

第 5 章 验证性实验

5.1 THCO MIPS 指令系统实验

THCO MIPS 指令系统是 THINPAD 教学计算机建议的指令系统,熟悉指令系统中各指令的功能及格式后,可为在后续实验中实现该指令系统打好基础。

5.1.1 实验目的

- (1) 熟悉教学计算机 THINPAD 指令系统,包括指令功能、指令格式和寻址方式等。
- (2) 了解教学计算机模拟器的功能,并能熟练使用。
- (3) 掌握 THINPAD 教学计算机汇编语言程序的基本编程方法,能编写简单程序。

5.1.2 实验环境

- (1) 硬件环境: 安装有 Windows 7 或以上操作系统的微机。
- (2) 软件环境: THINPAD 教学计算机模拟器、汇编器。

5.1.3 实验内容

- (1) 编写汇编语言程序,求前 10 个 Fibonacci 数,将结果保存到起始地址为 0x8500 的连续 10 个字中。
- (2) 编写汇编语言程序,将 ASCII 字符表在终端上输出。
- (3) 自行编写一个能完成一定功能的汇编语言程序并完成调试。

5.1.4 实验原理

本实验通过模拟器的使用,帮助学生学习和掌握 THCO MIPS 指令系统的指令功能和指令格式,并体会计算机机器语言程序的执行过程。实验提供的模拟器可直接模拟汇编语言程序的执行,也可模拟 THINPAD 教学计算机硬件运行监控程序的过程,在监控程序的管理下运行用户程序。

5.1.5 主要实验步骤

本实验所有的 3 个汇编语言程序均在模拟器的两种工作模式下运行。

1. 指令系统功能模拟

指令系统功能模拟模式下,启动模拟器以后,使用模拟器提供的命令,输入汇编语句并逐句完成汇编,然后再执行用户程序。具体步骤如下:

- (1) 启动模拟器程序。
- (2) 用 ea 命令,逐句输入编写好的程序并汇编,注意程序的起始地址。

(3) 用 v 命令观察输入的程序的二进制代码,分析指令格式,指明操作码、操作数和寻址方式。

(4) 用 c 命令运行程序,并用 v 命令查看运行结果并记录。请注意程序的首地址,必要时可先用 goto 命令保证程序的首地址的正确性。

(5) 使用模拟器中其他命令,体会各命令的功能。

(6) 按以上步骤完成 3 个程序的调试,并记录过程和结果。

2. 硬件系统功能模拟

硬件系统功能模拟是指用模拟器来充当 THINPAD 教学计算机的硬件,再用终端程序 Term 来做模拟器的输入输出设备,连接模拟器运行。具体步骤如下:

(1) 启动模拟器,并确保监控程序 kernel. bin 在当前目录下。

(2) 在模拟器下运行 server 指令,启动模拟 THINPAD 硬件执行。

(3) 启动 Term 程序,并运行 SIM 命令,与模拟器模拟的硬件连接,这样,Term 程序将作为模拟器的终端,与连接真正的硬件运行时完全一样。

(4) 在监控程序的命令提示符下,输入 A 命令,然后输入编写好的汇编程序,输入完成后,退出编辑和汇编状态。

(5) 用 G 命令运行用户程序,注意程序的首地址。

(6) 用 U 命令逐条指令查看指令格式,并指出其寻址方式。

(7) 按以上步骤完成 3 个程序的调试,并记录过程和结果。

5.1.6 思考题

(1) 根据程序中每条指令的机器代码,给出其指令格式及寻址方式。

(2) 指出程序中寻址方式为立即数寻址的指令中,立即数的机器表示和实际数值各是多少?

(3) 如果要求程序计算前 50 个 Fibonacci 数,应如何修改程序?

(4) 读懂监控程序和 Term 程序的代码,并画出两个程序的框图。

5.2 监控程序扩展实验

监控程序直接运行于 THINPAD 硬件平台之上,相当于教学计算机的简单的操作系统,也被后续实验作为验证硬件设计是否正确的一个测试软件。了解监控程序功能及实现过程,掌握其使用方法,一方面可继续熟悉指令系统,另一方面可为后续实验的调试打好基础。

5.2.1 实验目的

(1) 熟悉教学计算机监控程序,了解监控程序功能的具体实现。

(2) 熟悉教学计算机终端程序 Term,了解监控程序与 Term 的相互关系。

(3) 将监控程序支持的指令集扩充,为后续实验打好基础。

5.2.2 实验环境

(1) 硬件环境: 安装有 Windows 7 或以上操作系统的微机。

(2) 软件环境: THINPAD 教学计算机模拟器、终端程序 Term、监控程序 Kernel。

5.2.3 实验内容

- (1) 阅读监控程序和终端程序 Term 的源代码,并给出程序框图。
- (2) 标准的 Term 和监控程序只能汇编基本指令集中的指令,请将其扩充为可汇编和反汇编包含你们要实现的指令集的全部指令。

5.2.4 实验原理

本实验通过对相关程序的源代码的分析,要求学生了解后续实验中需要运行的软件的结构,进一步熟悉指令系统和实验环境。

教学计算机中的监控程序功能相对简单,仅完成了最基本的字符输入和输出功能,而对这些字符进行分析和正确显示是通过高级语言编写的 Term 来完成的。实验中请注重了解两个程序的配合关系。

5.2.5 主要实验步骤

本实验主要是对程序的阅读和分析,具体操作中仅仅是对完成修改后的程序进行验证。根据本组所要完成的指令系统,用 A 命令汇编基本指令 and 全部扩展指令,并用 U 命令进行反汇编,用 D 命令检查结果是否正确。

5.2.6 思考题

- (1) 读懂监控程序和 Term 程序的代码,并画出两个程序的框图。
- (2) 监控程序是否也能完成汇编和反汇编功能? 如何让监控程序连接标准的终端程序实现目前的功能? 请简要分析实现的难点。

5.3 算术逻辑部件 ALU 实验

算术逻辑部件 ALU 是计算机的核心功能部件,完成数据的算术和逻辑运算,功能和结构相对简单,实现也不太复杂,但却是所有计算机的基础。

5.3.1 实验目的

- (1) 熟悉硬件描述语言及开发环境,了解硬件系统开发的基本过程。
- (2) 掌握 ALU 基本设计方法和简单运算器的数据传送通路。
- (3) 验证 ALU 的功能。

5.3.2 实验环境

- (1) 硬件环境: 安装有 Windows 7 或以上操作系统的 PC, THINPAD 教学计算机。
- (2) 软件环境: FPGA 开发工具软件 Xilinx ISE 12.3 或以上。

5.3.3 实验内容

- (1) 根据实验原理中的要求,用 VHDL 语言实现一个简单的 ALU。
- (2) 在教学计算机 THINPAD 上验证实现的 ALU 的功能。

5.3.4 实验原理

本实验通过设计一个简单的 ALU 帮助学生更好地理解数据通路和 ALU 的工作原理，并通过该实验使同学们熟悉 VHDL 硬件描述语言，为接下来的实验打好基础。

算术逻辑部件(ALU)的主要功能是对二进制数据进行定点算术运算、逻辑运算和各种移位操作等。算术运算包括定点加减乘除运算；逻辑运算主要有逻辑与、逻辑或、逻辑异或和逻辑非等操作。ALU 通常有两个数据输入端 A 和 B 输入操作数，一个数据输出端 Y 以及标志位输出结果，通过输入操作码 op 来确定所要进行的操作。本实验通过实现一个状态机，根据状态机状态的变化来输入操作数及操作码，并最终实现不同的运算，将结果和标志位呈现出来。

本实验中的 ALU 要求实现基本的算术运算、逻辑运算、移位运算等，具体功能如表 5.1 所示。

表 5.1 ALU 功能表

操作码	功能	描 述	操作码	功能	描 述
ADD	$A+B$	加法	NOT	not A	取非
SUB	$A-B$	减法	SLL	A sll B	逻辑左移 B 位
AND	A and B	与	SRL	A srl B	逻辑右移 B 位
OR	A or B	或	SRA	A sra B	算数右移 B 位
XOR	A xor B	与或	ROL	A rol B	循环左移 B 位

要求：ALU 的数据输入 A、B 的长度为 16 位，操作码 op 为 4 位，算术运算时数据用补码表示。

5.3.5 主要实验步骤

本实验通过 VHDL 语言实现一个比较简单的 ALU 模块，如图 5.1 所示。

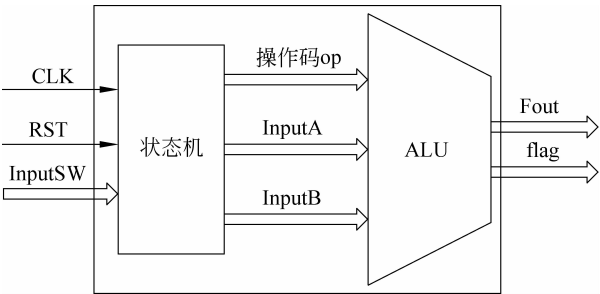


图 5.1 ALU 实验框图

利用 THINPAD 教学计算机完成 ALU 功能的验证，具体步骤如下：

(1) 用 VHDL 语言编写 ALU 功能代码，并用状态机对其进行控制，使其完成实验要求的操作。操作码和操作数的输入用微型开关 SW0~SW15，计算结果的输出用教学机上的 LED 灯来显示。

- (2) 将代码下载到教学机的 FPGA 芯片中,并调试完成。
- (3) 在 THINPAD 教学机上运行时,RST 和时钟均用手动开关或按钮,便于演示。操作码和操作数在开关 SW0~SW15 上输入;为便于观察和调试,每次 ALU 得到操作数,最好可以在 LED 上显示一下。最后的运算结果在 L0~L15 上显示,标志位可自行选择显示方法。
- (4) 状态机设计可以参考图 5.2。

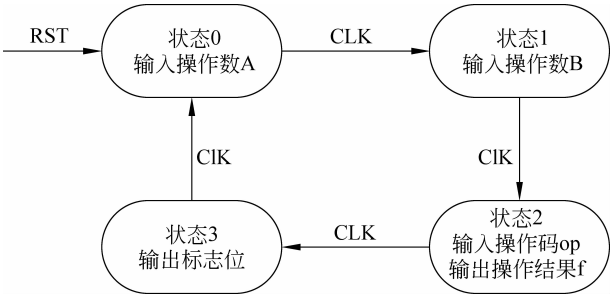


图 5.2 状态机参考设计

- (5) 记录实验结果。
- 注意：**教学机上 SW0~SW15 是 ON 端表示 0,相反表示 1;CLK、RST、K1~K4 均是按下为 0,放开为 1。本实验使用的管脚分配如表 5.2 所示。

表 5.2 管脚分配表

端口性质	端口	管脚	说 明	端口性质	端口	管脚	说 明
输入	CLK	U9	单步时钟 CLK	输出	L0	D10	发光二极管运算结果及标志位输出
	RST	U10	复位键 RST		L1	E10	
	SW0	T7	数据及操作码输入		L2	A11	
	SW1	R7			L3	B11	
	SW2	U8			L4	C11	
	SW3	V4			L5	D11	
	SW4	V3			L6	E11	
	SW5	V2			L7	F11	
	SW6	U1			L8	A12	
	SW7	R4			L9	E12	
	SW8	R1			L10	F12	
	SW9	N2			L11	A13	
	SW10	N1			L12	B13	
	SW11	M1			L13	D13	
	SW12	K7			L14	E13	
	SW13	K2			L15	A14	
	SW14	J7					
	SW15	J6					

5.3.6 实验数据

将实验过程中进行的操作与结果数据记录在表 5.3 中。

表 5.3 实验数据

输入数据			实际输出		与预期一致性
操作码	操作数 A	操作数 B	运算结果	标志位	

5.3.7 思考题

- (1) ALU 进行算术逻辑运算所使用的电路是组合逻辑电路还是时序逻辑电路？
- (2) 如果给定了 A 和 B 的初值,且每次运算完后结果都写入到 B 中,再进行下次运算。这样一个带暂存功能的 ALU 要增加一些什么电路来实现？

5.4 内存储器系统实验

存储器系统是计算机中存放程序 and 数据的场所,分为内存和外存。THINPAD 教学计算机设置了两片 RAM 作为内存,使用 Flash 存储器作为外存,本实验主要完成内存存储器的读写访问过程。

5.4.1 实验目的

- (1) 熟悉 THINPAD 教学计算机内存存储器的配置及与总线的连接方式。
- (2) 掌握教学机内存(RAM)的访问时序和方法。
- (3) 理解总线数据传输的基本原理。

5.4.2 实验环境

- (1) 硬件环境：安装有 Windows 7 或以上操作系统的微机、THINPAD 教学计算机。
- (2) 软件环境：FPGA 开发工具软件 Xilinx ISE 12.3 或以上。

5.4.3 实验内容

使用教学计算机上的 FPGA 芯片,设计一个状态机和内存读写逻辑,完成对存储器 RAM 的访问。具体要求如下：

- (1) 写 RAM1。将拨码开关上的数据写入到 RAM1 的相应存储单元中。首先,从拨码开关拨入写入地址单元的地址,按 CLK 键后,再在拨码开关上拨入要写入该单元的数据,再

按 CLK 键,数据应写入到 RAM1 的对应单元中。继续按 CLK 键,则地址和数据各加 1 后写入,共写 10 个数据。过程中,应在 LED 灯上分别显示地址和数据。

(2) 读 RAM1。RAM1 中写入 10 组数据后,按 CLK 键后开始读 RAM1 的内容。按 CLK 键,逐个将刚写入的 10 个数据从存储单元中读出,送到 LED 上显示。

(3) 写 RAM2。将前面 RAM1 中已保存的 10 个数据,各减 1 后写入到 RAM2 的同样的地址单元内,写入过程中要求在 LED 灯上输出地址和被写入的数据。

(4) 读 RAM2。要求同读 RAM1。

注意: 后两步为选做实验,在实验过程中可以使用七段数码管监测状态机当前状态,发光二极管只有 16 个,可以只显示地址的低 8 位和数据的低 8 位。

5.4.4 实验原理

对于存储器芯片的访问,首先要熟悉存储器芯片的访问时序,并了解 THINPAD 教学机上存储器芯片与 FPGA 芯片的具体连接方式,也就是说,如何在 THINPAD 教学机上具体实现对 RAM 芯片的访问时序。

THINPAD 教学计算机上使用的是两片 $256 \times 1024 \times 16b$ 的 SRAM,SRAM 的访问时序如图 5.3 所示。

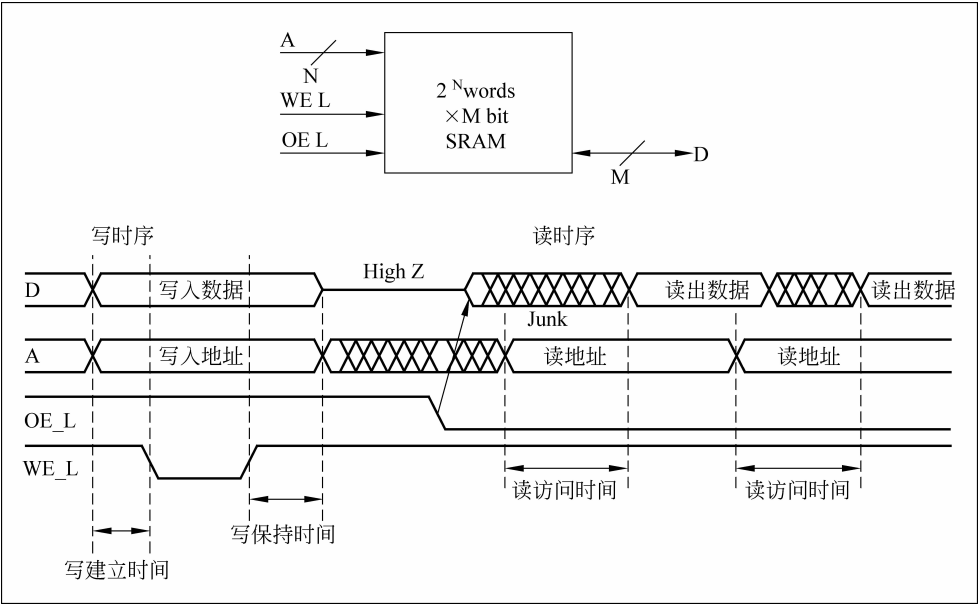


图 5.3 SRAM 的访问时序

从时序要求上看,读取 SRAM 时要提前准备好地址,并将数据线设置成高阻,然后就可以读出数据了;写 SRAM 时要提前准备好地址数据,然后将写信号拉低即可将数据写入相应地址。

要完成实验,就要给内存芯片提供地址、数据和控制信号。初始地址和数据均来自拨码开关,需要有寄存器接收并保存好,然后分别送到地址总线 and 数据总线。控制信号有 3 个,以 RAM1 为例,分别是 RAM1_EN、RAM1_OE 和 RAM1_RW,对应连接到 RAM 芯片的/

CE、/OE 和 /WE 管脚,需要根据对存储芯片的访问要求来正确设置。

需要指出的是,由于 RAM1 和串口共同连接在一条总线上,需要使串口不工作,这可以通过控制信号(/OE)的选择来实现。

实验中用到的主要芯片连接关系如图 5.4 所示。其中,内存控制器用 FPGA 实现,基本内存和扩展内存分别指 RAM1 和 RAM2 两片芯片。

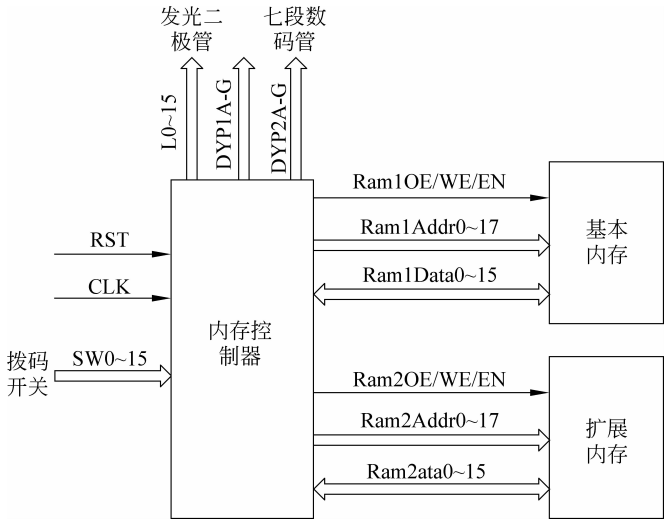


图 5.4 存储器访问实验器件连接图

5.4.5 主要实验步骤

本实验的完成需要用 FPGA 芯片来控制存储芯片的读写,根据实验要求,需要在 FPGA 中实现地址寄存器、数据寄存器,并实现一个简单的状态机来控制对存储芯片的访问。具体步骤如下:

1. 定义输入信号

- (1) 拨码开关决定读写地址或数据。
- (2) 绑定 RAM1 的数据线、地址线。
- (3) 绑定 LED 灯,用来显示从内存中读出的数据。

2. 定义状态机

- (1) 读写状态控制的状态机:控制当前控制器是处于读状态还是写状态。
- (2) 内存读周期、写周期的状态机:根据时钟进行跳转,控制读写的过程。

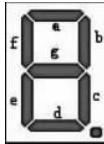
3. 验证过程

- (1) 将编译好的内存控制器烧入实验板。
- (2) 用手拨动拨码开关,预设一个值,每个写周期,作为写入内存的数据;每个读周期,该值作为内存地址,从该内存位置处读出数据。该预设值在同一个读/写周期内不改变。
- (3) 写周期,LED 灯显示当前写入的地址;读周期,LED 灯最终显示从内存中读出的数据。
- (4) 注意观察 LED 上地址和数据的变化,是否符合实验的设计要求。
- (5) 记录实验结果。
- (6) 可以自行设计该实验展示的方式。

注意：教学机上 SW0~SW15 是 ON 端表示 0,相反表示 1;CLK、RST、K1~K4 均是按下为 0,放开为 1。本实验使用的管脚分配如表 5.4 所示。

表 5.4 管脚分配表

端口性质	端口	管脚	说 明	端口性质	端口	管脚	说 明
输入	CLK	U9	单步时钟 CLK	输出	Ram1Addr0	C17	RAM1 地址
	RST	U10	复位键 RST		Ram1Addr1	C18	
	SW0	T7	拨码输入初始地址和数据		Ram1Addr2	D16	
	SW1	R7			Ram1Addr3	D17	
	SW2	U8			Ram1Addr4	E15	
	SW3	V4			Ram1Addr5	E16	
	SW4	V3			Ram1Addr6	F14	
	SW5	V2			Ram1Addr7	F15	
	SW6	U1			Ram1Addr8	F17	
	SW7	R4			Ram1Addr9	F18	
	SW8	R1			Ram1Addr10	G13	
	SW9	N2			Ram1Addr11	G14	
	SW10	N1			Ram1Addr12	G15	
	SW11	M1			Ram1Addr13	G16	
	SW12	K7			Ram1Addr14	H14	
	SW13	K2			Ram1Addr15	H15	
	SW14	J7			Ram1Addr16	H16	
	SW15	J6			Ram1Addr17	H17	
输出	L0	D10	发光二极管显示地址及数据	双向	Ram1Data0	J12	RAM1 数据
	L1	E10			Ram1Data1	J13	
	L2	A11			Ram1Data2	J14	
	L3	B11			Ram1Data3	J15	
	L4	C11			Ram1Data4	J16	
	L5	D11			Ram1Data5	J17	
	L6	E11			Ram1Data6	K12	
	L7	F11			Ram1Data7	K13	
	L8	A12			Ram1Data8	K14	
	L9	E12			Ram1Data9	K15	
	L10	F12			Ram1Data10	L15	
	L11	A13			Ram1Data11	L16	
	L12	B13			Ram1Data12	L17	
	L13	D13			Ram1Data13	L18	
	L14	E13			Ram1Data14	M13	
	L15	A14			Ram1Data15	M14	

端口性质	端口	管脚	说 明	端口性质	端口	管脚	说 明
输出	Ram1OE	M16	RAM1 输出使能	双向	Ram2Data6	R14	RAM2 数据
	Ram1WE	M18	RAM1 写使能		Ram2Data7	R12	
	Ram1EN	M15	RAM1 使能		Ram2Data8	R11	
输出	Ram2Addr0	N14	RAM2 地址		Ram2Data9	R10	
	Ram2Addr1	N15			Ram2Data10	P13	
	Ram2Addr2	N18			Ram2Data11	P12	
	Ram2Addr3	P16			Ram2Data12	P11	
	Ram2Addr4	P17			Ram2Data13	P10	
	Ram2Addr5	P18			Ram2Data14	N12	
	Ram2Addr6	R15			Ram2Data15	N11	
	Ram2Addr7	R16		输出	Ram2OE	R9	RAM2 输出使能
	Ram2Addr8	R18			Ram2WE	M9	RAM2 写使能
	Ram2Addr9	T17			Ram2EN	N10	RAM2 使能
	Ram2Addr10	T18		输出	DYP0A	C7	七段数码管测试 输出 
	Ram2Addr11	U18			DYP0B	D7	
	Ram2Addr12	V15			DYP0C	E7	
	Ram2Addr13	V13			DYP0D	F7	
	Ram2Addr14	V12			DYP0E	A8	
	Ram2Addr15	V9			DYP0F	E8	
	Ram2Addr16	U16			DYP0G	F8	
双向	Ram2Addr17	U15	输出	DYP1A	C9	七段数码管测试 输出	
	Ram2Data0	U13		DYP1B	D9		
	Ram2Data1	T16		DYP1C	E9		
	Ram2Data2	T15		DYP1D	F9		
	Ram2Data3	T14		DYP1E	G9		
	Ram2Data4	T12		DYP1F	A10		
	Ram2Data5	R13		DYP1G	B10		

5.4.6 实验数据

将实验过程中对内存读写的数据记录在表 5.5 中。

表 5.5 实验数据

写 内 存		读 内 存		读写一致性
地址	数据	地址	数据	