

计算机组成与结构专题实验

实验报告

第一次 节拍发生器

一、实验目的

1. 掌握节拍脉冲发生器的设计方法，深入理解其工作原理。
2. 通过实际操作，能够熟练运用相关知识，设计并实现节拍脉冲发生器，进一步加强对计算机组成与结构的理解。

二、实验原理

计算机能够依据预设顺序执行操作或运算，关键在于其控制部分可按顺序发出相应控制信号，这要求计算机必须配备时序电路，以确保各操作与运算按精确的时间顺序依次进行。

1. 连续节拍发生电路

连续节拍发生电路由 4 个 D 触发器构成，能够产生 4 个等间隔的时序信号 $T1 \sim T4$ 。其中，CLK 为时钟信号。当 RST 为低电平时， $T1$ 输出为 “1”， $T2$ 、 $T3$ 、 $T4$ 输出为 “0”。一旦 RST 从低电平变为高电平， $T1 \sim T4$ 将在 CLK 的输入脉冲作用下，依次轮流输出正脉冲，电路随之进入连续运行状态（EXEC）。

2. 单步节拍发生电路

单步节拍发生电路与连续节拍发生电路中的 RST 信号相关。每当 STEP（即 RST）出现一个脉冲后，该电路仅输出一组 $T1$ 、 $T2$ 、 $T3$ 、

T4 节拍信号，并且在 STEP 出现下一个脉冲之前，电路不会再输出新的节拍信号。

三、实验任务

实验任务 1：连续节拍发生电路设计实验

根据连续节拍发生电路的电路图，将其绘制并编译，随后下载至 FPGA 中。选择实验模式 1，具体连接方式为：用单线一端连接主系统时钟源(4Hz)，另一端连接 J17 的 CLKB0 端，同时键 8 用于控制 RST1。当 RST1 处于高电平时，发光管 1、2、3、4 将分别显示出 T1、T2、T3、T4 的输出电平。最后，将实验结果与仿真波形图进行比较分析。

实验任务 2：单步节拍发生电路设计实验

根据单步节拍发生电路的电路图，将其绘制并编译，随后下载至 FPGA 中。该电路主要用于对微程序进行单步运行调试。具体表现为：每当 RST1 出现一个负脉冲后，电路仅输出一组 T1、T2、T3、T4 节拍信号，并且在 RST1 出现下一个负脉冲之前，不会输出新的节拍信号。选择实验模式 1，时钟频率设定为 4Hz（选择范围是 1Hz-20MHz），键 8 用于控制 RST1。每出现一个负脉冲，发光管 1、2、3、4 将分别显示一次 T1、T2、T3、T4 的输出电平。最终，将实验结果与仿真波形图进行比较分析。

实验任务 3：单步/连续节拍发生电路设计实验

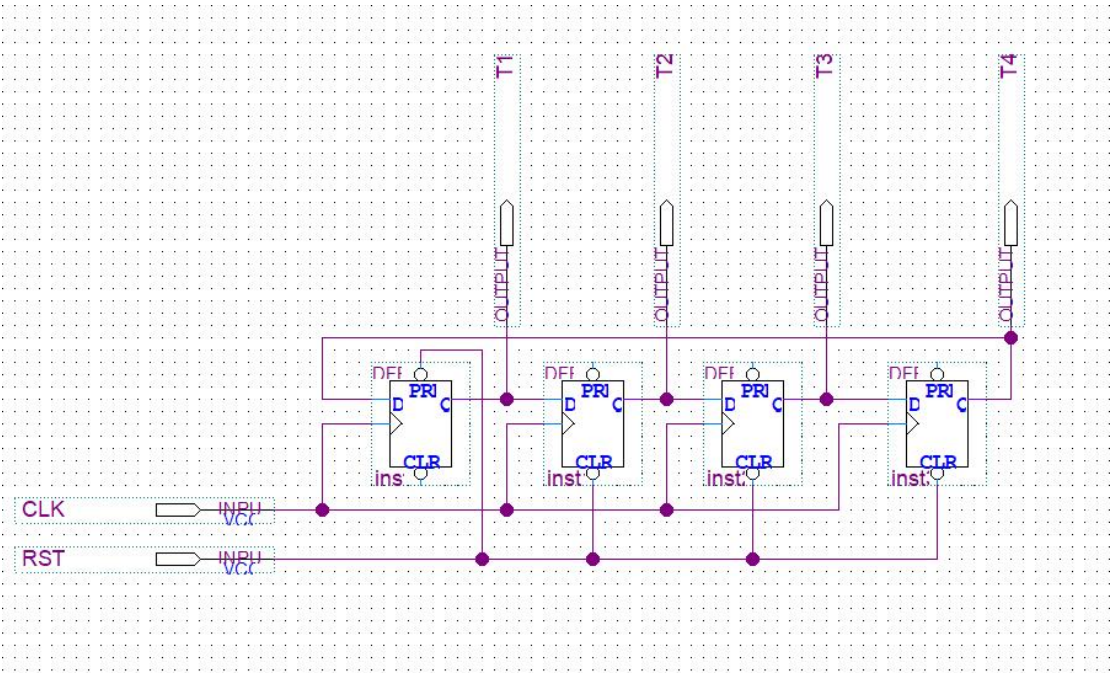
根据单步/连续节拍发生电路的电路图，将其绘制并编译，随后下载至 FPGA 中。在该电路中，增加了两个 2-1 多路选择器，S0 作为单步或连续节拍发生的控制信号。具体工作模式为：当 S0=0 时，选

择单步运行方式；当 S0=1 时，选择连续运行方式。选择实验模式 1，时钟源同上一个实验一样接 4Hz，键 8 用于控制 RST1，键 7 用于控制 S0。发光管 1、2、3、4 将分别显示 T1、T2、T3、T4 的输出电平。最终，将实验结果与仿真波形图进行比较分析。

四、 实验步骤及结果

实验任务 1：连续节拍发生电路设计实验

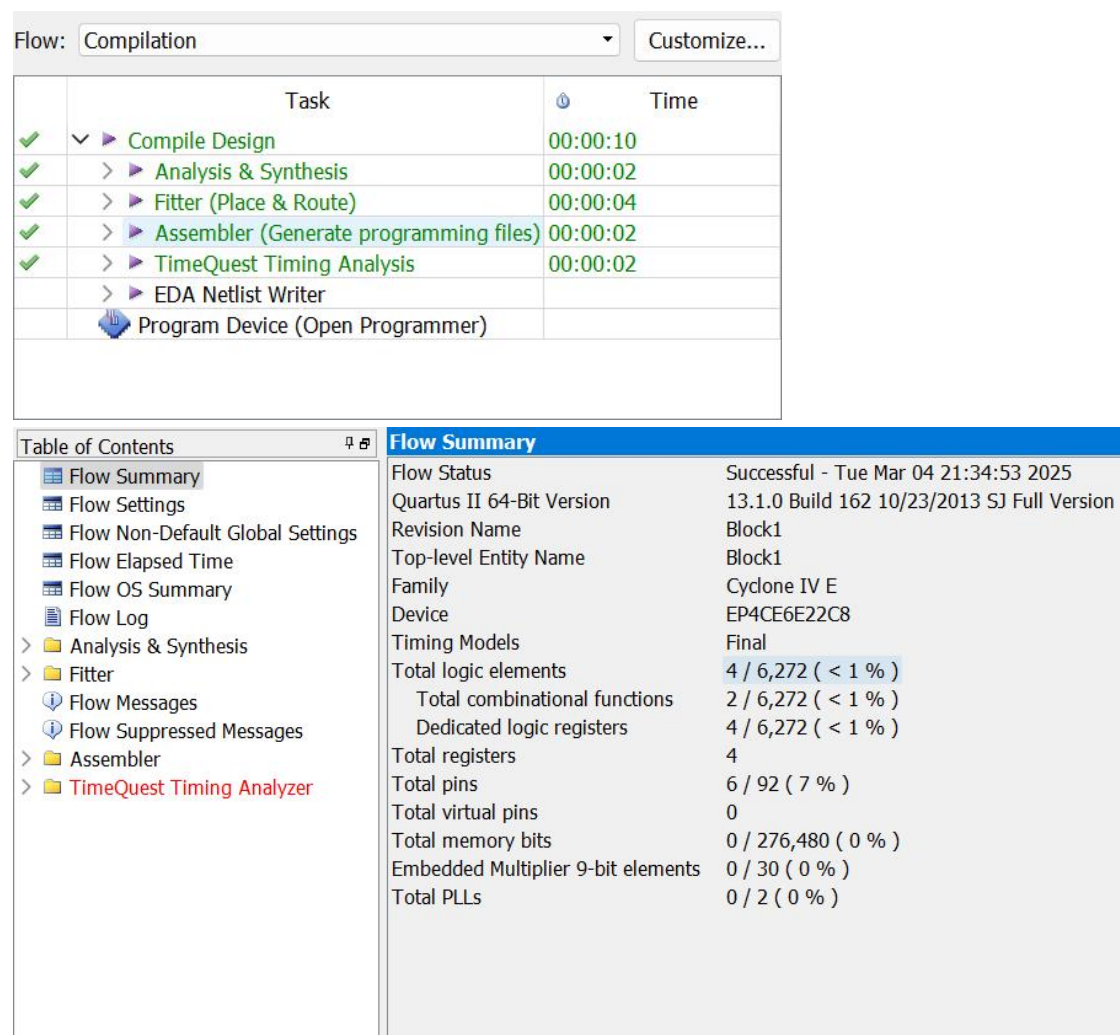
绘制电路图：



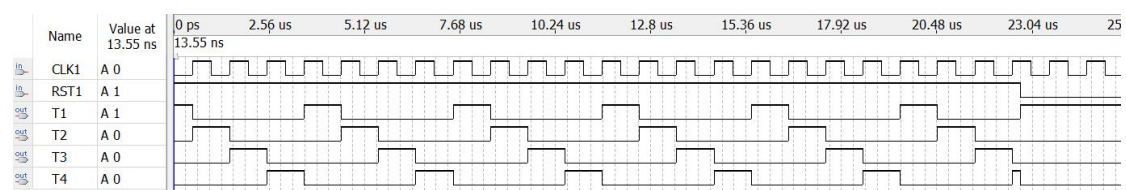
引脚绑定：

Pin	Status	From	To	Assignment Name	Value	Enabled	Entity	C
1	✓		in CLK1	Location	PIN_90	Yes		
2	✓		in RST1	Location	PIN_53	Yes		
3	✓		out T1	Location	PIN_101	Yes		
4	✓		out T2	Location	PIN_100	Yes		
5	✓		out T3	Location	PIN_113	Yes		
6	✓		out T4	Location	PIN_105	Yes		
7		<<new>>	<<new>>	<<new>>				

编译：



仿真：

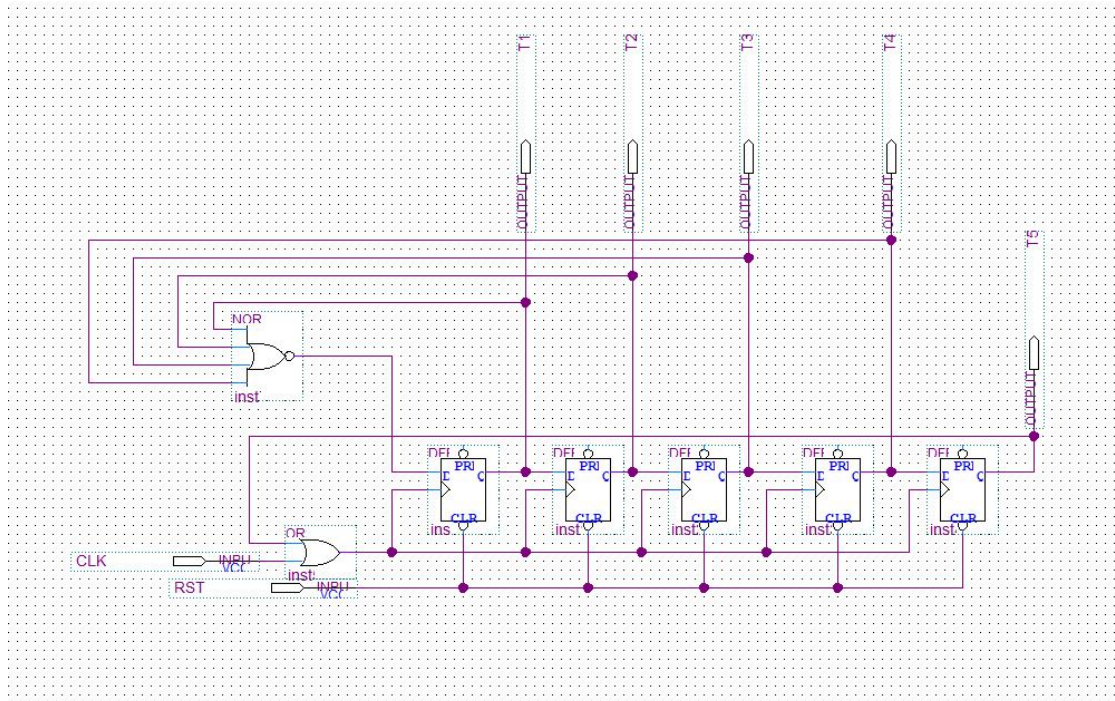


下载至 FPGA，根据实验任务对开发板进行操作，实验结果与预期结果一致。







分析：RST1 从低电平转换至高电平，T1 随之从高电平转换至低电平，而 D 触发器通过将状态反转，使 T2 变为高电平，依次下去，直到 T4 也变成高电平，当 T4 变为低电平后，其输出作为第一个 D 触发器的输入，使 T1 再变为高电平，从而产生连续节拍发生的效果。

实验任务 2：单步节拍发生电路设计实验

绘制电路图：



引脚绑定:

	statu:	From	To	Assignment Name	Value	Enabled	Entity
1	✓		 RST1	Location	PIN_53	Yes	
2	✓		 T1	Location	PIN_101	Yes	
3	✓		 T2	Location	PIN_100	Yes	
4	✓		 T3	Location	PIN_113	Yes	
5	✓		 T4	Location	PIN_105	Yes	
6	✓		 CLK1	Location	PIN_90	Yes	

编译:


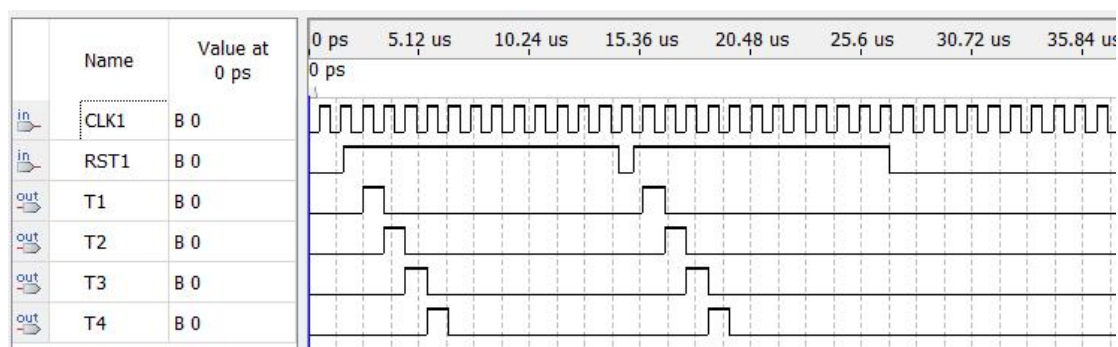
	Task	Time
✓	▼ ▶ Compile Design	00:00:10
✓	> ▶ Analysis & Synthesis	00:00:02
✓	> ▶ Fitter (Place & Route)	00:00:04
✓	> ▶ Assembler (Generate programming files)	00:00:02
✓	> ▶ TimeQuest Timing Analysis	00:00:02
	> ▶ EDA Netlist Writer	
	 Program Device (Open Programmer)	

Table of Contents	Flow Summary
Flow Summary	Flow Status Successful - Tue Mar 04 21:40:22 2025
Flow Settings	Quartus II 64-Bit Version 13.1.0 Build 162 10/23/2013 SJ Full Version
Flow Non-Default Global Settings	Revision Name Block1
Flow Elapsed Time	Top-level Entity Name Block1
Flow OS Summary	Family Cyclone IV E
Flow Log	Total logic elements 6 / 6,272 (< 1 %)
> Analysis & Synthesis	Total combinational functions 2 / 6,272 (< 1 %)
> Fitter	Dedicated logic registers 5 / 6,272 (< 1 %)
Flow Messages	Total registers 5
Flow Suppressed Messages	Total pins 7 / 92 (8 %)
> Assembler	Total virtual pins 0
> TimeQuest Timing Analyzer	Total memory bits 0 / 276,480 (0 %)
	Embedded Multiplier 9-bit elements 0 / 30 (0 %)
	Total PLLs 0 / 2 (0 %)
	Device EP4CE6E22C6
	Timing Models Final

仿真：

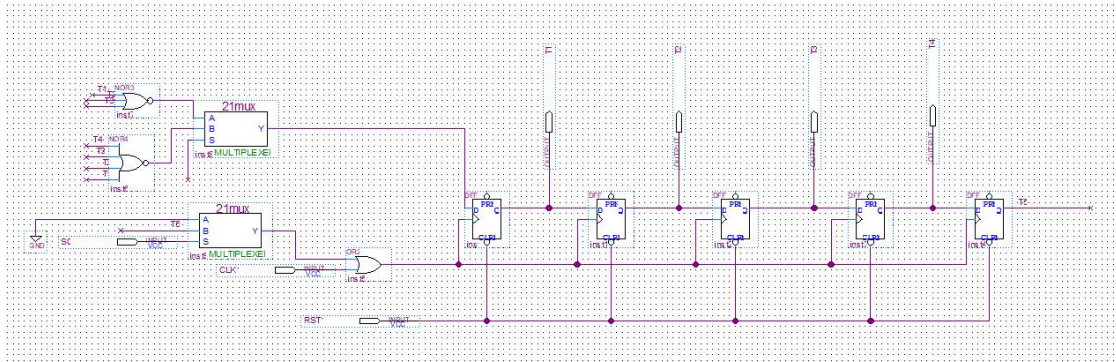


下载至 FPGA，根据实验任务对开发板进行操作，实验结果与预期结果一致。

分析：当按下按键后 RST 从低电平转为高电平，开始正常工作，最初 T1-4 均为低电平，第一个触发器的输入为高电平，在第一个时钟上升沿到来后 T1 变为高电平，之后从 T2-5 依次变为高电平，在 T5 变为高电平后，时钟信号与 T5 进行或运算始终为高电平，输入不发生变化，直到下一次复位即按下 RST1。

实验任务 3：单步/连续节拍发生电路设计实验

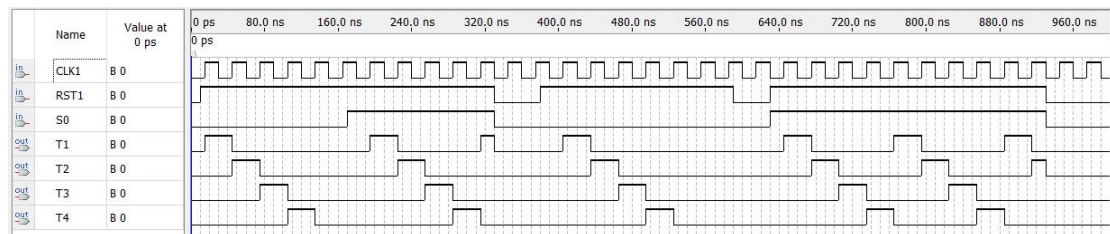
绘制电路图：



编译：

Tasks			⌵ ⌵ ⌵
Flow: Compilation			Customize...
	Task	Time	
✓	✓ ▶ Compile Design	00:00:12	
✓	> ▶ Analysis & Synthesis	00:00:02	
✓	> ▶ Fitter (Place & Route)	00:00:04	
✓	> ▶ Assembler (Generate programming files)	00:00:02	
✓	> ▶ TimeQuest Timing Analysis	00:00:02	
✓	> ▶ EDA Netlist Writer	00:00:02	
	🔧 Program Device (Open Programmer)		

仿真：



S0=0 时，为单步运行模式，S0=1 时，为连续运行模式。下载至 FPGA，根据实验任务对开发板进行操作，实验结果与预期结果一致。

分析：运用 2-1 多路选择器实现单步/连续节拍发生器的切换，当 S0 为低电平时输出为 B 的值，为高电平时，输出为 A 的值，从而实现模式切换。当键 7 控制 S0 输出低电平时，选择的运行模式为单步运行。此时按下键 8 使输出为高电平时，出现一个负脉冲，此时发光管 D1、D2、D3、D4 分别显示 T1、T2、T3、T4 的输出电平一次，当按下键 8

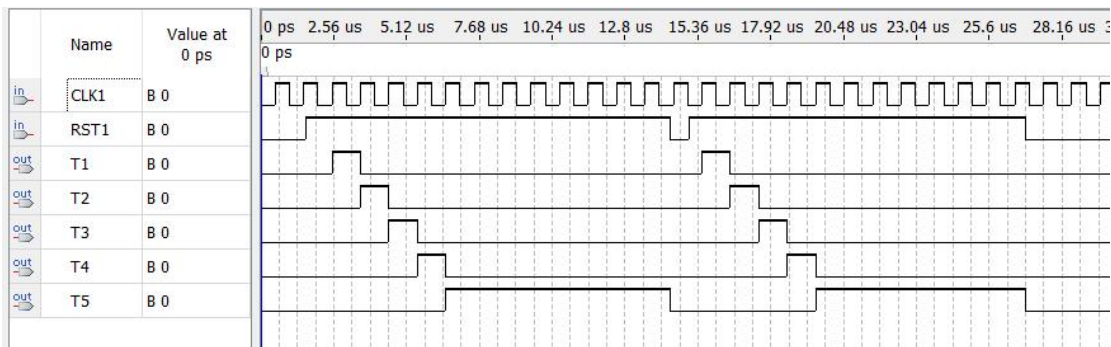
使输出为低电平时，发光管不亮。当键 7 控制 S0 输出高电平时，选择的运行模式为连续运行。当按下键 8 使 RST1 由低电平变为高电平后，T1-4 将在 CLK1 的输入脉冲作用下，周期性地轮流输出正脉冲，体现为 D4、D3、D2、D1 轮流亮光。

五、 实验总结及问题分析

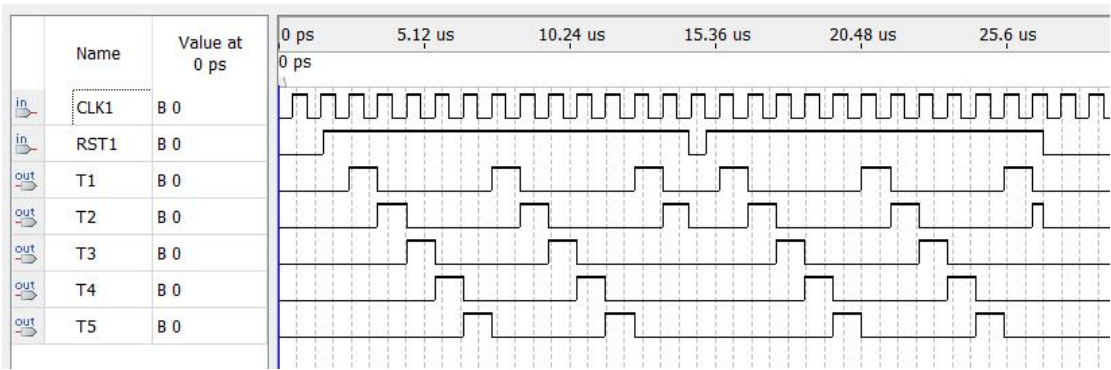
1. 给出每个电路的原理图分析说明为何能产生所需节拍。

实验电路图如实验步骤中所示。分析详见实验原理和实验任务部分，对节拍的产生进行了详尽的分析。

2. 对于实验任务 2，对比没有 t5 输入时的仿真时序图，借此说明 t5 的作用。



有 T5 输入时的仿真时序图



没有 T5 输入时的仿真时序图

可见，电路效果变为五个节拍的连续节拍生成器。对比包含 T5

输入的时序图可知，T5 的作用是在一个节拍生成周期后，使每个触发器的时钟信号输入均为高电平，从而使整个电路停留在 T1、T2、T3、T4 为 0，T5 为 1 的状态。

3. 思考题 1，单步运行与连续运行有何区别，它们各自的使用环境怎样？

单步运行是逐步执行程序，每次执行一条指令或一个语句后暂停，等待用户命令继续执行下一条指令。这种方式主要用于调试程序，帮助开发者观察程序的执行过程，逐步检查变量值和程序状态，从而快速定位问题。单步运行适用于开发和调试阶段，尤其是当程序出现逻辑错误或难以复现的问题时，通过逐条跟踪执行流程，可以更清晰地理解程序的行为。

连续运行则是一次性执行完整程序，从开始到结束不间断地运行，无需用户干预。这种方式适用于性能测试和实际运行环境，因为它能够快速完成任务，适合处理大量数据或执行复杂计算。连续运行通常用于生产环境，例如服务器、自动化脚本或定时任务等，这些场景需要程序高效、自动化地运行。

选择哪种运行方式取决于具体需求：如果目标是调试程序、排查问题或理解程序逻辑则选择单步运行；而如果需要测试程序性能或在实际环境中运行程序，则应选择连续运行。

4. 思考题 2，在实验任务 3 时，如何进行单步和连续运行工作方式的切换？

在实验任务 3 时，运用 2-1 多路选择器实现单步/连续节拍发生

器的切换，当 S0 为低电平时输出为 B 的值，从而可以等效为单步运行时的电路图，为高电平时，输出为 A 的值，可以等效为连续运行时的电路图，从而实现模式切换。