

Embedded Systems Project Proposal (webrenderer)

Alex Gajewski (apg2162) and Allison Ghuman (ag3910)

We're planning on building a hardware renderer for a small subset of HTML/CSS. The motivation is that rendering webpages is really slow, and even GPU acceleration incurs a lot of overhead that could probably be cut down with truly specialized hardware.

The circuit is going to render a small patch of the window at a time (however much we can fit in the FPGA's memory), with all the pixels within this patch in parallel. Within each patch, it will draw one object at a time, modifying a pixel buffer on each pass, drawing objects with lower z-index first and moving up. The input will be the current state of the pixel buffer, along with a description of the current object, initially just a width, a height, and some coordinates, but eventually including a background color, maybe a border and border radius, and box shadows if time allows. The output will be a new version of the pixel buffer, after everything has been rendered.

Text would be difficult, and we'll only attempt it if we think we have time. The way this would work is we would have a bunch of bitmaps in DRAM, one for each character, and if an object contains some text, we would have a separate text-rendering circuit that would take one of these bitmaps and draw it onto the current buffer. This could be either one character at a time or a few characters in parallel.

While we of course won't be able to render the entire HTML/CSS specification, we hopefully will be able to get some usable subset working and see some actual speedups relative to CPU rendering at least. A nice quality of this project is that it's very flexible: if basic rendering turns out to be very difficult, we can scale back and stick to colored rectangles. If it turns out to be simpler, we can do more ambitious things like text and box shadows and borders.