Abstract

This document describes the potential signal race situation in the CMOS Sensor Interface (CSI) module of the i.MX processors. The signal race occurs between the HSYNC signal and the PIXCLK signal when there is poor routing in the PCB layout. When the signal race situation is triggered, an invalid pixel clock edge goes into the internal logic and results in dummy pixel data being received by the RxFIFO. The side effect is loss of frame sync and mis-alignment of lines.

The situation only occurs in gated-clock mode with the sensor running a continuous pixel clock.

This document applies to the following i.MX devices, collectively called i.MX throughout:

- MC9328MX1
- MC9328MXL
- MC9328MXS

Contents

1 Abstract .............................................. 1
2 Normal Operation ................................. 2
3 Potential Race Situation ....................... 2
4 How to Avoid Race Situation ............... 3
5 Revision History ................................. 3
2 Normal Operation

Figure 1 illustrates the normal operation of gated-clock mode.

![Correct Valid Edge Generation](image1)

The pixel clock from the i.MX processor’s CSI module is continuous. The pixel clock signal is synchronized with the HSYNC signal. HSYNC = 1 indicates the pixel clock is valid. Inside the i.MX processor, these two signals are ANDed to generate valid pixel clock edges. Data is latched into the RxFIFO at the valid pixel clock edges.

3 Potential Race Situation

Figure 2 illustrates the scenario that causes the race situation.

![Incorrect Valid Edge Generation](image2)

Depending on the PCB design, the HSYNC signal will skew ahead of the PIXCLK signal. If the skew is great enough, for example, a half pixel clock cycle, the HSYNC signal overlaps the previous pixel clock signal. This generates an error to the valid pixel clock signal used by the internal logic to latch data.
If the skew is small, however, the situation will not happen. The maximum amount of drift \( t_d \) allowed is dependent on the sensor, not the i.MX processor. There are two factors affecting the amount of drift allowed:

- Relative timing position of HSYNC to PIXCLK.
  Theoretically there is at most \( \frac{1}{2} \) PIXCLK period allowed for the drift.
- Frequency of PIXCLK. Higher frequency results in a smaller margin.

A printed circuit board design in which the routing for the PIXCLK signal is longer than that of the HSYNC signal, allows a delay to be inserted to the PIXCLK signal. Delay in PIXCLK is equivalent to a backward drift in HSYNC and results in the signal race situation.

4 How to Avoid Race Situation

The key to avoid the signal race situation is to ensure that HSYNC never overlaps with the previous pixel clock. Here are suggested solutions:

1. Restrict the routing of PIXCLK not to exceed that of HSYNC.
2. Insert a buffer/inverter to the HSYNC signal when the measured delay on PIXCLK is too long. This compensates for the delay on PIXCLK.
3. Gate PIXCLK with HSYNC. To accomplish, feed PIXCLK and HSYNC to an AND gate located close to the sensor output. The output of the AND gate then feeds the PIXCLK input of the i.MX processor. This inhibits PIXCLK when HSYNC is low and is a more robust solution than either solution 1 or 2. For high data rates, a similar AND gate may be needed in the pixel data path to compensate for the added propagation delay in the PIXCLK path.

5 Revision History

Table 1 details the document revision history.

<table>
<thead>
<tr>
<th>Revision Number</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>2</td>
<td>Editorial review, clarification to title and text. Applied new Freescale template.</td>
</tr>
</tbody>
</table>