

## Introduction

Recovering from  $H_{SYNC}$  loss in LCD monitors caused by poor signal coding implementation is important to maintaining good video imagery on many LCD monitors.

## Problem

It is not uncommon to experience corruption of  $H_{SYNC}$  in a Sync On Green (SOG) signal during Vertical retrace. Low cost SOG sources use AND gates to remove the  $H_{SYNC}$  or a XOR gate to invert the  $H_{SYNC}$  during the  $V_{SYNC}$  time. This implantation can cause the  $H_{SYNC}$  to be missing or be out of sync with the source. Also, not all monitors have adequate phase lock loop recovery time to recover from such corrupted  $H_{SYNC}$  signals. Such PLLs trying to recover the corrupted  $H_{SYNC}$  will generate distortion at the top of the LCD displays or not sync at all, causing a loss of the entire image.

The PLL in some CCD displays have such high loop gain and slow lock ability that they have a very hard time stabilizing in a short time with the replacement  $H_{SYNC}$  pulses (even with low error) so they will show a very small amount of jitter at the top 10% to 20% of the screen. Without the replacement  $H_{SYNC}$  pulses, the display is very badly corrupted and not useful. A much more complex design using a VCXO for the each video format would be needed.

This technical brief presents a simple circuit that supports SOG for game and computer video signals plus regenerate any corrupted or an entire loss of  $H_{SYNC}$  during Vertical retrace. Technical Brief 474 has a SOG design with true  $H_{SYNC}$  timing if you have control of the SOG sources yet; if you do not have control of the SOG input, this circuit will regenerate  $H_{SYNC}$  during  $V_{SYNC}$  pulse time regardless of the state of the incoming  $H_{SYNC}$ .

## Basic Circuit

This circuitry will be the source for all the  $H_{SYNC}$  inputs to the monitor. First, we need to detect and phase lock to a correct  $H_{SYNC}$ . Next, will be the detection of a corrupted or lost  $H_{SYNC}$  and re-synchronize  $H_{SYNC}$  to a non-corrupted  $H_{SYNC}$ . Finally we will need to remove  $H_{SYNC}$  from the incoming SOG and forward only the green video information to the monitor.

## Detecting $H_{SYNC}$

Intersil offers a simple solution to the  $H_{SYNC}$  detector in the ISL59885. We selected the ISL59885 due to it's multi video mode capability. With normal inputs, the ISL59885 will generate the  $V_{SYNC}$  and  $C_{SYNC}$  outputs. When we have missing input  $H_{SYNC}$ , such as with SOG, the LCD monitor may not properly sync. We can use the  $H_{SYNC}$  or the  $C_{SYNC}$

outputs with an external PLL to regenerate the missing  $H_{SYNC}$  during the  $V_{SYNC}$  time. This will result in a usable image on the monitor and will also support multi video modes.

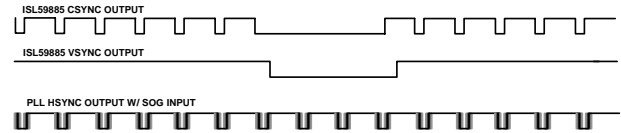


FIGURE 1. ISL59885  $H_{SYNC}$  OUTPUT WITH SOG INPUT

Using this sync separator, we can detect the  $H_{SYNC}$  including a High Definition  $H_{SYNC}$  as well as normal NTSC/PAL. The ISL59885 extracts video sync timing information from both standard and non-standard video. To make use of this sync extractor, we first need to terminate the input cable using a  $75\Omega$  resistor to ground. Next, we need to filter out the chroma by using a simple RC filter on the input. Since HD has a no chroma signal, we can simply switch out the second capacitor in parallel when HD is detected to change the filter characteristics for HD. The ISL59885 will now output  $H_{SYNC}$  from the source and be referred to as 'source  $H_{SYNC}$ ' in this document.

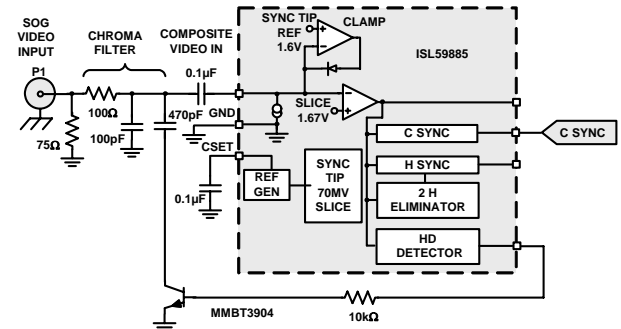


FIGURE 2.  $H_{SYNC}$  DETECTION

## Determine the Status of $H_{SYNC}$

We need to compare a reference  $H_{SYNC}$  with the incoming source  $H_{SYNC}$  to determine if the incoming  $H_{SYNC}$  is valid. To do so, we need to generate the reference  $H_{SYNC}$ . The most efficient design would center on using a Phase Lock Loop that would synchronize to the correct  $H_{SYNC}$  from the source.

## Phase Lock

We selected a simple low power, low cost digital PLL (the HC4046) to be the basic source for the  $H_{SYNC}$  substitution. This PLL needs to have a wide lock range to support a wide range of video  $H_{SYNC}$  rates from 16kHz to 64kHz (standard NTSC to HDTV). A wide locking range will cause the PLL to have a jitter that is too large to support a clean video  $H_{SYNC}$

for much longer than the  $V_{\text{SYNC}}$  interval. Using the PLL oscillator running at a high rate (say 32x that of the  $H_{\text{SYNC}}$ ), we can use a digital divider to help reduce the jitter by a factor of the divider, in this case by 32. This will keep the display PLL in lock with only a small error during the  $V_{\text{SYNC}}$  time when there is no active video. It will also keep the monitor's on-board PLL in a stable region.

We selected the PLL output to be 32x the desired  $H_{\text{SYNC}}$ . 32x  $H_{\text{SYNC}}$  is within the normal operational range of the PLL and large enough to take advantage of the inherent reduction of the  $H_{\text{SYNC}}$  jitter. By using a simple digital counter, the HC4040 (12 stage divider), we can divide down the PLL oscillator to the correct  $H_{\text{SYNC}}$  frequency and also divide any jitter by the same 32x.

The PLL generated  $H_{\text{SYNC}}$  will be referred to as the PLL  $H_{\text{SYNC}}$  for the remainder of this document.

**Detection Circuit**

We now need to develop a detection circuit, which will compare the PLL  $H_{\text{SYNC}}$  with the ISL59885 source  $H_{\text{SYNC}}$  and indicate the status of the source  $H_{\text{SYNC}}$ . We can use a simple digital approach to the problem by using a basic D-flip/flop with the source  $H_{\text{SYNC}}$  driving the D input and the PLL  $H_{\text{SYNC}}$  driving the clock. If the reference  $H_{\text{SYNC}}$  is high at the start of the PLL  $H_{\text{SYNC}}$  rising edge, the flip/flop Q output will go high on the rising edge of the PLL's  $H_{\text{SYNC}}$ , thus indicating a missing/corrupt source  $H_{\text{SYNC}}$  pulse.

The ISL59885 was selected because it support multi video  $H_{\text{SYNC}}$  rates. We can use the  $C_{\text{SYNC}}$  output as it is a comparator output of both  $H_{\text{SYNC}}$  and  $V_{\text{SYNC}}$ . Now, all we need is a pulse former on the  $C_{\text{SYNC}}$  output to extract the  $H_{\text{SYNC}}$  portion of the input. Otherwise  $C_{\text{SYNC}}$  output will stay low during the  $V_{\text{SYNC}}$ , time and give a false  $H_{\text{SYNC}}$  response.

Another issue to consider is the external PLL cannot sync to a corrupted source  $H_{\text{SYNC}}$ . We will need to open the PLL control loop and allow the PLL to coast for the non-active video time, while maintaining a proper  $H_{\text{SYNC}}$  to the monitor. If we can NAND the Q and delay Q', the NAND gate output can be used to open the PLL loop via a switch. The coast time should be about three valid  $H_{\text{SYNC}}$  times (the number of  $H_{\text{SYNC}}$  during the  $V_{\text{SYNC}}$  sync window). At the same time, select PLL  $H_{\text{SYNC}}$  pulses to drive the monitor.

We can delay Q' to the NAND-Gate by using a series 10kΩ with a 0.047μF capacitor to ground. This RC ramp would delay Q' from going low for about 300μs. The output of the NAND gate drives a simple analog switch, ISL43110, and keep the PLL from trying to synchronize to the false  $H_{\text{SYNC}}$  signal and let the PLL coast for the same 300μs. Before 300μs is over, the input source  $H_{\text{SYNC}}$  would typically be valid again and the PLL will return to run in a synchronized mode. If the input source  $H_{\text{SYNC}}$  is not valid within 300μs then the PLL will be forced to go back to the synchronized mode and start to re-sync lock on the next  $H_{\text{SYNC}}$ .

A subtle issue, but none the less important, is the timing relationship of these signals.

The ISL59885's source  $H_{\text{SYNC}}$  is delayed into the PLL to allow the detection circuitry to determination of the status of  $H_{\text{SYNC}}$  and the selection circuit to select the correct  $H_{\text{SYNC}}$ . The delays have to support the setup and hold times for the D-flip/flop to avoid a race condition. To bound the delays you have to remember that the longer the  $H_{\text{SYNC}}$  delay, the more you have left side blanking of the screen. However, if you have too little delay, you have more chance you have of a race condition and false  $H_{\text{SYNC}}$  detection.

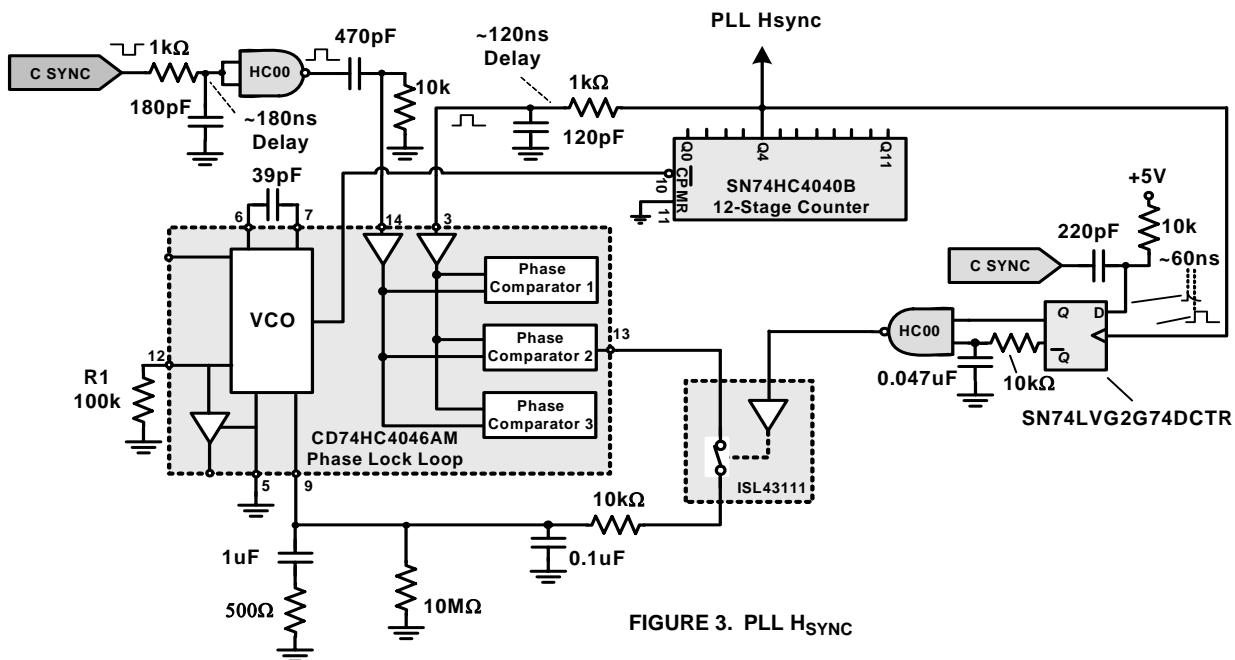


FIGURE 3. PLL Hsync

**Delay Computation**

The source H<sub>SYNC</sub> delay needs to be greater than the time it takes for the PLL to go into the coast mode and take into account other propagation delays as well. This would be the sum of the delays:

**DETECTION CIRCUITRY**

- 74HC00 - Nand-Gate ~20ns
- 74HC74 - D flip/flop propagation delay ~45ns

**PLL CIRCUITRY**

- CD4040 and CD4046 – not an issue since the VCO phase comparator will force the input such that the output will be phase synchronous with the delayed H<sub>SYNC</sub>.
- ISL43110 – Switch turn-on/off time ~80ns

The sum would be about 145ns, but we should add a safeguard to insure that we allow for any other parasitic delays. A 25% buffer would support our needs. Thus, the total delay of H<sub>SYNC</sub> is about 180ns delay. A simple series 1kΩ resistor and 180pF capacitor to ground into a logic inverter will give us about 180ns delay or about 5 to 6 pixel widths on the left side of the monitor. This is adequate time to allow for the propagation and to allow the detection circuitry to respond and select the proper H<sub>SYNC</sub> source but not impact the video image. This is the delayed source H<sub>SYNC</sub> signal.

One remaining timing issue is the detection flip/flop setup and hold time. In our design, we use the source H<sub>SYNC</sub> as the D input and the reference H<sub>SYNC</sub> as the clock. We need to delay the clock by the setup and hold time after applying the source H<sub>SYNC</sub>.

Remember, connecting the delayed PLL H<sub>SYNC</sub> to the Comp IN of the HC4040 and the reference H<sub>SYNC</sub> into the SIG IN, will force the VCO of the PLL to be in phase at the input to the PLL. For correct timing, the PLL H<sub>SYNC</sub> needs to be delayed by only enough to compensate for the setup and hold of the D flip/flop. Delaying the PLL Q4 (PLL H<sub>SYNC</sub>) by 120ns and delaying the clock source H<sub>SYNC</sub> by 180ns, (180ns - 120ns = 60ns) will compensate for the setup and hold time for the D-flip/flop. Using an RC 100Ω serial and 120pF to ground will generate the necessary delay.

Controlling the PLL loop when running in coast mode will require a heavily dampened feedback network supports the pre-open loop H<sub>SYNC</sub>. A loop filter of a series 10kΩ resistor and 1μF capacitor to ground with a 0.1μF capacitor in parallel will keep PLL at the proper frequency while in coast mode. The PLL was not designed to operate in coast mode so a 10mΩ resistor in parallel to the loop filter has to be added to ensure the PLL will relock with very low jitter after coast mode. We selected the loop VCO to be 32x that of the H<sub>SYNC</sub> to give the PLL a wide enough operational range to support H<sub>SYNC</sub> modes from 16kHz, 32kHz and 64kHz, and to reduce jitter.

**H<sub>SYNC</sub> Selector Circuitry**

At the time we place the PLL in coast mode, we need to select the PLL H<sub>SYNC</sub> output as the H<sub>SYNC</sub> input to the monitor.

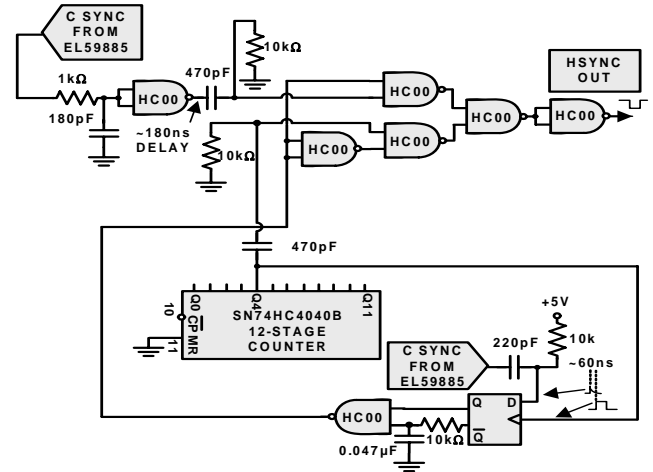


FIGURE 4. FALSE H<sub>SYNC</sub> DETECTOR

We use the same signal which places the PLL in coast mode to select the counter output as the replacement H<sub>SYNC</sub>. The simple arrangement of NAND gates is used to make the selection and proper polarity for H<sub>SYNC</sub>. Since H<sub>SYNC</sub> has a defined pulse width, using a series capacitor to couple the H<sub>SYNC</sub> into the NAND gate will properly pulse shape the signal. We use a series 470pF capacitor and 10kΩ to ground to set the proper pulse width to about 3μs.

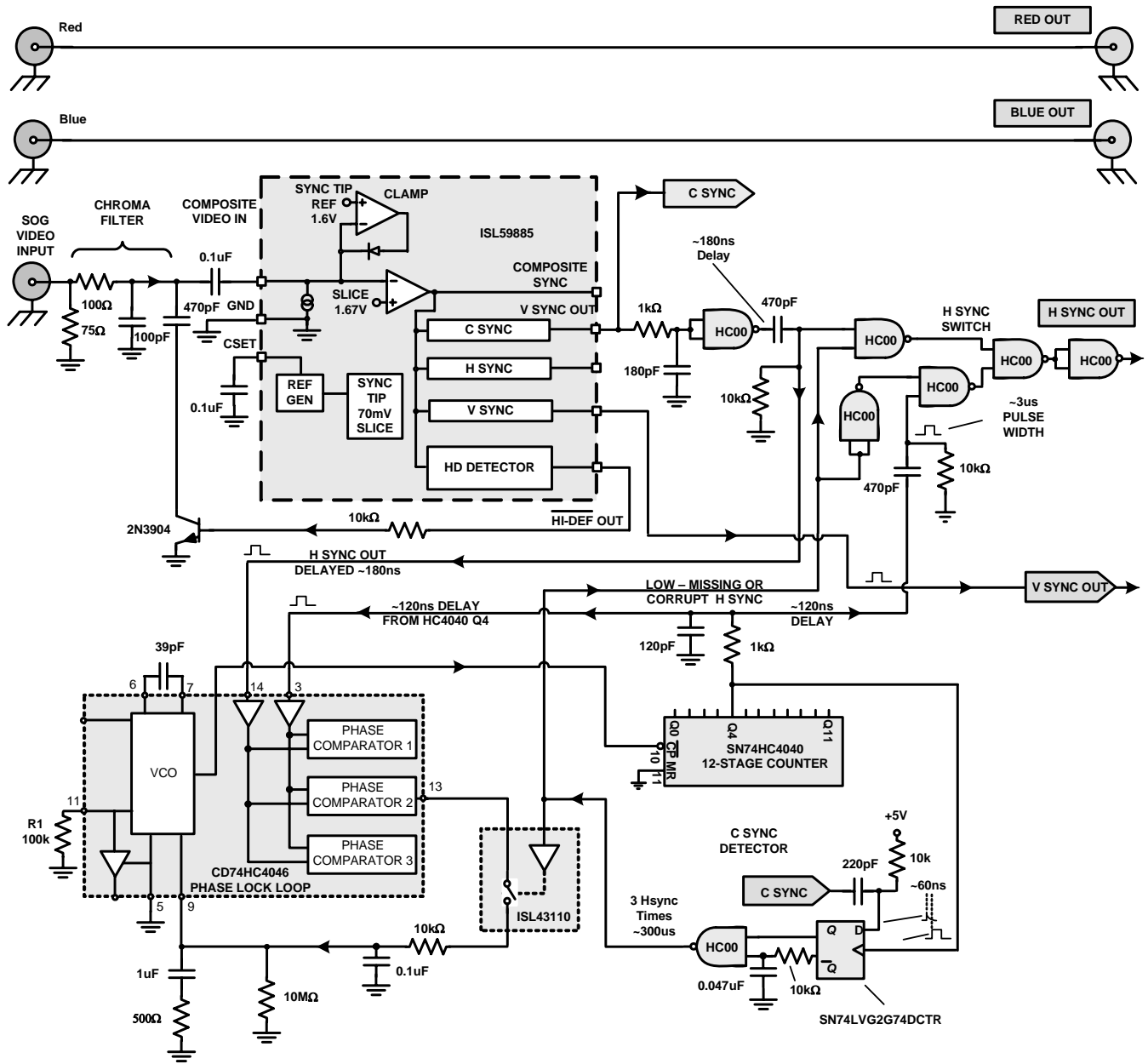


FIGURE 5. COMPLETE H<sub>SYNC</sub> CIRCUITRY

Summary

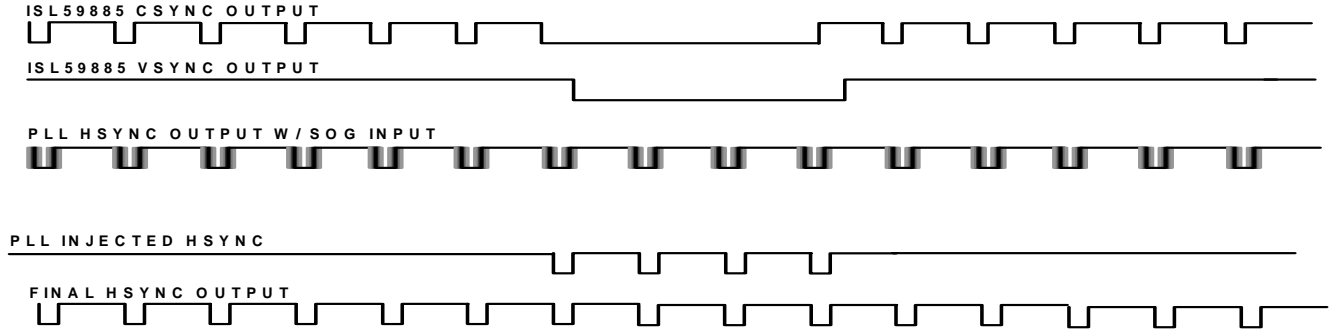


FIGURE 6. CORRECTED H<sub>SYNC</sub> OUTPUT

The replacement H<sub>SYNC</sub> is generated by the PLL H<sub>SYNC</sub> sync'd to a delayed version of the source H<sub>SYNC</sub>. When the incoming SOG or H<sub>SYNC</sub> is lost or corrupted during vertical retrace, the PLL will run open loop and continue to supply a valid H<sub>SYNC</sub>. When a valid source H<sub>SYNC</sub> is detected, this external PLL will resynchronize to the source H<sub>SYNC</sub> and the detector will switch the H<sub>SYNC</sub> output back to the source H<sub>SYNC</sub>. By using a heavy damping in the feedback, the PLL drift will be greatly reduced if not eliminated for the short time duration needed. Thus, even a low cost monitor's PLL will be able to track the re-sync to the source's H<sub>SYNC</sub> with little to no impact on the active video region of the monitor.

**Additional Comments Concerning Digital Sync**

**SOG Sync**

If, at the H<sub>SYNC</sub> switch input, you short the 470pF capacitor from Q4 to the NAND gate and open/removing the 10k resistor, the H<sub>SYNC</sub> will be a C<sub>SYNC</sub> output which can be used for SOG sync source with valid H<sub>SYNC</sub> in V<sub>SYNC</sub> time. The H<sub>SYNC</sub> pulse during the V<sub>SYNC</sub> will be on half the length of the H<sub>SYNC</sub> time and easily decoded with a sync separator.

**C<sub>SYNC</sub> (Digital SYNC Input)**

You might have a logic 5V level C<sub>SYNC</sub> with a bad H<sub>SYNC</sub> signal applied to the input of the sync separator. C<sub>SYNC</sub> could over-drive the input to the ISL59885. If so, it will become necessary to reduce the C<sub>SYNC</sub> input signal level. This can be done simply by using a 1kΩ resistor in series with the input 75Ω resistor to reduce load on the source and attenuate the input to the sync separator.

**Remove C<sub>SYNC</sub> from SOG**

Failing to remove C<sub>SYNC</sub> from SOG will shift the Green level up by the C<sub>SYNC</sub> negative level. On an LCD RGB display,

this will result in a strong green tint on the screen image. This can be corrected by removing the C<sub>SYNC</sub> portion on the SOG so that only the green video signal goes to the LCD. This can be easily done with an ISL4089 as it will DC-restore the SOG video back porch near ground to within 10mV and clip off the C<sub>SYNC</sub> portion of the SOG signal to this 10mV offset. Thus, no unwanted green tint is visible (see Figure 7).

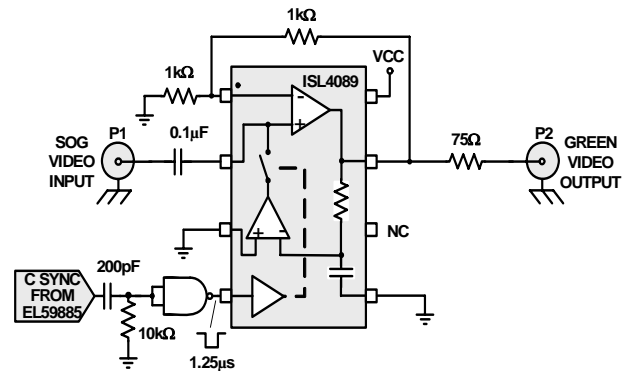


FIGURE 7. RECOVERING GREEN VIDEO FROM SOG

We selected the ISL59885 as the sync separator to cover a wide variety of video sync rates. The problem is the ISL59885 outputs do not have a back porch. Therefore, we have to create the back porch such that the ISL4089 knows when to do the DC restore. This is easy to do by using a RC network (a series 200pF and 10kΩ to ground) to form a 1.25μs pulse plus from the ISL59885 C<sub>SYNC</sub> output. The NAND gate inverts this pulse to form a 1.25μs pulse back porch for the ISL4089 DC restore.

Intersil Corporation reserves the right to make changes in circuit design, software and/or specifications at any time without notice. Accordingly, the reader is cautioned to verify that the Application Note or Technical Brief is current before proceeding.

Schematic and PCB Layout

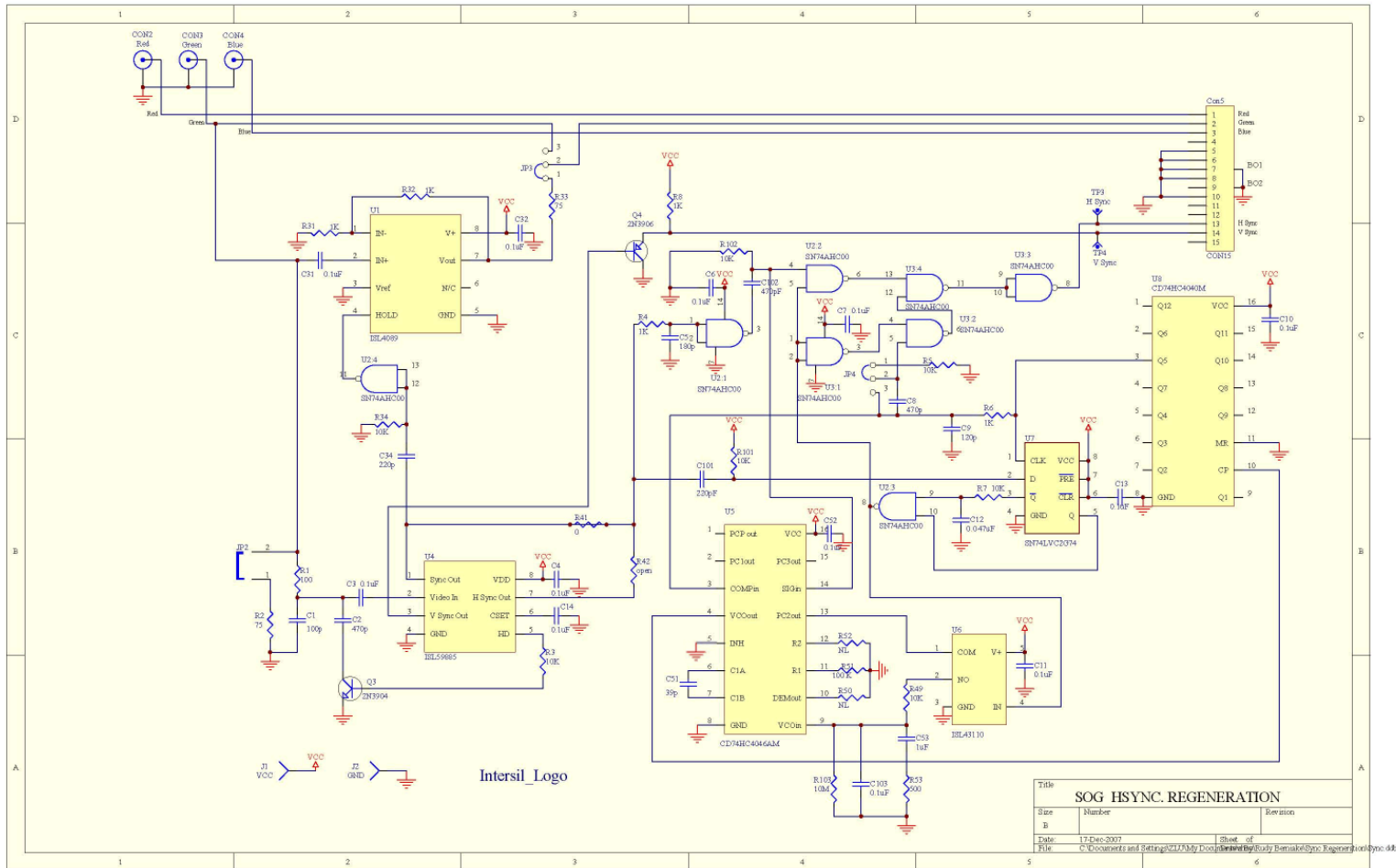


FIGURE 8. COMPLETE SCHEMATIC

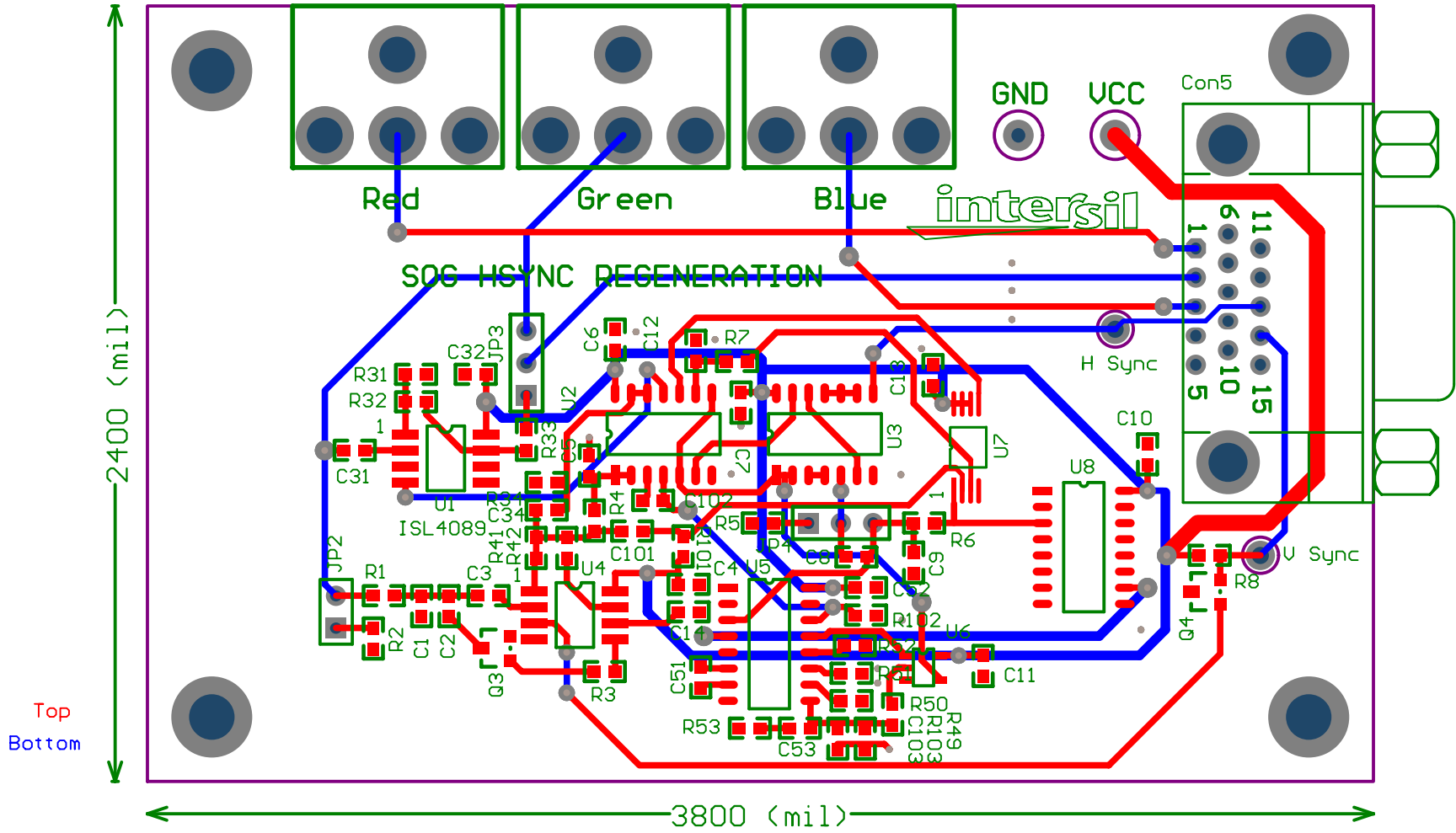


FIGURE 9. PCB LAYOUT