SADR[04]	Output
C9	DQ[49]	I/O	E3	V_{DD}	1.5V/1.65V
D1	TRST_L	Input	E4	V_{SS}	Ground
D10	VSS	Ground	F1	ID[1]	Input
D11	V_{DD}	1.5V/1.65V	F2	ID[2]	Input
D12	V_{DD}	1.5V/1.65V	F23	V_{SS}	Ground
D13	V_{DD}	1.5V/1.65V	F24	V_{DD}	1.5V/1.65V
D14	V_{DD}	1.5V/1.65V	F25	SADR[06]	Output
D15	V_{DD}	1.5V/1.65V	F26	V_{DDQ}	2.5V/3.3V
D16	V_{DD}	1.5V/1.65V	F3	V_{DD}	1.5V/1.65V
D17	V_{SS}	Ground	F4	V_{SS}	Ground
D18	V_{SS}	Ground	G1	ID[3]	Input
D19	V_{SS}	Ground	G2	ID[4]	Input
D2	TDO	Output-T	G23	V_{SS}	Ground
D20	V_{SS}	Ground	G24	V_{DD}	1.5V/1.65V
D21	V_{SS}	Ground	G25	SADR[08]	Output
D22	V_{SS}	Ground	G26	SADR[07]	Output
D23	V_{SS}	Ground	G3	V_{DD}	1.5V/1.65V
D24	V_{DD}	1.5V/1.65V	G4	V_{SS}	Ground
D25	SADR[03]	Output	H1	LHI[0]	Input
D26	SADR[02]	Output	H2	LHI[1]	Input
D3	V_{DD}	1.5V/1.65V	H23	V_{SS}	Ground
D4	V_{SS}	Ground	H24	NC	No Connect
D5	V_{SS}	Ground	H25	V_{DDQ}	2.5V/3.3V
D6	V_{SS}	Ground	H26	SADR[09]	Output
D7	V_{SS}	Ground	H3	NC	No Connect
D8	V_{SS}	Ground	H4	V_{SS}	Ground
D9	V_{SS}	Ground	J1	LHI[2]	Input
E1	ID[0]	Input	J2	LHI[3]	Input
E2	V_{DDQ}	2.5V/3.3V	J23	V_{SS}	Ground

Table 18-1. Pinout Descriptions for Pinout Diagram (continued)

Package Ball Number	Signal Name	Signal Type	Package Ball Number	Signal Name	Signal Type
E23	V_{SS}	Ground	J24	SADR[11]	Output
J25	SADR[12]	Output	M2	BHI[0]	Input
J26	SADR[10]	Output	M23	V_{DD}	1.5V/1.65V
J3	V_{DDQ}	2.5V/3.3V	M24	V_{DD}	1.5V/1.65V
J4	V_{SS}	Ground	M25	V_{DDQ}	2.5V/3.3V
K1	LHI[6]	Input	M26	SADR[17]	Output
K2	LHI[4]	Input	M3	V_{DD}	1.5V/1.65V
K23	V_{SS}	Ground	M4	V_{DD}	1.5V/1.65V
K24	SADR[13]	Output	N1	BHI[1]	Input
K25	V_{DDQ}	2.5V/3.3V	N11	V_{SS}	Ground
K26	SADR[14]	Output	N12	V_{SS}	Ground
K3	LHI[5]	Input	N13	V_{SS}	Ground
K4	V_{SS}	Ground	N14	V_{SS}	Ground
L1	LHO[0]	Output-T	N15	V_{SS}	Ground
L11	V_{SS}	Ground	N16	V_{SS}	Ground
L12	V_{SS}	Ground	N2	BHI[2]	Input
L13	V_{SS}	Ground	N23	V_{DD}	1.5V/1.65V
L14	V_{SS}	Ground	N24	V_{DD}	1.5V/1.65V
L15	V_{SS}	Ground	N25	SADR[19]	Output
L16	V_{SS}	Ground	N26	SADR[18]	Output
L2	LHO[1]	Output-T	N3	V_{DD}	1.5V/1.65V
L23	V_{DD}	1.5V/1.65V	N4	V_{DD}	1.5V/1.65V
L24	V_{DD}	1.5V/1.65V	P1	BHO[0]	Output-T
L25	SADR[15]	Output	P11	V_{SS}	Ground
L26	SADR[16]	Output	P12	V_{SS}	Ground
L3	V_{DD}	1.5V/1.65V	P13	V_{SS}	Ground
L4	V_{DD}	1.5V/1.65V	P14	V_{SS}	Ground
M1	V_{DDQ}	2.5V/3.3V	P15	V_{SS}	Ground
M11	V_{SS}	Ground	P16	V_{SS}	Ground
M12	V_{SS}	Ground	P2	MULTI_HIT	Output-T
M13	V_{SS}	Ground	P23	V_{DD}	1.5V/1.65V
M14	V_{SS}	Ground	P24	V_{DD}	1.5V/1.65V
M15	V_{SS}	Ground	P25	SADR[21]	Output
M16	V_{SS}	Ground	P26	SADR[20]	Output
P3	V_{DD}	1.5V/1.65V	U24	OE_L	Output-T
P4	V_{DD}	1.5V/1.65V	U25	PHS_L	Input
R1	V_{DDQ}	2.5V/3.3V	U26	CLK1X/CLK2X	Input
R11	V_{SS}	Ground	U3	FULI[1]	Input
R12	V_{SS}	Ground	U4	V_{SS}	Ground
R13	V_{SS}	Ground	V1	FULI[2]	Input
R14	V_{SS}	Ground	V2	FULI[3]	Input
R15	V_{SS}	Ground	V23	V_{SS}	Ground
R16	V_{SS}	Ground	V24	CE_L	Output-T

Table 18-1. Pinout Descriptions for Pinout Diagram (continued)

Package Ball Number	Signal Name	Signal Type	Package Ball Number	Signal Name	Signal Type
R2	BHO[1]	Output-T	V25	V_{DDQ}	2.5V/3.3V
R23	V_{DD}	1.5V/1.65V	V26	WE_L	Output-T
R24	V_{DD}	1.5V/1.65V	V3	FULI[4]	Input
R25	SADR[22]	Output	V4	V_{SS}	Ground
R26	V_{DDQ}	2.5V/3.3V	W1	V_{DDQ}	2.5V/3.3V
R3	V_{DD}	1.5V/1.65V	W2	FULI[5]	Input
R4	V_{DD}	1.5V/1.65V	W23	V_{SS}	Ground
T1	BHO[2]	Output-T	W24	NC	No Connect
T11	V_{SS}	Ground	W25	CMDV	Input
T12	V_{SS}	Ground	W26	ALE_L	Output-T
T13	V_{SS}	Ground	W3	NC	No Connect
T14	V_{SS}	Ground	W4	V_{SS}	Ground
T15	V_{SS}	Ground	Y1	FULI[6]	Input
T16	V_{SS}	Ground	Y2	FULO[0]	Output-T
T2	V_{SS}	Ground	Y23	V_{SS}	Ground
T23	V_{DD}	1.5V/1.65V	Y24	V_{DD}	1.5V/1.65V
T24	V_{DD}	1.5V/1.65V	Y25	CMD[1]	Input
T25	CLK_MODE	Input	Y26	CMD[0]	Input
T26	SADR[23]	Output	Y3	V_{DD}	1.5V/1.65V
T3	V_{DD}	1.5V/1.65V	Y4	V_{SS}	Ground
T4	V_{DD}	1.5V/1.65V			
U1	FULI[0]	Input			
U2	V_{DDQ}	2.5V/3.3V			
U23	V_{SS}	Ground			

19.0 Ordering Information

Table 19-1 provides ordering information.

Table 19-1. Ordering Information

Part Number	Description	I/O Voltage	Frequency	Temperature Range
CYNSE70128–66BGC	NSE	2.5V/3.3V	66 MHz	Commercial
CYNSE70128–66BGI	NSE	2.5V/3.3V	66 MHz	Industrial
CYNSE70128–83BGC	NSE	2.5V/3.3V	83 MHz	Commercial
CYNSE70128–83BGI	NSE	2.5V/3.3V	83 MHz	Industrial
CYNSE70128–100BGC	NSE	2.5V/3.3V	100 MHz	Commercial

Note:
27. All V_{DDQ} pins should be set to 2.5V or 3.3V (CYNSE70128).

© Cypress Semiconductor Corporation, 2003. The information contained herein is subject to change without notice. Cypress Semiconductor Corporation assumes no responsibility for the use of any circuitry other than circuitry embodied in a Cypress Semiconductor product. Nor does it convey or imply any license under patent or other rights. Cypress Semiconductor does not authorize its products for use as critical components in life-support systems where a malfunction or failure may reasonably be expected to result in significant injury to the user. The inclusion of Cypress Semiconductor products in life-support systems application implies that the manufacturer assumes all risk of such use and in doing so indemnifies Cypress Semiconductor against all charges.

20.0 Package Diagram

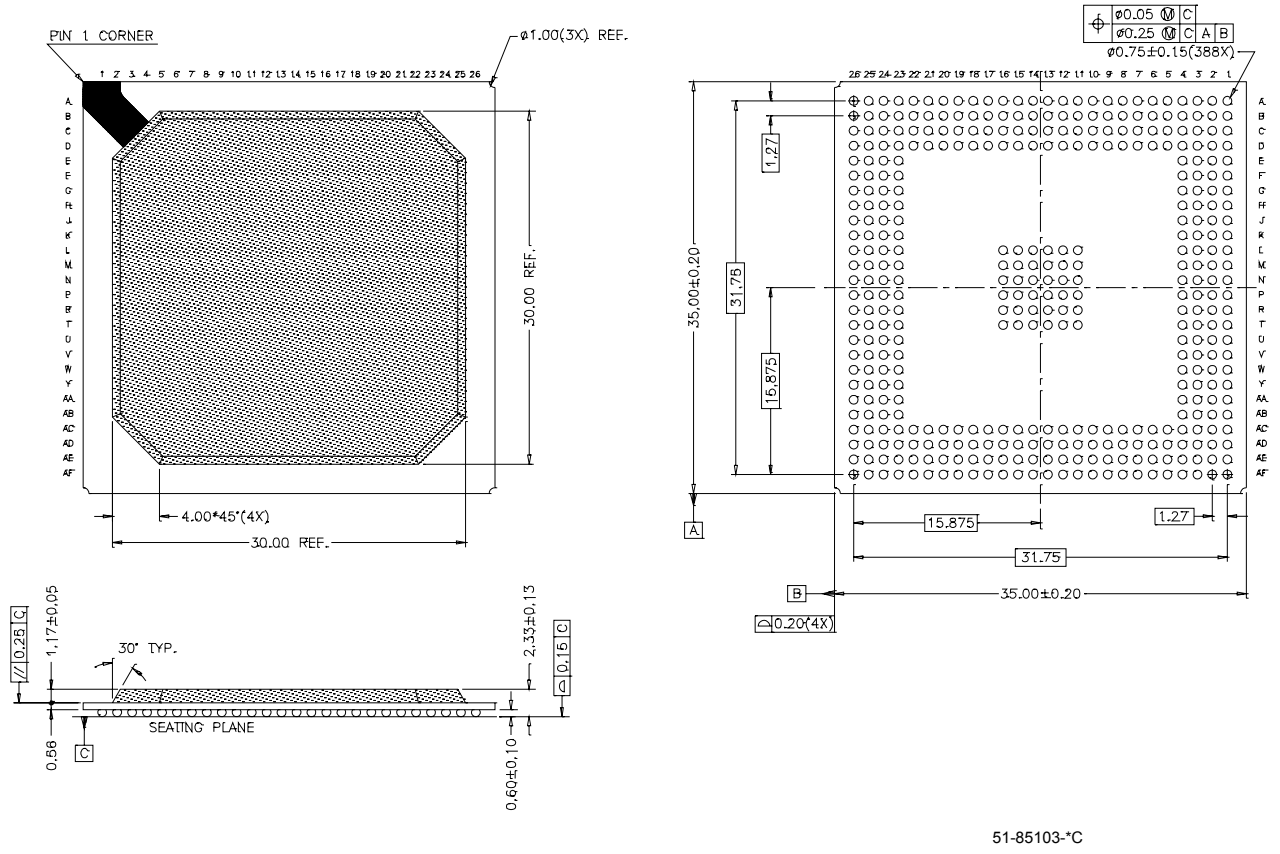


Figure 20-1. 388-lead Ball Grid Array (35 x 35 x 2.33 mm) BG388

APT is a trademark of Cypress Semiconductor. All product and company names mentioned in this document are the trademarks of their respective holders.

Document History Page

Document Title: CYNSE70128 Network Search Engine Document Number: 38-02040				
REV.	**ECN NO.**	**Issue Date**	**Orig. of Change**	**Description of Change**
**	111436	01/29/02	AFX	New Data Sheet
*A	116609	08/28/02	OOR	Updated AC Timing, DC Char, JTAG, Pinout Diagram and Pinout Description. Removed references to TEST signals from Pinout Diagram, Pinout Description and Signal Description. Added Power section covering power-up sequence and power consumption. Removed all references to 1.8V I/O. Removed all references to CLK_TUNE[3:0] and set it to 100% (“1001”).
*B	119295	12/16/02	ED	Added availability of 66- and 83-MHz Industrial parts. Removed CLK1X power-up sequence diagrams. Added Note to power-up sequence instructions. Added operating temperature range of Industrial parts. Added Minimum Output Data Hold (t_{CKHDZ} and t_{CKHSHZ}) of 0.5 ns. Changed Cascaded Input Hold Time (t_{ICHCH}) to 2.0 ns. Corrected Pinout Signal Name for AE26, AF2: V_{DD} to V_{SS}. Added Industrial parts ordering information. Removed Alternative power-up sequence instructions: TOC, *Figure 13-3*.
*C	123794	02/20/03	KOS	Added 3.3V to the I/O Voltage of CYNSE70128-83BGI ordering information. Corrected Section 10.6.10 Figure Number Reference from Figure 12-37 to *Figure 10-71*. Corrected Section 10.6.10 Figure Number Reference from Figure 12-38 to *Figure 10-72*. Changed Cascaded Input Hold Time (t_{ICHCH}) to 2.0ns. Changed Input Hold Time to CLK1X edge (t_{IHCH}) for CYNSE70128-66 to 0.6 ns. Changed Input Hold Time to CLK1X edge (t_{IHCH}) for CYNSE70128-83 to 0.6 ns. Changed Input Hold Time to CLK1X edge (t_{IHCH}) for CYNSE70128-100 to 0.5 ns. Added Input Hold Time to CLK2X edge (t_{IHCH}) AC characteristic for Industrial parts. Added Input Hold Time to CLK1X edge (t_{IHCH}) AC characteristic for Industrial parts.
*D	126021	05/08/03	ITL	Updated *Figure 13-1* on page 122 to reflect the correct waveforms. Also corrected the power-up sequence above the figure. Corrected the pin description of AE26 in the pin assignment table. The description is changed to “ground.”
*E	127445	06/25/03	DCU	Clarified description of HIGH_SPEED pin. Amended Learn description to include restriction at >83 MHz.
*F	313477	See ECN	AOG	Modified Section 3.1 Modified Figure 5-1 Added explanation in Section 5.0 Changed Section 10.5 on parallel write Added point 5 in Section 13.1 Power-up sequence Added Section 12.9 and Table 12-2 about idle cycle between successive operations.