|  |
| --- |
| **北 京 邮 电 大 学**  **实 验 报 告**  **课程名称\_数字逻辑与数字系统\_\_\_\_\_**  **实验名称\_\_实验五\_\_十进制计数器七段显示**  **实验六\_（1）\_60进制计数器\_\_\_**  **实验六\_（2）\_读写寄存器\_\_\_\_\_\_\_**  **\_\_计算机\_\_\_\_学院\_\_307\_\_\_\_班 姓名\_魏生辉\_\_**  **教师\_\_赵学达\_\_\_\_\_\_ 成绩\_\_\_\_\_\_**  **\_2024\_\_\_年\_\_12\_\_月\_9\_\_\_日** |

|  |
| --- |
| **一．硬件测试环境描述：**  EPM 7128 1片；  TEC-8数字电路实验系统 1台；  逻辑笔 1个；  示波器 1台**；**  ISP1032；  **二．实验详述：**  **实验五 十进制计数器七段显示：**  1.实验目的：  用Verilog语言设计一个8421码十进制计数器及七段数码管显示系统。  学习使用Quartus II软件。  学习EPM7128的编程和下载。  2**.**实验内容：  在TEC-8实验台上，使用单脉冲作计数器时钟，复位 信号接电平开关。输出接最左边的数码管进行验证，计 数器能够从0-F，可以增加一个STOP按键。   1. 代码模块设计与流程必要说明：   **①模块设计概述**：  **1**.**十进制计数器模块**负责在时钟的驱动下对输入信号进行计数，并根据计数值更新输出。计数范围为0到15，当计数达到15后会自动清零。  **2.7段显示译码器模块：** 根据计数值生成对应的7段显示器控制信号（高电平有效），将计数值显示为十进制数字或字母。  **②模块功能分析**  **1.十进制计数器**  输入信号：clk：时钟信号，用于触发计数器递增。  clr：清零信号，低电平有效，清除计数值，重置为0。  stop：停止信号，低电平有效，暂停计数器的递增。  输出信号：count：4位二进制计数值，表示当前计数值，范围为0到15。  seg：7段显示器的控制信号，用于显示当前count值。  **2.7段显示译码器**  译码器根据count值生成7段显示器控制信号，使其显示对应的十进制数字或字母（0-9、A-F）。  同时要保证seg信号中的**高电平**表示对应的段点亮。  **③核心代码展示**    **④引脚说明：**     1. 实验现象与分析：   **1. 计数器的正常工作**  在系统上电并初始化时，clr信号被置为高电平，计数器处于初始状态，count值为0。  当clk信号开始工作时，计数器开始按时钟上升沿递增。每次时钟脉冲到来时，count值会递增1，直到达到15（4'd15）。此时，count会重新回到0，开始新的循环。  实验现象：计数器能够按时钟脉冲依次**显示0到15的数字**，7段显示器的显示效果符合预期。  **2. 清零功能的验证**  当clr信号被拉低时，计数器立即清零，count值归0，无论计数器当前的状态是什么。  实验现象：无论count当前值是多少，clr信号低电平有效后，计数器会迅速将count值清零，并通过7段显示器显示**数字0**。  **3. 暂停功能的验证**  当stop信号被拉低时，计数器停止递增，保持当前count值不变。  实验现象：在stop信号激活（低电平）时，计数器**停止增加**，无论时钟如何变化，count值保持在当前显示的数字，不会变化。  **4. 7段显示器的显示**  实验现象：7段显示器按预期显示计数值。例如，当count值为4'd0时，7段显示器显示数字0；当count为4'd10时，显示字母A；当count为4'd15时，显示字母F。  **5. 计数器的复位与回绕**  当count达到最大值15时，计数器会回绕并从0重新开始计数。  实验现象：count值从4'd0开始，依次增加到**4'd15，然后再次从4'd0**开始循环。7段显示器会相应显示出数字或字符的循环。 |

|  |
| --- |
| **实验六（1） 六十进制计数器/秒表：**   1. 实验目的：   用Verilog语言设计一个秒表，计数从00到59 ，以两个数码管显示输出。  2.实验内容：  设计并实现一个**BCD计数器**，该计数器能够计数从0到59，并将结果以二进制编码十进制（BCD）的方式显示。计数器应具有复位功能和暂停功能。  3.代码模块设计与流程必要说明：  **①模块设计概述**：  本设计基于Verilog硬件描述语言实现了一个同步BCD计数器。该计数器的设计包括两个主要部分：  十位计数器（tens）：从0计数到5。  个位计数器（ones）：从0计数到9。  计数器的控制信号包括：  clk：时钟信号，驱动计数器按时钟周期递增。  rst：复位信号，低电平有效，用于将计数器重置为0。  pause：暂停信号，低电平有效，用于暂停计数器的计数。  **②模块功能分析**  **输入信号**  clk：时钟信号，计数器的时序由时钟脉冲控制。  rst：复位信号，低电平有效，用于将计数器清零。  pause：暂停信号，低电平有效，用于暂停计数器。  **输出信号**  tens：十位BCD，4位二进制数表示十位上的数字（0到5）。  ones：个位BCD，4位二进制数表示个位上的数字（0到9）。  **功能描述**  个位计数器：从0到9，达到9后自动清零并使十位计数器加1。  十位计数器：从0到5，达到5后自动回绕至0。  复位：当rst信号被激活时，tens和ones都被清零，计数器返回初始状态。  暂停：当pause信号被激活时，计数器停止递增，保持当前计数状态。  **③核心代码展示**    **④引脚说明：** |

|  |
| --- |
| 4.实验现象与分析：  **1.计数器正常工作**  **实验现象：**  当时钟信号clk开始工作时，计数器按照预期的十进制数值逐步递增。  ones（个位）计数器从0到9递增，达到9时会清零，并且ones清零时，tens（十位）计数器加1。  tens（十位）计数器从0递增到5，当ones归零后，tens会增加1，直到tens达到5后再清零，重新开始。  **分析：**  该现象符合设计预期：ones计数器在0到9之间循环，tens计数器在0到5之间循环。当ones达到9时，ones清零并触发tens加1，这使得计数器在00到59之间循环递增。  **2.复位功能验证**  **实验现象：**  当rst信号（复位信号）为低电平时，计数器的ones和tens值立即清零（即显示为00），无论计数器当前的状态如何。  此时计数器停止计数，并返回到初始状态。  **分析：**  该现象符合预期：复位信号rst被拉低时，计数器无条件清零。  **3. 暂停功能验证**  **实验现象：**  当pause信号为低电平时，计数器停止计数，保持当前的ones和tens值不变。  计数器的状态不发生变化，直到pause信号被拉高，计数器恢复正常工作。  **分析：**  pause功能的实现使得计数器能够在不中断其他操作的情况下暂停计数。暂停信号通过控制计数器的状态机，当pause为低电平时，计数器“冻结”在当前值，不再增加，符合实验设计目标。  **4. 计数器回绕与最大值处理**  **实验现象：**  当ones计数器达到9时，ones会自动清零，tens计数器加1。  当tens计数器达到5时，tens清零，并且ones重新开始从0计数。 |

|  |
| --- |
| **实验六（2） 提高项目：读写寄存器：**   1. 实验目的：   设计并实现一个**寄存器文件**模块。该模块具有四个8位寄存器，通过控制信号来实现读写操作。实验的目标是验证寄存器文件的基本功能，包括寄存器写入、读取以及高阻态输出。  2.实验内容：  用硬件描述语言设计四个寄存器，8bit数据输入， 8bit数据三态输出，在实验箱上显示输出。  3.代码模块设计与流程必要说明：  **①模块设计概述**：  本设计通过Verilog语言实现了一个四个8位寄存器的寄存器文件模块。寄存器文件能够根据输入的地址和控制信号进行相应的读写操作。  **输入信号：**  clk：时钟信号，用于驱动写操作。  crw：控制信号，用于选择读操作或写操作。当crw为高电平时，表示进行写操作；为低电平时，表示进行读操作。  a[1:0]：2位地址信号，用于选择要读写的寄存器（共有四个寄存器：r0、r1、r2、r3）。  r[7:0]：8位输入数据，用于写入寄存器。  **输出信号：**  d[7:0]：8位输出数据，通过寄存器文件的读取操作获得。读取时，输出对应寄存器的值。  **②模块功能分析**  **写操作（crw = 1）：**  根据地址a[1:0]选择对应的寄存器（r0、r1、r2、r3），将输入数据r[7:0]写入选中的寄存器。  **读操作（crw = 0）：**  根据地址a[1:0]选择要读取的寄存器，并将其值输出到d[7:0]。如果ct为高电平，则输出d为高阻态（8'bz），表示禁用输出。  **寄存器文件：**  四个8位寄存器r0、r1、r2、r3存储数据，能够根据控制信号进行读写操作。 |

|  |
| --- |
| **③核心代码展示** |

|  |
| --- |
| **④引脚说明：**     1. 实验现象与分析：   **1.写操作（crw = 1）**  **实验现象：**  当写使能信号crw为高电平时，输入数据r[7:0]会被写入到指定的寄存器中。通过地址信号a[1:0]，可以选择不同的寄存器进行写操作：a = 2'b00：数据写入寄存器r0。  a = 2'b01：数据写入寄存器r1。  a = 2'b10：数据写入寄存器r2。  a = 2'b11：数据写入寄存器r3。  **分析：**  该现象与设计预期一致。每个寄存器的数据在时钟的正沿时被更新，且不同地址对应不同的寄存器。  **2. 读操作（crw = 0）**  **实验现象：**  当crw为低电平时，表示进入读操作模式。此时，地址a[1:0]会选择要读取的寄存器，输出信号d[7:0]会显示对应寄存器的数据：  a = 2'b00：输出寄存器r0的数据。  a = 2'b01：输出寄存器r1的数据。  a = 2'b10：输出寄存器r2的数据。  a = 2'b11：输出寄存器r3的数据。  **分析：**  在读操作时，模块能够根据地址信号选择对应寄存器并输出数据。ct信号的作用是控制数据输出的有效性，当ct = 1时，d处于高阻态，表示禁止输出数据。这是寄存器文件设计中常见的功能，用于确保只有一个模块能够输出数据，避免总线冲突。  **3. 高阻态输出（ct = 1）**  **实验现象**：  当控制信号ct为高电平时，输出d会变为高阻态（8'bz），即禁止数据输出。  在实验中，当ct = 1时，不管crw的值如何，d都不会输出有效数据，而是保持高阻态。  **分析：**  高阻态功能的测试结果与预期一致，ct = 1时，d正确处于高阻态，其他信号不受影响。  **4. 寄存器数据保持性**  **实验现象：**  在进行读操作时，寄存器中的数据始终保持有效，除非进行写操作。  写操作后，寄存器的值在没有其他操作的情况下不会改变，这保证了数据在寄存器中保持直到下一次写操作。   1. **综合实验总结：**   通过本次数字逻辑与数字系统课程的实验，我深入学习了如何使用Verilog硬件描述语言设计和实现数字电路，尤其是在计数器、寄存器文件以及7段数码管显示等模块的设计与验证方面积累了实践经验。  其中，最让我感到汗流浃背的其实是第一个实验——7段数码管显示，虽然在现在的视角看起来这只是小菜一碟，但是对于当时没有接触过软硬件结合实验的我来说，属实让人破防。  最开始，我踌躇满志写完代码，准备大展拳脚，不过一会就倒在了引脚连接上，这也是我每个实验都会附带上引脚说明的原因，这个过程对于我这种心浮气躁的人来说真是场折磨，不过后来吸取教训，后来者居上。  最后的提高实验也是一度让我崩溃，我明明是最快完成的但是却没有显示我想要的效果，后来仔细检查实验箱，发现很多地方被上一位“好心人”动了手脚。之后就顺利完成了。  说实话，这学期的数电实验真的很有魅力，有魅力的老师，有魅力的设计，在学期初期，听闻老师说要学一门新的编程语言，我当时可谓是十分害怕，加上之前就一直对实验很胆怯，所以一直很担心这门课的实验，但是到现在，只叹一句——轻舟已过万重山！  感谢赵老师的指导！ |