

# 嵌入式处理器与芯片设计基础实验报告

## wujian 平台介绍和 FPGA 调试

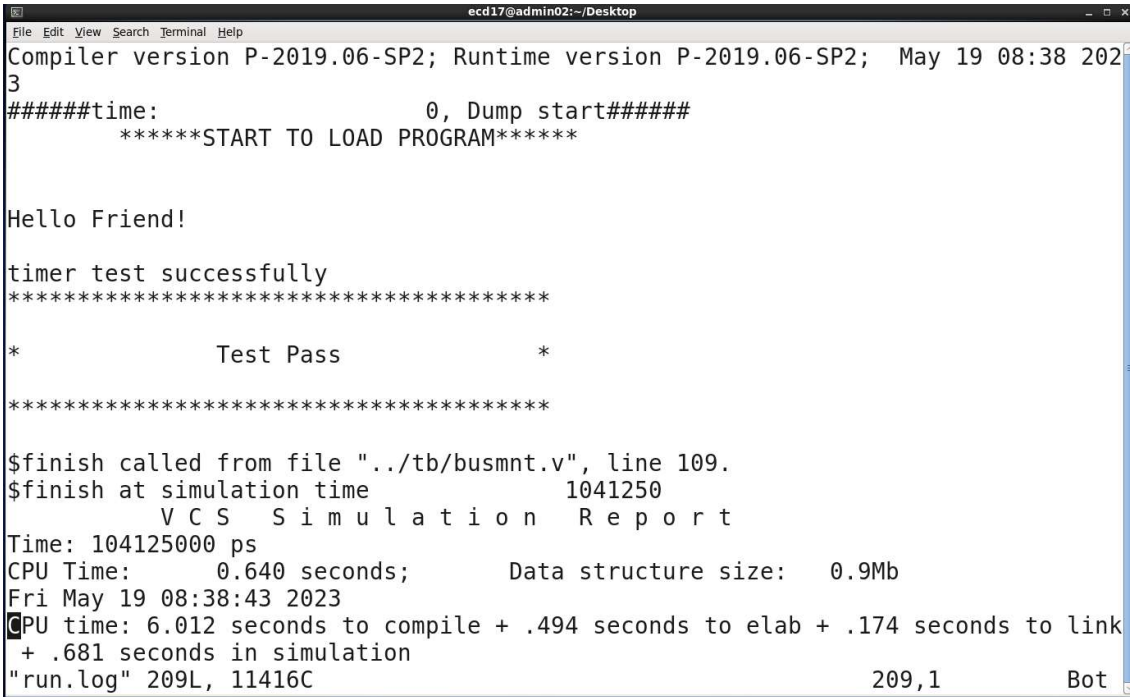
陈之逸 21307130003

### 一、实验目的

了解 wujian100 SoC 的组成架构，掌握 wujian 平台的前端仿真和 fpga 调试。

### 二、实验步骤（包括实验结果，数据记录、截图等）

1. 依照（二、wujian100 SoC 的前端仿真）的步骤完成 case 目录下的示例的 timer\_test.c 仿真，完善 case/mytest/mytest.c 实现任一简单算法并对 mytest.c 仿真，对以上输出结果进行截图。
- 1) 获取 wujian100 项目文件，利用 cp 语句将其复制到本地的路径中。
- 2) 利用 vcs 对示例 case 进行仿真，得到输出“Hello Friend!”以及 test pass 的结果。



```
ecd17@admin02:~/Desktop
File Edit View Search Terminal Help
Compiler version P-2019.06-SP2; Runtime version P-2019.06-SP2; May 19 08:38 2023
#####time: 0, Dump start#####
*****START TO LOAD PROGRAM*****

Hello Friend!

timer test successfully
*****

* Test Pass *

*****

$finish called from file "../tb/busmnt.v", line 109.
$finish at simulation time 1041250
V C S S i m u l a t i o n R e p o r t
Time: 104125000 ps
CPU Time: 0.640 seconds; Data structure size: 0.9Mb
Fri May 19 08:38:43 2023
CPU time: 6.012 seconds to compile + .494 seconds to elab + .174 seconds to link
+ .681 seconds in simulation
"run.log" 209L, 11416C 209,1 Bot
```

- 3) mytest 使用的是自行编写的快速排序算法 C 程序代码，仿真成功。

```

E: > Study > SoCDesign > 实验课 > 实验十一 > 2023_05_17_上板实验 > C QuickSort.c > quicksort(int, int)
1  #include <stdio.h>
2  int arr[] = {7, 5, 9, 8, 1, 2, 8, 6};
3  int getpivot(int left, int right)
4  {
5      // 每个循环的条件都是left<right
6      int pivot = arr[left]; // 把最左侧的值赋给支点
7      while (left < right)
8      {
9          while (arr[right] >= pivot && left < right)
10             right--;
11         arr[left] = arr[right];
12         while (arr[left] < pivot && left < right)
13             left++;
14         arr[right] = arr[left];
15     }
16     arr[right] = pivot; // 这里left==right 所以都可以 下同
17     return left;
18 }
19 void quicksort(int left, int right)
20 {
21     if (left < right)
22     {
23         int pivot = getpivot(left, right); // 找中间位置左右分割
24         quicksort(left, pivot - 1); // 处理左边, 这里是一个递归的过程
25         quicksort(pivot + 1, right); // 处理右边, 这里是一个递归的过程
26     }
27 }
28 int main(void)
29 {
30     int i;
31     quicksort(0, 7);
32     for (i = 0; i < 8; i++)
33         printf("%d", arr[i]);
34     return 0;
35 }

```

```

run.log
~/czy21307130003/wujian100_Project/wujian100_open/workdir

All of 71 modules done
rm -f _csrc*.so pre_vcsobj *.so share_vcsobj *.so
if [ -x ../simv ]; then chmod a-x ../simv; fi
g++ -o ../simv -m32 -rdynamic -WL,-rpath='$ORIGIN'/simv.daidir -WL,-
rpath=../simv.daidir -WL,-rpath=/apps/EDAs/synopsys/2019/vcs/P-2019.06-SP2/linux/lib -
L/apps/EDAs/synopsys/2019/vcs/P-2019.06-SP2/linux/lib -WL,-rpath-link=/ obj
amQw.d.o 215640 archive.1.so SIM.l.o rmapats.mop.o rmapats.o rmar.o
rmar.nd.o rmar.llvm.0.1.o rmar.llvm.0.0.o -lvrsim -lerrorinf -lsnpsmalloc
-lvcs -lvcsnew -lsimprofile -luclina -lative /apps/EDAs/synopsys/2019/vcs/P-2019.06-
SP2/linux/lib/vcs.tls.o -WL,-whole-archive -lvcsucl1 -WL,-no-whole-
archive /apps/EDAs/synopsys/2019/vcs/P-2019.06-SP2/linux/lib/
vcs_save_restore_new.o /apps/EDAs/synopsys/2019/vcs/P-2019.06-SP2/linux/lib/ctype-
stubs.32.a -ldl -lc -lm -lpthread -ldl
../simv up to date
Notice: timing checks disabled with +notimingcheck at compile-time
Chronologic VCS simulator copyright 1991-2019
Contains Synopsys proprietary information.
Compiler version P-2019.06-SP2; Runtime version P-2019.06-SP2; May 19 08:47 2023
#####time: 0, Dump start#####
*****START TO LOAD PROGRAM*****

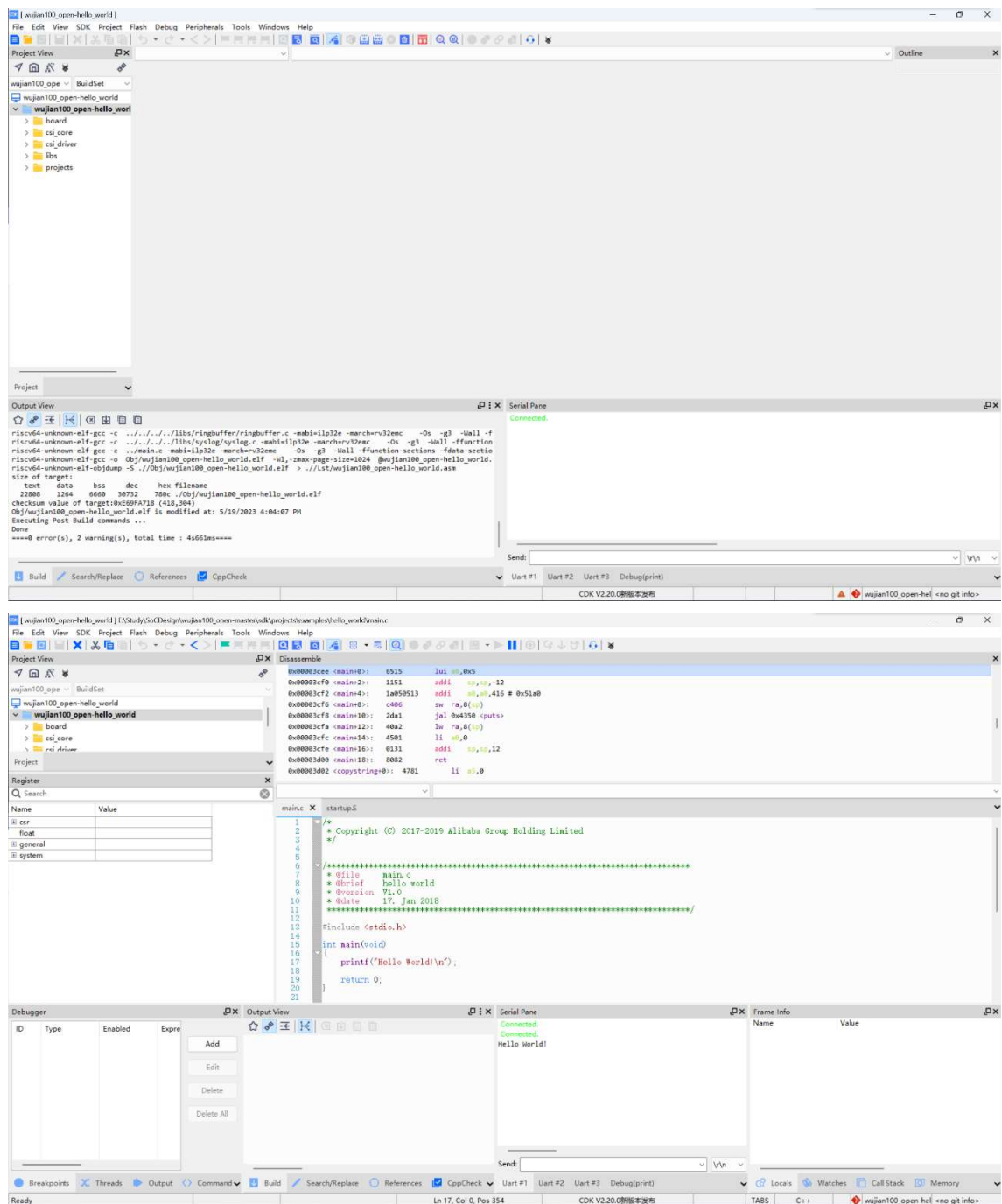
Test Your Program Here!
12567889*****
* Test Pass *
*****

$finish called from file "../tb/busmnt.v", line 109.
$finish at simulation time 1896750
VCS Simulation Report
Time: 189675000 ps
CPU Time: 0.780 seconds; Data structure size: 0.9Mb
Fri May 19 08:47:26 2023
CPU time: 6.021 seconds to compile + .491 seconds to elab + .176 seconds to link + .
814 seconds in simulation

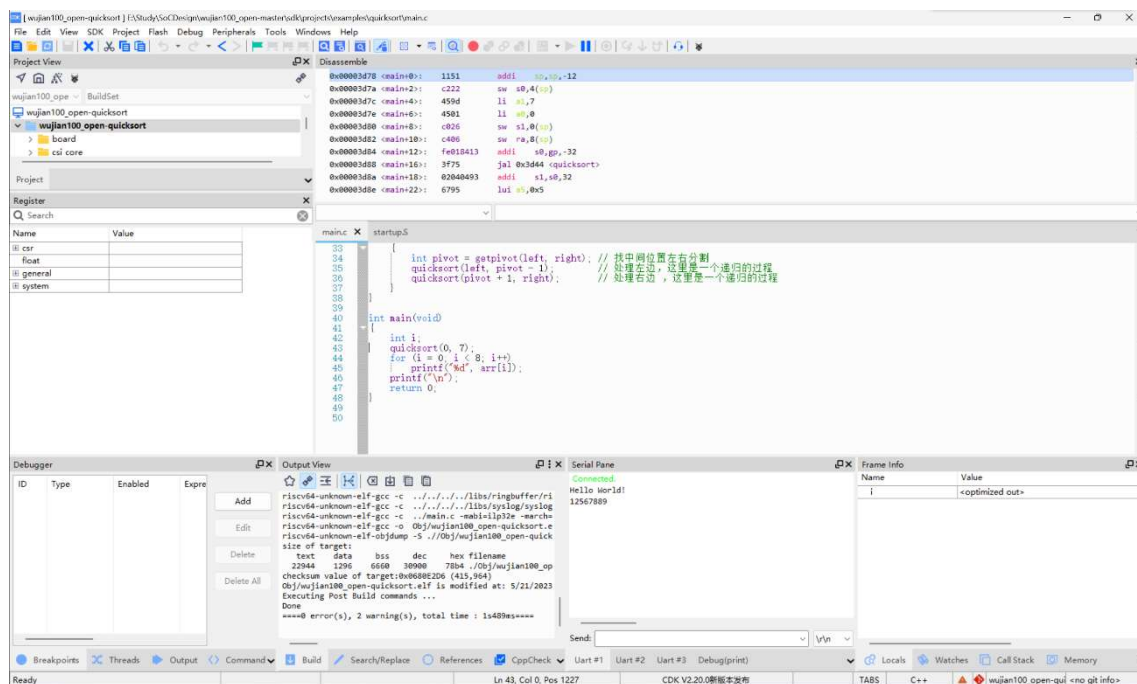
```

2. 依照(三、wujian100 SoC 的 FPGA 原型测试)的步骤完成示例的 hello\_world 项目测试, sdk\projects\benchmark 目录下的两个示例 cdk 项目测试, 对测试结果

进行截图。



3. 自行编写一个 C 语言的简单算法程序，在板子上进行测试，步骤参考（三、wujian100 SoC 的 FPGA 原型测试-4），对测试结果进行截图。



阅读 projects 下其他程序，比如 core\vic 实现定时器驱动程序的主要功能，应得到得到相应的结果，“example run successfully”；driver\dmac 实现 DMAC 驱动程序的主要功能，应得到相应的结果，“test damc successfully”；driver\gpio 实现 GPIO 驱动程序的主要功能，应得到相应的结果，“test gpio successfully”；driver\pwm 实现 PWM 驱动器的主要功能，应得到相应的结果，“pwm\_signal\_test OK”；driver\rtc 实现 RTC 驱动程序的主要功能，应得到相应的结果，“test rtc successfully”。由于耗

时会很长，没有进行仿真验证。

#### 4. 自行编写代码的结果

对快速排序算法进行了仿真，得到输出“12567889”以及 test pass 的结果。

### 四、实验收获、存在问题、改进措施或建议等

#### 1. 存在问题

- 1) 在下载相应的软件以及资源包的时候，需要注意路径中不能有中文字符，否则可能会出错。
- 2) 仿真前应为 vivado 的解压预留出足够的磁盘空间，否则会因内存不足而无法安装。

#### 2. 实验收获

通过本次实验，首先了解了 wujian100 SoC 的组成结构，接触 CDK 并基本掌握了 wujian 平台的前端仿真的基本流程，通过对提供给的例子以及自己的 c 程序的仿真测试，完成了 c 语言的仿真实现。