

**本科生实验报告**

**（2024~2025第一学期）**

**实验课程 数字电路与逻辑设计**

**学院名称 核自院**

**专业名称 测控技术**

**学生姓名**

**学生学号**

**指导教师 王 敏**

**实验地点 6C802**

**实验成绩**

**二〇二四 年 十一 月 —— 二〇二五 年 一 月**

**说 明**

1. 正文部分内容采用宋体，小四，1.25倍行间距；
2. 程序代码部分采用times new roman，五号，单倍行间距；
3. word中可插入文本框，程序代码写在文本框中；
4. “实验步骤”部分重点突出实验项目的设计思路，可用框图说明，须必要文字描述；
5. “实验程序”部分不能只粘贴代码，须对代码做必要的解释；
6. “实验结果”部分指的是对实验现象的描述，可附图；
7. 实验课程结束后一周内提交实验报告电子档和纸质档。

目录

[实验一 计时器 1](#_Toc183803403)

[1.1 实验目的 1](#_Toc183803404)

[1.2 实验内容 1](#_Toc183803405)

[1.3 实验步骤 1](#_Toc183803406)

[1.3.1 实验框架 1](#_Toc183803407)

[1.3.2 主要代码 2](#_Toc183803408)

[1.4 实验程序 2](#_Toc183803409)

[1.5 实验结果 3](#_Toc183803410)

[实验二 测量占空比 4](#_Toc183803411)

[2.1实验目的 4](#_Toc183803412)

[2.2 实验内容 4](#_Toc183803413)

[2.3 实验步骤 4](#_Toc183803414)

[2.3.1 实验框架 4](#_Toc183803415)

[2.3.1 时序图 5](#_Toc183803416)

[2.3.2 主要代码 5](#_Toc183803417)

[2.4 实验程序 5](#_Toc183803418)

[2.5 实验结果 6](#_Toc183803419)

[实验三 数码管静态显示 7](#_Toc183803420)

[3.1 实验目的 7](#_Toc183803421)

[3.2 实验内容 7](#_Toc183803422)

[3.3 实验原理 7](#_Toc183803423)

[3.3.1数码管简介 7](#_Toc183803424)

[3.3.2静态显示 8](#_Toc183803425)

[3.3.3 74HC573简介 8](#_Toc183803426)

[3.4 硬件资源 9](#_Toc183803427)

[3.5实验程序 10](#_Toc183803428)

[3.6实验结果 11](#_Toc183803429)

[**实验四 动态显示** 12](#_Toc183803430)

[4.1 实验目的 12](#_Toc183803431)

[4.2 实验内容 12](#_Toc183803432)

[4.3 实验原理 12](#_Toc183803433)

[4.3.1 数码管动态显示简介 12](#_Toc183803434)

[4.4实验程序 13](#_Toc183803435)

[4.5实验结果 14](#_Toc183803436)

[实验心得 16](#_Toc183803437)

# 实验一 计时器

## 1.1 实验目的

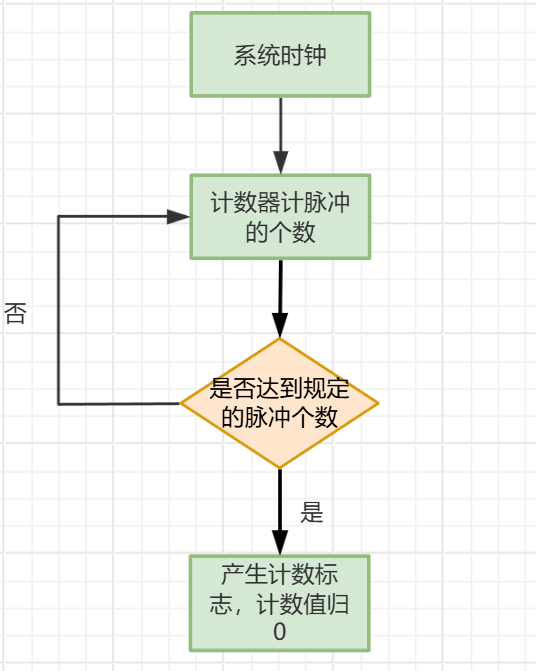
1. 了解计时器工作原理；
2. 依据计时器工作原理，使用Verilog语言对50MHz时钟计时，分别计时1us、1ms、1s。
3. 根据TB文件模板，学会编写仿真文件并仿真计数器实验。

## 1.2 实验内容

本实验的任务是使用寄存器对50MHz时钟进行计时，编写Verilog代码并使用modelsim仿真。

## 1.3 实验步骤

### 1.3.1 实验框架



* **系统时钟：**在仿真代码中通过always语句与#延迟符号编写出50MHz时钟，通过例化连接到工程系统时钟。
* **计数器计脉冲个数：**使用在循环语句中，当时钟脉冲上升沿时，计数寄存器加一。
* **是否达到规定的脉冲个数：**根据时钟频率与周期的关系计算出实验要求计时的脉冲个数，通过if判断语句判断是否达到计数要求。
* **产生计数标志，计数值归0：当计数达到要求时，计数标志寄存器赋1，计数寄存器赋0**。

### 1.3.2 主要代码

reg，assign，always@(posedge i\_clk,posedge i\_rst)，if else if else，initial

## 1.4 实验程序

**例：后续需要将此处示例代码删除，仅需添加主要代码即可**

module XXX(

input i\_XXX,

output o\_XXX

);

reg ro\_XXX;

assign o\_XXX = ro\_XXX

always @(State)

begin

if()

else if()

else

end

endmodule

**例：仿真模板**

**`timescale 1ns / 1ps**

**module XXX\_sim();**

**reg XX,XX;**

**initial begin**

**XX = 1;**

**#XX;**

**@(posedge XX) XX =0 ;**

**end**

**always begin**

**XX =0;**

**#XX;**

**XX =1;**

**#XX;**

**end**

**XXX XXX\_u(**

**. i\_X (XX),**

**. i\_X (XX),**

**. o\_X ()**

**);**

**endmodule**

## 1.5 实验结果

# 实验二 测量占空比

## 2.1实验目的

1. 了解测量占空比的原理；
2. 依据工作原理，使用Verilog语言求出给定波形的占空比；
3. 根据TB文件模板，学会编写仿真文件并仿真计数器实验。

## 2.2 实验内容

本实验的任务是使用边缘检测与计数求给定波形的占空比，编写Verilog代码并使用modelsim仿真。

## 2.3 实验步骤

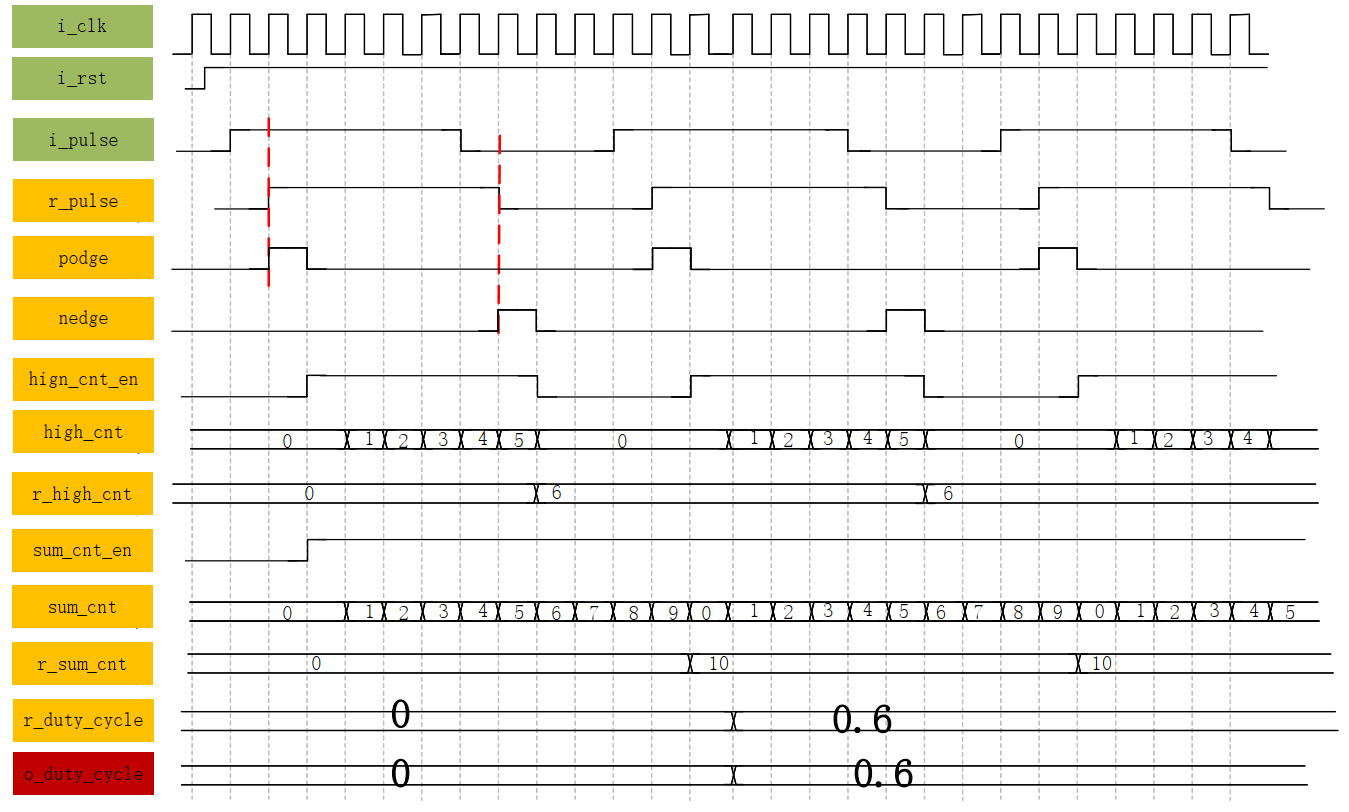
### 2.3.1 实验框架

图片包含 图示

描述已自动生成

* **i\_clk：**在仿真代码中通过always语句与#延迟符号编写出50MHz时钟，通过例化连接到工程系统时钟。
* **i\_pulse:** 在仿真代码中通过激励的方式得到
* **freq\_couunter:** 频率计

### 2.3.1 时序图



### 2.3.2 主要代码

reg，assign，always@(posedge i\_clk,posedge i\_rst)，if else if else，initial

## 2.4 实验程序

**例：后续需要将此处示例代码删除，仅需添加主要代码即可**

module XXX(

input i\_XXX,

output o\_XXX

);

reg ro\_XXX;

assign o\_XXX = ro\_XXX

always @(State)

begin

if()

else if()

else

end

endmodule

**例：仿真模板**

**`timescale 1ns / 1ps**

**module XXX\_sim();**

**reg XX,XX;**

**initial begin**

**XX = 1;**

**#XX;**

**@(posedge XX) XX =0 ;**

**end**

**always begin**

**XX =0;**

**#XX;**

**XX =1;**

**#XX;**

**end**

**XXX XXX\_u(**

**. i\_X (XX),**

**. i\_X (XX),**

**. o\_X ()**

**);**

**endmodule**

## 2.5 实验结果

# 实验三 数码管静态显示

## 3.1 实验目的

1. 了解数码管工作原理；
2. 了解数码管静态显示工作原理；
3. 依据数码管工作原理使用Verilog语言编程实现七位八段数码管静态显示。

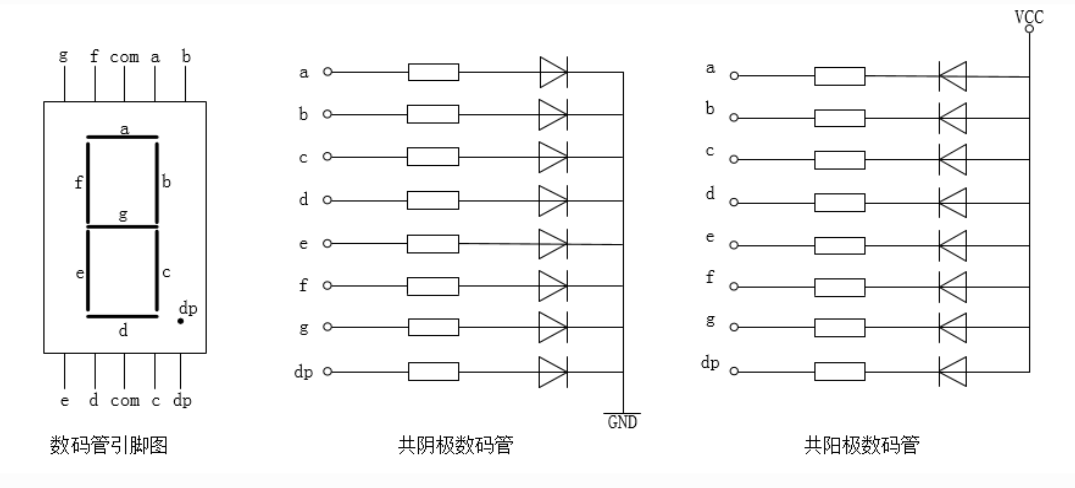
## 3.2 实验内容

本实验的任务是使用FPGA开发板上的8位数码管以静态显示的方式从0开始，进行十进制计数，每1s计数值增加一，当计数值从0增加到99999999后重新从0开始计数。

## 3.3 实验原理

### 3.3.1数码管简介

八段数码管的结构图如图所示。



由图可以看出，八段数码管是一个八字型数码管，分为八段：a、b、c、d、e、f、g、dp，其中dp为小数点，每一段即为一个发光二极管，这样的八段我们称之为段选信号。数码管常用的有10根管脚，每一段有一根管脚，另外两根管脚为一个数码管的公共端，两根互相连接。

数码管分为共阳极数码管和共阴极数码管。共阳极数码管就是把发光二极管的正极连接在一起作为一个引脚，负极分开。相反的，共阴极数码管就是把发光二极管的阴极连接在一起作为一个引脚，正极分开。这两者的区别在于，公共端是连接到地还是高电平，对于共阳极数码管需要给对应段低电平才会使其点亮，而对于共阴极数码管则需要 给其高电平才会点亮。本次实验使用的是共阳极数码管，也就是说给对应段低电平才会被点亮。给不同的段点亮可显示0~f的值。

### 3.3.2静态显示

段式数码管工作方式有两种：静态显示和动态显示。静态显示的特点是每个数码管的段选必须接一个8位数据线来显示字形，显示字形可一直保持，直到送入新字形码为止。

图示, 示意图

描述已自动生成

由上图可以看到每一个数码管都有一个位选信号，而这个位选信号就控制着数码管的亮灭。这样我们就可以通过位选信号去控制数码管亮，而在同一时刻，位选选通的数码管上显示的字形是一样的，因为我们将6个数码管相对应的段选连在了一起，数码管的显示自然就相同了，数码管的这种显示方式即为静态显示。

### 3.3.3 74HC573简介

由于直接控制数码管占用太多I/O口资源，所以使用74HC573（锁存器芯片）来控制数码管。

74HC573是一款典型的锁存器芯片，它的内部逻辑图如下图所示。

图示, 示意图

描述已自动生成

上图中，D0~D7为数据输入，Q0~Q7为数据输出。Output Enable为输出使能端， Latch Enable为锁存使能端。74HC573的真值表如下表所示。

表格

描述已自动生成

由真值表中可以看出，Output Enable为高电平时，无论Latch Enable与D端为何种电平状态，其输出都为高阻态，此时芯片处于不可控状态。所以在设计电路是我们直接将Output Enable接地，也就是74HC573一直处于使能状态。当Output Enable为低电平，且Latch Enable为高电平时，D与Q同时为H或L。当Latch Enable为低电平时，无论D为何种电平，Q都保持上一次的数据状态。锁存器就是把当前的状态锁存起来，使输入的数据在接口电路的输出端保持一段时间锁存后状态不再发生变化，直到解除锁存。锁存引脚高电平表示直通状态，为低电平表示锁存状态，可以通过控制锁存引脚，控制74HC573后端输出数据。

## 3.4 硬件资源

本实验各输入输出端口和开发板对应功能的硬件管脚如下表所示。

|  |  |
| --- | --- |
| 设计端口 | 芯片引脚（EP4CE6E22C8） |
| SEG\_A | 119 |
| SEG\_B | 121 |
| SEG\_C | 124 |
| SEG\_D | 115 |
| SEG\_E | 114 |
| SEG\_F | 120 |
| SEG\_G | 125 |
| SEL\_0 | 136 |
| SEL\_1 | 135 |
| SEL\_2 | 133 |
| SEL\_3 | 132 |
| SEL\_4 | 129 |
| SEL\_5 | 128 |
| SEL\_6 | 127 |
| SEL\_7 | 126 |

## 3.5实验程序

**例：后续需要将此处示例代码删除，仅需添加主要代码即可**

module XXX(

input i\_XXX,

output o\_XXX

);

reg ro\_XXX;

assign o\_XXX = ro\_XXX

always @(State)

begin

if()

else if()

else

end

endmodule

**例：仿真模板**

**`timescale 1ns / 1ps**

**module XXX\_sim();**

**reg XX,XX;**

**initial begin**

**XX = 1;**

**#XX;**

**@(posedge XX) XX =0 ;**

**end**

**always begin**

**XX =0;**

**#XX;**

**XX =1;**

**#XX;**

**end**

**XXX XXX\_u(**

**. i\_X (XX),**

**. i\_X (XX),**

**. o\_X ()**

**);**

**endmodule**

## 3.6实验结果

**实验四 动态显示**

## 4.1 实验目的

1. 了解数码管动态显示原理；
2. 依据动态显示原理使用Verilog语言编程实现八位八段数码管动态显示。

## 4.2 实验内容

让八位位数码管显示从十进制数0开始计数，每1s加1，一直到加到十进制数99999999。到达99999999之后回到0开始重新计数。

## 4.3 实验原理

### 4.3.1 数码管动态显示简介

在上一章节我们知道静态显示是让六个数码管的8位段选信号连在8根线上且六个数码管的位选信号同时选中点亮。但是如果我们每次只选中一个数码管点亮呢？这样我们段选信号点亮的就只是我们选中数码管的值了，那是不是就可以给每个数码管显示不一样的值了？但是这样我们又会发现一个新的问题：每次只点亮一个 数码管，那么同一时间六个数码管就只能看到一个数码管在亮，那不是同时显示不了六个不同的字符了吗？针对这个问题先为大家介绍两种现象：

首先是人眼视觉暂留：人眼在观察景物时，光信号传入大脑神经，需经过一段短暂的时间，光的作用结束后，视觉影像并不立即消失，这种残留的视觉称“后像”，视觉的这一现象则被称为“视觉暂留”。其次是数码管的余晖效应：当停止向发光二极管供电时发光二极管亮度仍能维持一段时间。根据这两种现象我们可以想到，如果让数码管轮流显示，而且轮流显示速度很快，这样会不会看起来六个数码管都在显示呢？事实证明是可以的，这种方式称为动态扫描。为帮助大家理解，打个比方：若一个数码管在1s内点亮两次，那么我们可以很明显的看到其亮了两次，若1s内点亮10次呢？我们可能只能看到其在快速的闪烁，若点 亮100次1000次呢？总有一个速度我们人眼是分辨不出来在闪烁的。所以说一个数码管让我们人眼感觉一直在亮并不用一直给其点亮，只要我们让其亮的间隔足够短就行。这样我们就可以用不在点亮的时间去点亮其他数码管，让其他数码管也达到这样的效果，这样就可以让我们人眼感觉所有数码管都在同时点亮了。那么多长的动态扫 描间隔可以让人眼感觉所有数码管都在亮而不会有闪烁感呢？实验证明，当我们的扫描间隔为1ms时不会有闪烁感。

最后为大家总结我们的动态驱动数码管的方式：使用1ms的刷新时间让六个数码管轮流显示：第1ms点亮第一个数码管，第2ms点亮第二个数码管，以此类推依次点亮六个数码管，6ms一个轮回，也就是说每个数码管每6ms点亮1ms，这样就能让人眼感觉到数码管一直在亮了。点亮相应数码管的时候给其显示相应的值，这样就 可以使六个数码管显示不同的值了，这就是驱动数码管动态显示的方法。

## 4.4实验程序

**例：后续需要将此处示例代码删除，仅需添加主要代码即可**

module XXX(

input i\_XXX,

output o\_XXX

);

reg ro\_XXX;

assign o\_XXX = ro\_XXX

always @(State)

begin

if()

else if()

else

end

endmodule

**例：仿真模板**

**`timescale 1ns / 1ps**

**module XXX\_sim();**

**reg XX,XX;**

**initial begin**

**XX = 1;**

**#XX;**

**@(posedge XX) XX =0 ;**

**end**

**always begin**

**XX =0;**

**#XX;**

**XX =1;**

**#XX;**

**end**

**XXX XXX\_u(**

**. i\_X (XX),**

**. i\_X (XX),**

**. o\_X ()**

**);**

**endmodule**

## 4.5实验结果

# 实验心得

|  |  |
| --- | --- |
| **学生**  **实验**  **心得** | 学生（签名）：  年 月 日 |
| **指导教师评语** | 成绩评定：  指导教师（签名）：  年 月 日 |