Common Resolutions (Full HD and Above)

FPGA developers most often use **1920×1080 (Full HD)** as a baseline resolution. In fact, Xilinx's official Linux setup for their FPGA boards defaults to 1920×1080 and explicitly recommends "using the 1920x1080 resolution for the most robust desktop experience" [1]. (The quoted document noted that 4K is *possible* but suggested 1080p as the safer default.) Modern laptops and monitors commonly ship at 1080p, so it is indeed a widespread starting point. That said, 1920×1080 can feel cramped when viewing large waveforms. In practice many engineers now favor higher resolutions. For example, one FPGA user lamented that on a 27″ 4K monitor *"everything is so freaking small it's actually a bit hard on the eyes"* once UI scaling is applied, whereas a 24″ 1080p display looks coarse [2]. (No standard "1980×1080" exists; the industry uses 1920×1080 for Full HD.)

- **Standard Full HD (1920×1080):** This is a very common resolution. It is widely supported and used by Vivado/ModelSim/GtkWave GUIs. Xilinx and others often assume 1080p for optimal UI performance [1]. It offers reasonable readability of signals on a single monitor.
- **Lower Resolutions (e.g. 1280×720):** On low-memory or embedded Linux systems (like the Xilinx ZCU104 board), even 1080p can be too much. Xilinx notes that *"resolutions higher than 1080p are not supported on the GNOME Desktop"* of that board, and even recommends dropping to 1280×720 for better responsiveness [3]. Thus, if system resources are tight, using a lower resolution may improve speed, at the cost of reduced screen real estate.
- **High-DPI Scaling:** On high-DPI or 4K displays, GTKWave's UI may need scaling. Some FPGA users report GTKWave looks blurry on Retina screens unless properly scaled. In general, running the software at 4K (3840×2160) without UI scaling can make text and waveforms very small. Vivado and other EDA GUIs often have scaling issues at 4K, so 1080p is still recommended for ease of use [1] [2].

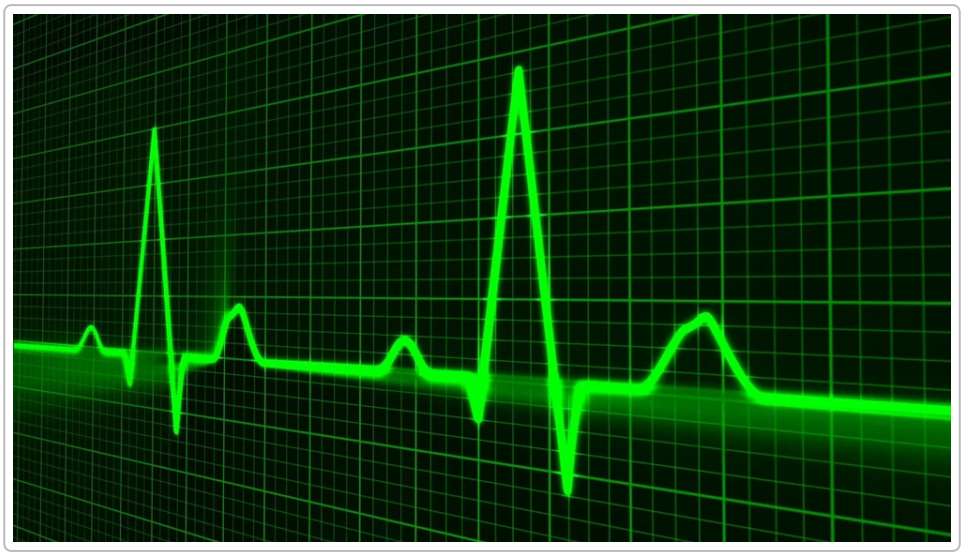

Figure: High-resolution waveform data is easier to examine on larger monitors. Users often prefer more pixels for waveform windows (source: an example heartbeat waveform image).

Beyond Full HD: 2K, 4K, and Ultrawide Displays

Many FPGA professionals extend their workspace with higher resolutions and/or multiple monitors. Higher pixel counts allow more signals and time span to be visible at once. Common choices include **2560×1440 (2K/QHD)** and **3840×2160 (4K/UHD)**. For example, a 27″ monitor at 2560×1440 provides ~77% more pixels than 1080p, making waveforms and text much crisper. A 4K screen (especially 32″ or larger) quadruples the pixel count of 1080p, at the expense of needing 200% UI scaling to keep fonts legible. In community forums, FPGA users praise high-resolution setups: "my favourite tool is a *high-resolution monitor* to actually see the waveforms on" [4]. One user summed it up:

- **Ultrawide Screens:** A popular extreme is the 49″ Dell Ultrasharp U4919DW (5120×1440 resolution, 32:9 aspect). This one user enthused *"49 inches. 32:9 format. 5120×1440 resolution. So much room for waveforms."* [5]. Ultrawide or superwide (e.g. 5120×1440 or 3440×1440) give vast horizontal space for long time-scale waveforms and side-by-side windows.
- **Multi-Monitor Setups:** Alternatively, many engineers use dual or triple monitors (each 1080p or 1440p). In one discussion a user noted *"I prefer multi-monitor…with two monitors it's easier to organize windows…"* instead of one giant display [6]. Multi-monitor setups let you keep waveform, source code, and other tools visible simultaneously. For example, one engineer uses three 27″ 1440p screens for ample workspace.
- **4K Caveats:** High DPI monitors offer sharpness but require proper scaling. A user commented that at 4K resolution they had to scale UI elements up, otherwise *"everything is so…small it's hard on the eyes."* [2]. If using 4K, ensure the OS/UI is scaled (e.g. 150–200% text scaling) or use a sufficiently large screen (40″+) so details remain readable. Vivado's GUI, for instance, does not automatically scale on 4K, leading many engineers to stick with 1080p or use X.org scaling tricks [2] [3].

In practice, **1920×1080 remains a common baseline**, but experienced FPGA developers often upgrade to at least 2560×1440 or beyond for debugging. A Reddit FPGA designer wryly noted that regular 24″ 1080p monitors *"look like an old Atari"* compared to modern high-res displays [7].

Screen Real Estate vs. Usability

While more pixels give more waveform view, there is a trade-off in usability. Some users report that extremely large or ultrawide displays can be unwieldy for single applications: one comment observed that after running a five-monitor setup, only three were truly useful and organizing windows across five screens was difficult [6]. Another user stuck with a modest 17″@1024×768 display due to vision issues, and used Windows scaling for comfort [8]. In summary, what matters most is readable text and waveform lines. A larger display at higher resolution must have appropriately scaled fonts and marker labels, or the extra pixels won't help.

Professionals often balance resolution with practical concerns: for example, using a 27″ QHD monitor or dual 24″ monitors. According to community consensus, **1920×1080 is common and adequate**, but *not necessarily optimal* if you can afford higher-res. As one developer put it, their favorite upgrade was simply getting a bigger, sharper screen: *"my favourite tool is a high resolution monitor to actually see the waveforms on."* [4].

Recommendations

- **If you're starting out:** 1920×1080 is fine and widely supported (as Xilinx notes [1]). It works with all FPGA tools without UI issues.

- **For heavy waveform analysis:** Consider at least 2560×1440 or 3840×2160. More horizontal pixels allow viewing longer time spans and more signals simultaneously. Higher DPI yields crisper waveforms [4], but remember to scale UI elements for readability.
- **Ultrawide or multiple monitors:** If budget allows, an ultrawide (e.g. 5120×1440) or a dual-monitor setup can dramatically increase productive space [5] [6]. Many FPGA engineers split code on one screen and GTKWave on another. Tools like window managers or the monitor's own splitting features can help use extra space effectively.
- **Avoid too-low resolutions:** Resolutions below 1080p (e.g. 1366×768) are generally insufficient for modern EDA tools on large designs, as important labels and text become tiny or truncated.

In short: 1080p is the industry standard baseline and is often used, but for **GTKWave** debugging a higher-resolution display (2K, 4K, or ultrawide) will improve wave-data readability and workflow. These points are confirmed by official Xilinx documentation [1] and by FPGA developer forums [4] [5] [6], which unanimously emphasize that more screen space (higher resolution or multiple screens) makes waveform viewing and debugging much easier.

Sources: Official Xilinx documentation on display resolution [1] [3]; FPGA developer and forum discussions (Xilinx, Digilent, Reddit) [4] [9] [6] [10] [2]. These highlight both the prevalence of 1920×1080 and the real-world benefits of larger/high-DPI displays for GTKWave use.

---

[1] [3] Getting Started with Certified Ubuntu 20.04 LTS for Xilinx Devices - Xilinx Wiki - Confluence
https://xilinx-wiki.atlassian.net/wiki/spaces/A/pages/2037317633/Getting+Started+with+Certified+Ubuntu+20.04+LTS+for+Xilinx+Devices

[2] [5] [6] [7] [8] [9] [10] Ok this is not really related to FPGA: sbut who else wants this monitor for simulation : r/FPGA
https://www.reddit.com/r/FPGA/comments/cb4oyd/ok_this_is_not_really_related_to_fpgas_but_who/

[4] Your Favorite VCD File Viewer : r/FPGA
https://www.reddit.com/r/FPGA/comments/1dcp6f/your_favorite_vcd_file_viewer/