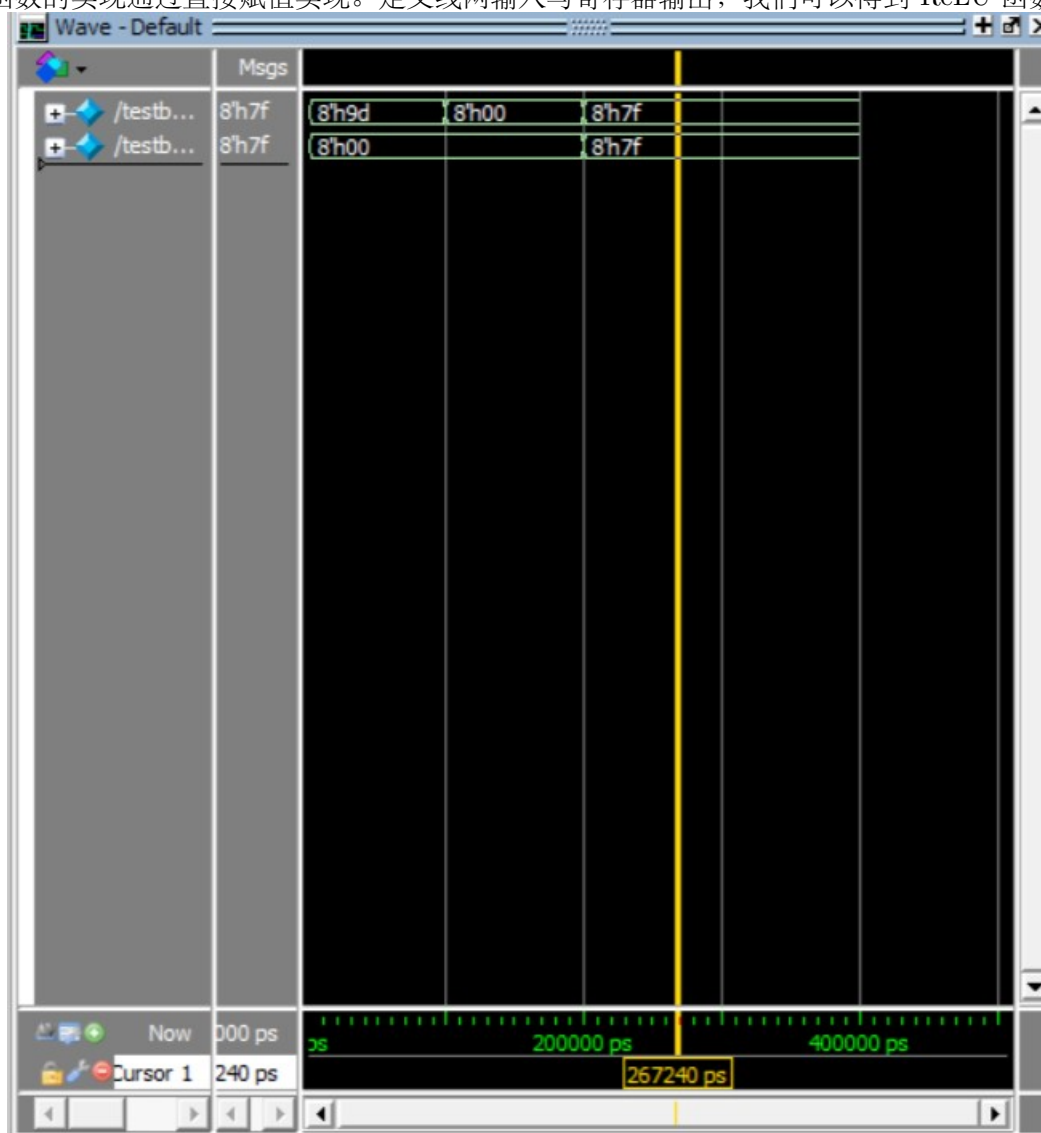


1 ReLU 函数的实现

ReLU 函数的实现通过直接赋值实现。定义线网输入与寄存器输出，我们可以得到 ReLU 函数的直接实现。



图中可以发现，当输入小于等于零时，仿真波形输出为 0. 大于零时输出与输入相同。

2 SiLU 函数的实现

SiLU 函数的表达式为 $f(x) = \frac{x}{1 + e^{-x}}$ ，因此在 Verilog 语言中，我们采用线性逼近的方式近似实现 SiLU 函数。通过查表法实现。实现波形如下：



可以看到，该波形符合线性逼近。

3 SiLU 函数多位宽线性插值

1. 在同学的帮助下，我们发现需要明确输入，中间和输出信号的符号性。默认为无符号型的数将会出现运算（乘法）的问题。
2. 首先，我们规定 16 位，24 位和 8 位的变量。最后过程的乘法可能会出现溢出问题，因此我们采用 24 位存储。
3. 其次，在处理过程中，我们将数字拓展到 16 位。具体的操作为：（1）将数存储到 24 位寄存器变量中。由于存储的特性为高位补 0，而我们的中间变量应该仅应该拓展小数位而非整数位。因此将输入数据左移 8 位。
4. 接着，采用移位的方法来实现乘法的功能。具体的方式请参照代码。而不同函数情况的界定，也将其表示成 4 位整数位和 12 位小数位组合而成的 16 位变量进行划分，从而实现不同的功能。
5. 在实现操作前，我们将输入数据绝对值化，即探测寄存器最高位（符号位）是否为 1。随后在最后得到结果后再判断是否需要输出 $1 - f(|x|)$ 。

