

ZJUQS-II SWORD 版模拟器使用说明

注：以下各个外设接口均为默认值。可在配置 - 自定义总线接口处进行重新配置。

一、 内存模拟

一个大数组模拟内存，加载用户程序就是将用户机器程序导入数据。内存空间 32K×32 位 WORD。地址空间为 00000000H~0000FFFFH 字地址，在此地址范围内 SW 和 LW 指令读取或写入此数组(内存)。

指令也存储于此内存，初始时 PC=00000000H。

二、 寄存器模拟

数组模拟 32 个寄存器实时显示。用户程序读/写访问寄存器时，由模拟器仿真实现(读出或写入)并实时显示。

三、 指令模拟

读取内存代码区机器指令，并模拟执行。支持 MIPS-32 指令。

指令模拟仅宏观模拟指令执行结果，不考虑延时槽。复位时 PC=00000000H。所有指令模拟执行后更新目的操作数同时，就更新外设和寄存器组显示。

四、 VGA 显示模拟

1. 显示接口状态寄存器

8×8 和 16×16 两种字符点阵所需 VRAM 容量分别是 1200 和 4800 单元，占用存储资源不多，VRAM 文本显示容量选用 4800(0~12BF)共享，区别在于定位计算略有不同，可以定义状态向量位切换。VRAM 图形容量 4B000(2000~4CFFF)。VRAM 起始地址由 VRAMSR 决定，缺省值为 000C0000H。

为方便后继设计扩展及作为计算机系统的标准接口，此处定义显示接口三个状态寄存器。

a) 模式控制寄存器



M: 模式位，置 0 为文本模式，置 1 为图形模式

C: 硬件光标使能位，置 1 时开启硬件光标

Font: 字库模式，1xx 保留。

Font = 000: 英文 8*8(标准/启动)，Font = 001: 英文 8*16

Font = 010: 中文 16*16，Font = 011: 中文 32*32

R: 分辨率选择，0000 表示关闭显示(VGA)输出，除下列外其余保留。

模式	分辨率@刷新率	长宽比	标准类型
1	640 * 480 @ 60Hz	4:3	工业标准(启动)

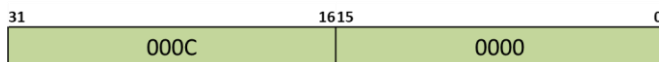
.....

模式寄存器缺省状态为 MODE=1100_0...0_000_0001。

b) 文本 VRAM 字符显示格式寄存器（也是 VRAM 存储内容格式）



c) VRAM 起始地址寄存器



VRAM 起始地址，文本和图形模式共用，缺省为 000C_0000H。实际就为 VRAMR+XX。

功能：汇编程序写入 VRAM，VGA 显示模拟读出 VRAM 在 VGA 窗口显示。

VRAM 地址：FFC00000H~FFC0FFFFH

2. 文本显示模拟

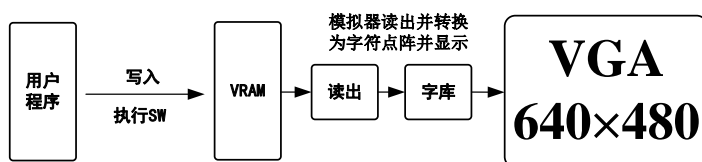
8*8 西文字库：一行 80 个字符，60 行(缺省状态，首先实现)，对应 VRAM 地址 VRAMSR+00000000H~VRAMSR+000012BFH

16*16 汉字字库，一行 40 个字符，30 行，对应 VRAM 地址 VRAMSR+00000000H~VRAMSR+000004AFH。

缺省时：VRAMSR=000C0000H

VRAM 用数组实现：存储字符编码，统一为 16 位。西文时低 8 位为 ASCII，高 8 位为颜色；汉字时低 13 位为汉字(压缩区位码)编码，高 3 位为颜色。参考上述 b)。

用户程序写入(ASCII 或汉字编码)，模拟器读出 ACSCII 或汉字编码并转换为对应字符在屏幕窗口显示。字符转换显示需要用对应字库。



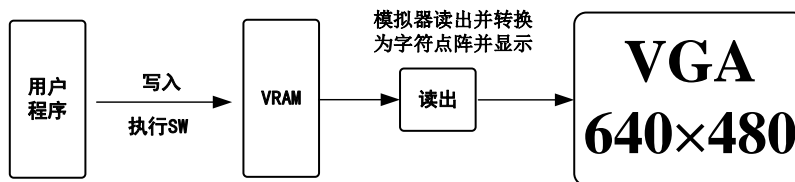
用户写入即为 SW 指令模拟，当地址范围在：

VRAMSR+00000000H~VRAMSR+000012BFH

3. 图形模式显示模拟

VRAM 存储 640*480 点像素，每个像素 16 位，低 12 位为像素点 RRRR_GGGG_BBBB，高 4 位为属性，具体定义待定。VRAM 地址 VRAMSR+00002000H~VRAMSR+0004CFFFH。

用户程序直接写入像素显示值(16 位)，模拟器读出并直接在屏幕窗口显示。



用户写入即为 SW 指令模拟，当地址范围在：

VRAMSR+00002000H~ VRAMSR+0004CFFFH

五、 七段显示模拟

1. 七段码地址：FFFFFE00/E0000000H

2. 模拟：当 SW 指令地址为 FFFFFE00/E0000000H 时。

若滑动开关 SW[0]=1，SW 输出内容在七段码上以 16 进制方式显示；

若滑动开关 SW[0]=0，偶数地址，SW 输出内容在低四位七段码上以图形点阵形式显示；奇数地址，SW 输出内容在高四位七段码上以图形点阵形式显示。

六、 GPIO 输出(写)模拟(LED 显示)

1. GPIO 地址：FFFFFFF00/F0000000H

2. 模拟：当 SW 指令地址为 FFFFFFF00/F0000000H 时。SW 指令输出内容对应如下：

a. SW 指令输出 D_{17~2} 16 位输出对应 LED 显示(LED_{15~0})。

b. D_{1~0} 二位控制硬件计数器,对应计数器 4 个通道 counter_ch_{0~3}。参见硬件计数器模拟

c. D_{31~18} 14 位输出对应硬件光标。其中 Cursor_x[6:0]=D_{24~18}，Cursor_y[6:0]=

D_{31~25}。

此

七、 GPIO 输入(读)模拟(滑动开关读取)

1. 鼠标或键盘可改变滑动开关状态，向上为开(1)，向下为关(0)。

2. 滑动开关地址：FFFFFFF00/F0000000H

2. 模拟：当 LW 地址为 FFFFFFF00/F0000000H 时，LW 指令模拟执行结果为将下列信息写入目的寄存器：

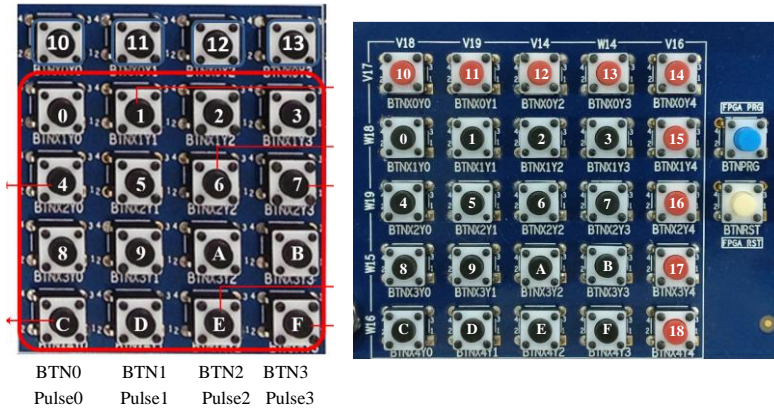
{counter0_out,counter1_out,counter2_out,8'h000,BTN[4:0],SW[15:0]}

其中：SW[15:0]为滑动开关当前状态，BTN[4:0]为阵列键盘列状态值(SWORDV2 是 4 位，SWORDV4 是 5 位，一律按 5 位输入)。counter0_out,counter1_out,counter2_out 为硬件计数器 0，1，2 溢出位。

BTN 定义参见阵列键盘模拟。

八、 阵列键盘模拟

1. 阵列键盘编码



(a) SWORD-V2

(b) SWORD-V4

用鼠标点击模拟按键。按下键盘时输出对应的编码 KCODE [4:0]，同时按键有效信号 KRDY=1。

输出键盘编码同时输出阵列键盘当前列键对应电平信号 BTN0、BTN1、BTN2 或 BTN3 为“1”，释放后即为“0”。信号集合组成为 32 位数：

$D_{31}=KRDY$ 、 $D_{4-0}=KCODE [4:0]$ ，其余暂定为 0。

阵列键盘访问地址：FFFFFC00/C0000000H

2. 模拟

当 LW 地址为 FFFFFC00/C0000000H 时，LW 指令模拟执行结果为将下列信息写入目的寄存器，同时 KRDY 清零：

{ KRDY, 21'h000000, BTN[4:0], KCODE [4:0]}

关于 KRDY 说明：当键按下时 KRDY=1，读取后 KRDY=0，此时只有按键释放后才能下次按键。也就是不支持连续按键。

若没有读取键盘则 KRDY 一直保持 1 不变，无论按键是否释放，但 KCODE [4:0]是最近按下的键编码。

九、 硬件计数器模拟

1. 计数器说明

硬件计数器为 32 位计数器，兼容 8253，共三个计数通道，其中溢出信号分别为 counter0_out、 counter1_out,和 counter2_out。暂模拟通道 0，工作方法 0，参考 8253。

计数器对应地址为 FFFFFFF0/F0000004。三个计数器和控制豁口由 GPIO 输出最底 2 位识别，其对应如下：

counter_ch=00，对应计数器 0。读写对应计数器 0

counter_ch=01，对应计数器 1。读写对应计数器 1

counter_ch=10，对应计数器 2。读写对应计数器 2

counter_ch=11，对应控制寄存器。读写对应计数器控制器。

控制寄存器格式 24 位，每个通道 8 位(可参考 8253)：

Counter0_Ctrl [7:0]= D_{7-0} = XX M2 M1 M0 X

Counter1_Ctrl [7:0]= D_{15-8} = XX M2 M1 M0 X

Counter2_Ctrl [7:0]= D_{23-16} = XX M2 M1 M0 X

2. 计数器模拟

暂时模拟通道 0 工作方式 0，也就是一个独立计数器，以上参数可以暂时忽略。

这样只要 SW 指令地址= FFFFFFF04 /F0000004，就重新加载计数值(无论当前计数是否结束)，重新开始再次计数开始，直到减到 0，输出 counter0_out=1，并一起保持，直到下一次 SW 指令地址= FFFFFFF04 /F0000004。

LW 指令地址= FFFFFFF04 /F0000004，指令模拟结果为当前计数器值写入 LW 指令的目的寄存器中。

十、 PS2 键盘模拟

1. 说明

当 VGA 窗口激活时，PS2 输入指向模拟器，若 PS2 键盘按下模拟发送键盘扫描码：“通码+F0+通码”或 E0+通码+E0+F0+通码(参考 PS2 键盘协议)。PS2 键盘地址=FFFFD000H。

2. 模拟：当 LW 指令地址= FFFFD000H 时，模拟读取 PS2 扫描码的当前 8 位，写入 LW 指令目的寄存器。格式如下：

{ps2_ready,23'h0, key};

其中 ps2_ready 是扫描码有效，读取即清零，key 为当前描述码中的 8 位(通码、F0(断码)或 E0(扩展码))。

每次按键 PS2 模拟分三次或 5 次发送完整的扫描码，每次发送 8 位，设定合理时间间隔。

注：接口模拟相当于读写只有一个单元的内存，也就是寄存器。