Mixing

// create a value of 0. or 1. from the value of x wrt edge:
float t = step(float edge, float x);

// create a value in the range 0. to 1. from the value of x wrt edge0 and edge1:
float t = smoothstep(float edge0, float edge1, float x);

// use the returned value from step( ) or smoothstep( ) to blend value0 to value1:
T out = mix(T value0, T value1, float t);

“SmoothPulse” in a Fragment Shader

void main() {
  float f = fract( uA*vX);
  float t = smoothstep(0.5-uP-uTol, 0.5-uP+uTol, f) - smoothstep(0.5+uP-uTol, 0.5+uP+uTol, f);
  vec3 rgb = vLightIntensity * mix(WHITE, vColor, t);
  gl_FragColor = vec3(rgb, 1.);
}

Fun With One

Moral: There are many ways to turn \([0. - 1.]\) into \([0. - 1.]\)
Why Do These Two Curves Match So Closely?

The Taylor Series expansion of \( y = \sin\left(\frac{\pi}{2}x\right) \) around \( x = 0.5 \) is:

\[
y = \left(\frac{1}{2} - \frac{\pi^2}{4} + \frac{\pi^4}{96}\right) x + \left(\frac{\pi^2}{2} - \frac{\pi^4}{16} + x\right) \left(\frac{\pi^2}{8} - x\right) \left(\frac{\pi^2}{12}\right)
\]

which is pretty close to: \( y = 3x^2 - 2x^4 \)

Cubic vs. Quintic

Both go from 0 to 1.
Both have initial and final slopes of 0.
The quintic has initial and final curvatures of 0.