

APPLICATION NOTE 3578

# Audio Data Transmission in MAX9217/MAX9218 Video Link

*Abstract: The MAX9217/MAX9218 serializer and deserializer chipset has been used for video data transmission with a single twisted-pair LVDS link in automotive and industrial applications. Video signals always have blanking periods in every displayed frame. One can use those blanking periods to 'piggyback' audio data. In this application note, we discuss the audio data transmission format, an approach for sending the data over the video link, and an example of system implementation.*

## Introduction

The MAX9217/MAX9218 chipset [1] is a transceiver pair in which the transmitter (MAX9217) converts parallel data to serial data sent the receiver (MAX9218), where it is converted back to parallel data. This chipset is designed to transmit video and control signals from a graphic controller (processor) to a LCD or plasma panel display through a single low-cost, twisted-pair cable like the UTP-CAT5 cable commonly used in Ethernet. The transmission distance can be more than 10 meters. The simple link structure and the use of low-cost cable made this chipset an ideal solution for video display applications in automotive, instrumentation, and medical equipment.

This chipset, however, is capable of much more than sending video signals between two distant points. Occasionally, people want the chipset to send an audio signal simultaneously. In this application note, we will discuss how to deliver audio data to the display panel side through the control signal channel during the video signal-blanking periods. We will also explain how to convert digital audio data to an analog audio signal, and then show the system structure for driving speakers in the panel side.

## MAX9217/MAX9218 Link Function and Video Data Format

The MAX9217 serializer has a 27-bit parallel input with a bus rate up to 35Mbps. Of these 27 bits, 18 bits are the video RGB data: 6 bits for each of the 3 primary colors and the remaining 9 bits for control signals. The first 3 bits of the 9-bit control signals are assigned to vertical, horizontal, and RGB data synchronizations: VSYNC (C0), HSYNC (C1) and ENAB (C2). The remaining 6 control bits (C3 to C8) can be used for other control signals. In our case, we use some of these 6 control bits for audio data transmission. The MAX9217 alternatively serializes the 18-bit RGB data or the 9-bit control data and transmits them through the single LVDS pair. The control data are sent during the blanking periods of video display, as indicated by the RGB data enable signal (ENAB).

When the serialized data are received at the MAX9218, they are converted back to parallel data in the same format as when they were inputs to the MAX9217. Also, when the parallel data are output from the MAX9218, a bus clock is regenerated based on the timing embedded in the serial LVDS link. **Figure 1** shows the block diagram of the link setup and connections for the video and control data between the MAX9217 and MAX9218. **Figure 2** illustrates the timing of the video data and control data. The duration ratio of control data over RGB data can be 1% to 5%, depending on the video format, display size, and link speed.

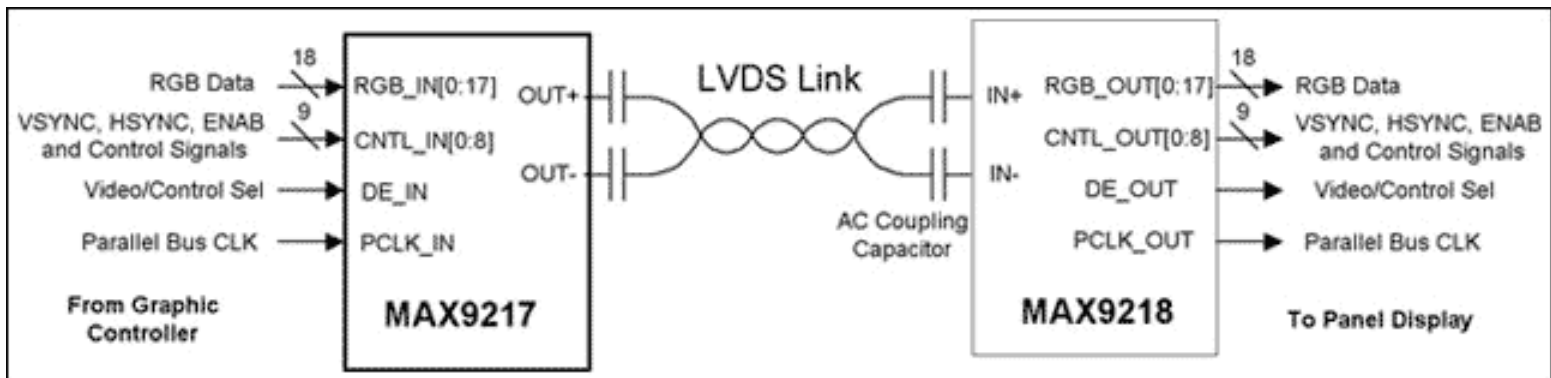


Figure 1. Video link setup with the MAX9217/MAX9218.

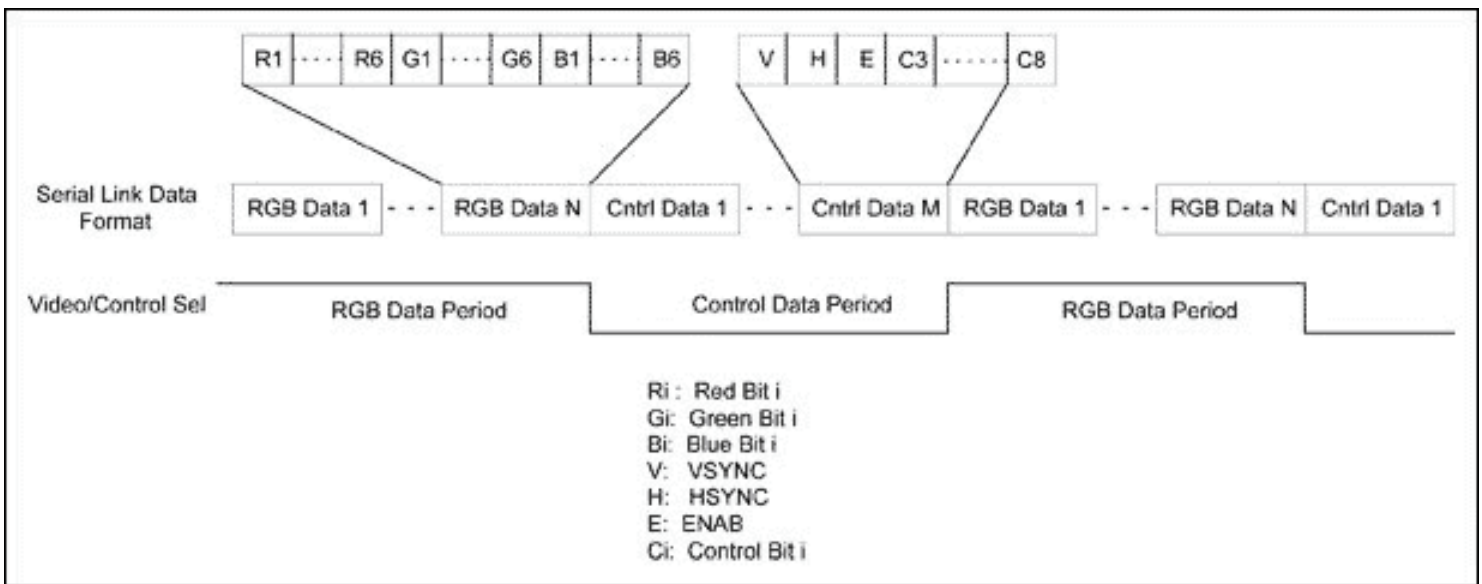


Figure 2. Video data and control data format in serial link.

## Digital Audio Data Types and Transmission Formats

There are many different formats for digital audio data. We will consider the three most popular formats: sampled digital audio (PCM), MPEG Layer 3 audio (MP3) [2], and the ATSC Digital Audio Compression Standard (AC3) [3].

PCM digital audio is the data format used in a CD ROM or DVD. The PCM digital signal is generated by sampling the left and right channel audio signals at the sampling rate of 44.1kHz and at a 16- or 32-bit resolution. Consequently, the PCM audio data rate is 1.41Mbps at 16 bits per sample or 2.42Mbps at 32 bits. A 700MB CD can hold about 60 minutes of music in the 16-bit PCM data format.

MP3 is the audio format used in the MP3 players, and is encoded by compressing the PCM audio data. The data rate of the stereo MP3 is from 112kbps to 128kbps. At such a data rate, the decoded MP3 sound has the same quality as the CD digital audio. AC3 is the digital audio coding standard used in digital TV, HDTV, and cinema. The coded data rate of the stereo AC3 is 192kbps.

To regenerate the sound, the coded audio data can be fed to an audio decoder chip which, in turn, creates the PCM digital data going to the audio DACs, which then restore the analog audio signal. In contrast, the noncoded digital audio data can be fed to audio DACs directly. (More details about that type of system implementation are shown later.)

The most popular serial audio digital interface for coded or noncoded audio data is the Inter-IC sound bus (I<sup>2</sup>S) [4]. **Figure 3** shows the I<sup>2</sup>S interface configuration and timing diagram. The boundary of each audio word is indicated by the signal, WS. Configuration Mode 1 will be used for our application. The data are latched into the receiver by the rising edges of the SCK signal, but when the SCK stays low, no data are received.

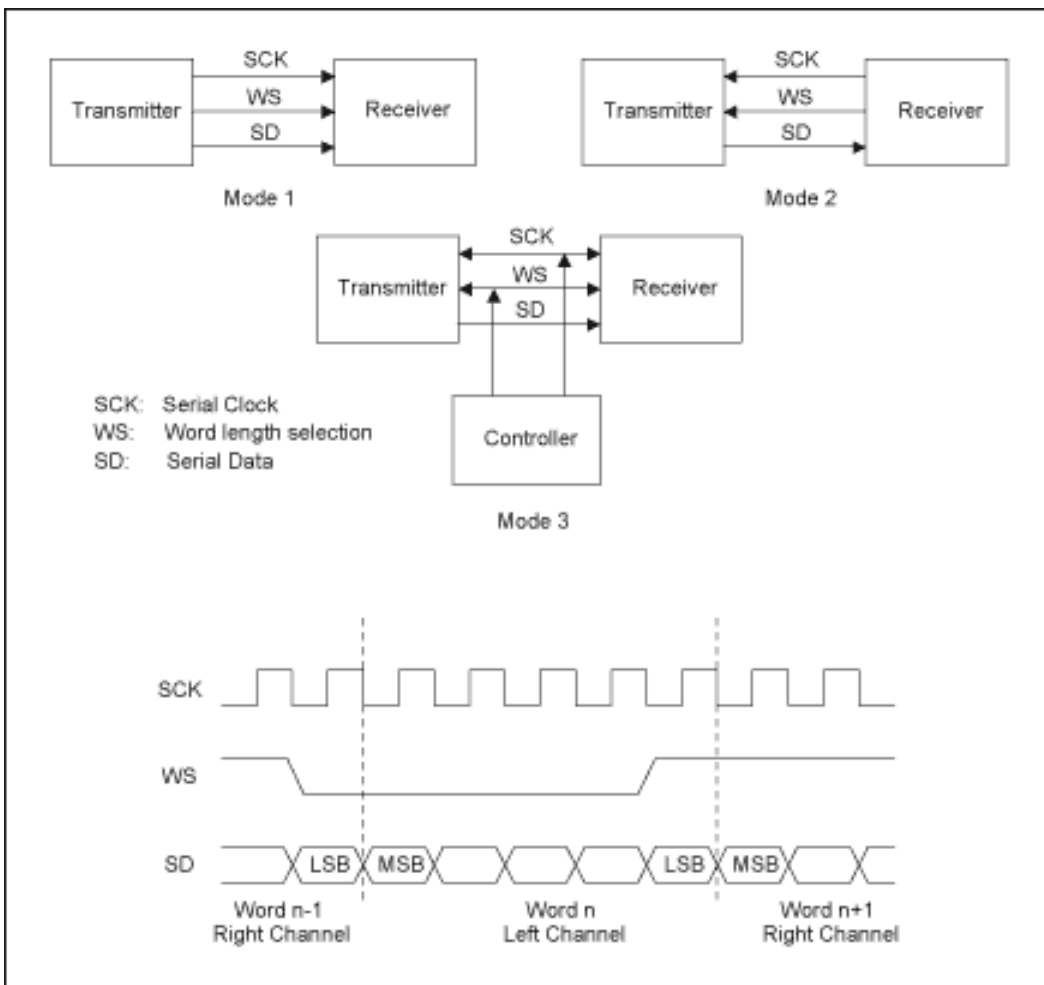


Figure 3. I<sup>2</sup>S interface configuration and timing.

We now use the serial link between the MAX9217 and MAX9218 to emulate an I<sup>2</sup>S interface, so we can transmit the audio data from the graphic controller side to a remote site. We assign the control bits C3 and C4 for the SD and WS signals, respectively. For the SCK clock, we can use the MAX9218's recovered pixel clock, PCLK\_OUT, directly if PCM digital audio is to be transmitted. For transmitting the MP3 or AC3 audio, the control bit, C5, can be used to generate a half or lower rate pixel clock for the SCK clock. **Figure 4** shows the timing waveforms of both cases. To prevent overflow in the receiver, throttling control is needed for most I<sup>2</sup>S interfaces and can be done easily by putting SCK to low when the data transmission should be held. If the SCK signal cannot be set to low during operation like that in Case1, we can use a chip select pin, active-low CS, to deactivate the receiver. In this case, we assign C6 for the active-low CS signal, as shown by Case 1 in Figure 4.

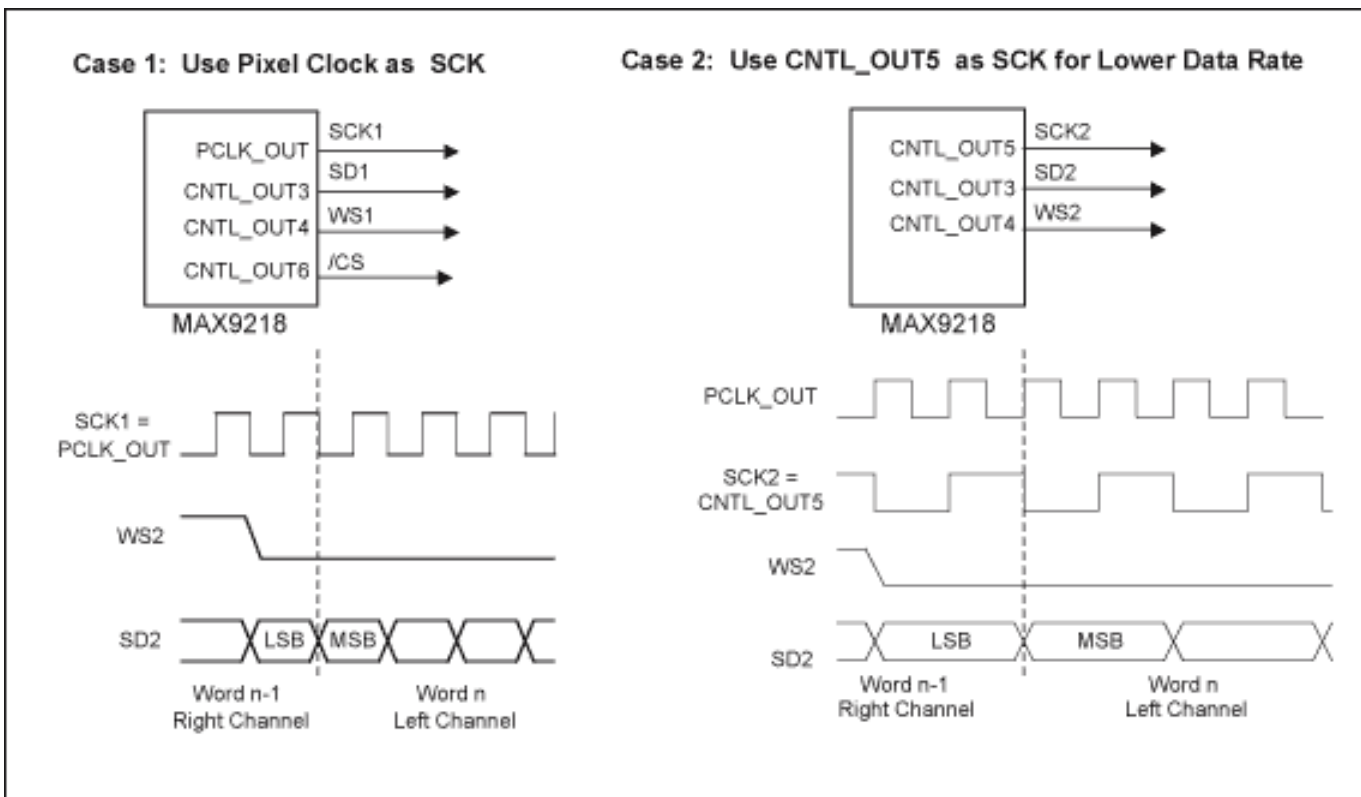


Figure 4. Waveforms of control data bits for an I<sup>2</sup>S interface.

## Blanking Ratio and Audio Data Throughput

Since the audio data are transmitted through the blanking periods of a video signal, we need to determine the required line-blanking ratio and the frame-blanking ratio with a given pixel frequency,  $f_p$ . **Figure 5** illustrates the line-blanking and frame-blanking periods on a display panel.

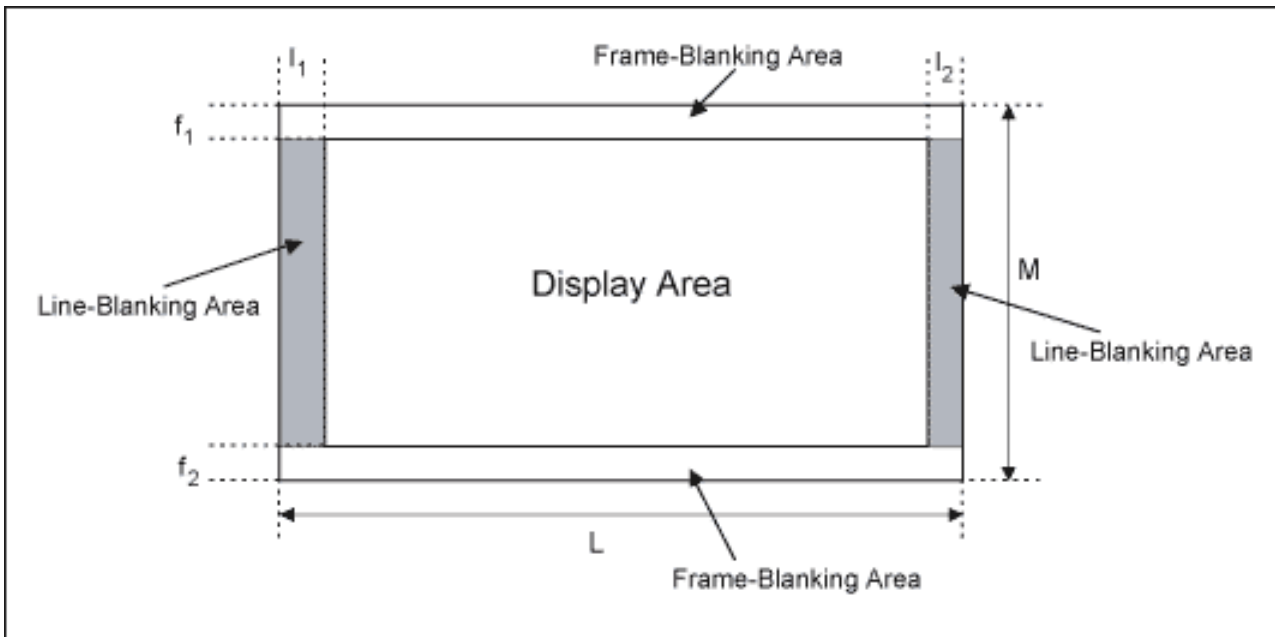


Figure 5. Line blanking and frame blanking.

Representing the line-blanking ratio as  $R_L$  and the frame-blanking ratio as  $R_F$ , we can calculate these ratios from Figure 5 as:

$$R_L = (l_1 + l_2) / L$$

and

$$R_F = (f_1 + f_2) / F$$

Thus, we can obtain the audio data throughput,  $R_A$ , as:

$$R_A = (R_F \delta_F + (1 - R_F) R_L \delta_L) f_P$$

where  $\delta_F$  and  $\delta_L$  are the utilization rates of audio data transmission during the blanking periods. Utilization rates describe the percentage of total blanking periods used for audio data transmission, are the result of throttling control. As examples, **Table 1** shows the parameters we can set to generate data rates for the three types of audio data.

**Table 1. Blanking Parameter Settings for Audio Data of Different Types**

Audio Data Type	$f_P$	$R_A$	$R_B$	$\delta_F$	$\delta_L$	Data Rate
16-Bit PCM Audio Data	35	0.02	0.03	81%	82%	1.41Mbps
MP3	17.5	0.01	0.01	35%	38.5%	128Kbps*
AC3	17.5	0.01	0.01	50.3%	60%	192Kbps*

\*Note: Both MP3 and AC3 audio data have head information. Counting those headers, the real coded data rates are slightly higher [2, 3].

## System Implementation

To replay the audio in the panel side, we need either to send the PCM data to audio DACs, or decode the MP3 or AC3 data and then send to the audio DACs. Because there is no reverse channel for sending a handshake signal back to the controller, the decoder master clock must be synchronized with the pixel clock to prevent the data overflow and underflow. **Figure 6** shows system block diagrams for implementing the audio replay for coded and noncoded data.

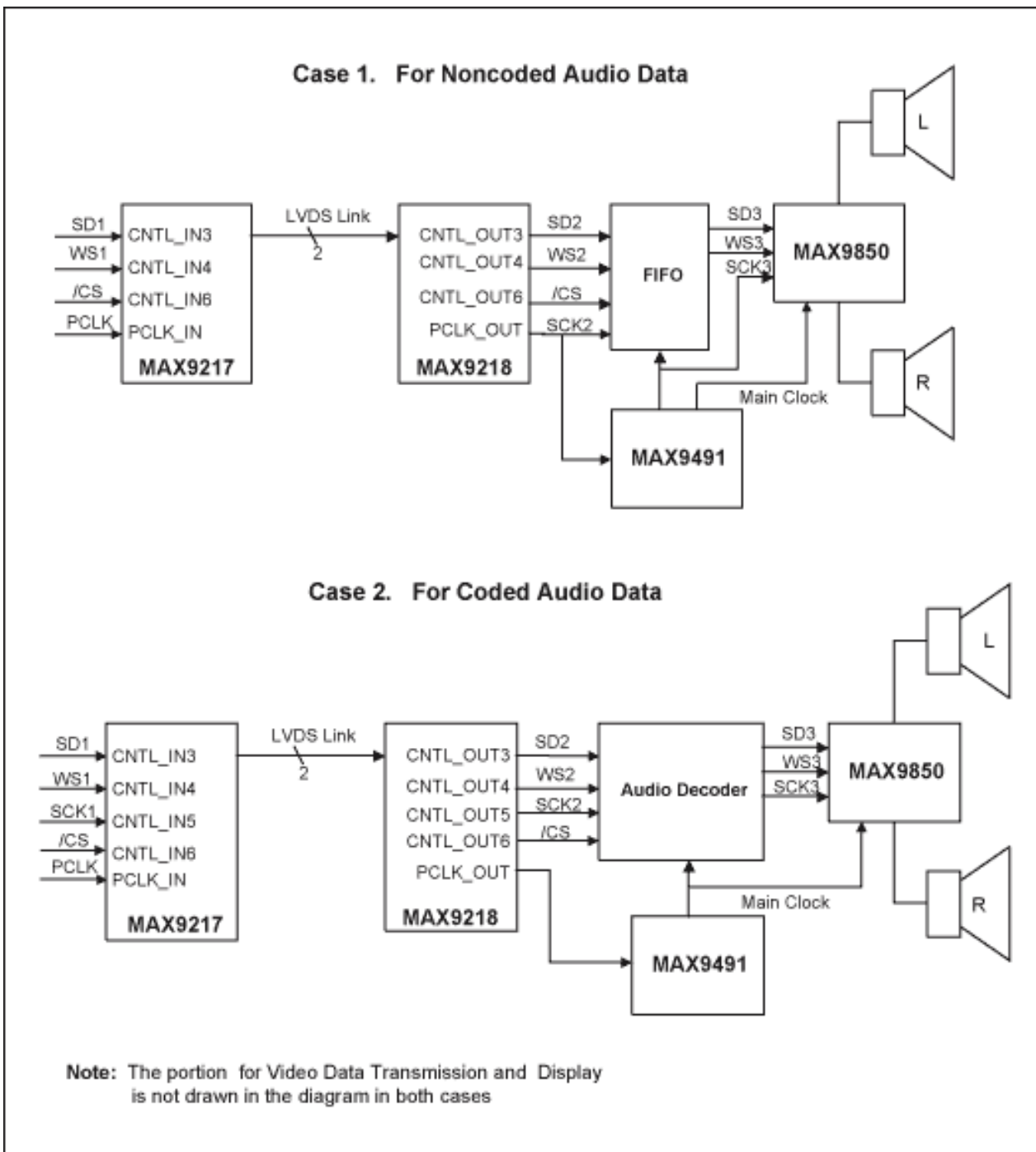


Figure 6. Implementation of audio replay in panel side.

In the block diagram above, the I<sup>2</sup>S interface is employed three times. The data rates for the first and second I<sup>2</sup>S interfaces from the left are the same, and can be as high as 35MHz. The rate for the third interface to the MAX9850 DirectDrive headphone amplifier [5] is fixed to multiples of audio sampling rates. The clock, SCK2, is fed to the MAX9491 multiple clock generator [6], which generates the synchronized clocks for decoder, FIFO, and the MAX9850. The MAX9491 has dual programmable PLLs with OTP and is an ideal frequency synthesizer for this purpose. The implementation in Case 1 above is for a graphic controller that can provide decoded PCM audio data, while Case 2 is for decoding the compressed data at the panel side. The throttling control in Case 1 is done by the active-low CS pin and in Case 2 by idling the SCK clock. Comparing the two implementations, we have seen that the Case 1 approach for the PCM audio data does not require much blanking time (Table 1) for transmission and, without using an audio decoder chip, it costs less than Case 2. Thus, sending the PCM data directly over the link is preferred, assuming that the graphic controller can generate it from coded audio data streams like MP3 or AC3.

**Reference**

- [1] [MAX9217](#) and [MAX9218](#) data sheets
- [2] [MPEG Audio Layer III frame](#)
- [3] [Digital Audio Compression \(AC-3\)](#)
- [4] [I<sup>2</sup>S Bus Specification](#)

[5] [MAX9850](#) data sheet

[6] [MAX9491](#) data sheet

---

Application Note 3578: [www.maxim-ic.com/an3578](http://www.maxim-ic.com/an3578)

#### More Information

For technical support: [www.maxim-ic.com/support](http://www.maxim-ic.com/support)

For samples: [www.maxim-ic.com/samples](http://www.maxim-ic.com/samples)

Other questions and comments: [www.maxim-ic.com/contact](http://www.maxim-ic.com/contact)

---

#### Automatic Updates

Would you like to be automatically notified when new application notes are published in your areas of interest? [Sign up for EE-Mail™](#).

---

#### Related Parts

MAX9217: [QuickView](#) -- [Full \(PDF\) Data Sheet](#) -- [Free Samples](#)

MAX9218: [QuickView](#) -- [Full \(PDF\) Data Sheet](#) -- [Free Samples](#)

MAX9485: [QuickView](#) -- [Full \(PDF\) Data Sheet](#) -- [Free Samples](#)

MAX9491: [QuickView](#) -- [Full \(PDF\) Data Sheet](#)

MAX9850: [QuickView](#) -- [Full \(PDF\) Data Sheet](#) -- [Free Samples](#)

AN3578, AN 3578, APP3578, Appnote3578, Appnote 3578

Copyright © by Maxim Integrated Products

Additional legal notices: [www.maxim-ic.com/legal](http://www.maxim-ic.com/legal)