



**THE DATASHEET OF  
CYFB0072V18L-133BGXI**



## 72-Mbit Video Frame Buffer

### Features

- Memory organization
  - Density: 72-Mbit
  - Organization: × 36
- Up to 133-MHz clock operation <sup>[1]</sup>
- Unidirectional operation
- Independent read and write ports
  - Supports simultaneous read and write operations
  - Reads and writes operate on independent clocks, upto a maximum ratio of two, enabling data buffering across clock domains.
  - Supports multiple I/O voltage standard: low voltage complementary metal oxide semiconductor (LVCMOS) 3.3 V and 1.8 V voltage standards.
- Input and output enable control for write mask and read skip operations
- Empty & Full status flags
- Flow-through mailbox register to send data from input to output port, bypassing the Frame Buffer
- Separate serial clock (SCLK) input for serial programming of configuration registers
- Master reset to clear entire Frame Buffer
- Partial reset to clear data but retain programmable settings
- Joint test action group (JTAG) port provided for boundary scan function
- Industrial temperature range: -40 °C to +85 °C

### Functional Description

The Video Frame Buffer is a 72-Mbit memory device which operates as a FIFO with a bus width of 36 bits. It has independent read and write ports, which can be clocked up to 133 MHz. The bus size of 36 bits enables a data throughput of 4.8 Gbps. The device also offers a simple and easy-to-use interface to reduce implementation and debugging efforts, improve time-to-market, and reduce engineering costs. This makes it an ideal memory choice for a wide range of applications including video and image processing or any system that needs buffering at high speeds across different clock domains.

The functionality of the Video Frame Buffer is such that the data is read out of the read port in the same sequence in which it was written into the write port. If writes and inputs are enabled ( $\overline{WEN}$  &  $\overline{IE}$ ), data on the write port gets written into the device at the rising edge of write clock. Enabling reads and outputs ( $\overline{REN}$  &  $\overline{OE}$ ) fetches data on the read port at every rising edge of read clock. Both reads and writes can occur simultaneously at different speeds provided the ratio between read and write clock is in the range of 0.5 to 2. Appropriate flags are set whenever the device is empty or full.

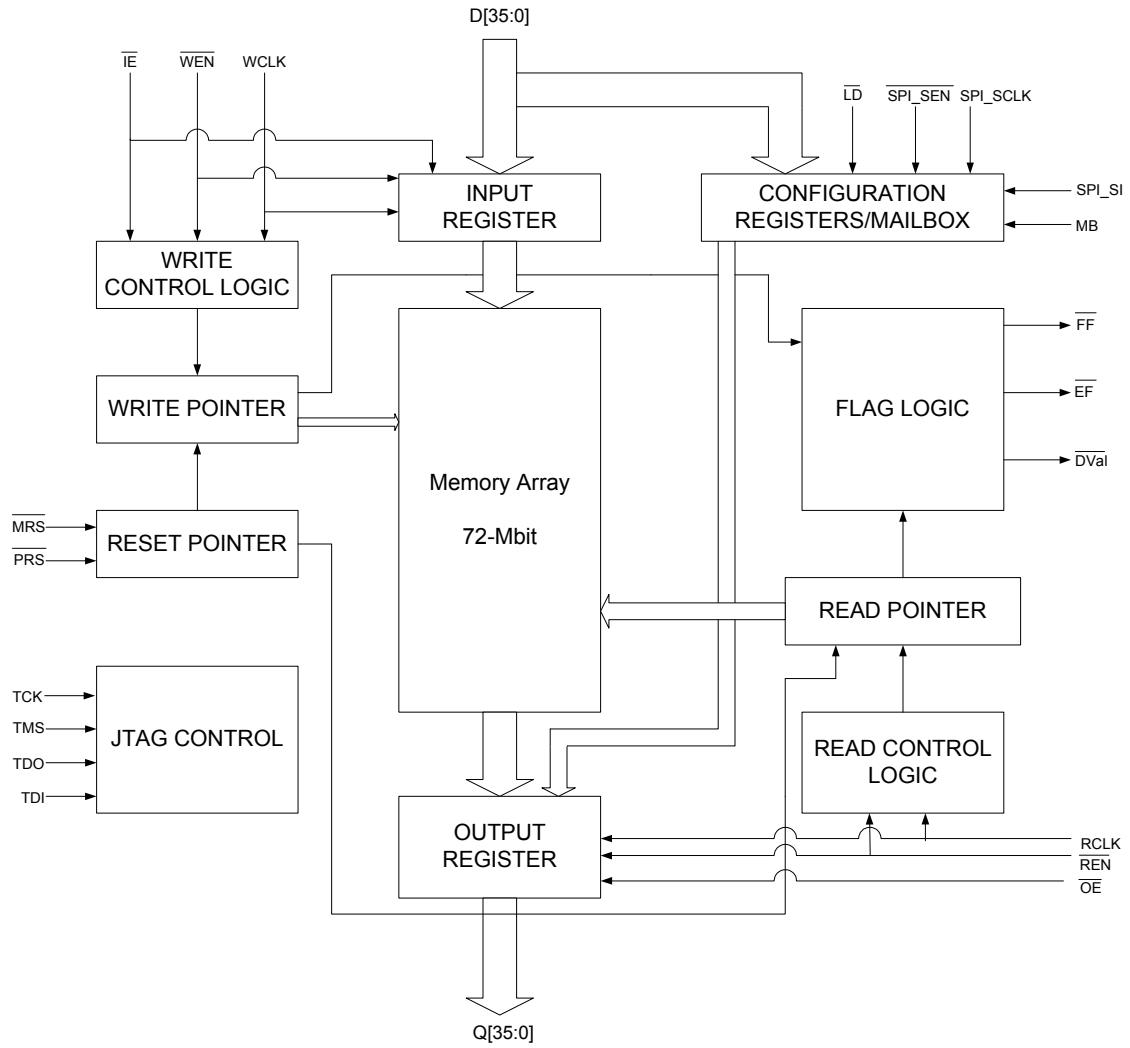
The device also supports a flow-through mailbox register to bypass the frame buffer memory

For a complete list of related documentation, [click here](#).

#### Note

1. For device operating at 150 MHz, Contact Sales.

**Logic Block Diagram**



## Contents

<b>Pin Configuration</b> .....	<b>4</b>	<b>Operating Range</b> .....	<b>14</b>
<b>Pin Definitions</b> .....	<b>5</b>	<b>Recommended DC Operating Conditions</b> .....	<b>14</b>
<b>Architecture</b> .....	<b>7</b>	<b>Electrical Characteristics</b> .....	<b>14</b>
Reset Logic .....	7	<b>I/O Characteristics</b> .....	<b>15</b>
Data Valid Signal (DVal) .....	7	<b>Latency Table</b> .....	<b>15</b>
Write Mask and Read Skip Operation .....	7	<b>AC Test Load Conditions</b> .....	<b>16</b>
Flow-through Mailbox Register .....	7	<b>Switching Characteristics</b> .....	<b>17</b>
Flag Operation.....	7	<b>Switching Waveforms</b> .....	<b>18</b>
Programming Configuration Registers .....	8	<b>Ordering Information</b> .....	<b>24</b>
Width Expansion Configuration .....	11	Ordering Code Definitions .....	24
Power Up .....	11	<b>Package Diagram</b> .....	<b>25</b>
Read/Write Clock Requirements .....	11	<b>Acronyms</b> .....	<b>26</b>
JTAG Operation .....	12	<b>Document Conventions</b> .....	<b>26</b>
Test Access Port .....	12	Units of Measure .....	26
Tap Registers .....	12	<b>Document History Page</b> .....	<b>27</b>
JTAG ID Codes .....	13	<b>Sales, Solutions, and Legal Information</b> .....	<b>28</b>
OPCODES Supported .....	13	Worldwide Sales and Design Support .....	28
JTAG Instructions .....	13	Products .....	28
Instruction Update and Bypass .....	13	PSoC® Solutions .....	28
TAP Controller State Diagram .....	13	Cypress Developer Community .....	28
<b>Maximum Ratings</b> .....	<b>14</b>	Technical Support .....	28

## Pin Configuration

**Figure 1. 209-ball FBGA pinout (Top View)**

	1	2	3	4	5	6	7	8	9	10	11
A	$\overline{FF}$	D0	D1	DNU	V <sub>PU</sub>	V <sub>PU</sub>	DNU	DNU	V <sub>PD</sub>	Q0	Q1
B	$\overline{EF}$	D2	D3	DNU	DNU	V <sub>PU</sub>	DNU	DNU	$\overline{REN}$	Q2	Q3
C	D4	D5	$\overline{WEN}$	DNU	V <sub>CC1</sub>	DNU	V <sub>CC1</sub>	DNU	RCLK	Q4	Q5
D	D6	D7	V <sub>SS</sub>	V <sub>CC1</sub>	DNU	$\overline{LD}$	DNU	V <sub>CC1</sub>	V <sub>SS</sub>	Q6	Q7
E	D8	D9	V <sub>CC2</sub>	V <sub>CC2</sub>	V <sub>CCIO</sub>	V <sub>CCIO</sub>	V <sub>CCIO</sub>	V <sub>CC2</sub>	V <sub>CC2</sub>	Q8	Q9
F	D10	D11	V <sub>SS</sub>	V <sub>SS</sub>	V <sub>SS</sub>	DNU	V <sub>SS</sub>	V <sub>SS</sub>	V <sub>SS</sub>	Q10	Q11
G	D12	D13	V <sub>CC2</sub>	V <sub>CC2</sub>	V <sub>CCIO</sub>	V <sub>CC1</sub>	V <sub>CCIO</sub>	V <sub>CC2</sub>	V <sub>CC2</sub>	Q12	Q13
H	D14	D15	V <sub>SS</sub>	V <sub>SS</sub>	V <sub>SS</sub>	V <sub>CC1</sub>	V <sub>SS</sub>	V <sub>SS</sub>	V <sub>SS</sub>	Q14	Q15
J	D16	D17	V <sub>CC2</sub>	V <sub>CC2</sub>	V <sub>CCIO</sub>	V <sub>CC1</sub>	V <sub>CCIO</sub>	V <sub>CC2</sub>	V <sub>CC2</sub>	Q16	Q17
K	DNU	DNU	WCLK	DNU	V <sub>SS</sub>	$\overline{IE}$	V <sub>SS</sub>	DNU	V <sub>CCIO</sub>	V <sub>CCIO</sub>	V <sub>CCIO</sub>
L	D18	D19	V <sub>CC2</sub>	V <sub>CC2</sub>	V <sub>CCIO</sub>	V <sub>CC1</sub>	V <sub>CCIO</sub>	V <sub>CC2</sub>	V <sub>CC2</sub>	Q18	Q19
M	D20	D21	V <sub>SS</sub>	V <sub>SS</sub>	V <sub>SS</sub>	V <sub>CC1</sub>	V <sub>SS</sub>	V <sub>SS</sub>	V <sub>SS</sub>	Q20	Q21
N	D22	D23	V <sub>CC2</sub>	V <sub>CC2</sub>	V <sub>CCIO</sub>	V <sub>CC1</sub>	V <sub>CCIO</sub>	V <sub>CC2</sub>	V <sub>CC2</sub>	Q22	Q23
P	D24	D25	V <sub>SS</sub>	V <sub>SS</sub>	V <sub>SS</sub>	$\overline{SPI\_SEN}$	V <sub>SS</sub>	V <sub>SS</sub>	V <sub>SS</sub>	Q24	Q25
R	D26	D27	V <sub>CC2</sub>	V <sub>CC2</sub>	V <sub>CCIO</sub>	V <sub>CCIO</sub>	V <sub>CCIO</sub>	V <sub>CC2</sub>	V <sub>CC2</sub>	Q26	Q27
T	D28	D29	V <sub>SS</sub>	V <sub>CC1</sub>	V <sub>CC1</sub>	SPI_SI	V <sub>CC1</sub>	V <sub>CC1</sub>	V <sub>SS</sub>	Q28	Q29
U	$\overline{DVal}$	DNU	D30	D31	$\overline{PRS}$	DNU <sup>[2]</sup>	SPI_SCLK	V <sub>REF</sub>	$\overline{OE}$	Q30	Q31
V	DNU	DNU	D32	D33	DNU	$\overline{MRS}$	MB	DNU	V <sub>PD</sub>	Q32	Q33
W	TDO	DNU	D34	D35	TDI	DNU	TMS	TCK	DNU	Q34	Q35

**Notes**

- This pin should be tied to V<sub>SS</sub> preferably or can be left floating to ensure normal operation.

## Pin Definitions

Pin Name	I/O	Pin Description
$\overline{\text{MRS}}$	Input	Master reset: $\overline{\text{MRS}}$ initializes the internal read and write pointers to zero, resets both flags and sets the output register to all zeroes. During Master Reset, the configuration registers are set to default values.
$\overline{\text{PRS}}$	Input	Partial reset: $\overline{\text{PRS}}$ initializes the internal read and write pointers to zero, resets both flags and sets the output register to all zeroes. During Partial Reset, the configuration register settings are retained.
WCLK	Input	Write clock: The rising edge clocks data into the frame buffer when writes are enabled ( $\overline{\text{WEN}}$ asserted). Data is written into the buffer memory when LD is high and into configuration registers when LD is low.
$\overline{\text{LD}}$	Input	Load: When $\overline{\text{LD}}$ is LOW, D[7:0] (Q[7:0]) are written (read) into (from) the configuration registers. When LD is HIGH, D[35:0] (Q[35:0]) are written (read) into (from) the buffer memory.
$\overline{\text{WEN}}$	Input	Write enable: Control signal to enable writes to the device. When $\overline{\text{WEN}}$ is low data present on the inputs is written to the buffer memory or configuration registers on every rising edge of WCLK.
$\overline{\text{IE}}$	Input	Input enable: $\overline{\text{IE}}$ is the data input enable signal that controls the enabling and disabling of the 36-bit data input pins. If it is enabled, data on input pins is written into the frame buffer memory or configuration registers. The internal write address pointer is always incremented at rising edge of WCLK if WEN is enabled, regardless of the $\overline{\text{IE}}$ level. This is used for 'write masking' or incrementing the write pointer without writing into a location.
D[35:0]	Input	Data inputs: Data inputs for a 36-bit bus.
RCLK	Input	Read clock: The rising edge initiates a read from the frame buffer when reads are enabled ( $\overline{\text{REN}}$ asserted). Data is read from the buffer memory when LD is high & from the configuration registers if LD is low.
$\overline{\text{REN}}$	Input	Read enable: Control signal to enable reads from the device. When $\overline{\text{REN}}$ is low data is read from the buffer memory or configuration registers on every rising edge of RCLK.
$\overline{\text{OE}}$	Input	Output enable: When $\overline{\text{OE}}$ is LOW, device data outputs are enabled; when $\overline{\text{OE}}$ is HIGH, the device's outputs are in High Z (high impedance) state.
Q[35:0]	Output	Data outputs: Data outputs for a 36-bit bus.
$\overline{\text{DVal}}$	Output	Data valid: Active low data valid signal to indicate valid data on Q[35:0].
MB	Input	Mailbox: When asserted the reads and writes happen to flow-through mailbox register.
$\overline{\text{EF}}$	Output	Empty flag: When $\overline{\text{EF}}$ is LOW, the frame buffer is empty. $\overline{\text{EF}}$ is synchronized to RCLK.
$\overline{\text{FF}}$	Output	Full flag: When $\overline{\text{FF}}$ is LOW, the frame buffer is full. $\overline{\text{FF}}$ is synchronized to WCLK.
SPI_SCLK	Input	Serial clock: A rising edge on SPI_SCLK clocks the serial data present on the SPI_SI input into the configuration registers if SPI_SEN is enabled.
SPI_SI	Input	Serial input: Serial input data in SPI mode.
$\overline{\text{SPI_SEN}}$	Input	Serial enable: Enables serial loading of configuration registers.
TCK	Input	Test clock (TCK) pin for JTAG.
TMS	Input	Test mode select (TMS) pin for JTAG.
TDI	Input	Test data in (TDI) pin for JTAG.
TDO	Output	Test data out (TDO) pin for JTAG.
V <sub>REF</sub>	Input Reference	Reference voltage: Reference voltage (regardless of I/O standard used)
V <sub>CC1</sub>	Power Supply	Core voltage supply 1: 1.8 V supply voltage
V <sub>CC2</sub>	Power Supply	Core voltage supply 2: 1.5 V supply voltage

**Pin Definitions** (continued)

Pin Name	I/O	Pin Description
V <sub>CCIO</sub>	Power Supply	Supply for I/Os
V <sub>SS</sub> <sup>[3]</sup>	Ground	Ground
V <sub>PU</sub>	CMOS Voltage Level	The pins A5, A6 and B6 of the FBGA Pins are required to be Pulled Up to the CMOS voltage level. <u>These pins should be powered up post the power supply V<sub>CC1</sub> &amp; V<sub>CC2</sub>, and should be stable prior MRST operation.</u>
DNU	–	Do not use: These pins need to be left floating.
V <sub>PD</sub> <sup>[3]</sup>	Input	Connect to GND (Short to V <sub>SS</sub> ).

**Note**

- 3. All V<sub>SS</sub> pins should be connected to the same ground plane.

## Architecture

The video frame buffer consists of a memory array of 72-Mbit along with the logic blocks to implement FIFO functionality and its associated features that are built around this memory array.

The input and output data buses have a maximum width of 36 bits. The input data bus goes to an input register and the data flow from the input register to the memory is controlled by the write control logic. The inputs to the write logic block are WCLK, WEN and IE. When the writes are enabled through WEN and if the inputs are enabled by IE, then the data on the input bus is written into the memory array at the rising edge of WCLK. This also increments the write pointer. Enabling writes but disabling the data input pins through IE only, increments the write pointer without doing any writes or altering the contents of the memory location.

Similarly, the output register is connected to the data output bus. Transfer of contents from the memory to the output register is controlled by the read control logic. The inputs to the read control logic include RCLK, REN, OE. When reads are enabled by REN and outputs are enabled using OE, the data from the memory pointed by the read pointer is transferred to the output data bus at the rising edge of RCLK along with active low DVal. If the outputs are disabled but the reads enabled, the outputs are in high impedance state, but internally the read pointer is incremented.

During write operation, the number of writes performed is always an even number (i.e., minimum write burst length is two and number of writes always a multiple of two), whereas during read operation, the number of reads performed can be even or odd (i.e., minimum read burst length is one).

### Reset Logic

The frame buffer can be reset in two ways: Master Reset (MRS) and Partial Reset (PRS). The MRS initializes the read and write pointers to zero and sets the output register to all zeroes. It also resets empty flag, full flag & the configuration registers to their default values. A Master Reset is required after power-up before accessing the frame buffer.

PRS resets the read pointer, write pointer to the first physical location in the memory array. It also resets the flags to their default values. PRS does not affect the programmed configuration register values.

### Data Valid Signal (DVal)

Data valid (DVal) is an active low signal, synchronized to RCLK and is provided to check for valid data on the output bus. When a read operation is performed, the DVal signal goes low along with output data. This helps user to capture the data without keeping track of REN to data output latency. This signal also helps when write and read operations are performed continuously at different frequencies by indicating when valid data is available at the output port Q[35:0].

### Write Mask and Read Skip Operation

As mentioned in [Architecture on page 7](#), enabling writes but disabling the inputs (IE HIGH) increments the write pointer without doing any write operations or altering the contents of the location.

This feature is called Write Mask and allows user to move the write pointer without actually writing to the locations. This “write masking” ability is useful in some video applications such as Picture In Picture (PIP).

Similarly, during a read operation, if the outputs are disabled by keeping the OE high, the read data does not appear on the output bus; however, the read pointer is incremented. This feature is referred to as a Read Skip Operation.

### Flow-through Mailbox Register

This feature transfers data from input to output directly bypassing the sequential buffer memory. When MB signal is asserted the data present in D[35:0] will be available at Q[35:0] after two WCLK cycles. Normal read and write operations are not allowed during flow-through mailbox operation. Before starting Flow-through mailbox operation reads should be completed to make data valid DVal high to avoid data loss from buffer memory.

### Flag Operation

This device provides two flag pins to indicate the condition of the video frame buffer.

#### Full Flag

The Full Flag (FF) operates on double word (burst length of two) boundaries and goes LOW when the device is full. Write operations are inhibited whenever FF is LOW regardless of the state of WEN. FF is synchronized to WCLK, that is, it is exclusively updated by each rising edge of WCLK. The worst case assertion latency for Full Flag is four. As the user cannot know that the frame buffer is full for four clock cycles, it is possible that user continues writing data during this time. In this case, the four data words written will be stored to prevent data loss and these words have to be read back in order for full flag to get de-asserted. The minimum number of reads required to de-assert full-flag is two and the maximum number of reads required to de-assert full flag is six. The latency associated with Full flag is explained in [Latency Table on page 15](#).

#### Empty Flag

The Empty Flag (EF) deassertion depends on burst writes and goes LOW when the device is empty. Read operations are inhibited whenever EF is LOW, regardless of the state of REN. EF is synchronized to RCLK, that is, it is exclusively updated by each rising edge of RCLK. The latency associated with Empty flag is explained in [Latency Table on page 15](#).

### Programming Configuration Registers

The CYFB0072V has ten 8-bit user configurable registers. The tenth register is the Fast CLK bit which indicates the faster clock domain.

This register can be programmed in one of two ways: serial loading or parallel loading method. The loading method is selected using the SPI\_SEN (Serial Enable) pin. A low on the SPI\_SEN selects the serial method for writing into the register. For serial programming, there is a separate SCLK and a Serial Input (SI). In parallel mode, a LOW on the load ( $\overline{LD}$ ) pin causes the write and read operation to these registers. When  $\overline{LD}$  is held LOW, write and read operations happen sequentially from the

first location (0x1) to the last location (0xA). If  $\overline{LD}$  is HIGH, the writes occur to the FIFO.

Register values can be read through the parallel output port regardless of the programming mode selected (serial or parallel). Register values cannot be read serially. The registers may be programmed (and reprogrammed) any time after master reset, regardless of whether serial or parallel programming is selected.

See [Table 1](#) and [Table 2](#) on [page 9](#) for access to configuration registers in serial and parallel modes.

In parallel mode, the read and write operations loop back when the maximum address location of the configuration registers is reached. Simultaneous read and write operations should be avoided on the configuration registers.

**Table 1. Configuration Registers**

ADDR	Configuration Register	Default	Bit [7]	Bit [6]	Bit [5]	Bit [4]	Bit [3]	Bit [2]	Bit [1]	Bit [0]
0x1	Reserved	0x00	X	X	X	X	X	X	X	X
0x2	Reserved	0x00	X	X	X	X	X	X	X	X
0x3	Reserved	0x00	X	X	X	X	X	X	X	X
0x4	Reserved	0x7F	X	X	X	X	X	X	X	X
0x5	Reserved	0x00	X	X	X	X	X	X	X	X
0x6	Reserved	0x00	X	X	X	X	X	X	X	X
0x7	Reserved	0x7F	X	X	X	X	X	X	X	X
0x8	Reserved	0x00	X	X	X	X	X	X	X	X
0x9	Reserved	0x00	X	X	X	X	X	X	X	X
0xA	Fast CLK Bit Register	1XXXXXXXXb	Fast CLK bit	X	X	X	X	X	X	X

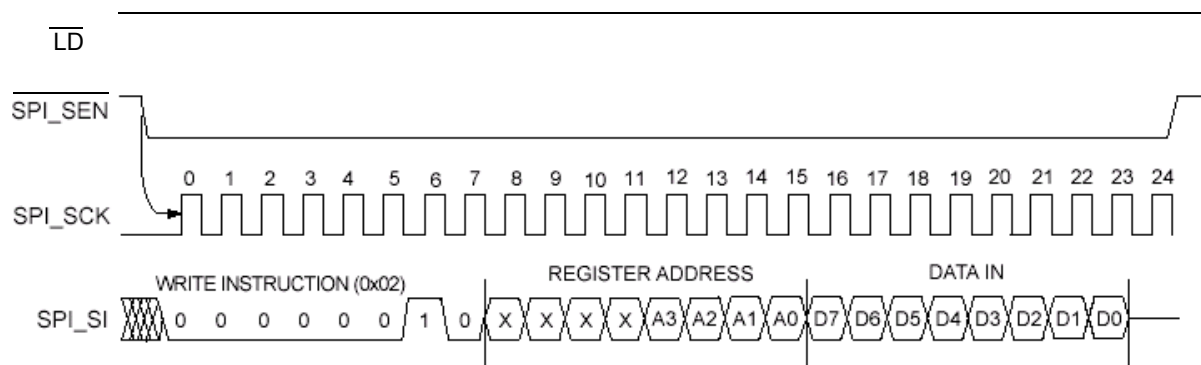
**Table 2. Writing and Reading Configuration Registers in Parallel Mode**

SPI_SEN	LD	WEN	REN	WCLK	RCLK	SPI_SCLK	Operation
1	0	0	1	↑ First rising edge because both LD and WEN are low	X	X	Parallel write to first register
1	0	0	1	↑ Second rising edge	X	X	Parallel write to second register
1	0	0	1	↑ Third rising edge	X	X	Parallel write to third register
1	0	0	1	↑ Fourth rising edge	X	X	Parallel write to fourth register
1	0	0	1	•	X	X	•
1	0	0	1	•	X	X	•
1	0	0	1	•	X	X	•
1	0	0	1	↑ Tenth rising edge	X	X	Parallel write to tenth register
1	0	0	1	↑ Eleventh rising edge	X	X	Parallel write to first register (roll back)
1	0	1	0	X	↑ First rising edge since both LD and REN are low	X	Parallel read from first register
1	0	1	0	X	↑ Second rising edge	X	Parallel read from second register
1	0	1	0	X	↑ Third rising edge	X	Parallel read from third register
1	0	1	0	X	↑ Fourth rising edge	X	Parallel read from fourth register
1	0	1	0	X	•	X	•
1	0	1	0	X	•	X	•
1	0	1	0	X	•	X	•
1	0	1	0	X	↑ Tenth rising edge	X	Parallel read from tenth register
1	0	1	0	X	↑ Eleventh rising edge	X	Parallel read from first register (roll back)
1	X	1	1	X	X	X	No operation
X	1	0	X	↑ Rising edge	X	X	Write to Frame Buffer memory
X	1	X	0	X	↑ Rising edge	X	Read from Frame Buffer memory
0	0	X	1	X	X	X	Illegal operation

Table 3. Writing into Configuration Registers in Serial Mode

SPI_SEN	LD	WEN	REN	WCLK	RCLK	SCLK	Operation
0	1	X	X	X	X	↑ Rising edge	Each rising of the SCLK clocks in one bit from the SI (Serial In). Any of the 10 registers can be addressed and written to, following the SPI protocol.
X	1	0	X	↑ Rising edge	X	X	Parallel write to Frame Buffer memory.
X	1	X	0	X	↑ Rising edge	X	Parallel read from Frame Buffer memory.
1	0	1	1	X	X	X	This corresponds to parallel mode (refer to <a href="#">Table 2 on page 9</a> ).

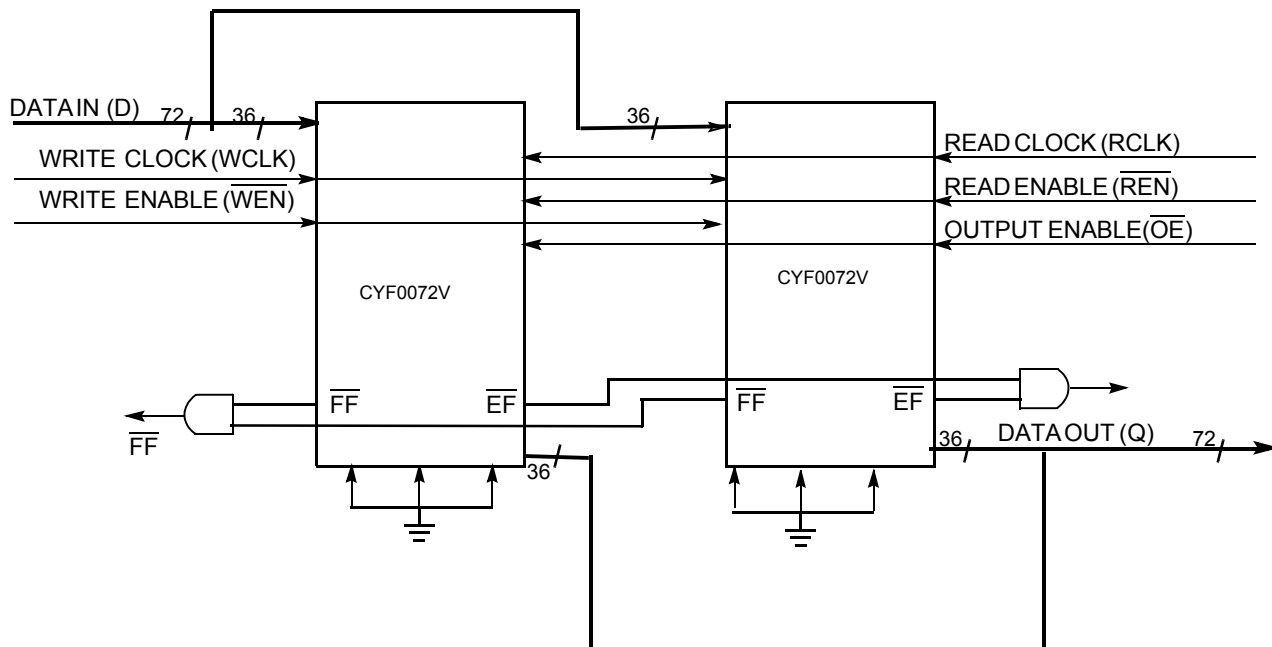
Figure 2. Serial WRITE to Configuration Register



### Width Expansion Configuration

The width of the frame buffer can be expanded to provide word widths greater than 36 bits. During width expansion mode, all control line inputs are common and all flags are available. Empty (Full) flags are created by ANDing the Empty (Full) flags of every Frame Buffer. This technique avoids reading data from or writing data to the device that is “staggered” by one clock cycle due to the variations in skew between RCLK and WCLK. Figure 3 demonstrates an example of 72 bit-word width by using two 36-bit word frame buffers.

Figure 3. Width Expansion



### Power Up

The device becomes functional after  $V_{CC1}$ ,  $V_{CC2}$ ,  $V_{CCIO}$ , and  $V_{REF}$  attain minimum stable voltage required as given in [Recommended DC Operating Conditions on page 14](#). The device can be accessed  $t_{PU}$  time after these supplies attain the minimum required level (see [Switching Characteristics on page 17](#)). There is no specific power sequencing required for the device.

### Read/Write Clock Requirements

The read and write clocks must satisfy the following requirements:

- Both read (RCLK) and write (WCLK) clocks should be free-running.
- The clock frequency for both clocks should be between the minimum and maximum range given in [Electrical Characteristics on page 14](#).
- The WCLK to RCLK ratio should be in the range of 0.5 to 2.

For proper frame buffer operation, the device must determine which of the input clocks – RCLK or WCLK – is faster. This is evaluated using counters after the MRS cycle. The device uses two 9-bit counters (one running on RCLK and other on WCLK), which count 256 cycles of read and write clocks after MRS. The clock of the counter which reaches its terminal count first is used as master clock inside the frame buffer.

When there is change in the relative frequency of RCLK and WCLK during normal operation of Frame Buffer, user can specify it by using “Fast CLK bit” in the configuration register (0xA).

“1” - indicates  $f_{req}(WCLK) > f_{req}(RCLK)$

“0” - indicates  $f_{req}(WCLK) < f_{req}(RCLK)$

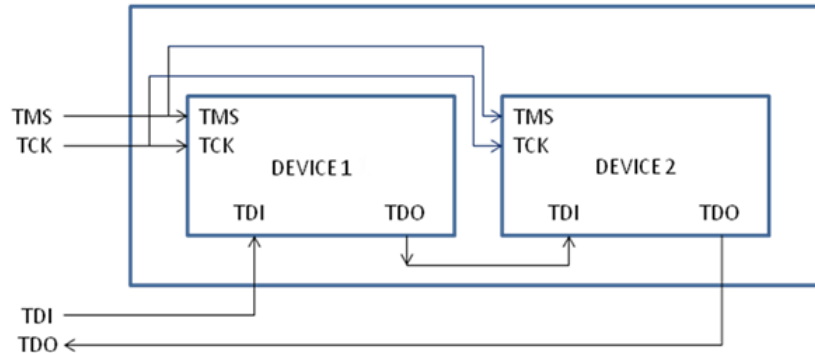
The fast clock bit configuration register(0xA), can be accessed by keeping  $\overline{LD}$  low for 10 clock cycles. The result of counter evaluated frequency is available in this register bit. User can override the counter evaluated frequency for faster clock by changing this bit.

Whenever there is a change in this bit value, user must wait  $t_{PLL}$  time before issuing the next read or write to buffer memory.

## JTAG Operation

The video frame buffer has two devices connected internally in a JTAG chain, as shown in figure [Figure 4](#).

**Figure 4. Device Connection in a JTAG Chain**



### Test Access Port

#### Test Clock (TCK)

The test clock is used only with the TAP controller. All inputs are captured on the rising edge of TCK. All outputs are driven on the falling edge of TCK.

#### Test Mode Select (TMS)

The TMS input is used to give commands to the TAP controller and is sampled on the rising edge of TCK. This pin may be left unconnected if the TAP is not used. The pin is pulled up internally, resulting in a logic HIGH level.

#### Test Data-In (TDI)

The TDI pin is used to serially input information into the registers and can be connected to the input of any of the registers. The register between TDI and TDO is chosen by the instruction that is loaded into the TAP instruction register. For information on loading the instruction register, see [Figure 5](#). TDI is internally pulled up and can be left unconnected if the TAP is unused in an application. TDI is connected to the most significant bit (MSB) on any of the TAP registers.

#### Test Data-Out (TDO)

The TDO output pin is used to serially clock data out from the registers. The output is active, depending upon the current state of the TAP state machine. The output changes on the falling edge of TCK. TDO is connected to the least significant bit (LSB) of any of the TAP registers.

**Note:** Reset is performed by forcing TMS HIGH ( $V_{DD}$ ) for five rising edges of TCK.

### Tap Registers

Registers are connected between the TDI and TDO pins to scan the data in and out of the test circuitry. Only one register can be selected at a time through the instruction registers. Data is serially loaded into the TDI pin on the rising edge of TCK. Data is output on the TDO pin on the falling edge of TCK.

#### Instruction Register

Three-bit instructions can be serially loaded into the instruction register. This register is loaded when it is placed between the TDI and TDO pins, as shown in [Figure 5](#). Upon power-up, the instruction register is loaded with the IDCODE instruction. It is also loaded with the IDCODE instruction if the controller is placed in a Reset state.

#### Bypass Register

To save time when serially shifting data through registers, it is sometimes advantageous to skip certain chips. The bypass register is a single-bit register that can be placed between TDI and TDO pins. This enables shifting of data through the device with minimal delay.

#### Boundary Scan Register

The boundary scan register is connected to all of the input and output pins on the device. Several No Connect (NC) pins are also included in the scan register to reserve pins for higher density devices.

The boundary scan register is loaded with the contents of the device input and output ring when the TAP controller is in the Capture-DR state and is then placed between the TDI and TDO pins when the controller is moved to the Shift-DR state. The EXTEST, SAMPLE/PRELOAD instructions can be used to capture the contents of the input and output ring. The MSB of the register is connected to TDI and the LSB is connected to TDO.

#### Identification (ID) Register

The ID register is loaded with a vendor-specific, 32-bit code during the Capture-DR state when the IDCODE command is loaded in the instruction register. The IDCODE is hardwired into the device and can be shifted out when the TAP controller is in the Shift-DR state. The ID register has a vendor code and other information described in [Table 4](#).

**JTAG ID Codes**

**Table 4. JTAG IDCODES**

	IR Register Length	Device ID (HEX)	Bypass Register Length
Device-1	3	“Ignore”	1
Device-2	8	1E3261CF	1

**OPCODES Supported**

**Table 5. OPCODES Supported**

Device-1	Opcode (Binary)	Device-2	Opcode (Binary)
BYPASS	111	BYPASS	11111111
		EXTEST	00000000
		HIGHZ	00000111
		SAMPLE/PRELOAD	00000001
		IDCODE	00001111

**JTAG Instructions**

*IDCODE*

The IDCODE instruction loads a vendor-specific, 32-bit code into the instruction register. It also places the instruction register between the TDI and TDO pins and shifts the IDCODE out of the device when the TAP controller enters the Shift-DR state. The IDCODE instruction is loaded into the instruction register at power-up or whenever the TAP controller is supplied a Test-Logic-RST state.

*SAMPLE/PRELOAD*

SAMPLE/PRELOAD is an IEEE 1149.1 mandatory instruction. When the SAMPLE/PRELOAD instructions are loaded into the instruction register and the TAP controller is in the Capture-DR state, a snapshot of data on the input and output pins is captured in the boundary scan register.

PRELOAD places an initial data pattern at the latched parallel outputs of the boundary scan register cells before the selection of another boundary scan test operation.

The shifting of data for the SAMPLE and PRELOAD phases can occur concurrently when required; that is, while the data captured is shifted out, the preloaded data can be shifted in.

*BYPASS*

When the BYPASS instruction is loaded in the instruction register and the TAP is placed in a Shift-DR state, the bypass register is placed between the TDI and TDO pins. The advantage of the BYPASS instruction is that it shortens the boundary scan path when multiple devices are connected together on a board.

*HIGHZ*

The HIGHZ instruction mode is used to set all the user I/O pins to an inactive drive state. These pins are tri-stated until a new JATG instruction is executed. When this instruction is selected, the bypass register is connected between the TDI and TDO ports.

*EXTEST*

The EXTEST instruction drives the preloaded data out through the system output pins. This instruction also connects the

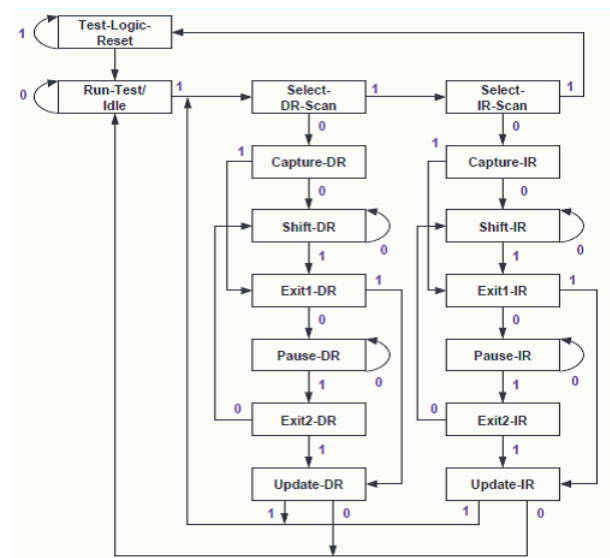
boundary scan register for serial access between the TDI and TDO in the Shift-DR controller state.

**Instruction Update and Bypass**

1. Every time an instruction is loaded through JTAG port, BYPASS command needs to be loaded on device 1. For example, to push PRELOAD command, BYPASS to device-1 “111” + PRELOAD to device-2 “00000001” needs to be sent.
2. When both devices are put on BYPASS, any pattern sent in should be observed on TDO after two TCK delay.

**TAP Controller State Diagram**

**Figure 5. TAP Controller State Diagram**



TAP controller is a Finite State Machine with 16 states as shown in Figure 5. State change is determined by the state of TMS on rising edge of TCK. Figure 5 shows the value of TMS for each state transition.

## Maximum Ratings

Exceeding maximum ratings may shorten the useful life of the device. These user guidelines are not tested.

Storage temperature (without bias) ..... -65 °C to +150 °C

Ambient temperature with power applied ..... -55 °C to +125 °C

Core supply voltage 1 ( $V_{CC1}$ ) to ground potential ..... -0.3 V to 2.5 V

Core supply voltage 2 ( $V_{CC2}$ ) to ground potential ..... -0.3 V to 1.65 V

Latch up current ..... >100 mA

I/O port supply voltage ( $V_{CCIO}$ ) ..... -0.3 V to 3.7 V

Voltage applied to I/O pins ..... -0.3 V to 3.75 V

Output current into outputs (LOW) ..... 24 mA

Static discharge voltage (per MIL-STD-883, Method 3015) ..... > 2001 V

## Operating Range

Range	Ambient Temperature
Industrial	-40 °C to +85 °C

## Recommended DC Operating Conditions

Parameter <sup>[4]</sup>	Description	Min	Typ	Max	Unit	
$V_{CC1}$	Core supply voltage 1	1.70	1.80	1.90	V	
$V_{CC2}$	Core supply voltage 2	1.425	1.5	1.575	V	
$V_{REF}$	Reference voltage (irrespective of I/O standard used)	0.7	0.75	0.8	V	
$V_{PU}$	Input CMOS Voltage level for the Frame Buffer	LVC MOS33	3.00	3.30	3.60	V
		LVC MOS18	1.70	1.8	1.90	V
$V_{CCIO}$	I/O supply voltage, read and write banks.	LVC MOS33	3.00	3.30	3.60	V
		LVC MOS18	1.70	1.8	1.90	V

## Electrical Characteristics

Parameter	Description	Conditions	Min	Typ	Max	Unit
$I_{CC}$	Active current	$V_{CC1} = V_{CC1MAX}$	-	-	300	mA
		$V_{CC2} = V_{CC2MAX}$ (All I/O switching, 133 MHz)	-	-	600	mA
		$V_{CCIO} = V_{CCIO MAX}$ (All outputs disabled)	-	-	100	mA
$I_I$	Input pin leakage current	$V_{IN} = V_{CCIO max}$ to 0 V	-15	-	15	μA
$I_{OZ}$	I/O pin leakage current	$V_O = V_{CCIO max}$ to 0 V	-15	-	15	μA
$C_P$	Capacitance for TMS and TCK	-	-	-	16	pF
$C_{PIO}$	Capacitance for all other pins except TMS and TCK	-	-	-	8	pF

### Note

4. Device operation guaranteed for a supply rate > 1 V / μs.

## I/O Characteristics

(Over the operating range)

I/O Standard	Nominal I/O Supply Voltage	Input Voltage (V)		Output Voltage (V)		Output Current (mA)	
		V <sub>IL</sub> (max)	V <sub>IH</sub> (min)	V <sub>OL</sub> (max)	V <sub>OH</sub> (min)	I <sub>OL</sub> (max)	I <sub>OH</sub> (max)
LVC MOS33	3.3 V	0.80	2.20	0.45	2.40	24	24
LVC MOS18	1.8 V	30% V <sub>CCIO</sub>	65% V <sub>CCIO</sub>	0.45	V <sub>CCIO</sub> - 0.45	16	16

## Latency Table

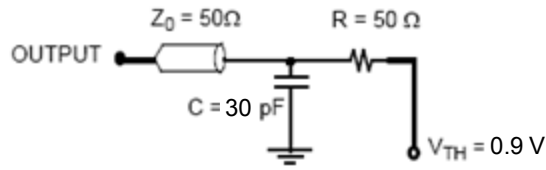
Latency Parameter	Number of Cycles	Details
L <sub>REN_TO_DATA</sub>	4	Latency when $\overline{\text{REN}}$ is asserted low to first data output from Frame Buffer.
L <sub>REN_TO_CONFIG</sub>	4	Latency when $\overline{\text{REN}}$ is asserted along with $\overline{\text{LD}}$ to first data read from configuration registers.
L <sub>IN</sub>	Max = 26 <sup>[5]</sup>	Initial latency for data read after Frame Buffer goes empty during simultaneous read/write.
L <sub>FF_ASSERT</sub>	Max = 4	Last data write to $\overline{\text{FF}}$ going low.
L <sub>EF_ASSERT</sub>	0	Last data read to $\overline{\text{EF}}$ going low.
L <sub>FF_DEASSERT</sub>	8 <sup>[5]</sup>	Read to $\overline{\text{FF}}$ going high.
L <sub>EF_DEASSERT</sub>	Max = 24 <sup>[5]</sup>	Write to $\overline{\text{EF}}$ going high.
L <sub>PRS_TO_ACTIVE</sub>	32 <sup>[5]</sup>	$\overline{\text{PRS}}$ de-assert to normal operation.
L <sub>MAILBOX</sub>	2	Latency from write port to read port when MB = 1 (w.r.t. WCLK).

### Note

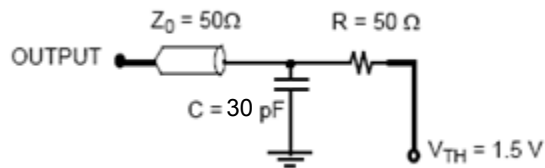
5. These latency values are valid for a clock ratio of 1.

**AC Test Load Conditions**

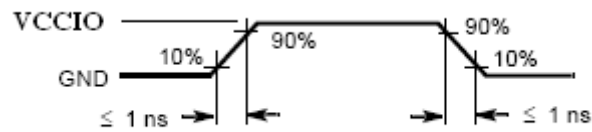
**Figure 6. AC Test Load Conditions**



(a)  $V_{CCIO} = 1.8$  Volt



(b)  $V_{CCIO} = 3.3$  Volt



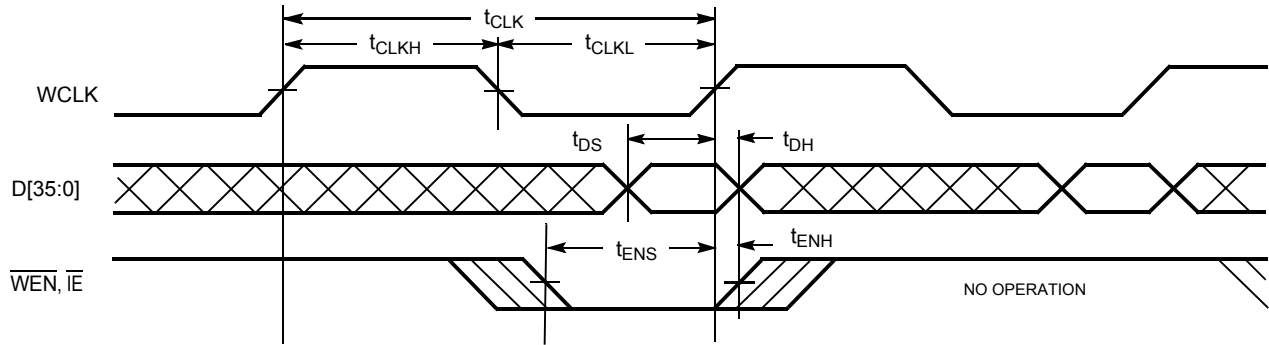
(c) All Input Pulses

## Switching Characteristics

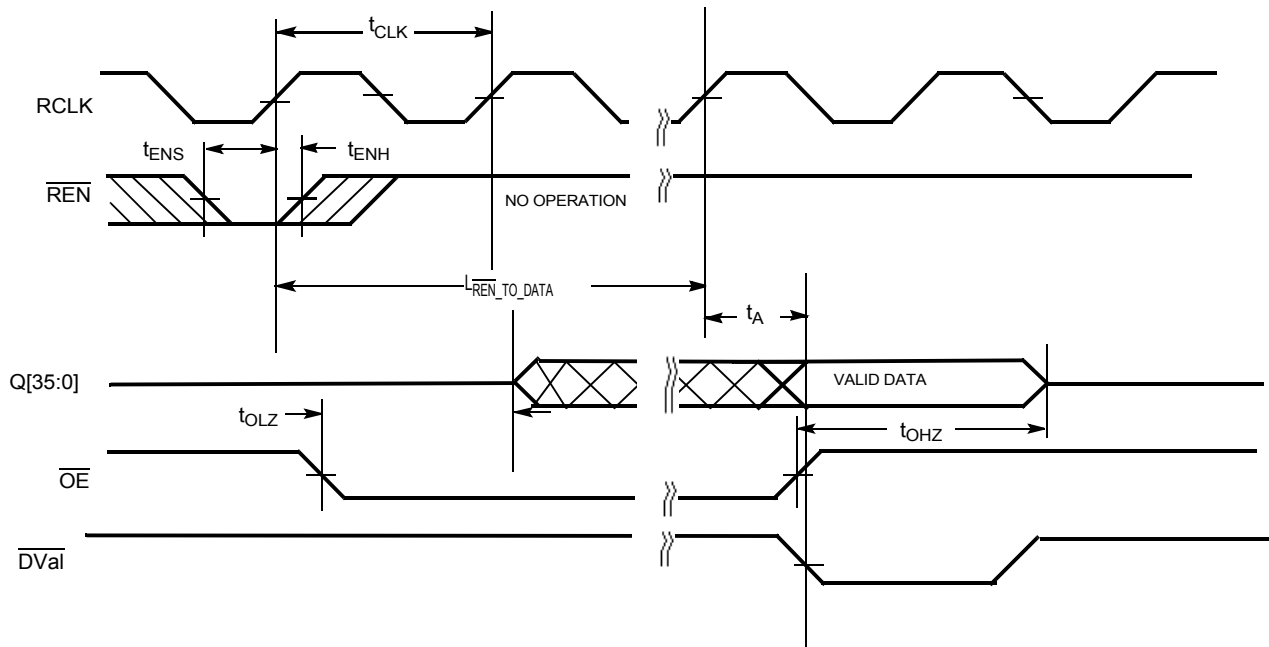
Parameter	Description	-133		Unit	
		Min	Max		
t <sub>PU</sub>	Power-up time after all supplies reach minimum value	–	2	ms	
t <sub>S</sub>	Clock cycle frequency	3.3 V LVCMOS	24	133	MHz
t <sub>S</sub>	Clock cycle frequency	1.8 V LVCMOS	24	133	MHz
t <sub>A</sub>	Data access time	–	10	ns	
t <sub>CLK</sub>	Clock cycle time	7.5	41.67	ns	
t <sub>CLKH</sub>	Clock high time	3.375	–	ns	
t <sub>CLKL</sub>	Clock low time	3.375	–	ns	
t <sub>DS</sub>	Data setup time	3	–	ns	
t <sub>DH</sub>	Data hold time	3	–	ns	
t <sub>ENS</sub>	Enable setup time	3	–	ns	
t <sub>ENH</sub>	Enable hold time	3	–	ns	
t <sub>ENS_SI</sub>	Setup time for SPI_SI and $\overline{\text{SPI\_SEN}}$ pins	5	–	ns	
t <sub>ENH_SI</sub>	Hold time for SPI_SI and $\overline{\text{SPI\_SEN}}$ pins	5	–	ns	
t <sub>RATE_SPI</sub>	Frequency of SCLK	–	25	MHz	
t <sub>RS</sub>	Reset pulse width	100	–	ns	
t <sub>RSF</sub>	Reset to flag output time	–	50	ns	
t <sub>OLZ</sub>	Output enable to output in Low Z	4	15	ns	
t <sub>OE</sub>	Output enable to output valid	–	15	ns	
t <sub>OHZ</sub>	Output enable to output in High Z	–	15	ns	
t <sub>WFF</sub>	Write clock to $\overline{\text{FF}}$	–	8.5	ns	
t <sub>REF</sub>	Read clock to $\overline{\text{EF}}$	–	8.5	ns	
t <sub>PLL</sub>	Time required to synchronize PLL	–	1024	cycles	
t <sub>RATE_JTAG</sub>	JTAG TCK cycle time	100	–	ns	
t <sub>S_JTAG</sub>	Setup time for JTAG TMS,TDI	8	–	ns	
t <sub>H_JTAG</sub>	Hold time for JTAG TMS,TDI	8	–	ns	
t <sub>CO_JTAG</sub>	JTAG TCK low to TDO valid	–	20	ns	

**Switching Waveforms**

**Figure 7. Write Cycle Timing**



**Figure 8. Read Cycle Timing**



Switching Waveforms (continued)

Figure 9. Reset Timing

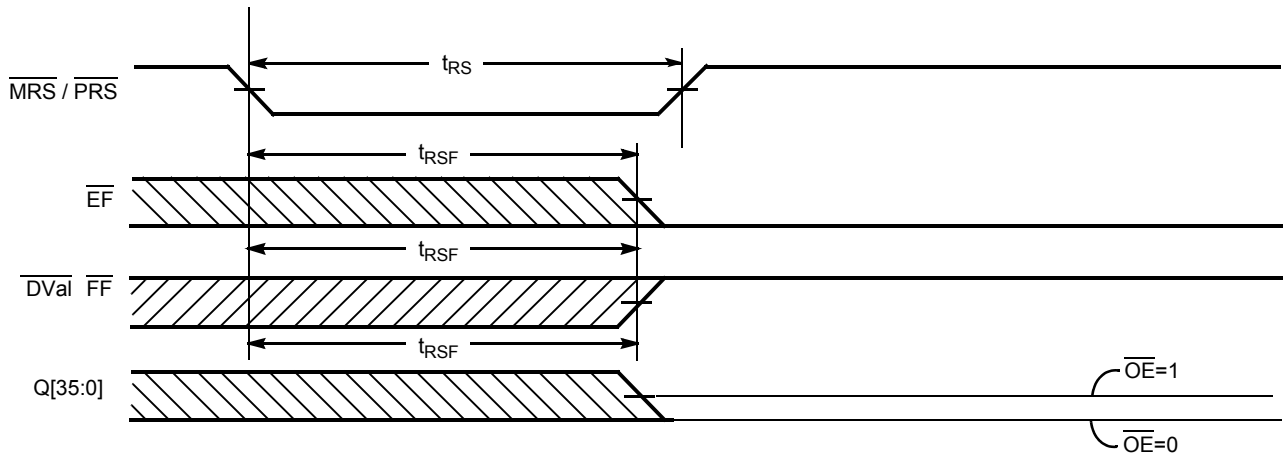


Figure 10. Empty Flag Timing

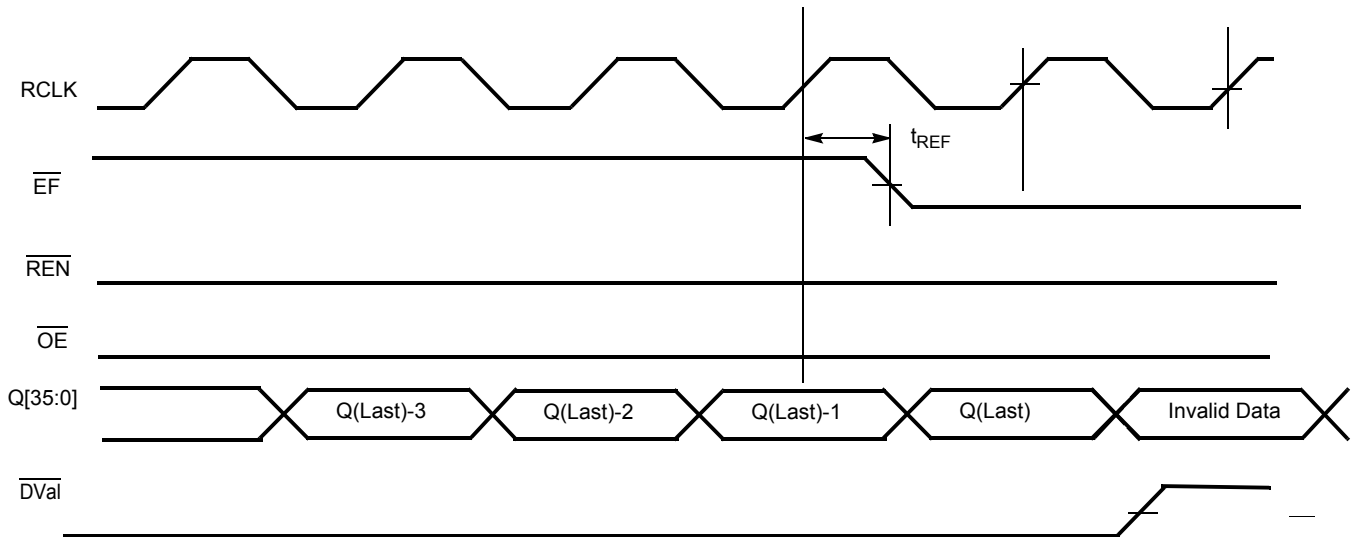
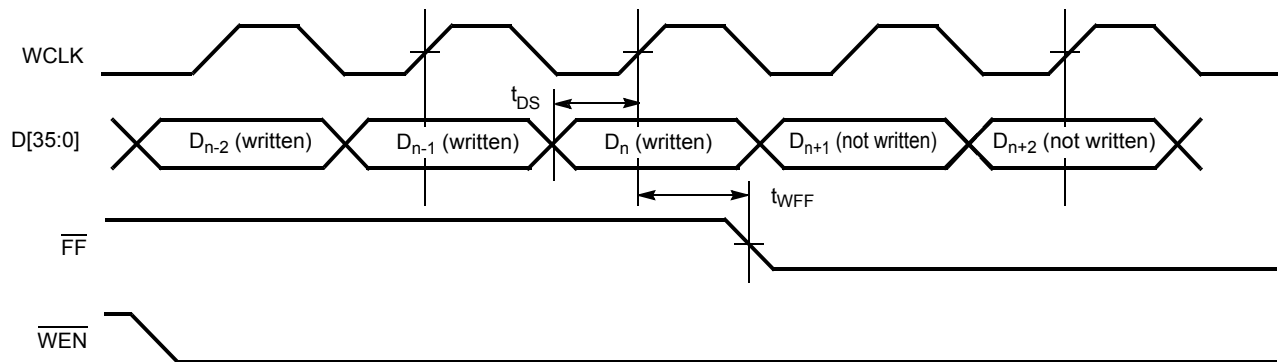


Figure 11. Full Flag Timing



Switching Waveforms (continued)

Figure 12. Initial Data Latency

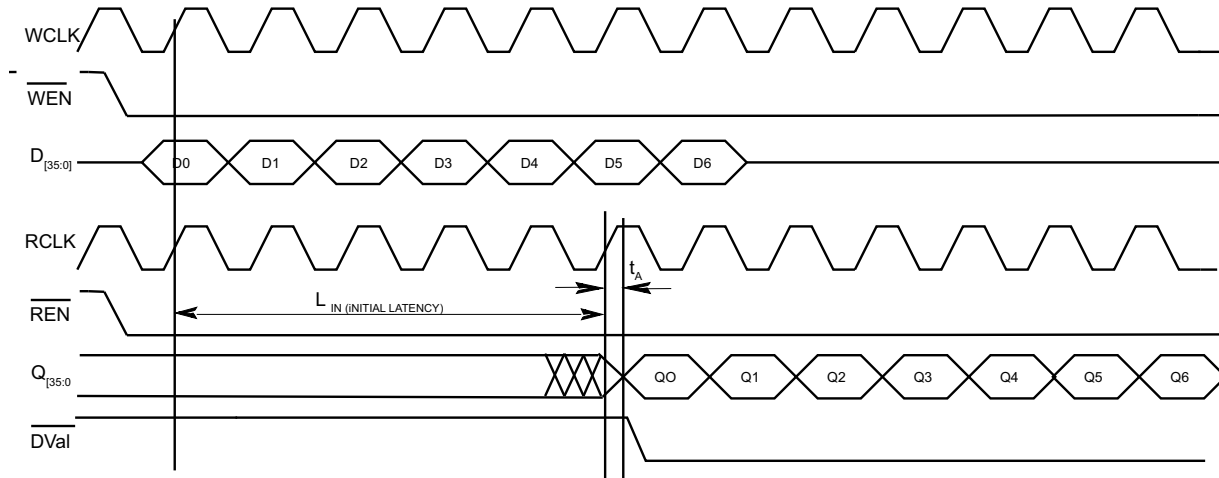
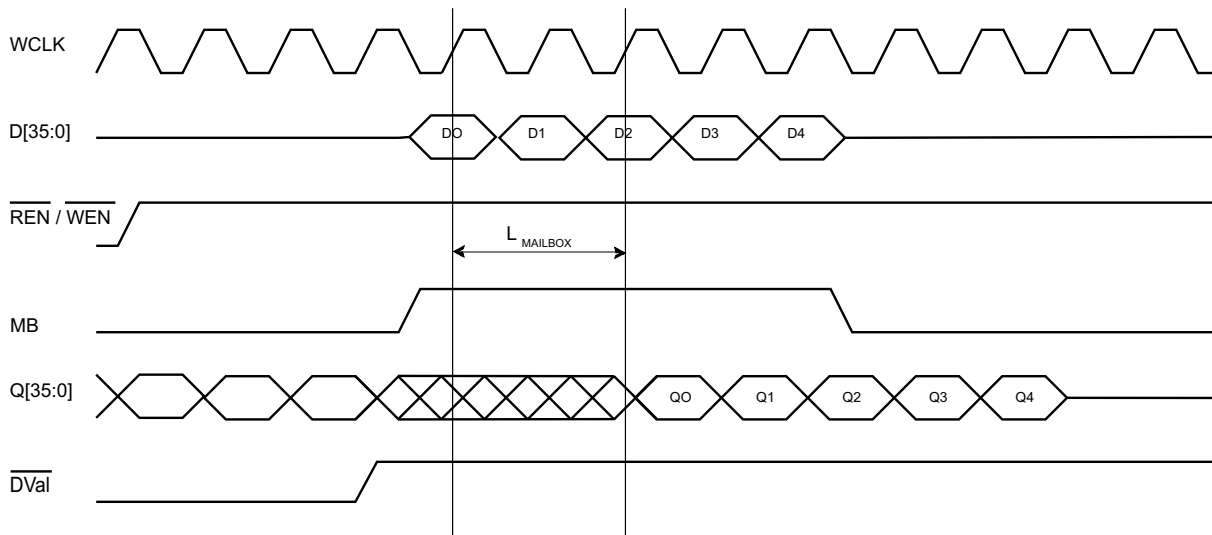


Figure 13. Flow-through Mailbox Operation



Switching Waveforms (continued)

Figure 14. Configuration Register Write

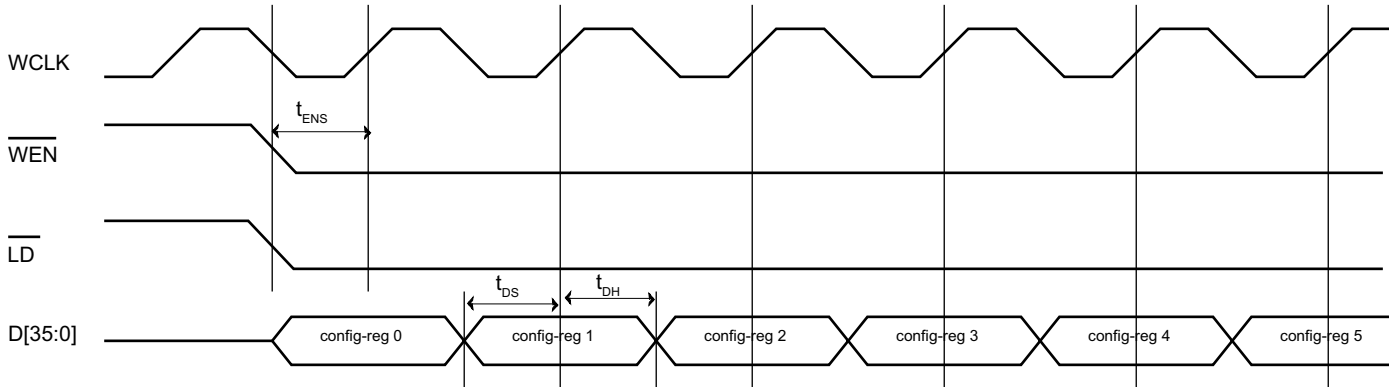


Figure 15. Configuration Register Read

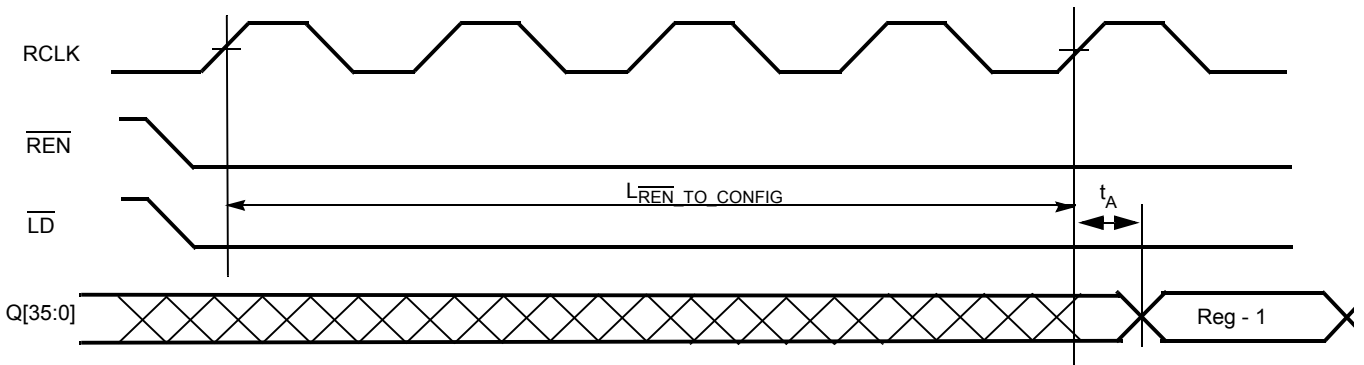
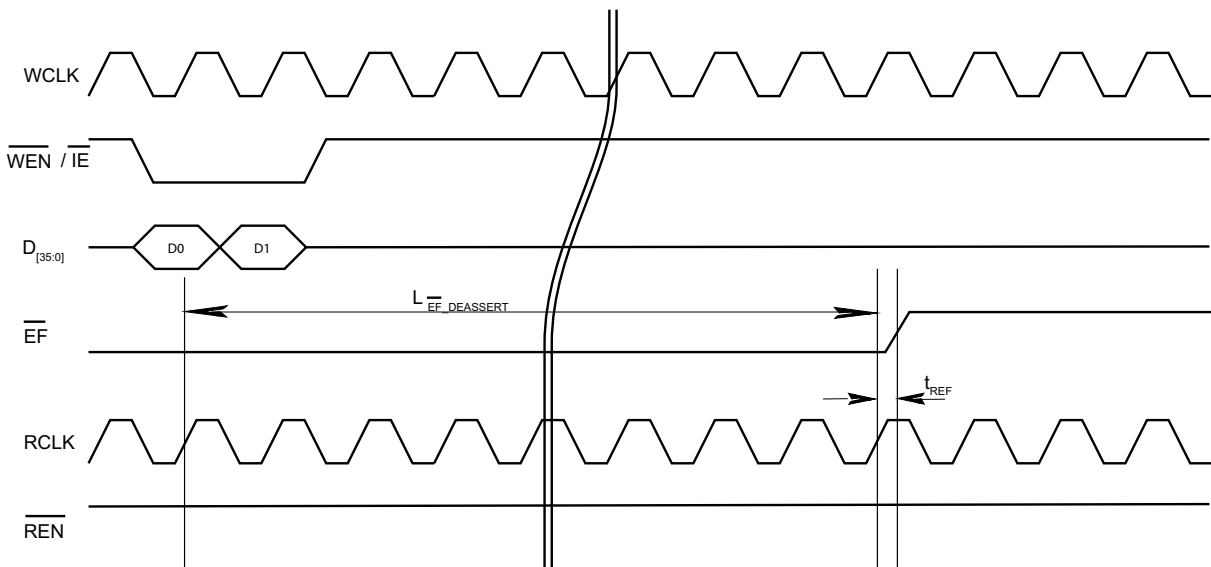
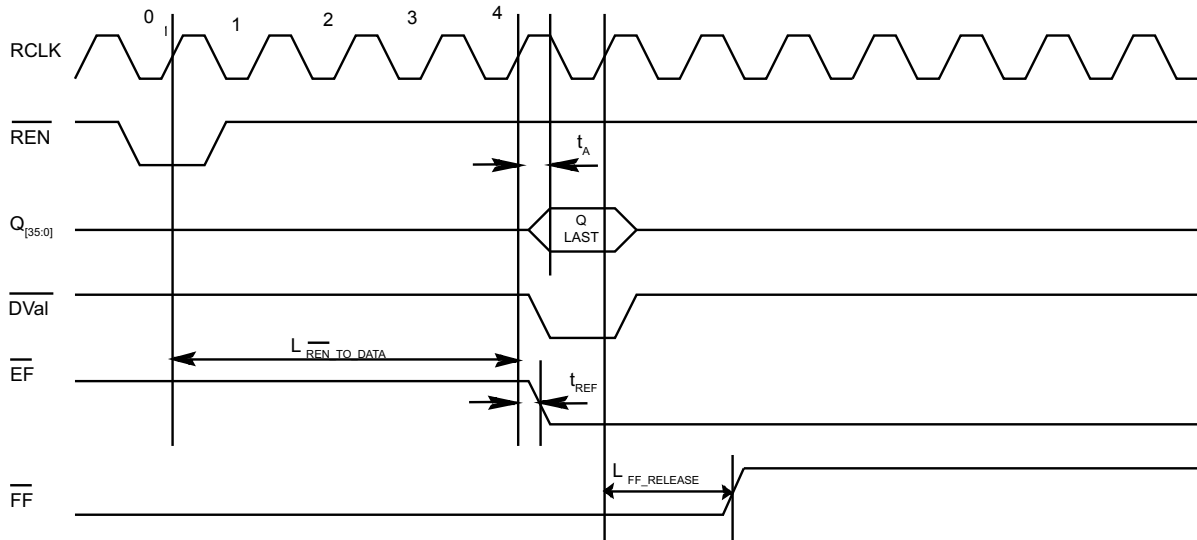


Figure 16. Empty Flag Deassertion

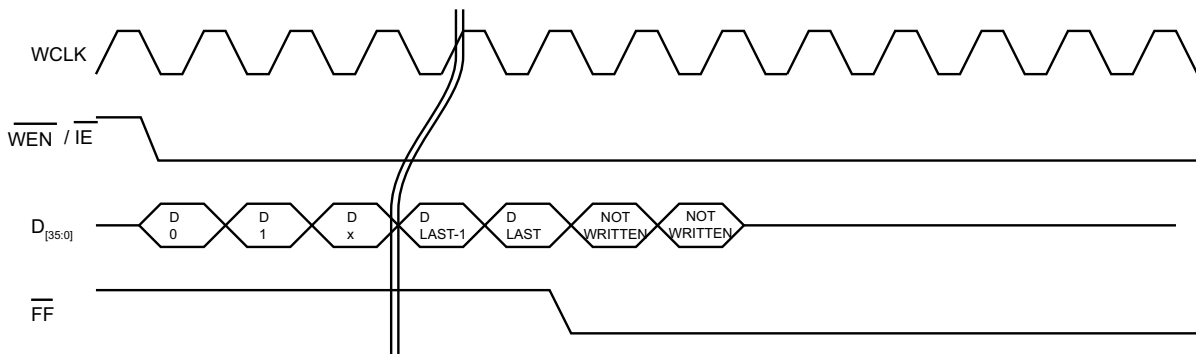


**Switching Waveforms** (continued)

**Figure 17. Empty Flag Assertion**

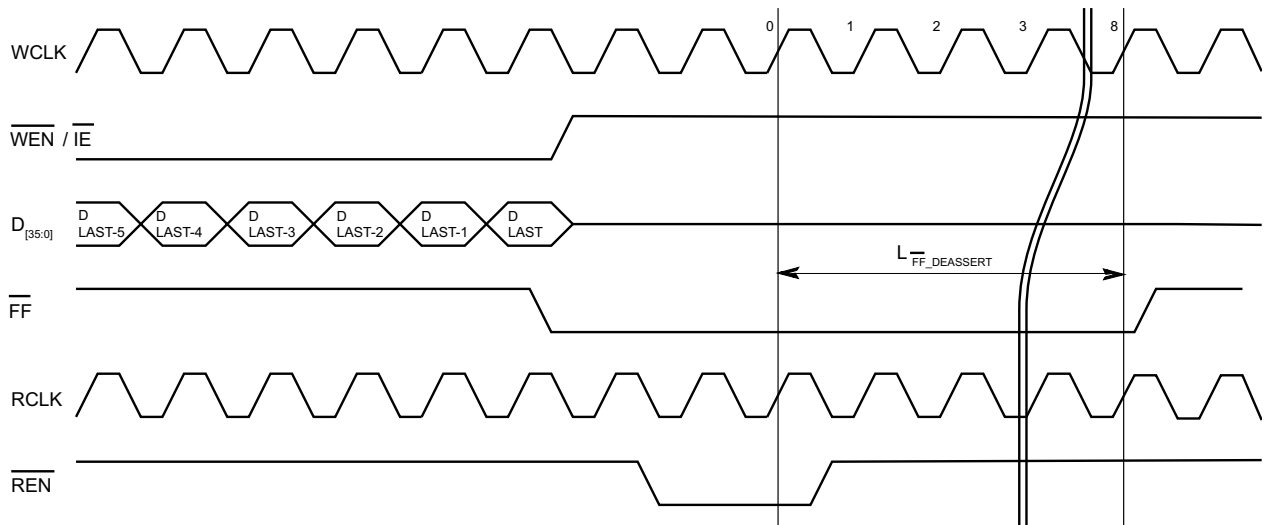


**Figure 18. Full Flag Assertion**



**Switching Waveforms** (continued)

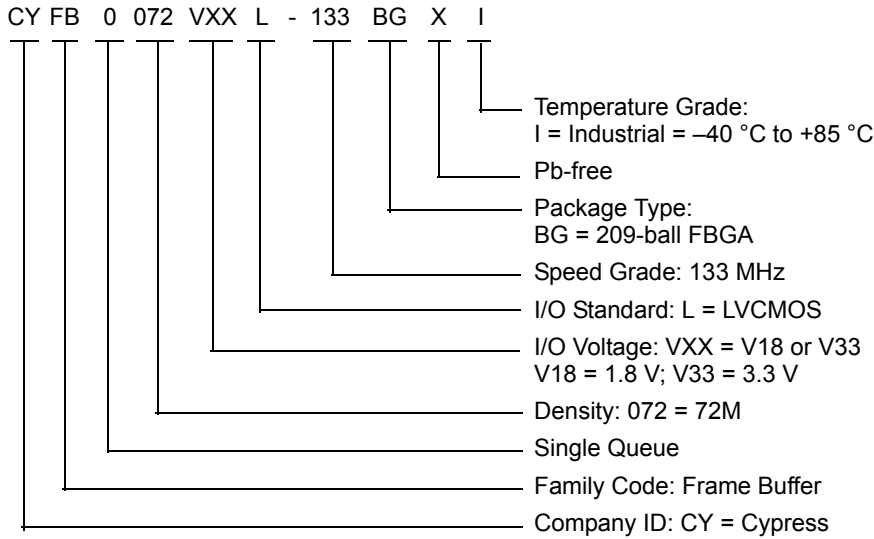
**Figure 19. Full Flag Deassertion**



### Ordering Information

Speed <sup>[6]</sup> (MHz)	Ordering Code	Package Diagram	Package Type	Operating Range
133	CYFB0072V18L-133BGXI	51-85167	209-ball FBGA (14 × 22 × 1.76 mm)	Industrial
133	CYFB0072V33L-133BGXI	51-85167	209-ball FBGA (14 × 22 × 1.76 mm)	Industrial

### Ordering Code Definitions

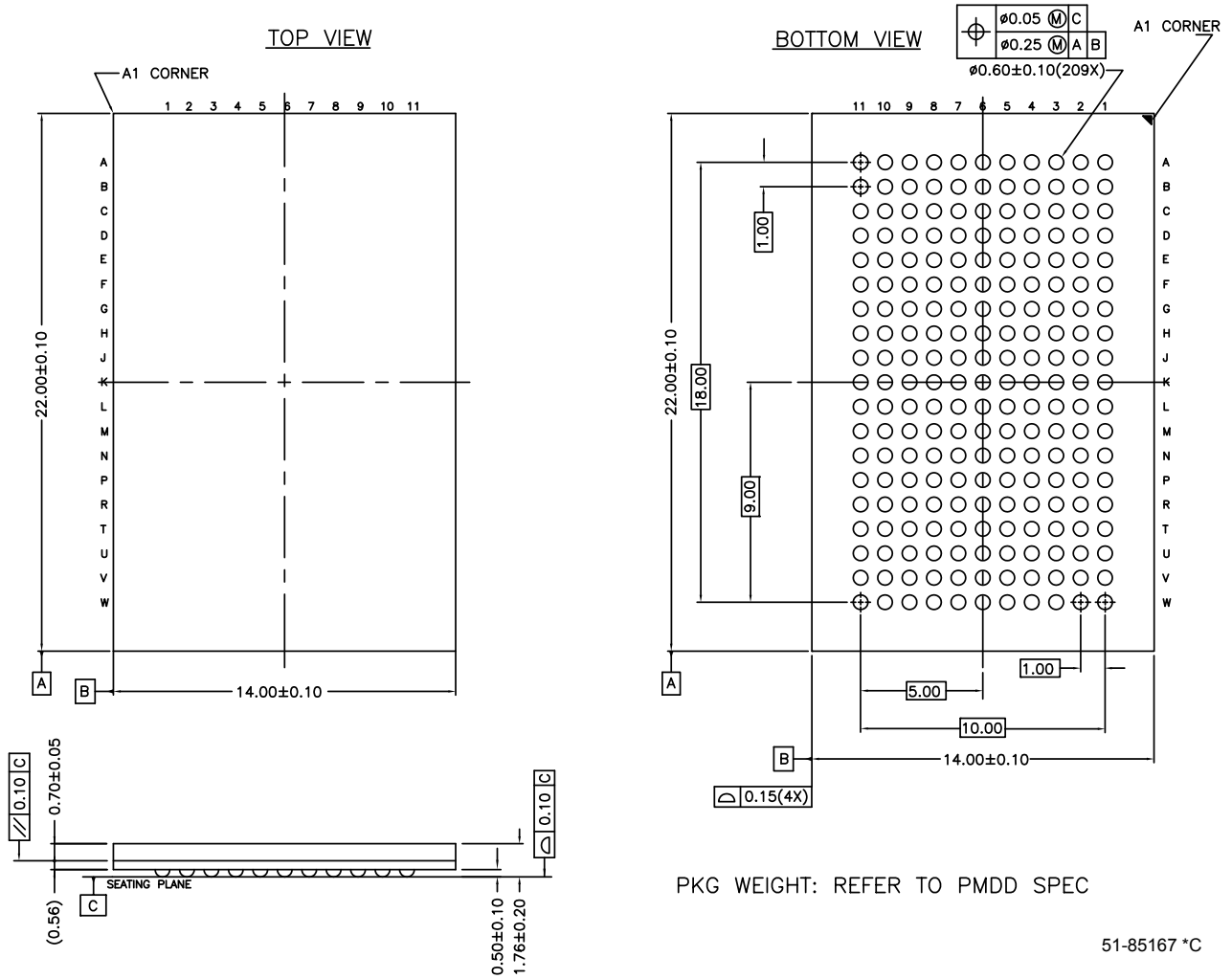


**Note**

6. For device operating at 150-MHz, Contact Sales.

Package Diagram

Figure 20. 209-ball FBGA (14 × 22 × 1.76 mm) BB209A Package Outline, 51-85167



## Acronyms

Acronym	Description
FF	Full Flag
FIFO	First In First Out
$\overline{IE}$	Input Enable
I/O	Input/Output
FBGA	Fine-Pitch Ball Grid Array
JTAG	Joint Test Action Group
LSB	Least Significant Bit
LVC MOS	Low Voltage Complementary Metal Oxide Semiconductor
MB	Mailbox
$\overline{MRS}$	Master Reset
MSB	Most Significant Bit
$\overline{OE}$	Output Enable
$\overline{PRS}$	Partial Reset
RCLK	Read Clock
$\overline{REN}$	Read Enable
RCLK	Read Clock
SCLK	Serial Clock
TCK	Test Clock
TDI	Test Data In
TDO	Test Data Out
TMS	Test Mode Select
WCLK	Write Clock
$\overline{WEN}$	Write Enable

## Document Conventions

### Units of Measure

Symbol	Unit of Measure
°C	degree Celsius
MHz	megahertz
μA	microampere
mA	milliampere
mm	millimeter
ms	millisecond
ns	nanosecond
Ω	ohm
pF	picofarad
V	volt
W	watt

## Document History Page

Document Title: CYFB0072V 72-Mbit Video Frame Buffer Document Number: 001-88646				
Revision	ECN	Orig. of Change	Submission Date	Description of Change
**	4084048	SMCH	09/16/2013	New data sheet.
*A	4136715	SMCH	09/26/2013	Changed status from Preliminary to Final. Updated <a href="#">Features</a> : Updated the sub-features under Memory organization. Updated <a href="#">Pin Configuration</a> : Updated <a href="#">Figure 1</a> . Updated <a href="#">Pin Definitions</a> : Added V <sub>PD</sub> pin details. Added Note 3 and referred the same note for V <sub>SS</sub> and V <sub>PD</sub> pins. Updated <a href="#">Architecture</a> : Updated <a href="#">Reset Logic</a> : Updated description. Updated <a href="#">Programming Configuration Registers</a> : Updated description.
*B	4581652	SMCH	11/25/2014	Added related documentation hyperlink in page 1.
*C	5323855	SMCH	06/28/2016	Updated <a href="#">Logic Block Diagram</a> . Updated <a href="#">Pin Configuration</a> . Updated <a href="#">Pin Definitions</a> . Updated <a href="#">JTAG Operation</a> . Updated CY Logo and Sales Disclaimer.
*D	5378669	DEVM	07/29/2016	Updated <a href="#">Pin Configuration</a> : Replaced TRST\ with DNU.
*E	5966514	AESATMP8	11/14/2017	Updated logo and Copyright.

## Sales, Solutions, and Legal Information

### Worldwide Sales and Design Support

Cypress maintains a worldwide network of offices, solution centers, manufacturer's representatives, and distributors. To find the office closest to you, visit us at [Cypress Locations](#).

#### Products

ARM® Cortex® Microcontrollers	<a href="http://cypress.com/arm">cypress.com/arm</a>
Automotive	<a href="http://cypress.com/automotive">cypress.com/automotive</a>
Clocks & Buffers	<a href="http://cypress.com/clocks">cypress.com/clocks</a>
Interface	<a href="http://cypress.com/interface">cypress.com/interface</a>
Internet of Things	<a href="http://cypress.com/iot">cypress.com/iot</a>
Memory	<a href="http://cypress.com/memory">cypress.com/memory</a>
Microcontrollers	<a href="http://cypress.com/mcu">cypress.com/mcu</a>
PSoC	<a href="http://cypress.com/psoc">cypress.com/psoc</a>
Power Management ICs	<a href="http://cypress.com/pmic">cypress.com/pmic</a>
Touch Sensing	<a href="http://cypress.com/touch">cypress.com/touch</a>
USB Controllers	<a href="http://cypress.com/usb">cypress.com/usb</a>
Wireless Connectivity	<a href="http://cypress.com/wireless">cypress.com/wireless</a>

#### PSoC® Solutions

[PSoC 1](#) | [PSoC 3](#) | [PSoC 4](#) | [PSoC 5LP](#) | [PSoC 6](#)

#### Cypress Developer Community

[Forums](#) | [WICED IOT Forums](#) | [Projects](#) | [Video](#) | [Blogs](#) | [Training](#) | [Components](#)

#### Technical Support

[cypress.com/support](http://cypress.com/support)

---



© Cypress Semiconductor Corporation, 2013-2017. This document is the property of Cypress Semiconductor Corporation and its subsidiaries, including Spansion LLC ("Cypress"). This document, including any software or firmware included or referenced in this document ("Software"), is owned by Cypress under the intellectual property laws and treaties of the United States and other countries worldwide. Cypress reserves all rights under such laws and treaties and does not, except as specifically stated in this paragraph, grant any license under its patents, copyrights, trademarks, or other intellectual property rights. If the Software is not accompanied by a license agreement and you do not otherwise have a written agreement with Cypress governing the use of the Software, then Cypress hereby grants you a personal, non-exclusive, nontransferable license (without the right to sublicense) (1) under its copyright rights in the Software (a) for Software provided in source code form, to modify and reproduce the Software solely for use with Cypress hardware products, only internally within your organization, and (b) to distribute the Software in binary code form externally to end users (either directly or indirectly through resellers and distributors), solely for use on Cypress hardware product units, and (2) under those claims of Cypress's patents that are infringed by the Software (as provided by Cypress, unmodified) to make, use, distribute, and import the Software solely for use with Cypress hardware products. Any other use, reproduction, modification, translation, or compilation of the Software is prohibited.

TO THE EXTENT PERMITTED BY APPLICABLE LAW, CYPRESS MAKES NO WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, WITH REGARD TO THIS DOCUMENT OR ANY SOFTWARE OR ACCOMPANYING HARDWARE, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. To the extent permitted by applicable law, Cypress reserves the right to make changes to this document without further notice. Cypress does not assume any liability arising out of the application or use of any product or circuit described in this document. Any information provided in this document, including any sample design information or programming code, is provided only for reference purposes. It is the responsibility of the user of this document to properly design, program, and test the functionality and safety of any application made of this information and any resulting product. Cypress products are not designed, intended, or authorized for use as critical components in systems designed or intended for the operation of weapons, weapons systems, nuclear installations, life-support devices or systems, other medical devices or systems (including resuscitation equipment and surgical implants), pollution control or hazardous substances management, or other uses where the failure of the device or system could cause personal injury, death, or property damage ("Unintended Uses"). A critical component is any component of a device or system whose failure to perform can be reasonably expected to cause the failure of the device or system, or to affect its safety or effectiveness. Cypress is not liable, in whole or in part, and you shall and hereby do release Cypress from any claim, damage, or other liability arising from or related to all Unintended Uses of Cypress products. You shall indemnify and hold Cypress harmless from and against all claims, costs, damages, and other liabilities, including claims for personal injury or death, arising from or related to any Unintended Uses of Cypress products.

Cypress, the Cypress logo, Spansion, the Spansion logo, and combinations thereof, WICED, PSoC, CapSense, EZ-USB, F-RAM, and Traveo are trademarks or registered trademarks of Cypress in the United States and other countries. For a more complete list of Cypress trademarks, visit [cypress.com](http://cypress.com). Other names and brands may be claimed as property of their respective owners.

## Looking for pricing, stock, or lifecycle information?

Click below to explore more details on WIN SOURCE:

-  [View CYFB0072V18L-133BGXI on WIN SOURCE](#)
-  [Infineon Technologies](#) Information

## Optimize Your Supply Chain with WIN SOURCE Solutions

-  Global Sourcing Solution
-  Obsolete Management
-  Cost Control Management
-  Shortage Management
-  Alternative Solution
-  Excess Inventory Management