调试软件使用方法

通用调试器(Unidebugger)目前最新的版本是 2.0。该软件将调试用到的所有功能集成 在一个可执行的程序里,即 Unidebugger.exe。

1 软件界面

双击 Unidebugger.exe 程序,即启动通用调试器。程序启动界面如下:



图1 启动界面

进入系统以后,显示如图2的界面。

🕅 UniDebugger - [0024018A5M.asm] 工具	💥 💷 🗙
🔮 文件(E) 编辑(E) 视图(Y) 串口(5) 汇编(A) 调试(D) 系统(N) 窗口(W) 帮助(H)	_ I 뢴 ×
新建打开一保存一剪切复制粘贴一打印 连接断开一系统设置 新点 单步扩展 可断运行 傍	計上 中止 复位
地址 十六进制数据 ASCII码 ■ ■ ■ ■ ■ ■ ■ ■ ■ ■ ■ ■ ■ ■ ■ ■ ■ ■	
0000 00 0F F7 5F 37 12 F7 0B7 mov r1,#1	
0010 07 EF B7 51 7E, EF, EP, BF	
0018 FD FF 7F AF E政机体及F	
0030 OE FF FF FF 73 FF BF FBs Ida 8001	
0038 3E 7F 4F DF FE B7 9F BF >.0 mov @r1,a	
0040 AF EB 52 FE FF 4B BF BFRK	
RAM V Micro mov @r0,a	
□□□1-1	
若要获取帮助,请按 F1	

图 2 各个工作区域

2 操作

(1). 检查硬件线路连接情况

PC 机通过一根串口线与实验仪连接,进行通信数据的传送。所以首先要检查串口连接 是否正常。调试器的默认串口连接是 COM1,如果你需要连接到其他串口,可以从菜单操作。

	串口 (S)	汇编(<u>A</u>)
1	连接 @	υ 📔
H	断开 (C	9
	设置 (S	9
ľ		

选择"串口 --> 设置",会出现串口设置对话框,实验仪的串口协议是:波特率 9600bps,数据位 8 位,校验位无,停止位 1 位,数据流控制无。

串	口设置		X
	串行口设置	₽ 	
	端口:	COM1 -	数据流控制————
	波特率:	9600 💌	●无
	数据位:	8 💌	C RTS / CTS
	校验:	•	C XON / XOFF
	停止位:	1 💌	
		确定	取消

图 3 串口设置

(2). 设置实验方式

首先必须在系统菜单中选择实验方式。



或从工具栏选择



然后会出现如下的系统设置对话框:

系统设置		×
┌实验方式 ─────	_ 晶振	总线方式
○ 手动数字逻辑	C 8MHZ	● 8位
○ 自动数字逻辑	4MHZ	○ 全16位
● 8位微程序方式	C 2MHZ	〇 16位,WRL,WRH
○ 16位微程序方式	C 1MHZ	〇 16位,WR,BHE
○ 8位硬接线方式	C 1.024ms	
○ 16位硬接线方式	C 8.196ms	确定 取消
		浏览
└────────────────────────────────────		Saline 1
519程序文件		
		浏览
└────────────────────────────────────		
		浏览

图 4 选择实验方式

对 8 位微程序控制计算机实验,实验方式为 8 位微程序方式,晶振频率可选为 8MHz。 总线方式为 8 位。按下确定后计算机与实验仪进行连接,读取微程序存储器以及 RAM 中的 数据在数据区显示。

对 16 位 RSIC 计算机实验,实验方式为 16 位硬接线方式,晶振频率可选为 8MHz。总 线方式为全 16 位或 16 位,WRL,WRH; 16 位,WR,BHE 等(后两者属于 8 位编址方式)。

另外,如果你是继续前面的实验的话,在这个对话框里还可以设置用户配置文件、微程 序文件、S19 文件等。

确定后可自动下载微程序文件、S19文件、用户配置文件,如图5、6所示。



图 5 程序文件下载提示



图 6 微程序文件下载提示

最后还会读入并显示 RAM 和微程序存储器内容,并进行配置设定。如不设置 这些文件,将直接读入实验仪中 RAM 和微程序存储器中的上电初始化时或原来的 内容。

-

(3) 微程序存储器

在微程序控制计算机实验中,可在 Micro 窗口(见图 3-15)中实验仪的微程序 存储器的内容。首次使用时,可在微程序窗口中按照地址分配直接输入 16 进制微程序代码 (每个地址四位 16 进制数)。

地址	ーナナ	∀进	制数	据	二进制
0000	00	25	70	04	00000000001001010111000000000100 🚍
0001	00	00	00	00	000000000000000000000000000000000000000
0002	00	00	00	00	000000000000000000000000000000000000000
0003	00	00	00	00	00000000000000000000000000000000000
0004	00	00	00	00	000000000000000000000000000000000000000
0005	00	00	00	00	000000000000000000000000000000000000000
0006	00	00	00	00	000000000000000000000000000000000000000
0007	37	24	ΕO	07	00110111001001001110000000000111
0008	00	25	70	04	00000000001001010111000000000100
0009	00	00	00	00	000000000000000000000000000000000000000
A000	00	00	00	00	000000000000000000000000000000000000000
000B	00	00	00	00	000000000000000000000000000000000000000
000C	00	00	00	00	000000000000000000000000000000000000000
000D	00	00	00	00	000000000000000000000000000000000000000
000E	00	00	00	00	000000000000000000000000000000000000000
000F	21	20	ΕO	07	00100001001000001110000000000111
0010	00	25	70	04	00000000001001010111000000000100
0011	00	00	00	00	000000000000000000000000000000000000000
0012	00	00	00	00	000000000000000000000000000000000000000
0013	00	00	00	00	00000000000000000000000000000000000
0014	00	00	00	00	000000000000000000000000000000000000000
0015	00	00	00	00	000000000000000000000000000000000000000
0016	00	00	00	00	00000000000000000000000000000000
•					
🔶 RA	M) M	licr	. [

图 7 微程序存储器窗口

然后从 File 菜单将其保存为一个微程序定义文件, 后缀为.m19(见图 7 和 8)。

新建 打开 保存 另存为(<u>A</u>) 关闭(<u>C</u>)	Ctrl+N(N) Ctrl+O(O) Ctrl+S(S)
新建项目 打开项目 保存项目 关闭项目	
下载519程序文件 下载微程序文件 下载测试向量文件 保存程序文件	
保存微程序文件	
打印 打印预览(⊻) 打印设置(<u>R</u>)	Ctrl+P(P)
<u>1</u> F:\mda\\0024018ASM.asm <u>2</u> asmall.asm <u>3</u> F:\mda\\PRINTA.LST	
退出(X)	

图 8 微程序文件的下载及保存之一

另存为					<u>? ×</u>
保存在 (<u>t</u>):	🔁 微程序		•	+ 🗈 💣 🎟+	
<u>③</u> 历史	atest.m19				
東面 泉直					
我的电脑					
网上邻居	, 文件名 (M): 保存类型 (T):	test.m19 S19 Files (*.m19)		v	保存 (<u>S</u>) 取消

图 9 微程序文件的下载及保存之二

Save Data	x
起始地址(16进制): 0000	
终止地址(16进制): 0100	
保存取消	

图 10 微程序文件的下载及保存之三

在保存时,应先输入起始地址(一般为0)和结束地址(微程序的最后一条微指令的地址)(见图 3-18)。以后再次上机时,可用 File 菜单读出该微程序并下载 至实验仪中。

(4) RAM

存储器(RAM)窗口(见图 3-19)显示实验计算机 RAM 的当前内容。RAM 的内容可在 RAM 窗口中直接输入或修改。

. . .

Lik J.I			- 1960 -		-	-	-	_	1 gazzIJ		_
地址	+7	て进る	制数	焟					ASCIINA		
0000	38	00	39	01	ЗA	08	ЗВ	02	8.9.:.;.	-	
0008	60	00	08	40	80	01	19	60	`0`		
0010	00	OF	40	80	01	18	00	23	@#		
0018	08	01	23	09	2A	50	00	08	#.*P		
0020	38	ΟA	30	10	18	38	09	30	8.08.0		
0028	00	18	38	08	18	38	00	39	88.9		
0030	08	ЗA	02	ЗB	07	ЗC	09	ЗD	.:.;.<.=		
0038	01	10	0E	2E	06	В9	ЗE	ΟA	>.		
0040	2E	50	00	4E	19	07	2F	30	.P.N/O		
0048	01	BC	10	58	00	50	26	19	X.P«.		
0050	00	22	08	2B	50	00	39	58	.".+P.9X		
0058	00	5A	05	ЗE	01	2E	50	00	.Z.>P.		
0060	6E	ЗD	00	38	01	39	09	ЗB	n=.8.9.;		
0068	08	ЗC	0A	58	00	39	78	01	.<.X.9x.		
0070	6D	78	00	EЗ	60	00	74	3C	mx`.t<		
0078	08	OD	ЗF	01	25	0E	04	2F	?.*/		
0080	50	00	88	0C	06	58	00	7C	PX.		
0088	06	80	38	00	39	0B	ЗA	08	8.9.:.		
0090	ЗB	01	10	19	00	23	08	10	;#		
0098	78	00	77	0E	2E	06	В9	19	x.w		
00A0	38	02	39	00	ЗA	08	3B	01	8.9.:.;.		
00A8	10	19	00	23	08	10	78	00	#x.		
00B0	77	0E	2E	06	В9	19	38	04	w8.		-1
I										Þ	
	w 🔽) N	licr	•							

图 11 存储器窗口

但一般可先用通用汇编软件输入源程序,并汇编成目标码文件(*.S19)。然后用 File 菜单下载 S19 程序文件装入 RAM 中(见图 12)。

新建 打开 保存 另存为(<u>A</u>) 关闭(<u>C</u>)	Ctrl+N(N) Ctrl+O(O) Ctrl+S(S)
新建项目 打开项目 保存项目 关闭项目	
下载519程序文件	
下载微程序文件 下载测试向量文件	
保存程序文件 保存微程序文件	
打印 打印预览(⊻) 打印设置(<u>R</u>)	Ctrl+P(P)
1 F:\mda\\0024018ASM.asm 2 asmall.asm 3 F:\mda\\PRINTA.LST	
退出(X)	

图 12 S19 文件的下载

(5) 用户配置文件

本实验仪的实验 FPGA 的输入、输出引脚共 104 个,包括:

- 地址总线 AB15-0
- 数据总线 DB15-0
- 读写控制信号 MWR、MRD、IOW、IOR
- 备用控制信号 CTR1-4
- 可设置内部输出信号 CI31-0 (MUX = 000 时)
- 可设置输入/输出信号 CO31-0。

另外有 224 路扩展观察信号 EX1_31-0, ---, EX7_31-0。

它们可在单步和扩展单步时被采样并显示在调试数据观察窗口中,也可在全速运行(包括断点和连续运行)时由硬件电路采样并存储在跟踪存储器中(仅 I/O 引脚),最大存储深度为 255 项(每项为 104 位)。

×	No.	BIT	ADDR	DATA	CTRL	IOR	IOW	CRD	CWR	
	0									
	1									
	2									
	3									
	4									
	5									
	6									
	7									
	8									
		1	1							

图 13 数据观察窗口

在系统菜单中选择输出装置。选择菜单项"系统 → 输出设置 → 新设置":

帮助(H)
新设置(1)
装载设置(L)

即可弹出设置对话框,如图 14 所示:

输出显示	设置									X
● 数据区 ● CI	[0 со	O EX1	O EX2	O EX3	O EX4	O EX5	O EX6	O EX7	<添加>
31 30	29 2	8 27 26	25 24 23 22	2 21 20 19 18	17 16 15 1	4 13 12 11	10987	654	3210	< <mark>删除></mark>
										编辑
显示格	大						显示标题	2		保存
○ 2 j	悲制	C	8进制	○10进制	. 01	16进制			~	退出

图 14 显示设置

在这个对话框里,可设置 CI31-0、CO31-0 和 EX1-7 的 31-0 对应的数据格式 和名称。例如对 8 位微程序控制计算机实验,按用户 VHDL 程序中 CI 的选择,可 设置 CI31-24 为 ACC, CI23-16 为 IR, EX1 的 31-16 为 PC, EX2 的 31-16 为 ADR, EX3 的 31-24 为 R0,、、、、。还可设置 CO31-0 为 MIR (为清楚起见,也可按每个 微操作的名称来设置,例如 SEL 为 ALU 功能选择,接于 CO31-29,可设置 CO31-29 为 SEL); CI9-0 为 MPC。以后在单步、连续调试时可按名称和数据格式显示这些 信号,而在设置断点、屏蔽时也可使用这些名称来设置。这样可大大方便调试的进 行。在设置时可选择显示格式,一般情况应选择 16 进制。

注意, EX1-EX7 等为扩展观察信号。在执行扩展单步时,将按用户 FPGA 内部电路,由 MUX 选择从 CI31-0 输出其他 32 X 7 位数据。MUX = 000 为 CI31-0; MUX = 001 为 EX1_31-0; MUX = 010 为 EX2_ 31-0; 、、、、、; MUX = 111 为 EX7_31-0。所以在执行扩展单步时,总共可采样、显示 104 + 32 X 7 位内部数据。 但必须注意,扩展部分的数据必须在内部编程设计一个 8 选一多路开关(32 位) 才有效,并且它们不能用于设置断点,也不能在全速执行时存放于跟踪存储器中。 编辑完成后,按"保存"可将该设置存放于文件中(*.CFG),以后可通过"编辑"读出该设置文件,也可对它进行编辑修改。

(6). 断点和断点屏蔽

实验仪有一个 104 位的断点寄存器和 104 位的屏蔽寄存器。这 104 位对应于 FPGA 的 104 个输入输出脚,可用于设置断点。

本实验仪有一个 104 位的断点寄存器。断点可按 ADDR (地址总线)、DATA (数据总线)、CTRL (四个备用控制位)、MRD、MWR、IOR、IOW 和用户设置的 CI31-0 (按用户定义的标号和格式) 来分别设置。这些设置为"与"的关系,即设置的断点应 全部符合,才产生断点信号。可通过设置'X'来屏蔽不需要的断点条件。一般情况应 屏蔽大部分的断点,仅使用一个或少数几个断点条件。为此,在设置断点时,除了写入 数据的位置 (标号),其它未写入的位置均自动置为屏蔽状态。如可设置断点为访问某 个地址 (16 位),这时应屏蔽其它位 (88 位);也可设置在某个内部寄存器 (接于 CI 或 CO 上)为某个值时产生断点等。

按工具栏按钮可开启断点设置窗口:

Ð	3	<u>E8</u>	Ħ	1	•7
连接	断开	系统	设置		6 A

弹出的断点设置窗口如图 15 所示:

Ķ	设置断点					
[标记	备注	断点值	^	取消断卢	
	ADDR	16进制数,格式(FFFF)				
	DATA	16进制数,格式(FFFF)			设塑版片	
	CTRL	2进制数,格式(1111)			以且 时是	
	IOR	2进制数,格式(1)			PR 41 20 PR	
	IOW	2进制数,格式(1)		~	退出攻直	

图 15 断点设置

断点列表包括三项:

- 标记:设置断点的数据名称(ADDR、DATA、CTRL、MRD、MWR、IOR、 IOW 等为固定的名称,其他的由用户自定义)
- 备注:规定了断点的数据格式
- 断点值:当数据域的值为此值时产生断点信号

三个按钮:

- 取消断点:清除所有已设断点
- 设置断点:进行断点设置的合法性检查,并设置断点
- 退出设置

(7) 调试功能和使用方法

以下功能可由调试菜单、图标或功能键执行,对应每个调试功能,均设置了相应的 快捷键。

调试 (D)	系统 (N)
单步	F10
扩展单	步 F7
运行	F8
断点运	行 175
停止	F6
设断点	F9
终止运	行ESC
复位	F4

另外,还有相应的工具栏快捷按钮,见图 16。



图 16 运行方式

单步和扩展单步运行

每执行一次单步或扩展单步运行,对微程序控制的计算机实验先发一个 CLK(和 CLKG),如这时从用户 FPGA 输出的 MCLK 为 1,则再发一个 CLK。所 以如按前面介绍的 MCLK 产生电路,这时每次单步发两个 CLK。对硬接线控制计 算机实验,一般 MCLK 等于 CLK,故每步发一个 CLK。

在 CLK 和 MCLK 均为高电平时,读出 FPGA 外部的 104 位的数据。如为扩展 单步,则还将依次置 MUX 为 001-111,读出扩展的 32 X 7 位数据。这些数据按用 户设置的格式显示在屏幕下半部的窗口中。如为单拍运行,则扩展的数据显示为"一 一"。显示的数据可用点击右键清除。

在单步或扩展单步运行后,实验计算机处于运行状态(RUN = 1)。可发停止 命令停止运行,也可发复位命令复位实验计算机。

单步运行与扩展单步运行的区别是:扩展单步运行返回的调试信息包括了扩展数据部分,也就是用户设置中的 EX1~EX7 的部分。

断点运行

断点运行启动实验计算机以全速(按用户设置的 CLK 频率)运行,在遇到 断点后停止运行。在未遇断点前,保持运行状态,这时可通过发中止命令停止运行, 也可发复位命令复位实验计算机。

在启动运行后,立即开始跟踪存储器的采样存储。在跟踪存储器满以后停止采样。 并提示用户可读入显示跟踪存储器的内容。跟踪存储器的容量为 104 位 X 255。除 了总线信息外,它按用户设置的内容进行显示(仅 CI31-0、CO31-0)。

● 连续运行

连续运行启动实验计算机以全速(按用户设置的CLK频率)运行,不停止

运行。但可发中止命令停止运行,也可发复位命令复位实验计算机。 在运行后,遇到断点条件符合后,启动跟踪存储器的采样存储,在跟踪存储器满以 后停止采样。并提示用户可读入显示跟踪存储器的内容。

● 停止运行

停止程序的运行。一般用于单步运行时清零 RUN 信号。

● 中止运行

中止程序的运行。一般用于全速运行时停止程序的运行。

复位

在初次运行前或修改 RAM 后,一般应先发复位命令。它向用户的 FPGA 芯 片发一个低电平有效的 RESET 脉冲。该脉冲应用于复位实验 CPU 的 PC 和 MPC。

复位命令也可用于停止实验 CPU 的运行,这时它清零 RUN 信号,同时也发出 RESET 脉冲。

(8) 文本编辑显示功能

本软件有一个编辑窗口,可使用"打开"图标打开某个文件,并显示在右上部, 见图 17。一般可打开正在调试程序的列表文件(*.lst),它包括源程序、地址、目 标码、行号等信息。这些可大大方便对硬件和程序调试。



图 17 文本编辑区