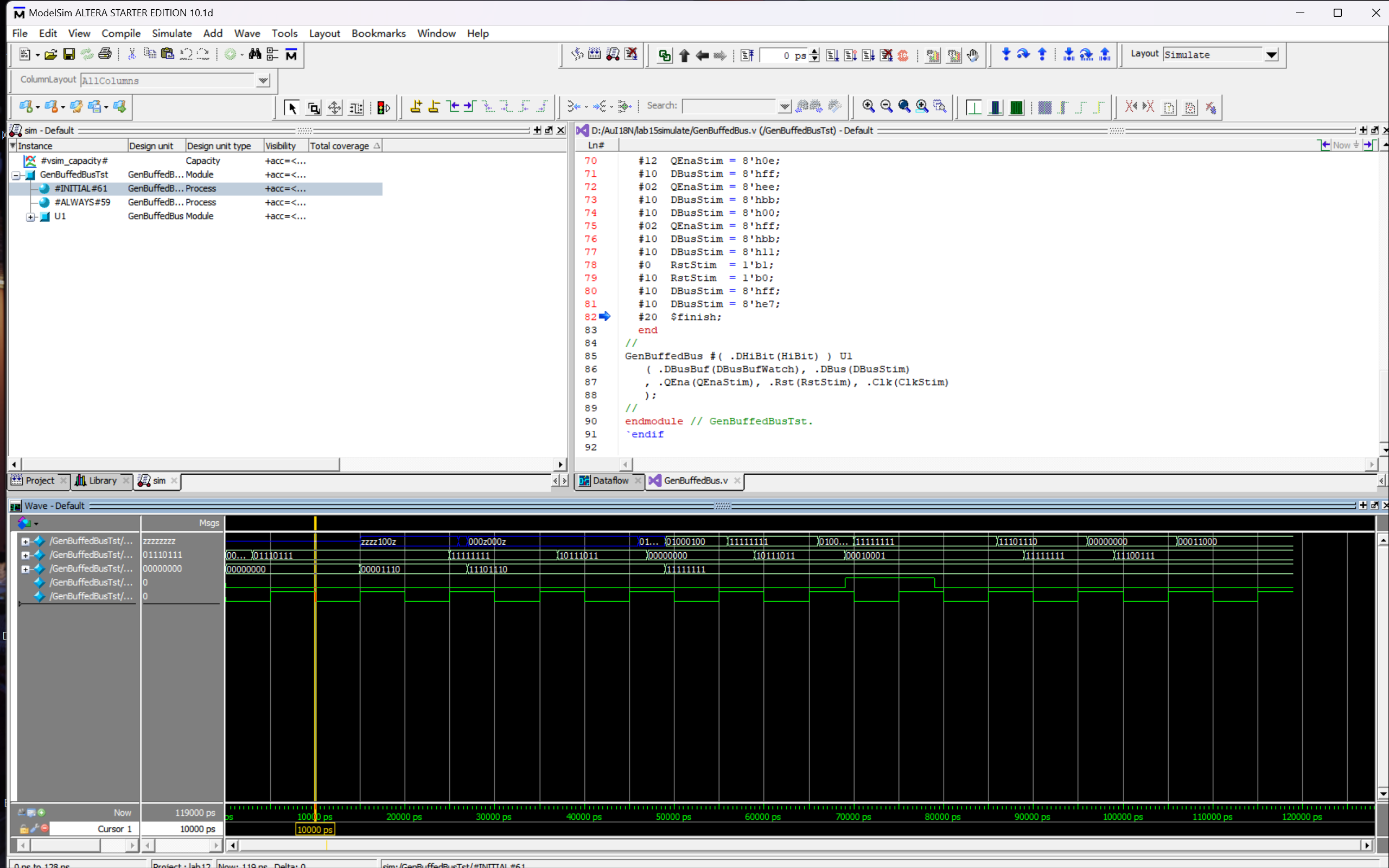
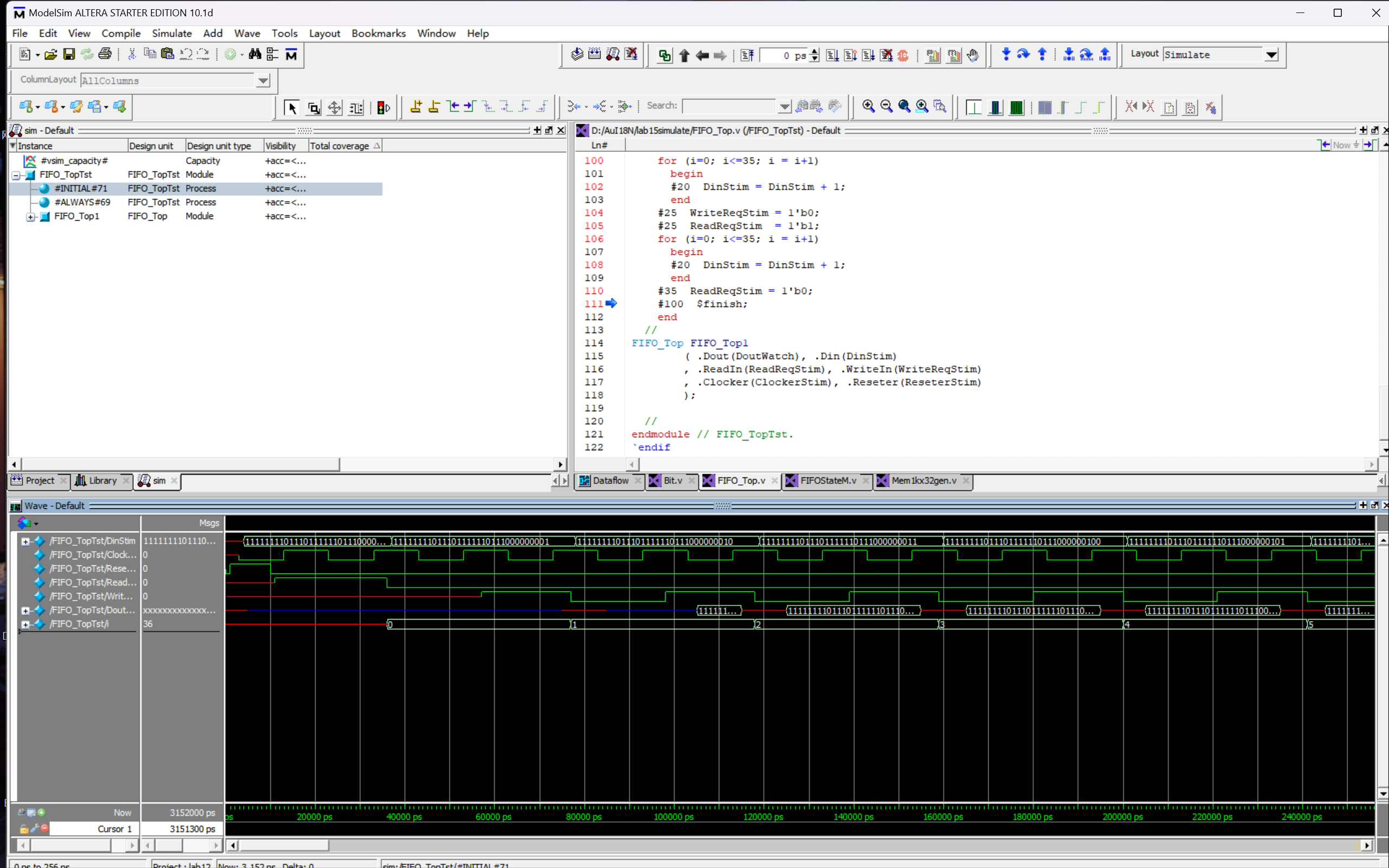
generate是调用过程控制语句的并行结构。generate里可以使用条件语句if或者case，但

通常都是使用for来产生对象数组。generate语句里可以包含门级器件，模块实例，寄存器，线，--连续赋值表达式，always或initial块等。但是在generate语句里不允许再包含generate语句





思考1：在 Verilog 中，一个整型变量和一个 genvar 类型的变量的名称是可以相同的。Genvar 变量是用于 generate 块中的循环和条件语句，用于生成不同的实例。整型变量用于存储整数值。这两种类型的变量在作用域上是不同的，因此它们可以使用相同的名称而不会冲突。

思考2：在 Verilog 中，generate 块具有自己的作用域（scope）。这意味着在 generate 块内部定义的变量、实例以及其他命名对象在块外部是不可见的，并且不会与外部作用域中的对象发生冲突。这为 generate 块提供了一种独立的命名空间，使得你可以在块内部使用与外部作用域相同的名称而不会产生冲突。需要注意的是，虽然 generate 块具有独立的作用域，但它仍然可以访问外部作用域中的对象。如果在 generate 块内部引用了外部作用域中的对象，它们会被当作外部作用域的对象处理。但是，在 generate 块内部定义的对象不会影响外部作用域中的对象。