

OVERVIEW

The LF3312 is good tool for synchronizing video or data streams with arbitrary timing to a set system or 'master' timing source. The timing sources are typically in the form of a Clock and a field/frame sync signal. Input video timing information such as the F, V, or H SYNC signals can be used to set or clear the Write pointer.

This application note focuses on synchronizing a single video stream with the LF3312 in single-channel mode using frame-based synchronization. For high-resolution video, multiple LF3312s can be cascaded and used in the same manner described in this application note. For lower resolution video or smaller data-frames in which 2 full field/frames can be stored within 12Mbits, the LF3312 in dual-channel mode can synchronize the 2 streams of video.

NOTE: In single-channel mode, AWEN and BWEN must be tied together, as must AREN/BREN, AIEN/BIEN, and AWCLK/BWCLK.

SYNCHRONIZING VIDEO USING EXTERNAL CONTROL SIGNALS

The first step in this application is to determine a common reference point that both the external and master/system timing domains can identify. Many methods can be used to set, clear, or mark address pointer locations within the LF3312. The method chosen for this application note is referencing address 0 within the memory as the start of a frame of video.

The F sync signal of the external video stream can be tied to ACLRb pin, which can clear the Write pointer to address 0 on field or frame boundaries. Another method for clearing the Write pointer at field/frame boundaries is using the embedded TRS data from the video stream. This is described in another application note.

The operational mode of the LF3312 should be set to single channel FIFO (OPMODE Register 8[2:0]=011). In the case that you wish to use a single LF3312 to synchronize 2 streams, use the dual-channel FIFO mode (OPMODE Register 8[2:0]=100).

Register 8[7:6] corresponds to the width of the data word. See the table below.

Word Width	Register 8 [7:0]
8 bit	0X
10 bit	10
12 bit	11

Using external F Sync to Reset/Clear Write Pointer

As the F sync of the external video signal falls, indicating the start of a new field and frame, it clears the internal Write pointer. This frame-based synchronization method only clears the write pointer on frame boundaries – not field boundaries. The rising edge of AWCLK that latches a LOW on ACLR (clearing the Write pointer) also latches the data that will be sent to address 0 in the memory. ACLR can be programmed to be level or falling edge sensitive (Register A). All SET and CLR pins are, by default, set to falling edge triggered – which means we do not need to actively program the

appropriate register in this application. (NOTE: Register A controls the SET and CLR trigger edge/level sensitivities).

Using Master/System F Sync to Reset Read Pointer

Upon indicating that a new frame is to begin on the system/master domain, its F-sync signal will go LOW, clearing the Read pointer. The F-sync (or a derivative) should be tied to the RCLR pin.

The timing of issuing the RCLR “frame/field sync” signal is referenced to RCLK, the system/master video clock.

In all FIFO modes, it takes 9 AREN/BREN-enabled rising edges of RCLK for the contents of address 0 to arrive on AOUT, the first sample of the new frame – after bringing RCLR LOW.

Masking Unwanted Data from Overwriting Previous Data (Ex: Freeze Frame)

If data exists in the input data that is unwanted, and should not overwrite previous data (and yet the buffer system must maintain synchronization), the LF3312’s Input Masking feature should be used. The two input mask control pins are AIEN and BIEN. These mask pins allow the writing of data into the memory to be disabled, and yet the write pointer can continue to sequentially increment. That is, the actual “writing” and “pointer increment” enables are two separate control pins, AIEN/BIEN and AWEN/BWEN respectively.

A simple example of the using write masking is during a freeze-frame situation. Here, we want to retransmit the previous frame over and over again and ignore the input video stream. We also want a simple mechanism to make this happen, maintaining synchronization and latency between write and read pointers – without anticipating timing and requiring pointer resets, etc... AIEN/BIEN is useful in that it only affects what is written into the memory and yet has no effect on timing.

The write pointer enable pins, AWEN and BWEN, must be brought LOW 2 AWCLK cycles before valid data is to be latched into the current pointer address and the pointer enabled to increment on the next rising edge of AWCLK.

The data write enable pins, AIEN and BIEN, must be brought LOW 2 AWCLK cycles before valid data on the AIN port is to be written into memory.

Keeping the ‘data-writing’ enabled (AIEN/BIEN brought LOW) while ‘pointer-incrementing’ is disabled (AWEN/BWEN brought HIGH) simply writes the data into the same address.

PIN	INITIALIZE	DESCRIPTION
AWCLK	----	Connect to external video source clock
BWCLK	Tie to AWCLK	In single channel mode, always tie to AWCLK
RCLK	----	Connect to master video clock
AIN	----	Connect to external video data source
BIN	Tie HIGH or LOW	Not Used – but must tie off
CHIP_ADDR	Tie all bits LOW	Serial interface ID bits not used
SCL	Tie HIGH	Not using serial interface
PADDR	----	Used for parallel interface
PDATA	----	Used for parallel interface
SDA	Tie HIGH	Not using serial interface
BOUT	UNCONNECTED	Not Used
ACL	Tie HIGH	No external pointer manipulation in this example
BCLR	Tie HIGH	Not used in this example
ASET	Tie HIGH	Not used in this example
BSET	Tie HIGH	Not used in this example
AMARK	Tie HIGH	Not used in this example
BMARK	Tie HIGH	Not used in this example
RSET	Tie HIGH	Not used in this example
RCLR	----	Connect to master F-sync
AWEN	Tie LOW	This example expects continuous input data stream
BWEN	Tie to AWEN	In single channel mode, always tie to AWEN
AIEN	Tie LOW	Always writing to memory
BIEN	Tie LOW	Always writing to memory
AREN	Tie LOW	This example expects continuous input data stream
BREN	Tie to AREN	In single channel mode, always tie to AREN
PROGRAM	Tie HIGH	This example uses parallel interface
LOAD	----	LOW at start of operation, bring HIGH after configuration
RESET	----	LOW for ≥ 1 cycles prior to configuration (while LOAD is LOW)
AOE	Tie LOW	This example never disables AOUT
BOE	Tie HIGH	BOUT is not used
CSB	----	Used for parallel interface
WEB	----	Used for parallel interface
REB	----	Used for parallel interface

Configuration Register Address	Data[7:0]	Description
8	83	Single Channel FIFO Mode, 10bit data word