P7设计文档及思考题

顶层设计

包含四个部件

//调用流水线CPU

CPU CPU(

//以下为基础的时钟和复位信号

.clk(clk),

.reset(reset),

//以下为P7新加的指令

.HWInt(HWInt), //6个硬件中断请求

.PrRD(PrRD), //从Bridge模块读入的数据

.PrWe(PrWe), //cpu写使能

//.PrBE(PrBE), //4位字节使能

.PrWD(PrWD), //输出至Bridge模块的数据

.PrAddr(PrAddr), //32位地址总线

.addr(addr)

);

//调用Bridge进行连接CPU与IO进行数据交换

Bridge Bridge(

//从CPU到IO的32位地址

.PrAddr(PrAddr),

.DEVAddr(DEVAddr),

//从CPU到IO的写使能信号

.PrWe(PrWe),

.DEV0We(DEV0We),

.DEV1We(DEV1We),

//从CPU到IO的写入数据

.PrWD(PrWD),

.DEVWD(DEVWD),

//从IO到CPU的返回值PrRD

.DEV0RD(DEV0RD),

.DEV1RD(DEV1RD),

.PrRD(PrRD)

);

//调用Timer0进行计时

Timer Timer0(

.clk(clk),

.reset(reset),

.Addr(DEVAddr[31:2]),

.WE(DEV0We),

.Din(DEVWD),

.Dout(DEV0RD),

.IRQ(IRQ0)

);

//调用Timer1进行计时

Timer Timer1(

.clk(clk),

.reset(reset),

.Addr(DEVAddr[31:2]),

.WE(DEV1We),

.Din(DEVWD),

.Dout(DEV1RD),

.IRQ(IRQ1)

);

思考题

1.我们计组课程一本参考书目标题中有“硬件/软件接口”接口字样，那么到底什么是“硬件/软件接口”？

软硬件接口就是传输CPU与外设的数据，传输硬件数据，并将硬件映射到内存地址的Bridge、片选控制器和总线等。

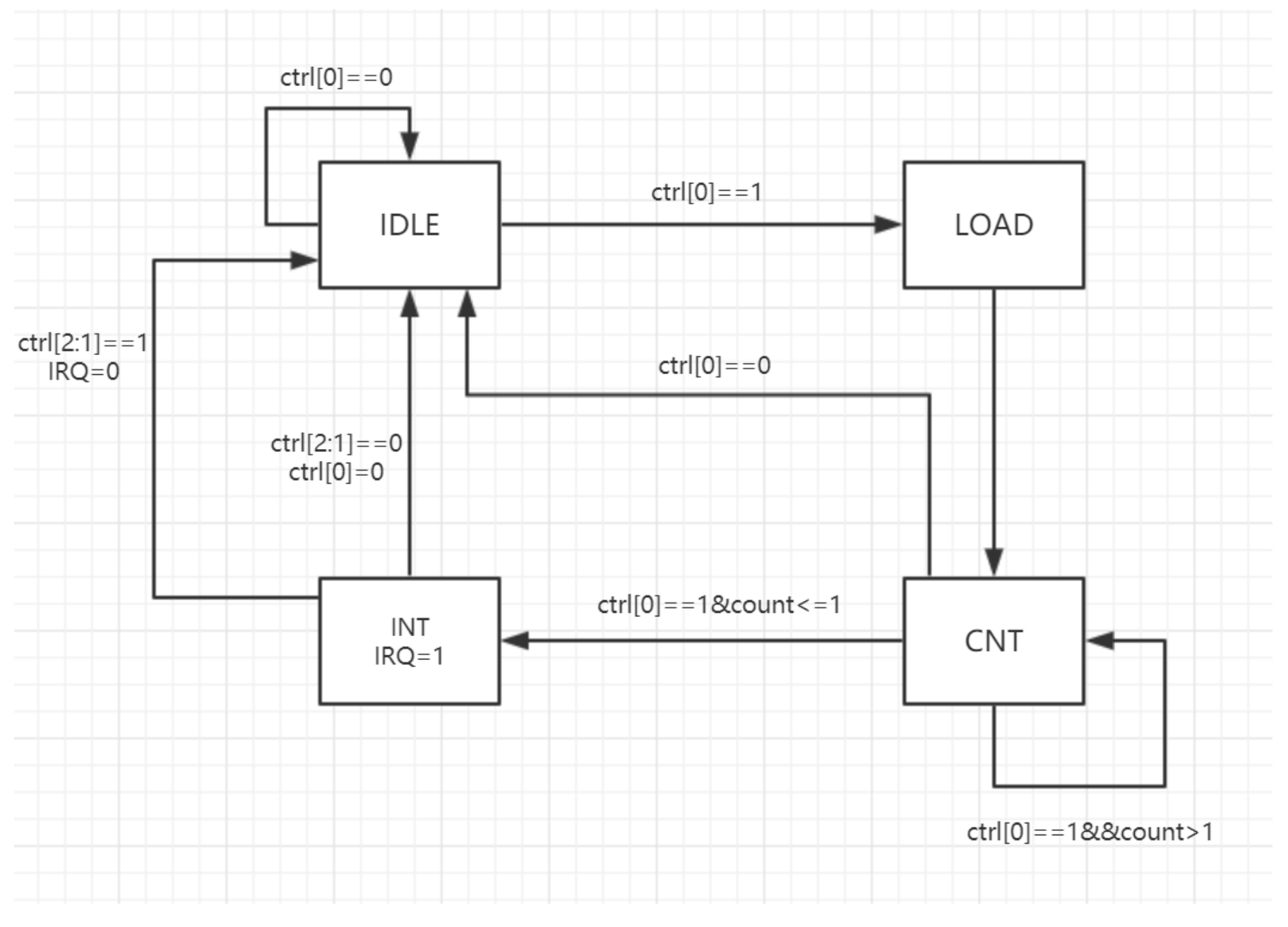
2 .在我们设计的流水线中，DM 处于 CPU 内部，请你考虑现代计算机中它的位置应该在何处。

在CPU外部，并通过主板上的北桥与CPU连接。

3.BE 部件对所有的外设都是必要的吗？

不一定，对于按字保存的外设就没有必要。

4.请阅读官方提供的定时器源代码，阐述两种中断模式的异同，并分别针对每一种模式绘制状态转移图



计时器使用说明：

计时器内有三个寄存器，CTRL,PRESENT,COUNT

CTRL 是用来控制计时器的模式和使用，CTRL[2:1]选择模式 0 或 1，CTRL[0]是计时器使能，CTRL[3]是中断屏蔽位，控制寄存器决定该计数器启停控制，初值寄存器为 32 位计数器提供初始值，使用 store 类指令修改寄存器的优先级高于 timer 自身修改的优先级。

模式 0：

当计数器倒计时为 0 后，计数器停止计数，此时控制寄存器中的使能CTRL[0]变为 0，当其再次被设置为 1 后，初值寄存器被再次加载到计数器，计数器重新启动倒计数。模式 0 用来产生定时中断，中断屏蔽位可以屏蔽中断。

模式 1：

当计数器计数为 0 后，初值寄存器被自动加载到计数器，计数器继续倒计时，模式 1 用来产生周期性脉冲，模式 1 下计数器每次计数循环只产生一周期的中断信号。

5.开发一个主程序以及定时器的handler

主程序及Handler：

.text 0x00003000 # Main.v

ori $t0, $0, 0x0000FC01 # SR: Allow All Interrupt

mtc0 $t0, $12

addu $s0, $0, 0x00007F00 # Timer 1

addu $s1, $0, 0x00007F10 # Timer 2

ori $t0, $0, 7

sw $t0, 4($s0) # init = 7

sw $t0, 4($s1) # init = 7

ori $t0, $0, 9

sw $t0, 0($s0) # start mode 0

sw $t0, 0($s1) # start mode 0

……

……

(while(True): pass)

.ktext 0x00004180 # Handler.v

addu $s2, $0, 0x00007F00 # Timer 1

addu $s3, $0, 0x00007F10 # Timer 2

ori $t0, $0, 9

sw $0, 0($s2) # CTRL: 0,0

sw $0, 0($s3)

sw $t0, 0($s2) # CTRL: 1,1

sw $t0, 0($s3)

eret

6.键盘鼠标的输入信号：低速设备通过南桥和北桥与CPU沟通。用中断信号让CPU响应。当键盘/鼠标触发一个动作时，设备发出一个中断信号。CPU接收中断信号后，根据不同的中断信号执行不同的中断处理程序，从而特异性的响应外设动作：将设备存入的寄存器 / 设备内存中的数据保存。