First, we'll consider a progressive-scan VGA (640 × 480) CMOS sensor that connects to a processor's video port, sending 8-bit 4:2:2 YCbCr data at 30 frames/sec.

$$(640 \times 480 \text{ pixels/frame})(2 \text{ bytes/pixel})(30 \text{ frames/sec}) \approx 18.4 \text{ Mbytes/second}$$

This represents the raw data throughput into the processor. Often, there will also be blanking information transmitted on each row, such that the actual pixel clock here would be somewhere around 24 MHz. By using a sensor mode that outputs 16 bits at a time (luma and chroma on one clock cycle), the required clock rate is halved, but the total throughput will remain unchanged, since the video port takes in twice as much data on each clock cycle.

Now, let's switch over to the display side. Let's consider a VGA LCD display with a "RGB565" characteristic. That is, each RGB pixel value is packed into 2 bytes for display on an LCD screen (as we'll discuss soon). The nuance here is that LCDs usually require a refresh rate somewhere in the neighborhood of 50 to 80 Hz. Therefore, unless we use a separate LCD controller chip with its own frame memory, we need to update the display at this refresh rate, even though our sensor input may only be changing at a 30 frames/sec rate. So we have, for example,

$$(640 \times 480 \text{ pixels/frame})(2 \text{ bytes/pixel})(75 \text{ frames/sec refresh}) = 46.08 \text{ Mbytes/second}$$

Since we typically transfer a parallel RGB565 word on every clock cycle, our pixel clock would be somewhere in the neighborhood of 25 MHz (accounting for blanking regions).

Sample Video Application Walk-Through

Let's walk through the sample system of Figure 6.23 to illustrate some fundamental video processing steps present in various combinations in an embedded video application. In the diagram, an interlaced-scan CMOS sensor sends a 4:2:2 YCbCr video stream through the processor's video port, at which point it is deinterlaced and scan-rate converted. Then it passes through some computational algorithm(s) and is prepared for output to an LCD panel. This preparation involves chroma resampling, gamma correction, color conversion, scaling, blending with graphics, and packing into the appropriate output format for display on the LCD panel. Note that this system is only provided as an example, and not all of these components are necessary in a given system. Additionally, these steps may occur in different order than shown here.